

Università degli studi di Milano Bicocca

Dipartimento di Fisica "Giuseppe Occhialini"

Corso di Laurea magistrale in Fisica



Qubit characterization and calibration on a RFSoC-based control system

The long journey from the hardware setup to a possible quantum sensing application, developing an economic alternative to commercial control devices

Supervisor:
Andrea Giachero

Candidate:
Rodolfo Carobene

External supervisor:
Stefano Carrazza

Student Number:
838092

Co-supervisor:
Matteo Borghesi

External co-supervisor:
Javier Serrano

ACADEMIC YEAR 2022/2023

Abstract

Quantum computing has attracted significant interest from its early beginnings, even when there were no physical implementations of quantum computers yet. The proposal and development of algorithms (such as those by Shor and Grover), that are theoretically more efficient than what can be designed with classical computers, have led to significant progress in this field over the last decade.

These early discoveries have led to an intensification of research and development, aided by the active involvement of private companies, such as Google and IBM, which, within a few years, have invested in new technologies and built computers efficient enough to claim the achievement of a "Quantum Primacy". Although these advancements have yielded noticeable results, current computers are still not precise enough to be practically useful. Indeed, we are in what is referred to as the "Noisy Intermediate-Scale Quantum" (NISQ) era, where quantum computers exist, but are too noisy to support quantum algorithms without encountering errors.

For this reason, after years of research focusing on increasing the number of qubits (the quantum computing equivalent of classical bits), current research is shifting its focus toward designing and building more reliable and efficient individual qubits. In order to achieve this goal, research is working simultaneously in multiple directions: experimenting with various types of qubits (superconducting, trapped ions, etc.), trying to enhance the highly delicate qubit fabrication processes, and improving the methods for qubit control and operation.

My thesis work is part of the research concerning the last point: Quantum Control, specifically for superconducting qubits.

A superconducting qubit can be represented as a LC circuit where the classical inductance is replaced by a non-linear inductance obtained with a Josephson junction. This junction exploit the phenomenon of superconductivity to create a non-harmonic Hamiltonian, allowing the separation of the first two states (those used for computation) from the others. The qubit is then coupled to a classical resonator, described by a harmonic Hamiltonian, which is used for readout. All transition frequencies of this system, which can be precisely studied within the framework of cQED (circuit Quantum ElectroDynamics) [1], are typically on the order of GHz. By applying voltage pulses at these frequencies, it is possible to read the state of the qubit and modify it as desired. However, synthesizing the required pulses is itself a technological challenge, as typical Digital to Analog Converters (DACs) reach at most hundreds of MHz, requiring

up/down-conversion schemes that necessitate the use of additional equipment. This complicates the experimental setup and introduce additional sources of noise into a system that is highly sensitive to external disturbances. All parameters describing the pulses are precisely calibrated to maximize the qubit's performance (relaxation time T_1 , dephasing time T_2 , accuracy), and even minor variations can significantly affect coherence times or lead to leakage outside the computational space. This requires the use of specialized instruments.

Several companies have developed proprietary tools for the synthesis and definition of these pulses. However, these tools are often expensive, and researchers are frequently constrained in signal quality due to typical up/down-conversion schemes. Additionally, each instrument is controlled using different software and programming languages, which complicates integration into a unified control and readout framework.

A recent, more versatile, and cost-effective alternative is the use of RFSoC boards (Radio Frequency System on Chip). These boards are a specific type of FPGA (Field Programmable Gate Arrays) composed of programmable integrated circuits, all connected on a single board, which also incorporates several high-performance RF components. The most notable feature of RFSoC boards is their ability to synthesize GHz pulses with great precision and full control [2], achieved through a technique known as Direct Digital Synthesis (DDS).

RFSoC boards, initially developed for communications purposes, have quickly become appealing also for quantum computing due to their synthesis capabilities and the inclusion of a single electronic board that houses a CPU, multiple high-performance DACs, and ADCs. Additionally, since 2020, the Fermilab's **Qick** project [3] has provided open-source firmware, enabling the programming of these FPGAs for qubit control and readout purposes.

In my thesis work, I utilized **Qick** as a foundation to develop **Qibosoq**, an open-source software designed for the control and readout of superconducting qubits using RFSoC boards. Additionally, **Qibosoq** integrates RFSoC boards into the **Qibo** framework, a high-level software platform that allows for the definition of quantum algorithms and circuits to be executed on hardware later on.

During my thesis period at the Technology Innovation Institute (TII) in Abu Dhabi, I focused on the characterization and calibration of qubits, by using **Qibosoq** in conjunction with various RFSoC boards for the control of single fixed-frequency qubits (physically implemented with a single Josephson junction in a 3D cavity) and for the simultaneous control of multiple variable-frequency qubits (constructed with pairs of Josephson junctions coupled to planar resonators).

The characterization of qubits is a process that involves various experiments aimed at identifying all the key parameters specific to the qubit itself and to the resonant cavity to which it is coupled for readout. These parameters are then used along with those obtained from additional calibration experiments to optimize the control of the qubit, making it suitable for applications in quantum computing or quantum sensing. The sequence of the experiments required is typically composed of at least fifteen different experiments. Each of these experiments involves the synthesis of specific pulse sequences

and measurements aimed at optimizing one or more of the parameters that describe the system.

A system based on RFSoC boards, such as the one developed for this thesis, which does not require additional instruments beyond the board itself, can have a significant impact in simplifying the characterization and calibration operations.

Furthermore, it's important to note that once a qubit control system is developed, it can be used for a wide range of experiments beyond just calibration and characterization.

The work at TII has always been focused on quantum computing, meaning the use of qubits for the execution of quantum algorithms. In this context, the developed system has already been used to run quantum machine learning algorithms aimed at estimating Parton Distribution Functions (PDFs) on qubits [4]. However, in the future, it will also become an important tool for applications in quantum sensing, using qubits as particle detectors.

In particular, in 2021 the italian INFN (*Istituto Nazionale di Fisica Nucleare*) presented the Qub-IT project, which aims to use qubits for the search for dark matter, specifically for dark photons and axions [5]. The fundamental idea behind Qub-IT is to couple a qubit to a cavity that, in resonance with potential dark matter particles, could induce the conversion of these particles into photons. The various photons present in the cavity influence the coupled qubit, making it possible to measure the number of photons with high precision through a specific pulse sequence not too dissimilar from the one required in the previously mentioned Ramsey experiment. In the project proposal document, one of the key points is the development of a qubit control and readout system using FPGAs that is precise, reliable, and scalable. The work of my thesis represents an important step in this direction.

The thesis provides the theoretical elements of cQED necessary to understand how the control of superconducting qubits works. It illustrates the experimental setup and hardware configuration required for the various RFSoC boards used, along with a detailed explanation of the unique features of RFSoC boards, including the synthesis and acquisition methods used to work directly in the GHz range. Additionally, the software **Qibosoq** and **Qibolab**, developed and used during the thesis work, are briefly introduced and commented upon.

The core of the thesis consists of a detailed explanation of the main experiments required for the characterization and calibration of single qubits and two-qubit gates. Each experiment includes graphs to be obtained under ideal conditions, some real-world graphs, and solutions to some of the most common experimental problems. This section is designed to potentially serve as a practical guide for the calibration and characterization of superconducting qubits.

Finally, the thesis concludes with two examples of possible applications of the developed control and readout system: the first in quantum machine learning (already tested on hardware), and the second in quantum sensing, related to the experiment for the search for dark matter.

Contents

Abstract	i
List of figures	ix
List of tables	ix
1 Theory	1
1.1 Dark matter search	1
1.1.1 Observational evidences	1
1.1.2 DM candidates	5
1.2 Qubits	8
1.2.1 Josephson junction	9
1.2.2 Transmon qubits	12
1.3 Transmon interactions	14
1.3.1 Control and readout	17
2 Experimental setup and software	19
2.1 Experimental setup	19
2.1.1 Cryostat	19
2.1.2 FPGA & RFSoC	21
2.1.3 Qubits Configuration	32
2.1.4 Configuration and setup for RFSoC4x2	32
2.1.5 Configuration and setup for ZCU111	34
2.1.6 Configuration and setup for ZCU216	35
2.2 Developed Control Software	36
2.2.1 Qibolab	37
2.2.2 Qibosoq	40
3 Single qubit characterization and calibration	43
3.1 Time of flight measurement	47
3.2 Resonator spectroscopy (bare frequency)	50
3.3 Resonator punchout	54
3.4 Flux resonator spectroscopy	57
3.5 Qubit spectroscopy	59

3.6	Flux Qubit Spectroscopy	63
3.7	Rabi oscillation	66
3.8	Ramsey experiment	69
3.9	Flipping	74
3.10	T1 measurement	75
3.11	Hahn’s spin echo (T2)	76
3.12	Single-shot classification	78
3.13	Readout optimization	81
3.14	AllXY	83
3.15	DRAG tuning (pulse shape)	85
3.16	Randomize benchmarking	86
3.17	Other possible calibration experiments	88
4	Two-qubits characterization and calibration	91
4.1	Avoided crossings	92
4.2	Chevron plot	94
4.3	Flux pulse shape correction	96
4.4	Virtual phase correction	98
5	Applications	101
5.1	Quantum machine learning: determining probability density functions . .	101
5.2	Quantum sensing applications	103
6	Conclusions	107
	Bibliography	111

List of Figures

1.1	Flat rotation curves	2
1.2	Graphical depiction of gravitational lensing	3
1.3	Dark Matter quantity expected from observations and theory	4
1.4	Dark Matter candidates	6
1.5	Highlight of the main qubit technologies and companies involved in the field. Credits [14].	8
1.6	Schematic drawing of a Josephson junction	9
1.7	Scheme of a Cooper pair box	11
1.8	Schematic layout of a SQUID.	12
1.9	Comparison of qubit Hamiltonian eigenvalues for different ratios of E_J/E_C	13
1.10	Basic scheme for the complete control of a single flux-tunable transmon.	15
1.11	Basic circuit scheme for the complete control of a single flux-tunable transmon.	15
2.1	Phase diagram for He as a function of ^3He concentration.	20
2.2	Outline of the dilution process.	21
2.3	Bluefors XLD.	22
2.4	A simple Direct-Digital-Synthesizer, where the function stored in PROM is a sine wave. Credits [29].	23
2.5	Details of the phase accumulator component of a DDS system	24
2.6	Scheme of a generic NCO DDS system	25
2.7	Attenuation-frequency relationship at different Nyquist zones, for different reconstruction waveforms	26
2.8	Picture of a RFSoC4x2 board.	29
2.9	Picture of a ZCU111 board.	30
2.10	Picture of a XM500 board.	31
2.11	Example of a single qubit in a 3D cavity	32
2.12	Topology of the Contralto QuatWare chip	33
2.13	Schematics of the lines of a single non-flux-tunable qubit in a 3D cavity.	33
2.14	Schematics of the lines of a single flux-tunable qubit in the 25 qubit chip.	34
2.15	Scheme used for down and up-conversion.	35
2.16	Schematic overview of Qibo software components.	36
2.17	Schematic overview of Qibolab supported instruments.	38

3.1	Scheme of the characterization and calibration experiments as presented for Qibocal auto-calibration module.	46
3.2	Plot of a generic time of flight experiment.	48
3.3	Plot of a time of flight experiment.	49
3.4	Plot of a time of flight experiment. the vertical line is merely the effect of a badly implemented fitting procedure.	49
3.5	Plot of a generic resonator spectroscopy experiment.	50
3.6	Resonator spectroscopy, for a 3D cavity.	52
3.7	Resonator spectroscopy, for multiple multiplexed 2D resonators with a local oscillator for upconversion.	53
3.8	Expected plot for a resonator punchout experiment. From this we could choose an amplitude of 0.3 at ≈ 480 MHz.	55
3.9	Different plots for the resonator punchout experiment.	56
3.10	Resonator spectroscopy of a 3D cavity in the low power regime.	57
3.11	Expected plot for a flux resonator spectroscopy experiment.	58
3.12	Plot a the flux resonator spectroscopy experiment.	59
3.13	Sketch of the plot expected for a qubit spectroscopy experiment.	60
3.14	Example of a qubit spectroscopy experiment with clear aliasing.	61
3.15	Example of a qubit spectroscopy experiment.	62
3.16	Example of a qubit spectroscopy experiment with zero relaxation.	62
3.17	Sketch of the plot expected for a flux qubit spectroscopy experiment with comparison between a "bad" and "good" step for the bias scan.	64
3.18	Plots for flux qubit spectroscopies.	65
3.19	Expected behaviour in Rabi (length) oscillation.	67
3.20	Possible behaviour in Rabi oscillation, when plotting amplitudes.	67
3.21	Plot of a Rabi-amplitude-length oscillations experiment. On the x-axis we have "pulse length", on the y-axis "pulse amplitude".	68
3.22	Plot of a Rabi-length oscillations experiment.	68
3.23	Plot of a Rabi-amplitude oscillations experiment.	69
3.24	Schematic representation of a Ramsey experiment. Credits [62].	71
3.25	Sketch of the plot of a Ramsey experiment in ideal conditions.	72
3.26	Plot for a Ramsey experiment with the presence of a spurious coupling.	72
3.27	Plot for a Ramsey experiment at the wrong drive frequency.	73
3.28	Plot of a flipping experiment.	75
3.29	Plot of a T1 experiment.	76
3.30	Plot of a Hahn's Spin Echo experiment.	78
3.31	Ideal plot for a single-shot classification experiment.	79
3.32	At the top, a possible bad plot in a single-shot classification experiment. At the bottom, a good one.	80
3.33	Plot of a dispersive shift experiment.	82
3.34	Ideal plot for a allXY experiment.	84
3.35	Plot of a allXY experiment.	84
3.36	Comparison between a Gaussian and a DRAG shapes [70].	85

3.37	Sketch of the plot of a standard randomized benchmarking experiment	87
3.38	Plot of a standard randomized benchmarking experiment executed with a RFSoC4x2	88
4.1	Possible behaviour of two qubits flux dependence.	93
4.2	Avoided crossing.	93
4.3	Avoided crossing for CZ and iSWAP.	94
4.4	Large scan where all the 4 interaction biases are visible.	95
4.5	Zoom in on an avoided crossing (iSWAP). The bias is applied only to the high frequency qubit, while the lower is left at the sweetspot. Measuring the low-frequency qubit gives the avoided crossing.	96
4.6	Expected plot for a two-qubits chevron.	97
4.7	Chevron plot for CZ.	97
4.8	Chevron plot for CZ. The distortion, that will be fixed in the next section, causes the asymmetry higher-lower bias. The leakage causes the chevron to not close and to not appear in the high frequency qubit.	98
4.9	Rectangular pulse distortion.	98
4.10	Calibrated virtual phase correction plot.	99
5.1	PDF of the u -quark, fitted with the QML model with a Qibosoq -controller RFSoC.	103
5.2	Hardware setup required of the axion-detection experiment.	105
5.3	Overview of the pulse sequence to be executed for the axion-detection experiment.	106
6.1	Speed benchmark of a Qibosoq system Vs. commercial alternatives	109

List of Tables

2.1	Characteristics of the supported RFSoCs	29
2.2	Supported features and limitations of Qibolab 0.1.0	39
2.3	Comparison of the supported features of Qibolab , Qick and Qibosoq	42

Chapter 1

Theory

In this chapter, the main theory concepts required to follow the experimental work will be provided. This includes a first section on dark matter, since the detection of axions (a dark matter candidate) is one interesting application of qubits, as well as the elements of circuit QED required to fully understand the used working principle of a superconducting qubit device.

1.1 Dark matter search



Picture of some Dark Matter

To add as soon as possible!

The nature of dark matter (DM) is currently one of the most intriguing open questions in physics [6]. It's an active topic of research in astrophysics and cosmology, where it is implied as the accepted solutions for various phenomena, but is also of great interest for particle physicists, since it will inevitably go beyond the standard model.

Its existence, supported by theoretical models and experimental observations, it's widely accepted by the research community.

However, no direct proof has yet be found.

1.1.1 Observational evidences

There have been different evidences of the presence of DM.

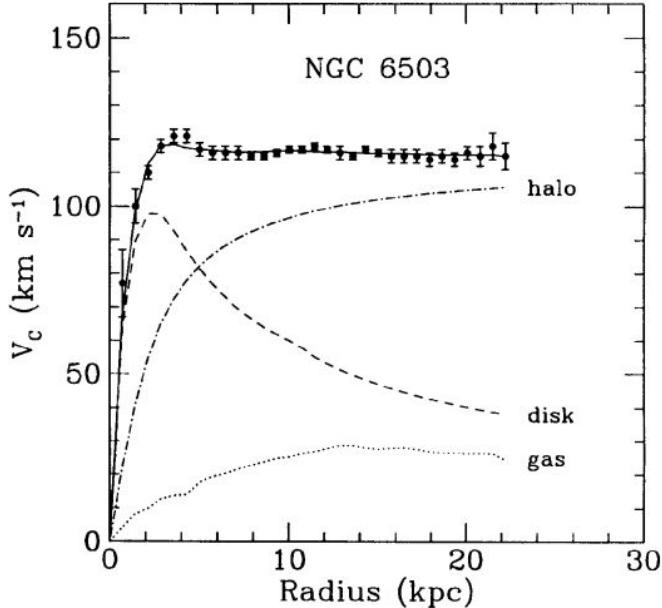


Figure 1.1: Galactic rotation curve for NGC 6503 showing disk and gas contribution plus the dark matter halo contribution needed to match the data. Credits to [8]

Spiral galaxy rotational speed In spiral galaxies, the virial theorem dictates that the velocity of a star at distance R from the center of the galaxy should follow [7]:

$$v(R) = \sqrt{G \frac{M(R)}{R}} \quad (1.1)$$

where $M(R)$ is the total mass contained a sphere with radius R . Far from the center of the galaxy, where practically no mass is left and $M(R)$ is nearly constant, the velocity decreases as $v(R) \propto R^{-1/2}$.

However, experimental measurements of stars around spiral galaxies, performed with extremely precise Doppler shift measurements, shows a different picture. In particular, it was observed that the velocity of stars becomes independent from the radius. The phenomenon, visible for example in the *flat rotation curves* of which fig. 1.1 are typical examples, can be explained with a halo of invisible mass of:

$$M(R) = R \frac{v_0^2}{G} \quad (1.2)$$

Since this mass, that seems to represent up to 80-90% of the total mass of galaxies, appears to not interact in any way except by gravitational force, we identify it and call it as Dark Matter (DM).

Galaxy clusters dynamic A Galaxy cluster is a massive object composed of hundreds or thousands of galaxies bound together. In the intergalactic medium, among galaxies,

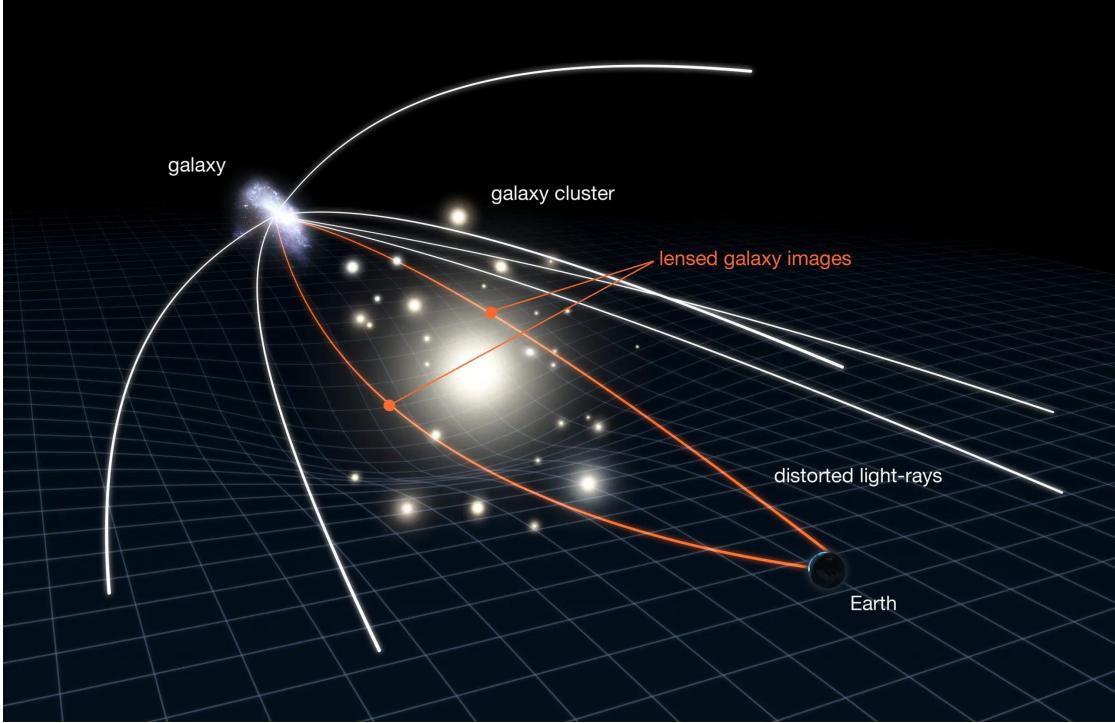


Figure 1.2: This illustration depicts the gravitational lensing phenomenon. The scale has been greatly exaggerated in this diagram. Credits to [9]

it contains large quantity of gases that reach high velocities and can emit x-rays via bremsstrahlung. This effect can be used as a way of measuring the intensity of the gravitational force and, therefore, the distribution of masses in the cluster. The idea is that photons emitted from the center of the cluster should lose more energy than photons emitted from the edge, because of a stronger gravity. This effect, known as *gravitational redshift*, showed that a large portion of mass was not visible and distributed all around the cluster.

This method leads to results with high uncertainties, so it has been superseded by gravitational lensing techniques.

Gravitational lensing The observation of distant galaxies and galaxy clusters is usually affected by the gravitational lensing phenomenon depicted in fig. 1.2. Namely, the light does not follow a straight line, being distorted by any mass between source and observer (the earth) forming the so-called *Einstein circle*.

The radius of an Einstein circle is related to the mass which causes the light deflection following:

$$\theta_E = \sqrt{\frac{4GM}{c^2} \frac{(D_S - D_L)}{D_S D_L}} \quad (1.3)$$

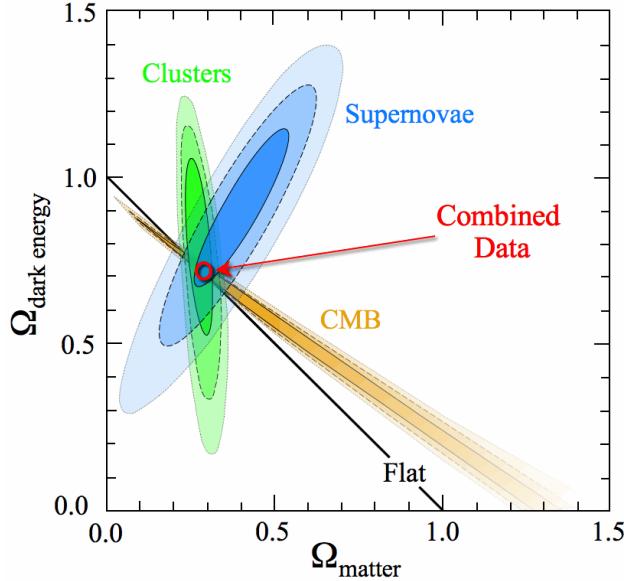


Figure 1.3: Convergence of the experimental observations and of the Λ CDM model. Leading to a a percentage of 95% of non-baryonic energy. Credits to [11]

with θ_E being the angular radius, M the mass of the lens (in between) , D_L the distance to the lens and D_S the distance to the source. This technique to weigh galaxies (and other astrophysical structures) has been used with success for many years. Numerous studies, however, consistently reported a much larger mass measured than seen. Again, this is explainable with presence of DM.

Cosmology The currently most widely accepted cosmological theory requires dark matter to explain the formation of structures in the early universe.

In particular, in the early universe, the energy distribution (so the mass distribution) was more or less uniform. Two opposite processes were involved in the generation of structures: gravitational forces that were favouring structures and the expansion of the universe itself, that was "countering" structures.

In this situation, it is possible to differentiate the role of baryonic and non baryonic matter that is collisionless. Various simulations, and in particular the Millennium Simulation [10], were able to simulate all the early stages of the universe, with the requirement of dark matter and dark energy. Without DM, the current model was not able to explain nor the structures formation nor the CMB spectrum.

The current best theory, the Λ CDM (Lambda cold dark matter) shows a remarkable agreement with all the observations from disparate scales as shown in fig. 1.3.

1.1.2 DM candidates

Now that we saw the many evidences supporting the existence of dark matter, the challenge becomes to determine its nature.

We are looking for a particle (or, for what we know, a set of particles) that is stable over billions of years, collisionless, non-baryonic and interacting mostly gravitationally.

Within the Standard Model (SM), the only neutral non-baryonic particles are the neutrinos that are, however, very light particles. This leads neutrinos to be generally relativistic, constituting an example of fast or "hot" DM. The CMB spectrum, however, cannot be properly explained with only "hot" (relativistic) DM and rather requires the presence of a large quantity of "cold" (non relativistic) particles. Therefore the neutrinos can eventually explain just a minimal portion of the hidden mass. We need to find new particles, beyond SM.

Tens of candidates, of which the main ones are shown in fig. 1.4, have been proposed over the years, but none has been yet successfully discovered. Indeed this is also referred to as the "dark matter candidates zoo", because of the large and never ending number of proposed new particles.

Since this thesis is not directly focused on DM, we will briefly mention the two main candidates: WIMPs (weakly interacting massive particles) and axions.

WIMPs were, until very recently, the most prominent DM candidate. With the name of WIMPs, various particles were described. They were supposed to have been produced thermally in the early universe (as the SM particles) with a mass of ≈ 100 GeV. The introduction of WIMPs was often referred to as the "WIMPs miracle" since these particles are "automatically" introduced with a minimal extension of the SM to a super-symmetric theory. Moreover, WIMPs should have been detectable both indirectly, looking at annihilation products, and directly in collision experiments. No experiment¹ has ever found direct evidence and, lately, the hype for super-symmetric theories has also been decreasing, since no evidence has yet to be found at LHC and other colliders (although it was indeed expected).

Dark Photons [13] introduce an intriguing dimension to the realm of dark matter, akin to WIMPs and axions. These hypothetical particles, often associated with hidden or "dark" forces that operate outside the framework of the Standard Model of particle physics, are captivating for their unique characteristics. Dark photons are postulated as mediators of a novel force, commonly termed the "dark force," which interacts exceptionally weakly with ordinary matter. Just as with their counterparts, dark photons may trace their origins back to the early universe, potentially emerging through mechanisms reminiscent of WIMPs. Dark Photons still are a viable candidate, but require a large theoretical extension of the Standard Model and, being also particularly difficult to detect, are often discarded in comparison to other candidates.

¹With the notable exception of the non-reproducible results of DAMA [12].

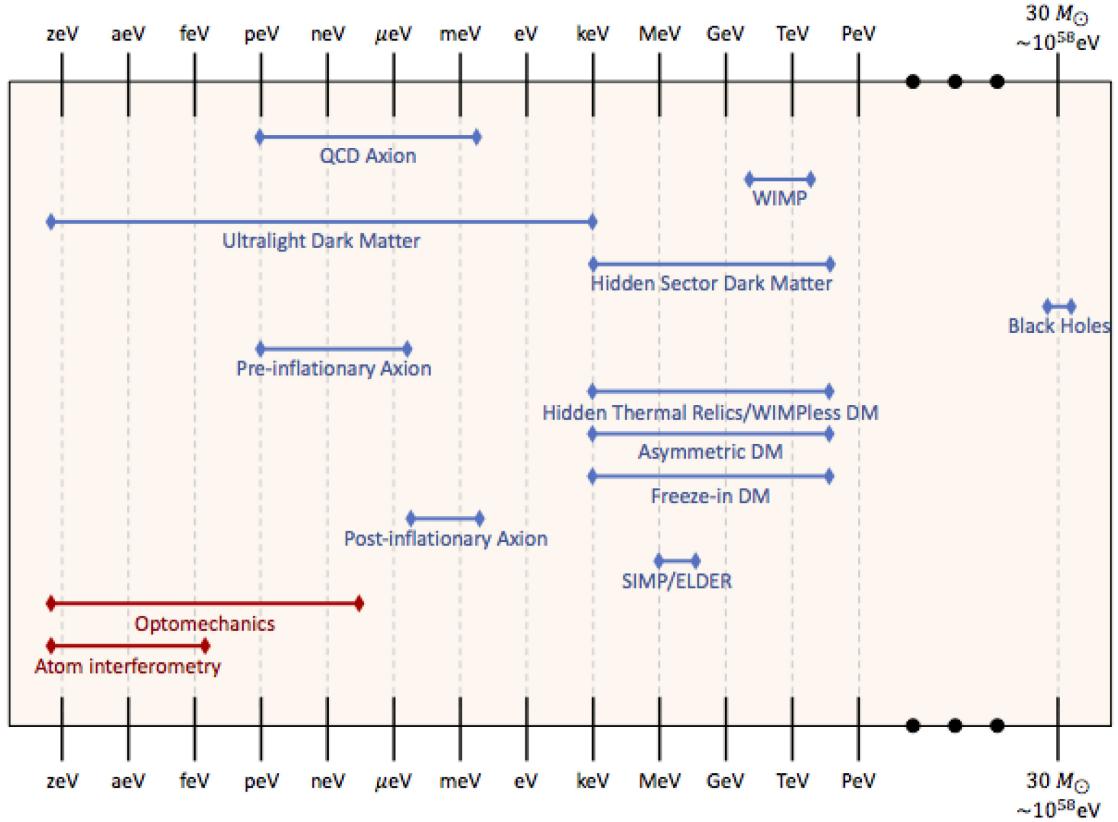


Figure 1.4: DM candidates, in respect to the possible masses. Credits to [7]

Axions The original axion model is related to the strong violation of the charge conjugation (C) and parity (P) symmetries (CP-violation). The QCD Lagrangian can be written as

$$\mathcal{L}_{QCD} = \mathcal{L}_{QCD, \text{perturbative}} + \bar{\theta} \frac{g^2}{32\pi^2} G_{\mu\nu}^a \bar{G}_{\mu\nu}^a \quad (1.4)$$

where $G_{\mu\nu}^a$ is the gluon tensor, $\bar{G}_{\mu\nu}^a$ its dual and $\bar{\theta}$ a constant. The first term is the standard perturbative Lagrangian and the second one is an effective term coming from the topological properties of non-Abelian gauge theories and from the diagonalization of the quark mass matrix. The problem with this term is that it violates the P and CP symmetries and measurements on the neutron electric dipole moment reveal no violation of those symmetries for the strong-force. A solution for this missing violation was proposed by Peccei and Quinn, in which the constant is substituted with a dynamical term coming from a new chiral global symmetry $U(1)$. This new symmetry spontaneously breaks and gives rise to a new scalar boson, called "axion". The symmetry also generates a new term in the Lagrangian, that removes the violation. From theory we have that:

$$m \approx \frac{\Lambda_{QCD}^2}{f_a} \quad (1.5)$$

where m is the expected mass of the new particle (ranging from μeV and eV), $\Lambda_{QCD} \approx 200 \text{ MeV}$ and f_a is a constant with the dimension of an energy, of the order of the symmetry breaking scale.

Axions are an interesting candidate for these reasons: they were not introduced for DM and solve two huge problems of modern physics. They could indeed explain the missing mass of the universe, while at the same time solving the strong CP problem.

1.2 Qubits

A possible way of detecting dark matter, in particular axions, is through the use of superconducting qubits.

A qubit is defined, in quantum computing, as a two-state quantum-mechanical system. One of the simplest systems that can show quantum properties.

A qubit can be built with very different technologies: some examples include qubits composed of polarized photons, ions and cold atoms.

Qubit Type	Pros/Cons	Select Players
Superconducting	Pros: High gate speeds and fidelities. Can leverage standard lithographic processes. Among first qubit modalities so has a head start. Cons: Requires cryogenic cooling; short coherence times; microwave interconnect frequencies still not well understood.	      
	Pros: Extremely high gate fidelities and long coherence times. Extreme cryogenic cooling not required. Ions are perfect and consistent. Cons: Slow gate times/operations and low connectivity between qubits. Lasers hard to align and scale. Ultra-high vacuum required. Ion charges may restrict scalability.	   
Photronics	Pros: Extremely fast gate speeds and promising fidelities. No cryogenics or vacuums required. Small overall footprint. Can leverage existing CMOS fabs. Cons: Noise from photon loss; each program requires its own chip. Photons don't naturally interact so 2Q gate challenges.	  
	Pros: Long coherence times. Atoms are perfect and consistent. Strong connectivity, including more than 2Q. External cryogenics not required. Cons: Requires ultra-high vacuums. Laser scaling challenging.	   
Silicon Spin/Quantum Dots	Pros: Leverages existing semiconductor technology. Strong gate fidelities and speeds. Cons: Requires cryogenics. Only a few entangled gates to date with low coherence times. Interference/cross-talk challenges.	    

Figure 1.5: Highlight of the main qubit technologies and companies involved in the field. Credits [14].

In fig. 1.5 some of the main companies researching in quantum computing are shown, together with indication on the qubit technology used. The preferred qubits in the last years were superconducting-based. In this thesis, these will be the utilized qubits.

Superconducting qubits are often called artificial atoms and the main idea behind them, is to build a circuit with the first two level very distinguishable from the others. So the qubit is obtained by restraining the system.

The superconducting technology has reached very good results [15], but it is still dealing with huge problems also in the hardware and design section.

1.2.1 Josephson junction

To observe quantum behaviour at a macroscopic level, (so at the size of circuit components), a strong degree of coherence is needed. In superconducting qubits this is achieved exploiting the phenomenon of superconductivity that is found in some specific metals at cryogenic temperatures [16].

At these temperatures (i.e. $T \ll T_c$ where T_c is the critical temperature of the material), electrons bound together forming *Cooper pairs* composed of two electrons with equal and opposite momenta [17]. Cooper pairs have a bosonic nature, allowing them to form a condensate, where they are all described by a single quantum ground state, effectively creating a macroscopic quantum phenomenon. To assure that the number of Cooper pairs remains stable and increase coherence times, superconducting qubits are usually operated below 50 mK by exploiting complex cryogenics system called cryostat [18].

The core of a superconducting qubit is the *Josephson junction* [19]: a superconducting inductor that enables the transformation from a harmonic oscillator (standard LC circuit) to an an-harmonic oscillator, needed to separate the 0-1 states and build a qubit. A Josephson junction is composed by a thin insulative gap (order of nm) between two superconductors as presented in fig. 1.6. The small thickness of the insulator enables the Cooper pairs wavefunction in the two superconductors to overlap, creating a coherent tunneling phenomenon where Cooper pairs can "jump" from one superconductor to the other without any applied voltage. The resulting current (often referred to as *supercurrent*) has a nonlinear dependence on the voltage applied over the junction, causing the anharmonicity.

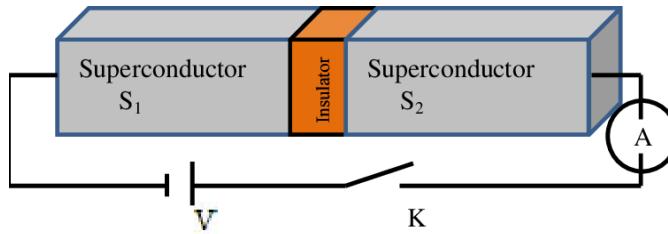


Figure 1.6: Schematic drawing of a Josephson junction. Credits [20].

The Hamiltonian of the Josephson junction can be found from the two equations

under the name of *Josephson effect* [21, 22]:

$$I = I_c \sin \phi \quad (1.6)$$

$$\frac{\partial \phi}{\partial t} = \frac{2e}{\hbar} V \quad (1.7)$$

where I is the supercurrent flowing through the junction, $I_c = 2eE_J/\hbar$ is the critical current over which the junction starts to exhibit dissipation, V is the voltage applied over the junction and ϕ , called Josephson phase, is the phase difference between the wavefunction of the two superconductors. E_J , called Josephson energy, is the energy associated with a single Cooper pair that tunnels through the junction and can be computed as $E_J = L_J I_c^2$ with $L_J = \hbar/2eI_c$ being the inductance of the junction.

The relations expressed in eq. (1.6) and eq. (1.7) can be used to find the I-V relation:

$$\frac{\partial I}{\partial t} = \frac{\partial I}{\partial \phi} \frac{\partial \phi}{\partial t} = I_c \frac{2e}{\hbar} V \cos \phi \quad (1.8)$$

That gives an effective inductance term of $L(\phi) = L_J / \cos \phi$ and a inductance Hamiltonian of:

$$H_L = -E_J \cos \phi \quad (1.9)$$

The physical arrangement of the junction is extremely similar to a parallel plate capacitance and, indeed, this add to the Hamiltonian a capacitive part:

$$H_C = \frac{Q^2}{2C} = \frac{(2en)^2}{2C} = \frac{4e^2 n^2}{2C} = 4E_C n^2 \quad (1.10)$$

where we defined the charging energy of a single *electron* as $E_C = e^2/2C$ and n is the number of electron on the capacitor.

Overall, the Hamiltonian of the junction is:

$$H = H_C + H_L = 4E_C n^2 - E_J \cos \phi \quad (1.11)$$

For now, n and ϕ where simple classical variables, now we elevate them to be quantum operators that satisfy the commutator $[\hat{\phi}, \hat{n}] = i$. For clarity, we will use a *Cooper pair box* (CPB) as system and not a standalone junction [23]. A scheme is presented in fig. 1.7.

The Hamiltonian is almost identical, but we add a n_g term that is the offset charge induced by a voltage source:

$$H = 4E_C(n - n_g)^2 - E_J \cos \phi \rightarrow \hat{H} = 4E_C(\hat{n} - n_g)^2 - E_J \cos \hat{\phi} \quad (1.12)$$

For a typical CPB we have $E_J/E_C < 1$ so we can express the Hamiltonian in term of *charge states* $|N\rangle$, namely the eigenstates of \hat{n} (note that $[\hat{H}, \hat{n}] = 0$). Note also that with N we indicate the difference of Cooper pairs between the two islands and *not* the

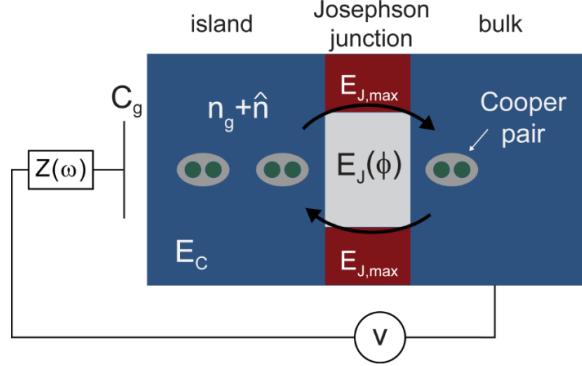


Figure 1.7: Scheme of a Cooper pair box (CPB). Credits [24].

total number of Cooper pairs that would be impractical.

We can write [20]:

$$4E_C(\hat{n} - n_g)^2 = 4E_C \sum_N (N - n_g)^2 |N\rangle\langle N| \quad (1.13)$$

$$-E_J \cos \hat{\phi} = \frac{E_J}{2} \sum_N (|N\rangle\langle N+1| + |N+1\rangle\langle N|) \quad (1.14)$$

Note that $|N\rangle\langle N|$ is essentially the number operator for the Cooper pairs, while $|N\rangle\langle N+1|$ represents the coherent transfer of a single Cooper pair from island to the reservoir (and the other combination the opposite transfer).

Flux-tunable Josephson junction

A single Josephson junction can be used to build a qubit, but its resonance frequency will be fixed by a relation with E_J . For various reasons, and in particular to achieve two-qubits interactions, it is important to have the capability of "freely" controlling the resonance frequency.

To achieve this tunability, we substitute the Josephson junction with a *superconducting quantum interference device* (SQUID), namely two junctions connected in a loop. The SQUID, depicted in fig. 1.8, roughly behaves like a single junction, but its Josephson energy is dependent on an external magnetic flux that may run through the loop, thus giving us tunability.

Let us return to the inductance Hamiltonian (non as an operator), that for a SQUID is:

$$H_L = -E_J \cos \phi_1 - E_J \cos \phi_2 \quad (1.15)$$

where we are assuming that the two junctions are identical (so $E_{J1} = E_{J2} + E_J$). This simplifies the calculations here, but note that in reality the two energies are purposefully made different to limit the tuning range of the SQUID [25].

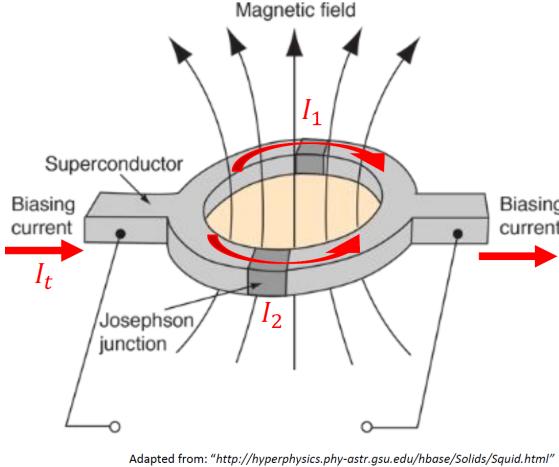


Figure 1.8: Schematic layout of a SQUID.

Between the two phases there is a strict relation:

$$\phi_1 - \phi_2 = 2\pi l + 2\pi\Phi/\Phi_0 \quad (1.16)$$

where we considered two phenomena: the difference in phase, without external flux, must be a multiple of 2π (l here is an integer) so that the wavefunction is single-valued; and the addition of an external flux Φ changes the overall phase by a factor dependent on the ratio between Φ and $\Phi_0 = h/2e$, the superconducting flux quantum.

So we can now write:

$$H_L = -2E_J \cos\left(\frac{\phi_1 - \phi_2}{2}\right) \cos\left(\frac{\phi_1 + \phi_2}{2}\right) = -E_{J\Phi} \cos\phi \quad (1.17)$$

where we defined $\phi = (\phi_1 + \phi_2)/2$ and $E_{J\Phi} = 2E_J \cos(\pi\Phi/\Phi_0)$.

So we re-obtain the same exact Hamiltonian form than the simple Josephson junction, but with a tunable E_J .

1.2.2 Transmon qubits

A Cooper pair box can already work as a qubit, but its properties are not ideal. In general, we use a variation called *transmon*.

A transmon qubit is a CPB shunted by a capacitor that is large in respect to the capacity of the junction. The result is the characteristic ratio $E_J/E_C \gg 1$ (CPB usually have $E_J/E_C < 1$).

The physical implementation of a transmon is usually very different from a CPB (in particular for XMONs, the main transmon architecture), but the Hamiltonian is always of the same form. The eigenvalues, however, are extremely dependent on the E_J/E_C ratio, as visible in fig. 1.9. In the transmon limit the energy levels are extremely flat in respect to n_g , so much more stable, but also more harmonic. Since the sensitivity

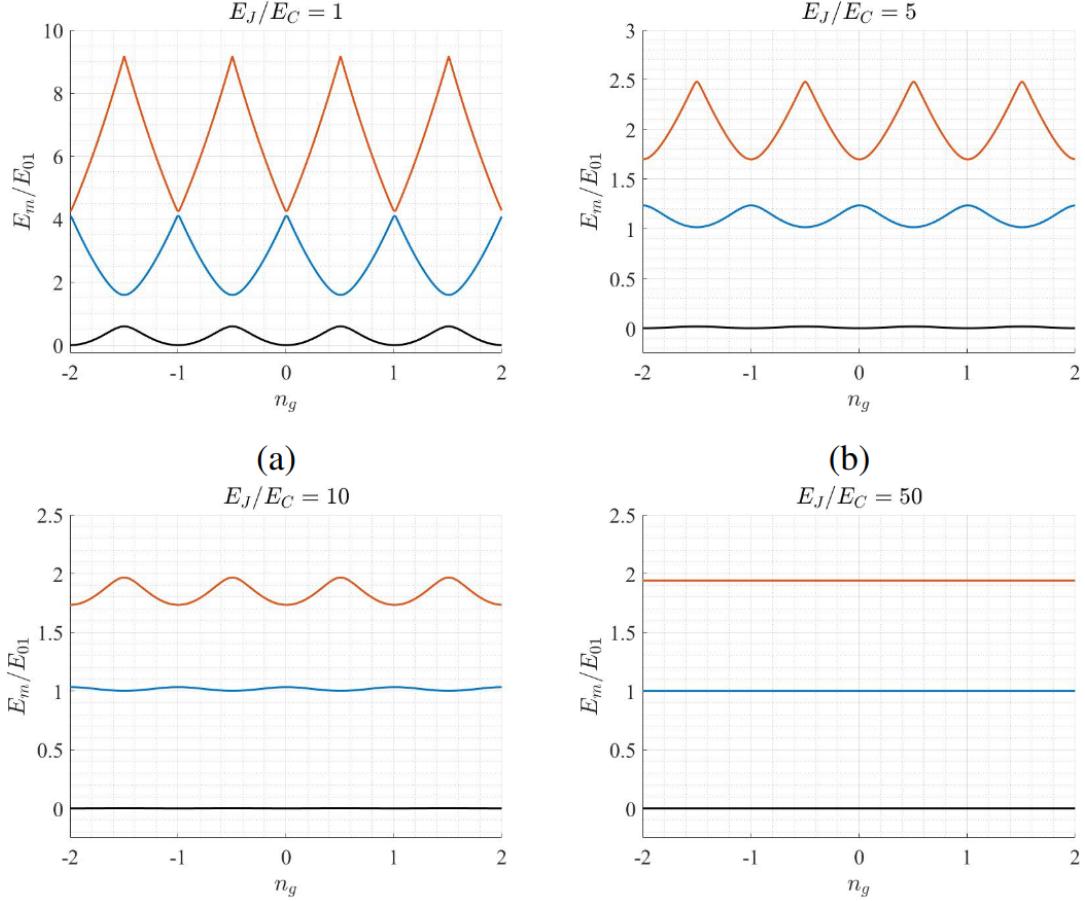


Figure 1.9: Comparison of the Hamiltonian eigenvalues for different ratios of E_J/E_C . Credits [26].

from charge noise decreases faster than the anharmonicity, then we achieve a practical insensitivity from charge noise, while still having enough anharmonicity to consider the system a qubit (typically we have transition of ≈ 10 GHz and anharmonicities of ≈ 200 MHz).

The Hamiltonian of this system, derivable from classical circuits Hamiltonians, can be written as:

$$\hat{H} = 4E_c(\hat{n} - n_g)^2 - E_J \cos \phi \quad (1.18)$$

Flux tunable qubits

The introduction of a SQUID instead of a single junction leads, as we have seen in section 1.2.1, to flux tunability of E_J . For identical junctions in the SQUID we have:

$$E_J = E_{j,max} |\cos(\pi\Phi/\Phi_0)| \quad (1.19)$$

where the external magnetic flux is Φ .

This flux dependency will be exploited in two-qubit gates applications, but also has some major drawbacks.

The idea is that, changing E_J , the first effect is to change the transition frequency 0-1 of the qubit. This follows the relation:

$$f(\Phi) \approx \frac{1}{h} \left(\sqrt{8E_J E_C \left| \cos\left(\pi \frac{\Phi}{\Phi_0}\right) \right|} - E_C \right) \quad (1.20)$$

This gets translated into saying that the frequency is tunable by applying external flux, but also that spurious magnetic currents will affect the qubit.

It is currently not possible to completely remove magnetic fluxes "trapped" in a cryostat and the problem becomes worse when you know that, for every warm up the trapped flux changes and the qubit needs to be re-calibrated.

However, flux-tunable qubits offers a straightforward way of implementing two-qubit interactions, "moving" one qubit frequency to the same value of a neighbor qubit, causing hybridization of states and creation of polarons.

Note that the flux-tunable technology is not the only one for implementing two-qubit interactions. For example, IBM, one of the most important player in the superconducting qubit development, never used flux tunable qubits for their backend and even Nakamura's lab (one of the most prominent labs in the word) is moving away from them, because of the problems related to flux-noise.

In any case, it is possible to minimize the sensitivity of the transition frequency, using a special bias value. The charge-degeneracy point $n_g = 1/2$, that identifies the so-called *sweetspot*. The idea is to move the qubit to its maximum frequency, where we have that higher *and* lower biases both lead to a decrease.

Since the charge dispersion has no slope there, linear noise contributions cannot change the qubit transition frequency. With this procedure, the unfavorable sensitivity of CPBs to charge noise can be improved significantly, potentially raising T2 times from the nanosecond to the microsecond range. Unfortunately, the long-time stability of CPBs at the sweet spot still suffers from large fluctuations which drive the system out of the sweet spot and necessitate a resetting of the gate voltage.

1.3 Transmon interactions

An isolated transmon offers no mean of control and readout, so it is completely useless. We have to add components for readout, drive (control of the qubit state) and flux (control of the qubit frequency).

For a single qubit, we can expect a scheme like the one represented in fig. 1.10.

In fig. 1.10 we can see all the standard lines present in a single qubit system:

Readout input: a line where a signal can be send to probe a resonator that is coupled to the qubit and, because of this, dependent on its state;

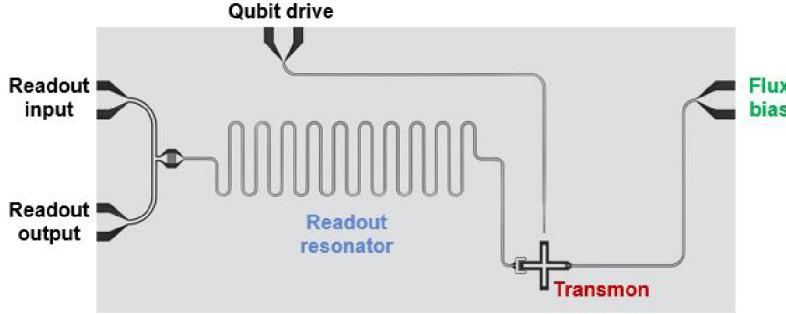


Figure 1.10: Basic scheme for the complete control of a single flux-tunable transmon.

Readout output (feedback): a line where the sent readout signal gets acquired, after the resonator interaction. This signal can be the input *reflected signal* (as in the scheme presented) or, in some configurations, the *transmitted signal*

Drive: a line coupled to the qubit, that we can use to change the qubit state itself;

Flux: a line that contains an inductor, so that a DC current passing through it can generate a magnetic field to change the qubit frequency.

An illustration, in circuit form, of the system is presented in fig. 1.11.

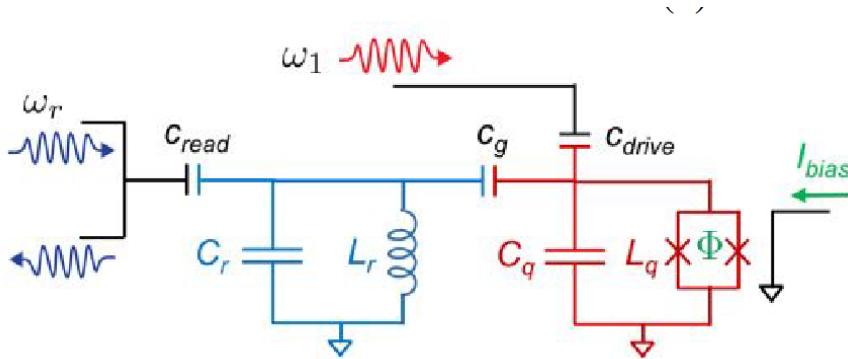


Figure 1.11: Basic circuit scheme for the complete control of a single flux-tunable transmon.

To study the system we can start writing the Hamiltonian of the resonator-qubit system. We can express the diagonalized qubit Hamiltonian, non limited to the first two states, as:

$$\hat{H}_T = \sum_j \hbar \omega_j |j\rangle \langle j| \quad (1.21)$$

where ω_j is the eigenvalue associated with the $|j\rangle$ eigenstate.

The resonator Hamiltonian can be written in the standard harmonic oscillator form:

$$\hat{H}_R = \hbar\omega_r \hat{a}^\dagger \hat{a} \quad (1.22)$$

The total Hamiltonian of the system will add one more term:

$$\hat{H} = \hat{H}_T + \hat{H}_R + \hat{H}_I \quad (1.23)$$

The interaction term can be written as:

$$\hat{H}_I = 2e \frac{C_g}{C_g + C_q} \hat{V}_r \hat{n} = 2e \frac{C_g}{C_g + C_q} V_{rms} (\hat{a} + \hat{a}^\dagger) \hat{n} \quad (1.24)$$

where \hat{V}_r is the resonator voltage operator and $V_{rms} = \sqrt{\hbar\omega/2C_r}$.

If we write the qubit charge operator in terms of the transmon eigenstates, define the coupling $g_j = 2e \frac{C_g}{C_g + C_q} V_{rms} \langle j-1 | \hat{n} | j \rangle$ and simplify the equation considering only the relevant terms:

$$\hat{H} = \sum_j \hbar\omega_j |j\rangle \langle j| + \hbar\omega_r \hat{a}^\dagger \hat{a} + \sum_j g_j (|j-1\rangle \langle j| \hat{a}^\dagger + |j\rangle \langle j-1| \hat{a}) \quad (1.25)$$

this equation is usually called *Jaynes-Cummings Hamiltonian*.

To understand the simplification we can write the generalized Jaynes-Cumming Hamiltonian as:

$$\hat{H} = \omega_r (\hat{a}^\dagger \hat{a}) - \frac{1}{2} \omega_1 \sigma_z - g(\hat{a}^\dagger + \hat{a})(\sigma_- + \sigma_+) \quad (1.26)$$

where σ_z is the Z Pauli matrix and the σ_\pm are the transmon ladder operators.

This representation does not add anything to the previous one, but it could maybe be a bit more easy to understand when we focus on the interacting Hamiltonian:

$$\hat{H}_I = \hat{a}^\dagger \sigma_- + \hat{a} \sigma_+ + \hat{a}^\dagger \sigma_+ + \hat{a} \sigma_- \quad (1.27)$$

The first term describes a qubit decay while creating a new photon in the resonator and so on.

Now it's easy to identify that, of those 4 terms, only 2 conserve the energy and are much less likely to occur. Actually, using the *Rotating Wave Approximation* (RWA), we can remove these terms, reaching again the form of eq. (1.25).

In the dispersive regime ($\Delta = |\omega_j - \omega_r| \gg g_j$) we can define the *dispersive shift* $\chi = g_1^2 / |\omega_1 - \omega_r|$ and, considering only $|0\rangle$ and $|1\rangle$ as the first two transmon states, write:

$$\hat{H} = \hbar(\omega_r - \chi |1\rangle \langle 1| + \chi |0\rangle \langle 0|) \hat{a}^\dagger \hat{a} + \frac{\hbar}{2}(\omega_1 + \chi)(|0\rangle \langle 0| - |1\rangle \langle 1|) \quad (1.28)$$

or better:

$$\hat{H} = (\omega_r - \chi \sigma_z) \hat{a}^\dagger \hat{a} - \frac{1}{2} \omega_q \sigma_z \quad (1.29)$$

where σ_z is the Pauli matrix and $\omega_q = \omega_1$.

This is the relation exploited in the standard transmon readout scheme.

1.3.1 Control and readout

Driving a qubit

Ignoring the resonator, we can consider the dynamic Hamiltonian of a driven qubit with:

$$H_{\text{drive}} = -\frac{1}{2}\omega_q\sigma_z - E(t) \cdot \hat{d} \quad (1.30)$$

where we are considering a classical coherent signal with dipole interaction: $E(t) = E \cos(\omega_d t)$. If the dipoles are aligned we can simplify the Hamiltonian as:

$$H_{\text{drive}} = -\frac{1}{2}\omega_q\sigma_z - A \cos(\omega_d t)\sigma_x \quad (1.31)$$

We can describe the interacting qubit dynamic considering a state:

$$|\psi(t)\rangle = C_0(t)e^{+i\frac{\omega_q}{2}t} |0\rangle + C_1(t)e^{-i\frac{\omega_q}{2}t} |1\rangle \quad (1.32)$$

Solving the Schrodinger equation we obtain:

$$C_0(t) = \frac{e^{-i\frac{\Delta_d}{2}t}}{\Omega_R} \left(\Omega_R \cos\left(\frac{\Omega_R}{2}t\right) + i\Delta_d \sin\left(\frac{\Omega_R}{2}t\right) \right) \quad (1.33)$$

$$C_1(t) = i \frac{A e^{i\frac{\Delta_d}{2}t}}{\Omega_R} \sin\left(\frac{\Omega_R}{2}t\right) \quad (1.34)$$

where we defined $\Delta_d = \omega_q - \omega_d$, $A = Ed$ and $\Omega_R = \sqrt{A^2 + \Delta_d^2}$.

These equations describe the dynamic of a driven qubit. We can extract the solution in terms of probabilities:

$$P_1(t) = |C_1(t)|^2 = \frac{A^2}{A^2 + \Delta_d^2} \sin^2\left(\frac{\sqrt{A^2 + \Delta_d^2}t}{2}\right) \quad (1.35)$$

Measurements

To measure the state of a qubit, we will never probe it directly but rather use eq. (1.29) to our advantage.

The idea is that the effective resonance frequency of the resonator will be dependent on the state of the qubit as: $\omega = \omega_r - \chi\sigma_z$. So it will be sufficient to measure this frequency to infer the qubit state.

Note that the resonance frequency corresponds to the excitation energy.

Considering the resonator at the ground state, we can identify two states: $0(G)$, corresponding to the qubit in state $|0\rangle$, and $0(E)$ corresponding to the qubit in $|1\rangle$. To these states, different transition energies (or frequency) will be available so we can write,

defining γ as a photon with the transition energy tuned to the excitation of the state in the underscript:

$$\begin{aligned}\gamma_{0(G)} + 0(G) &\rightarrow |1\rangle(G) \\ \gamma_{0(G)} + 0(E) &\rightarrow \gamma_{0(G)} + 0(E)\end{aligned}$$

this difference is explained considering that, in the second case, the sent photon does not have the energy tuned to the transition.

Consider first a circuit with a planar resonator. A planar resonator is a type of resonant structure designed on a single plane, typically on the surface of a printed circuit board (PCB) or a similar substrate. The readout line, often called transmission line, will be coupled to the resonator just by proximity. This means that, if we send a certain number of photons through the line and they are tuned to the transition frequency, they will get absorbed by the resonator that will get excited. We will therefore have "missing photons" at the end of the readout line.

The situation is slightly different in the case of a 3D resonator, in which the readout line passes within. 3D resonators are resonant structures that extend into three dimensions, and they are not confined to a single plane. Usually, in quantum computing, the typical structure is a cavity resonator. Indeed, the phenomenon appears here in the opposite way, with an increase in the photon number for on resonance energies. This can be explained with the classic constructing interference effect seen in all kind of standing waves, as they are the ones in the 3D resonator.

So this is the very simple idea behind superconducting qubit measurements. But what happens in case of superposition states?

From the principles of quantum mechanics, we will have the wave-function collapsing in the measured state. This should not come as a surprise, but what it's interesting to note is the non-destructiveness of this protocol. The idea is that two consecutive measurements, should lead to the same exact result, because the measurement should not disturb the state of the qubit (note that the wavefunction collapse is not considered a disturb here). This property is desired and expected in ideal conditions, and it is also required in numerous error correction schemes however, in real hardware, it is often not completely achievable.

Chapter 2

Experimental setup and software

2.1 Experimental setup

In this section I will schematically present the experimental setup used for this thesis. Note however that, being this an experimental work, the setup changed various times and it is difficult to give a complete description of it, for all the states it had.

The schematics presented here correspond to the final setup.

2.1.1 Cryostat

Since the qubits that we are considering are transmons, they work exploiting the superconductivity phenomenon. To reach the superconductivity regime, we need a device capable of refrigerating it to the mK region. The device employed for this task is called cryostat.

In this thesis I used both the SD and XLD cryostats manufactured by Bluefors, but the majority of the data presented here was acquired with qubits in the XLD so I will focus on the description of those last setups.

The technique used by cryostats to achieve these temperatures is the dilution refrigeration [27, 28]: a process that exploit a mixture of ^4He and ^3He to reach temperatures below the minimal one achievable through pressure process. The two isotopes are quite different: the former is a boson with spin 0 that can be found in natural gas reservoirs, while the latter is a fermion (spin 1/2) and is the byproduct of tritium fabrication in nuclear reactors.

^3He can be diluted in ^4He at any concentration, but when the mixture is cooled below ~ 0.85 K (note that pure liquid ^3He can be cooled to a minimum of 0.3 K) it undergoes a spontaneous phase separation.

Two phases now coexist: a concentrated phase composed essentially only of ^3He and a diluted phase that, on the contrary, is mostly composed of ^4He .

The graph in fig. 2.1 shows equilibrium concentrations: intersections of the phase separation line and a horizontal isotherm line correspond to the concentrations of the two different phases (if below the 0.85 mK threshold).

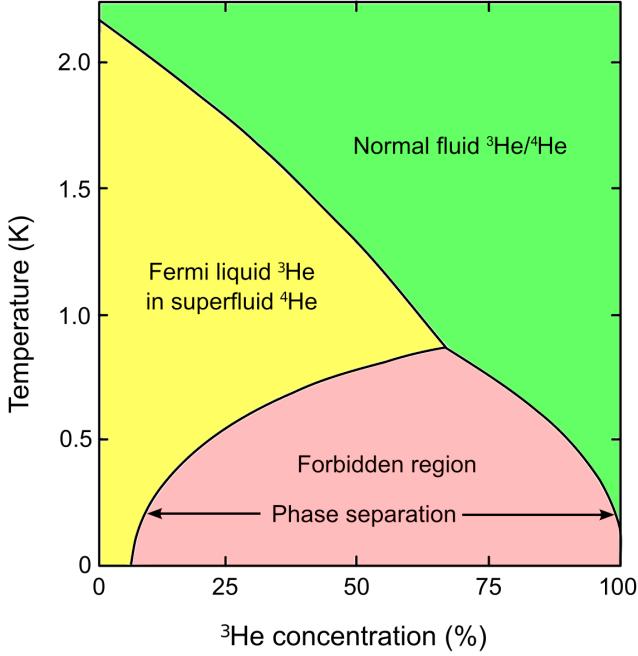


Figure 2.1: Phase diagram for He as a function of ³He concentration.

In the coldest part of the cryostat, the mixing chamber, ³He atoms cross the interface between the concentrated and the diluted phase. This transition is endothermic and removes heat from the mixing chamber. When ³He dissolves in ⁴He at such low temperatures, it undergoes a process similar to evaporation in a vacuum and produces vapor. The diluted phase is connected to another chamber called still (short for distilling pot) that is held by a heater at ~ 0.7 K and continuously pumped to remove the vapor which is mainly composed of ³He because of its greater vapor pressure. The depletion of ³He within the still produces an osmotic pressure that drives more ³He to the still from the mixing chamber where phase transition can continue to happen. ³He pumped from the still is reintroduced to the mixing chamber through a flow impedance that enables the helium to condense and that cools it down via heat exchangers that work making use of the cold ³He that is flowing upwards toward the still. This process is schematized in fig. 2.2.

In fig. 2.3 some picture of the XLD are present. Without any thermal load can reach 8 mK, while with minimal thermal load (the qubits mounted but not probed) is stable at 9 mK.

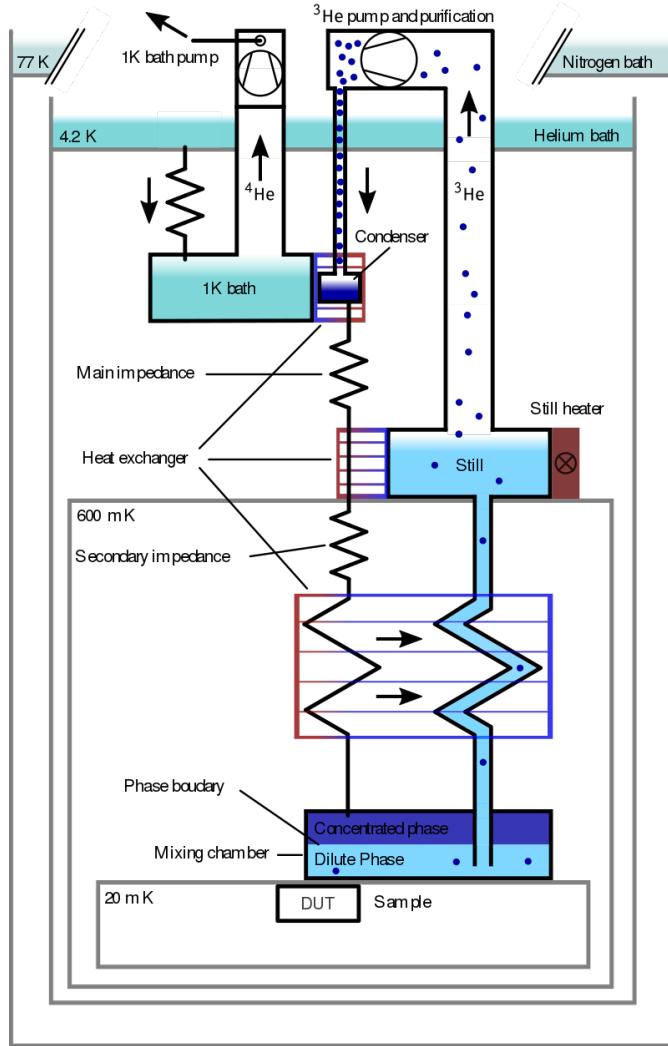


Figure 2.2: Outline of the dilution process.

2.1.2 FPGA & RFSoC

A Field-Programmable Gate Array (FPGA) is an electronic board that supports complete firmware configuration *after* manufacturing as well as partial re-configurations while executing a specific program.

While integrated circuits serve a specific purpose and cannot be re-configured after fabrication, FPGAs can contain arrays of programmable logic blocks that can serve simple logic purposes (as Boolean operations) or complex functions, depending on the loaded firmware. Since they are completely re-configurable and since they can improve the performance of standard CPU (with task-specific firmwares) without the large energy conception required by GPUs, FPGAs are rapidly becoming of widespread use. More-



Figure 2.3: Bluefors XLD.

over, since the same FPGA can serve different purposes in extremely different fields, it is usually a cheap alternative to standard systems.

The drawback of all of this is in the overhead that programming a FPGA introduces. To compile a FPGA firmware one first has to define the logic to use via Hardware Description Languages (HDL) then design the connections internal to the FPGA logic and conduct precise tests on the produced firmware.

The type of FPGAs used in my thesis are RFSoC, namely Radio Frequency System on Chip: their main characteristic is the capability of synthesizing directly frequency up to ≈ 10 GHz, as well as the capability of acquiring extremely high frequency by exploiting the downsampling technique. All of this combined in the same board, along with a standard CPU, RAM and memory: enabling the creation of a full on-board system.

In this section I will provide some explanation on the techniques used for high-frequency synthesis and high frequency acquisition. Also, some details on the FPGAs used will be provided.

Direct Digital Synthesis (DDS)

The technique used for direct microwave synthesis is called *Direct Digital Synthesis* (DDS) and it enables to use the DACs as arbitrary waveform generators for high frequencies [29].

In general, a DDS system consist in a precision reference clock (often a crystal clock), an address counter, a programmable read-only memory (PROM) and a DAC (see fig. 2.4 for reference).

In the PROM, a complete cycle of a periodic function is stored and the address counter steps through it, accessing sequentially all the PROM's memory locations (so the PROM is working a lookup table). The DAC then converts the input from the PROM, to analog outputs.

Note also that the PROM can be composed of two different lookup tables, for complex I-Q signals, often referred to "in-phase and quadrature signals," are a pair of time-varying signals used in signal processing and communications. They represent:

In-Phase (I) Component The "I" component represents the signal's amplitude or phase information. It corresponds to the real part of the complex signal.

Quadrature (Q) Component The "Q" component is also a real part of the complex signal but is phase-shifted by 90 degrees relative to the "I" component. The "Q" component carries information orthogonal to that of the "I" component.

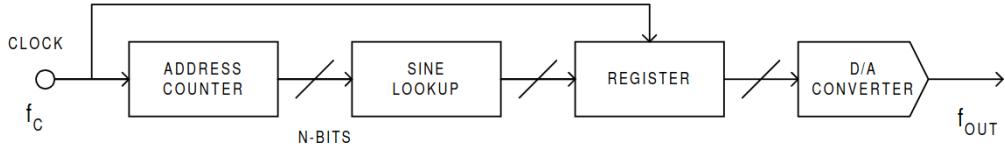


Figure 2.4: A simple Direct-Digital-Synthesizer, where the function stored in PROM is a sine wave. Credits [29].

This implementation of DDS is a bit simplified and, although it works, it lacks tuning flexibility and the output frequency is effectively fixed to the reference clock one.

A modern DDS system adds to the shown signal chain a phase accumulator, creating an architecture usually referred to as numerically-controlled oscillator NCO (see fig. 2.5). The address counter is replaced with an N -bit variable-modulus counter and a phase register: effectively storing not a "random" real number, but an integer mapped to a "phase wheel" that indicates the function points to be generated. A typical scheme is presented in fig. 2.6.

This architecture, in respect to the more standard *Phase-Locked Loop* (PLL), offers the advantages of micro-Hertz tuning resolutions, unparalleled matching and control of I-Q synthesized outputs, extremely fast changing speed in frequency and phase.

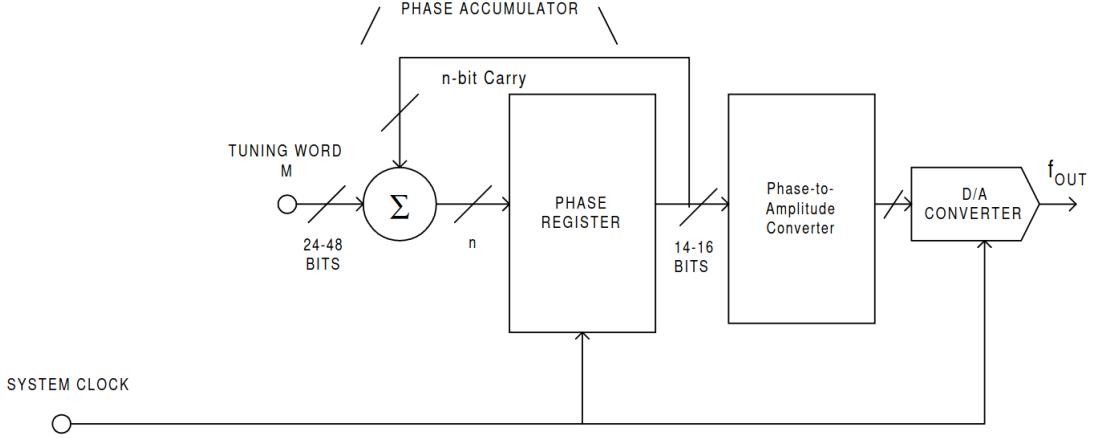


Figure 2.5: Details of the phase accumulator component of a DDS system. Credits [29].

Aliasing and synthesis at higher Nyquist zones

Now that we have an idea of the hardware involved for the synthesis, let's try to understand analytically how the DACs behave, how can they be used to synthesize directly in the microwave region, and what are some of the problems involved.

We can write the output of the DACs as an analytical function [2]. Given f_s the sampling frequency, so the rate of conversion from digital samples of a generic analytical function $x(t)$ to analog outputs $v(t)$, we can write:

$$v(t) = \left[x(t) \sum_{k=-\infty}^{\infty} \delta(t - kT) \right] * r(t) \quad (2.1)$$

Where $T = 1/f_s$ is the sampling period, inverse of the sampling frequency, and $*$ is the convolution in time operator. The $r(t)$ function is called *reconstruction waveform* and describes the DAC behaviour between two subsequent samples (so it is non-zero only for $0 \leq t \leq T$). So, trying to understand eq. (2.1), the summation if a function that is always zero, but in multiples of T , where the value between brackets reproduces the $x(t)$ function. Then we have to "connect the dots" to reach an analog continuous function, and this is the role of $*r(t)$.

We now apply Fourier Transform to $v(t)$ to obtain the frequency spectrum (capital functions are the FT of the corresponding time domain function):

$$V(\omega) = \mathcal{F}\{v(t)\} = \left[X(\omega) * \mathcal{F}\left\{ \sum_{k=-\infty}^{\infty} \delta(t - kT) \right\} \right] R(\omega)$$

For the sum we can recognize the T periodicity:

$$\sum_{k=-\infty}^{\infty} \delta(t - kT) = \sum_{n=-\infty}^{\infty} c_n e^{2\pi n/T}$$

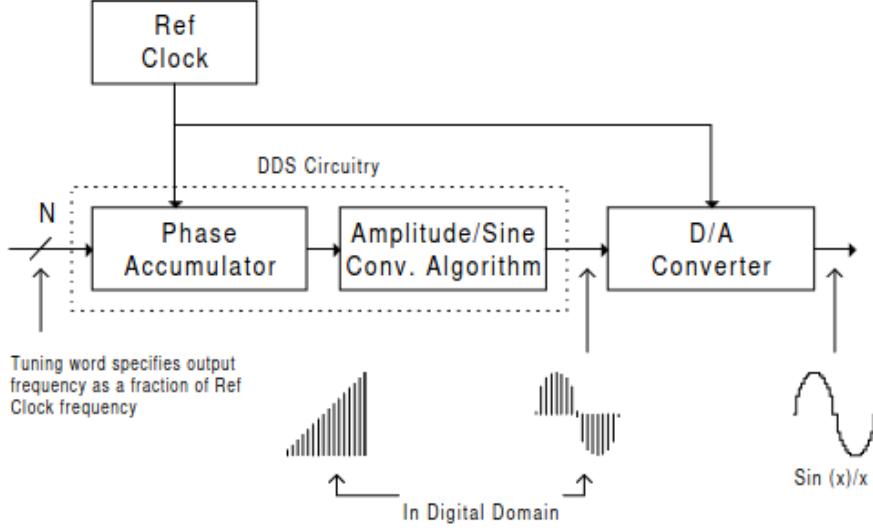


Figure 2.6: Scheme of a generic NCO DDS system. Credits [29].

From $k = 0$ we can extract:

$$c_n = \frac{1}{T} \int_{T/2}^{T/2} \delta(t) e^{2\pi n/T} dt = \frac{1}{T}$$

So we can compute the remaining part of FT:

$$\mathcal{F} \left\{ \sum_{n=-\infty}^{\infty} \frac{1}{T} e^{2\pi n/T} \right\} = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta(\omega - 2\pi n/T) = \sum_{n=-\infty}^{\infty} \delta(\omega T - 2\pi n)$$

And finally:

$$V(\omega) = R(\omega) \left[X(\omega) * \sum_{n=-\infty}^{\infty} \delta(\omega T - 2\pi n) \right] \quad (2.2)$$

The summation represents a series of peaks at multiple of the sampling frequency f_s and the convolution with $X(\omega)$ creates copies of the original signal spectrum at each peak. If $X(\omega)$ has a bandwidth larger than $f_s/2$ these copies will overlap, leading to a noise phenomenon called *aliasing*. The frequency bands to be considered, each extending for $f_s/2$, with the first starting at 0 are called *Nyquist zones*. So it is not possible to synthesize a signal in a single zone: if $X(\omega)$ is confined in a single one, then every odd zone will have an identical image of the signal, while every even zone will have an inverted image.

This is why we generally need to use band pass filters: to exclude the images of spurious Nyquist zones and reduce aliasing effects.

The reconstruction waveform $r(t)$ manifest itself as a frequency-dependant attenuation and different functions will be used for maximize the output power for different

Nyquist zones. In particular, the three most common choices are (note that they are all defined in the interval $[0 \leq t \leq T]$):

$$\text{non-return-to-zero (NRZ): } r(t) = 1 \rightarrow R(\omega) = Te^{-i\omega T/2} \frac{\sin(\omega T/2)}{\omega T/2}$$

$$\text{return-to-zero (RZ): } r(t) = \theta(-t + 0.5T) \rightarrow R(\omega) = \frac{T}{2} e^{-i\omega T/4} \frac{\sin(\omega T/4)}{\omega T/4}$$

$$\text{mix-mode-rf (MIX): } r(t) = -\text{sign}(t - 0.5T) \rightarrow R(\omega) = \frac{\omega T^2}{4} e^{-\frac{i}{2}(\omega T - \pi)} \left(\frac{\sin(\omega T/4)}{\omega T/4} \right)^2$$

In fig. 2.7 it's shown the attenuation-frequency dependence for different Nyquist zones and different reconstruction waveforms.

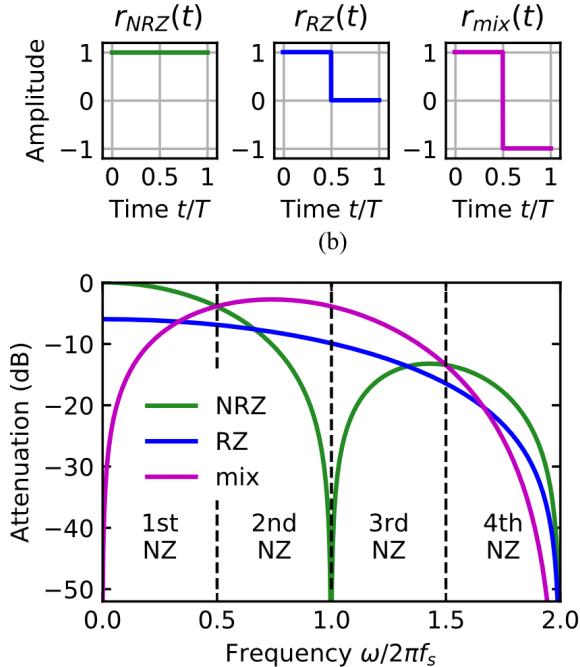


Figure 2.7: Attenuation-frequency relationship at different Nyquist zones, for different reconstruction waveforms. Credits [2].

The RFSoC-based system that I developed used RNZ for the first Nyquist zone and MIX for all the others.

Undersampling, decimation and Direct Down Conversion (DDC)

In last section we saw the main principles and ideas that enable high-performance digital-to-analog conversion, now let's illustrate analog-to-digital.

The starting point for this discussion is the Nyquist-Shannon sampling theorem that establish a sufficient condition for a discrete-time signal to capture all the information

¹ θ is the Heaviside function.

contained in a continuous-time signal. Without entering in any details regarding the mathematical demonstration, we can express the theorem as²:

Theorem 1 *If a function $x(t)$ does not contain any frequencies higher than M Hz, then it can be completely described by finite samples spaced less than $1/(2M)$ s.*

So, a signal must be sampled at a rate greater than twice its maximum frequency in order to assure complete and unambiguous reconstruction. In general, the effect of not meeting the Nyquist-Shannon criterion is again aliasing.

Sampling with a frequency $\geq 2M$ is called *normal sampling* or *oversampling*.

If we use sampling frequency that are less than twice the maximum frequency than we use *undersampling*. Here we have another theorem, also called Nyquist-Shannon sampling theorem:

Theorem 2 *If a function $x(t)$ has a bandwidth of B Hz, then it can be completely described by finite samples spaced less than $1/(2B)$ s.*

The two theorems are similar, but note that they apply at the same time, so we will have aliasing. While aliasing usually is a non-desired effect, with undersampling we try to exploit it. The aliased signal always appear at:

$$f_{ADC} = f_s - f_{in} \quad (2.3)$$

where f_s is the sampling frequency, f_{in} the input frequency and f_{ADC} the acquired one. If we know in advance that the signal is aliased, than we can easily recover the actual frequencies using the inverse of eq. (2.3).

Since in our system we are acquiring a reflected/transmitted signal that was produced by us, we know exactly what frequencies we have in input and is very easy to reconstruct the signal. Also note that the undersampling technique has several advantages:

- higher sampling rate increases data rates to FPGAs and this can deteriorate performances or increase the cost of the FPGAs;
- higher data rates usually need more time to setup, meaning larger dead times;
- higher sampling rate are linked to higher power consumption;
- to reach a situation where the Nyquist-Shannon criterion is met we usually need to involve external Local Oscillators and Mixing, that are not needed for undersampling.

And there are also some disadvantages:

²This theorem is often simply referred to as Nyquist sampling theorem.

- worse S/N ratios ³;
- we need to plan in advance what frequency we want to sample, otherwise we will not be able to directly apply eq. (2.3);
- undersampling cannot be used to sample high signal bandwidths.

The RFSoC-base system developed in this thesis used the undersampling, with the great advantage of not requiring external local oscillator.

Moreover, at the output of each ADC, internally to the FPGA logic, another operation was conducted: an eight-time *decimation*. The idea of decimation is that it is inefficient to transmit a wideband spectrum when only a narrow band is required, so we conserve only a small periodic portion of the ADC samples. The results is to effectively reducing the sample rate of the ADC: for our system, a decimation by 8 means that 7/8 of the total samples were completely discarded.

The ADC also contains a NCO and a digital filters that remove the out-of-band noise.

All this system, that can usually be called *Digital Down Conversion* (DDC) in analogy with DDS preserves all the information in the frequency band of interest, while also removing much all noise at different frequencies.

In the RFSoC-base system for quantum control it will be used to efficiently acquire complex (I-Q) samples.

Used RFSoC boards

The system for quantum control and readout that was developed during this thesis can in theory be used without any modification with all the boards supported by the **Qick** [3] package: a firmware and software library developed at Fermilab for using RFSoC boards for quantum computing. However, all the results produced and experiment conducted where obtained using a **RFSoC4x2** and a **ZCU111** (for some part of the thesis I also worked with a **ZCU216** which shares the main characteristics with the **ZCU111**). Some details about these boards are presented in table 2.1.

Both of these boards have Ethernet connectors so they were always controlled over the internet.

The **RFSoC4x2** is much smaller than the **ZCU111**, while it is smaller, the **RFSoC4x2** is a FPGA of third generation, meaning it has much higher sampling frequencies. Since there are just two DACs, this board can be used to control a single non-flux-tunable qubit. The connections will be:

³The Signal-to-Noise Ratio (S/N ratio) is a measure used in signal processing and communication systems to quantify the relative levels of a desired signa to unwanted background noise. It is typically expressed in decibels (dB) and is calculated as:

$$\text{S/N ratio (dB)} = 10 \cdot \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{noise}}} \right)$$

Where: P_{signal} is the power of the signal (the strength of the desired information-carrying component of the signal) and P_{noise} is the power of the noise (the unwanted or interfering components of the signal, typically background or random variations).

	RFSoC4x2	ZCU111	ZCU216
Number of ADCSS	4 (1 used)	8 (2 used)	16 (2)
Number of DACs	2	8	16
DAC sampling frequency	9.85 GSPS	6.55 GSPS	9.85 GSPS
ADC sampling frequency	5.00 GSPS	4.09 GSPS	2.5 GSPS
Chip generation	third gen.	first gen.	third gen.

Table 2.1: Outline of the main characteristics of the **RFSoC4x2**, **ZCU111** and **ZCU216** boards.

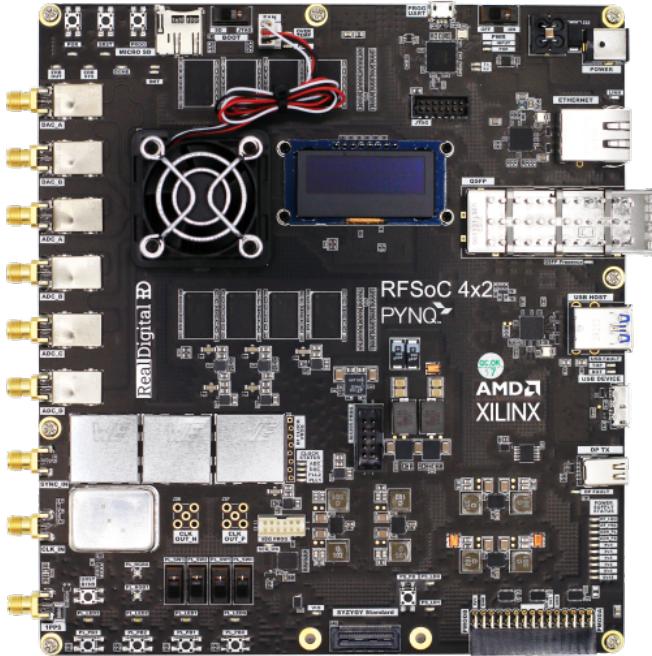


Figure 2.8: Picture of a RFSoC4x2 board.

- **DAC 1:** drive line
- **DAC 2:** readout line
- **ADC 1:** feedback line (for readout)
- **ADC 2:** not active
- **ADC 3:** not active
- **ADC 4:** not active

As visible in fig. 2.8 **RFSoC4x2** has direct connections to SMA cables so it doesn't require any specific additional connector. By *not active* I mean that the firmware used never activates it; a different firmware potentially could.

On the other hand, the **ZCU111** board is a bit more complex. As is visible in fig. 2.9 there are no SMA connectors and an additional board is needed to convert from the black band connectors (the two band on the left in the figure) to SMAs.

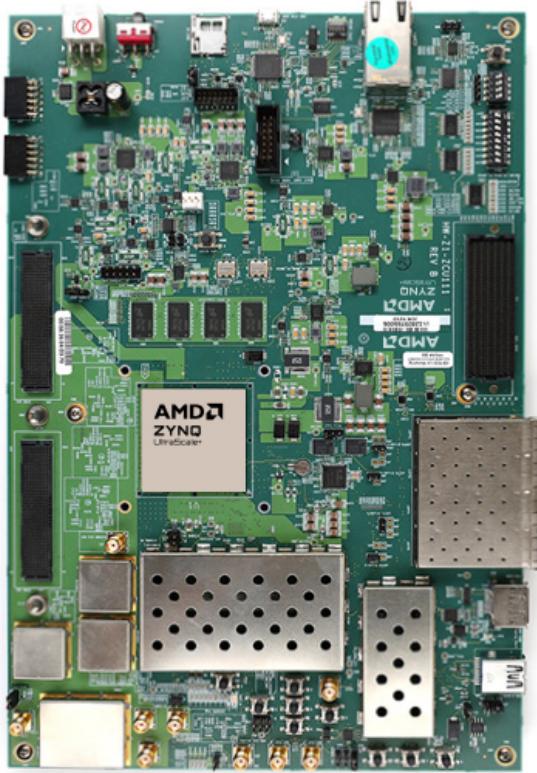


Figure 2.9: Picture of a ZCU111 board.

Fortunately, with the **ZCU111** is sold by default a **XM500** board, shown in fig. 2.10. Unfortunately, the **XM500** board is not just a convertor, but also adds some filtering and changes the output in non trivial ways. In particular, the **XM500** has:

- **external clock inputs/outputs:** not active;
- **4 single output DACs:** these are all active, but they have a balun filter added to it. That is a high-pass filter that removes DC components. So these cannot be used for DC flux control;
- **2 ADC:** active and also with baluns (but there are no downsides here);
- **3 differential output DACs:** these do not have baluns, so they can be used for flux lines, but require specific differential amplifiers to subtract the two outputs to convert them to a single SMA (also, the vast majority of differential amplifiers does not work for DC signals).

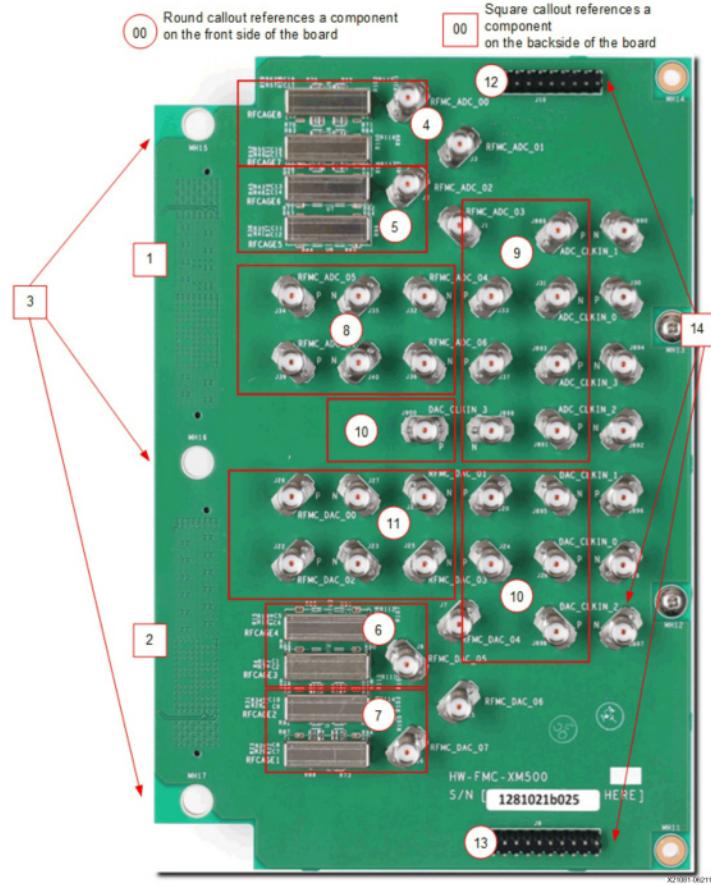


Figure 2.10: Picture of a XM500 board.

For the **ZCU111**, the Qick project offers two different firmwares: one uses both ADCs, so can be used to control up to two flux-tunable qubits, the other uses a single ADC multiplexed. The multiplex firmware is of particular interest and has been extensively used during the thesis. Loading it, one of the DACs become capable of sending 4 different pulses at different frequencies but at the same time, while one of the ADC becomes capable of acquiring 4 different pulses (again at 4 different frequencies) at the same time.

This is one of the perks of using a re-configurable FPGA: since the ADCs and DACs are products of the FPGA logic is possible to reconfigure them so that a single ADC is actually composed of 4 with smaller bandwidth.

For this configuration, anyway, we used a standard upconversion system for the DAC so that, even with a smaller f_s , the output was in the first Nyquist zone (for power reasons).

2.1.3 Qubits Configuration

At TII, the XLD cryostat was used at its full capacity and contained, at the mixing chamber level:

- 3 single non-flux-tunable qubit manufactured by TII, in custom 3D cavities;
- a 5 flux-tunable qubits chip, with a star connectivity (namely a single qubit in the middle that can couple to the other four);
- a 5 flux-tunable qubits with couplers (namely flux-tunable qubits between other qubits, used to mediate the interaction between the two qubits);
- a 25 qubit chip with 5 readout lines and various qubits connected to each other.

In the work for this thesis, I tested and interacted, in particular, with the single qubit devices and one readout line of the 25 qubit chip.

The single qubits, that were changed and replaced over my period at TII, generally share the same schematic and the reader can imagine them as presented in fig. 2.11.

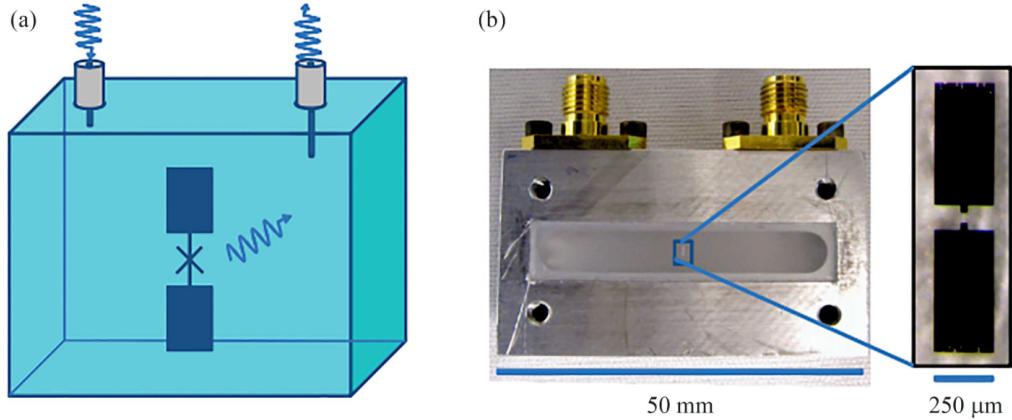


Figure 2.11: Example of a single qubit in a 3D cavity. Credits [30].

The 25 qubits chip is schematically presented in fig. 2.12

Among all the present qubits, the **ZCU111** was connected to line D.

Note that every line has 5 qubits plus an isolated one (the yellow) that are qubit used only for testing.

Moreover, all the black interconnecting lines are possible two-qubits gates.

2.1.4 Configuration and setup for RFSoC4x2

During this thesis work, the **RFSoC4x2** was connected to multiple single qubit in 3D cavities. Therefore, the setup changed several times, but it always had some stable attributes. For example, in a single qubit (transmon) non flux tunable, there is only a

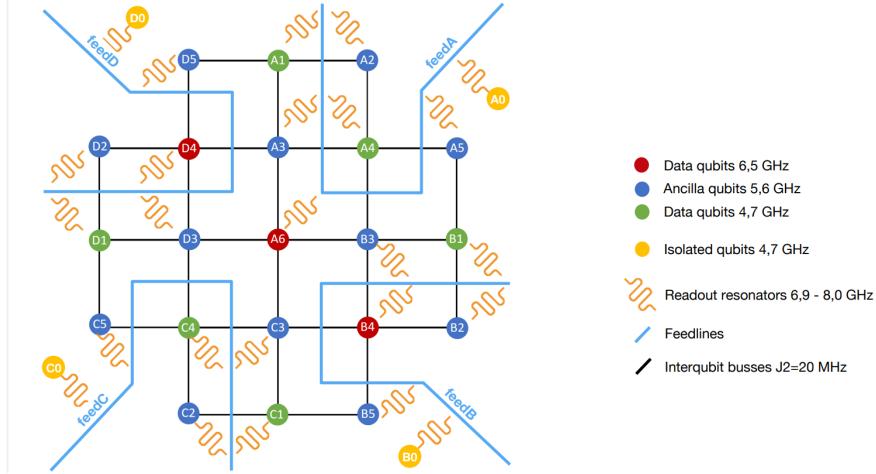


Figure 2.12: Topology of the Contralto QuatWare chip. Credits [31].

single line for input (were both readout and drive pulses are fired) and a single readout output (*feedback*).

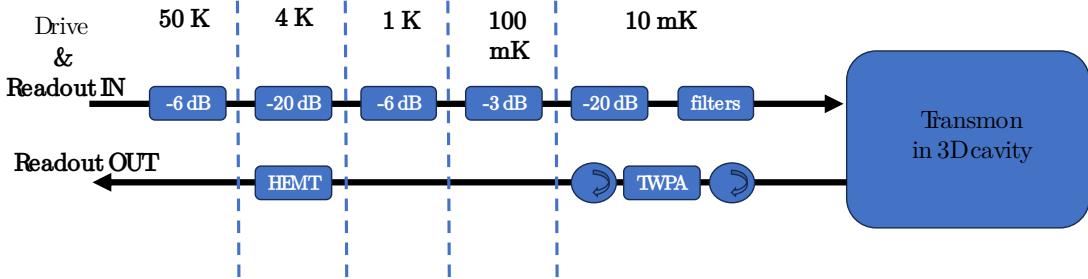


Figure 2.13: Schematics of the lines of a single non-flux-tunable qubit in a 3D cavity.

In fig. 2.13 the two lines (in-fridge) are presented. All the attenuation present in the IN line is needed to help to reach the single-photon level (or near-single-photon) and its spread to avoid excessive thermal load to a specific stage of the dilution. The filters represented at the mixing chamber level included both IR filter and low-pass filters to clean in the best possible way any noise.

In the readout OUT line, the first element is TWPA [32] manufactured by Silent Waves [33] continuously pumped (with another line not present in the scheme) during the experiments. The role of the TWPA is to amplify the signal, inducing a noise amplification quantum limited. Note that the TWPA was not present in the majority of the setups, but is inserted in the scheme for completeness.

At the 4 K level, a cryogenic HEMT [34] is present so that the signal, extremely low in the input, becomes measurable.

At room temperature the *feedback signal* is amplified with a LNA (Low Noise Am-

plifier) manufactured by MiniCircuit [35] (max +8 dB) and sent to the **RFSoC4x2** ADC 0.

The drive source is DAC 0 (B on the board) and sometimes an amplification of max +8 dB was needed. The readout IN was connected to DAC 1 (A on the board) and merged to the drive by using a passive *splitter*.

At room temperature, some band-pass filters were added to remove spurious created by the synthesis process and eventual high frequency noise.

2.1.5 Configuration and setup for ZCU111

Similarly as per single qubit control, in fig. 2.14 are showed the in-fridge lines for the control of the multiplexed qubits of the 25 qubits chip.

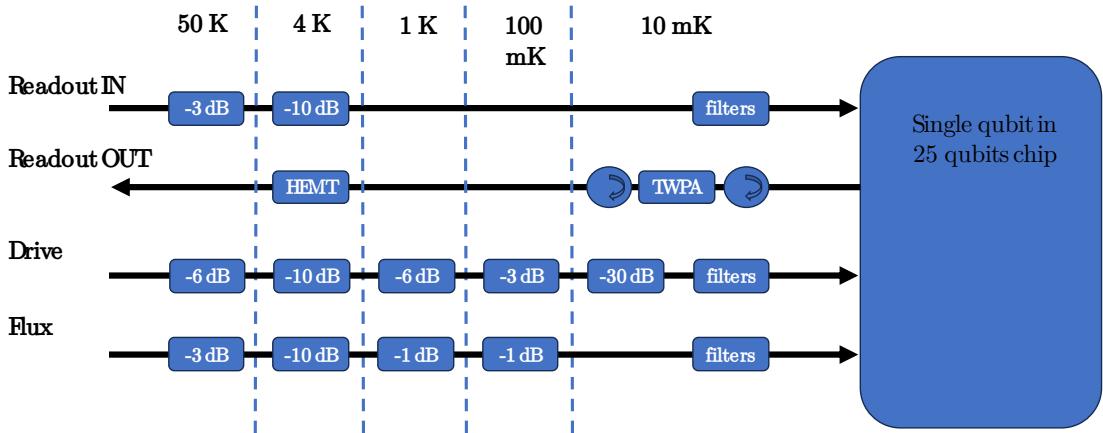


Figure 2.14: Schematics of the lines of a single flux-tunable qubit in the 25 qubit chip.

Note first that there are multiple input lines and, in particular, that the readout in and the drive line are separated. This is mainly because this is a 2D chip and not a cavity.

In any case, both for the readoutIN and the drive the filters are the same already named in the single qubit setup.

For the flux line, on the other hand, the filters are extremely low frequency filters since in theory we just need a DC current.

The TWPA is one of the differing elements. In fact it was not used a standard fixed-band TWPA as per the single qubit setup, but a prototype of a variable-band TWPA [36]. This TWPA needs a pump to work and a DC current to tune the amplification band and gain. Both these lines are not showed in the scheme since they are more or less secondary.

Some more differences are present at room temperature, where the setup is a bit more complex. In particular note that the schematics presented in fig. 2.14 represent a single qubit, but this is a system with 5+1 multiplexed qubits. Therefore, at room temperature we will have the DAC 6 of the **ZCU111** connected to the readoutIN while

a mux firmware is loaded. This basically cause the bandwidth to go from 6 GHz to few MHz, is therefore required a local oscillator with an up-conversion scheme as presented in fig. 2.15. As it shown, the same local oscillator is also used for down-conversion.

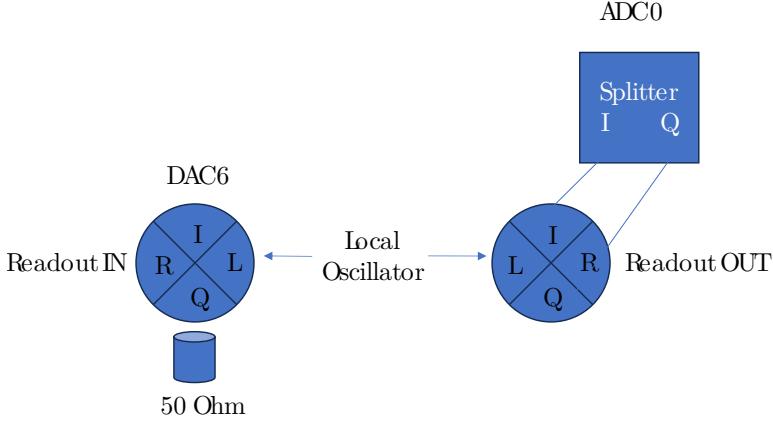


Figure 2.15: Scheme used for down and up-conversion.

Since this process uses an IQ mixer, in the down-conversion I and Q are splitted and gets merged back again using a splitter connected to the board ADC 0.

For the drive lines is sometimes needed additional amplification that was provided through a LNA. They were connected to DAC 3-4-5 to achieve full control of 3 qubits at the same time.

For the flux lines some more work is needed since the outputs of the companion XM500 board (that exposes the SMA connections) have baluns (i.e. DC filters) on its single-ended connection. There are two option to proceed: it is possible to buy/project a new companion boards or to leverage the 3 differential outputs of the XM500. This second option was chosen for simplicity. Note that the differential outputs P-N can be merged together via a simple subtraction P-N, something that standard differential amplifiers easily do, but that since we are aiming for DC currents these are not an option because they use capacitors and operates as high-pass filters. Special active differential amplifiers are needed in this case.

Having bought 3 of them, we were able to bias 3 flux-tunable qubits using DACs 0-1-2.

2.1.6 Configuration and setup for ZCU216

The **ZCU216** was used with the same setup composed for the **ZCU111**. This was a temporary solution that was also limiting the board potential.

In particular, the **ZCU216** could potentially be used to fully control 7 flux tunable qubits (considering 7 drive lines, 7 flux lines and a maximum of 2 readout inputs), while with the used scheme only three qubits were usable. To fully unblock the board potential, however, it is needed to make some changes to the **Qick** firmware and this requires particular work.

2.2 Developed Control Software

My thesis work was included in the development of **Qibolab** [37], the instrument-related component of the **Qibo** [38] framework.

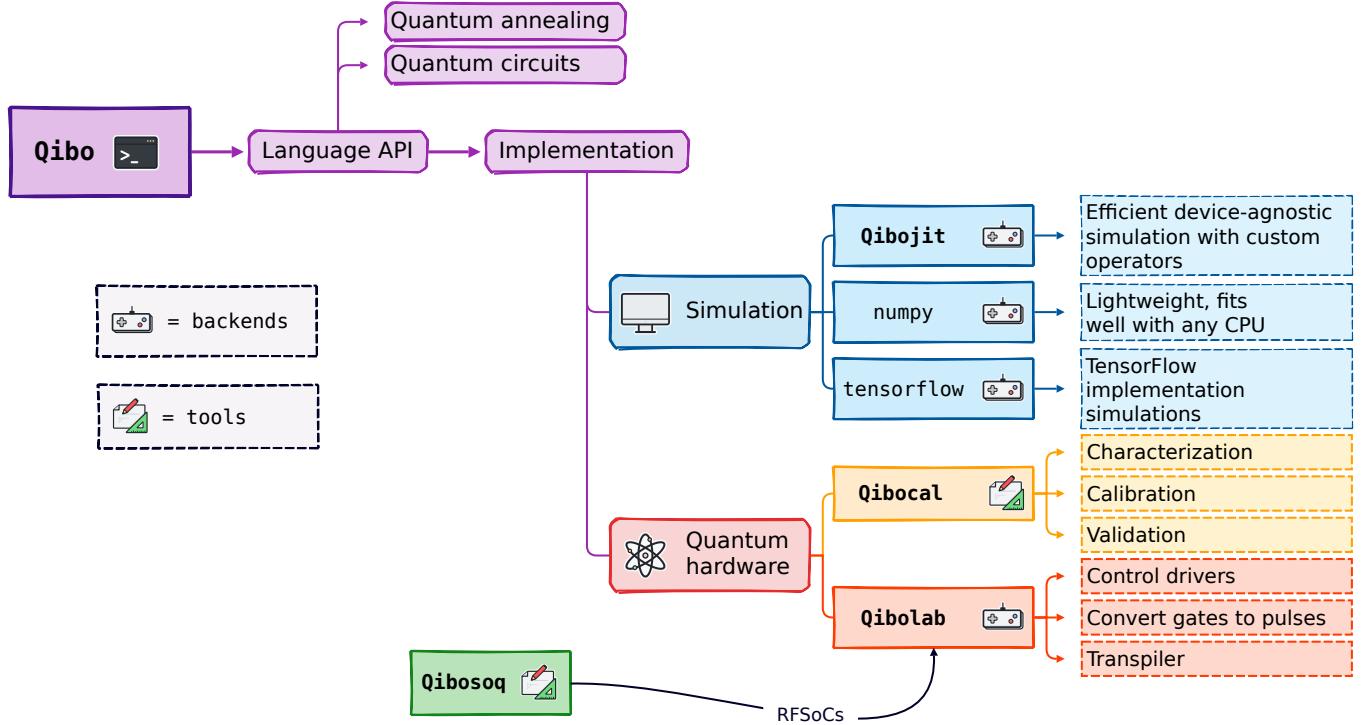


Figure 2.16: Schematic overview of **Qibo** software components.

Qibo, schematically illustrated in fig. 2.16, is an open-source [39–41] full stack application for quantum simulation that can be used to define and efficiently simulate circuits. It supports various backends for the execution of circuits: all of them are related to simulations, but one.

Qibolab is the backend dedicated to hardware control and for deployment of circuits on real-quantum-hardware. The idea behind **Qibolab** is that of an hardware-agnostic software for hardware control that can be used in different laboratories without any development effort.

Qibolab supports different control devices: in particular the systems of Quantum Machines, QBloq, Zurich-Instruments and, after my thesis, several RFSoC FPGAs.

In particular, I developed a two-side driver: composed of a client (the **Qibolab** component) and a server called **Qibosoq**. The latter is a server that runs on the FPGA

CPU present on the RFSoC boards and has the task of translating the programs received by **Qibolab** (that are in the form of sequences of pulses) to a **Qick** program. **Qick** contains both the firmware of the FPGAs and a low-level software to interact with the FPGA logic.

In this section I will give a general overview of the **Qibolab** project, its components and abstraction levels and I will provide some details on the implementation of the RFSoC driver as well as of the **Qibosoq** package.

2.2.1 Qibolab

Qibolab is the **Qibo** backend dedicated to hardware execution.

Ideally, **Qibolab** wants to be a universal control software for quantum computing, usable in the control of different qubit technologies by different electronics. The philosophy that is guiding the development of this tool is the *middleware* one, from RedHat.com:

Middleware is software and cloud services that provide common services and capabilities to applications and help developers and operators build and deploy applications more efficiently. Middleware acts like the connective tissue between applications, data, and users.

So that's the idea. In a moment in which, for quantum computing, no instrument is compatible with another one, where every technology and company implements a new programming language, where there is no convention between different constructors, **Qibolab** wants to be the glue.

The key features of the API (Application Programming Interface, namely the software abstractions and functionalities), as it is now, are:

- support for **Qibo** circuits deployment on quantum hardware;
- support for different instruments for custom lab setups and easy integration of new drivers;
- support for different heterogeneous platforms (qubit systems);
- re-usability of calibration experiments via **Qibocal**.

Qibolab is a complex library and is currently composed of roughly 25000 lines of Python code (79000 total lines including also documentation and tests), its main components are:

Platform this is the object that orchestrates different instruments in a laboratory, it describes a set of controlled qubits, controller device and any additional required device to coordinate;

Qubits objects that contain characterization/calibration parameters, information about native calibrated gates and the connected channels;

Channels physical wires in a laboratory that describe the setup and the connection among instruments and between the controller and the qubit;

Pulses extensive API to work with pulses and pulse sequences. It includes support of various shapes;

Sweepers these are objects crucial for experiments since they enable fast scans directly looping on FPGA logic;

Transpiler and Compiler two components that, working together, are in charge of converting a circuit to a list of pulses (first as a circuit of native gates and then as a list of "native pulses");

Instruments one of the most important parts of Qibolab, comprehensive support of different instruments under the same interface.

The main instruments, often called controllers, are shown in fig. 2.17, while a list of supported features is available in table 2.2.

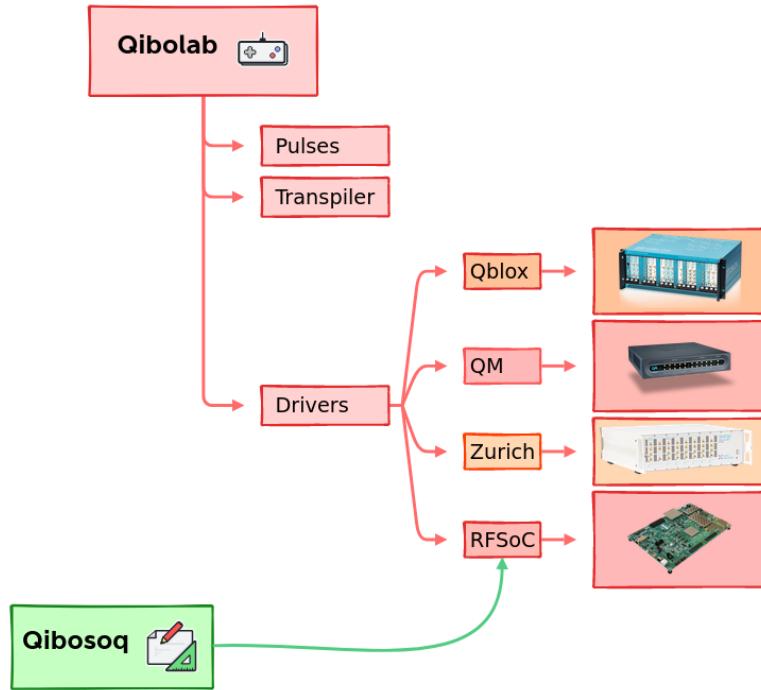


Figure 2.17: Schematic overview of Qibolab supported instruments.

The following is a description of the features presented in table 2.2.

Feature	RFSoCs	Qblox	QM	Zhinst
Arbitrary pulse sequences	✓	✓	✓	✓
Arbitrary waveforms	✓	✓	✓	✓ ⁴
Multiplexed readout	✓	✓	✓	✓
Hardware classification	✗	✓	✓	✓
Fast reset	■	■	■	■
Device simulation	✗	✗	✓	■
RTS frequency	✓ ⁵	✓	✓	✓
RTS amplitude	✓	✓	✓	✓
RTS duration	✗	✓	✓	✓
RTS start	✓	✓	✓	✓
RTS relative phase	✓	✓	✓	✓
RTS 2D any combination	✓	✓	✓	✓
Sequence unrolling	■	■	■	■
Hardware averaging	✓	✓	✓	✓
Singleshot (No Averaging)	✓	✓	✓	✓
Integrated acquisition	✓	✓	✓	✓
Classified acquisition	✓	✓	✓	✓
Raw waveform acquisition	✓	✓	✓	✓

Table 2.2: Features or limitations of the main drivers supported by Qibolab 0.1.0. The features denoted by “✓” are supported, “✗” means not supported and “■” under development.

Arbitrary pulse sequences the capability of executing arbitrary pulse sequences defined in Qibolab, which is a fundamental requirement of a driver. This feature is not related to the execution of pulses with arbitrary *waveform shapes*.

Arbitrary waveforms the capability of executing pulse waveforms of arbitrary shape. For drivers that do not support this feature, rectangular, Gaussian and Derivative Removal by Adiabatic Gate (DRAG [42]) waveforms can still be synthesized.

Multiplexed readout allows playing and acquiring multiple multiplexed pulses through the same line. It is particularly useful for multi-qubit chips where the readout line is commonly shared among multiple qubits.

Hardware classification the capability of doing single shot measurement classification *during the execution* of a pulse sequence.

Fast reset the capability of actively resetting the state of a qubit to zero after a measurement. This feature requires hardware classification and enables faster executions of repeated pulse sequences.

Device simulation the possibility of simulating in advance the pulses to be executed, without directly using quantum hardware.

RTS frequency RTS (*Real Time Sweeper*) refers to the capability of executing a pulse sequence multiple times with different values of, in this case, the frequency of a pulse. This feature facilitates faster qubit characterization and experiments.

RTS amplitude real-time sweeping of the amplitude of a pulse.

RTS duration real-time sweeping of the duration of a pulse.

RTS start real-time sweeping of the start time of a pulse.

RTS relative phase real-time sweeping of the relative phase of a pulse.

RTS 2D the capability of combining two RTS scans on different parameters.

Sequence unrolling the capability of converting a sequence repeated multiple times, in a minor number of longer sequences that each contains multiple measurements, in order to decrease the overall time spent on pulse compilation and communication to the devices.

Hardware averaging the capability of repeating the same experiment multiple times and obtain, directly from the device, averaged results.

Singleshot (No Averaging) the capability of obtaining from the devices all the non-averaged results.

Integrated acquisition the capability of acquiring complex signals [43] with "in-phase" and "quadrature" (IQ) components demodulated and integrated for the measuring time.

Classified acquisition the capability of performing 0-1 state classification after the integrated acquisition.

Raw waveform acquisition the capability of acquiring non-integrated IQ waveform values.

Qibolab was also object of a recent publication, in which the contribution of the work related to this thesis was not indifferent: CITE.

2.2.2 Qibosoq

A large part of my work for this thesis focused on the development of the required software to integrate RFSoCs into the **Qibo** framework through **Qibolab** driver. To do so, it was needed to build a comprehensive library that bridged the differences between **Qibolab** and **Qick** (that has the task of controlling the FPGA logic). This were the premises for **Qibosoq**.

Qibosoq is a an open-source server-side software package designed for RFSoC for executing arbitrary pulse sequences on self-hosted quantum processing units.

It's layout is essentially divided in three main elements:

Client: the user of the RFSoC, connected remotely via any network connection;

Server: the always-listening server running directly on the board. It takes care of receiving commands from the client and to execute the experiments;

Language API: a set of objects that define a common language between client and server.

The server role is to take commands and programs defined in the **Qibosoq** language and compile them as an executable that can be run by the **Qick tProcessor**: a timed processor on the FPGA logic that takes care of orchestrating firing pulses.

Qibosoq exposes various tools and abstractions to the user: the **programs**, representing **Qick** programs, and eventually taking care of the experiment compilation into the **tProcessor** language, the **components**, high-level structures used in the *programs* construction, the **server** implementation, and **client** utilities, to manage the communication.

The **programs** bridge the gap between the high-level interfaces (components) and the low-level execution on quantum hardware. Eventually, only two distinct programs are directly used, but the full hierarchy also includes intermediate abstractions. Considering all layers, the defined **programs** are:

- the abstract **base** program, that contains functions shared among all possible experiments and executions. It serves as the foundation for all the other **Qibosoq** programs;
- the abstract **flux** program, that collects the additional elements required for controlling flux-tunable qubits. In addition to the functionalities defined in **base**, it includes support for bias voltages and fast DC (direct current) pulses.
- the **sequences** and **sweepers** programs, that contain the different elements used, respectively, in the execution of fixed parameters pulse sequences and real-time sweeps. They inherit all the functionalities defined in **base** and **flux**.

The **components** play the crucial role of establishing a common language for communication, easing the implementation of a **Qibosoq** client in **Qibolab** or by other parties. The main elements defined within the **components** submodule include:

- the **Config** object that contains essential general information required for execution on hardware, such as the number of repetitions for the experiment and the waiting time between repetitions;
- the **Pulse** base object that serves as the foundation for different implemented pulse shapes. Rectangular, Gaussian and DRAG [42] pulses are natively supported, as well as custom waveform shapes defined by their "in-phase" and "quadrature" (IQ) [43, 44] values;
- the **Qubit** object that describes a qubit and holds information about any necessary bias required for its operation [45];

- the **Sweeper** and **Parameter** objects that are used to describe real-time on-hardware scans.

Feature	Qick	Qibolab	Qibosoq
Arbitrary pulse sequences	✓	✓	✓
Arbitrary waveforms	✓	✓	✓
Multiplex readout	✓ ⁶	✓	✓
Feedback	✓	✓	⚙
RTS frequency drive	✓	✓	✓
RTS frequency readout	✗	✓	✗
RTS amplitude	✓	✓	✓
RTS duration	✗	✓	✗
RTS start	✓	✓	✓
RTS relative phase	✓	✓	✓
RTS N-Dimensional	✓	✓	✓
Hardware averaging	✓	✓	✓
Singleshot (No Averaging)	✓	✓	✓
Integrated acquisition	✓	✓	✓
Raw waveform acquisition	✓	✓	✓

Table 2.3: Main features and limitations of **Qick**, **Qibosoq** and **Qibolab** compared. The features denoted by “✓” are supported, “✗” means not supported and “⚙” under development.

The last two fundamental elements are the **client** and the **server**. The **client** is composed of a set of tools used to connect to the server, convert components into a serialized form, and send them following the **Qibosoq** communication protocol. The **server** implements the on-board server, continuously listening for connections, and executing received instructions by initializing and running the required programs on the quantum hardware.

In analogy of what was presented for **Qibolab**, in table 2.3 are presented the main features supported by **Qibosoq** in comparison with what is supported by **Qick** and **Qibolab**.

Qibosoq was developed for this thesis and is now a small complete library of 2500 lines of python code (total of 8000 lines) has been recently object to arXiv publication: [46].

Chapter 3

Single qubit characterization and calibration

The focus of this thesis is the characterization and calibration of superconducting qubits [26].

The two words, *characterization* and *calibration*, are often used as synonyms, but it is useful to introduce a difference:

- characterization means finding the values of *intrinsic* properties of the qubit (for example it's resonance frequency, it's T_1 ...);
- calibration means finding the optimal parameters to *control* the qubit.

The following parameters are the main ones to be obtained in characterization experiments:

Qubit resonance frequency The qubit resonance frequency refers to the specific frequency at which a qubit undergoes a transition between its first two energy levels. This parameter that determines the operational frequency of the qubit.

Resonator resonance frequency The resonator resonance frequency is the natural frequency at which a resonator oscillates. This frequency is dependent on the physical characteristics of the resonator and is essential for readout operations.

Q factors Quality factors measure the damping of oscillations in a resonant system. A high Q factor indicates minimal damping and high energy efficiency, while a low Q factor indicates significant damping. In the context of quantum systems, a high Q factor is desirable for maintaining coherence and prolonging the lifetime of quantum states. It is defined as $Q = \frac{f_r}{\Delta f_r}$

Coupling g The coupling parameter g represents the strength of the interaction between two quantum components, such as a qubit and a resonator. It determines the rate at which energy is exchanged between the components.

Dispersive shift The dispersive shift is a phenomenon in quantum systems where changes in the energy levels of a qubit are observed due to its interaction with a resonator. This shift depends on the state of the resonator and can be used for qubit readout and manipulation.

Anharmonicity Anharmonicity refers to the nonlinearity of the energy levels of a quantum system and signifies that the energy difference between different quantum states is not strictly equal.

Relaxation time T_1 The relaxation time T_1 characterizes the time it takes for a qubit to return to its ground state from an excited state. It quantifies the qubit's tendency to lose energy and coherence, often due to interactions with the environment.

Dephasing time T_2 The dephasing time T_2 represents the duration during which a qubit can maintain coherence without undergoing energy-changing transitions. It measures the qubit's resilience against fluctuations and noise in its environment.

As we can see, characterization parameters are intrinsic properties of the qubit system and come mostly from fabrication.

On the other hand, we have calibration experiments that analyze optimal control parameters:

Drive frequency The optimal frequency of the pulses required to change the state of the qubit.

Drive amplitude The optimal amplitude of the pulses used to change the rotate the state of the qubit on the Bloch sphere, as well as the mapping between amplitudes and rotated degrees.

Drive duration & shape The optimal duration and shape of the control pulses to change the qubit state as required and minimizing leakage to higher excited states.

Assignment fidelity Parameters to classify the states, zero and one, from IQ values as well with assignment fidelities.

Time of flight time required for a signal to travel from the RFSoC to the qubit and back, required for pulse control and readout.

Readout frequency The optimal frequency of the pulses required to read the state of the qubit.

Readout amplitude The optimal amplitude of the pulses required to read the state of the qubit

Readout duration & shape The optimal duration and shape of the pulses required to optimize the read-out the state of the qubit

Gate fidelity The accuracy relative to specific operation that differ from an exact matrix, by some measurable factor.

The difference is often minimal and many experiments fall in both categories at the same time, but nevertheless the difference is substantial. In particular, the objective of an experimenter may be just in characterization, so with no interest in optimizing the control of the qubit, or just in calibration, with only interest in reaching good control without interest in knowing parameters such as quality factors or coupling values.

In this chapter (and this thesis), I will try to give a description and some examples for all the most important experiments in single qubit characterization and calibration. Note that the list hereby provided is not exhaustive at all and, in particular for calibration, the possible experiments are almost unlimited.

The main sources for this chapter, that are maybe more precise and advanced than this thesis are [47–50].

This chapter, on the other hand, tries to present the experiments in the clearest possible way, so that they can easily reproduced. For every experiment, ideal and real-case plots are provided so that the reader can have a better grasp of the differences between the ideal situation and the reality.

Moreover, at the end of every experiment a short recap will be provided, in the form of a blue color box.

In the case of flux-tunable devices, some more experiments are needed: these experiments will always have names starting with "Flux ..." and will be differentiated by the others also by the use of a red recap box. These experiment can/have to be avoided in the case of non-flux-tunable qubit characterization.

In fig. 3.1, an outline of the main experiments described in this chapter is presented as per the proposal for the auto-calibration module of **Qibocal**. Several experiments are mentioned in the scheme and not yet available, so not described in this thesis.

Note that, while they are in order of priority (top to bottom), characterization and calibration **is not** a sequential process. It will happen many times, as an experimenter, to go back in the experiments tree and repeat some procedure, to better optimize some parameters.

The burden of understanding when to repeat an experiment is necessarily left to the reader/experimenter and the apparent sequentiality of this chapter is just a matter of simplicity.

A note: the majority of the plots presented in this chapter are produced by **Qibocal** and shown here complete of a left amplitude plot and a right phase plot. Our discussion will always focus only on the amplitude plot for simplicity, but it has to be clear that the majority of information can also be extracted from the phase plot. In the end, there is direct amplitude measured, but always two I-Q values from which we can compute amplitudes and phases.

Autocalibration 0.2

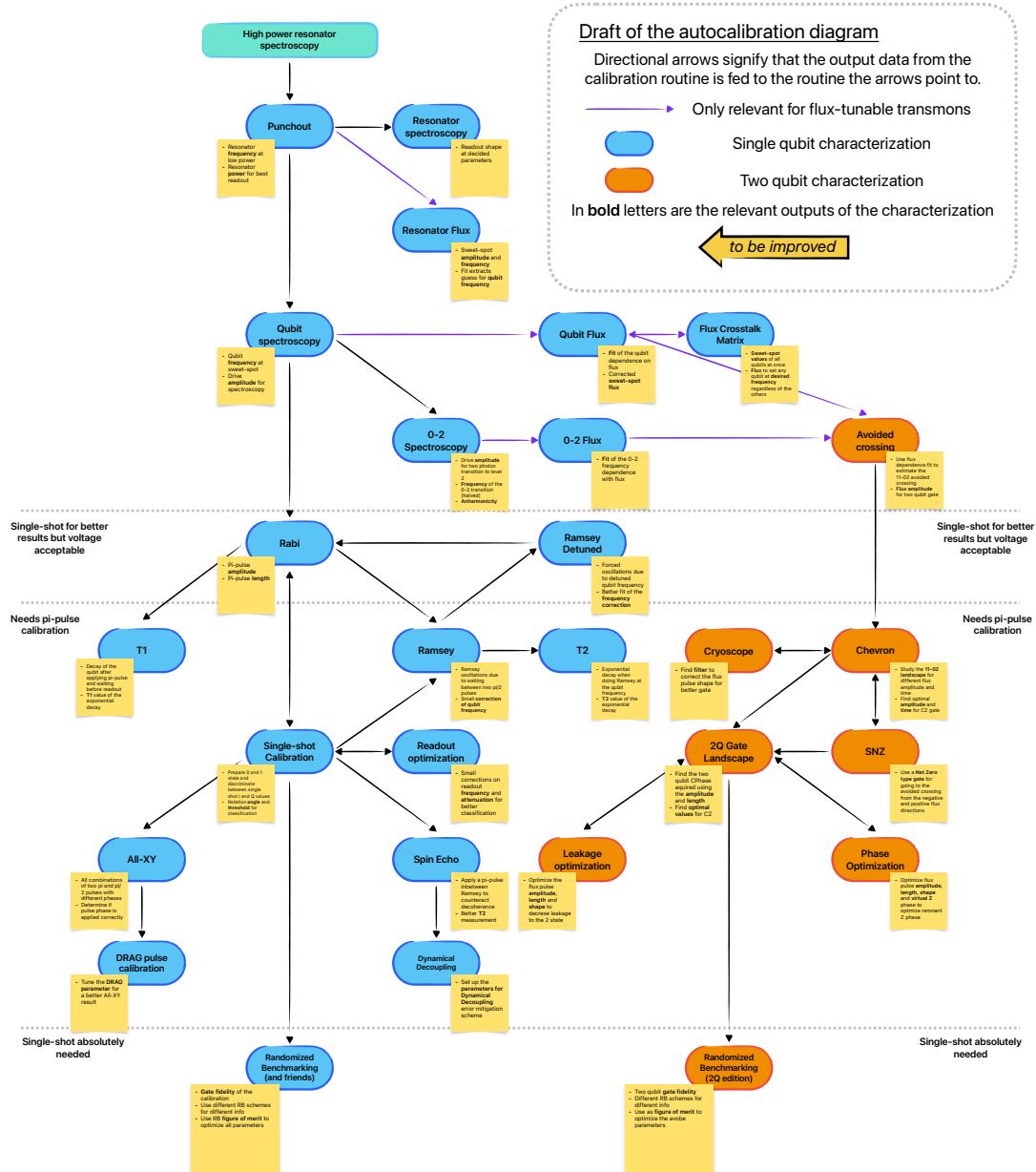


Figure 3.1: Scheme of the characterization and calibration experiments as presented for Qibocal auto-calibration module.

3.1 Time of flight measurement

The first calibration experiment, on top of which all other experiments will build on, is the *Time Of Flight* (TOF) experiment.

The TOF is defined as the time that a signal produced by the RFSoC needs to reach the acquisition system, after having traveled to the core of the cryostat through the readout line. The starting and ending point of the signal are, in the case of a RFSoC-based system, included on the same board but in general these could be two different devices. The TOF is not a characteristic of the qubit (or of the resonator coupled with it), but rather of our experimental setup: in particular, of the length of the lines used to carry the signals and of their material.

When measuring the state of a qubit, we first send a readout pulse of fixed length that changes in amplitude (and phase), depending on the qubit state, since the signal gets partially absorbed/amplified if it is at the same frequency. We then want to start acquiring, through the ADC, the "returning" waveform.

This acquisition generally last a time equal to the pulse length.

The timing of the acquisition, in particular of the delay between the firing of the readout pulse and the start of the acquisition is crucial and extremely relevant. If we start acquiring too early, the initial part of the acquisition will be empty, since the pulse has not yet reached the FPGA. If we start acquiring too late, we will have the same problem for the last part of the acquisition.

Since we are talking about readout, this experiment is sometimes referred to as *time of flight readout*, since the same idea can also be applied for the drive.

A possible solution to be sure of capturing the whole signal would be to acquire for a longer period of time but, in that case, a certain percentage of our acquisition will be identical for the two different qubit states, reducing the discriminating power of the readout system. Moreover, the acquisition systems have all a finite memory allocated for a single readout and a longer acquisition can easily become problematic.

Because of that, a carefully conducted TOF experiment will lead to better fidelities at the end of the qubit calibration, making it easier to differentiate between the ground and excited state.

The experimental procedure is fairly simple: we send a fixed length readout pulse and immediately start acquiring. When plotting the acquired amplitude against time, we will initially see nothing. When the pulse finally reaches the ADC, after the TOF time, we will roughly see the pulse waveform or, at least, a difference in amplitude.

An example of a ideal plot is presented in fig. 3.2.

Note that the unit of measurement for the x axis, the time axis, is in ADC *clock ticks*. This is because, in general, it is not really relevant the s/ns conversion ¹.

Of course, the plot presented in fig. 3.2 is an ideal one: in a real case scenario, we will have a much noisier signal and the edges of the rectangular pulse (both the rising and falling edges) will never appear "vertical").

To counter the effect of noise, we can repeat the experiment multiple times (usually called

¹In any case, we roughly have 4 clock ticks ≈ 1 ns

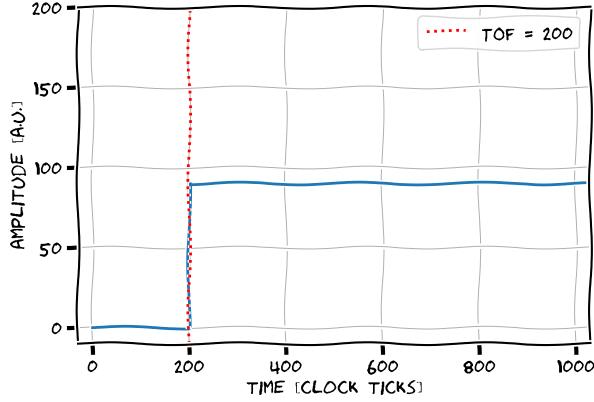


Figure 3.2: Plot of a generic time of flight experiment.

shots), averaging the acquired waveforms. We can also increase the initial amplitude of the pulse, however this has to be done carefully, since too much power could damage the components.

In any case, it is not important to be extremely precise with the time of flight estimation.

Note, anyway, that this experiment can be done with much more precision in later stages of the characterization: in particular after finding the resonator frequency.

Consider for example a system with a cavity, if we send through it a pulse that is not on resonance, the signal will mostly get absorbed and the acquired wave will need a lot of averages to become clearer. If we know the exact resonator frequency, on the other hand, the signal will be highly amplified and the experiment will be easier.

In fig. 3.3 a real plot is presented. We can see that the tone we sent (in this case at the resonator frequency) some effects can happen that distort the signal. However the shape sketched in fig. 3.2 is still clearly visible.

From this, a TOF value of 200 was chosen as the time offset between readout pulse firing and acquisition.

The oscillating effect present in fig. 3.3 has different possible explanation, but can sometimes be linked to the use of a frequency (for the pulse) just slightly different from the resonance one. In this case, the signal is amplified but with an initial detuning effect. For a cavity, when this detuning is missing, we can expect something like fig. 3.4, where we can see the characteristic charging and discharging effect of the cavity that behaves as a capacitor.

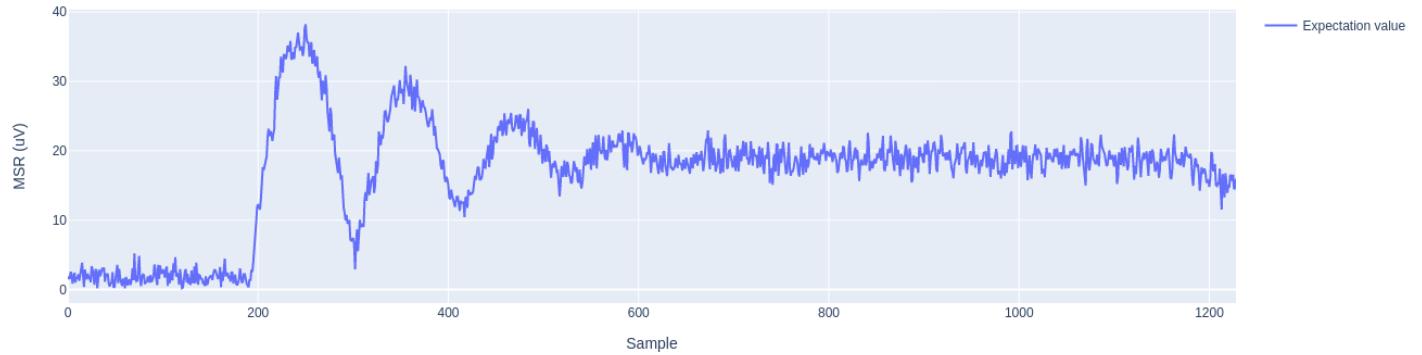


Figure 3.3: Plot of a time of flight experiment.

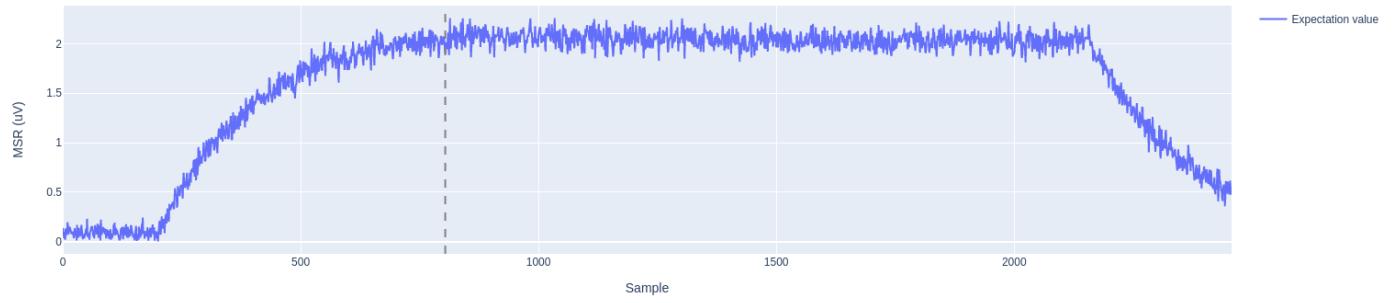


Figure 3.4: Plot of a time of flight experiment. the vertical line is merely the effect of a badly implemented fitting procedure.

Experiment recap: Measurement Time of Flight

Scope:

readout calibration.

Parameters to extract:

time of flight (adc offset).

Brief description:

a readout pulse gets fired, the ADC acquisition starts immediately. We plot acquired amplitude Vs time, we extract the TOF as the time where we visibly see a change in amplitude.

3.2 Resonator spectroscopy (bare frequency)

After calibrating the time of flight parameter, we can start calibrating the readout pulse through resonator characterization.

The first thing to do is to find the resonator frequency, that is to say its transition frequency.

At this frequency we will be able to observe a clear difference in the transmitted signal: if the resonator is a 3D cavity we will observe an amplified signal, while for a 2D resonator we will observe a higher absorption. In both cases, we expect to see a Lorentzian peak (positive for 3D cavity or negative for 2D resonators):

$$y = y_0 + \frac{w^2}{4(x - x_0)^2 + w^2} \quad (3.1)$$

Where w is equal to half of the peak width.

In the experiment, we send a readout pulse with fixed duration and amplitude and, after waiting for the TOF, we acquire a waveform that we average, obtaining a single point. We then repeat the experiment for different frequencies, ideally obtaining a plot like the one sketched in fig. 3.5.

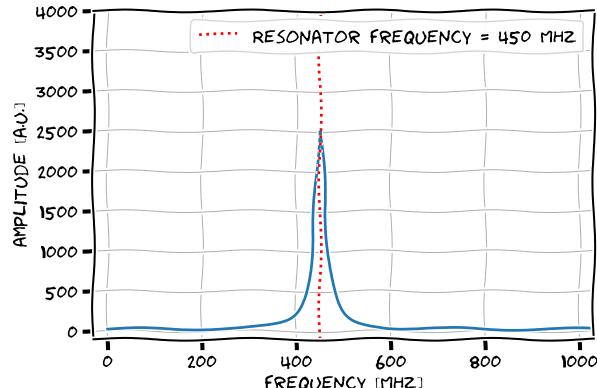


Figure 3.5: Plot of a generic resonator spectroscopy experiment.

In the time of flight experiment we didn't really have many variables, for the resonator spectroscopy, on the other hand, the situation is very different. In particular, we are extremely dependent on the amplitude of the pulse and on its duration². Since the objective of this experiment is to find the resonator frequency, without any readout optimization (something that we will have to do afterwards), we can fix the duration of the pulse at $2\mu s$.

For the amplitude the discussion is slightly more complex and there are several elements to take into consideration:

²Note that, for the readout pulse we usually use a rectangular waveform. For the control pulse, the shape is a much more relevant parameter, that will require a specific calibration.

- higher amplitudes usually correspond to better S/N;
- at high amplitudes the signal breaks superconductivity, therefore resonator is not effectively coupled to the qubit (we talk of **bare resonator frequency**);
- at intermediate amplitudes the peak could completely disappear and is, in general, not Lorentzian;
- very high amplitudes could damage the components³.

In the first stages of characterization, we don't need to be at low power to operate: even if the resonator is not coupled to the qubit it is not a problem⁴. Indeed it is preferable to have higher S/N so that the peak is clearly identifiable. With standard attenuation inside the fridge (-55 dB + line resistance), a reasonable voltage could be around 0.5 V. However, different instruments have different way of controlling the power: some may offer attenuation control, some just a linear amplification. The experimenter will, inevitably, have to decide power levels dependently from its setup.

Another parameter connected to the amplitude, is also the *relaxation time* (in some literature also referred to as *repetition duration*) and the number of shots. The number of shots represents the number of repetitions of the same experiment (at the same frequency), while the relaxation time is the waiting time between repetitions. A higher number of shots will increase the S/N ratio by averaging the noise, but will also slow down the acquisition. As per the relaxation time, for this experiment in particular we can leave it at zero: since we are not exciting the qubit we do not particularly care about it. However note that, for 3D cavities, we could end up damaging the qubit if we send too much energy over a small period of time so it could be worth to increase the relaxation time.

Last but not least, we have to choose which frequencies are probed during the scan: a very wide scan can be useful if nothing is known about the studied resonator, but in general we have at least the design parameters. These are often not exact, but can give an idea of the region to scan (for standard cavities around 7 GHz). Also, a very small step between two subsequent frequency points is not needed and could really slow down the experiment (from seconds to tens of minutes) if chosen incorrectly. Usually, a step of 200 MHz is fine enough.

A plot for a resonator spectroscopy experiment is presented fig. 3.6. This plot was obtained probing a 3D cavity resonator with a **RFSoC4x2** board that, since can synthesize frequency up to 9.8 GHz, didn't require any external local oscillator for the upconversion.

The use of a 3D resonator is a special case that generally leads to better S/N, as well as less overall noise and a horizontal background, in respect to planar chips.

³This is especially true in the case of a 3D cavity, where the delicate qubit is inside the cavity and the line is shared both by control and readout.

⁴By "not coupled with the qubit" we mean that, since we send a huge number of photons through our readout line, we are essentially exceeding the Josephson junction critical current and "turning off" the qubit.

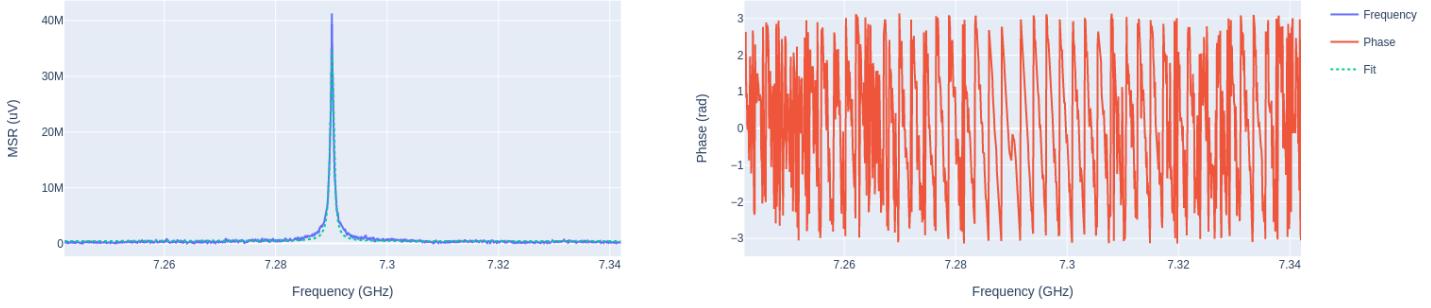


Figure 3.6: Resonator spectroscopy, for a 3D cavity.

For example, in fig. 3.7, we can see a resonator spectroscopy done on a chip namely containing 5 different multiplexed⁵ resonators (was not possible to identify one of them). The experiment was conducted with a **ZCU111** board, with a firmware that enables multiplexed readout (so the use of a common DAC and ADC for the simultaneous readout of multiple resonators), with the drawback of a limited bandwidth. Because of that, a local oscillator is needed along with an IQ mixer. Unfortunately, this configuration carries some problem, most notably:

- the background is noisier and less horizontal;
- the upconversion scheme creates multiple "images" of the upconverted signals;
- background-resonance interaction can severely change the shape of the peak (from a Lorentzian to a Fano resonance [51, 52]);
- the frequency value of the local oscillator has a strong effect on the overall background.

All of these phenomena can be seen in fig. 3.7 (resonator 4, for example, is a Fano resonance, while resonator 3 isn't).

Common problems that could increase the difficulty of this experiment generally involve the power level (amplitude) set for the pulse, that could be too low to clearly see the peak, or the use of very large frequency step that could reduce the number of samples actually part of the resonance.

Note that this experiment doesn't really require the use of a FPGA or a RFSoC and can be initially performed with a VNA (generally easier/faster to use). However, being able to perform all the experiments without needing to change the hardware setup can be extremely beneficial.

⁵By multiplexed we mean that the resonator are coupled to the same readout line and can be probed simultaneously.

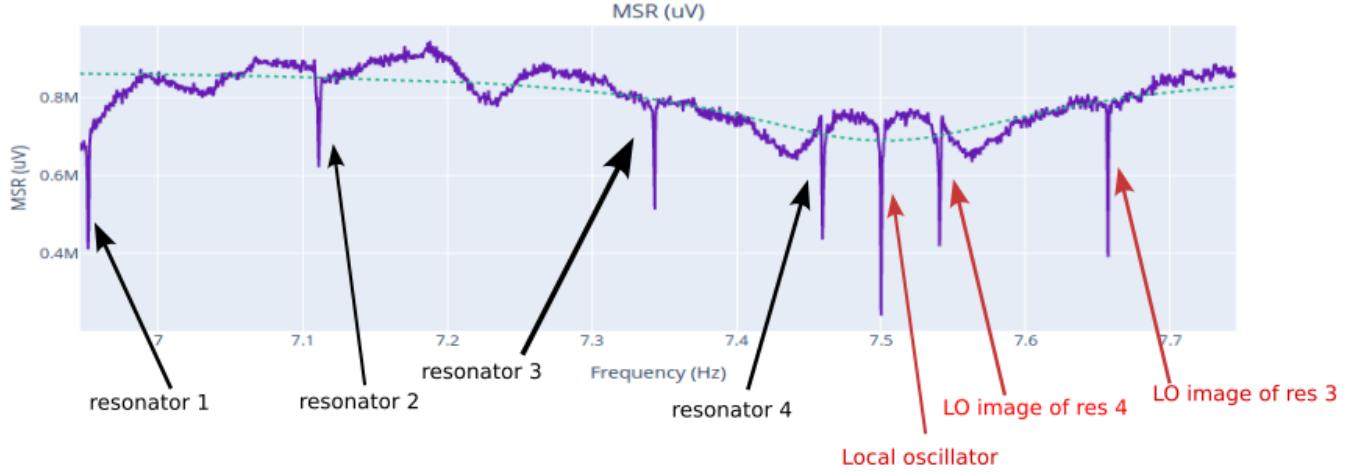


Figure 3.7: Resonator spectroscopy, for multiple multiplexed 2D resonators with a local oscillator for upconversion.

Focusing on a single peak, both for the 3D and the 2D case, at this point we should have roughly found the resonant frequency: that is the main objective of this experiment. Although it is not strictly required, we can fit the peak with a Lorentzian/Fano resonance on a background, to extract the frequency with more precision and to compute the resonator quality factor *in the high power regime* that can be defined equivalently as:

$$Q = \frac{f_r}{\Delta f_r} = \frac{\omega_r}{\Delta\omega_r} = 2\pi f_r \frac{\text{energy store}}{\text{power loss}} \quad (3.2)$$

Where f_r is the resonator frequency, Δf_r the peak width and $2\pi f = \omega$.

These information are not strictly required for qubit calibration, but could help to understand the overall quality of the system.

Experiment recap: Resonator spectroscopy (high power regime)
Scope:

readout calibration.

Parameters to extract:

bare resonator frequency,
resonator quality factor (in high power).

Brief description:

a measurement (readout pulse fired, averaged acquisition after TOF) is executed for different frequencies. We plot acquired amplitude Vs frequency, we look for a Lorentzian peak (or Fano resonance) and extract the bare resonator frequency as the minimum (2D resonators) or maximum (3D cavity).

3.3 Resonator punchout

With the last experiment we found the bare resonator frequency (also called *high power frequency*), but this is not the frequency that will be used in measurements. We need to find the frequency in the low power regime, where the resonator is actually coupled to the qubit [53]. To do that, we first have to find the correct amplitude of the readout pulse.

Also this experiment can be initially performed with a VNA, to check everything is working and to obtain a rough estimation of the parameters to be used by the control devices.

We do again a resonator spectroscopy experiment, measuring at different frequencies, but this time scanning in a narrow frequency span and also for different pulse amplitudes. We expect the resonator frequency to have a strong dependency on the amplitude: in particular we expect it to be fixed at first (in the high power regime) then undergo a transition phase and then be fixed again at a different frequency because of the coupling with the qubit. Eventually, we want to have a plot like fig. 3.8

This experiment is the first one where we actually are "seeing" the qubit and it's extremely important also to check that the qubit is working properly. During a characterization, various experimental problems can happen and can lead the experimenter to believe that the qubit is no longer working: this experiment gives us an easy way to check it.

Moreover, from this experiment we can actually already have an estimate of the qubit frequency [26, 54, 55] using:

$$\omega_{rh} - \omega_{rl} = \chi = \frac{g^2}{\Delta} \quad (3.3)$$

Where $\Delta = \omega_{rl} - \omega_q$.

So, if we know, maybe from design specifications, the expected value of g , we can have an estimate of the qubit frequency. If, as in most cases, we don't have information

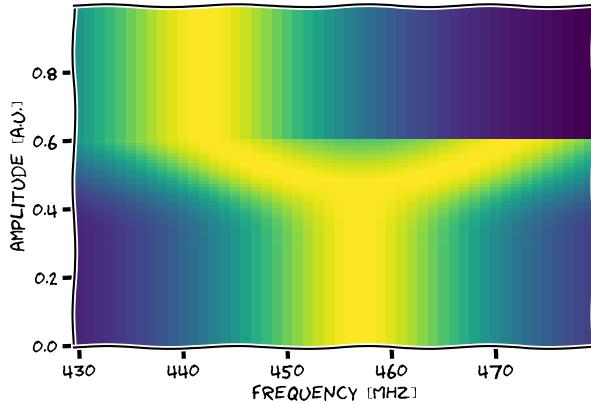


Figure 3.8: Expected plot for a resonator punchout experiment. From this we could choose an amplitude of 0.3 at ≈ 480 MHz.

on g , than we still can infer if the qubit frequency is higher or lower than the resonator frequency.

To obtain a clear plot, this experiment usually requires multiple tries, to choose "good parameters". In particular:

- the amplitude range and step must be chosen carefully because punchout can lead to extremely long experiments (that we would like to avoid), but large steps will inevitably confuse the plot;
- the frequency usually shifts very little, but depending on g and Δ , so it is difficult to set the frequency span and step;
- if the scan in amplitude is linear, it will be difficult to have a clear view of all the three regimes at the same time. If it is possible, it may be worth to do a logarithmic scan.

Note also that, in literature, this plot is often presented as a scan in the *attenuation* of the readout line. The effect is the same as of changing the amplitude (although it is by default a logarithmic scan), but the plot is reversed: at the top (high attenuation) we will see the low power regime and at the bottom (low attenuation) the high power one.

In fig. 3.9 different punchout plots are shown, so that is understandable how much they can differ. Note that "the colors are normalized" for every amplitude and we actually have very high amplitudes in high power, and much lower amplitude elsewhere.

From these plots we can extract few things: first of all, the pulse amplitude that we will use for the next experiments for the readout pulse. We should try to use the amplitude that maximize the S/N ratio for the resonator, so in general the highest amplitude at low power. However, we still need to be sure to not enter the transition

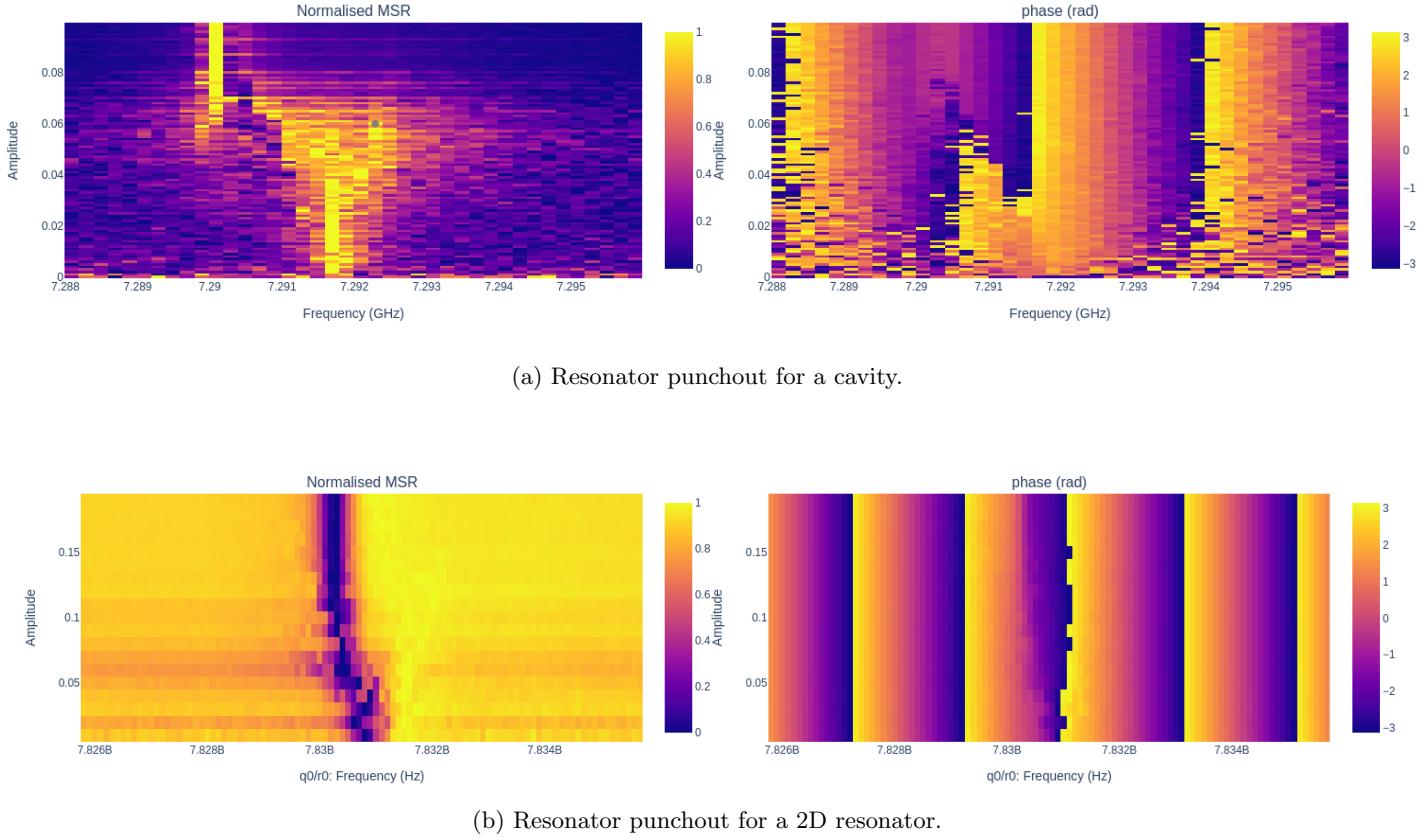


Figure 3.9: Different plots for the resonator punchout experiment.

regime, something that could lead to very noisy experiments and eventually prevent us to reach any sensible result. To be sure that, at the given pulse amplitude, we are not in the transition regime, we can repeat the standard resonator spectroscopy with a finer scan, checking if the peak is Lorentzian.

A resonator spectroscopy is also useful to better check the resonator frequency at low power that will present with a lower quality factor as shown in fig. 3.10.

Last but not least, we can write down also the maximum value of the peak.

Since we are not interacting directly with the qubit, we are effectively measuring the amplitude of the ground state and, from now on we will not change the resonator frequency, so we expect to always measure this amplitude for the zero state. Here it is not needed to do a precise measurement, but it is nevertheless very useful to have an approximate value, that can later be used to check, in other experiments, that we are not exciting the qubit by error (if we see a change in amplitude, then maybe the qubit state has changed).

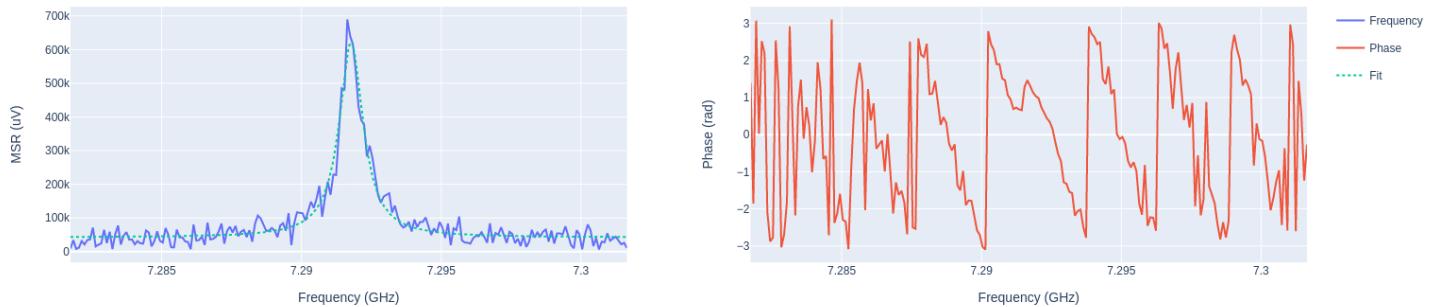


Figure 3.10: Resonator spectroscopy of a 3D cavity in the low power regime.

Experiment recap: **Resonator punchout**

Scope:

readout calibration.

Parameters to extract:

amplitude for the readout pulse,
frequency of the resonator in low power (frequency of readout),
signal amplitude for qubit ground state,
estimation of the qubit frequency.

Brief description:

a measurement is executed with pulses of different frequencies and amplitude. We do a 3D plot with transmitted signal vs frequency vs amplitude. We extract amplitude and frequency for low power regime. We estimate the qubit frequency and the signal amplitude of the qubit ground state.

3.4 Flux resonator spectroscopy

This experiment regards only flux-tunable qubits since it studies the flux dependency of the resonator frequency.

As we saw in section 1.2.2, for flux-tunable qubits there is a bias level, called *sweetspot*, where the qubit frequency is almost non dependant from flux oscillations. It's extremely important to find, at least roughly, this bias level before doing qubit spectroscopy (so before finding the qubit frequency) otherwise the qubit will have a completely different frequency than the one at normal working bias (we are talking of differences of hundreds of MHz or even GHz).

Because of that, we can proceed with a variant of a resonator spectroscopy experiment

where we also change the bias [56]. Note that the resonator is not directly affected by it, but it changes its resonance frequency in accordance to the qubit one that is directly dependent on the bias.

Since we are trying to exploit the coupling between qubit and resonator, we need to operate in the low power regime.

The experiment, repeated for a certain number of shots, for different frequencies and biases, will have the following layout:

- we send a constant DC current at the qubit through the flux line. The voltage amplitude of this "pulse"⁶ will be the bias level;
- after waiting few nanoseconds to be sure that the qubit is stable, we can measure the resonator at the amplitude level found with the punchout experiment;
- we set again the bias to zero (this is extremely important because DC currents tend to heat up the cryostat).

We expect to see a parabola-like dependence as in fig. 3.11. From this plot we can

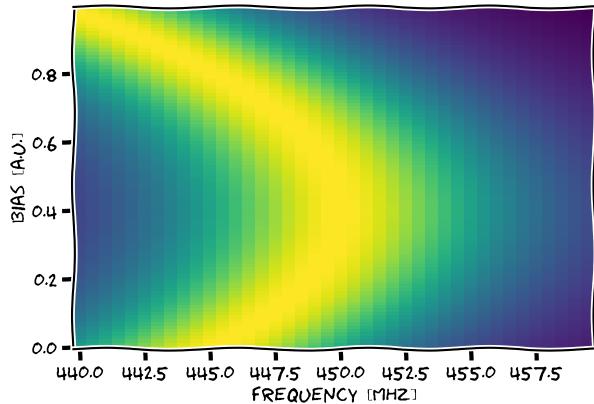


Figure 3.11: Expected plot for a flux resonator spectroscopy experiment.

set the sweetspot level at ≈ 0.4 [a.u.]. Note that we do not have a great resolution and we will later have to execute a similar experiment with the qubit resonance, that is much more dependent on the flux.

A real plot is presented in fig. 3.12.

Something that is also very important is to consider all the qubits at once (if there is more than one). We need to set all qubits at their sweetspot even for experiments that do not concern them, because they influence and change the sweetspots of all the other qubits. For example, if we measure qubit_0 alone we could find a sweetspot of 0.5, but when we measure it at the same time of qubit_1 (that has a non-zero sweetspot), then the first sweetspot shifts to 0.2.

⁶This can be done by connecting a constant source or, as in our case, sending a long constant and not modulated pulse.

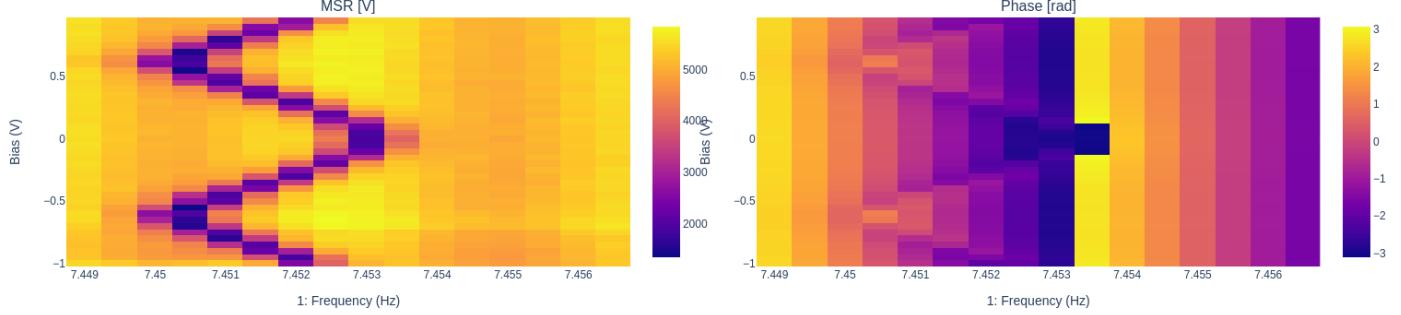


Figure 3.12: Plot a the flux resonator spectroscopy experiment.

Experiment recap: Flux resonator spectroscopy

Scope:

readout calibration, flux dependence calibration.

Parameters to extract:

estimation of the flux sweetspot,
resonator frequency at the sweetspot.

Brief description:

a measurement is executed for different frequencies and different biases (fixing the amplitude). We do a 3D plot with transmitted signal vs frequency vs bias. We extract bias and frequency for the sweetspot (highest frequency).

3.5 Qubit spectroscopy

The past experiments were all regarding *readout calibration*, from those we extracted the main parameters required for measurements:

- waiting time for acquisition (*time of flight*);
- frequency of the readout pulse (so of the resonator);
- amplitude of the readout pulse;
- estimation of ground state;
- estimation of the *sweetspot*. For flux tunable systems every experiment will be conducted at the sweetspot unless otherwise stated.

With these components we can proceed to look directly for the qubit that for now we saw only indirectly with the *punchout* experiment (and in flux resonator spectroscopy).

If the resonator spectroscopy experiments is a *single-tone spectroscopy*, to find the qubit frequency we need a *two-tone spectroscopy*. The idea is to send a first tone through the qubit drive line and then execute a measurement (so a second tone). The first tone will be changed in frequency and won't have any effect on the qubit state unless the frequency matches the qubit transition one. Looking at eq. (1.35) we will have $\Delta_d^2 \gg A \rightarrow P_1(t) = 0$.

However, if the drive frequency is "near" the resonant qubit frequency, the qubit will get excited, thus changing the amplitude of the acquired signal.

The qubit spectroscopy experiment [57, 58] is very similar to the resonator one: we fire a drive pulse at a certain frequency; *after* it, we perform a measurement, we wait for the qubit to relax, we repeat the experiment at different frequencies and then plot drive frequency vs measured amplitude, looking for a Lorentzian (downward if the resonator was upward and vice versa) as in fig. 3.13.

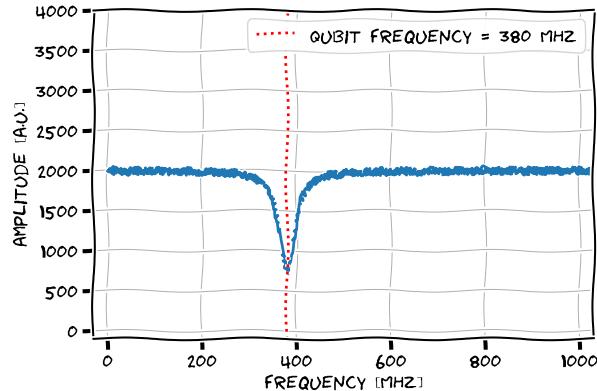


Figure 3.13: Sketch of the plot expected for a qubit spectroscopy experiment.

As always there are several variables and phenomena to take into account:

- the drive pulse will be, at the end of the calibration, a DRAG pulse with a length of ≈ 40 ns, but here we are interested in calibrating only the frequency: therefore we use rectangular pulses with length of $1 - 2 \mu\text{s}$;
- it is important to wait for the drive pulse to end so that we avoid the *Stark effect/shift* [57], but we should not wait any time more than strictly necessary. Because of that it is important to have all the cables of the same length;
- if we send too much power we could end up exciting the qubit even at "wrong" frequencies. We can check this by looking at the average amplitude measured: if it is not at the same level of what we defined in the resonator spectroscopy experiments as the ground state, than we are probably always exciting the qubit and we should reduce the amplitude of the drive. At the same time if we do not see any excitation we could need to increase the amplitude;

- having a large relaxation time is not strictly needed, in the sense that we could even initially perform this experiment without waiting for relaxations. However this will lead to not exact results and to an asymmetric Lorentzian: in fact we will excite the qubit with frequency A, measure the excited state, then apply frequency B that wouldn't excite the qubit, but still measure the excited state. Anyway, having a way of executing a fast version of the experiment may still be useful;
- the frequency step of the scan is clearly very important. A step of 100 MHz should be enough fine;
- it's important to be aware of potential aliasing phenomena and signal images caused by local oscillators or DDS. For every found peak深深, always ask yourself if it can be caused by these phenomena and, eventually, exclude it using the proper filters.

Without knowing anything about the qubit frequency, this experiment is not particularly easy: it can happen that the scan is at the right frequencies, but that there is no excitation for different reasons (for example the drive pulse has not enough amplitude). Or that we focus on the wrong frequencies because in some scans there seem to be a peak (maybe because of aliasing or just noise).

In fig. 3.14 we can see a first, fairly unfortunate, qubit spectroscopy done for a single qubit chip with the **RFSoC4x2**. See that the S/N ratio is not good at all, the peaks are very asymmetric (since the relaxation time was set to 0) and, clearly, there are two peaks for what should be a single qubit.

Multiple peaks can be caused by different phenomena, but here it is easy to see that the two peaks are equally distant (and opposite) from $f_s/2$ (half of the DAC sampling frequency) so we are actually seeing the effect of aliasing (as described in section 2.1.2). To check which one of the two peaks is the correct one, to properly characterize the

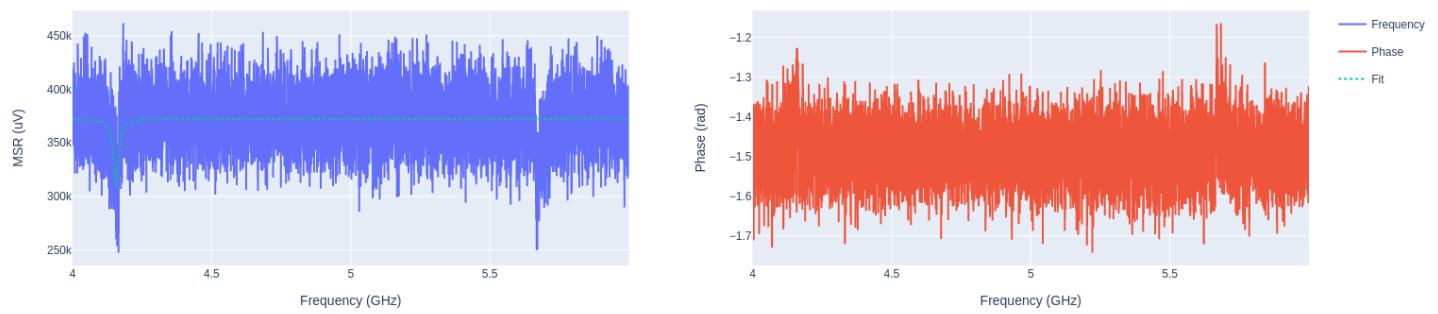


Figure 3.14: Example of a qubit spectroscopy experiment with clear aliasing.

qubit, the experimenter can add a frequency filter to remove one of the two peak: if it

does not disappear, it's the image.

Usually, anyway, just a single peak will be present, as in fig. 3.15.

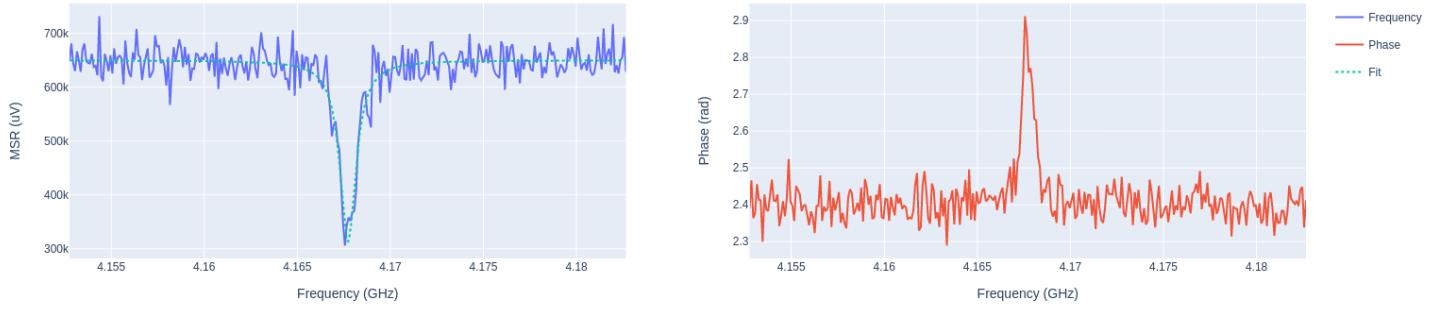


Figure 3.15: Example of a qubit spectroscopy experiment.

Note that now, having both the qubit frequency and the resonator frequency for high and low power, we can also get the value of the coupling factor from the dispersive shift relation:

$$\omega_{rh} - \omega_{rl} = \chi = \frac{g^2}{\Delta} = \frac{g^2}{\omega_{rl} - \omega_q} \rightarrow g = \sqrt{(\omega_{rl} - \omega_q)(\omega_{rh} - \omega_{rl})} \quad (3.4)$$

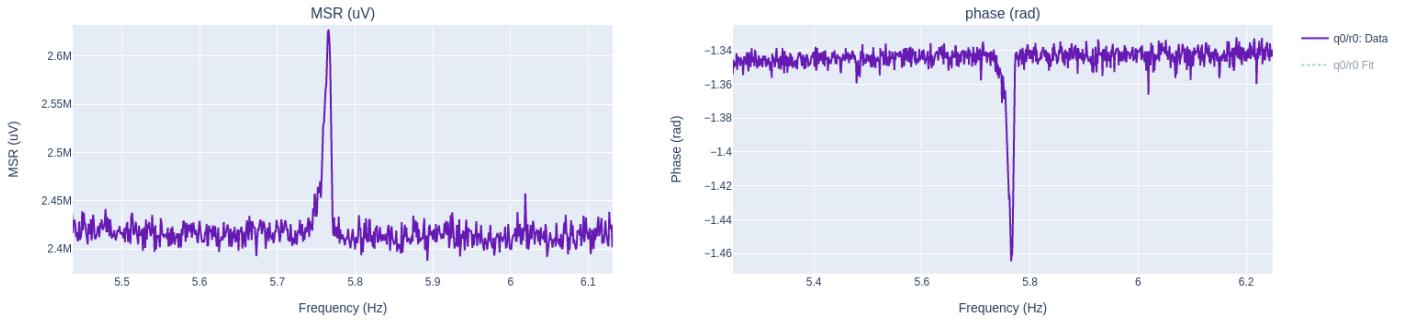


Figure 3.16: Example of a qubit spectroscopy experiment with zero relaxation.

In fig. 3.16 we can see the plot for a qubit spectroscopy performed without relaxation time: the physical peak is centered at 5.7 GHz, but the apparent peak is asymmetric and shifted.

Experiment recap: Qubit spectroscopy

Scope:

drive calibration.

Parameters to extract:

- frequency of the qubit,
- value of Δ ,
- value of the coupling,
- quality factor of the qubit.

Brief description:

two tone spectroscopy. First a tone is sent to the drive line, then a measurement is executed. We repeat the experiment for different frequencies for the first tone and plot the measured amplitude vs the first tone frequency. We expect to find a Lorentzian shaped peak in the opposite direction of the resonator one.

3.6 Flux Qubit Spectroscopy

In section 3.4 we found a rough estimation of the sweetspot, exploiting the coupling between qubit and resonator. The resonance frequency of the resonator changed (weakly) with the flux and we had to find the bias level that maximized it.

We now repeat a similar experiment with qubit spectroscopy, expecting to find a much more sensitive dependence between flux and qubit frequency.

The experiment it's a bit of a merge between "Flux Resonator Spectroscopy" and "Qubit Spectroscopy":

- we send a constant DC current at the qubit through the flux line. The voltage amplitude of this pulse will be the bias level;
- we perform a two-tone qubit spectroscopy;
- we repeat for different bias levels and different drive frequencies.

In respect to the experiments already executed, we can focus on much more narrow scans: in particular for the flux one of which we already have an estimation of the sweetspot. Looking at fig. 3.11, for example, we could decide to focus our scan between 0.3 and 0.5. To obtain a sensible and overall *good* plot, it's important to decrease the bias step; the effect of a too large step is visible in fig. 3.17.

Another note on the parameter choices, in addition to all the ones that we already saw in the previous sections, we have here also the choice of the frequency span: in fact we already know the qubit frequency at the current estimation for the sweetspot and this can be the center of our scan, but how large should the scan be? This is extremely

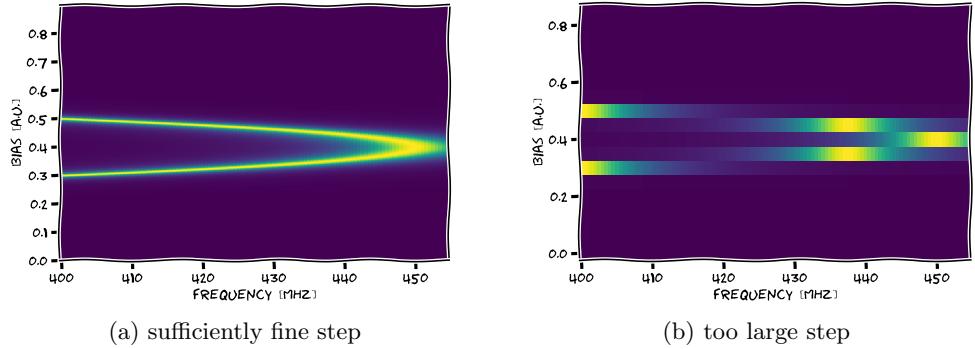


Figure 3.17: Sketch of the plot expected for a flux qubit spectroscopy experiment with comparison between a "bad" and "good" step for the bias scan.

dependent on the qubit parameters and fabrication, but we can say fro the beginning that the dependence will be much faster and steep than the one of the resonator.

It is useful to obtain a first realistic estimation of the parameters, to do an initial scan with larger steps (obtaining something like the first plot of fig. 3.17) to have a better idea of the strength of the qubit-flux dependence.

Some plots, for real case scenarios, are presented in fig. 3.18. Note that these plots come from acquisitions done at the same time, on qubits of the same chip (so manufactured with the same fabrication procedure) but nevertheless obtaining very different dependencies.

It is also important to repeat a *caveat* already outlined in section 3.4: if the goal is to control multiple qubits at the same time, then this experiment *must* be done at the same time for all the needed qubits. And, in particular, centering the sweetspots at the same time.

From these plots, we extract the sweetspot values as 0.753, 0.022 and -0.01 .

After choosing the new sweetspot it may be worth to re-do resonator and qubit spectroscopies: the resonator one should not have changed much, but it is better to be sure; on the other hand, for sure the qubit one has changed, it is true that every information is already in the 3D plot, but a more straightforward visualization is helpful.

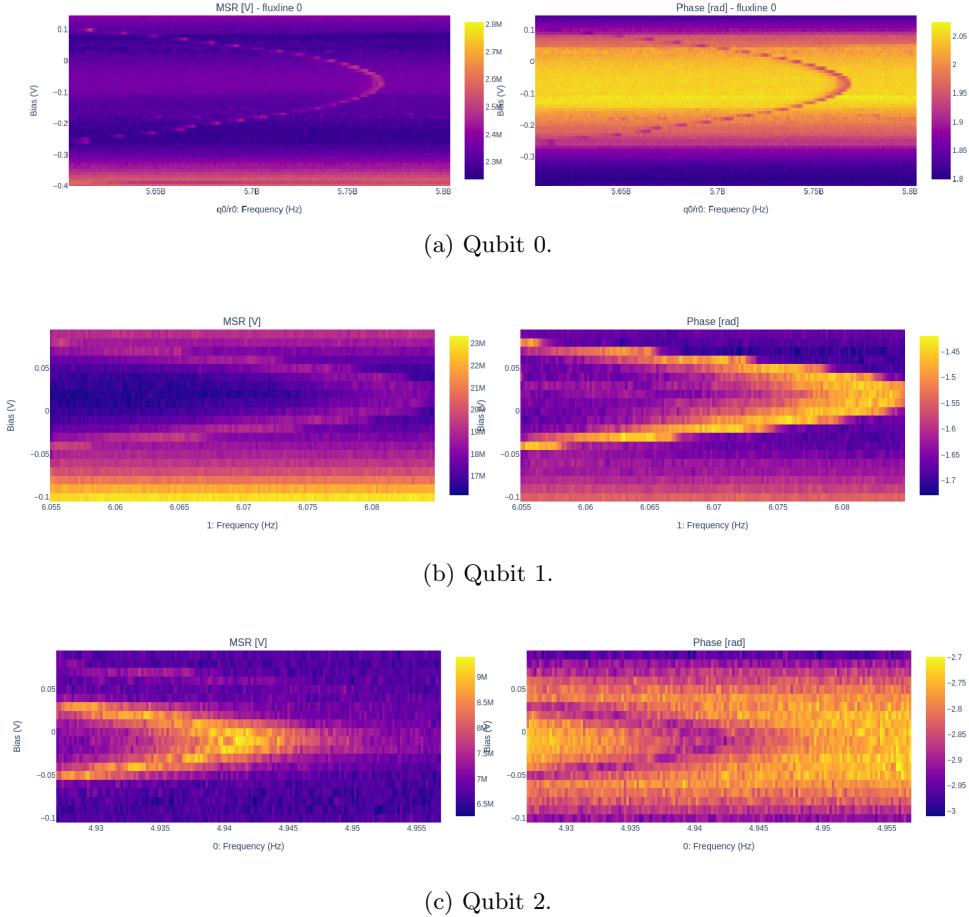


Figure 3.18: Plots for flux qubit spectroscopies.

Experiment recap: Flux qubit spectroscopy

Scope:

drive calibration, flux dependence calibration.

Parameters to extract:

flux sweetspot,
qubit frequency at the sweetspot.

Brief description:

a qubit spectroscopy experiment is executed for different frequencies and different biases (fixing the amplitude). We do a 3D plot with transmitted signal vs drive frequency vs bias. We extract bias and frequency for the sweetspot (highest frequency).

3.7 Rabi oscillation

In the last experiments, after defining all the readout parameters, we have also found the main parameter for qubit control, namely the frequency of the drive pulses. Now we have to calibrate the drive pulse so that we are able to completely control rotations on the Bloch sphere in ≈ 40 ns [59, 60].

Initially, we focus on calibrating a π -pulse, namely a X gate:

$$\begin{aligned} |0\rangle &\xrightarrow{X} |1\rangle \\ |1\rangle &\xrightarrow{X} |0\rangle \end{aligned}$$

The first elements to reach this control are included in eq. (1.35). They will lead us to define:

- the amplitude of the π -pulse;
- the duration of the π -pulse.

Practically, we can do three different, but similar, experiments: rabi-length, rabi-amplitude and rabi-amplitude-length. All these experiments have the same structure:

- "Prepare" the qubit in state 0 (usually just waiting for a long enough time);
- drive the qubit with a pulse with a specific amplitude and length;
- measure;
- update the pulse parameter (amplitude, length or both) and iterate;

In principle, considering the single-parameter experiments, we expect to see a \sin^2 as in fig. 3.19 and we can select the new parameter value for the π -pulse as half the period. In this way we are choosing the length and amplitude so that, from the ground-state we can go to the furthest point: the excited state.

From a practical point of view, however, we still do not have any way of computing *probabilities* and we are just plotting amplitudes: this could lead to deformed curves as, for example, the one plotted in fig. 3.20. Since we are plotting $AMP = \sqrt{i^2 + q^2}$ it can happen that the oscillation between the two center points in the IQ plane does not resembles the expected \sin^2 .

In any case, we should be able to correctly fit the obtained curve and, eventually, re-do this experiment once we have a way of computing probabilities. Note that the only important parameter here is the period of the oscillation.

For the two-parameters experiment, on the other hand, we expect something like fig. 3.21. We can see that this plot basically has the two plots combined.

What are the difference between the three experiment? Should you execute them all?

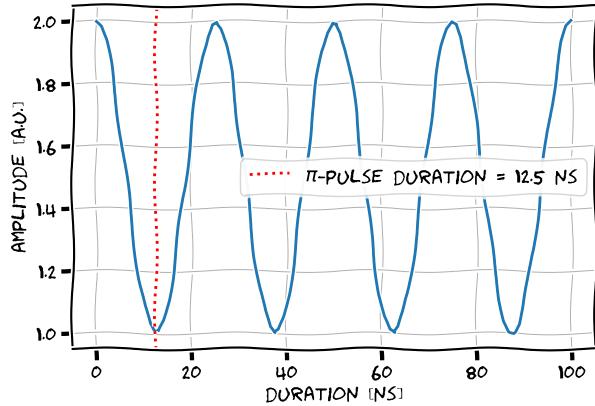


Figure 3.19: Expected behaviour in Rabi (length) oscillation.

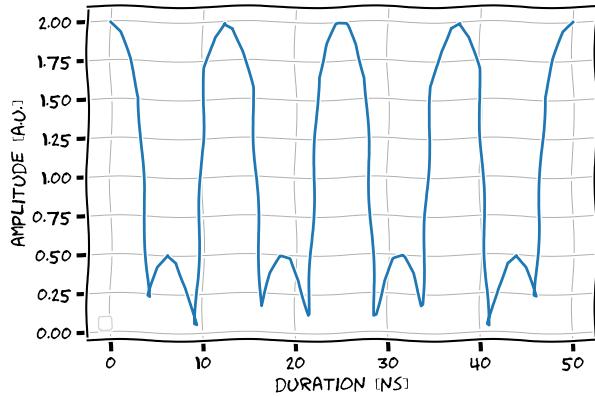


Figure 3.20: Possible behaviour in Rabi oscillation, when plotting amplitudes.

In general, we can avoid to do the two-parameter scan: while it is complete, it is much slower than the single-parameter ones and probes length-amplitude configurations not too useful. In a practical situation, where no information is given about the qubit, one could simply choose an ansatz on the duration of the pulse: since we want, in the end, fast pulses, a good starting choice could be 40 ns. A Rabi amplitude experiment is, usually, more precise than a duration one, but it also more susceptible to range problems: it can indeed happen that the physical range of the DACs is not enough to see a full oscillation. It's not important to see it, but the fit could be much harder and, in any case, the π -pulse amplitude must be reachable so a new amplifier at the drive line (or less attenuation) could be needed. Clearly, there is always the possibility to use longer pulses.

On the other hand, Rabi-duration doesn't incur in these problems, but in general it doesn't provide the same fine tuning capability. It can be useful, if we have an idea of the amplitude order of magnitude, to confirm it and to do an estimation of the length. Moreover, this can help to find the average amplitude for the excited state which we still

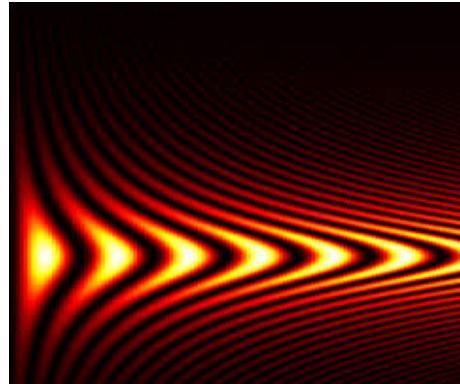


Figure 3.21: Plot of a Rabi-amplitude-length oscillations experiment. On the x-axis we have "pulse length", on the y-axis "pulse amplitude".

didn't estimate properly.

In figs. 3.21 to 3.23, some plots of the Rabi oscillations experiments are presented.

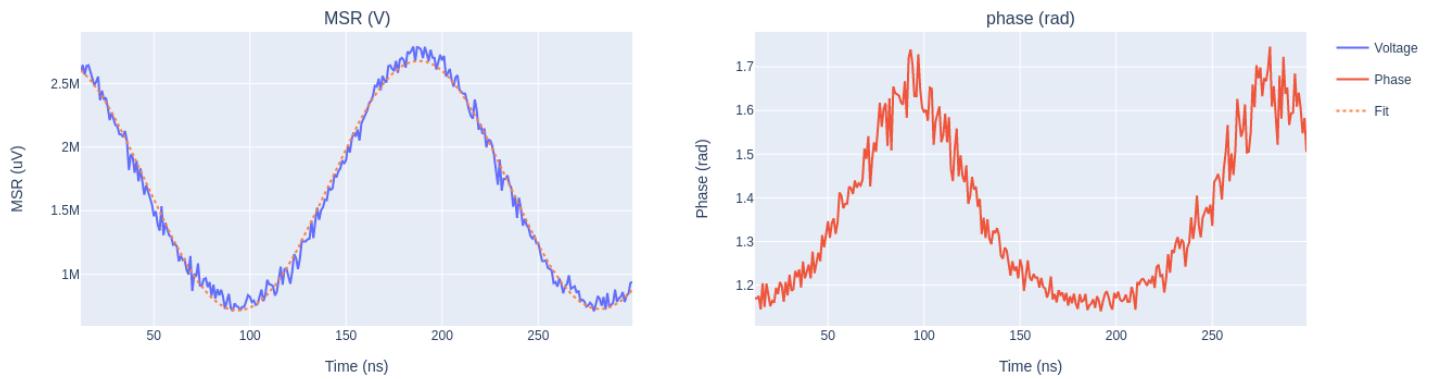


Figure 3.22: Plot of a Rabi-length oscillations experiment.

In this case, we chose a length of 30 ns and an amplitude of 0.203. In fig. 3.21 these numbers would describe the shortest central oscillations, namely the one that maximize the amplitude of the sinusoidal.

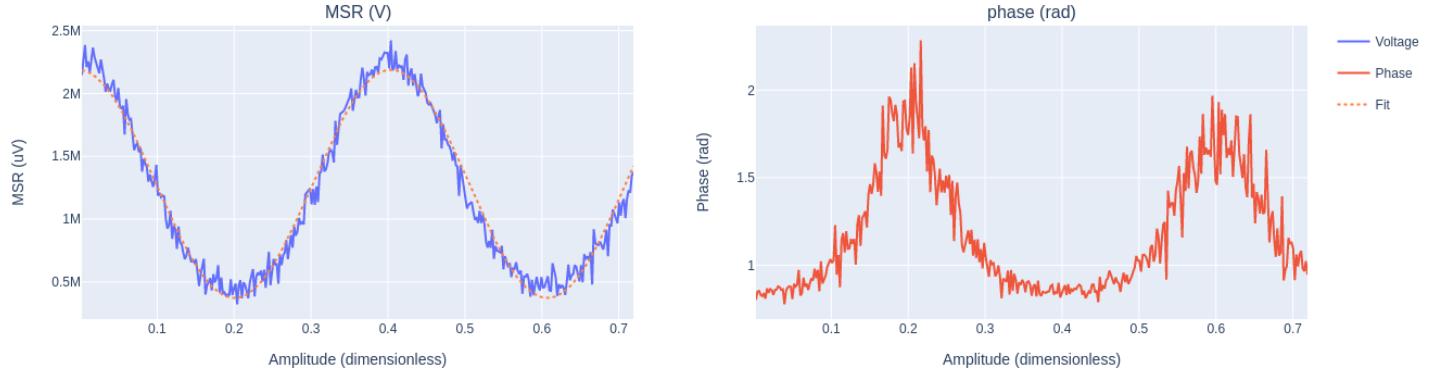


Figure 3.23: Plot of a Rabi-amplitude oscillations experiment.

Experiment recap: Rabi oscillations

Scope:

drive calibration.

Parameters to extract:

duration of the π -pulse,
amplitude of the π -pulse,
estimation of the amplitude relative to the ground and excited state.

Brief description:

a drive pulse with a specific amplitude and length is sent to the qubit. Afterwards, we perform a measurement of the qubit state. We repeat the experiment for different amplitudes of the drive pulse (or different lengths). We plot the transmitted/reflected amplitude vs the pulse duration/length. We perform a fit with a \sin^2 , taking half of the period as the π -pulse parameter.

3.8 Ramsey experiment

In the last experiment we calibrated the π -pulse and, having already calibrated the measurement pulse, we have all the components needed for control, technically we could already perform algorithmic experiments. However, we still do not know much about the qubit itself and, in particular, the values of T_1 and T_2 . Moreover, we didn't perform a complete calibration and, if we tried to run circuits now, we will not achieve good fidelities.

The *Ramsey experiment* [61] is a simple routine that allows us to address all these

problem at once. In particular, with it we can:

- fine tune the drive qubit frequency;
- perform a sanity check on our setup, confirming we are correctly controlling signals phases;
- compute T_2 (technically we obtain T_2^* , see footnote)

The experiment can be performed in a *simpler* version, often called *non-detuned Ramsey* and in a *detuned* version.

Although usually the non-detuned experiment can be avoided, it is useful as a starting point for the discussion. So, let's start from it.

First, we send to the drive line a $\pi/2$ -pulse. That is a drive pulse with the same characteristics of a π -pulse, but with the amplitude halved. This pulse should work as follows:

$$\begin{aligned} |0\rangle &\xrightarrow{\frac{\pi}{2}} \frac{|0\rangle + i|1\rangle}{2} \\ |1\rangle &\xrightarrow{\frac{\pi}{2}} \frac{|0\rangle - i|1\rangle}{2} \end{aligned}$$

After the $\pi/2$ -pulse, we wait for a certain time and fire a second $\pi/2$ -pulse followed by a measurement. We then repeat the experiment for increasing waiting times between pulses and we plot the measured amplitude against the waiting times. *In the ideal case*, we expect to see an exponential decay from the amplitude corresponding to $|1\rangle$ to the medium point between ground and excited state. At this point we can fit the obtained curve with an exponential and obtain T_2 as the decaying constant⁷.

It is not immediate to understand what is happening to the qubit, but we can try starting at the limit values:

- at wait time 0, we send two consecutive $\pi/2$ -pulse that have, together, the same effect of a π -pulse, so we expect to measure the excited state;
- at wait time $\gg T_2$, we expect the qubit to decohere during the waiting time (we are not considering relaxation since in general $T_1 \gg T_2$). So we expect to be again in complete superposition when we measure;

but what happens for an intermediate waiting time?

After the first pulse, the qubit starts to decohere so a variable phase gets added to the state. We can visualize this decoherence as a rotation on the xy plane, in the Bloch sphere. When we apply the second pulse, we have an equal probability to get to the ground state or to the excited one since the phase added is not constant.

We can see a schematic representation of the experiment in fig. 3.24.

The ideal plot is presented in the sketch in fig. 3.25

⁷Technically we obtain T_2^* in this experiment. See section 3.11 for more details.

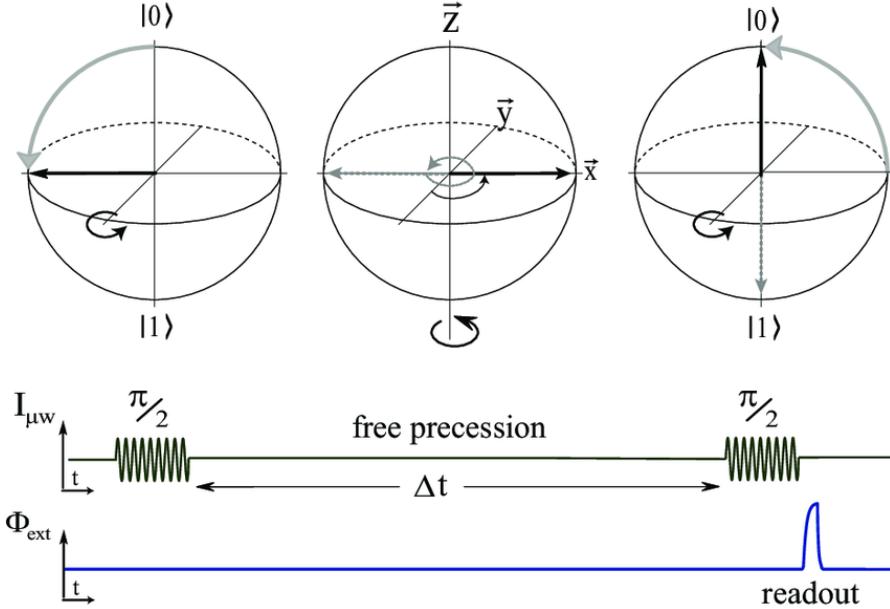


Figure 3.24: Schematic representation of a Ramsey experiment. Credits [62].

However, several effects could lead to spurious and unexpected behaviours. Consider for example fig. 3.26. A part from some noise that has to be expected, here the plot is almost ideal and the curve, obtained by points average 1000 times, can be easily fitted with an exponential (a $T_2 = 6 \mu\text{s}$ was extracted). But what happens around 13 thousands ns and again around 17 and 20 is not just noise: indeed we can see that the qubit state is "constantly" having a different final state. This can happen when the qubit, because of a non perfect environmental isolation, couples to spurious cavity modes, thus effectively changing the 0-1 transition frequency.

Another possible plot that one could obtain is presented in fig. 3.27. Actually, the majority of the times, when we perform a Ramsey experiment we will initially obtain something like this. Note that we still have the exponential decay but convoluted with a sinusoidal. This can be explained with a not-properly-calibrated π -pulse (and therefore a not-properly-calibrated $\pi/2$ -pulse). Indeed if our drive pulse is not sending $|0\rangle$ into $\frac{|0\rangle+i|1\rangle}{\sqrt{2}}$, but in something near it *outside* the xy plane, than the decohering will happen as a rotation also outside the xy plane. So, for example, when we have wait time equal to zero, then we are not doing exactly a rotation over x of π , but of $\pi \pm \epsilon$ where ϵ is usually small.

Therefore we are seeing the decoherence of Ramsey and the oscillations of Rabi at the same time.

From the period of the oscillations we can fine tune the drive frequency. In the plot of fig. 3.27, for example, the oscillations have a period of $\approx 1.8 \mu\text{s}$, this means that the

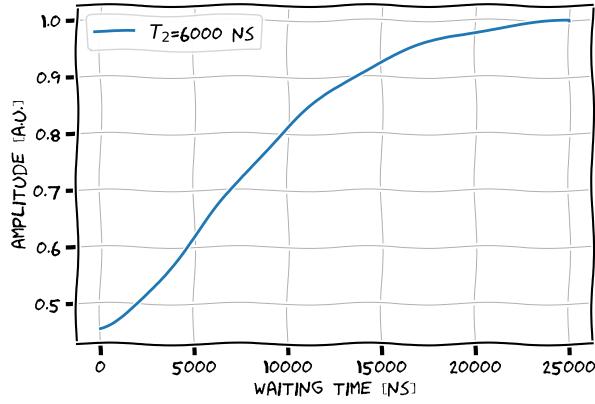


Figure 3.25: Sketch of the plot of a Ramsey experiment in ideal conditions.

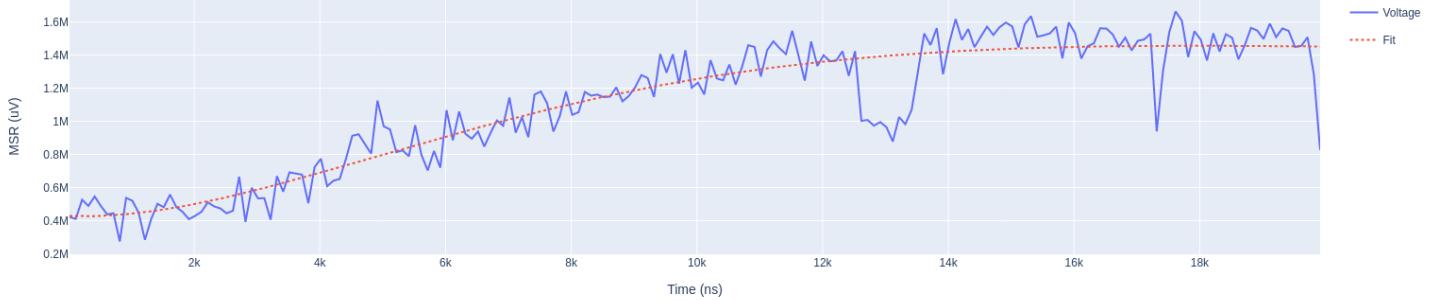


Figure 3.26: Plot for a Ramsey experiment with the presence of a spurious coupling.

frequency is off by $\frac{1}{1.8\mu s} = 555555 \text{ Hz}$. Basically, we are performing a fit with:

$$y = p_0 + p_1 \sin(2\pi\Delta x + p_3) e^{-\frac{x}{T_2}} \quad (3.5)$$

with the final correction being Δ .

The problem, at this point, is knowing the sign of the correction to which we are currently not sensible at all. We have the option of trying the two values and choose the one that produces the most exponential-like curve. Note, also, that for every change in the frequency, it is recommended to perform a new Rabi experiment to re-calibrate the π -pulse.

The other option, to avoid the sign ambiguity, is to perform the detuned experiment. We start defining the maximum wait time we will probe as `max_wait` and a number of desired oscillations as `n`. For every second $\pi/2$ -pulse, we will add a spurious phase defined as:

$$\phi = (\text{start_time})2\pi \frac{n}{\text{max_wait}} \quad (3.6)$$

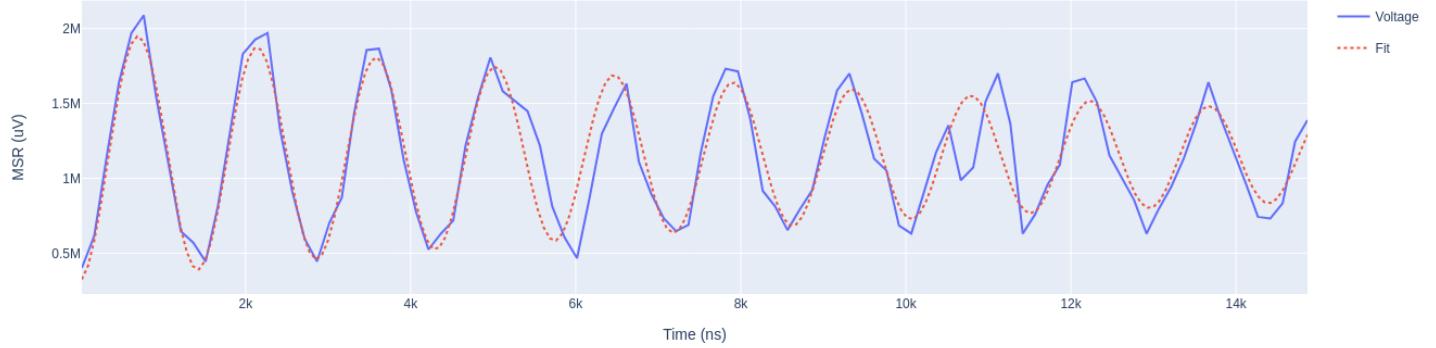


Figure 3.27: Plot for a Ramsey experiment at the wrong drive frequency.

The effect of this detuning, if the drive frequency is the exact one will present itself as precisely n oscillations in the plots. On the other hand, if we are driving the qubit at a higher frequency we expect to see more oscillations, if we are not driving the qubit enough we will see less. The fit function is again eq. (3.5), but the final correction become:

$$\Delta_{corr} = \Delta - \frac{n}{max_wait} \quad (3.7)$$

with the new drive frequency being $\omega_q = \omega_d - \Delta_{corr}$ ⁸.

After the correction and after a new *required* Rabi experiment, is always advisable to repeat the non-detuned experiment, to really check if the oscillation is disappeared (so to check if the drive frequency is now correct).

⁸A wrong interpretation of the phase, that here is defined as positively growing with growing frequencies, will lead to $\omega_q = \omega_d + \Delta_{corr}$.

Experiment recap: Ramsey experiment

Scope:

drive calibration, qubit characterization.

Parameters to extract:

fine tuned qubit/drive frequency,
characteristic dephasing time T_2 (T_2^*).

Brief description:

a $\pi/2$ -pulse is sent to the qubit through the drive line, after a variable wait time we execute a new $\pi/2$ -pulse and measure. If we do not see any oscillation, but just an exponential decay, we can find T_2 . Otherwise, depending on the number of oscillations we can fine tune the drive frequency with a sign ambiguity, perform a Rabi experiment and repeat the Ramsey experiment. In the detuned version of the experiment, a spurious and controlled detuning is introduced as a phase on the second $\pi/2$ -pulse. The effect is to fix a desired number of oscillations as $\neq 0$ and, if this number is different from what expected, we can again fine tune the frequency without sign ambiguity.

3.9 Flipping

Another possible way of refining the π -pulse is the *Flipping* experiment.

In particular, if the Ramsey experiment is used to fine tune the frequency, the Flipping experiment can be used to fine tune the amplitude, sometimes with notable differences in respect to the value found with Rabi.

In this context, we define a flip as two consecutive π -pulses that, in the ideal condition, should in the end lead to the initial state.

For the experiment, we choose a maximum number of flips M and step number S . We now have all the elements to perform the experiment:

- with the qubit in $|0\rangle$, we apply a $\pi/2$ -pulse;
- we apply N flips (with $N \in [0, M]$ with steps of S);
- we measure;
- we repeat the experiment with a higher number of flips.

In the ideal case we should see just a straight line, always being in a superposition state (so in the middle point between the two amplitude states). In the case of a wrong amplitude, we encounter oscillations. We can perform a fit analog to the Ramsey experiment, or even proceed by manual tries, since the experiment is generally extremely fast (after all, there is no scan).

In fig. 3.28 we can see a possible plot for the flipping experiment. In this case the

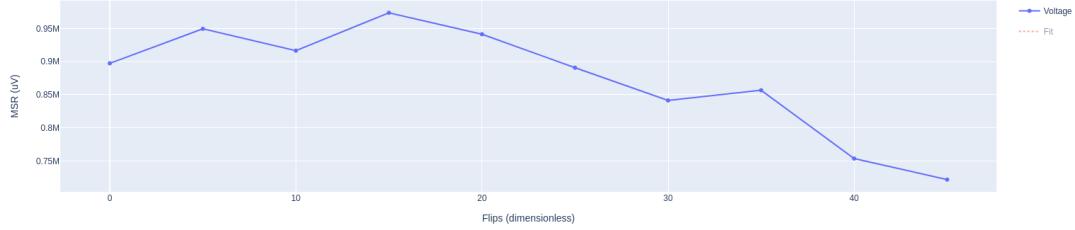


Figure 3.28: Plot of a flipping experiment.

amplitude is almost correct, since we cannot see any clear oscillation, but can for sure be fine tuned to remove the present "decay".

While this experiment can be interesting, it has the problem of involving a large number of gates and this may cause memory problems in the control system.

Experiment recap: Flipping

Scope:

drive calibration.

Parameters to extract:

fine tuned qubit/drive frequency,
characteristic dephasing time T_2 (T_2^*).

Brief description:

after a $\pi/2$ -pulse, an increasing number of flips is executed (a couple of π -pulse). If the drive frequency is correct, the resulting plot will just present exponential decay, otherwise an oscillation from which we can extract the correct frequency..

3.10 T1 measurement

In this experiment we measure the characteristic T_1 time of the qubit [63]. This is the first experiment that cannot be considered calibration since no control parameter gets changed, but rather we are trying to understand better the qubit so we can talk of pure *characterization*.

The experiment is easy to understand and follow: we first excite the qubit with a π -pulse, then wait some time and measure. If the waiting time is zero, then we expect to measure the amplitude of the excited state $|1\rangle$, if the wait time is ∞ then we expect the qubit to have completely relaxed to $|0\rangle$ before the measurement. For intermediate wait times, we have an time-exponentially decaying probability of relaxation so, for a sufficiently large number of shots, we expect to see the same shape for the measured amplitude.

Therefore, we can fit the obtained curve with a simple exponential:

$$y = p_0 - p_1 e^{-\frac{x}{T_1}} \quad (3.8)$$

In fig. 3.29 a real T_1 experiment is shown.



Figure 3.29: Plot of a T_1 experiment.

Experiment recap: T1 measurement

Scope:

qubit characterization.

Parameters to extract:

characteristic relaxation time T_1 .

Brief description:

a π -pulse is sent to the qubit through the drive line, after a variable wait time we perform a measurement. We plot the measurement amplitude against the waiting time and fit the curve with a simple exponential function. The decay constant will be T_1 .

3.11 Hahn's spin echo (T2)

In section 3.8 we computed the value of T_2 but we overlooked something: from Chapter 5 of Nii Quantum Information Lectures [64] we read:

” An applied DC field H_0 is not completely uniform in all space points. If many spin qubits are placed in such an inhomogeneous DC field, they have different Larmor frequencies. This leads to the dephasing effect if we compare the phase difference between

different qubits. A time constant for this dephasing process is determined by the spatial (not temporal) inhomogeneous broadening of the dc field and distinguished from T_2 process. A new time constant is often referred to as T_2^* . ”

This sentence is not immediately clear, but it's basically saying that a spurious DC field, coming from the other qubits near the one under analysis, or from a near flux line or again just from noise, can lead to an additional dephasing effect not included in the standard definition of T_2 . In section 3.8 we did not consider any of these effects, so we actually measured T_2^* that is a good approximation of T_2 in a nearly-noiseless environment.

To measure the pure dephasing T_2 , we have to perform higher order experiments that take into account these potential impurities. One of these is the *Hahn's spin echo* experiment [65].

The experiment can be seen as a non-detuned Ramsey experiment with the addition of a π -pulse between the two $\pi/2$ -pulse, in particular the experiment is as follows:

- a $\pi/2$ -pulse is sent to the qubit so that the state of it is now $\frac{|0\rangle+i|1\rangle}{\sqrt{2}}$;
- we wait for $\Delta t/2$ so that the state is $\frac{|0\rangle+e^{-i\Delta t/2}i|1\rangle}{\sqrt{2}}$;
- a π -pulse is sent to the qubit. The state is now $\frac{|1\rangle+e^{-i\Delta t/2}i|0\rangle}{\sqrt{2}}$;
- we wait for $\Delta t/2$ so that the state is $e^{-i\Delta t/2}\frac{i|0\rangle+|1\rangle}{\sqrt{2}}$;
- we apply a $\pi/2$ -pulse so the qubit reaches $e^{-i\Delta t/2}|0\rangle$;
- finally, we measure the qubit state.

The main difference with Ramsey, the π -pulse in the middle of the experiment, guarantees that any DC impurity applies the same phase to $|0\rangle$ and $|1\rangle$, therefore leading to a negligible global phase.

We can now perform an exponential fit where we can extract the decay time T_2 .

Note that there are some experiments of even higher order that take into account other possible impurities, but in general Hahn's spin echo is the used technique for T_2 . These experiments include the *Carr-Purcell sequence* and the *Meiboom-Gill sequence* [66].

Also note that, in literature, there is the tendency to write both Ramsey- T_2 and Echo- T_2 simply as T_2 so one has to be particularly careful when talking about the dephasing time.

A plot for a Echo experiment is presented in fig. 3.30 and is, as the ideal Ramsey, a simple exponential decay. However, this time the starting state is $|0\rangle$, so the curve is flipped horizontally.

qubit	Fitting Parameter	Value
0	T2 Spin Echo	9,304 ns.

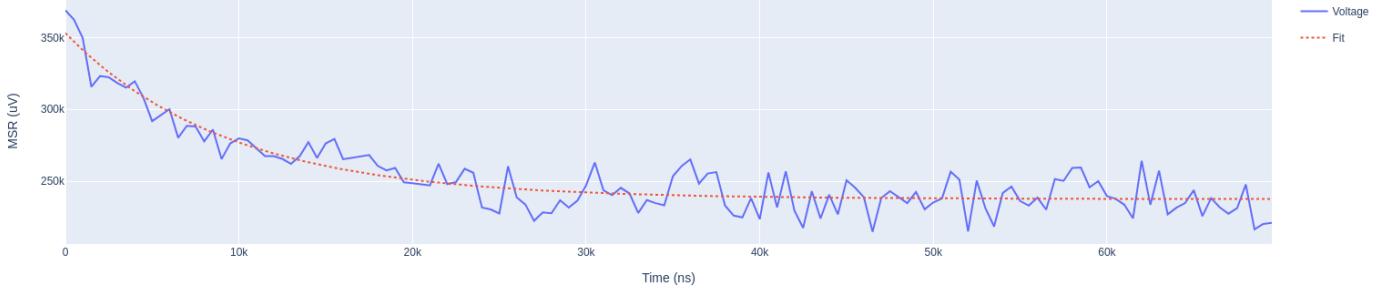


Figure 3.30: Plot of a Hahn's Spin Echo experiment.

Experiment recap: Hahn's Spin Echo experiment

Scope:

qubit characterization.

Parameters to extract:

characteristic dephasing time T_2 .

Brief description:

a $\pi/2$ -pulse is sent to the qubit through the drive line, after a variable wait time $t/2$ we execute a π -pulse and wait again $t/2$. After a new $\pi/2$ -pulse we measure the qubit state. We can now fit the obtained curve with an exponential curve and obtain T_2 as the decay constant. This experiment differs from Ramsey for the introduction of a π -pulse between the two $\pi/2$ -pulse that enables the removal of a spurious dephasing term linked to external DC currents, leading to better estimations of T_2 .

3.12 Single-shot classification

In the past experiments we calibrated the measurement pulses and the drive pulses so, at this point, what is missing from a basic calibration point of view is just the states classification [67, 68].

The experiment is very easy, but can still be fine-tuned for readout calibration. The procedure is as follows:

- we prepare the qubit in state $|0\rangle$ (waiting enough time) and we measure *without* averaging;

- we prepare the qubit in state $|1\rangle$ (applying a π -pulse) and we measure *without* averaging;
- we repeat the experiment and we plot in the IQ plane the single shots.

In fig. 3.31 an ideal sketch-plot for the experiment is presented.

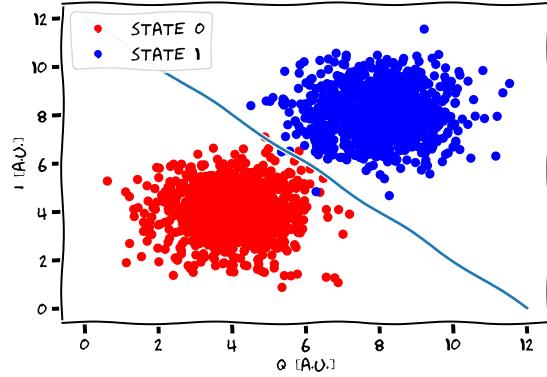


Figure 3.31: Ideal plot for a single-shot classification experiment.

The task now is to define a classification scheme that enables to assign a state (0-1) to every point in the IQ plane. Different scheme more or less efficient can be employed (based on Machine Learning, ...) but usually the definition of a simple straight division line is enough. To find it, we proceed as follows:

- we find the mean points for state 0 and 1;
- we perform a rotation of the angle identified by the segment between the two mean points. Namely, we rotate the IQ plane so that the two mean points are on the same horizontal line;
- we focus on the real parts and perform the cumulative distribution of the points of state 0 and state 1 (considering all the points);
- we compute the difference point-by-point between the two cumulative distributions and select as threshold the value that maximize this difference.

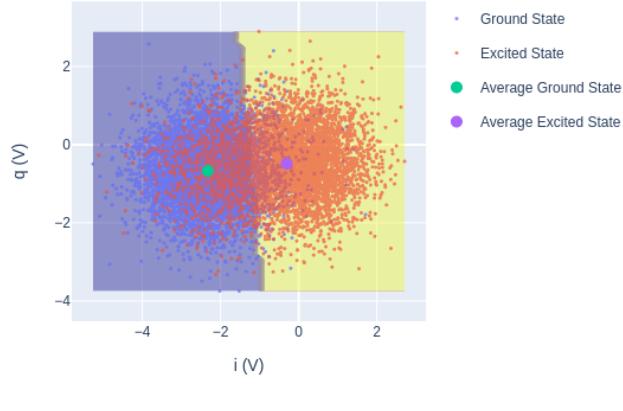
In this way we are maximizing, with a single straight line, the *assignment fidelity* defined as:

$$\text{assignment fidelity} = \frac{\text{correct predictions}}{\text{total number of points}} \quad (3.9)$$

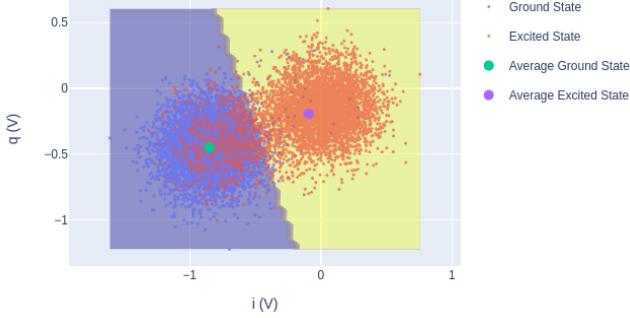
The two parameters that we have to save to use them in later experiments and circuit executions are the rotation angle and the threshold ⁹.

⁹Note that we can consider these two parameter sufficiently stable, but still this experiment needs to be executed every now and then even in ideal cases. Moreover, for RFSoCs systems, the rotation angle can change every time the firmware is re-loaded into the FPGA logic.

The assignment fidelity is the first single number that describes the "goodness" of the qubit and its calibration. For the qubit to work in the best way, this number should be higher than 90-95% but note that the real fidelity number (namely gate-fidelity) will be computed with a different experiment.



(a) Assignment fidelity = 0.80



(b) Assignment fidelity = 0.92

Figure 3.32: At the top, a possible bad plot in a single-shot classification experiment. At the bottom, a good one.

In real case scenario, sadly too often, the classification plot can look as fig. 3.32(a). The problem can be something hardware related, caused to the qubit fabrication and inherent S/N ratio so cannot be improved arbitrarily, or fixable with a good calibration. In particular, a longer measurement pulse can help to distance the two "blobs" in the IQ plane and also an increased measurement amplitude can improve the S/N leading to the same effect. If the fidelity is still very low, a good hardware solution is to add a

quantum limited amplifier (TWPA or JPA) to the readout line inside the cryostate. In any case, the fact that the single shot fidelity is not great can be partially reduced as a problem by increasing the number of shots to execute whatever circuit of future experiment.

On the other hand, sometimes the calibration and qubit quality is enough to obtain a nice fidelity as in the bottom plot of fig. 3.32.

Experiment recap: Single-shot classification

Scope:

qubit characterization.

Parameters to extract:

threshold and rotation angle for classification,
assignment fidelity.

Brief description:

we prepare the qubit in the state zero and we measure the IQ values without averaging. We then prepare the qubit in state one and measure. We repeat the measurements a high number of times and we plot the non-averaged results. We find the separation line that maximize the assignment fidelity, namely the fraction of right assignments.

3.13 Readout optimization

After section 3.12 we should have a working qubit, fairly calibrated. But still, we are not taking the most out of it. In particular, we never really calibrated the readout pulse. This is because we didn't really have a way to determine which values were the best. In the last experiment, however, we defined the assignment fidelity and we can use it as figure of merit.

The amplitude of the pulse was already calibrated in section 3.3 and its shape will not affect much the fidelity. What we can improve is:

- the duration of the pulse;
- the frequency of the pulse.

For the duration of the pulse there is no specific experiment nor a specific relationship. The experimenter will have to try different values and choose the one that maximize the assignment fidelity.

Note also that there is a bit of a trade-off: longer measurement should improve the S/N ratio but will also mean longer execution times. Moreover, if the resonator lifetime is not long enough, this could lead to weird results. Usually a measurement around $2\ \mu s$ is good enough.

Note also that the two "blobs" in the classification experiment do not have to be too much separated, but should always barely touch. This is because two distant blobs usually also cause a lower T_1 (since from the energy point of view, the two states are very distant).

For the frequency we can do something more interesting. Until now, we used the resonator frequency as readout frequency, but this was in fact a bit arbitrary. Let us return to describe the principle behind the measurement: the resonator is coupled to the qubit and, depending on the qubit state the resonator peak *shifts* independently from the measurement. We then measure at a fixed frequency and induce, from the amplitude, if the resonator is shifted or not.

We are never fully exciting the resonator (at least on purpose) so there is no reason to send the tone exactly at the resonator frequency! This is the basis of the *dispersive shift* experiment:

- we first do a standard resonator spectroscopy (not considering the qubit that remains in $|0\rangle$);
- we perform a second resonator spectroscopy, but with the qubit in $|1\rangle$ (sending a π -pulse before every measurement);
- we plot the two curves on the same plot and we choose as readout frequency the one that maximizes the difference between the two states.

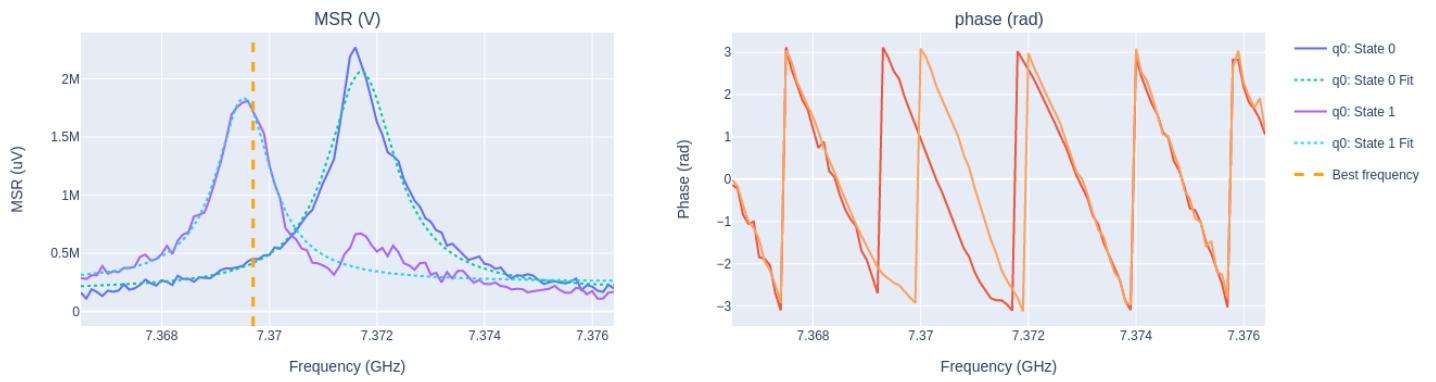


Figure 3.33: Plot of a dispersive shift experiment.

As we can see in fig. 3.33, the optimal readout frequency does not correspond to the resonator peak for either of these plots.

This experiment is just an ulterior optimization and is not strictly required, but can be used to improve the assignment fidelity. Note also that, after a change in frequency for the measurement pulse, new classification parameters need to be computed.

Experiment recap: **Readout optimization (dispersive shift)**

Scope:

readout calibration.

Parameters to extract:

optimized readout frequency,
readout duration,
dispersive shift.

Brief description:

for the calibration of the pulse length, we try readout pulses with different length and choose the one that maximize the assignment fidelity. For the frequency, we perform a dispersive shift experiment: performing one resonator spectroscopy with the qubit in $|0\rangle$ and one resonator spectroscopy with the qubit in $|1\rangle$. We then choose the frequency that maximize the difference between the two states.

3.14 AllXY

In this experiment we do not calibrate any pulse nor we characterize any qubit parameter, but rather we perform a practical evaluation of the quality of the calibration as well as a sanity check of the control system.

The experiment involves a set of gate defined as: $\{I, X, Y, X/2, Y/2\}$ where I is the identity, $X - Y$ are π -pulse with relative phase of zero and $\pi/2$ respectively, and $X/2, Y/2$ are the same pulses with halved amplitude. Talking in terms of rotation: we have π and $\pi/2$ rotations along the X and Y axis, as well as two identity (correspondent to no pulse).

In an *AllXY* experiment, we apply all the possible couples of gates of the defined set, one at a time, and perform a measurement. In general, this experiment is referred in literature quoting not the probability of measuring the zero state nor the amplitude, but the expectation value of σ_z (1 for $|0\rangle$ and -1 for $|1\rangle$). So, we expect to measure 1 for $\{II, XX, YY, XY, YX\}$, we expect 0 for $\{IX/2, IY/2, XX/2, XY/2, X/2X, X/2Y, YX/2, YY/2, Y/2X, Y/2Y\}$ and finally -1 for $\{IX, IY, X/2X/2, Y/2Y/2\}$.

So, in the ideal case, a *allXY* plot looks like fig. 3.34.

In a real case, usually, we have points more close to 0 (the dephased state) as in fig. 3.35.

AllXY is extremely fast since it does not include any scan, but allows to see if the assignment fidelity is high enough and if the gate are working "together" properly in a single experiment.

Note also that, by looking at the kind of errors of *allXY*, we can infer which section of the calibration is most wrong [47].

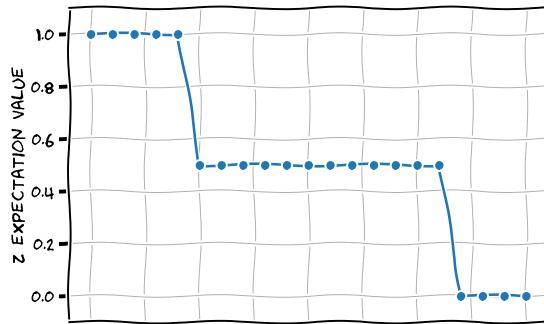


Figure 3.34: Ideal plot for a allXY experiment.

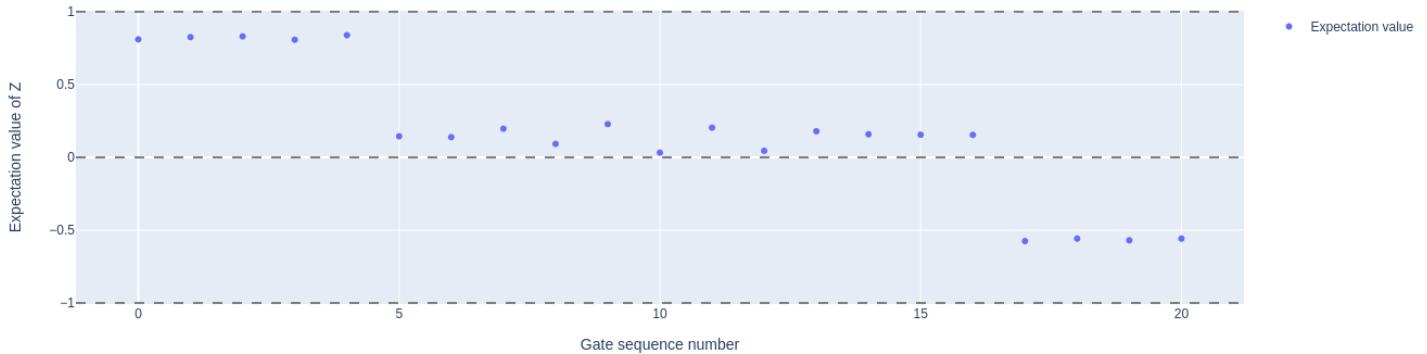


Figure 3.35: Plot of a allXY experiment.

Experiment recap: AllXY

Scope:

Healthcheck.

Parameters to extract:

None.

Brief description:

select all the possible combination of couple of gates in the set composed of identity, π -pulse and $\pi/2$ -pulse with 0 and $\pi/2$ phase. Execute a couple and measure, for all the couples. Plot the expectation value of σ_z for all the circuits performed and compare the results with the expect values.

3.15 DRAG tuning (pulse shape)

This section does not really describe a single experiment, and could actually be expanded to cover a whole new thesis. The study of the optimal pulse shapes (for the drive/control pulse in particular) is one of the most active research topics in the quantum control field. It is important to understand that this section just provides an example.

To introduce the DRAG [42] (Derivative Removal by Adiabatic Gate) technique we can start remembering that a superconducting qubit is just an approximation of a two-level system. Moreover, the difference in frequency between the transition 0-1 and 1-2 is not even particularly large. The DRAG is the analytical solution to force the interaction Hamiltonian expressed in eq. (1.31) to be restricted to the computational space defined as the state 0-1.

From the theory point of view, we can solve for $A(t)$:

$$H_{I\ eff}(t) = A^\dagger(t)H(t)A(t) + i\dot{A}^\dagger(t)A(t) \quad (3.10)$$

Leaving to more reliable sources [69] the burden of the proof, this translates in a modification of our Gaussian pulses, with the addition of a derivative term:

$$i = \text{Gaussian}(x, amp, sigma) = amp * e^{-\frac{1}{2}\left(\frac{x - duration/2}{sigma}\right)^2} \quad (3.11)$$

$$q = -1j * \beta \left(\frac{x - duration/2}{\sigma^2} \right) * \text{Gaussian} \quad (3.12)$$

From the shape point of view of the shape this produces an effect visible in fig. 3.36.

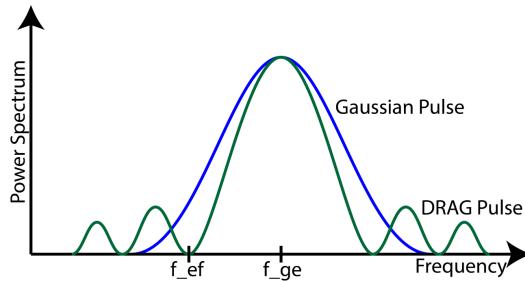


Figure 3.36: Comparison between a Gaussian and a DRAG shapes [70].

From another point of view, we can consider the DRAG pulse as a Gaussian pulse to which it got removed the 1-2 transition frequency Δ_{12} . In fact we have that the β parameter, that is a free, to-be-calibrated parameter is of the magnitude order of $1/\Delta_{12}$.

The calibration of β is rather empirical and consists in just repeating the allXY experiment for different hypothesis of β .

Moreover, it is important to note that pulse shape optimization usually does not lead to extremely better results, although it can slightly improve performances.

Experiment recap: Drag tuning

Scope:

drive pulse optimization.

Parameters to extract:

DRAG parameter β .

Brief description:

the allXY experiment is repeated for different values of β . The value that minimizes the distance between the measured points and the expected ones is selected as the calibrated β .

3.16 Randomize benchmarking

Randomized benchmarking [71–74] (RB) is the collective name of a series of popular protocols for characterizing the quality of a quantum processor. In particular, the objective of RB is to give a single value that describes the fidelity of a general gate. Here we talk about *gate fidelity* that is a parameter much more important than the *assignment fidelity* that we encountered in the last experiments.

Different RB protocols currently exist and are used, but they all share one core idea. They consist in the generation of random Clifford circuits plus their conjugates, so that the application of both should be equal to the identity. We repeat the experiment for increasing depths of the circuits and we extract the error rate per single qubit from the resulting exponential decay.

With Clifford circuit we intend a circuit composed of gates from the universal Clifford set.

Standard Randomized Benchmarking

With the standard version of the experiment we can compute an average gate fidelity or a fidelity differentiated between the different Cliffords.

The experiment is conceptually simple:

- we choose a maximum, a minimum and a step depth for the scan;
- for a given depth, we randomly generate a circuit;
- we execute the circuit C applying $C^\dagger C |0\rangle$;
- we measure and iterate for all the chosen depths.

A usual plot for the standard RB is presented in fig. 3.37.

We can easily compute the averaged error per gate $\mathcal{E} = 1 - \epsilon$ (ϵ is the fidelity) by fitting the curve with:

$$y = A\epsilon^n + B \quad (3.13)$$

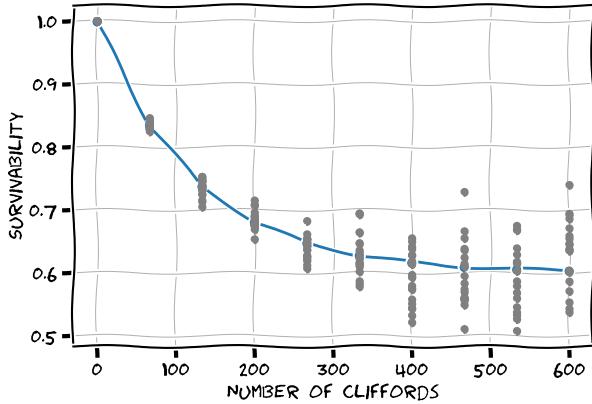


Figure 3.37: Sketch of the plot of a standard randomized benchmarking experiment .

where n is the number of gates for circuit.

To compute the error related to a specific gate e_i we can use:

$$\mathcal{E} = 1 - \prod_i (1 - e_i)^n \sim \sum_i n_i e_i + O(e^2) \quad (3.14)$$

It seems that we cannot distinguish the values of \mathcal{E} and e_i , however by defining an error ratio r_i , with respect to a chosen value e_0 we can compute the error for each basis gate:

$$\mathcal{E} \sim e_0 \sum_i n_i r_i \rightarrow e_i \sim r_i e_0 = \frac{r_i \mathcal{E}}{\sum_i n_i r_i} \quad (3.15)$$

The *pulse fidelity*, often used as figure of merit for qubit characterization, has a definition derived from the fidelity:

$$\text{pulse fidelity} = 1 - \frac{\frac{1-\epsilon}{2}}{\text{num pulses per Clifford} \approx 1.85} \quad (3.16)$$

In general, this experiment can be complex to perform with success on hardware: in particular if you are trying to reach high fidelities (we usually "count the 9s": 0.98, 0.998, 0.9998...) it usually involves very long sequences (since at lower depths the curve should be almost flat). In particular for the **RFSOC4x2** it proved to be a problem, since the long sequences (for 3 nines 500 gates are needed) filled the limited memory available. For example, the experiment presented in fig. 3.38 would be much more reliable if it did reach depths of 100/200.

Note that in this experiment is advised to not use a large number of shots, but rather to execute multiple time the circuit with different Cliffords, hence the different iterations present in the plots.

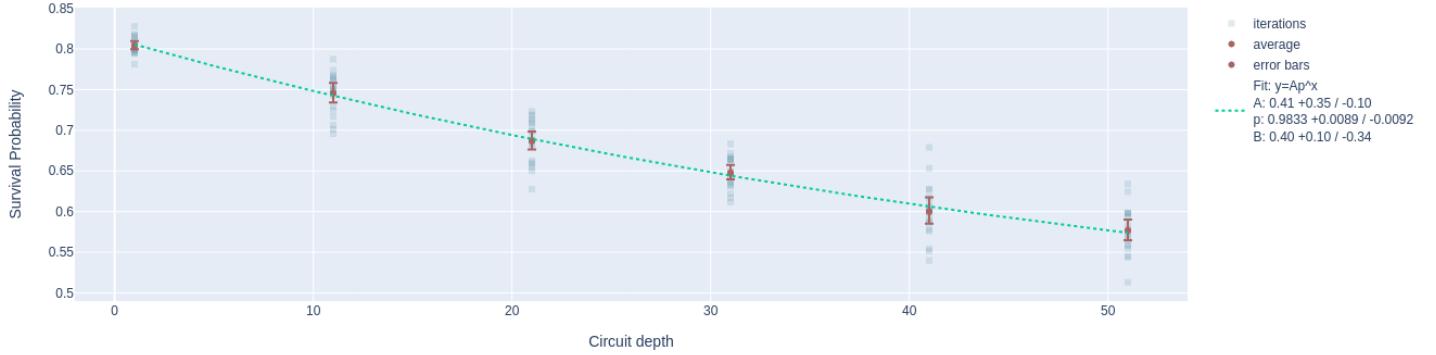


Figure 3.38: Plot of a standard randomized benchmarking experiment executed with a **RFSoC4x2**.

3.17 Other possible calibration experiments

As said in this chapter introduction, the one presented is just a limited collection of possible experiments that, nevertheless, is enough to achieve a proper calibration and a full characterization.

In the following list, some other experiments and calibration routines will be briefly presented so that the reader/experimenter, eventually, can dive in on its own.

Smearing correction: this technique can improve the assignment fidelity [75]: it consist in removing the initial and final part of the acquired waveform. Looking at fig. 3.4, for example, we can see that in the transients (start and end of the pulse) the separation from signal to no-signal is not maximized. Through smearing we can remove those parts from the computation, achieving higher readout fidelities.

Integration weights: again a routine for readout optimization [76]. We usually acquire a certain waveform and then average all the points to a single one (in the IQ plane), this process is referred to as *integration* and we are implicitly considering all the points as having the same importance. But we execute a time of flight experiment with state 0 and state 1, not every point will be separated the same. In particular, some points will be more significant and some less. By introducing weights for the integration of the waveform, we can exploit this to increase assignment fidelities (so the separation between the two states).

Direct calibration of $\pi/2$ -pulses: with the Rabi and Flipping experiments, we calibrated the π -pulses to the best of our capabilities, however in several routines/experiments/circuits we end up using different pulses for different rotations. It may be worth to spend some time calibrating directly $\pi/2$ -pulses or even $\pi/4$ -pulses

to achieve finer tuned control and in the end higher gate fidelities. This could be done using experiments similar to the one already encountered as flipping, but only using sets of $\pi/2$ -pulses.

Readout-qubit frequency: for flux tunable qubits we have found the sweetspot and we always used that as the bias level to operate them. In particular, this is required while driving the qubit since at different biases the frequency of the qubit is much more subject to noise changes. However, for reading out the qubit, so to interact with the resonator, it could be useful to use a different bias level, playing with the effective coupling qubit-resonator. This could potentially improve coherence times and fidelities.

Fast reset: in the experiments we saw, after each singular shot we need to wait for the qubit to relax. Generally, this time is chosen to be around 5 times T_1 . While working with qubits with low T_1 this is sustainable, but if T_1 is large, this can become a huge waste of time. It may be worth to implement and calibrate a *fast reset* (or active reset) technique [77]. The idea is that, instead of waiting, we measure the final state and check if it's zero or not¹⁰. If the qubit is already in $|0\rangle$ we can run another shot of our experiment, otherwise we can fire a π -pulse and go on with another shot. Note that while this idea is extremely powerful, it may need a very fine calibration since every π -pulse has the possibility of mistakes. A possible way of reducing this problem is to measure multiple times, even after the π -pulse.

Memory-based reset: a different approach to the same problem of fast reset is based on a memory approach. Instead of trying to reset the qubit to $|0\rangle$, we can just record the measurement value and proceed to the next shot. If the measured value was $|0\rangle$ nothing has to be done, otherwise the final result of the successive shot will require the application, in post-processing, of a X gate.

Quantum non-demolitioness of the readout: this parameter estimation is almost required for the active reset methods. Basically we always assumed that the readout pulses, since they are interacting only with resonators and not with qubits, are not changing its state. This is a fair assumption, but with active reset we require measurement that have QND probabilities $\gg 0.9$ or they will not make any sense. Therefore, it is important to compute the QND probability for a measurement [47].

Cross-talk measurements: when working with multiple qubits it is extremely interesting to measure the cross-talk matrix so, basically, how every qubit disturb drive and readout of the others. The field of cross-talk measurement has roots in the field of noise modeling for qubits that is extremely vast and interesting. Different protocols [78] already exist for this kind of experiment.

¹⁰This must be done in real time, at the FPGA logic level, so must be supported by the device.

Quantum Zeno: the Quantum Zeno effect [79, 80] basically allows to slow down the time evolution of a quantum system by applying multiple frequent measurements. This can potentially lead to an increase of the effective T_1 and T_2 , thus leading to better gate fidelities. Also this technique needs the measurements to be "fully" non-destructive.

Quantum tomography: also called "full state tomography" [81] is the process to reconstruct completely a quantum state, by using multiple gates and measurements on different replica of the same qubit state. It is a powerful, and slow, tool to fully understand if a gate it's behaving as expected.

Chapter 4

Two-qubits characterization and calibration

After the last chapter, and with some more work and fine tuning, the experimenter should be able to reach a proper single qubit calibration. However, to execute "useful" circuits, the capability of entangling multiple qubits and executing gates between them is critical. Indeed, the real difference between classical and quantum computations *is* the entanglement effect: from a probability point of view, all the results that we obtained are still compatible with classical theories (or hidden-variables theories) and we didn't yet encounter any real proof of quantum probability.

In this section, we will learn the principles of two-qubits gates and how to perform a basic calibration of those.

There are two main two-qubits gates that can be implemented: CZ and iSWAP. The calibration is more or less the same and the complexity to achieve them does not differ. I will focus on the iSWAP gate since I found better literature about it, but everything can be easily transposed for the CZ [75, 82, 83].

The interaction of two qubits is done via a coupling capacity that produces an interaction term of:

$$H_{qq} = g\sigma_{y1} \otimes \sigma_{y2} \quad (4.1)$$

This Hamiltonian can be decomposed in σ^\pm (ladder operators) and, dropping the fast rotating terms, we can arrive at:

$$H_{qq} = g(e^{\delta\omega_{12}t}\sigma^+\sigma^- + e^{-\delta\omega_{12}t}\sigma^-\sigma^+) \quad (4.2)$$

where we have $\delta\omega_{12} = \omega_{q1} - \omega_{q2}$.

The last equation is not particularly clear and easy to understand, but this all changes when the two qubits share the same frequency:

$$H_{qq} = g(\sigma^+\sigma^- + \sigma^-\sigma^+) = \frac{g}{2}(\sigma_x\sigma_x + \sigma_y\sigma_y) \quad (4.3)$$

And this equation, when applied as a time evolution, produces an iSWAP gate.

The iSWAP gate can be written in the canonical basis as:

$$iSWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -i & 0 \\ 0 & -i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.4)$$

while the time evolution of H_{qq} is:

$$U_{qq}(t) = e^{-i\frac{g}{2}(\sigma_x\sigma_x + \sigma_y\sigma_y)t} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(gt) & -i\sin(gt) & 0 \\ 0 & -i\sin(gt) & \cos(gt) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.5)$$

with exact equality for $T = \frac{\pi}{2g}$.

So, what we have just demonstrated can be summarized in saying that the iSWAP can be implemented simply by activating the two-qubits interaction for a fixed time.

4.1 Avoided crossings

The first step required to reach two-qubits gates is the avoided crossing experiment [84, 85]. The objective here is to find, approximately, the flux and frequency where the interaction will happen.

In particular, since we are considering the iSWAP, we want to find the point in the flux-frequency space, where we can have a jump of states between $|01\rangle$ and $|10\rangle$.

We are considering to have two flux-controllable qubits already calibrated (at the single qubit level) so we are supposing to have already studied the flux dependency of those, but let's repeat some information. Both qubits have a dependency between frequency and the flux passing through their SQUID. They are characterized by a certain sweetspot, namely a certain flux value that moves the qubit to a frequency that is the maximum frequency reachable. The application of less or more flux will reduce the frequency of the qubits.

Note that we now want to control both qubits at the same time and a flux applied at one will probably change also the other qubit. It's critical to take this in consideration in the research for the sweetspots.

Anyway, considering one qubit (A) at the sweetspot and the second qubit (B), that has a higher frequency at the sweetspot, swept in bias, we can obtain a plot similar to the one in fig. 4.1.

Note that in fig. 4.1 the lines are the ones drawn by the peaks in qubit spectroscopies that describes the transition $|0\rangle \leftrightarrow |1\rangle$.

Note also that, if we define the two-qubits state as $|A\rangle \otimes |B\rangle = |AB\rangle$, we can identify the orange curve in the plot as the one of the $|00\rangle \leftrightarrow |01\rangle$ transition and the blue one as the $|00\rangle \leftrightarrow |10\rangle$ transition. So at the green points we have precisely the condition required for the iSWAP and, to reach it, it seems that is just needed to apply two

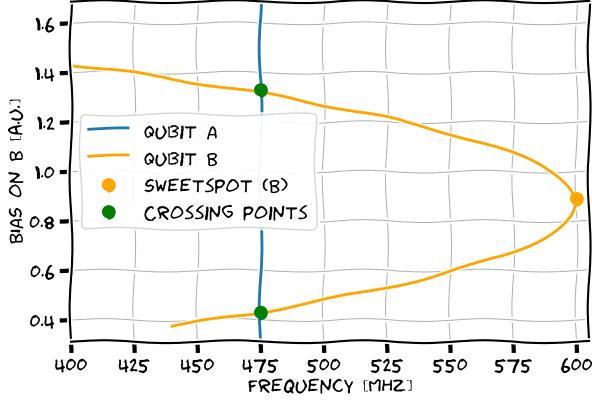


Figure 4.1: Possible behaviour of two qubits flux dependence.

contemporary flux pulses at two qubits (with an amplitude computed from the distance from the sweetspot).

What actually is happening at the crossing points however, is an hybridization of the two states that can be solved analytically or just seen directly by zooming in the plot presented. See for example fig. 4.2 (note that the axes are swapped).

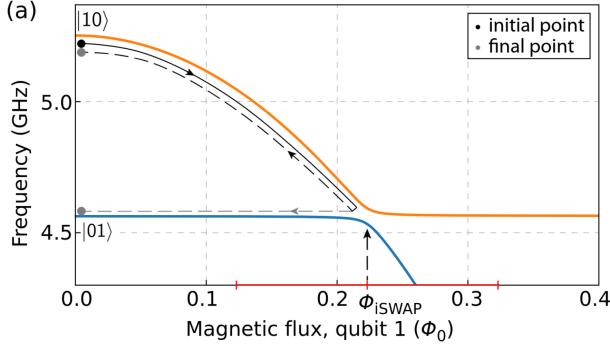


Figure 4.2: Avoided crossing.

In fig. 4.2 it's defined also an initial and final state, that will be used for the iSWAP.

Finding the avoided crossings is the first step for two-qubits gates characterization. After that, we know how much flux is needed to move the higher frequency qubit to the interaction point and we have proved (through avoided crossing) that an interaction is indeed present (but we still need to calibrate it).

Note that there are always two interaction points, but this are usually at different flux absolute values, so it's preferred to use the lower one (although the interaction would happen equally).

Note also that we now saw the avoided crossings for the iSWAP, but for the CZ the experiment is similar. For the CZ we look for the interaction $|02\rangle \leftrightarrow |20\rangle$. To see the second level transition frequency, it is sufficient to perform a qubit spectroscopy

with more amplitude so that, just before the $|0\rangle \leftrightarrow |1\rangle$ transition peak, also the $\frac{|0\rangle + |2\rangle}{\sqrt{2}}$ can appear (divided by two because usually two combined photons can give those high energies).

A larger scan in frequencies can give us fig. 4.3.

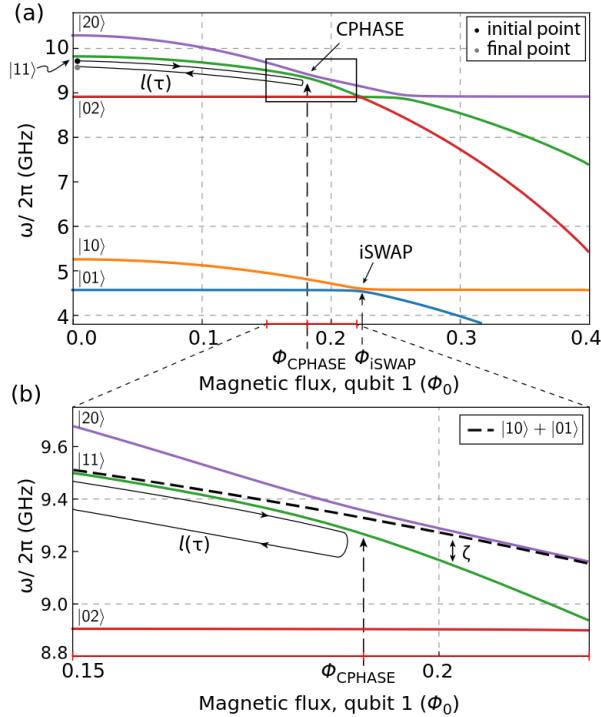


Figure 4.3: Avoided crossing for CZ and iSWAP.

In fig. 4.4 are shown the results of a scan large both in flux and frequency. The 4 horizontal lines are a low resolution visualization of the 4 avoided crossings (2 for CZ and 2 for iSWAP) with the inner two being the ones for CZ and the outer two of iSWAP.

In fig. 4.5 a zoom in on one of the iSWAP crossing points show the states hybridization and the avoided crossing phenomenon.

4.2 Chevron plot

Now that we found an estimation of the interaction point, it's time to calibrate the gate [86].

First we look back at fig. 4.2 that was also showing the evolution of a state starting in $|10\rangle$. Indeed if we prepare a state $|10\rangle$, using an already calibrated π -pulse on the first qubit, we can follow it up with a second pulse that moves the state along the orange curve. This second pulse will be applied to qubit B and sent through the flux line on the qubit. Ideally, there is no movement along the line, but only jumps: in the sense that,

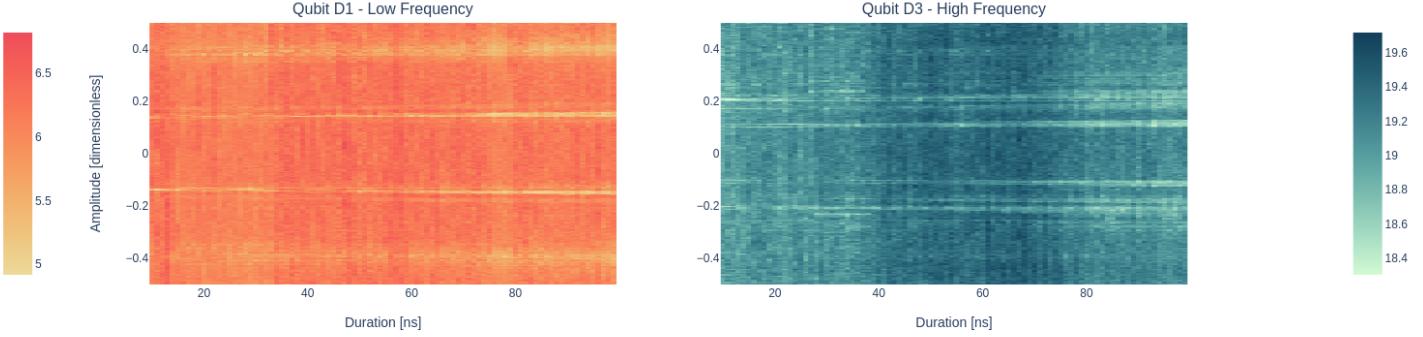


Figure 4.4: Large scan where all the 4 interaction biases are visible.

if we want to send a certain flux, we would like to not have any transient. Because of this the flux pulse is initially defined as a rectangular pulse.

Let's try to understand the arrows drawn in fig. 4.2. If we send a flux pulse with a flux of 0.1 (considering the numbers in the plot) this will not be enough to reach the interaction point and, after the flux pulse is ended, measuring the two qubits will show that we still have a $|10\rangle$ state. The same thing happen, more or less, if more flux than needed is sent.

If the flux value is correct, Rabi-like oscillations are produced between the two states as indicated in eq. (4.5). Actually, even at wrong fluxes we still achieve some Rabi-like oscillations, but with lower amplitude, higher frequencies, and lots of leakage to higher levels.

What we want to calibrate then, is the exact value of the flux needed for the two-qubit gate and the time T to achieve the gate. We perform an experiment with the following pulse sequence:

- a first π -pulse is sent to qubit A (lower frequency);
- afterwards, a flux pulse is sent to qubit B (higher frequency);
- afterwards, a measurement is performed on both qubits;
- the experiment is repeated for various biases (around the expected crossing point) and various lengths.

The results of this experiment are plotted in a 3D graph with pulse-flux Vs pulse-length Vs P_{01} . The z axis can actually represent different parameters: in particular probabilities regarding single qubit measurements (both for qubit A and B) or two-qubits measurement or, again, amplitude (defined as $\sqrt{i^2 + q^2}$) measurements. Something that can happen, in particular for CZ where the second excited state is involved, is leakage

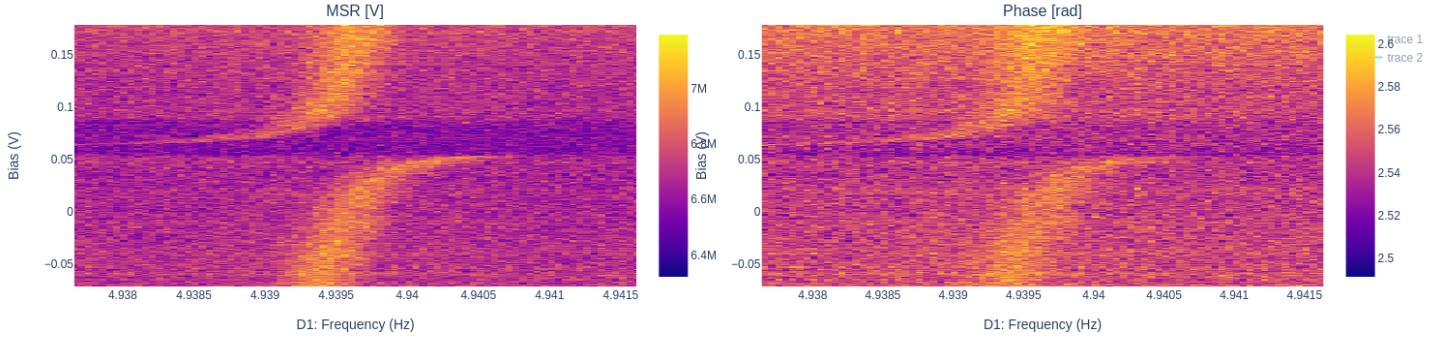


Figure 4.5: Zoom in on an avoided crossing (iSWAP). The bias is applied only to the high frequency qubit, while the lower is left at the sweetspot. Measuring the low-frequency qubit gives the avoided crossing.

outside the computing space that is usually not discriminated¹. Therefore it's usually preferred to use amplitude in case of CZs, while for iSWAP both probabilities and amplitude should work fine.

The expected plot for this experiment is presented in fig. 4.6.

This experiment, performed with the ZCU216 using the coded `Qibocal` routines is presented in fig. 4.7.

The flux required for the two-qubits gate is the center one in the chevron plot, while the length of the pulse is chosen as half of the distance between two consecutive peaks at that flux (exactly as in the Rabi experiment).

4.3 Flux pulse shape correction

The result of the last experiment should be exactly symmetric, as presented in fig. 4.6, but this is hardly the case. Usually the result is much more distorted as presented in fig. 4.8, for example.

The problem in this case is the effective shape of the flux pulse. We are now sending a rectangular pulse, with the idea that we can use it to perform a jump between two different flux values, without any transients. In reality, the pulse gets partially deformed by the quality of the line, in particular its capacitance, that transform the pulse from a rectangular to something like what is presented in fig. 4.9.

The solution to this problem is to change the shape of the flux pulse to an ideal rectangular to something that will produce an effective rectangular [87, 88]. To achieve this, there are different possibilities, but the most straightforward consists in just implementing a new pulse shape and fine tune it manually using the symmetry of the chevron

¹A possibility it's then to discriminate also the state 2 (on top of 0 and 1) in the IQ plane.

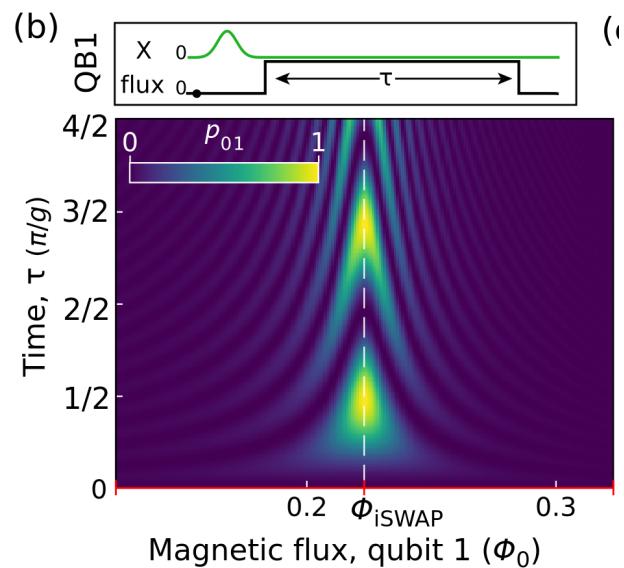


Figure 4.6: Expected plot for a two-qubits chevron.

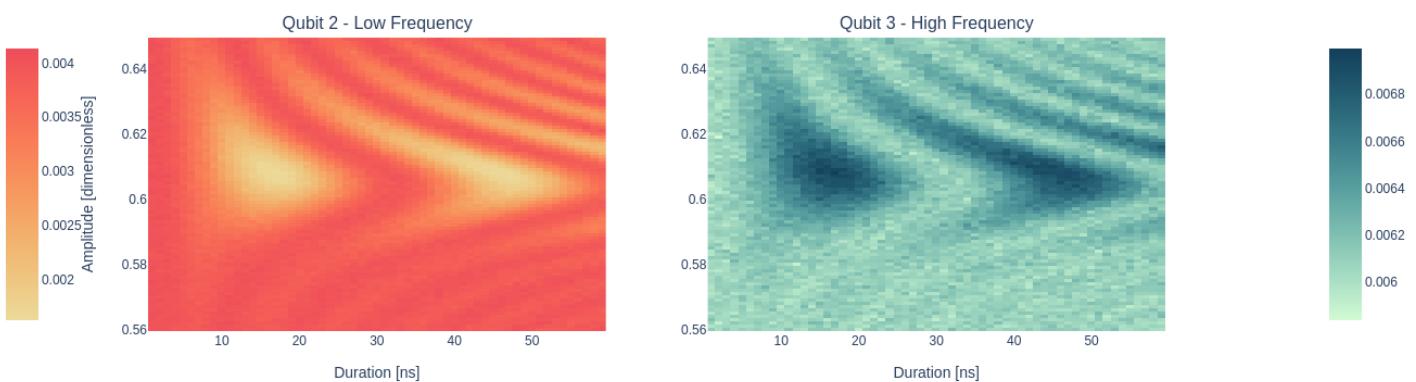


Figure 4.7: Chevron plot for CZ.

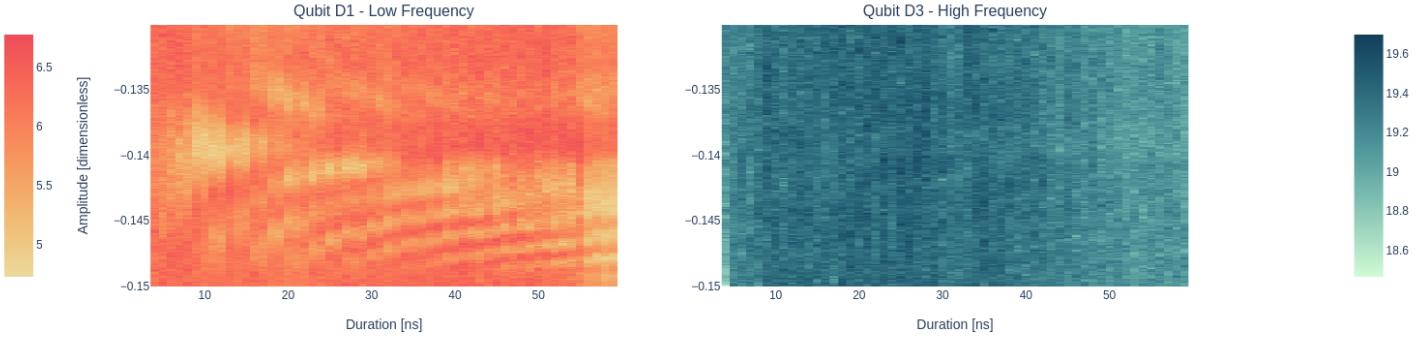


Figure 4.8: Chevron plot for CZ. The distortion, that will be fixed in the next section, causes the asymmetry higher-lower bias. The leakage causes the chevron to not close and to not appear in the high frequency qubit.

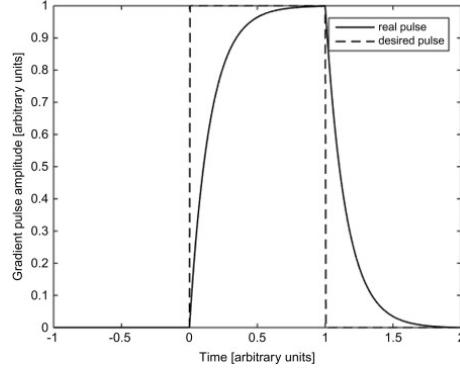


Figure 4.9: Rectangular pulse distortion.

plot of the last experiment as figure of merit.

A sensible ansatz for the flux pulse is presented in eq. (4.6).

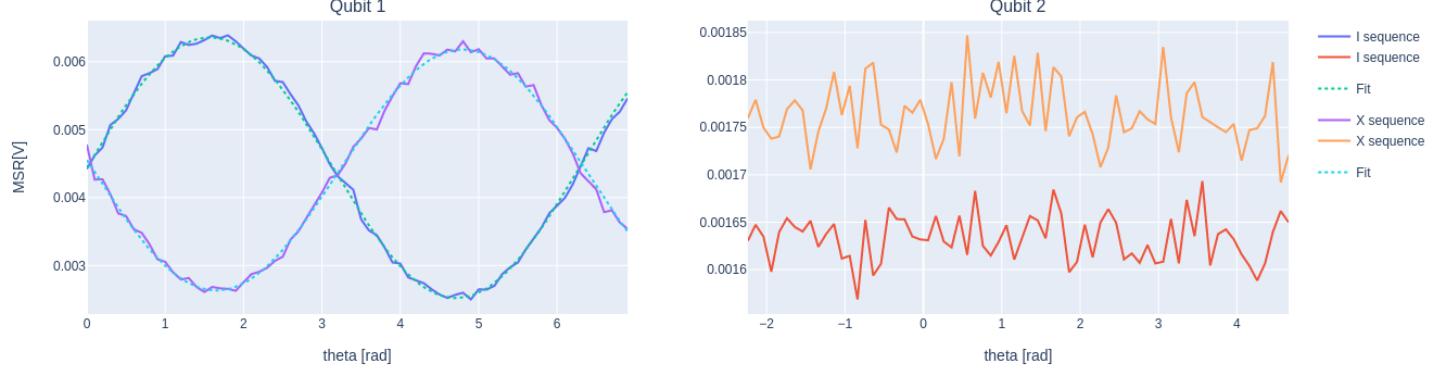
$$y = A * \frac{(\exp(-x/v)) + g * \exp(-x/\tau)}{1 + g} \quad (4.6)$$

where basically the rectangular pulse is corrected with an initial exponential function, to counter the initial transient.

4.4 Virtual phase correction

After finding the right shape, we still have to consider a last calibration parameter: the phase. The discussion here will be around the CZ since it presents some more problems in respect to the iSWAP, but as always everything is more or less interchangeable.

Phase correction Qubit 1



Phase correction Qubit 2

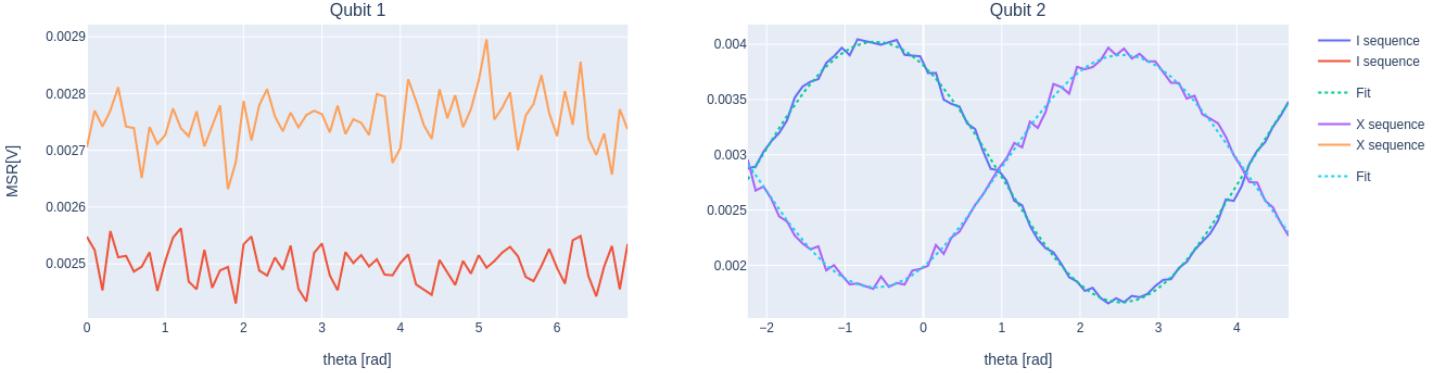


Figure 4.10: Calibrated virtual phase correction plot.

The problem is that so far we have ignored the role of the single-qubit phases acquired by tuning the qubit frequency (caused by simple time evolution). So, effectively we never implemented a CZ (or a iSWAP) but a CZ with the addition of a spurious relative phase between the qubits. Critically, this phase is not dependent on the state of the control qubit, but is constant in all cases, so it will be fairly easy to implement a virtual correction once calibrated (so consider the presence of this phase for all successive pulses and measurements).

The experiment we perform is the following:

- we initialize the system performing a $Y90$ pulse on the low frequency qubit and either a I or a X gate on the high frequency one;
- we apply the flux pulse for the two-qubit interaction;
- we undo the initial rotation on the high frequency qubit, by applying I or X again;

- we apply a $\Theta(90)$ pulse, so a rotation of 90° around a certain angle θ ;
- we measure the two qubits;
- we repeat the measurement for multiple angles and for both I and X initial states.

Ideally, the high frequency qubit should not be affected at all by the gate, eventually some leakage to the $|2\rangle$ state can be visible.

For the low frequency qubit, we should see, depending on the initial state of the control qubit, two different sinusoidal with a 90° phase difference. What we will actually see is presented in fig. 4.10.

The phase difference is not 90° as expected but $(90 + \epsilon)^\circ$ with ϵ being the virtual phase to consider for later sequences and measurements.

Chapter 5

Applications

The qubits control system developed within this thesis can be used for various experiments in different quantum technology fields. Although the integration in Qibolab is focused on quantum computing, the final objective is to have a complete instrument able to fully control of a qubit state. So it is possible to imagine also non-computing applications.

In any case, let's focus on the two main applications where Qibosoq could be useful and could be used in the short term. These are:

- in quantum computing, for quantum machine learning applications.
- in quantum sensing, in particular for the QubIT project;

5.1 Quantum machine learning: determining probability density functions

The RFSoC-based control system that was developed for this thesis has already been used in quantum machine learning applications, in particular to fit probability density functions [4].

This is a relevant short-term application because even a single not-optimally-tuned qubit can be work fine.

To the reader, using a quantum computer for fitting, might seem useless. It's a fair doubt, but there are still situations where the analytical underlying distribution is not known and the problem cannot be efficiently solved classically.

For example, the reliable determination of probability density functions (PDF) from data samples is still an active research topic in studies of fundamental physics.

With quantum computing and, specifically, adiabatic evolution algorithms, it is possible to approximate the underlying distribution via a circuit-based quantum device.

Given a one-dimensional function $f(t)$ with $t \in [0, T]$, we can build a regression model by choosing an observable such that two Hamiltonians H_0 and H_1 exist and respect the condition of having their ground state energy as the two values defining the range to which the function will be bounded.

We can then interpret the regression problem as the search for a time dependent Hamiltonian such that its ground state energy evolves as $f(t)$:

$$\langle H(t) \rangle = f(t) \quad (5.1)$$

The Hamiltonian can be also written in terms of a new function $s(t; \theta)$, called *scheduling function*, that describes a variational circuit defined by a set of parameters θ :

$$H(t) = [1 - s(t; \theta)]H_0 + s(t; \theta)H_1 \quad (5.2)$$

In this way, the problem is reduced to finding the right parameters θ .

The complete, schematic, procedure is at follows:

1. we generate a sample of variables $\{x\}$ from a chosen distribution;
2. we compute the empirical *cumulative density function* (CDF) from the extracted samples;
3. we select N_{train} data samples such that their values match some of the evolution times controlled by s ;
4. we map each pair (x_i, f_i) into (τ_i, E_i) with the second pair representing two generic values of the evolution time/energy of the target observable evaluated at the evolved state with $\tau = t/T$;
5. we define a loss function to minimize as:

$$J = \frac{1}{N_{train}} \sum_{j=1}^{N_{train}} (f_j - E_j(\theta))^2 \quad (5.3)$$

It's important to have a monotonic regression function, eventually, we could add extra penalty terms that, increasing with j , would assure the monotony;

6. we perform a training of the parameters θ to minimize the loss function. This can be done with classic, quantum or hybrid schemes.

As example, we take some data samples from the NNPDF4.0 [89] PDF grid. Using a variational ansatz composed of consecutive rotations [90], we encode the values of x and proceed with the optimization minimising the MSE using the ADAM [91] optimizer. In this case, the optimization was done partially on classical hardware, as a pre-training, and part on a TII single qubit device controlled by **Qibosoq**.

In fig. 5.1, an example of a plot obtained with this procedure is presented. Both the expected PDF and the fitted one, with **Qibosoq** and the **RFSoC4x2**, are presented.

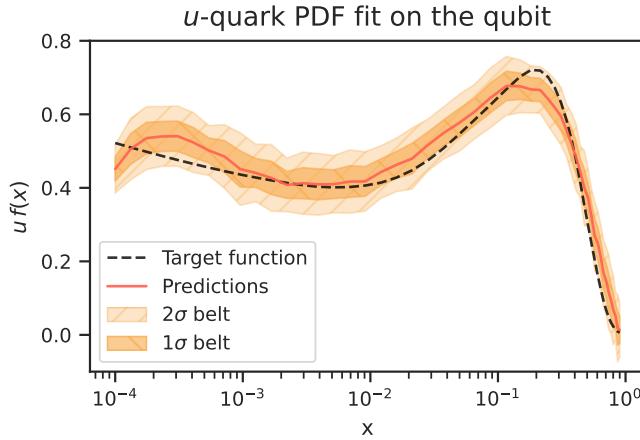


Figure 5.1: PDF of the u -quark, fitted with the QML model with a **Qibosoq**-controller RFSoC.

5.2 Quantum sensing applications

For the moment, **Qibosoq** has not been used in sensing applications, but it has been designed to be flexible and to support different kind of applications, from the circuits-based QML that we saw in last section to more pulse-based optimization protocols (for optimal-control, for examples) or for sensing applications.

What are the special needs of quantum sensing, from a control device / software point of view?

It is difficult to have a complete list of required features, but we could say that the main elements are:

- extensive control of the pulses to execute (shaping, timing, frequency modulation, phases etc.);
- fast and precise acquisition;
- support for continuous measurements;
- good scaling properties;
- support for highly multiplexed systems;

Qibosoq natively has, leveraging different **Qick** functionalities, most of these characteristics.

In particular, through the Python API, the user has total control of the pulses and can both work optimizing speed and dead times, by leveraging all the pre-defined pulse shapes included in **Qibosoq**, or focusing on optimal control, by defining in the client specific pulses from their IQ values.

The intrinsic characteristic of a RFSoC system give full control and flexibility on the frequency requests. In particular, when comparing the three currently supported Xilinx boards with the other commercial systems available for control at TII (Qblox, Quantum Machines, Zurich Instruments) the bandwidth of a single RFSoC channel (that can go above 10 GHz) is not-comparable in respect to the standard bandwidths of ≈ 200 MHz (without local oscillators). This speeds up *incredibly* any experiment where there is the need to probe very different frequencies.

Currently, **Qick** does not support continuous measurements and this reflects on **Qibosoq** with the same limitation. The problem, from the point of view of **Qick**, stands in the use of memory that quickly fills the buffer size defined in FPGA logic. Even tho this is a big limitation for sensing application, it's partially patchable through **Qibosoq**. Since **Qibosoq** can be included easily in any Python client and is designed to minimize any latency, it should be possible to apply repeated measurement windows one after the other, without loosing excessive statistics in between them.

In any case, this issue should be taken into account and precisely measured for any real application of rare phenomena.

Regarding the scaling capabilities, the **Qick** team is currently updating their firmwares so the is possible to synchronize different RFSoCs and use them collectively *as a cluster-like system*. The main **Qick** developers admitted that this firmware upgrade will cause some problem in the **Qick** software department: indeed the current proposed way of managing a RFSoC directly with **Qick** involves a direct connection to the board, this is not possible when dealing to multiple boards at the same time. In this problem, the **Qibosoq** layout offers an easy solution: having already an external client responsible of managing communication with the board(s), it should be easy to give to it the new responsibility of splitting the program into different subsection for the different boards.

The high multiplexability is a concept that, for qubits, has not really been under development. However, **Qick** has specific firmwares to control highly multiplexed arrays of Microwave Kinetic Inductance Detector [92], often referred to as MKIDs, (up to 4096 detectors with a single board). Although at the moment, the **Qibosoq** development has been focused on qubit control, so it does not support this special firmware, it has been designed with flexibility as one of the main priority, since we knew from the start that **Qick** would be upgraded independently from it and possibly without much interest in preserving backwards compatibility.

Thanks to this, it should not be difficult to extend support for the MKIDs firmware, although no development has yet been done in this direction.

Photon counter for dark matter detection

As a theoretical example of quantum sensing application, we can take an experiment for axions detection [93].

In this section, we will present the code required to replicate the experiment, showcasing how a complex experiment can become trivial using the **Qibosoq** API. The experiment itself requires a hardware setup that stops me to be able to actually replicate it now, but still is a good example of how easily the implementation can become.

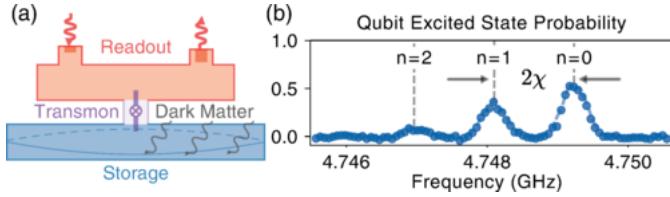


Figure 5.2: Hardware setup required of the axion-detection experiment.

This hardware setup is presented in fig. 5.2 (a). The detection system requires a superconducting qubit (transmon) used as a bridge between a readout resonator (in the image depicted as a cavity) and an additional storage-cavity.

The idea of this detector is to exploit the Primakoff effect that should cause potential axions to convert to photons in resonance to the storage frequencies. Now, we can use the qubit to build an extremely precise photon counter and, counting the average number of photons we may see statistically relevancy of axions existence.

We use, for this, two concepts already encountered in chapter 3:

- dispersive shift;
- Ramsey interferometry.

We exploit the dispersive shift considering that a different photons number has a different effect on the qubit frequency. Indeed the Hamiltonian of the system can be written as (ω_c is the cavity frequency; a and a^\dagger the ladder operators of the cavity; ω_q the frequency of the qubit; χ the dispersive shift and σ_z the well-known Pauli matrix):

$$H/\hbar = \omega_c a^\dagger a + \frac{1}{2} (\omega_q + 2\chi a^\dagger a) \sigma_z \quad (5.4)$$

So, effectively, the qubit frequency changes with $2\chi a^\dagger a$.

To measure the different "detuning" caused by a different photons population in the storage cavity, we can use the Ramsey protocol. In the Dixit paper [93] that proposes this experiments, the measurement protocol works not with two $\pi/2$ -pulses but with a $+\pi/2$ -pulse and a $-\pi/2$ -pulse. The effect is more or less the same and it is easily coded in **Qibosoq**.

Finally, to achieve exponential suppression of the readout errors, we have to repeat the measurement (the two drive pulses and the readout one) multiple consecutive times, leveraging the non-destructiveness of the measurement. A scheme of the sequence to be executed is present in fig. 5.3.

To increase the fidelity of the experiment, a Markov-chain analysis can be employed and it already proved to be of use. For this example, however, we will focus on the acquisition part that can be performed using **Qibosoq**, while we will leave out the data processing phase.

Using the **Qibolab** and **Qibosoq** API, the experimental procedure will be coded as:

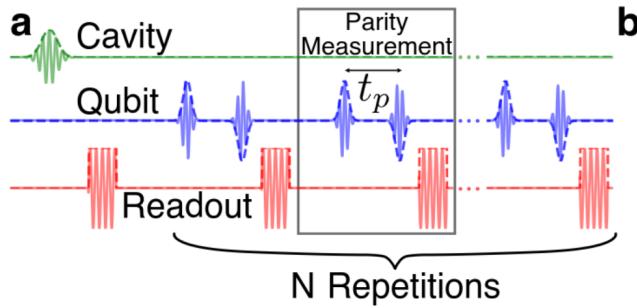


Figure 5.3: Overview of the pulse sequence to be executed for the axion-detection experiment.

```

1 from qibolab import create_platform
2 from qibolab.pulses import PulseSequence
3
4 # we instantiate the platform (qubit + RFSoC)
5 platform = create_platform("platform_name")
6
7 # we initialize an empty pulse sequence
8 sequence = PulseSequence()
9
10 N_repetitions = 30
11 for i in N_repetitions:
12     # we add a first pi-half pulse
13     sequence.add(platform.create_RX90_pulse())
14     # we add the - pi-half pulse
15     sequence.add(platform.create_RX90_pulse())
16     sequence[-1].amplitude = - sequence[-1].amplitude
17     # we add a measurement
18     sequence.add(platform.create_MZ_pulse())
19
20 # we perform the experiment, executing the sequence
21 results = platform.execute_pulse_sequence(sequence)

```

See that, in 21 lines (with an abundance of comments), we coded and performed the full experiment. Clearly, some work has been hidden here, in particular the calibration parameters of the qubit, that have to be computed via the various experiments detailed in the last sections, as well as all the data processing.

This example shows how much Qibosoq can simplify and accelerate research.

Chapter 6

Conclusions

In this thesis, after presenting the main theoretical concepts and the main instruments (hardware and software) used, I focused on the characterization and calibration experiments required to fully control a superconducting quantum device. A detailed description of all the experiments has been provided along with ideal and real plots for different scenarios.

Software development Although the explanation of these experiments is the main part of this thesis, the time spent at the Technology Innovation Institute was in large part dedicated to develop the software tools required for the experiments themselves. This included the development of parts of **Qibolab** and of **Qibosoq** of which some details were given in section 2.2.

Both of these software tools are open source and are being officially presented to the research community via papers [37] currently under review for the Quantum journal.

It is useful, for the purpose of better explaining my role in the development of these tools, to provide a brief description of them when I first started and when I finished the thesis.

When I first arrived, **Qibolab** was in an alpha-testing phase and was not fully released yet. Remember that the main idea behind **Qibolab** is to support, with the same interface, multiple devices and instruments. At that time, however, it supported only Quantum Machines devices and some Qblox devices (with outdated firmware). My main role was to add to **Qibolab** the full support of the RFSoC FPGAs compatible with the **Qick** project, but I also worked in the development of the general interface, helping reaching the first stable release of the software.

For what concerns **Qibosoq**, when I first arrived at TII, it was nothing but a prototype script, it was not compatible with the **Qibolab** interface and therefore was not of any real use. I developed it to the first stable release and reached a stage where all the major features supported by **Qick** and **Qibolab** are supported and integrated in the software. The role of **Qibosoq** could be critical for the research community, since it creates a connection between an open source software for control, **Qibolab**, and a high-precision and economically-feasible hardware solution as the RFSoCs and **Qick**.

Calibration experiments To develop and test the software solutions introduced in Qibolab and Qibosoq, continuous hardware testing on real qubits was required, as well as the software implementation of various calibration and characterization experiments.

In this thesis, I detailed the experiments required to fully calibrate a single qubit device, as well as the first main experiments required for two-qubit gates calibration. I tried to set up the calibration sections as a practical manual, so that it may be used by novice experimenters in the future. Indeed many more experiments can be developed to increase readout and gate fidelities, but following the layout of routines detailed in this thesis will give full and acceptable control capacities on a standard superconducting qubit. Moreover, the user of this "manual" will also gain a good enough knowledge on the underlying quantum-mechanical and cQED principles exploited for readout and control. A small list of other possible calibration experiments was also provided in section 3.17, to show the "infinite" possible number of experiments.

Algorithmic applications The RFSoC-based setup I developed during this thesis, along with the qubits I characterized and calibrated, was already used both by me and by other colleagues with no knowledge of the underlying complexity of Qibosoq and Qick. This demonstrates how relevant it is to produce a tool with a simple interface, that can be used by experimenters without needing to split focus between software and the experiment itself.

Qibosoq has been used for algorithmic applications, of which the fitting procedure in section 5.1 is an example. Moreover, it has also been used to develop a generative quantum neural network (article soon to be published).

In general, the RFSoC-based system has been proved to be as reliable as the commercial solutions and slightly faster than those, in particular for circuit-based experiments (so algorithmic applications) as presented in fig. 6.1 [37, 46].

Outlook and quantum sensing applications For the moment, Qibosoq and the RFSoC-based setup was used just for computing applications. The software too was developed with only calibration experiments and circuits applications in mind.

To fully unlock the potential of qubits in the particle physics research world, however, and in particular to offer Qibosoq as a complete control solution, independently from the type of final use, some development and testing is still needed.

Various experimental protocols are already been proposed to use qubits as detectors and, in any case, quantum sensing is one of the most interesting active topic of research in quantum technologies. The experiment presented in section 5.2 is an example of feasible application.

Note that the majority of elements required for sensing (in general, we are always talking about pulses and measurements) are already supported, but some changes in the interface and some more features are required to make Qibosoq a reliable tool (for example, the support for continuous acquisition modes).

Moreover, RFSoC FPGAs are not limited to the control of superconducting qubits and could be used also for different types of technologies such as photonic quantum

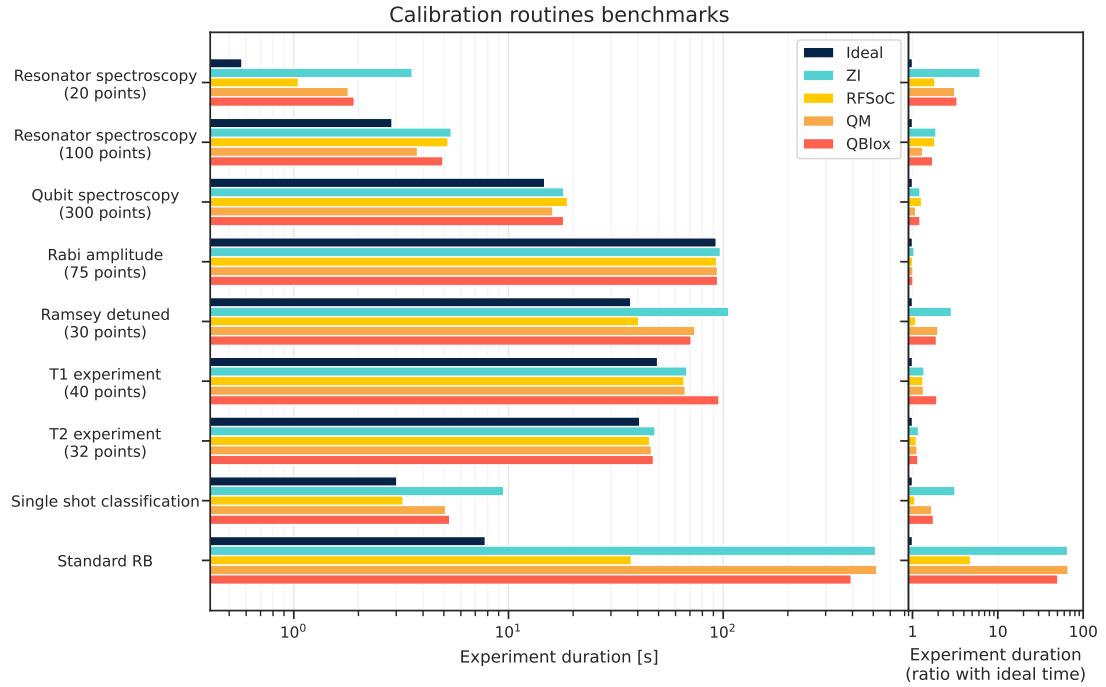


Figure 6.1: Execution time of different qubit calibration routines on various electronics. On the left side there is the absolute times in seconds for each experiment. The ideal time (black bar) shows the minimum time the qubit needs to be affected in each experiment. On the right side the ratio between actual execution time and ideal time is shown.

computing (with also some hardware modification).

In the future, **Qibosoq** will be extended for applications different from quantum computing and, in particular, for quantum sensing researches.

Bibliography

1. Blais, A., Grimsmo, A. L., Girvin, S. & Wallraff, A. Circuit quantum electrodynamics. *Reviews of Modern Physics* **93**. <https://doi.org/10.1103/revmodphys.93.025005> (May 2021).
2. Kalfus, W. D. *et al.* High-Fidelity Control of Superconducting Qubits Using Direct Microwave Synthesis in Higher Nyquist Zones. *IEEE Transactions on Quantum Engineering* **1**, 1–12 (2020).
3. Stefanazzi, L. *et al.* The QICK (Quantum Instrumentation Control Kit): Readout and control for qubits and detectors. *Review of Scientific Instruments* **93**. <https://doi.org/10.1063/5.0076249> (Apr. 2022).
4. Robbiati, M., Cruz-Martinez, J. M. & Carrazza, S. *Determining probability density functions with adiabatic quantum computing* 2023. <https://arxiv.org/abs/2303.11346>.
5. *Qub-IT: Quantum sensing with superconducting qubits for fundamental physics* <https://web.infn.it/qub-it/>. 2020.
6. Arbey, A. & Mahmoudi, F. Dark matter and the early Universe: A review. *Progress in Particle and Nuclear Physics* **119**, 103865. <https://doi.org/10.1016/j.ppnp.2021.103865> (July 2021).
7. Belenchia, A. *et al.* Quantum physics in space. *Physics Reports* **951**, 1–70. <https://doi.org/10.1016/j.physrep.2021.11.004> (Mar. 2022).
8. *Dynamics in Disk Galaxies* 2015. <https://pages.astronomy.ua.edu/keel/galaxies/diskdyn.html>.
9. *Gravitational Lenses: Spyglasses Into the Universe* 2022. <https://scitechdaily.com/gravitational-lenses-spyglasses-into-the-universe>.
10. Springel, V. *et al.* Simulations of the formation, evolution and clustering of galaxies and quasars. *Nature* **435**, 629–636. <https://doi.org/10.1038/nature03597> (June 2005).
11. Einasto, J. Dark Matter. <https://arxiv.org/abs/0901.0632> (2009).
12. Petriello, F. J. & Zurek, K. M. DAMA and WIMP dark matter. *Journal of High Energy Physics* **2008**, 047–047. <https://doi.org/10.1088/1126-6708/2008/09/047> (Sept. 2008).

13. Caputo, A., Millar, A. J., O'Hare, C. A. & Vitagliano, E. Dark photon limits: A handbook. *Physical Review D* **104**. <https://doi.org/10.1103/physrevd.104.095029> (Nov. 2021).
14. *Quantum Computing Modalities: A Qubit Primer Revisited* <https://quantumtech.blog/2022/10/20/quantum-computing-modalities-a-qubit-primer-revisited/>.
15. Arute, F. *et al.* Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510. <https://doi.org/10.1038/s41586-019-1666-5> (Oct. 2019).
16. Tinkham, M. *Introduction to Superconductivity: v. 1* 2nd ed. (Dover Publications, Mineola, NY, June 2004).
17. Langford, N. K. *Circuit QED - Lecture Notes* 2013. <https://arxiv.org/abs/1310.1897>.
18. Pobell, F. *Matter and Methods at Low Temperatures* <https://doi.org/10.1007/978-3-540-46360-3> (Springer Berlin Heidelberg, 2007).
19. Kockum, A. F. & Nori, F. in *Fundamentals and Frontiers of the Josephson Effect* 703–741 (Springer International Publishing, 2019). https://doi.org/10.1007/978-3-030-20726-7_17.
20. Vool, U. & Devoret, M. Introduction to quantum electromagnetic circuits. *International Journal of Circuit Theory and Applications* **45**, 897–934. <https://doi.org/10.1002/cta.2359> (June 2017).
21. Tafuri, F. <https://vdoc.pub/documents/fundamentals-and-frontiers-of-the-josephson-effect-3agaqrlc6u70>.
22. Josephson, B. Possible new effects in superconductive tunnelling. *Physics Letters* **1**, 251–253. [https://doi.org/10.1016/0031-9163\(62\)91369-0](https://doi.org/10.1016/0031-9163(62)91369-0) (July 1962).
23. Bladh, K., Duty, T., Gunnarsson, D. & Delsing, P. The single Cooper-pair box as a charge qubit. *New Journal of Physics* **7**, 180–180. <https://doi.org/10.1088/1367-2630/7/1/180> (Aug. 2005).
24. ETH. https://qudev.phys.ethz.ch/static/content/courses/QSIT11/QSIT11_V06_slides.pdf.
25. Koch, J. *et al.* Charge-insensitive qubit design derived from the Cooper pair box. *Physical Review A* **76**. <https://doi.org/10.1103/physreva.76.042319> (Oct. 2007).
26. Roth, T. E., Ma, R. & Chew, W. C. *An Introduction to the Transmon Qubit for Electromagnetic Engineers* 2021.
27. McClintock, P. Cryogenics. *Encyclopedia of Physical Science and Technology*, 37–59 (Jan. 2003).
28. Craig, N. & Lester, T. Hitchhiker's Guide to the Dilution Refrigerator (2004).
29. Devices, A. <https://www.ieee.li/pdf/essay/dds.pdf>.

30. Huang, H.-L., Wu, D., Fan, D. & Zhu, X. Superconducting quantum computing: a review. *Science China Information Sciences* **63**. <https://doi.org/10.1007/s11432-020-2881-9> (July 2020).
31. *Contralto chip* 2022. <https://www.quantware.eu/product/contralto>.
32. Yaakobi, O., Friedland, L., Macklin, C. & Siddiqi, I. Parametric amplification in Josephson junction embedded transmission lines. *Physical Review B* **87**. <https://doi.org/10.1103/physrevb.87.144301> (Apr. 2013).
33. *Silent Waves* <https://silent-waves.com/>.
34. Baulieu, G. *et al.* HEMT-Based 1 K Front-End Electronics for the Heat and Ionization Ge (CryoCube) of the Future Ricochet Experiment. *Journal of Low Temperature Physics* **209**, 570–580. <https://doi.org/10.1007/s10909-022-02896-5> (Nov. 2022).
35. *MiniCircuit* <https://www.minicircuits.com/>.
36. Ranadive, A. *et al.* Kerr reversal in Josephson meta-material and traveling wave parametric amplification. *Nature Communications* **13**. <https://doi.org/10.1038/s41467-022-29375-5> (Apr. 2022).
37. Efthymiou, S. *et al.* *Qibolab: an open-source hybrid quantum operating system* 2023.
38. *Qibo Quantum Computing Library* <https://www.qibo.science/>.
39. Qibo Team. *Qibo* <https://github.com/qiboteam/qibo>.
40. Qibo Team. *Qibolab* <https://github.com/qiboteam/qibolab>.
41. Qibo Team. *Qibosoq* <https://github.com/qiboteam/qibosoq>.
42. Gambetta, J. M., Motzoi, F., Merkel, S. T. & Wilhelm, F. K. Analytic control methods for high-fidelity unitary operations in a weakly nonlinear oscillator. *Phys. Rev. A* **83**, 012308. <https://link.aps.org/doi/10.1103/PhysRevA.83.012308> (1 Jan. 2011).
43. Naidu, P. S. V. *Modern Digital Signal Processing* (Alpha Science International, Sept. 2003).
44. Franks, L. E. *Signal Theory* (Prentice Hall, Old Tappan, NJ, Sept. 1969).
45. Hutchings, M. D. *et al.* Tunable Superconducting Qubits with Flux-Independent Coherence. *Physical Review Applied* **8**. <https://doi.org/10.1103/physrevapplied.8.044003> (Oct. 2017).
46. Carobene, R. *et al.* *Qibosoq: an open-source framework for quantum circuit RFSoC programming* 2023. <https://arxiv.org/abs/2310.05851>.
47. Gao, Y. Y., Rol, M. A., Touzard, S. & Wang, C. *A practical guide for building superconducting quantum devices* 2021. <https://arxiv.org/abs/2106.06173>.
48. Kim, Y. *et al.* Evidence for the utility of quantum computing before fault tolerance. *Nature* **618**, 500–505. <https://doi.org/10.1038/s41586-023-06096-3> (June 2023).

49. Naghiloo, M. *Introduction to Experimental Quantum Measurement with Superconducting Qubits* 2019.
50. Chen, Z. *Metrology of Quantum Control and Measurement in Superconducting Qubits* PhD thesis (UC Santa Barbara, 2018).
51. Fano, U. Effects of Configuration Interaction on Intensities and Phase Shifts. *Physical Review* **124**, 1866–1878 (Dec. 1961).
52. Limonov, M. F., Rybin, M. V., Poddubny, A. N. & Kivshar, Y. S. Fano resonances in photonics. *Nature Photonics* **11**, 543–554 (Sept. 2017).
53. Gao, J. *The Physics of Superconducting Microwave Resonators* PhD thesis (UC Santa Barbara, 2008).
54. Majer, J. *et al.* Coupling superconducting qubits via a cavity bus. *Nature* **449**, 443–447 (Sept. 2007).
55. Gu, X., Kockum, A. F., Miranowicz, A., Liu, Y.-x. & Nori, F. Microwave photonics with superconducting quantum circuits. *Physics Reports* **718-719**, 1–102 (Nov. 2017).
56. Greene, A. *Calibration and Utilization of High-Fidelity Two-Qubit Operations* PhD thesis (Massachusetts Institute of Technology, 2008).
57. Schuster, D. I. *et al.* ac Stark Shift and Dephasing of a Superconducting Qubit Strongly Coupled to a Cavity Field. *Physical Review Letters* **94**. <https://doi.org/10.1103/physrevlett.94.123602> (Mar. 2005).
58. Schuster, D. I. *et al.* Resolving photon number states in a superconducting circuit. *Nature* **445**, 515–518. <https://doi.org/10.1038/nature05461> (Feb. 2007).
59. Rabi, I. I. On the Process of Space Quantization. *Physical Review* **49**, 324–328. <https://doi.org/10.1103/physrev.49.324> (Feb. 1936).
60. Wallraff, A. *et al.* Approaching Unit Visibility for Control of a Superconducting Qubit with Dispersive Readout. *Physical Review Letters* **95**. <https://doi.org/10.1103/physrevlett.95.060501> (Aug. 2005).
61. Ramsey, N. F. A Molecular Beam Resonance Method with Separated Oscillating Fields. *Physical Review* **78**, 695–699. <https://doi.org/10.1103/physrev.78.695> (June 1950).
62. *Experiments on superconducting Josephson phase quantum bits* https://www.researchgate.net/figure/Illustration-of-the-Ramsey-experiment-in-the-Bloch-sphere-representation-of-the-qubit_fig39_47041555.
63. Klimov, P. *et al.* Fluctuations of Energy-Relaxation Times in Superconducting Qubits. *Physical Review Letters* **121**. <https://doi.org/10.1103/physrevlett.121.090502> (Aug. 2018).
64. *Quantum Information Lecture Series* <https://www.nii.ac.jp/qis/first-quantum/e/forStudents/lecture/index.html>.

65. Hahn, E. L. Spin Echoes. *Physical Review* **80**, 580–594. <https://doi.org/10.1103/physrev.80.580> (Nov. 1950).
66. Carr-Purcell-Meiboom-Gill Sequence 2023. <https://www.sciencedirect.com/topics/chemistry/carr-purcell-meiboom-gill-sequence>.
67. Caban, P., Rembieliński, J., Smoliński, K. A. & Walczak, Z. Classification of two-qubit states. *Quantum Information Processing* **14**, 4665–4690. <https://doi.org/10.1007/s11128-015-1121-y> (Sept. 2015).
68. Dietrich, H., de Graaf, W. A., Marrani, A. & Origlia, M. Classification of four qubit states and their stabilisers under SLOCC operations. *Journal of Physics A: Mathematical and Theoretical* **55**, 095302. <https://doi.org/10.1088/1751-8121/ac4b13> (Feb. 2022).
69. Motzoi, F., Gambetta, J. M., Rebentrost, P. & Wilhelm, F. K. Simple Pulses for Elimination of Leakage in Weakly Nonlinear Qubits. *Phys. Rev. Lett.* **103**, 110501. <https://link.aps.org/doi/10.1103/PhysRevLett.103.110501> (11 Sept. 2009).
70. Balasius, S. *Single-Qubit Gates Calibration in PycQED using Superconducting Qubits* (ETH, 2017).
71. Magesan, E., Gambetta, J. M. & Emerson, J. Characterizing quantum gates via randomized benchmarking. *Physical Review A* **85**. <https://doi.org/10.1103/physreva.85.042311> (Apr. 2012).
72. Kelly, J. *et al.* Optimal Quantum Control Using Randomized Benchmarking. *Physical Review Letters* **112**. <https://doi.org/10.1103/physrevlett.112.240504> (June 2014).
73. Proctor, T. J. *et al.* Direct Randomized Benchmarking for Multiqubit Devices. *Physical Review Letters* **123**. <https://doi.org/10.1103/physrevlett.123.030503> (July 2019).
74. Chow, J. M. *et al.* Randomized Benchmarking and Process Tomography for Gate Errors in a Solid-State Qubit. *Physical Review Letters* **102**. <https://doi.org/10.1103/physrevlett.102.090502> (Mar. 2009).
75. Krantz, P. *et al.* Single-shot read-out of a superconducting qubit using a Josephson parametric oscillator. *Nature Communications* **7**. <https://doi.org/10.1038/ncomms11417> (May 2016).
76. *Integration Weights Measurement* 2023. https://docs.zhinst.com/shfqa_user_manual/tutorial_qubit_weight.html.
77. Magnard, P. *et al.* Fast and Unconditional All-Microwave Reset of a Superconducting Qubit. *Physical Review Letters* **121**. <https://doi.org/10.1103/physrevlett.121.060502> (Aug. 2018).
78. Barrett, C. N. *et al.* Learning-based Calibration of Flux Crosstalk in Transmon Qubit Arrays 2023. <https://arxiv.org/abs/2303.03347>.

79. Harrington, P., Monroe, J. & Murch, K. Quantum Zeno Effects from Measurement Controlled Qubit-Bath Interactions. *Physical Review Letters* **118**. <https://doi.org/10.1103/physrevlett.118.240401> (June 2017).
80. Itano, W. M. Perspectives on the quantum Zeno paradox. *Journal of Physics: Conference Series* **196**, 012018. <https://doi.org/10.1088/1742-6596/196/1/012018> (Nov. 2009).
81. *Quantum State Tomography* 2004. http://research.physics.illinois.edu/QI/Photonics/tomography-files/tomo_chapter_2004.pdf.
82. DiCarlo, L. *et al.* Demonstration of two-qubit algorithms with a superconducting quantum processor. *Nature* **460**, 240–244. <https://doi.org/10.1038/nature08121> (June 2009).
83. McKay, D. C., Wood, C. J., Sheldon, S., Chow, J. M. & Gambetta, J. M. Efficient gates for quantum computing. *Physical Review A* **96**. <https://doi.org/10.1103/physreva.96.022330> (Aug. 2017).
84. Silveri, M. P. *et al.* Stückelberg interference in a superconducting qubit under periodic latching modulation. *New Journal of Physics* **17**, 043058. <https://doi.org/10.1088/1367-2630/17/4/043058> (Apr. 2015).
85. Sun, X.-M., Cong, L., Eckle, H.-P., Ying, Z.-J. & Luo, H.-G. Application of the polaron picture in the two-qubit quantum Rabi model. *Physical Review A* **101**. <https://doi.org/10.1103/physreva.101.063832> (June 2020).
86. Gu, X. *et al.* Fast Multiqubit Gates through Simultaneous Two-Qubit Gates. *PRX Quantum* **2**. <https://doi.org/10.1103/prxquantum.2.040348> (Dec. 2021).
87. Rol, M. A. *et al.* Time-domain characterization and correction of on-chip distortion of control pulses in a quantum processor. *Applied Physics Letters* **116**. <https://doi.org/10.1063/1.5133894> (Feb. 2020).
88. Ferreira, V. S., Kim, G., Butler, A., Pichler, H. & Painter, O. *Deterministic Generation of Multidimensional Photonic Cluster States with a Single Quantum Emitter* 2022.
89. Ball, R. D. *et al.* The path to proton structure at 1% accuracy. *The European Physical Journal C* **82**. <https://doi.org/10.1140/epjc/s10052-022-10328-7> (May 2022).
90. Pérez-Salinas, A., Cervera-Lierta, A., Gil-Fuster, E. & Latorre, J. I. Data re-uploading for a universal quantum classifier. *Quantum* **4**, 226. <https://doi.org/10.22331/q-2020-02-06-226> (Feb. 2020).
91. Kingma, D. P. & Ba, J. *Adam: A Method for Stochastic Optimization* 2014. <https://arxiv.org/abs/1412.6980>.
92. Magniez, A., Bardou, L., Morris, T. J. & O'Brien, K. *MKID: an energy sensitive superconducting detector for the next generation of XAO systems* in *Adaptive Optics Systems VIII* (eds Schmidt, D., Schreiber, L. & Vernet, E.) (SPIE, Aug. 2022). <https://doi.org/10.1117/12.2627026>.

93. Dixit, A. V. *et al.* Searching for Dark Matter with a Superconducting Qubit. *Physical Review Letters* **126**. <https://doi.org/10.1103/physrevlett.126.141302> (Apr. 2021).