



UNIVERSITÀ DEGLI STUDI DI MILANO
FACOLTÀ DI SCIENZE E TECNOLOGIE

Master degree in Physics

**Development of an open-source calibration framework for
superconducting qubits**

Supervisor:

Prof. Dr. Stefano Carrazza

Co-supervisor:

Dr. Alessandro Candido

Co-supervisor:

Dr. Andrea Pasquale

Co-supervisor:

Dott. Edoardo Pedicillo

Elisa Stabilini
Matricola n° 28326A
A.A. 2024/2025

Contents

1 Superconducting qubits	1
1.1 Introduction	1
1.2 Transmon qubits	2
1.2.1 Josephson Junctions	3
1.2.2 CPB qubit	4
1.2.3 Transmon qubit	5
1.2.4 Flux-tunable transmon	6
1.3 Qubit readout	7
1.4 Qubit control	9
1.5 Qubit state degradation	10
1.5.1 Qubit decoherence	10
1.5.2 Noise models	11
2 Qubit calibration	13
2.1 Experimental setup	13
2.2 Single qubit calibration experiments	15
2.2.1 Resonator characterization	15
2.2.2 Qubit characterization	16
2.2.3 T1 & T2 measurement	17
2.2.4 Single shot classification	18
2.2.5 Gate calibration	18
2.2.6 Fine calibration	18
2.2.7 Standard Randomized Benchmarking	18
2.2.8 DRAG experiment	18
3 RB fidelity optimization	19
3.1 Randomized Benchmarking	19
3.2 Scipy optimization methods	20
3.2.1 Algorithm description	20
3.2.2 Results	21
3.3 CMA-ES	21
3.3.1 Algorithm description	21
3.3.2 Results	22
3.4 Optuna	22
3.4.1 Algorithm description	22
3.4.2 Results	23
3.5 RB optimization conclusions	23
4 Pulses analysis and tuning	25
4.1 RX90 calibration	25
4.2 Flux pulse correction	25
4.2.1 Cryoscope	25
4.2.2 Corrected pulse	26
4.2.3 Filter determination	26
4.2.4 Dimostrazione del conto	28
5 Conclusions	29

Summary

Chapter 1

Superconducting qubits

The electronics that modern computers rely on contain components that operate based on quantum mechanics; however, their computational processes are still governed by classical laws. For this reason, they are referred to as "classical computers."

Quantum computing emerged from Richard Feynman's idea that simulating quantum systems efficiently requires quantum mechanical resources [1]. Classical computers struggle to model complex quantum interactions due to the exponential growth of computational requirements with system size, making exact simulations infeasible beyond small systems [2]. Quantum computers, taking advantage of quantum mechanics phenomena like superposition and entanglement, offer a natural framework for such simulations and have been demonstrated to provide exponential speedups for certain quantum systems [3].

Beyond quantum simulation, current theoretical advancements suggest that quantum algorithms can outperform classical counterparts in solving specific problems [4].

1.1 Introduction

The physical realization of quantum computing necessitates the development of a system capable of functioning as quantum bits (qubits).

Similar to classical logic, where the bits 0 and 1 are associated with two physical levels, typically represented by high and low voltage states, a qubit can, to a first approximation, be considered as a two-level physical system.

Mathematically, this system is described within a two-dimensional complex Hilbert space, where the basis states $|0\rangle$ and $|1\rangle$ correspond to two orthonormal vectors. Any general state of the qubit can be expressed as a superposition of these basis states:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \rightarrow \begin{pmatrix} \alpha \\ \beta \end{pmatrix}, \quad (1.1)$$

where $\alpha, \beta \in \mathbb{C}$. If the normalization condition $|\alpha|^2 + |\beta|^2 = 1$ holds, the state $|\psi\rangle$ represents a qubit. The basis $\{|0\rangle, |1\rangle\}$ is called computational basis and the information is stored in the complex numbers α and β .

A possible geometric representation of qubit states is given by the Bloch sphere, which offers a visualization of two level quantum systems as vectors on a unit sphere. A qubit state is depicted as a vector originating from the center of the sphere, with the computational basis states $|0\rangle$ and $|1\rangle$ positioned at the north and south poles, respectively. The axis connecting these states defines the z -axis. The transverse x - and y - axes correspond to the equal superposition states $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$ and $|\pm i\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm i|1\rangle)$, respectively.

A vector of unit length on the Bloch sphere is characterized by the polar angle θ , with $0 \leq \theta \leq \pi$ and the azimuthal angle φ , with $0 \leq \varphi \leq 2\pi$, each unit vector represent a possible pure state of the qubit.

The qubit states $|0\rangle$ and $|1\rangle$ can also be associated with energy eigenstates of a physical system, where $|0\rangle$ represents the ground state with energy E_0 and $|1\rangle$ represents the excited state with energy

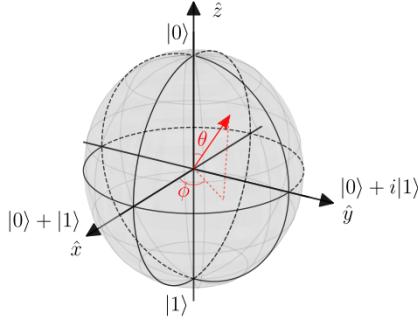


Figure 1.1: Example of qubit state representation on the Bloch sphere

Source: Metrology of Quantum Control and Measurement in Superconducting Qubits [5]

E_1 , assuming $E_0 < E_1$. In this energy eigenbasis, the Hamiltonian of the qubit is given by

$$\hat{H}_q = E_0 |0\rangle\langle 0| + E_1 |1\rangle\langle 1| = \begin{pmatrix} E_0 & 0 \\ 0 & E_1 \end{pmatrix}. \quad (1.2)$$

Since only energy differences are physically relevant, it is possible to redefine the zero-point energy by subtracting the constant term $E_0(|0\rangle\langle 0| + |1\rangle\langle 1|)$, leading to the simplified Hamiltonian

$$\hat{H}_q = (E_1 - E_0) |1\rangle\langle 1| = \hbar\omega_q |1\rangle\langle 1| = \hbar\omega_q \hat{\sigma}^+ \hat{\sigma}^- = \begin{pmatrix} 0 & 0 \\ 0 & \hbar\omega_q \end{pmatrix}, \quad (1.3)$$

where $\omega_q = (E_1 - E_0)/\hbar$ is the qubit transition frequency, and we have used the relation $\hat{\sigma}^+ \hat{\sigma}^- = |1\rangle\langle 1|$. For convenience, the Hamiltonian can also be rewritten in terms of the Pauli z -matrix, $\hat{\sigma}_z$, by adding a term proportional to the identity:

$$\hat{H}_q = \hbar\omega_q |1\rangle\langle 1| - \frac{\hbar\omega_q}{2} \mathbb{I} = \begin{pmatrix} -\frac{\hbar\omega_q}{2} & 0 \\ 0 & \frac{\hbar\omega_q}{2} \end{pmatrix} = -\frac{\hbar\omega_q}{2} \hat{\sigma}_z. \quad (1.4)$$

Qubits can be implemented through various physical mechanisms; however, their practical realization remains a significant challenge due to their susceptibility to environmental interactions, which lead to decoherence and reduce their coherence time. Despite the diversity of possible physical implementations, any functional quantum computing system must satisfy a set of fundamental criteria. These requirements, known as the DiVincenzo criteria, establish the essential conditions for the construction and operation of a viable quantum computer [6], [7]:

1. The physical system used as quantum computer must comprise a set of qubits, meaning that the quantum system must be well-characterized, and scalable such that quantum computing can be realized.
2. It must be possible to initialize the qubits in a reliable state, such as the ground state.
3. The coherence time of the qubits must be longer than the typical gate time, ideally should be possible to perform $> 10^4$ operations, that is the number which allows for realizing effective error corrections.
4. It must be possible to implement a universal set of quantum gates.
5. It must be possible to measure the qubits in the computational basis.

In the present work, I will focus on superconducting qubits, which constitute the hardware I have worked on and where the experiments were conducted. However, several of the experiments described later can also be implemented using different physical systems.

1.2 Transmon qubits

In this section, I provide a review of the structure and operation of superconducting transmon qubits. The content of this section is based on the *Quantum Information Science* manual [7], *The Metrology of Quantum Control and Measurement in Superconducting Qubits* [5], the notes from quantum computing lectures held by Professor Olivares [8], and the original transmon paper [9].

1.2.1 Josephson Junctions

The Josephson junction (JJ) is formed by a thin oxide layer positioned between the two superconductors which acts as an insulating barriers. An example of Josephson junction is show in figure (1.2), a side view in image (1.2a) and a top view in image (1.2b).



Figure 1.2: Figure (1.2a): Side viwe of a Josephson junction, the two superconducting pads are coloured in red and blue and indicating by the letter S. In grey, indicated by letter I is represented the insulating barrier of oxide. The superconductors and the oxide are layered over a substrate.

Figure (1.2b): Electron microscope image of a $2\mu\text{m} \times 2\mu\text{m}$ cross-type junction: I. Josephson junction. II. Base electrode. III. Contact to the top electrode.

Source:<https://www.ims.kit.edu/english/2551.php>

Superconductivity is a phenomenon observed in certain materials where, when cooled well below a critical temperature T_c , which depends on the material, their electrical resistance drops to zero, allowing them to behave as perfect conductors. According to the BCS (Bardeen-Cooper-Schrieffer) theory, superconductivity arises, from the formation of Cooper pairs, which are bound states of electrons with opposite momenta and spins. These pairs collectively forms a macroscopic quantum states described by a single waveform $\psi(\mathbf{r})$ which can be expressed as

$$\psi(\mathbf{r}) = \sqrt{\rho(\mathbf{r})} e^{i\theta(\mathbf{r})} \quad (1.5)$$

where $\rho(\mathbf{r})$ is the density of Cooper pairs in the metal, which is typically uniform in the bulk of the superconductor, and $\theta(\mathbf{r})$ is the macroscopic phase of the superconducting wavfunction.

For this reason the wavefunctions on the two sides of the JJ can be denoted as

$$\psi_1(\mathbf{r}, t) = \sqrt{\rho_1(\mathbf{r}, t)} e^{i\theta_1(\mathbf{r}, t)}, \psi_2(\mathbf{r}, t) = \sqrt{\rho_2(\mathbf{r}, t)} e^{i\theta_2(\mathbf{r}, t)} \quad (1.6)$$

The dynamics of the system can be described by the two equations

$$i\hbar \frac{d\psi_1}{dt} = E_1 \psi_1 + K \psi_2, \quad (1.7)$$

$$i\hbar \frac{d\psi_2}{dt} = E_2 \psi_2 + K \psi_1. \quad (1.8)$$

By substituting the expression of ψ_i into the Schrödinger equation (1.7), (1.8) we obtain

$$\frac{d\rho_1}{dt} = \frac{2K}{\hbar} \sqrt{\rho_1 \rho_2} \sin(\theta_2 - \theta_1), \quad (1.9)$$

$$\frac{d\rho_2}{dt} = -\frac{2K}{\hbar} \sqrt{\rho_1 \rho_2} \sin(\theta_2 - \theta_1). \quad (1.10)$$

Since the derivative of the charge density is the current, from equations (1.9) and (1.10) we obtain the first Josephson equation

$$I = I_c \sin \phi \quad (1.11)$$

where $I_c = \frac{2K}{\hbar} \sqrt{\rho_1 \rho_2}$ is the critical current and ϕ is the superconducting phase difference $\theta_2 - \theta_1$.

Instead, from the real part of the Schrödinger equation (1.7), (1.8) and a few calculations, we obtain the second Josephson equation

$$\frac{d\phi}{dt} = \frac{2e}{\hbar} V(t). \quad (1.12)$$

which can be rewritten as

$$\frac{d\phi}{dt} = \frac{2\pi}{\Phi_0} V(t). \quad (1.13)$$

where $\Phi_0 = \frac{h}{2e}$ is the superconducting flux quantum, with h is the Planck's constant and $2e$ is the charge of a Cooper pair.

The time derivative of the first Josephson equation (1.11) yields:

$$\dot{I}_J = I_C \cos \phi \frac{\partial \phi}{\partial t}, \quad (1.14)$$

equation (1.14) suggests a nonlinear relation between the current the voltage. Using the Josephson voltage-phase relation and the fact that $\dot{I} = \frac{V}{L}$ it is possible to define an effective nonlinear inductance for the Josephson junction:

$$L_J = \frac{1}{\cos \phi} \frac{\Phi_0}{2\pi I_c}. \quad (1.15)$$

In addition to the inductive behaviour the Josephson junction also exhibits capacitive properties due to its inherent capacitance C_J with a corresponding energy of

$$E_{C_J} = \frac{Q^2}{2C_J} \quad (1.16)$$

From equation (1.15) it is possible to compute the energy stored in the nonlinear inductance as

$$E_{L_J} = \int_0^t d\tau I_J(\tau) V(\tau) = \int_0^t d\tau I_c \sin \phi(\tau) \frac{\partial \phi(\tau)}{\partial \tau} \frac{\Phi_0}{2\pi} \quad (1.17)$$

$$= \frac{\Phi_0 I_c}{2\pi} (1 - \cos \phi) = E_J (1 - \cos \phi) \quad (1.18)$$

where E_J represents the energy due to the behaviour of the junction as nonlinear inductor .

1.2.2 CPB qubit

A first example of superconducting qubit is the Cooper Pair Box (CPB), which consists of a small superconducting island connected to a reservoir of superconducting electrons through a Josephson junction [10], with an external gate voltage controlling the charge state. The circuit corresponding to CPB is similar to the circuit of a parallel resonator where the linear inductance is substituted by a Josephson junction which simply acts as a nonlinear inductance.

Combining the energy associated to the capacitance C and the energy of the Josephson junction (1.17) it is possible to write the classical Hamiltonian of the circuit

$$H_J = 4E_C n^2 - E_J \cos \phi \quad (1.19)$$

where the constant term was ignored as it acts simply as a constant offset without influencing the dynamics of the system and where E_C is the charging energy defined as

$$E_C = \frac{e^2}{2C}. \quad (1.20)$$

To control the number of Cooper pairs on the island, it is possible to connect a DC voltage source V_g to the system through a gate capacitor C_g , as shown in Figure (1.3b). When $V_g = 0$, both the gate and qubit capacitors remain uncharged. As V_g increases, a charge $Q_g = C_g V_g$ accumulates on the gate capacitor, inducing an equal and opposite charge on the island to maintain charge neutrality. When $Q_g \approx 2e$, a Cooper pair tunnels from the reservoir to the island, discharging the qubit capacitor.

The presence of the external voltage source introduces an additional control parameter for the number of Cooper pairs on the island, modifying the system's Hamiltonian. The resulting Hamiltonian of the Cooper Pair Box (CPB) takes the form

$$\hat{H} = 4E_C(\hat{n} - n_g)^2 - E_J \cos \hat{\phi} \quad (1.21)$$

where $n_g = \frac{C_g V_g}{2e}$ represents the normalized gate charge.

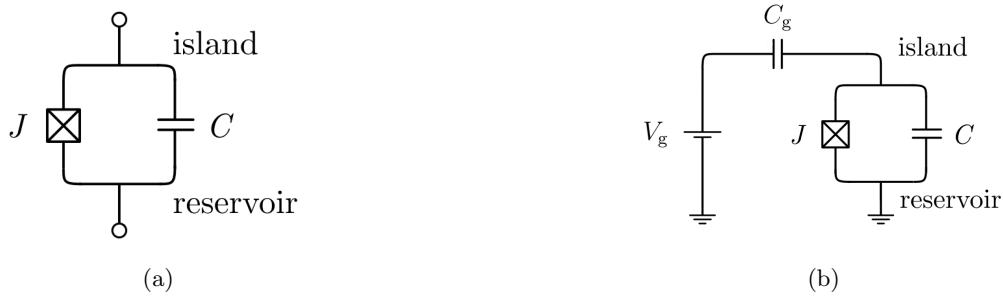


Figure 1.3: Figure (1.3a): corresponding circuit of a CPB which consists of Josephson junction shunted by a capacitor. Source: [7]. Figure (1.3b): electrical circuit of a CPB capacitively coupled to a DC voltage source through a capacitor C_g . Source: [7].

A key feature of the system is the presence of a Josephson junction in place of a linear inductance. Unlike a standard LC circuit—corresponding to a quantum harmonic oscillator—this results in a non-equidistant energy level spectrum. In particular, the energy levels are anharmonic, allowing the first two levels to be spectrally isolated from the higher ones. This anharmonicity enables the use of the subspace spanned by the ground state $|0\rangle$ and the first excited state $|1\rangle$ as a qubit.

The qubit frequency is defined as the frequency associated with the energy difference between these two states: $f_{01}(n_g) = f_Q = \frac{(E_1 - E_0)}{\hbar}$. This frequency can be tuned by varying the externally applied DC voltage, which modifies the system's parameters and, consequently, its energy level spacing.

1.2.3 Transmon qubit

One of the main drawbacks of the Cooper Pair Box (CPB) qubit, which ultimately led to its replacement by other qubit architectures, is its limited coherence time. The transmon qubit was introduced specifically to address this issue, with the goal of improving the dephasing time of the CPB. The key idea behind the transmon is to reduce the sensitivity of the energy levels to fluctuations in the gate charge—effectively flattening the energy bands—by increasing the ratio between the Josephson energy $E_J/E_C \geq 1$, this architecture was first proposed in [9], the first and more straightforward method to increase this ratio is to enlarge the capacitance of the qubit, which reduces the charging energy E_C .

Since the CPB and the transmon qubit have the same electrical circuit they are also described by the same Hamiltonian (1.19). The difference is that in this case the transmon satisfies the condition $E_J/E_C \geq 1$ it is possible to expand the cosine term in (1.19) with a Taylor series and neglect the higher order terms:

$$\hat{H} \approx 4E_C\hat{n}^2 + \frac{1}{2}E_J\hat{\phi}^2 - \frac{E_J}{4!}\hat{\phi}^4 \quad (1.22)$$

where the last term, proportional to $\hat{\phi}^4$, makes the potential of the transmon slightly anharmonic.

As happens in the standard harmonic oscillator case, the operators $\hat{\phi}$ and \hat{n} satisfy the canonical commutation relation $[\hat{\phi}, \hat{n}] = i\mathbb{I}$, it is possible to introduce the raising and lowering operators \hat{b}, \hat{b}^\dagger as

$$\hat{\phi} = \sqrt{\xi}(\hat{b} + \hat{b}^\dagger), \quad \hat{n} = -\frac{i}{2\sqrt{\xi i}}(\hat{b} - \hat{b}^\dagger), \quad (1.23)$$

where $\xi = \sqrt{2E_C/E_J}$.

Substituting equations (1.23) in the Hamiltonian, equation (1.22) becomes

$$\hat{H} = \sqrt{8E_J E_C} \hat{b}^\dagger \hat{b} - \frac{E_C}{12} (\hat{b} + \hat{b}^\dagger)^4. \quad (1.24)$$

Given equation (1.24) it is possible to solve the eigenvalue problem $\hat{H}|k\rangle = E_k|k\rangle$ and calculate the energy levels E_k . The first term of Hamiltonian (1.24) is the harmonic oscillator contribution with eigenstates $|k\rangle$ and eigenvalues $\sqrt{8E_J E_C}k$. Since $E_C \ll E_J$, the second term $\hat{V} = -\frac{E_C}{12}(\hat{b} + \hat{b}^\dagger)^4$ represents a small perturbative contribution to the Hamiltonian and can be treated using perturbation theory. The first-order correction to the energy levels is given by the diagonal matrix elements of the perturbation

operator: $\Delta E_k^{(1)} = \langle k | \hat{V} | k \rangle$. It is possible to verify that $\langle k | \hat{V} | k \rangle = -\frac{E_C}{12}(6k^2 + 6k + 3)$. Thus the eigenenergies of the transmon Hamiltonian are

$$E_k \approx \sqrt{8E_J E_C} k - \frac{E_C}{2}(k^2 + k). \quad (1.25)$$

As mentioned before, the qubit frequency is defined as $f_Q = f_{01} = (E_1 - E_0)/h$ which yields

$$f_{01} \approx (\sqrt{8E_J E_C} - E_C)/h \quad (1.26)$$

As explained at the beginning of this section, a large E_J/E_C ratio makes the transmon qubit significantly less sensitive to charge noise. However, this improvement comes at the expense of reduced anharmonicity in the energy level spectrum. The anharmonicity η is defined as the difference between the second and first transition energies, relative to the first transition energy:

$$\eta = \frac{(E_2 - E_1) - (E_1 - E_0)}{\hbar} = \omega_{12} - \omega_{01}. \quad (1.27)$$

For a transmon, the anharmonicity η is negative, reflecting the fact that the level spacing decreases with increasing energy. Ideally, the absolute value $|\eta|$ should be sufficiently large to allow external microwave drives to selectively address the $|0\rangle \leftrightarrow |1\rangle$ transition without inadvertently exciting higher-energy states.

1.2.4 Flux-tunable transmon

To implement certain two-qubit gate schemes, such as swap interactions, it is essential to tune the qubit frequency. A common approach to achieving this is by adding an extra junction to the transmon, the most common configuration is the SQUID (Superconducting QUantum Interference Device). In the SQUID configuration two Josephson junctions are connected in parallel on a superconducting loop, as shown in Figure (1.4).

Starting from the Hamiltonian of the single Josephson junction it is possible to write the Hamiltonian of a SQUID:

$$\hat{H} = 4E_C \hat{n}^2 - E_{J1} \cos \hat{\phi}_1 - E_{J2} \cos \hat{\phi}_2 \quad (1.28)$$

where E_{J1} and E_{J2} are the Josephson energies of the two junctions, and the operators $\hat{\phi}_1$ and $\hat{\phi}_2$ are the phase differences across the junctions.

Because of the quantization of the magnetic flux through the SQUID, the quantities $\hat{\phi}_1$ and $\hat{\phi}_2$ are not independent. In particular, as shown in [7], the difference between $\hat{\phi}_1$ and $\hat{\phi}_2$ follows the following relation:

$$\hat{\phi}_1 - \hat{\phi}_2 = \frac{2\pi}{\Phi_0} \Phi_{\text{ext}} \pmod{2\pi} \quad (1.29)$$

where Φ_{ext} is the flux of external magnetic field defined as the integral of the magnetic field over the SQUID area.

Equation (1.29) can be simplified and rewritten as

$$\hat{H} = 4E_C \hat{n}^2 - E_J(\Phi_{\text{ext}}) \cos \hat{\varphi} \quad (1.30)$$

where $\hat{\varphi} = \frac{\hat{\phi}_1 + \hat{\phi}_2}{2}$, and the Josephson energy is flux-dependent:

$$E_J(\Phi_{\text{ext}}) = (E_{J1} + E_{J2}) \left| \cos \left(\pi \frac{\Phi_{\text{ext}}}{\Phi_0} \right) \right| \sqrt{1 + d^2 \tan^2 \left(\pi \frac{\Phi_{\text{ext}}}{\Phi_0} \right)}, \quad (1.31)$$

where $d = \frac{E_{J1} - E_{J2}}{E_{J1} + E_{J2}}$ is the relative junction asymmetry.

Then, it's easy to see that the frequency f_Q of a two-junction transmon depends on the magnetic flux $\Phi_Q(t)$ through the SQUID loop, for symmetric junctions is given by

$$f_Q(\Phi_Q) \approx \frac{1}{h} \left(\sqrt{8E_J E_C \cos \left(\pi \frac{\Phi_Q}{\Phi_0} \right)} - E_C \right). \quad (1.32)$$

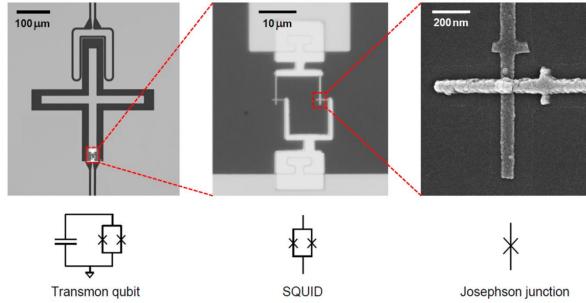


Figure 1.4: Images of a flux tunable transmon qubit. From left to right: the flux tunable transmon qubit, consisting of a large cross-shaped capacitance in parallel with a SQUID to ground, and its corresponding circuit. A zoom in of the SQUID (center), a single Josephson junction (right). Source: [11]

1.3 Qubit readout

Up to this point, I have discussed the physical structure of a transmon qubit. However, in order to perform quantum computing, it is essential to be able to control and measure its quantum state. One approach to achieve this is to capacitively couple the qubit to both a drive line and a readout line. The drive line is used to manipulate the state of the qubit, while the readout line is employed to measure it. An example of such a system is shown in Figure (1.5), along with the corresponding circuit diagram.

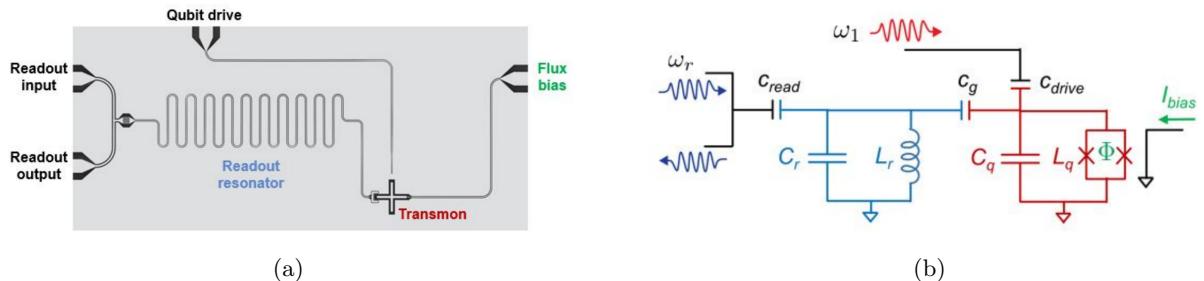


Figure 1.5: Figure (1.5a) shows an example of a single transmon device. Figure (1.5b) shows the equivalent lumped element circuit model of the device in (1.5a), in blue is represented the resonator circuit while in red the transmon qubit circuit. Source: [11]

As mentioned in the introduction of this section, in circuit quantum electrodynamics (cQED), the qubit state is measured via a dispersive interaction between a qubit and a far-detuned microwave resonator.

In Figure (1.5b), in blue is represented the resonator circuit capacitively coupled to the (red) qubit that is used for the readout of the qubit state. The resonator circuit is characterized by a an inductance L_r and a capacitance C_r , then the characteristic frequency is $\omega_r = 1/\sqrt{L_r C_r}$.

The classical Hamiltonian of the resonator can be written as

$$H_r = \frac{Q^2}{2C_r} + \frac{1}{2}C\omega_r^2\Phi^2, \quad (1.33)$$

where Φ is the generalized flux, defined as the time integral of the voltage across the capacitor:

$$\Phi(t) = \int_{-\infty}^t V(t')dt'. \quad (1.34)$$

The quantization of the Hamiltonian involves replacing the classical conjugate variables with their corresponding hermitian operators $\hat{\Phi}$ and \hat{Q} with $[\hat{\Phi}, \hat{Q}] = i\hbar\mathbb{I}$ which leads to

$$\hat{H}_r = \frac{\hat{Q}^2}{2C_r} + \frac{1}{2}C\omega_r^2\hat{\Phi}^2. \quad (1.35)$$

To find the eigenvalues and eigenvectors of \hat{H}_r it is often convenient to introduce the raising and lowering operators, \hat{a} and \hat{a}^\dagger which satisfy $[\hat{a}, \hat{a}^\dagger] = \mathbb{I}$. These two non-hermitian operators are defined as

$$\hat{\Phi} = \Phi_{\text{zpf}}(\hat{a} + \hat{a}^\dagger), \quad (1.36)$$

$$\hat{Q} = -iQ_{\text{zpf}}(\hat{a} - \hat{a}^\dagger) \quad (1.37)$$

where $\Phi_{\text{zpf}} = \sqrt{\frac{\hbar}{2C_r\omega_r}}$ and $Q_{\text{zpf}} = \sqrt{\frac{C_r\hbar\omega_r}{2}}$ are the zero-point fluctuations.

Then it is possible to write the Hamiltonian of a microwave resonator in a quantum regime:

$$\hat{H}_r = \hbar\omega_r \left(\hat{a}^\dagger \hat{a} + \frac{1}{2} \right), \quad (1.38)$$

The system of the capacitively coupled transmon and resonator is built in such a way that there is a maximum coupling between the qubit and the resonator. The Hamiltonian of the system reads

$$\hat{H} = \hbar\omega_r \hat{a}^\dagger \hat{a} + 4E_C (\hat{n} + \frac{C_{\text{read}} \hat{V}}{2e})^2 - E_J \cos \hat{\phi} \quad (1.39)$$

$$= \hbar\omega_r \hat{a}^\dagger \hat{a} + 4E_C \hat{n}^2 - E_J \cos \hat{\phi} + \frac{4E_C}{e} \hat{n}_C \hat{V} \quad (1.40)$$

where $\hat{V} = \hat{Q}/C_r$ is the voltage across the resonator capacitor. Using equations (1.23) and (1.38) the Hamiltonian becomes

$$\hat{H} = \hbar\omega_r \hat{a}^\dagger \hat{a} + \sqrt{8E_J E_C} \hat{b}^\dagger \hat{b} - \frac{E_C}{12} (\hat{b} + \hat{b}^\dagger)^4 + \hbar g (\hat{b}^\dagger - \hat{b})(\hat{a} - \hat{a}^\dagger) \quad (1.41)$$

where was introduced the parameter g , known as coupling strength, that quantifies the strength of the coupling between the qubit and the resonator.:

$$g = \frac{2E_C}{\hbar e} \frac{C_{\text{read}}}{C_r} Q_{\text{zpf}} \sqrt{\xi} = \frac{E_C}{\hbar e} \left(\frac{E_J}{2E_C} \right)^{1/4} \frac{C_{\text{read}}}{C_r} \sqrt{2\hbar\omega_r C_r}. \quad (1.42)$$

The coupling strength can be adjust by varying the capacitance coupling the qubit and the resonator.

When $g \ll \omega_r$ and $g \ll \omega_q$ it is possible to use the rotating wave approximation (RWA) and write the Hamiltonian in the form

$$\hat{H} = \hbar\omega_r \hat{a}^\dagger \hat{a} + \sqrt{8E_J E_C} \hat{b}^\dagger \hat{b} - \frac{E_C}{12} (\hat{b} + \hat{b}^\dagger)^4 + \hbar g (\hat{b}^\dagger \hat{a} + \hat{b} \hat{a}^\dagger). \quad (1.43)$$

Focusing on the first two levels of the transmon we obtain the Jaynes-Cummings Hamiltonian that reads

$$\hat{H} = \hbar\omega_r \hat{a}^\dagger \hat{a} - \frac{\hbar\omega_{01}}{2} \hat{\sigma}_z + \hbar g (\hat{\sigma}_+ \hat{a} + \hat{\sigma}_- \hat{a}^\dagger), \quad (1.44)$$

that describes the interaction between an atom, in this case an artificial atom, with an electromagnetic field in the approximation of the two-level system.

When the coupling strength g is much smaller than the detuning between the qubit and the resonator $\Delta = \omega_q - \omega_r$ the system operates in the dispersive regime.

The Jaynes-Cummings Hamiltonian in the dispersive regime, which is the condition in which we operate to perform the qubit readout, can be approximated as

$$\hat{H}_{\text{disp}} = \hbar(\omega_r - \chi \hat{\sigma}_z) \hat{a}^\dagger \hat{a} - \frac{\hbar}{2} (\omega_{01} + \chi) \hat{\sigma}_z \quad (1.45)$$

where χ is the dispersive shift defined as

$$\chi = \frac{g^2}{\Delta}. \quad (1.46)$$

Equation (1.45) shows that there is a shift in the resonator frequency from ω_r to $\omega_r - \chi$ if the qubit is in the ground state and a shift from ω_r to $\omega_r + \chi$ if the qubit is in the excited state.

The dispersive shift equation (1.46) was derives assuming that the qubit can be approximated as a two level system. Considering also the higher energy levels of the qubit a more accurate expression of the dispersive shift is given by

$$\chi = \frac{g^2}{\Delta(1 + \Delta/\eta)} \quad (1.47)$$

where η is the qubit anharmonicity.

1.4 Qubit control

Another necessary element for performing quantum computation is the implementation, starting with single-qubit gates. A qubit can be driven into any arbitrary superposition state by applying an electrical pulse with a carefully controlled amplitude, duration, and phase. This pulse is generated by an AC voltage source located outside the dilution refrigerator that hosts the qubit. The driving pulse is brought to the qubit by an on-chip waveguide which is capacitively coupled to the qubit as shown in Figure (1.5b), where the coupling capacitance is indicated with c . This signal path is commonly referred to as a control line or an XY line. The pulse that arrives at the device has the analytical form

$$V_d(t) = A\varepsilon(t) \sin \omega_d t + \alpha \quad (1.48)$$

where A is the pulse amplitude in volts, ω_d is the drive frequency in rad/s, α is the phase of the pulse and $\varepsilon(t)$ is the modulation of the pulse; the maximum of $\varepsilon(t)$ is fixed at one. As a first approximation, the envelope of the drive pulse is often chosen to have a Gaussian shape, which is preferred over a square pulse due to its smaller frequency bandwidth, minimizing the excitation of higher energy levels. However, the study of pulse shapes that minimize leakage to states outside the computational basis remains an active area of research [12].

In a similar way to what was previously done to study the capacitive coupling between the qubit and the resonator, it is possible to derive the Hamiltonian of the transmon capacitively coupled to the control line starting from the circuit shown in Figure (1.5b)

$$\hat{H} = 4E_C(\hat{n} + \frac{C_d V_d(t)}{2e})^2 - E_J \cos \hat{\phi} \quad (1.49)$$

, where $E_C = e^2/2C_\Sigma$ and $C_\Sigma = C_d + C_q$. By expanding the parenthesis and dropping a constant term the Hamiltonian can be re-written as

$$\hat{H} = 4E_C\hat{n}^2 - E_J \cos \hat{\phi} + 2e \frac{C_d}{C_\Sigma} V_d(t) \hat{n}. \quad (1.50)$$

Since $\hat{n} = -i(\hat{b} - \hat{b}^\dagger)/2\sqrt{\xi}$ (from equation (1.23)), the last term of the Hamiltonian can be rewritten:

$$\hat{H}_d = -2e \frac{C_d}{C_\Sigma} V_d(t) \hat{n} \quad (1.51)$$

$$= -i \frac{e}{\sqrt{\xi}} \frac{C_d}{C_\Sigma} V_d(t) (\hat{b} - \hat{b}^\dagger) \quad (1.52)$$

$$= -i \frac{e}{\sqrt{\xi}} \frac{C_d}{C_\Sigma} V_d(t) (|0\rangle\langle 1| - |1\rangle\langle 0|) \quad (1.53)$$

$$= \frac{e}{\sqrt{\xi}} \frac{C_d}{C_\Sigma} V_d(t) \hat{\sigma}_y \quad (1.54)$$

Regarding the first part of equation (1.50), by focusing on the first two levels of the transmon the qubit Hamiltonian can be rewritten as $\hat{H}_q = -\frac{1}{2}\hbar\omega_q \hat{\sigma}_z$. Then, by substituting the qubit and drive Hamiltonians (equation (1.51)) and the analytical form of $V_d(t)$ in equation (1.50) we obtain

$$\hat{H} = \hat{H}_q + \hat{H}_d = -\frac{\hbar\omega_q}{2} \hat{\sigma}_z + \frac{e}{\sqrt{\xi}} \frac{C_d}{C_\Sigma} A \varepsilon(t) \sin(\omega_d t + \alpha) \hat{\sigma}_y \quad (1.55)$$

$$= -\frac{\hbar\omega_q}{2} \hat{\sigma}_z + \hbar\Omega \epsilon(t) \sin(\omega_d t + \alpha) \hat{\sigma}_y, \quad (1.56)$$

where Ω is the Rabi frequency,

$$\Omega = \frac{e}{\hbar\sqrt{\xi}} \frac{C_d}{C_\Sigma} A. \quad (1.57)$$

The Rabi frequency quantifies the coupling between the control line and the qubit.

To study the dynamics of the system it is convenient to use the rotating frame, the Hamiltonian

becomes¹:

$$\hat{H}' = -\frac{\hbar(\omega_q - \omega_d)}{2}\hat{\sigma}_z + \hbar\Omega\varepsilon(t)\sin(\omega_dt + \alpha)(\hat{\sigma}_x\sin\omega_dt + \hat{\sigma}_y\cos\omega_dt) \quad (1.58)$$

$$= -\frac{\hbar(\omega_q - \omega_d)}{2}\hat{\sigma}_z + \hbar\Omega\varepsilon(t)\left(\frac{\hat{\sigma}_x}{2}(-\cos(2\omega_dt + \alpha) + \cos\alpha) + \frac{\hat{\sigma}_y}{2}(\sin(2\omega_dt + \alpha) + \sin\alpha)\right) \quad (1.59)$$

$$= -\frac{\hbar(\omega_q - \omega_d)}{2}\hat{\sigma}_z + \frac{\hbar\Omega}{2}\epsilon(t)(\hat{\sigma}_x\cos\alpha + \hat{\sigma}_y\sin\alpha) \quad (1.60)$$

where in the last step I used the rotating wave approximation, meaning that as $\omega_d \approx \omega_q$ the terms $\cos(2\omega_dt + \alpha)$ and $\sin(2\omega_dt + \alpha)$ oscillate rapidly and their contribution to the dynamics can be neglected.

For example, if $\omega_d = \omega_q$ and $\alpha = \frac{\pi}{2}$, the state evolution is described as follows:

$$|\psi\rangle = \exp\left\{\frac{i}{\hbar}\int_0^{+\infty}\hat{H}(t')dt'\right\}|0\rangle = \exp\left\{-i\frac{\Omega}{2}\int_0^{+\infty}\epsilon(t')dt'\right\}|0\rangle = e^{-i\frac{\theta}{2}\hat{\sigma}_y}|0\rangle \quad (1.61)$$

where

$$\theta = \Omega\int_0^{+\infty}\varepsilon(t')dt' \quad (1.62)$$

In general, a microwave pulse given by $V_d = A\epsilon(t)\sin(\omega_dt + \alpha)$ implements a single-qubit rotation $R_{\hat{n}(\alpha)}(\theta)$ around an axis \hat{n} that lies on the equator of the Bloch sphere, so that

$$R_{\hat{n}(\alpha)}(\theta) = e^{-\frac{i}{2}\hat{n}(\alpha)\cdot\vec{\sigma}\theta} = e^{-\frac{i}{2}(\hat{\sigma}_x\cos\alpha + \hat{\sigma}_y\sin\alpha)\theta} \quad (1.63)$$

1.5 Qubit state degradation

The ability to apply quantum gates to qubits is essential for executing complex circuits and operations. However, to ensure their effective implementation, it is crucial to study the qubit decoherence time, as it directly impacts the reliability and performance of quantum computations.

1.5.1 Qubit decoherence

By definition, decoherence refers to the loss of coherence in a quantum system, meaning that the relative phase between quantum states becomes randomized due to interactions with the environment. Decoherence is typically characterized by two key time constants: the energy relaxation time T_1 and the dephasing time T_2 . The energy relaxation time describes the characteristic time over which a qubit in an excited state decays to its ground state due to interactions with the environment. The decay follows an exponential law:

$$p_e(t) = p_e(0)e^{-\Gamma_1 t} \quad (1.64)$$

where $p_e(t)$ is the probability of the qubit being in the excited state at time t , $p_e(0)$ the probability of the qubit being in the excited state at time $t = 0$ and $\Gamma_1 = 1/T_1$ is the decay rate of the qubit from state $|0\rangle$ to state $|1\rangle$. Energy relaxation is primarily due to transverse noise, namely noise that couples the qubit through $\hat{\sigma}_x$ and/or $\hat{\sigma}_y$, it is caused by coupling to electromagnetic noise and other dissipative processes that induce energy exchange with the environment.

Longitudinal noise instead is due to environmental fluctuations coupling to σ_z that randomize the phase of the superposition state without causing energy exchange with the environment. The characteristic dephasing time of a qubit è indicato con $T_2^* = 1/\Gamma_\phi$ dove Γ_ϕ is the pure dephasing rate.

The transverse relaxation rate Γ_2 describes the combined effets of pure dephasing and energy relaxation and is defined as

$$\Gamma_2 = \frac{\Gamma_1}{2} + \Gamma_\phi, \quad (1.65)$$

or equivalently

$$\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_2^*}. \quad (1.66)$$

T_2 is known as decoherence time and quantifies how quickly a quantum superposition state loses phase coherence due to environmental fluctuations. For this reason, in an ideal system where pure dephasing is absent T_2 is simply limited by energy relaxation via the relaxation

$$T_{2,\max} = 2T_1. \quad (1.67)$$

¹For the complete derivation of the Hamiltonian in the rotating frame see Appendix ...

This result follows from the Bloch equations [13], where both energy relaxation and phase damping contribute to coherence loss. However, in real systems, additional dephasing mechanisms, such as low-frequency noise from charge or flux fluctuations, reduce T_2 below this upper bound and limits it to (1.66)

1.5.2 Noise models

Quantum noise mentioned in the previous subsection can be described using quantum operations known as quantum channels. A quantum operation is a mathematical transformation that describes how a quantum state changes as a consequence of a physical process. Formally, it is a map \mathcal{E} that transforms a quantum state described by a density operator $\hat{\rho}$ into another state described by a new density operator $\hat{\rho}'$:

$$\mathcal{E}(\rho) = \rho'. \quad (1.68)$$

Amplitude damping channel

The amplitude damping channel models energy dissipation due to interaction with a zero-temperature environment. The quantum map which describes the process is

$$\mathcal{E}_{\text{ad}}(\hat{\rho}) = \hat{E}_0 \hat{\rho} \hat{E}_0^\dagger + \hat{E}_1 \hat{\rho} \hat{E}_1^\dagger \quad (1.69)$$

where the Kraus operators:

$$\hat{E}_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix}, \quad \hat{E}_1 = \begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix}, \quad (1.70)$$

where $p = 1 - e^{-t/T_1}$ represents the probability of relaxation.

Generalized amplitude damping channel

The generalized amplitude damping channel models energy dissipation in a thermal environment where both relaxation and thermal excitation occur.

$$\mathcal{E}_{\text{gad}}(\hat{\rho}) = p \left(\hat{E}_0 \hat{\rho} \hat{E}_0^\dagger + \hat{E}_1 \hat{\rho} \hat{E}_1^\dagger \right) + (1-p) \left(\hat{E}_2 \hat{\rho} \hat{E}_0^\dagger + \hat{E}_3 \hat{\rho} \hat{E}_1^\dagger \right) \quad (1.71)$$

where:

$$\hat{E}_0 = \sqrt{p} \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix}, \quad = \sqrt{p} \begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix}, \quad (1.72)$$

$$\hat{E}_2 = \sqrt{1-p} \begin{bmatrix} \sqrt{1-\gamma} & 0 \\ 0 & 1 \end{bmatrix}, \quad \hat{E}_3 = \sqrt{1-p} \begin{bmatrix} 0 & 0 \\ \sqrt{\gamma} & 0 \end{bmatrix}, \quad (1.73)$$

where γ represents the thermal excitation probability and p describes the system-bath interaction strength.

Phase Damping channel

The phase damping channel describes pure dephasing, where the qubit phase is randomized without energy exchange. The Kraus operators for this channel are:

$$\hat{E}_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\lambda} \end{bmatrix}, \quad \hat{E}_1 = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{\lambda} \end{bmatrix}, \quad (1.74)$$

where $\lambda = 1 - e^{-t/T_2^*}$ quantifies the probability of phase randomization.

Depolarizing channel

The depolarizing channel captures the combined effects of relaxation (T_1) and pure dephasing (T_2^*), making it a natural model for total decoherence. The transformation applied by this channel is given by:

$$\rho \rightarrow (1-q)\rho + \frac{q}{3}(\sigma_x \rho \sigma_x + \sigma_y \rho \sigma_y + \sigma_z \rho \sigma_z), \quad (1.75)$$

meaning that the depolarizing channel leaves $\hat{\rho}$ unchanged with probability $(1-q)$, while with probability $q/3$ one of the Pauli operators is applied to it. The corresponding Kraus operators are:

$$\hat{E}_0 = \sqrt{1-q}I, \quad \hat{E}_1 = \sqrt{\frac{q}{3}}\sigma_x, \quad \hat{E}_2 = \sqrt{\frac{q}{3}}\sigma_y, \quad \hat{E}_3 = \sqrt{\frac{q}{3}}\sigma_z. \quad (1.76)$$

This channel models both energy relaxation and phase randomization, effectively representing total decoherence.

An alternative representation of the depolarizing channel describes the process in which the system state ρ is replaced by the maximally mixed state $\frac{\mathbb{I}}{2}$ with probability d :

$$\mathcal{E}_{dc}(\rho) = p \frac{\mathbb{I}}{2} + (1 - p)\rho. \quad (1.77)$$

where the following relation holds between p and q , $q = 3p/4$

Chapter 2

Qubit calibration

In this chapter I will describe the process of calibration for superconducting flux-tunable transmon on the hardware located in the QRC (Quantum Research Center) Laboratory of the TII (Technology & Innovation Institute) in Abu Dhabi.

2.1 Experimental setup

All the results presented in this work were obtained using the Contralto-D chip [14], which offers up to 21 fully connected qubits and 4 isolated qubits, for a total of 25 physical qubits. The distinction between fully connected and isolated qubits is important as only the fully connected subset supports direct two-qubit gate operations, which are essential for implementing entangling gates and complex quantum circuits. Isolated qubits, while still operational for single-qubit tasks, do not participate in multi-qubit interactions and thus are not functionally equivalent in terms of computational capabilities. The topology of the qubit is shown in figure (2.1b).

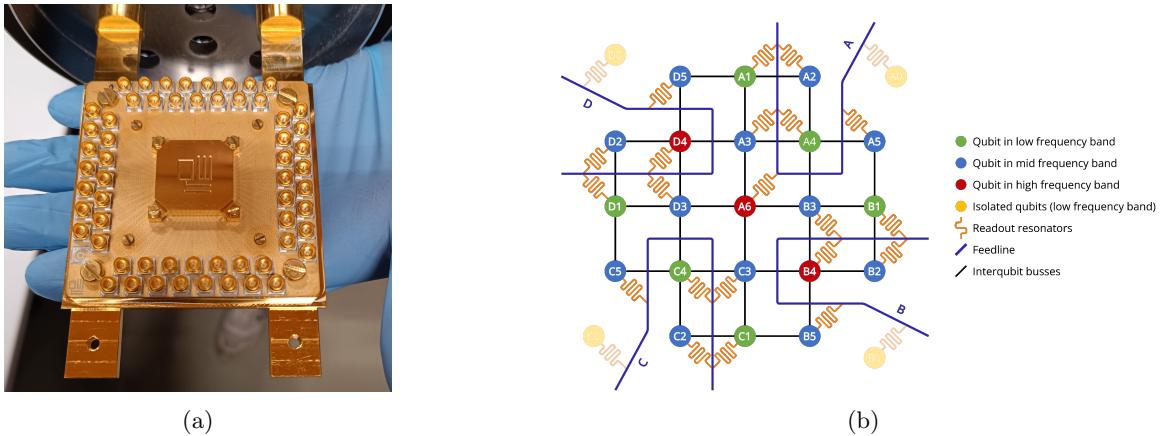


Figure 2.1: Figure (2.1a): Picture of the Contralto-D chip from QuantWare. Figure (2.1b): Topology of the Contralto-D chip from QuantWare.

As discussed in the previous chapter, the behavior of Josephson junctions and SQUIDs relies critically on the superconducting state of the materials involved. To achieve and maintain this regime, it is essential that the superconducting elements operate well below their critical temperature. For this reason, the Contralto-D chip is installed at the lowest temperature stage of the cryostat, where the required thermal conditions for superconductivity are met. This ensures the proper functioning of the quantum hardware and enables the realization of coherent quantum operations.

These systems achieve ultra-low temperatures by exploiting the unique quantum properties of helium-3 (^3He) and helium-4 (^4He) isotopes in a dilution process. At the core of a dilution refrigerator is a mixing chamber, where the cooling mechanism takes place. When a mixture of ^3He and ^4He is cooled below approximately 870 millikelvin, the two isotopes phase-separate into a ^3He -rich phase and a ^3He -dilute phase. The key principle is that when ^3He atoms cross the phase boundary—from the concentrated phase into the dilute phase—they absorb energy from their surroundings. This process is endothermic and is

the fundamental source of cooling in the dilution refrigerator.

The system operates as a closed loop: ^3He gas is circulated using a combination of sorption pumps and still pumps, which remove ^3He vapor from the still (typically at 600–800 mK), recondense it at a higher stage, and reintroduce it into the mixing chamber. The refrigerator includes several thermalization stages—typically at 50 K, 4 K, 800 mK, 100 mK, and finally below 20 mK—each connected to a corresponding cooling stage and separated by radiation shields and thermal filters to minimize heat load and noise from higher-temperature stages. Dilution refrigerators are highly stable and capable of reaching base temperatures below 10 mK, with hold times on the order of days or even weeks. These temperatures are crucial for achieving the low thermal noise and long coherence times necessary for high-fidelity quantum operations in superconducting circuits. Specifically the cryostat employed in the lab is the XLDsl from Bluefors [15], an image of the cryostat is shown in figure (2.2).

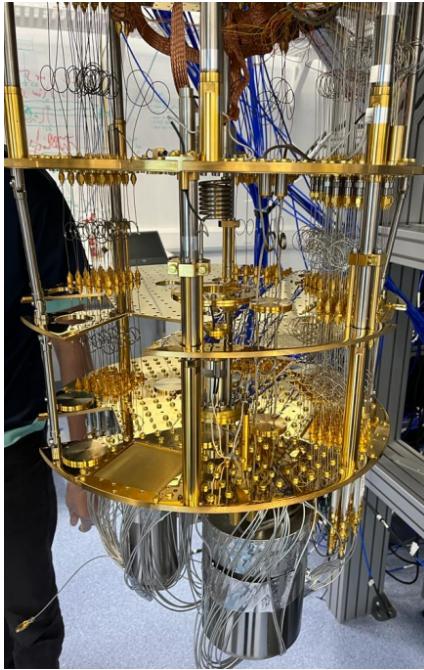


Figure 2.2: Picture of the XLDsl dilution refrigerator at the QRC Lab

Outside the cryostat, the control and readout of superconducting qubits are managed by dedicated room-temperature electronics. These systems are responsible for generating the microwave pulses used to drive single- and two-qubit gates, as well as for acquiring and processing the output signals that encode the qubit states. Typically they include arbitrary waveform generators (AWGs), microwave sources, mixers, digitizers, and field-programmable gate arrays (FPGAs). The generated microwave pulses are shaped and modulated at room temperature before being attenuated and routed to the cryogenic environment. Similarly, signals returning from the qubits are amplified and digitized for state discrimination and further processing. The electronics employed in the lab for the control of the `qw11q` is the OPX1000 platform by Quantum Machines [16].

The software I used for the calibration of the qubits and the subsequent experiments is **Qibocal**([17], [18], [19]), while the backend for communication with the laboratory instruments is **Qibolab**([20], [21], [22]). **Qibolab** is the control layer responsible for managing and executing low-level instructions on the hardware, bridging high-level quantum models and physical quantum platforms. It is designed to support diverse experimental setups and allows the researcher to define custom hardware configurations through a platform abstraction and to execute custom pulse sequences using both commercial and open-source firmware. The communication between **Qibolab** and the quantum hardware is structured and modular, relying on a stack that includes instrument drivers, pulse control logic, and a compiler that translates abstract quantum gates into hardware-specific instructions. This structure enables compatibility with heterogeneous platforms and facilitates the development of experimental drivers tailored to different laboratory environments. **Qibocal** interfaces directly with **Qibolab** to apply calibration protocols on the physical device. The routines deployment takes place through the interpretation of declarative runcards written in YAML. **Qibocal** allows an easy execution of pulse sequences, collection of measurement data, and interpretation of the results through the reports that are automatically generated upon completion

of the routine.

2.2 Single qubit calibration experiments

The first task that I needed to complete at the beginning of my thesis work was the calibration of at least a line of the superconducting qubits of the Contralto-D chip using the `Qibocallibrary`. From this point onward, for the sake of brevity, I will refer to the chip interchangeably as Contralto-D or `qw11q`, which is the name of the node under which it is registered on the QRC computing cluster. In the following I will describe the experiments that I performed and commenting on the results.

2.2.1 Resonator characterization

Resonator spectroscopy

The first step to calibrate the readout pulse is to characterize the resonator is to find the resonator frequency, that is the transition frequency for the resonator. At this frequency, a distinct difference in the transmitted signal can be observed depending on the type of resonator used. In the case of a 3D cavity resonator, the signal appears amplified, whereas for a 2D planar resonator, the signal tends to be more strongly absorbed. Regardless of the resonator type, the response typically exhibits a Lorentzian-shaped peak: this peak is positive for 3D cavities due to the amplification effect, and negative for 2D resonators due to their greater absorption.

The outcome of this experiment is strongly influenced by the amplitude of the excitation pulse. To reliably determine the resonator frequency, the pulse duration can be fixed on the order of microseconds, which is sufficient to observe the relevant signal response. However, selecting an appropriate amplitude requires more careful consideration. When the amplitude is high, the signal becomes more prominent, improving the signal-to-noise ratio and making it easier to identify the resonator's response. If the amplitude is increased too much, however, it can drive the system out of the superconducting regime. In this case, the resonator becomes effectively decoupled from the qubit, and the frequency observed corresponds to the so-called bare resonator frequency. An example of measurement of the bare resonator frequency is shown in Figure (2.3).

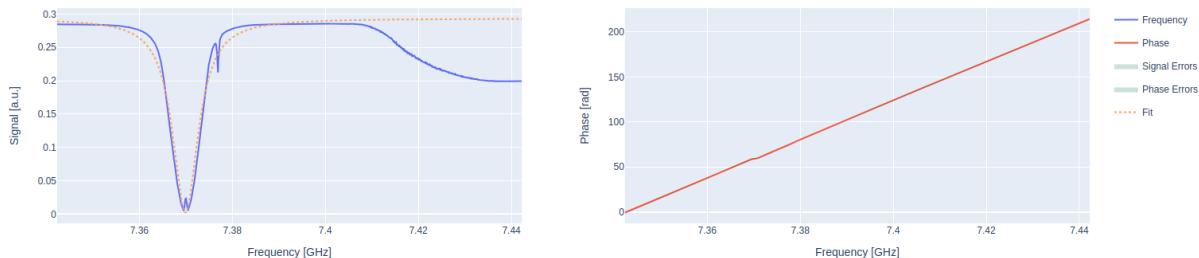


Figure 2.3: Output of resonator spectroscopy with high power on qubit B2.

Una misura della frequenza di risonanza del resonatore low power è invece riportato in Figure (2.4)

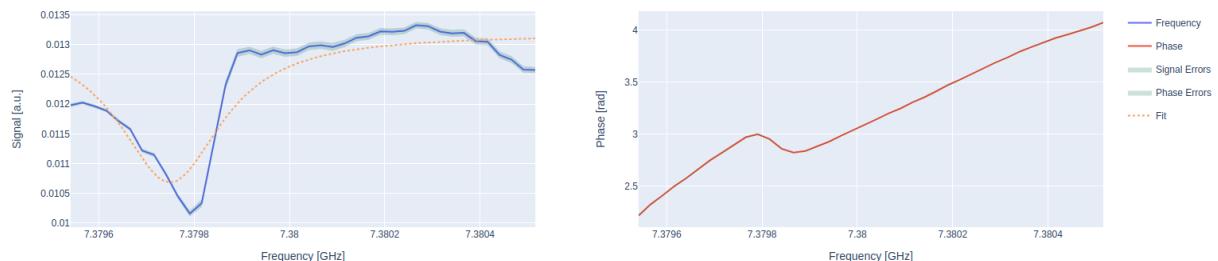


Figure 2.4: Output of resonator spectroscopy with low power on qubit B2.

Uno studio più preciso della risposta del risonatore può essere realizzato eseguendo la routine di resonator spectroscopy cambiando la funzione di fit, questa variazione dell'esperimento è descritta nell'appendice Appendix A

Resonator punchout

To identify the resonator frequency under qubit coupling, it is necessary to first determine the appropriate readout pulse amplitude. This can initially be explored using a vector network analyzer to verify system functionality and obtain a rough estimate of the relevant parameters. Then it is possible to repeat the spectroscopy, this time over a narrower frequency range and for varying pulse amplitudes. The resonator frequency is expected to depend strongly on amplitude: it remains constant in the high-power regime, shifts during an intermediate transition phase, and stabilizes again at a different value once the qubit-resonator interaction becomes significant.

Resonator flux dependence

Resonator crosstalk

2.2.2 Qubit characterization

Dopo aver calibrato e ricavato i parametri del risonatore accoppiato al qubit è possibile procedere con la calibrazione dei parametri relativi al qubit stesso.

Qubit spectroscopy

To determine the resonance frequency of a qubit, a qubit spectroscopy experiment is performed, which, unlike resonator spectroscopy, requires a two-tone approach. While resonator spectroscopy is typically a single-tone measurement used to identify the resonator's response, qubit spectroscopy involves applying a drive tone to the qubit followed by a readout tone to detect the qubit state. This method becomes essential after an initial estimate of the readout frequency and amplitude has been obtained from a resonator punchout experiment. In this protocol, a drive pulse of variable frequency ω is sent through the qubit drive line. If the drive frequency is far detuned from the qubit transition frequency ω_q , it will have no appreciable effect on the qubit state, and the measured signal will remain unchanged. However, as ω approaches ω_{01} the drive pulse can induce transitions between the qubit states. This excitation modifies the qubit population and, consequently, the resonator response, which is sensitive to the qubit state due to their dispersive coupling. When the drive frequency is near resonance and the pulse is sufficiently long, the qubit may reach a maximally mixed state leading to a detectable change in the readout signal amplitude. By sweeping the drive frequency and recording the corresponding readout amplitudes, one can plot a spectroscopy curve that reveals a Lorentzian dip or peak centered at the qubit transition frequency—opposite in direction to the Lorentzian feature observed in the resonator spectroscopy, due to the nature of the state-dependent dispersive shift.

An example of the output of qubit spectroscopy experiment is shown in figure (2.5)

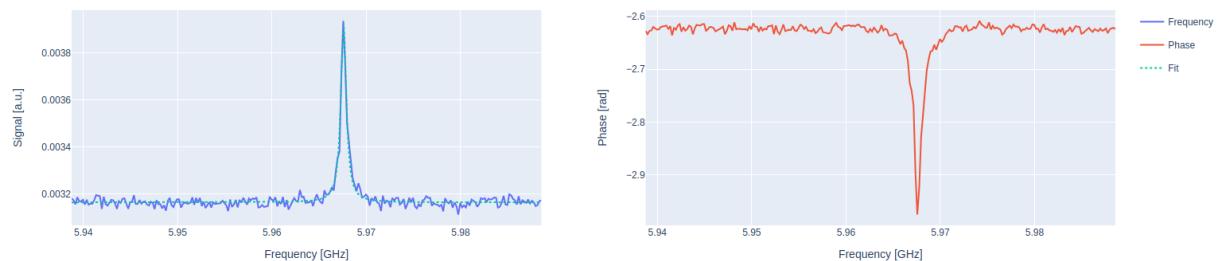


Figure 2.5: Output of qubit spectroscopy on qubit B2.

Qubit EF spectroscopy

Qubit spectroscopy can also be extended to probe transitions to higher excited states beyond the first excited state. Directly observing these higher-level transitions typically requires significantly increased drive power, which may exceed the safe operational limits of the experimental setup. An alternative and more controlled approach involves first preparing the qubit in state $|1\rangle$, followed by a standard spectroscopy sequence to induce the $|1\rangle \leftrightarrow |2\rangle$ transition.

Si noti che la descrizione di questa routine di calibrazione è stata inserita qui per una questione di chiarezza e continuità espositiva, tuttavia dato che richiede che il qubit all'inizio della spettroscopia si trovi nello stato $|1\rangle$ può essere eseguita solo dopo l'esecuzione di una single-shot classification(2.2.4).

Qubit flux dependence

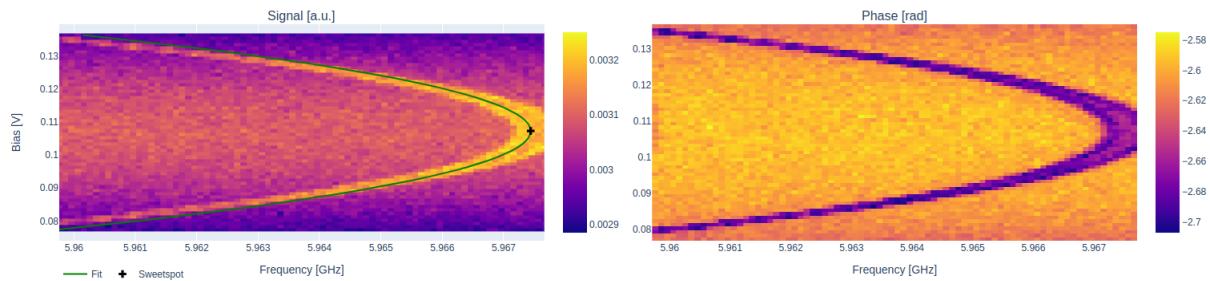


Figure 2.6: Output of the qubit flux dependence on qubit B2.

Qubit crosstalk

2.2.3 T1 & T2 measurement

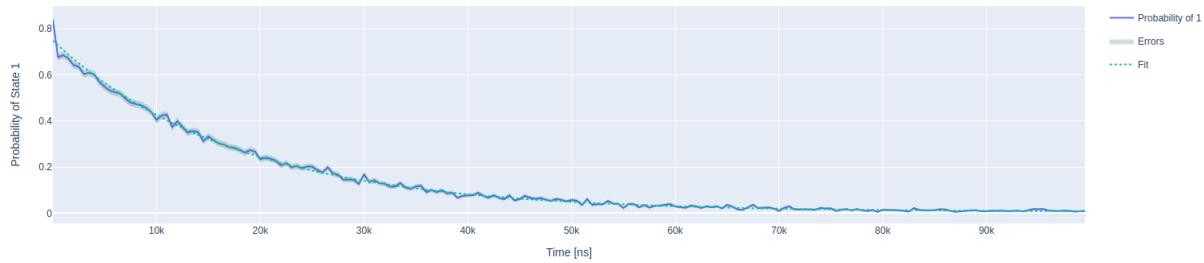


Figure 2.7: Output of the T_1 measurement on qubit B2.

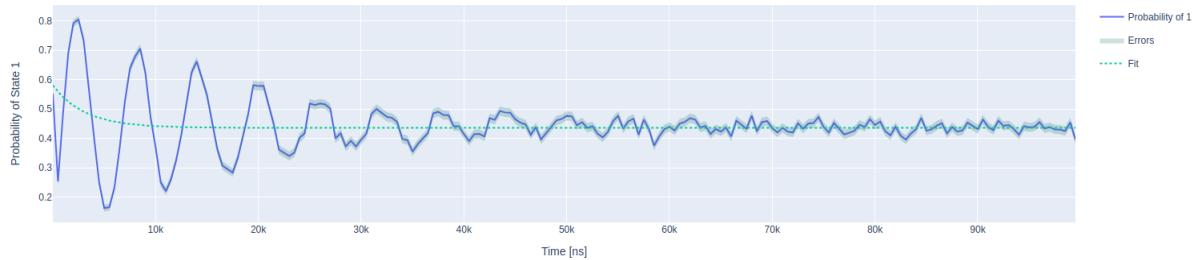


Figure 2.8: Output of the T_2 measurement on qubit B2.

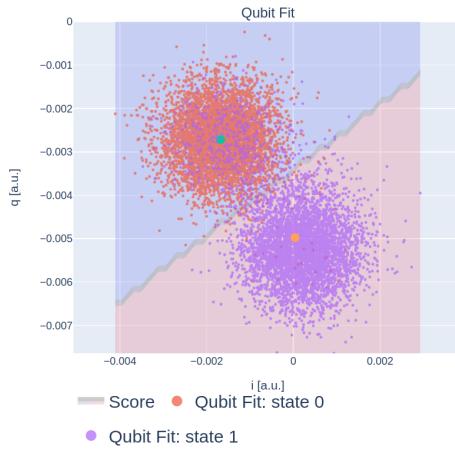


Figure 2.9: Output of the single shot classification on qubit B2.

2.2.4 Single shot classification

2.2.5 Gate calibration

Rabi amplitude

Rabi length

Rabi amplitude-length

2.2.6 Fine calibration

Ramsey experiment

Flipping experiment

Dispersive shift

Readout characterization

2.2.7 Standard Randomized Benchmarking

L'esperimento di standard randomized benchmarking verrà descritto più nel dettaglio nel capitolo successivo (see Section (3.1)), per il momento basti sapere che

2.2.8 DRAG experiment

[23][24]

Chapter 3

RB fidelity optimization

Tutti i risultati che sono presentati nel seguito sono stati ottenuti utilizzando il software di Qibolab per l'interazione con gli strumenti del laboratorio e Qibocal per il controllo delle operazioni sui qubit. L'hardware è un chip di QunatumWare. Durante il lavoro condotto per questo progetto di tesi entrambe le libererie, sia Qibocal che Qibolab undergo update and release, for this reason the first part of this work was realized using Qibocalv0.1 and Qibolabv0.1 while the second part of the work, dato che puntava anche allo sviluppo di routine che potessero essere utili per la calibrazione dei qubit è stato realizzato direttamente con Qibocalv0.2 e Qibolabv0.2.

3.1 Randomized Benchmarking

A strong limitation to the realization of quantum computing technologies is the loss of coherence that happens as a consequence of the application of many sequential quantum gates to the qubits. A possible approach to characterize gate error is the quantum process tomography which allows the experimenter to establish the behaviour of a quantum gates; the main drawback of this approach is that process tomography can be very time consuming since its time complexity scales exponentially with the number of qubits involved [25] and the result is affected by state preparation and measurements (SPAM) errors.

To overcome these limitations, randomized benchmarking (RB) was introduced and is currently widely used to quantify the average error rate for a set of quantum gates.

The main idea is that the error obtained from the combined action of random unitary gates drawn from a uniform distribution with respect to the Haar measure [26] and applied in sequence to the qubit will average out to behave like a depolarizing channel [27]. This last consideration simplifies the characterization of noise because it removes dependence on specific error structures and allows fidelity to be extracted through a simple exponential decay.

It was later shown that it is possible to simplify this procedure even more, by restricting the unitaries to gates in the Clifford group¹ and by not requiring that the sequence is strictly self-inverting [28].

The fundamental principle of RB is the application of sequences of randomly selected quantum gates from the Clifford group \mathcal{C} followed by an inversion gate which, in absence of noise, return the system to its initial state. For real systems, where noise is present, the observed survival probability provides an estimate of the average gate fidelity.

The standard RB protocols consist of the following steps:

1. Initialize the system in ground state $|0\rangle$
2. For each sequence-length m build a sequence of m randomly drawn Clifford gates C_1, C_2, \dots, C_m
3. Determine the inverse gate $C_{m+1} = (C_m \circ \dots \circ C_1)^{-1}$
4. Measure $C_{m+1} \circ C_m \circ \dots \circ C_1 |0\rangle$

The process must be repeated for multiple sequences of the same length and with varying length.

In ideal systems without noise we should have

$$C_{m+1} \circ C_m \circ \dots \circ C_1 |0\rangle = (C_m \circ \dots \circ C_1)^{-1} \circ (C_m \circ \dots \circ C_1) |0\rangle = |0\rangle \quad (3.1)$$

¹unitary rotations mapping the group of Pauli operators in itself

In real systems, where noise is present, eq. (3.1) does not hold; instead randomization with Clifford gates behave as a depolarizing channel (1.77) with depolarization probability d . The survival probability of the initial state $|0\rangle$ for different sequence lengths follows the exponential decay model

$$F(m) = Ap^m + B, \quad (3.2)$$

where $1 - p$ is the rate of depolarization and A and B capture the state preparation and measurement error but not the rate of decay p . Note that the exponential form arises naturally due to the assumption that each gate introduces independent noise.

The parameter p is directly related to the depolarization probability d through the average gate fidelity F which, for a depolarizing channel, is given by

$$F = 1 - \frac{d}{2^n - 1}. \quad (3.3)$$

For the details of the calculations to obtain eq. (3.3) see Appendix C.

Now we can derive the average error per Clifford gate $\varepsilon_{Clifford}$

$$\varepsilon_{Clifford} = 1 - F, \quad (3.4)$$

where F is the average gate fidelity. Substituting in (3.4) the formula for the average gate fidelity (3.3) we obtain

$$\varepsilon_{Clifford} = \frac{d}{2^n - 1} = \frac{1 - p}{1 - 2^{-n}}, \quad (3.5)$$

which shows how the average error per Clifford gate is directly connected to the exponential decay rate.

QUA Randomized Benchmarking

For the results we present in the following the technique used slightly differs from the one described in section (3.1)

3.2 Scipy optimization methods

3.2.1 Algorithm description

I primi metodi che abbiamo provato per l'ottimizzazione dei parametri sono quelli standard implementati nella libreria **Scipy** [29].

The first gradient-free optimization method to be tested was Nelder-Mead since in letteratura era già stato riportato il suo utilizzo per obiettivi simili [30].

The Nelder-Mead optimization method, originally introduced by Nelder and Mead in 1965 [31], is a widely used numerical optimization technique for unconstrained problems in multidimensional spaces. This derivative-free method operates using simplex, which is a polytope of $n + 1$ vertices in a n -dimensional space. The algorithm iteratively updates the simplex by replacing its worst-performing vertex with a new candidate point, thereby guiding the search towards an optimal solution. If the goal is to minimize a given function $f(\mathbf{x})$ where $\mathbf{x} \in \mathbb{R}^n$ the algorithm proceeds with the following steps:

1. If not otherwise initialized, $n + 1$ points are sampled for building the initial simplex
2. Order the test points according to their values at vertices: $f(\mathbf{x}_1) \leq f(\mathbf{x}_2) \leq \dots \leq f(\mathbf{x}_{n+1})$ and check whether the algorithm should terminate.
3. Calculate \mathbf{x}_0 , the centroid of all points except \mathbf{x}_{n+1} .
4. Reflection: Compute the reflected point $\mathbf{x}_r = \mathbf{x}_0 + \alpha(\mathbf{x}_0 - \mathbf{x}_{n+1})$ with $\alpha > 0$. If \mathbf{x}_r satisfies $f(\mathbf{x}_1) \leq f(\mathbf{x}_r) < f(\mathbf{x}_n)$, then a new simplex is obtained by replacing the worst-performing point \mathbf{x}_{n+1} with \mathbf{x}_r and then go to step 1.
5. Expansion: If \mathbf{x}_r is the current best point, meaning that $f(\mathbf{x}_r) < f(\mathbf{x}_1)$, then the expanded point is computed: $\mathbf{x}_e = \mathbf{x}_0 + \gamma(\mathbf{x}_r - \mathbf{x}_0)$ with $\gamma > 1$. If \mathbf{x}_e satisfies $f(\mathbf{x}_e) < f(\mathbf{x}_r)$, then a new simplex is obtained by replacing \mathbf{x}_{n+1} with the expanded point \mathbf{x}_e and then go to step 1.
If instead $f(\mathbf{x}_e) \geq f(\mathbf{x}_r)$, the new simplex is obtained by replacing \mathbf{x}_{n+1} with \mathbf{x}_r , and then go to step 1.

6. **Contraction:** In this case is certain that $f(\mathbf{x}_r) \geq f(\mathbf{x}_n)$ then:

- If $f(\mathbf{x}_r) < f(\mathbf{x}_{n+1})$: compute the contracted point $\mathbf{x}_c = \mathbf{x}_0 + \rho(\mathbf{x}_r - \mathbf{x}_0)$ with $0 < \rho \leq 0.5$. If \mathbf{x}_c satisfies $f(\mathbf{x}_c) < f(\mathbf{x}_r)$, then a new simplex is obtained by replacing \mathbf{x}_{n+1} with \mathbf{x}_c and go to step 1.
Else go to step 6.
- If $f(\mathbf{x}_r) \geq f(\mathbf{x}_{n+1})$: compute the contracted point $\mathbf{x}_c = \mathbf{x}_0 + \rho(\mathbf{x}_{n+1} - \mathbf{x}_0)$ with $0 < \rho \leq 0.5$. If \mathbf{x}_c satisfies $f(\mathbf{x}_c) < f(\mathbf{x}_{n+1})$, the a new simplex is constructed with \mathbf{x}_c and go to step 1.
Else go to step 6.

7. **Shrinkage:** Replace all points except the best, \mathbf{x}_1 , with $\mathbf{x}_i = \sigma(\mathbf{x}_i - \mathbf{x}_1), 0 < \sigma \leq 0.5$

The algorithm terminates when the standard deviation of the function values of the current simplex fall below a user-initialized tolerance. When the cycle stops the point of the simplex associated to the lower function value is returned as proposed optimum

The values of the parameters α, γ, ρ and σ were left to default of `scipy`: $\alpha = 1, \gamma = 2, \rho = 0.5, \sigma = 0.5$.

Per valutare eventuali miglioramenti nella performance abbiamo provato ad utilizzare un algoritmo che fosse gradient-based. Nello specifico ho provato ad utilizzare l'algoritmo di Sequential Least Squares Programming (SLSQP) nella versione implementata all'interno della libreria `scipy`.

3.2.2 Results

Di seguito riporto i risultati che abbiamo ottenuto utilizzando gli algoritmi descritti in precedenza per la minimizzazione del RB. Per prima cosa ho studiato come l'utilizzo di algoritmi di ottimizzazione che agiscono su ampiezza e durata ... sono in grado di migliorare la assignment fidelity partendo da una calibrazione non ottimale de qubit

3.3 CMA-ES

3.3.1 Algorithm description

Covariance Matrix Adaptation Evolution Strategy (CMA-ES [32]), is a population-based evolutionary algorithm designed for optimizing complex, non-convex, and high-dimensional functions.

It belongs to the broader class of Evolution Strategies (ES), a subset of Evolutionary Algorithms (EAs)(see [33]), and is particularly effective for black-box optimization where gradient information is unavailable.

Evolution Strategies (ES) are a class of optimization methods that employ self-adaptive mechanisms to explore the search space efficiently. Unlike classical optimization techniques that rely on gradient descent, ES leverage stochastic sampling to navigate rugged and multimodal landscapes. In this context, CMA-ES is an adaptive stochastic search method that iteratively refines a probability distribution over the search space. Unlike traditional Genetic Algorithms (GAs), which rely on crossover and mutation operators, CMA-ES employs a multivariate normal distribution to generate candidate solutions. The method adaptively updates the distribution's mean and covariance matrix based on the fitness of sampled points.

The fundamental idea behind CMA-ES is the use of a multivariate Gaussian distribution to model promising search directions. Let μ_t denote the mean of the distribution at iteration t , and Σ_t the covariance matrix. Then, a new population of λ candidate solutions $\mathbf{x}_t^{(t+1)} \sim \mu_t + \sigma_t \mathcal{N}(0, \Sigma_t)$, where σ_t is a step size controlling the exploration.

The CMA-ES algorithm follows the following steps:

1. If not otherwise specified, the initial parameters are set: mean vector μ_0 , covariance matrix Σ_0 ², step size σ_0 , population size λ
2. Generate λ new candidate solutions \mathbf{x}_i according to a multivariate normal distribution.
3. Evaluate the objective function $f(\mathbf{x}_i)$ for each candidate solution.
4. Sort the new candidate solutions based on fitness: $f(\mathbf{x}_0) \leq \dots \leq f(\mathbf{x}_\lambda)$.

² $\Sigma_0 = \mathbb{I}$ for isotropic search

5. Update the mean vector μ with the $m = \lfloor \lambda/2 \rfloor$ top performing solutions:

$$\mu \leftarrow \sum_{i=0}^m \mathbf{w}_i \mathbf{x}_i, \quad (3.6)$$

where \mathbf{w}_i are internally defined weights.

6. Update the isotropic and anisotropic evolution path $\mathbf{p}_\sigma, \mathbf{p}_c$ ³.
7. Update the covariance matrix:

$$C \leftarrow (1 - c_1 - c_\mu)C + c_1 \mathbf{p}_c \mathbf{p}_c^T + c_\mu \sum_{i=1}^\mu w_i \mathbf{y}_i \mathbf{y}_i^T, \quad (3.7)$$

where c_1 and c_μ are learning rates and \mathbf{y}_i represents the deviation of the i -th candidate solution from the mean \mathbf{mu} .

8. Update the step size using a cumulative path evolution mechanism

$$\sigma \leftarrow \sigma \cdot \exp \left(\frac{c_\sigma}{d_\sigma} (\|\mathbf{p}_\sigma\| - E\|\mathcal{N}(0, I)\|) \right), \quad (3.8)$$

where c_σ is the learning rate for step-size adaptation, d_σ is a damping factor $\|\mathbf{p}_\sigma\|$ is the length of the evolution path and $E\|\mathcal{N}(0, I)\|$ is the expected length of a standard normally distributed random vector.

Nel seguito, a meno che non sia diversamente specificato, i parametri sono stati inizializzati ai valori di default della libreria CMA-ES

3.3.2 Results

3.4 Optuna

3.4.1 Algorithm description

In addition to the optimization methods mentioned earlier, the Tree-Structured Parzen Estimator (TPE) method was employed, using its implementation available in the `optuna` library [34].

Tree-Structured Parzen Estimator (TPE) is a Sequential Model-Based Optimization (SMBO) approach [35]. SMBO methods sequentially construct models to approximate the performance of optimization parameters based on historical measurements, and then subsequently choose new parameters values to test based on this model. [36] At the heart of SMBO is the idea of building a surrogate model, which is used to predict the objective function's values for unseen parameters configurations. The surrogate model is iteratively updated as new observations are made, and the optimization process balances exploration, which focuses on uncertain regions of the search space, and exploitation, which focuses on areas that are more likely to improve the objective based on past evaluations. This balance ensures that the optimization process makes efficient use of resources and avoids wasting time on suboptimal regions.

The TPE algorithm is a probabilistic model-based optimization method that uses non-parametric density estimation to guide the search. The TPE algorithm differs from traditional Bayesian optimization approaches, such as Gaussian Process-based methods, in its modeling strategy. Rather than directly approximating the objective function, TPE constructs two separate probabilistic models:

- $p(x|y < y^*)$, the likelihood of observing a parameter configuration x given that the objective function value y is below a chosen threshold y^* .
- $p(x|y \geq y^*)$, the likelihood of observing X for less promising function values.

These probability densities are estimated using non-parametric methods such as kernel density estimation (KDE). New candidate points are then generated by sampling from $p(x|y < y^*)$, favoring

³For details on the update process of the evolution paths see [32].

configurations that are expected to yield lower objective values. The threshold y^* is typically set as a quantile of observed values, ensuring a focus on the most promising regions of the search space.

The TPE method is the default optimization strategy in `Optuna`, an advantage in the optimization algorithm as implemented in `optuna` is the addition of an automatic *pruning* mechanism that stops unpromising trials early, which can significantly speed up the optimization process by avoiding unnecessary computations. In our case, this is particularly relevant because the execution of the RB routine, which is performed at each call to the cost function, requires [insert approximate execution time]

As implemented in our code, the default pruner used is the median pruner `optuna.pruners.MedianPruner`. This pruner works by evaluating the intermediate results of a trial and comparing them to the median of completed trials at the same step. If the current trial's performance is worse than the median, it is pruned to prevent wasting computational resources on unpromising configurations.

3.4.2 Results

D1 - steps

D1 - 1000 steps

D2 - 1000 steps

3.5 RB optimization conclusions

Chapter 4

Pulses analysis and tuning

Having concluded that closed-loop optimization would not significantly improve fidelity, we shifted our focus towards improvement and implementation of individual protocols to improve the accuracy of qubit operations.

In this chapter, I present the results of two additions to the `Qibocal` software. The first is the inclusion of an *RX90* gate as a native gate, which can enhance the performance of protocols requiring qubit rotations of $\frac{\pi}{2}$. The second is the implementation of the cryoscope, a routine first described in [37], which is useful for correcting distortions in the magnetic flux pulse applied to the SQUID.

4.1 RX90 calibration

4.2 Flux pulse correction

4.2.1 Cryoscope

The experiment that we describe in this section was first introduced in [37], the goal is to determine predistortions that needs to be applied to a flux pulse signal so that the qubit receives the flux pulse as intended by the experimenter.

As explained in section (1.3), accurate dynamical control of qubit frequency is of key importance to realize single- and two-qubit gates. One of the on-chip control variable that is used on QunatumWare chip is the magneteic flux through a SQUID loop, the signal for magnetic flux control originates from an arbitrary waveform generator (AWG) which operates at room temperature.

As the signal propagates through various electrical components along the control line leading to the quantum device it undergoes linear dynamical distortions. If not properly compensated, these distortions can degrade gate performance, impacting experiments fidelity and repeatability.

In [37] is proposed a technique to characterize flux-pulse distortions induced by components inside the dilution refrigerator by directly measuring the qubit state. In this protocol we send the qubit a pulse sequence where a flux pulse of varying duration τ is embedded between two $\frac{\pi}{2}$ pulses which are always separated by a fixed interval T_{sep} .

The first $\frac{\pi}{2}$ pulse rotates the qubit of $\frac{\pi}{2}$ around the y axis of the Bloch sphere changing its state from $|0\rangle$ to $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$.

When a flux pulse $\Phi_{Q,\tau}(t)$ of duration τ is sent to the qubit¹ after the first $\frac{\pi}{2}$ pulse, the qubits evolve to the state $\frac{|0\rangle+e^{i\varphi\tau}|1\rangle}{\sqrt{2}}$ with relative quantum phase

$$\frac{\varphi_\tau}{2\pi} = \int_0^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau(t)}) dt = \int_0^\tau \Delta f_Q(\Phi_{Q,\tau(t)}) dt + \int_\tau^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau(t)}) dt \quad (4.1)$$

where in the second step we separated the contributions from flux response up to τ and the turn-off transient.

The experiment is then completed with a $\frac{\pi}{2}$ rotation aroud the y - or x -axis of the Bloch sphere to measure respectively the $\langle X \rangle$ or $\langle Y \rangle$ components of the Bloch vector when applying the measurement

¹To send a $\Phi_{Q,\tau}(t)$ flux pulse we are actually sending a $V_{in,\tau}(t)$ voltage pulse through the electronics

gate MZ . From the measurement of $\langle X \rangle$ and $\langle Y \rangle$ we can extract the relative phase ϕ_τ .

Then we can estimate $\Phi_Q(t)$ in the interval $[\tau, \tau + \Delta\tau]$ as follows. From the measurement of $\phi_{\tau+\Delta\tau}$ and ϕ_τ we can compute $\overline{\Delta f_R}$:

$$\overline{\Delta f_R} = \frac{\phi_{\tau+\Delta\tau} - \phi_\tau}{2\pi\Delta\tau} \quad (4.2)$$

$$= \frac{1}{\Delta\tau} \left(\int_0^{\tau+\Delta\tau} \Delta f_Q(\Phi_{Q,\tau+\Delta\tau}(t)) dt + \int_{\tau+\Delta\tau}^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau+\Delta\tau}(t)) dt \right) \quad (4.3)$$

$$- \frac{1}{\Delta\tau} \left(\int_0^\tau \Delta f_Q(\Phi_{Q,\tau}(t)) dt - \int_\tau^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau}(t)) dt \right) \quad (4.4)$$

$$= \frac{1}{\Delta\tau} \left(\int_\tau^{\tau+\Delta\tau} \Delta f_Q(\Phi_{Q,\tau+\Delta\tau}(t)) dt + \int_{\tau+\Delta\tau}^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau+\Delta\tau}(t)) dt - \int_\tau^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau}(t)) dt \right) \quad (4.5)$$

$$= \frac{1}{\Delta\tau} \int_\tau^{\tau+\Delta\tau} \Delta f_Q(\Phi_{Q,\tau+\Delta\tau}(t)) dt + \varepsilon \quad (4.6)$$

with

$$\varepsilon = \frac{1}{\Delta\tau} \left(\int_{\tau+\Delta\tau}^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau+\Delta\tau}(t)) dt - \int_\tau^{T_{sep}} \Delta f_Q(\Phi_{Q,\tau}(t)) dt \right)$$

The phase contribution from the turn-off transients is minimal due to the sharp return to the first-order flux-insensitive sweet spot of the nearly quadratic $\Delta f_Q(\Phi_Q)$; numerical simulations suggest that $|\varepsilon|/\Delta f_R$ remains within the range of approximately 10^{-2} to 10^{-3} for typical dynamical distortions in commonly used electronic components[38][39], for this reason it will be neglected.

Then we can obtain the reconstructed flux pulse $\Phi_R(t)$ inverting eq. (1.32).

Pulse reconstruction

Corrections study

4.2.2 Corrected pulse

4.2.3 Filter determination

IIR corrections

FIR corrections

Una volta trovati i feedback e feedforward taps per gli IIR filters, è necessario trovare gli FIR per correggere il segnale su scale di tempo più brevi.

for description and notes on CMA-ES see section (3.2)

Output filters in QM

Each analog output port of the OPX system used in this work is equipped with a digital filter that is applied to the signal in the digital domain before conversion to analog. The filters are provided to the OPX via the `parameters.json` file, which contains the coefficients for the feedforward and feedback components according to the equation

$$y[n] = \sum_{m=1}^M a_m y[n-m] + \sum_{k=0}^K b_k x[n-k], \quad (4.7)$$

where $y[n]$ is the output signal, $x[n]$ is the input waveform, a_m are the feedback coefficients, and b_k are the feedforward coefficients.

In our case, we have a set of feedback coefficients determined through IIR correction and two sets of feedforward coefficients: the first obtained from the IIR-based correction, and the second from FIR-based correction on short timescales. To uniquely determine the coefficients to be passed to the electronics, it is necessary to derive a single set of feedforward coefficients and a single set of feedback coefficients by combining the two sets of feedforward filters through convolution.

Infatti posso considerare il segnale di input x a cui applico un primo filtro IIR ottenendo così un segnale in output y tale che

$$y[n] = \sum_{m=1}^M a[m]y[n-m] + \sum_{k=0}^N b[k]x[n-k]. \quad (4.8)$$

al segnale y applico un secondo filtro IIR ottenendo quindi un segnale z in output con la sgeuente forma

$$z[n] = \sum_{m=1}^M a'[m]z[n-m] + \sum_{k=0}^N b'[k]y[n-k]. \quad (4.9)$$

Now I consider equation (4.8) and rewrite it as

$$y[n] - \sum_{m=1}^M a[m]y[n-m] = \sum_{k=0}^N b[k]x[n-k], \quad (4.10)$$

then by applying a Z-transform we obtain

$$Y(z) \left(1 - \sum_{m=1}^M a[m]z^{-m} \right) = X(z) \left(\sum_{k=0}^N b[k]z^{-k} \right) \quad (4.11)$$

so that

$$H_1(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^N b[k]z^{-k}}{1 - \sum_{m=1}^M a[m]z^{-m}} = \frac{B(z)}{A(z)} \quad (4.12)$$

$$\rightarrow Y(z) = H_1(z)X(z) = \frac{B(z)}{A(z)}X(z) \quad (4.13)$$

$$(4.14)$$

We can do the same also for equation (4.9) and rewrite it as

$$z[n] = \sum_{m=1}^M a'[m]z[n-m] + \sum_{k=0}^N b'[k]y[n-k], \quad (4.15)$$

which, by applying the Z-transform becomes

$$Z(z) \left(1 - \sum_{m=1}^M a'[m]z^{-m} \right) = Y(z) \left(\sum_{k=0}^N b'[k]z^{-k} \right). \quad (4.16)$$

Again we can write the the transfer function

$$H_2(z) = \frac{Z(z)}{Y(z)} = \frac{\sum_{k=0}^N b'[k]z^{-k}}{1 - \sum_{m=1}^M a'[m]z^{-m}} = \frac{B'(z)}{A'(z)} \quad (4.17)$$

$$\rightarrow Z(z) = H_2(z)Y(z) = \frac{B'(z)}{A'(z)}Y(z) = \frac{B'(z)}{A'(z)} \frac{B(z)}{A(z)}X(z) \quad (4.18)$$

$$= \left(\frac{\sum_{k=0}^N b'[k]z^{-k}}{1 - \sum_{m=1}^M a'[m]z^{-m}} \right) \left(\frac{\sum_{k=0}^N b[k]z^{-k}}{1 - \sum_{m=1}^M a[m]z^{-m}} \right) X(z) \quad (4.19)$$

$$(4.20)$$

From the transfer function we can obtain the expression for $Z(z)$ in terms of $X(z)$

$$\rightarrow Z(z) \left(1 - \sum_{m=1}^M a'[m]z^{-m} \right) \left(1 - \sum_{m=1}^M a[m]z^{-m} \right) = \left(\sum_{k=0}^N b'[k]z^{-k} \right) \left(\sum_{k=0}^N b[k]z^{-k} \right) X(z) \quad (4.21)$$

$$\rightarrow Z(z) \left(\sum_{m=0}^M a'[m]z^{-m} \right) \left(\sum_{m=0}^M a[m]z^{-m} \right) = \left(\sum_{k=0}^N b'[k]z^{-k} \right) \left(\sum_{k=0}^N b[k]z^{-k} \right) X(z) \quad (4.22)$$

$$(4.23)$$

where in the last step ho assorbito l'1 in the sum come coeffiente a_0 . By expanding the products we obtain

$$\left(\sum_{m=0}^M a'[m] \right) \left(\sum_{m=0}^M a[m] \right) = \sum_{m=0}^{2M} \sum_{i=0}^m a'[i]a[m-i] = c[k], \quad \text{with } m = 0, \dots, 2M, \quad (4.24)$$

$$\left(\sum_{k=0}^N b'[k] \right) \left(\sum_{k=0}^N b[k] \right) = \sum_{k=0}^{2N} \sum_{i=0}^k b[i]b[k-i] = d[k], \quad \text{with } k = 0, \dots, 2N. \quad (4.25)$$

$$(4.26)$$

It is then possible to re-write equation (4.23) using the new expression for the filters

$$Z(z) \left(\sum_{m=0}^{2M} c[m]z^{-m} \right) = \left(\sum_{k=0}^{2N} d[k]z^{-k} \right) X(z) \quad (4.27)$$

$$\rightarrow Z(z) \left(1 - \sum_{m=1}^{2M} c[m]z^{-m} \right) = \left(\sum_{k=0}^{2N} d[k]z^{-k} \right) X(z) \quad (4.28)$$

$$(4.29)$$

then we apply the inverse-Z-transform and obtain

$$z[n] - \sum_{m=1}^{2M} c[m]z[n-m] = \sum_{k=0}^{2N} d[k]x[n-k] \quad (4.30)$$

$$z[n] = \sum_{m=1}^{2M} c[m]z[n-m] + \sum_{k=0}^{2N} d[k]x[n-k] \quad (4.31)$$

$$(4.32)$$

where the feedforward (feedback) taps of the final filters, are given by the convolution of the feedforward (feedback) taps of the two filters as shown in equations (4.25) and (4.26).

4.2.4 Dimostrazione del conto

In general, for different forms of the detuning flux $\Delta f(\Phi) = a\Phi^k$, where $k \in \mathbb{Z}^+$, the phase φ_τ expressed in terms of the impulse response $h = \frac{ds}{dt}$ is the following,

$$\varphi_\tau = 2\pi a \int_0^\infty \left[\int_0^\infty h(t-t')dt' - \int_0^\infty h(t-\tau-t')dt' \right]^k dt \quad (4.33)$$

$$= 2\pi a \int_0^\tau \left[\int_0^t h(t-t')dt' \right]^k dt + 2\pi a \int_\tau^\infty \left[\int_0^\tau h(t-t')dt' \right]^k dt, \quad (4.34)$$

The demonstration of this equality is reported in Appendix B.

Chapter 5

Conclusions

Bibliography

- [1] Richard P. Feynman. “Simulating physics with computers”. In: *International Journal of Theoretical Physics* 21.6 (June 1982), pp. 467–488. ISSN: 1572-9575. DOI: 10.1007/BF02650179. URL: <https://doi.org/10.1007/BF02650179>.
- [2] Katherine L. Brown, William J. Munro, and Vivien M. Kendon. *Using Quantum Computers for Quantum Simulation*. 2010. DOI: 10.3390/e12112268. URL: <https://doi.org/10.3390/e12112268>.
- [3] I. M. Georgescu, S. Ashhab, and Franco Nori. “Quantum simulation”. In: *Reviews of Modern Physics* 86.1 (Mar. 2014), pp. 153–185. ISSN: 1539-0756. DOI: 10.1103/revmodphys.86.153. URL: <http://dx.doi.org/10.1103/RevModPhys.86.153>.
- [4] Ashley Montanaro. “Quantum algorithms: an overview”. In: *npj Quantum Information* 2.1 (Jan. 2016), p. 15023. ISSN: 2056-6387. DOI: 10.1038/npjqi.2015.23. URL: <https://doi.org/10.1038/npjqi.2015.23>.
- [5] Z. Chen. “Metrology of Quantum Control and Measurement in Superconducting Qubits”. ProQuest ID: Chen_ucsbs_0035D_13771, Merritt ID: ark:/13030/m50p5xxf. PhD thesis. UC Santa Barbara, 2018. URL: <https://escholarship.org/uc/item/0g29b4p0>.
- [6] David P. DiVincenzo. “The Physical Implementation of Quantum Computation”. In: *Fortschritte der Physik* 48.9–11 (Sept. 2000), pp. 771–783. ISSN: 1521-3978. DOI: 10.1002/1521-3978(200009)48:9/11<771::aid-prop771>3.0.co;2-e. URL: [http://dx.doi.org/10.1002/1521-3978\(200009\)48:9/11%3C771::AID-PROP771%3E3.0.CO;2-E](http://dx.doi.org/10.1002/1521-3978(200009)48:9/11%3C771::AID-PROP771%3E3.0.CO;2-E).
- [7] Riccardo Manenti and Mario Motta. *Quantum Information Science*. en. 1st ed. Oxford University PressOxford, Aug. 2023. ISBN: 978-0-19-878748-8 978-0-19-182959-8. DOI: 10.1093/oso/9780198787488.001.0001. URL: <https://academic.oup.com/book/51893> (visited on 02/24/2025).
- [8] S. Olivares. *Lecture Notes on Quantum Computing*. Nov. 2021.
- [9] Jens Koch et al. “Charge insensitive qubit design derived from the Cooper pair box”. en. In: *Physical Review A* 76.4 (Oct. 2007). arXiv:cond-mat/0703002, p. 042319. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.76.042319. URL: <http://arxiv.org/abs/cond-mat/0703002> (visited on 02/24/2025).
- [10] D. Vion et al. “Manipulating the quantum state of an electrical circuit”. In: *Science* 296.5569 (2002), pp. 886–889. DOI: 10.1126/science.1069372.
- [11] Thomas E. Roth, Ruichao Ma, and Weng C. Chew. “The Transmon Qubit for Electromagnetics Engineers: An introduction”. In: *IEEE Antennas and Propagation Magazine* 65.2 (Apr. 2023), pp. 8–20. ISSN: 1558-4143. DOI: 10.1109/map.2022.3176593. URL: <http://dx.doi.org/10.1109/MAP.2022.3176593>.
- [12] Ben Chiaro and Yaxing Zhang. *Active Leakage Cancellation in Single Qubit Gates*. 2025. arXiv: 2503.14731 [quant-ph]. URL: <https://arxiv.org/abs/2503.14731>.
- [13] Philip Krantz et al. “A Quantum Engineer’s Guide to Superconducting Qubits”. en. In: *Applied Physics Reviews* 6.2 (June 2019). arXiv:1904.06560 [quant-ph], p. 021318. ISSN: 1931-9401. DOI: 10.1063/1.5089550. URL: <http://arxiv.org/abs/1904.06560> (visited on 02/24/2025).
- [14] QuantWare Contralto-D. URL: <https://www.quantware.com/product/contralto>.
- [15] XLDsl dilution refrigerator, BLUEFORS. URL: <https://bluefors.com/products/4k-systems/xldsl-4k-measurement-system/>.
- [16] opx1000 hybrid control platform, QUANTUM MACHINES. URL: <https://www.quantum-machines.co/products/oxp1000/>.

- [17] Andrea Pasquale et al. *Qibocal: an open-source framework for calibration of self-hosted quantum devices*. en. arXiv:2410.00101 [quant-ph]. Oct. 2024. DOI: 10.48550/arXiv.2410.00101. URL: <http://arxiv.org/abs/2410.00101> (visited on 02/24/2025).
- [18] *Qibocal website*. URL: <https://qibo.science/qibocal/stable/>.
- [19] *Qibocal source code, GitHub*. URL: <https://github.com/qiboteam/qibocal>.
- [20] Stavros Efthymiou et al. “Qibolab: an open-source hybrid quantum operating system”. en. In: *Quantum* 8 (Feb. 2024). arXiv:2308.06313 [quant-ph], p. 1247. ISSN: 2521-327X. DOI: 10.22331/q-2024-02-12-1247. URL: <http://arxiv.org/abs/2308.06313> (visited on 02/24/2025).
- [21] *Qibolab website*. URL: <https://qibo.science/qioblab/stable/>.
- [22] *Qibolab source code, GitHub*. URL: <https://github.com/qiboteam/qibolab>.
- [23] F. Motzoi et al. “Simple Pulses for Elimination of Leakage in Weakly Nonlinear Qubits”. In: *Physical Review Letters* 103.11 (Sept. 2009). ISSN: 1079-7114. DOI: 10.1103/physrevlett.103.110501. URL: <http://dx.doi.org/10.1103/PhysRevLett.103.110501>.
- [24] J. M. Gambetta et al. “Analytic control methods for high-fidelity unitary operations in a weakly nonlinear oscillator”. In: *Physical Review A* 83.1 (Jan. 2011). ISSN: 1094-1622. DOI: 10.1103/physreva.83.012308. URL: <http://dx.doi.org/10.1103/PhysRevA.83.012308>.
- [25] M. Mohseni, A. T. Rezakhani, and D. A. Lidar. “Quantum-process tomography: Resource analysis of different strategies”. In: *Phys. Rev. A* 77 (3 Mar. 2008), p. 032322. DOI: 10.1103/PhysRevA.77.032322. URL: <https://link.aps.org/doi/10.1103/PhysRevA.77.032322>.
- [26] Antonio Anna Mele. “Introduction to Haar Measure Tools in Quantum Information”. In: *Quantum* 8 (May 2024), p. 1340. ISSN: 2521-327X. DOI: 10.22331/q-2024-05-08-1340. URL: <http://dx.doi.org/10.22331/q-2024-05-08-1340>.
- [27] Joseph Emerson, Robert Alicki, and Karol Życzkowski. “Scalable noise estimation with random unitary operators”. In: *Journal of Optics B: Quantum and Semiclassical Optics* 7.10 (Sept. 2005), S347. DOI: 10.1088/1464-4266/7/10/021. URL: <https://dx.doi.org/10.1088/1464-4266/7/10/021>.
- [28] E. Knill et al. “Randomized Benchmarking of Quantum Gates”. en. In: *Physical Review A* 77.1 (Jan. 2008). arXiv:0707.0963 [quant-ph], p. 012307. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.77.012307. URL: <http://arxiv.org/abs/0707.0963> (visited on 02/24/2025).
- [29] Pauli Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.
- [30] J. Kelly et al. “Optimal Quantum Control Using Randomized Benchmarking”. en. In: *Physical Review Letters* 112.24 (June 2014), p. 240504. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.112.240504. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.112.240504> (visited on 02/24/2025).
- [31] J. A. Nelder and R. Mead. “A Simplex Method for Function Minimization”. In: *The Computer Journal* 7.4 (Jan. 1965), pp. 308–313. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308. eprint: <https://academic.oup.com/comjnl/article-pdf/7/4/308/1013182/7-4-308.pdf>. URL: <https://doi.org/10.1093/comjnl/7.4.308>.
- [32] Masahiro Nomura and Masashi Shibata. *cmaes : A Simple yet Practical Python Library for CMA-ES*. 2024. arXiv: 2402.01373 [cs.NE]. URL: <https://arxiv.org/abs/2402.01373>.
- [33] Andrew N. Sloss and Steven Gustafson. *2019 Evolutionary Algorithms Review*. 2019. arXiv: 1906.08870 [cs.NE]. URL: <https://arxiv.org/abs/1906.08870>.
- [34] Takuya Akiba et al. “Optuna: A Next-generation Hyperparameter Optimization Framework”. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2019.
- [35] Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. “Sequential Model-Based Optimization for General Algorithm Configuration”. In: *Learning and Intelligent Optimization*. Ed. by Carlos A. Coello Coello. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 507–523. ISBN: 978-3-642-25566-3.
- [36] Bobak Shahriari et al. “Taking the Human Out of the Loop: A Review of Bayesian Optimization”. In: *Proceedings of the IEEE* 104.1 (2016), pp. 148–175. DOI: 10.1109/JPROC.2015.2494218.

- [37] M. A. Rol et al. “Time-domain characterization and correction of on-chip distortion of control pulses in a quantum processor”. en. In: *Applied Physics Letters* 116.5 (Feb. 2020). arXiv:1907.04818 [quant-ph], p. 054001. ISSN: 0003-6951, 1077-3118. DOI: 10.1063/1.5133894. URL: <http://arxiv.org/abs/1907.04818> (visited on 02/24/2025).
- [38] R. Sagastizabal et al. “Experimental error mitigation via symmetry verification in a variational quantum eigensolver”. In: *Phys. Rev. A* 100 (1 July 2019), p. 010302. DOI: 10.1103/PhysRevA.100.010302. URL: <https://link.aps.org/doi/10.1103/PhysRevA.100.010302>.
- [39] N. K. Langford et al. “Experimentally simulating the dynamics of quantum light and matter at deep-strong coupling”. In: *Nature Communications* 8.1 (Nov. 2017), p. 1715. ISSN: 2041-1723. DOI: 10.1038/s41467-017-01061-x. URL: <https://doi.org/10.1038/s41467-017-01061-x>.

Appendix A

Appendix B

In the following we will demonstrate the correctness of equation (4.33):

$$\varphi_\tau = 2\pi a \int_0^\infty \left[\int_0^\infty h(t-t')dt' - \int_0^\infty h(t-\tau-t')dt' \right]^k dt \quad (5.1)$$

$$= 2\pi a \int_0^\tau \left[\int_0^t h(t-t')dt' \right]^k dt + 2\pi a \int_\tau^\infty \left[\int_0^\tau h(t-t')dt' \right]^k dt, \quad (5.2)$$

for different forms of the detuning flux

$$\Delta f(\Phi) = a\Phi^k \quad (5.3)$$

where $k \in \mathbb{Z}^+$. From the calculations showed in (4.1) we know that the relative phase ϕ_τ for a general form of the detuning flux (5.3), with $T_{\text{sep}} = \infty$ is

$$\varphi_\tau = 2\pi \int_0^{+\infty} a [(s(t) - s(t-\tau))]^k dt = 2\pi a \int_0^{+\infty} [(s(t) - s(t-\tau))]^k dt \quad (5.4)$$

As hypothesis we know that voltage-to-flux step response of the control line is

$$s(t) = \left(1 - e^{-t/\tau}\right) \cdot u(t), \quad (5.5)$$

where $u(t)$ is the step function, and that the impulse response is

$$h(t) = \frac{ds}{dt} \quad (5.6)$$

If we substitute the expression of $s(t)$ given in (5.5) in equation (5.4) we obtain

$$\varphi_\tau = 2\pi a \int_0^{+\infty} \left[\int_0^{+\infty} h(t-t')dt' - \int_0^{+\infty} h(t-\tau-t')dt' \right]^k dt. \quad (5.7)$$

To do we have to show that

1.

$$s(t) = \int_0^{+\infty} h(t-t')dt' \quad (5.8)$$

2.

$$s(t-\tau) = \int_0^{+\infty} h(t-\tau-t')dt' \quad (5.9)$$

We start from the demonstration of equation (5.8). By definition (5.6) we can write

$$h(t) = \frac{d}{dt} \left[\left(1 - e^{-t/\tau}\right) u(t) \right] = \frac{e^{-t/\tau}}{\tau} u(t) + \left(1 - e^{-t/\tau}\right) \delta(t), \quad (5.10)$$

substituting Eq (5.10) in Eq (5.8) we obtain

$$\int_0^{+\infty} h(t-t')dt' = \int_0^{+\infty} \frac{e^{-\frac{(t-t')}{\tau}}}{\tau} u(t-t')dt' + \int_0^{+\infty} \left(1 - e^{-\frac{(t-t')}{\tau}}\right) \delta(t-t')dt', \quad (5.11)$$

by setting $t'' = t - t'$, $dt'' = -dt'$, we have $t'' \rightarrow -\infty$ for $t' \rightarrow +\infty$ and $t'' = t$ for $t' = 0$, the integral then becomes

$$\int_0^{+\infty} h(t - t') dt' = \int_t^{-\infty} -\frac{e^{-t''/\tau}}{\tau} u(t'') dt'' - \int_t^{-\infty} (1 - e^{-t''/\tau}) \delta(t'') dt'' \quad (5.12)$$

$$= \int_{-\infty}^t \frac{e^{-t''/\tau}}{\tau} u(t'') dt'' + \int_{-\infty}^t (1 - e^{-t''/\tau}) \delta(t'') dt'' \quad (5.13)$$

$$= \int_0^t \frac{e^{-t''/\tau}}{\tau} u(t'') dt'' + (1 - e^{-t''/\tau}) \Big|_{t''=0} \quad (5.14)$$

$$= \left[-e^{-t''/\tau} u(t'') \right]_0^t + 0 \quad (5.15)$$

$$= (1 - e^{-t/\tau}) u(t) \quad (5.16)$$

$$(5.17)$$

that concludes the demonstration of Eq (5.8). To demonstrate equation (5.9) we start again by using the definition of $s(t)$ to compute

$$h(t - t' - \tau) = \frac{d}{dt} \left[(1 - e^{-(t-t'-\tau)/\tau}) u(t - t' - \tau) \right] \quad (5.18)$$

$$= \frac{e^{-(t-t'-\tau)/\tau}}{\tau} u(t - t' - \tau) + (1 - e^{-(t-t'-\tau)/\tau}) \delta(t - t' - \tau) \quad (5.19)$$

$$(5.20)$$

We can substitute (5.20) in equation (5.9) and obtain

$$\int_0^{+\infty} h(t - t' - \tau) dt' = \int_0^{+\infty} \frac{e^{-(t-t'-\tau)/\tau}}{\tau} u(t - t' - \tau) dt' + \int_0^{+\infty} (1 - e^{-(t-t'-\tau)/\tau}) \delta(t - t' - \tau) dt' \quad (5.21)$$

by setting $t'' = t - t' - \tau$, $dt'' = -dt'$, we have $t'' \rightarrow -\infty$ for $t' \rightarrow +\infty$ and $t'' = t - \tau$ for $t' = 0$, the integral then becomes

$$\int_0^{+\infty} h(t - t' - \tau) dt' = \int_{-\infty}^{t-\tau} \frac{-e^{-t''/\tau}}{\tau} u(t'') dt'' - \int_{t-\tau}^{\infty} (1 - e^{-t''/\tau}) \delta(t'') dt'' \quad (5.22)$$

$$= \int_{-\infty}^{t-\tau} \frac{e^{-t''/\tau}}{\tau} u(t'') dt'' + \int_{-\infty}^{t-\tau} (1 - e^{-t''/\tau}) \delta(t'') dt'' \quad (5.23)$$

$$= \int_0^{t-\tau} \frac{e^{-t''/\tau}}{\tau} u(t'') dt'' + (1 - e^{-t''/\tau}) \Big|_{t''=0} \quad (5.24)$$

$$= \left[-e^{-t''/\tau} u(t'') \right]_0^{t-\tau} + 0 \quad (5.25)$$

$$= (1 - e^{-t''/\tau}) u(t'') \quad (5.26)$$

$$= s(t'') = s(t - t' - \tau) \quad (5.27)$$

With this, we demonstrated that

$$\varphi_\tau = 2\pi a \int_0^\infty \left[\int_0^\infty h(t - t') dt' - \int_0^\infty h(t - \tau - t') dt' \right]^k dt, \quad (5.28)$$

We now need to show that

$$2\pi a \int_0^\infty \left[\int_0^\infty h(t - t') dt' - \int_0^\infty h(t - \tau - t') dt' \right]^k dt \quad (5.29)$$

$$= 2\pi a \int_0^\tau \left[\int_0^t h(t - t') dt' \right]^k dt + 2\pi a \int_\tau^\infty \left[\int_0^\tau h(t - t') dt' \right]^k dt. \quad (5.30)$$

As first step we can try to rewrite the left-hand-side (LHS) in a different way:

$$2\pi a \int_0^\infty \left[\int_0^\infty h(t-t')dt' - \int_0^\infty h(t-\tau-t')dt' \right]^k dt \quad (5.31)$$

$$= 2\pi a \int_0^\infty [s(t) - s(t-\tau)]^k dt \quad (5.32)$$

$$= 2\pi\alpha \int_0^{+\infty} \left[(1 - e^{-t/\tau}) u(t) - (1 - e^{-(t-\tau)/\tau}) u(t-\tau) \right]^k dt \quad (5.33)$$

$$= 2\pi\alpha \int_0^\tau \left[(1 - e^{-t/\tau}) u(t) \right]^k dt + 2\pi\alpha \int_\tau^{+\infty} \left[(1 - e^{-t/\tau}) u(t) - (1 - e^{-(t-\tau)/\tau}) u(t-\tau) \right]^k dt \quad (5.34)$$

$$= 2\pi\alpha \int_0^\tau \left[\int_0^\infty h(t-t')dt' \right]^k dt + 2\pi\alpha \int_\tau^{+\infty} [s(t) - s(t-\tau)]^k dt \quad (5.35)$$

$$(5.36)$$

In the first step, to get equation (5.33) I simply used the equations (5.8) and (5.9) that were demonstarted before, then by substituting the definition of $s(t)$ we obtain Eq (5.34). It is possible to separate the integral in equation (5.34) by using the definition of $u(t)$ which is null for $t\tau$ and of $u(t-\tau)$ which is null also for $0 < t < \tau$, doing this we obtain Eq, (5.35) By substituting back (5.8) in the first term of equation (5.35) we obtain the first term of equation (5.29) which we want to demonstrate, in the second term of the sum instead, we can substitute back the definition of $s(t)$.

At this point we only have to show that for $t > \tau$, which is the interval we are considering in the second term,

$$2\pi\alpha \int_\tau^{+\infty} [s(t) - s(t-\tau)]^k dt = 2\pi a \int_\tau^\infty \left[\int_0^\tau h(t-t')dt' \right]^k dt. \quad (5.37)$$

To do this we can first evaluate $s(t) - s(t-\tau)$ in the interval $t > \tau$ so that $u(t) = u(t-\tau) = 1$, we have

$$s(t) - s(t-\tau) = 1 - e^{-\frac{t}{\tau}} - 1 + e^{-\frac{(t-\tau)}{\tau}} = e^{-\frac{t}{\tau}} (e^{\frac{\tau}{\tau}} - 1) = (e-1)e^{-\frac{t}{\tau}} \quad (5.38)$$

If we then compute $\int_0^\tau h(t-t')dt'$ we obtain

$$\int_0^\tau h(t-t')dt' = \int_0^\tau \left[\frac{e^{-\frac{(t-t')}{\tau}}}{\tau} u(t-t') - (1 - e^{-\frac{t-t'}{\tau}}) \delta(t-t') \right] dt' \quad (5.39)$$

$$= \int_0^\tau \frac{e^{-\frac{(t-t')}{\tau}}}{\tau} u(t-t')dt' \quad (5.40)$$

$$= \int_0^\tau \frac{e^{-\frac{(t-t')}{\tau}}}{\tau} dt' \quad (5.41)$$

$$= \int_0^\tau \frac{e^{-\frac{t}{\tau}} e^{\frac{t'}{\tau}}}{\tau} = e^{-\frac{t}{\tau}} e^{\frac{t'}{\tau}} \Big|_0^\tau = e^{\frac{-t}{\tau}} (e-1) \quad (5.42)$$

which is equal to (5.38) and then concludes the demonstration.

Appendix C

The average gate fidelity measures how well a noisy quantum gate \mathcal{E} approximates an ideal unitary gate U when averaged over all pure input state. It is defined as

$$F = \int d\psi \langle \psi | U^\dagger \mathcal{E}(\rho) U | \psi \rangle,$$

where the integral is taken over all the pure states $|\psi\rangle$ according to the Haar measure. In randomized benchmarking, we assume that errors behave like a depolarizing channel, meaning that after applying a noisy Clifford gate, the system is left in the correct state with probability $1 - d$ and in a completely mixed state with probability d . The quantum maps that represent a depolarizing channel is

$$\mathcal{E}(\rho) = (1 - d)U\rho U^\dagger + d\frac{I}{2^n} \quad (5.43)$$

When we substitute it in the average gate fidelity definition we get:

$$\begin{aligned} F &= \int d\psi \langle \psi | U^\dagger \left[(1 - d)U |\psi\rangle \langle \psi| U^\dagger + d\frac{\mathbb{I}}{2^n} \right] U | \psi \rangle \\ &= \int d\psi \langle \psi | U^\dagger [(1 - d)U |\psi\rangle \langle \psi| U^\dagger] U | \psi \rangle + \int d\psi \langle \psi | U^\dagger \left[d\frac{I}{2^n} \right] U | \psi \rangle \\ &= (1 - d) \int d\psi \langle \psi | |\psi\rangle \langle \psi| |\psi\rangle + \frac{d}{2^n} \int d\psi \langle \psi | U^\dagger \mathbb{I} U | \psi \rangle \\ &= (1 - d) + \frac{d}{2^n} \int d\psi \langle \psi | |\psi\rangle \\ &= (1 - d) + \frac{d}{2^n} = 1 - d + \frac{d}{2^n} = 1 - d \left(1 - \frac{1}{2^n} \right) \\ &= 1 - d \frac{2^n - 1}{2^n} \end{aligned}$$

Acknowledgement

In primo luogo non posso che ringraziare il professor Carrazza per avermi consentito di fare questo lavoro, per il supporto che mi ha fornito durante lo svolgimento del lavoro e soprattutto per la fiducia che mi ha accordato a scatola chiusa per poter fare questa tesi. Sicuramente ringrazio i miei correlatori senza i quali non avrei saputo da che parte cominciare, li ringrazio per avermi guidato senza mai impormi delle scelte e per non essersi mai scandalizzati per la mia ignoranza, è stato fondamentale. Ringrazio Alessandro perché soprattutto nelle due settimane passate al TII mi hai aiutata a inserirmi nel progetto, a capire come fosse meglio lavorare e sei sempre stato pronto a darmi consigli per migliorare. Mi ha aiutato molto la tua disponibilità e la tua attenzione al lato umano della ricerca, non immaginavo che avrei potuto trovare anche questo nel mondo della ricerca. Ringrazio Edoardo per essere stato sempre disponibile, soprattutto nel primo periodo quando stavo cercando di capire in quali parti del codice potessi mettere mano (e come mettere mano) non avrei potuto fare altrimenti. Ringrazio anche Andrea per essere stato sempre disponibile non appena ne ho avuto bisogno e per avermi aiutato tantissimo con il cryoscope senza farmi sentire stupida. Ringrazio tutto il gruppo di Qibo, è stato bello sentirsi parte di un progetto e mi avete accolta senza mai farmi sentire a disagio o non all'altezza di quello che dovevo fare. Ringrazio i miei genitori che hanno reso possibile questo percorso, senza opporsi alle scelte che non condividevano o non capivano ma fidandosi della mia capacità di orientarmi. Ringrazio le nonne, la nonna Rina e la nonna Palmira, che sono state una presenza rassicurante e stimolante, con sempre tanto affetto nei miei confronti. Ringrazio anche i nonni, il nonno Giuseppe e il nonno Mario che oggi non sono qui ma dai cui racconti ho imparato tanto e di cui sicuramente porto un pezzo dentro di me. Ci tengo moltissimo a ringraziare mia sorella Anna, la mia cincips, che mi ha sempre sostenuto, spronato ad andare avanti e spinta a continuare anche quando non sono stata bene, anche quando le cose sembravano non andare e mi ha dato la forza di continuare passo dopo passo. La ringrazio anche per tutte le foto di Wendy e Mila che mi ha mandato quando io non ero a casa ma lei sì e che mi hanno sempre strappato un sorriso o una lacrima per la tenerezza. Ringrazio tutti i miei compagni di corso con cui ho condiviso questo percorso fino a questo punto, le nostre strade probabilmente si divideranno ma è stato bello incontrarsi e poter condividere tanti momenti, spero che le nostr strade si incontrino di nuovo. Ringrazio Timo per tutti gli abbracci che mi ha regalato, Paolo per i le risate e il caffè la mattina, perchè cominciare la giornata insieme rende tutto un po' più facile, Ale per la carica di passione e voglia di fare che mi trasmette ogni volta che facciamo una chiacchierata. Ringrazio le mie compagne storiche del gruppo del laboratorio, Rosa e Daniela perchè studiare e confrontarmi con voi è stato incredibilmente arricchente, ha alleggerito le difficoltà, e fatto sembrare anche i momenti meno semplici comunque affrontabili. Ringrazio soprattutto Alessia, in questi cinque anni, ormai sei, in cui ci conosciamo la nostra amicizia si è evoluta moltissimo, sei stata un'ispirazione, con tutta la tua energia, i tuoi progetti, la tua positività e la tua simpatia, sei anche stata una spalla a cui mi sono potuta appoggiare ogni volta che ne ho avuto bisogno, sia per quanto riguarda tutto ciò che è successo nella mia vita in questi cinque anni che per le cose successe in università. Sono molto contenta di poter condividere con te tutti i miei pensieri e le mie idee sapendo che troverò sempre una persona pronta ad ascoltarmi e dirmi con onestà che cosa pensa senza mai giudicarmi. Un ringraziamento speciale è anche per tutte le persone che ho incontrato durante quella *vacanza* mervigliosa che è stato il tirocinio al Fermilab, ognuno in modo diverso mi avete aiutato tutti a crescere. Sicuramente ringrazio Tom per tutti i consigli che mi ha sempre dato, Marco che non ha mai paura di condividere le sue idee, molte in comune altre no ma su cui è sempre stato aperto a discutere, per non parlare della sua propensione alle avventure e al vivere sempre al massimo che ci ha coinvolto e caricati tutti durante i due mesi passati insieme. Ringrazionì moltissimo Sara e Silvia, grazie a voi ho sinceramente scoperto il valore di avere accanto due professioniste incredibili che ti sostengono e ti spronano a fare il tuo meglio, la nostra amicizia per me è speciale. Siete state le mie prime confidenti per qualsiasi problema io abbia avuto durante i due mesi in cui abbiamo vissuto insieme, sietet state una sorpresa fantastica di cui non sapevo nemmeno di avere bisogno. Come ci siamo già dette molte volte, vivere insieme ad altre due sconosciute avrebbe potuto essere difficile e invece l'avete resa la cosa più

bella e naturale che si possa immaginare, non vedo l'ora di poter vedere tutta la strada che percorrerete.