



Design of a control and readout system for quantum computing based on superconducting qubits

PROYECTO

presentado para optar
al Título de Grado en Ingeniería en Sistemas de
Telecomunicación por

Aleksei Nerushenko Savitskaia

bajo la supervisión de

Héctor Solar Ruiz

Donostia-San Sebastián, agosto de 2023



Tecnun
Universidad
de Navarra

ESCUELA DE INGENIERÍA
INGENIARITZA ESKOLA
SCHOOL OF ENGINEERING

CONTENTS

| | |
|--|----|
| INDEX OF FIGURES | 3 |
| LIST OF TABLES | 7 |
| 1 ABSTRACT | 9 |
| 2 INTRODUCTION | 11 |
| 2.1 What are quantum bits..... | 11 |
| 2.2 Introduction to the transmon..... | 12 |
| 2.3 Control and readout of a transmon qubit..... | 16 |
| 3 OBJECTIVES | 19 |
| 4 DESIGN & IMPLEMENTATION | 21 |
| 4.1 Qubit control for X,Y gates..... | 21 |
| 4.2 Qubit readout..... | 24 |
| 4.2.1 Steady-state readout..... | 25 |
| 4.2.2 Transient readout..... | 31 |
| 5 RESULTS | 35 |
| 5.1 Qubit control testbench..... | 35 |
| 5.2 Qubit steady-state readout testbench..... | 37 |
| 5.2.1 Simulations when $\omega_g = \omega_o^{ 0\rangle}$ | 37 |
| 5.2.2 Simulations when $\omega_g = \omega_o^{ 1\rangle}$ | 40 |
| 5.2.3 Simulations when $\omega_g = \omega_o$ | 42 |
| 5.3 Qubit transient state readout..... | 44 |
| 5.3.1 Simulations when $\omega_g = \omega_o^{ 0\rangle}$ | 46 |
| 5.3.2 Simulations when $\omega_g = \omega_o$ | 50 |
| 5.3.3 Simulations when $\omega_g = \omega_o^{ 1\rangle}$ | 54 |
| 5.3.4 Time delay between states..... | 57 |
| 6 CONCLUSIONS AND FUTURE WORK | 59 |

| | | |
|----------|--|-----------|
| 6.1 | Conclusions..... | 59 |
| 6.2 | Future work | 60 |
| 7 | BUDGET | 61 |
| 8 | REFERENCES | 63 |
| 9 | APPENDIXES | 65 |
| A. | Qubit control code | 65 |
| B. | Qubit steady-state readout code | 66 |
| C. | Qubit transient state readout code | 71 |
| D. | Gaussian envelope code..... | 75 |
| E. | Noise source code..... | 76 |

INDEX OF FIGURES

| | |
|--|----|
| Figure 1: Bloch sphere representation of a qubit [2]. | 11 |
| Figure 2: Drawn representation of a physical transmon qubit. | 12 |
| Figure 3: Microscopic view of a Josephson junction. | 13 |
| Figure 4: (a) Circuit model of a parallel LC oscillator (quantum harmonic oscillator, QHO), (b) energy levels of the QHO equidistantly spaced apart, (c) Josephson qubit circuit model with the non-linear inductance represented by a dashed orange box and (d) energy levels of the QHO when using a Josephson junction (blue) compared to an inductance (dashed red) [5]. | 14 |
| Figure 5: Circuit schematics of an LC oscillator (a) and a superconducting qubit (b) [3]. | 14 |
| Figure 6: Circuit schematics for the evolution of a Cooper pair box to a transmon. The Josephson junction is represented by “X” in (a) and C_j being the small junction capacitance. This pair is put together into one symbol represented as a box with an “X” through it like in (b) and (c) [4]. | 15 |
| Figure 7: Microscopic zoom of a typical transmon device. Microwave pulses are applied to the transmission lines to operate and read the qubit state [4]. | 15 |
| Figure 8: (a) shows the SQUID shunted by a large capacitance (“+” shape), (b) shows a zoom in of the SQUID and (c) does so with one of the Josephson junctions of the SQUID [4]. | 16 |
| Figure 9: Circuit schematic of the control and readout of a transmon qubit, including XY drive, Z drive and readout ports [13]. | 16 |
| Figure 10: Schematic of a typical qubit drive setup. | 18 |
| Figure 11: Schematic of an experimental setup for dispersive qubit readout. | 18 |
| Figure 12: Circuit schematic of the transmon coupled to the resonator | 21 |
| Figure 13: Circuit schematic for the control of the transmon qubit. | 22 |
| Figure 14: Circuit diagram of the capacitive coupling of a microwave drive line to a generic transmon [5]. | 23 |
| Figure 15: Flow diagram of the control testbench. | 23 |
| Figure 16: Feedline transmission as a function of frequency when the qubit is in the ground (blue, right) or excited (red, left) state. Along with the bare resonator frequency response (green, dotted) [10]. | 24 |
| Figure 17: Simplified circuit diagram of the readout. The transmon is capacitively coupled to the resonator and this one to the readout transmission line. The readout pulse is driven by a voltage source through the transmission line. | 24 |
| Figure 18: RLC circuit side-coupled to a transmission line [9] | 26 |
| Figure 19: Circuit schematic for the steady-state readout of the transmon qubit. | 29 |

| | |
|--|----|
| Figure 20: Flow diagram of the steady-state readout testbench. | 30 |
| Figure 21: Simplified circuit schematic of the transient state readout block | 31 |
| Figure 22: Circuit schematic for the transient state readout of the transmon qubit..... | 33 |
| Figure 23: Flow diagram of the transient state readout testbench. | 34 |
| Figure 24: Gaussian pulse obtained after mixing a sinusoidal voltage source with a Gaussian envelope of duration 600 ns and normalized amplitude to 1 V. | 35 |
| Figure 25: Accumulated rotation angle as a function of time | 36 |
| Figure 26: Probabilities of finding the qubit in state $ 0\rangle$ (blue curve) or state $ 1\rangle$ (orange curve) depending on the amplitude used for the control pulse with the qubit starting in state $ 0\rangle$ | 36 |
| Figure 27: Noiseless output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 38 |
| Figure 28: Noisy output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 38 |
| Figure 29: Noiseless output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 1\rangle}$ | 39 |
| Figure 30: Noisy output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 1\rangle}$ | 39 |
| Figure 31: Noiseless output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 1\rangle}$ | 40 |
| Figure 32: Noisy output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 1\rangle}$ | 40 |
| Figure 33: Noiseless output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 1\rangle}$ | 41 |
| Figure 34: Noisy output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 1\rangle}$ | 41 |
| Figure 35: Noiseless output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is ω_o | 42 |
| Figure 36: Noisy output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is ω_o | 42 |
| Figure 37: Noiseless output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is ω_o | 43 |
| Figure 38: Noisy output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is ω_o | 43 |

| | |
|--|----|
| Figure 39: Noiseless output V_o when the angular frequency of the voltage source is ω_o and the qubit is in state $ 1\rangle$ (blue, left) and state $ 0\rangle$ (orange, right) | 44 |
| Figure 40: S_{21} parameters of the bare resonator (blue, middle), resonator coupled to the qubit in state $ 1\rangle$ (grey, left) and resonator coupled to the qubit in state $ 0\rangle$ (orange, right). | 45 |
| Figure 41: Noiseless output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 46 |
| Figure 42: Noisy output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 46 |
| Figure 43: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 47 |
| Figure 44: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 and 601 ns when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 47 |
| Figure 45: Noiseless output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 48 |
| Figure 46: Noisy output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 48 |
| Figure 47: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 49 |
| Figure 48: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 ns and 601 ns when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{ 0\rangle}$ | 49 |
| Figure 49: Noiseless output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is ω_o | 50 |
| Figure 50: Noisy output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is ω_o | 50 |
| Figure 51: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is ω_o | 51 |
| Figure 52: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 and 601 ns when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is ω_o | 51 |
| Figure 53: Noiseless output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is ω_o | 52 |

| | |
|---|----|
| Figure 54: Noisy output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is ω_o | 52 |
| Figure 55: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is ω_o | 53 |
| Figure 56: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 ns and 601 ns when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is ω_o | 53 |
| Figure 57: Noiseless output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 54 |
| Figure 58: Noisy output V_o when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 54 |
| Figure 59: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 55 |
| Figure 60: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 and 601 ns when the qubit is in state $ 1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 55 |
| Figure 61: Noiseless output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 56 |
| Figure 62: Noisy output V_o when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 56 |
| Figure 63: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 57 |
| Figure 64: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 ns and 601 ns when the qubit is in state $ 0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$ | 57 |
| Figure 65: Time delay between the probing signals when the state of the qubit is changed while maintaining the angular frequency of the voltage source: $\omega_o^{(1)}$ (blue curve), $\omega_o^{(0)}$ (grey curve) and ω_o (orange curve)..... | 58 |

LIST OF TABLES

| | |
|---|----|
| Table 1: Summary of editable parameters of the steady-state testbench with the variable names used in the code inside round brackets. | 25 |
| Table 2: Summary of editable parameters of the transient state testbench with the variable names used in the code inside round brackets. | 32 |
| Table 3: Hardware costs..... | 61 |
| Table 4: Software costs | 61 |
| Table 5: Human resources | 61 |
| Table 6: Total costs | 61 |

1 ABSTRACT

This thesis will focus on the development of testbenches in Cadence Virtuoso for the verification of the readout and control of analog electronics involving transmon qubits. Taking the parameters of an existing qubit upon which the simulations will be compared to, the system will provide the resulting signals that are involved in the processes of control and readout.

Custom analog models will be programmed in order to implement all the arithmetic calculations needed to run the simulations. These programs can be separated into control and readout and they will be written using Verilog-A as the programming language. For the control of the qubit two different codes will be implemented, one of them will generate Gaussian pulses and the other one will perform the relevant operations. For the readout a distinction will be made between transient and steady-state readout.

The steady-state readout will consist of a single code that will output relevant values and signals based on the qubit parameters that the user inputs. The transient readout will study how the system operates in the first microseconds of the readout process and simulate how the resonator charges up before the system stabilizes. Both readout systems will simulate noisy and noiseless outputs.

After validating the proper functionality of the systems the testbenches will be available for other users to validate the analog electronics designed for the readout and control of a transmon-type qubit.

2 INTRODUCTION

2.1 What are quantum bits

Quantum bits (qubits) are, as the name implies, the quantum counterpart of classical bits and work as the basic unit of information in quantum computing. Classical bits can take the value 0 or 1 whereas qubits can represent a linear combination of both states due to a phenomenon called superposition. Qubit states are defined using notation called *Dirac notation*, the classic state 0 and 1 would be represented as $|0\rangle$ and $|1\rangle$ respectively and any other state the qubit is in could be represented as a linear combination of these two states [2].

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

Measuring a qubit collapses it yielding the state 0 with probability $|\alpha|^2$ or the state 1 with probability $|\beta|^2$. Since the sum of probabilities must equal 1 Equation (1) can be rewritten as [2].

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\gamma}\sin\frac{\theta}{2}|1\rangle$$

This linear combination can be represented as a point in the surface of a three-dimensional sphere, called the Bloch sphere, which is shown in Figure 1

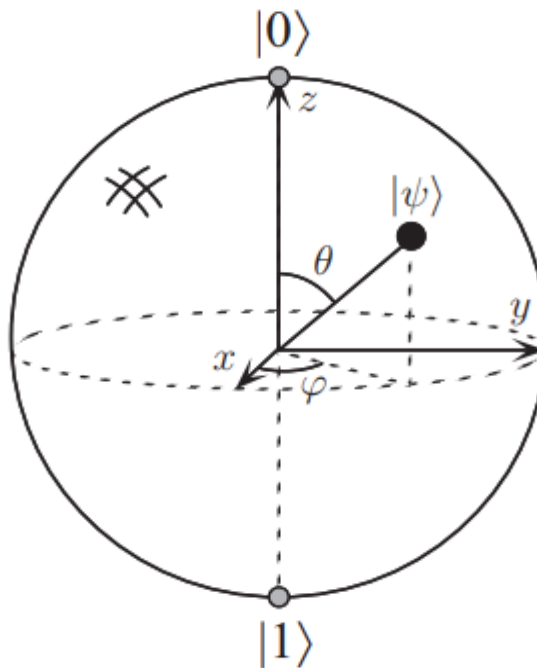


Figure 1: Bloch sphere representation of a qubit [2].

Thanks to the qubit it is possible to create quantum processors for quantum computing that reduce exponentially the time spent to compute certain algorithms.

Building, maintaining and scaling quantum systems are significant challenges despite all the benefits that quantum properties bring. Issues like short decoherence times mean that quantum processing must be done in temperatures as cold as milikelvins, making the electronics needed for these systems to be precise and fast. In the race to build a quantum computer, companies are pursuing many types of quantum bits each with its own strengths and weaknesses. Diamond vacancies, topological qubits, silicon quantum dots, trapped ions or superconducting loops are some of the most developed kinds of qubits [11]. This project will focus its attention on the latter type of qubits, more specifically on the transmon qubit. This type of qubit has been chosen because some of the most advanced quantum computers (Google, IBM) are built using this technology.

2.2 Introduction to the transmon

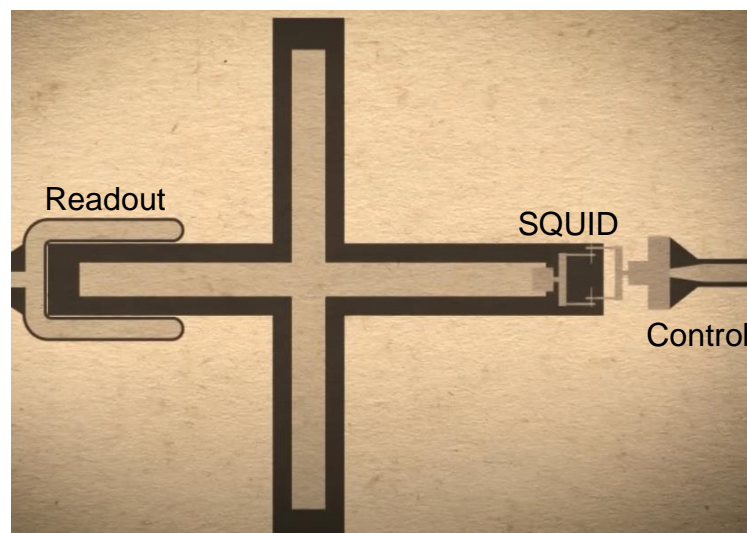


Figure 2: Drawn representation of a physical transmon qubit.

There are several types of superconducting qubits like the charge qubit, the flux qubit or the phase qubit to name a few of them. They have two important differences when compared to quantum two-level systems found in nature. First, they are multi-level systems and in order to be used effectively as qubits it is needed to confine all dynamics to two quantum levels, usually the ground state and the first excited state [3].

The second difference is that these qubits are manmade and this characteristic brings advantages and disadvantages. There is the freedom of designing the qubits as needed. However, this also implies that uncertainties in fabrication will make it impossible to make any two qubits the same. The most important benefit of this type of qubit is that it comes in a chip which can be interacted with using microwave equipment [3].

Superconducting qubits consist of superconducting electrodes, or islands, – which are materials that drop their resistance to near 0 at very low temperatures – connected through Josephson junctions, which can be viewed as nonlinear inductors. A Josephson junction is built by laying a thin layer of nonsuperconducting material between two layers of superconducting material. Metal alloys suffer a transition between its conducting state to a superconducting state when they are cooled to temperatures within twenty degrees or less of zero kelvin. In this state the material loses any electrical resistance, electrons lose their repulsive interaction and become slightly attractive forming pairs. If the middle layer of the Josephson junction is thin enough (in the order of angstroms for insulators and microns for nonsuperconducting metals) these electron pairs can tunnel across the barrier without any resistance until a critical current is reached [12]. Figure 3 shows a zoomed microscopic view of a Josephson junction.

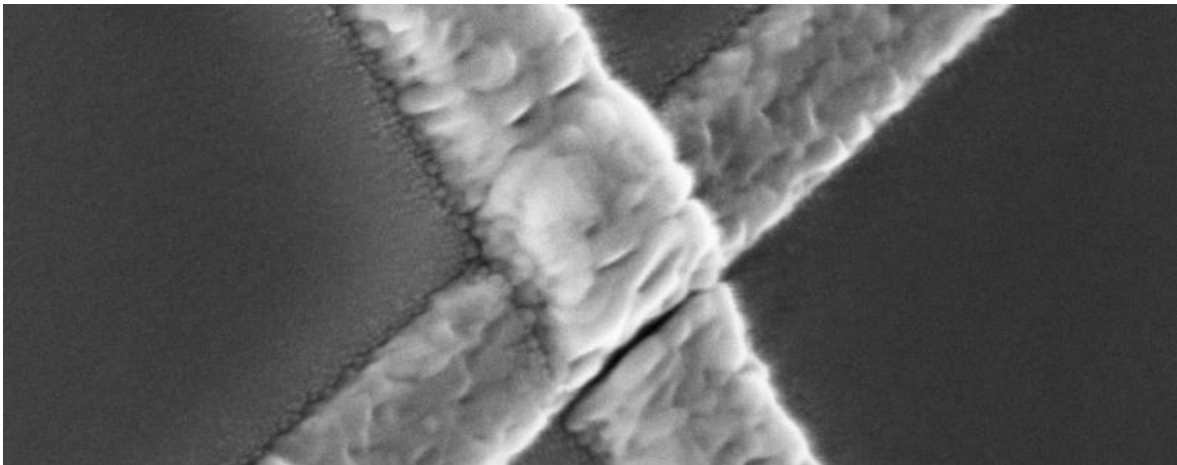


Figure 3: Microscopic view of a Josephson junction.

The transmon belongs to the family of charge superconducting qubits and its design holds similarities to an LC oscillator where the inductance has been replaced by a Josephson junction. This change is needed to properly confine the quantum dynamics to the two lowest energy levels of the system, the harmonicity of an LC oscillator would make it difficult to confine the dynamics because all the energy levels are equally separated in energy, so the Josephson junction is introduced to break this harmonicity with its nonlinear behavior and separating the energy levels [4]. The harmonicity of an oscillator refers to how the energy levels of the different harmonics of the fundamental frequency are equally separated, which may produce unintentional leakage. This leakage would be produced when the system is unintentionally transitioned to a higher energy level. The Josephson junctions solve this problem as can be appreciated in Figure 4.

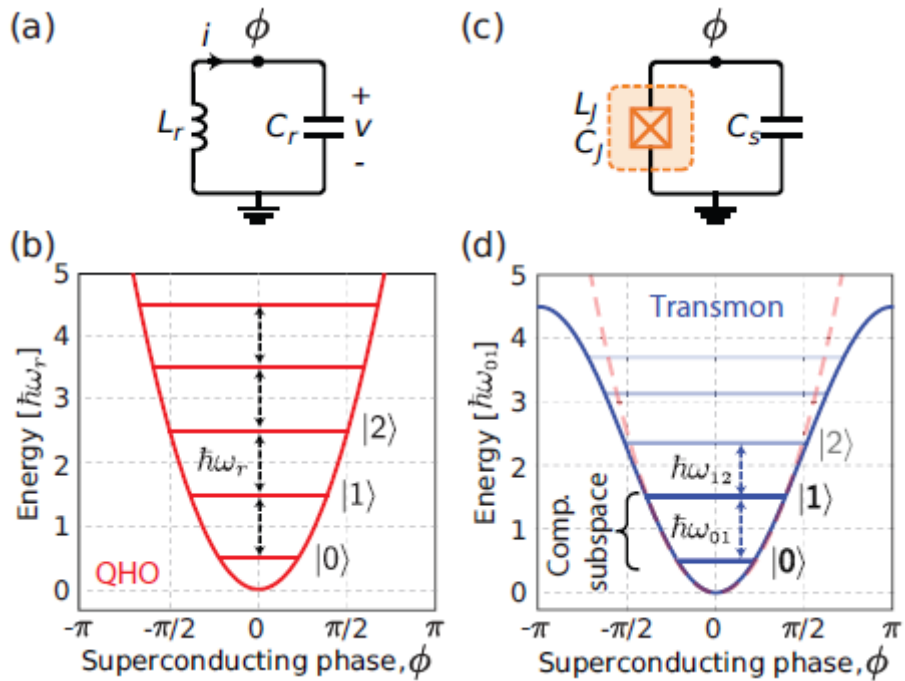


Figure 4: (a) Circuit model of a parallel LC oscillator (quantum harmonic oscillator, QHO), (b) energy levels of the QHO equidistantly spaced apart, (c) Josephson qubit circuit model with the non-linear inductance represented by a dashed orange box and (d) energy levels of the QHO when using a Josephson junction (blue) compared to an inductance (dashed red) [5].

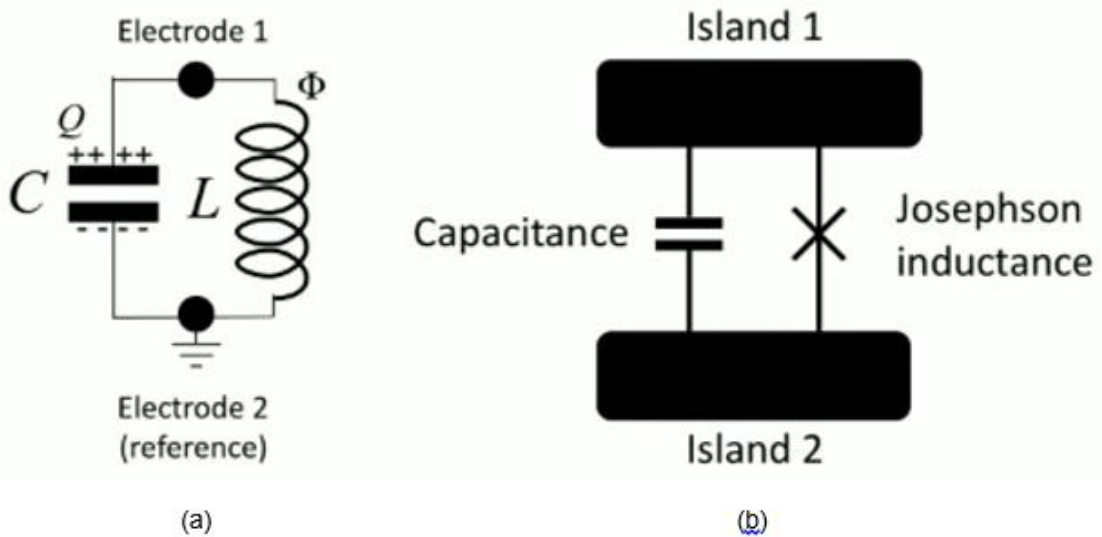


Figure 5: Circuit schematics of an LC oscillator (a) and a superconducting qubit (b) [3].

The transmon has evolved from a circuit with a single Josephson junction to a circuit like shown in Figure 6(c), where the transmon is built by two Josephson junctions shunted by a large capacitance C_B . This parallel combination of Josephson junctions is called a superconducting quantum interference device (SQUID).

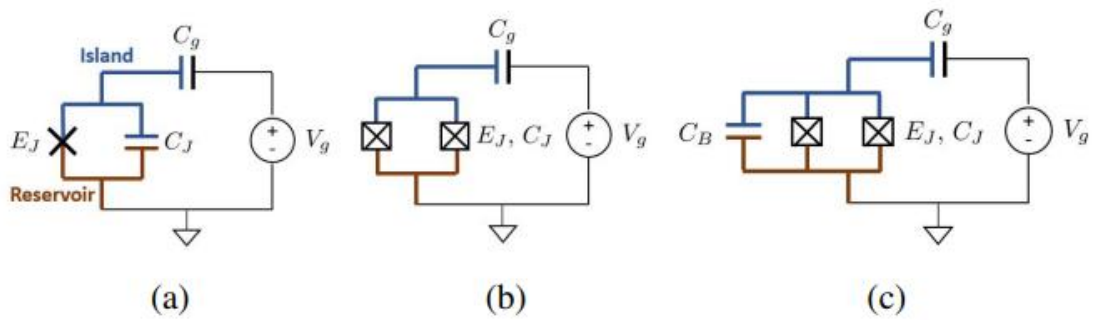


Figure 6: Circuit schematics for the evolution of a Cooper pair box to a transmon. The Josephson junction is represented by "X" in (a) and C_J being the small junction capacitance. This pair is put together into one symbol represented as a box with an "X" through it like in (b) and (c) [4].

In Figure 7 all the components of the circuit can be seen under microscopic zoom, included with the transmon are the $\lambda/4$ readout resonator and the transmission lines for control, readout and flux bias. Figure 8 shows close up images of the transmon qubit itself and its circuit model.

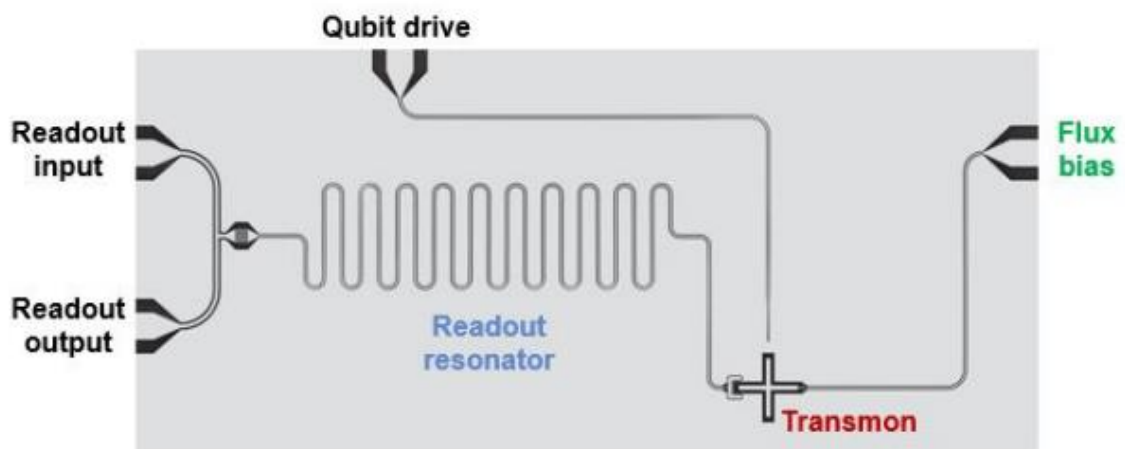


Figure 7: Microscopic zoom of a typical transmon device. Microwave pulses are applied to the transmission lines to operate and read the qubit state [4].

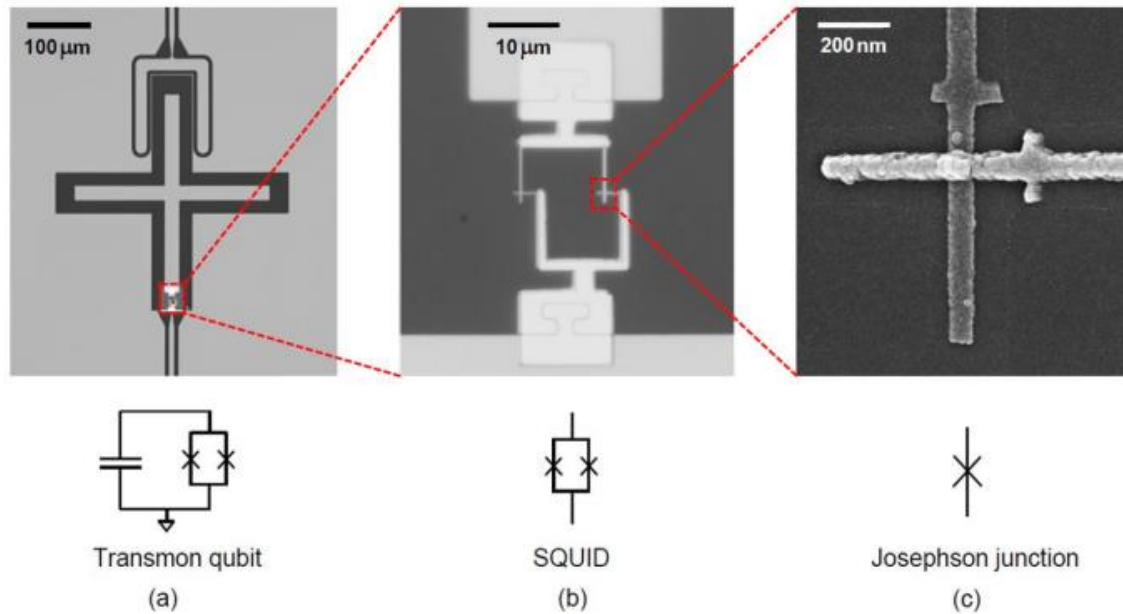


Figure 8: (a) shows the SQUID shunted by a large capacitance ("+" shape), (b) shows a zoom in of the SQUID and (c) does so with one of the Josephson junctions of the SQUID [4].

2.3 Control and readout of a transmon qubit

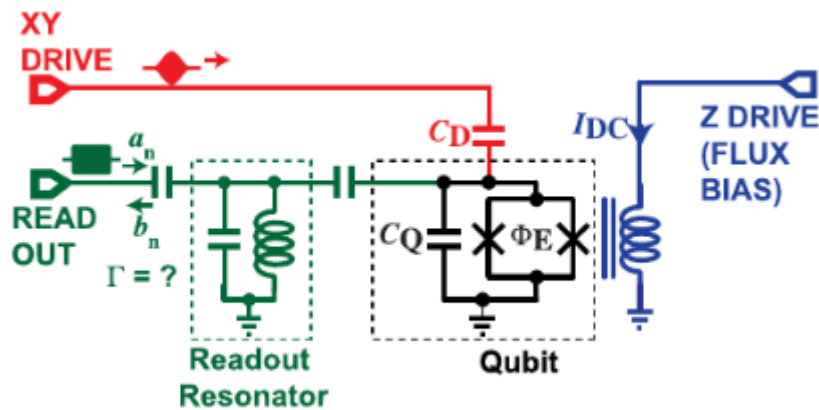


Figure 9: Circuit schematic of the control and readout of a transmon qubit, including XY drive, Z drive and readout ports [13].

This section will provide a brief explanation on how the control and readout of transmon qubits is performed before diving into details further on. The measurement of the qubit can be done using a resonator coupled to it. The resonator must be designed in such a way that its resonance frequency is far away from the qubit transition frequency in order to avoid interfering with the state of the qubit. The energy difference between two quantum states is the transition frequency, usually measured in Hertz and proportional to the gap between states. Transmon qubits have transition frequencies in the range of microwaves. Due to the coupling present between the resonator and the qubit, there will be a shift in the resonator's

frequency depending on the qubit state on the order of, typically, a few megahertz. This type of readout system is called the dispersive shift method.

To observe this shift a microwave pulse near the resonator frequency is injected into the resonator through a transmission line coupled to the resonator. This pulse is reflected by the resonator and by observing this response it is possible to infer in which state the qubit is in. The pulse needed for the readout must be shorter, in time, than the decoherence time of the qubit to minimize the effects of decoherence and improve measurement accuracy. Decoherence times in transmon qubits are usually in the range of microseconds while readout pulses tend to be in the range of hundreds of nanoseconds. The dispersive shift readout method is non-destructive meaning that the state of the qubit does not get altered after the readout process finishes, making it possible to apply quantum error correction protocols.

Similarly to the readout, the control of the qubit is performed by sending microwave pulses through a transmission line coupled to the qubit. Depending on the shape, duration and amplitude of these pulses it is possible to manipulate the energy levels of the transmon and perform different operations that produce rotations of the qubit state around the Bloch sphere. These rotations around the Z, X and Y axis are called quantum gates which are the counterpart of classic logical gates, some examples of universal quantum gates are the NOT gate or the Hadamard gate but analyzing quantum gates falls outside the scope of this project. Two different transmission lines are used to perform rotations, one of them is used for rotations along the X and Y axes and the other transmission line performs rotations along the Z axis. XY gate operations are in charge of performing changes in the state of the qubit by driving microwave pulses through the coupled transmission line while Z gate operations allow controlling the frequency of the qubit through an external DC current. This provides control of the qubit frequency and qubit initialization.

Although the development of electronics for the control and readout of a transmon qubit falls outside the scope of this project, schematics of such setups are provided for the control (Figure 10) and for the readout (Figure 11). It is worth mentioning that pursuing these designs could be future work after this project has concluded.

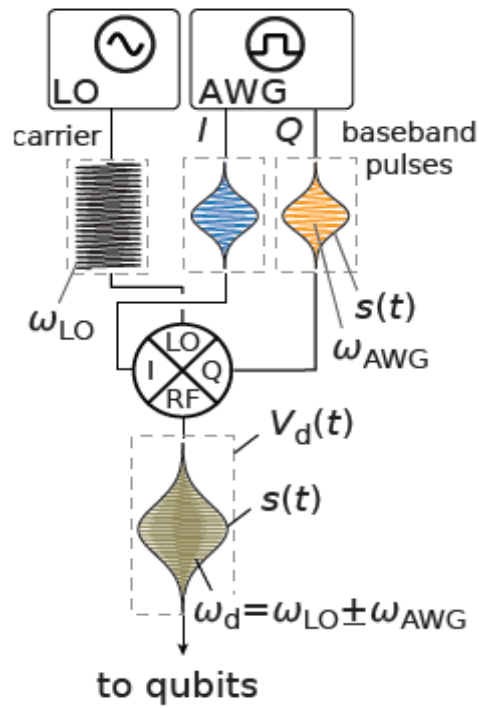


Figure 10: Schematic of a typical qubit drive setup.

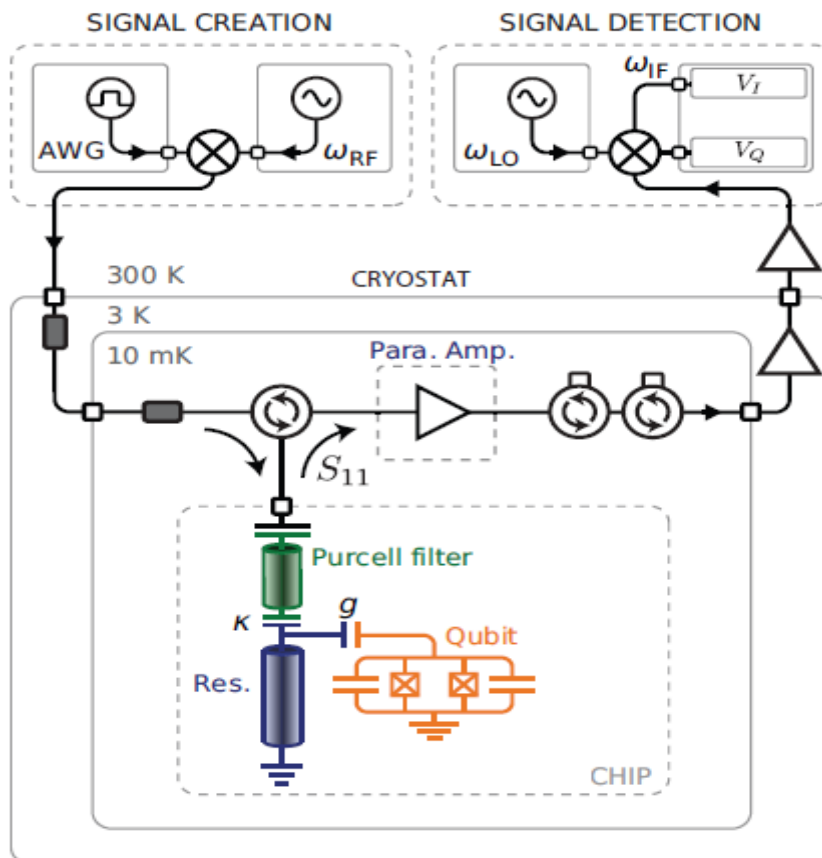


Figure 11: Schematic of an experimental setup for dispersive qubit readout.

3 OBJECTIVES

The main objective of this project is **the development of testbenches in Cadence Virtuoso for the verification of the readout and control analog electronics involving transmon qubits.**

To reach the main goal of the project a series of tasks will be set up that will work as milestones that allow keeping track of the progress made. Progress on the readout and control processes will be done separately because of the independence between them.

- **Choosing a viable programming language:** Because this project involves building custom analog blocks the first task is choosing a programming language compatible with Cadence Virtuoso in order to design and verify the functionality of the testbenches.
- **Find transmon parameters:** Once the programming language has been chosen the next step will be to find parameters of a real transmon qubit upon which the simulations carried out by the testbenches can be compared to. The results obtained from the testbenches must coincide with the results obtained from the measurements of the real qubit in order to confirm the viability of the testbenches
- **Design and code the analog blocks:** In order to design the testbenches several analog blocks must be implemented, this implementation process will consist of deciding which inputs and outputs the blocks will have, what internal parameters the user can modify by editing the blocks' properties in the schematics and proper coding of the arithmetic logic that the blocks will execute. The design of these testbenches will be done while trying to make them user-friendly for other users and third parties.
- **Emulate the blocks and study the results:** Finally, a set of simulations will be run varying parameters like the state of the qubit and the frequency of the probing pulse for the readout of the qubit or the amplitude and duration of driving pulses for the control of the qubit. These simulations will be displayed in this work and then compared to the expected results.

4 DESIGN & IMPLEMENTATION

The decision of which programming language is going to be used in this project is between Verilog-A and VHDL-AMS. Both languages hold a similar position in terms of its scope and application to synthesizable design but the best fit for the project will be Verilog-A. The reasons that justify this preference are that, on one hand, this language provides good support for simple and moderately complex chips and on the other hand Cadence Virtuoso has built-in capabilities that allow convenient and fast workflow between schematics and Verilog-A models [1].

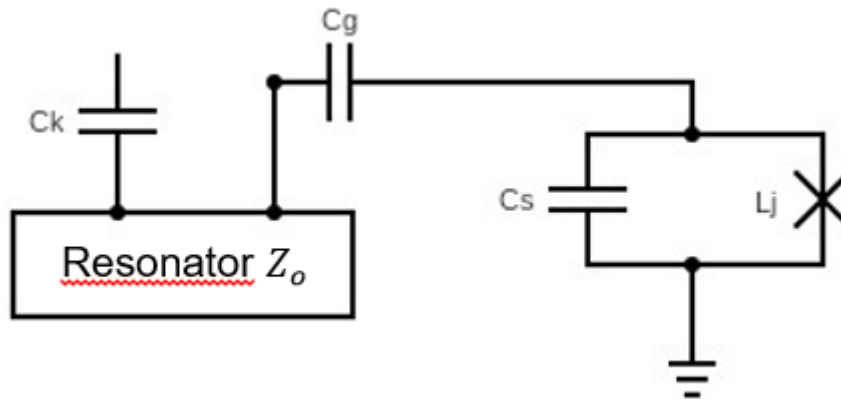


Figure 12: Circuit schematic of the transmon coupled to the resonator

The qubit chosen to test the proper functioning of the testbenches is taken from a real implementation and has a shunt capacitance of $C_s = 98.19$ fF. $C_g = 4.40$ fF and $C_k = 8.62$ fF for the qubit-resonator and resonator-feedline coupling capacitances respectively. The value for the Josephson junction inductance is $L_j = 11$ nH. The readout resonator will have a matching characteristic impedance of $Z_o = 50 \Omega$ and is 5.76 mm long [10]. Results obtained from the simulations will be provided in section 5 to verify these designs.

4.1 Qubit control for X,Y gates

In this section an analog block is going to be built to emulate how the qubit changes its state. Specifically, the control block outputs the angle of rotation of the qubit. Once this angle is obtained it is possible to calculate the probability of the qubit to be in state $|1\rangle$ or $|0\rangle$ as $\sin^2(\text{angle}/2)$ or $\cos^2(\text{angle}/2)$, respectively. To change the state of the qubit microwave pulses are sent through the transmission line and by performing parametric sweeps of either the amplitude of the envelope or the duration of the drive pulses it is possible to obtain a curve which depicts how the probability of the qubit to be in either state oscillates between 0 and 1. The results obtained determine the amplitude or duration required to perform rotations of the qubit in the Bloch sphere along the X or Y axis at arbitrary angles [6].

There is a degree of freedom that allows different waveforms to be used in these pulses like Rabi, Rosen-Zener, Gaussian, Demkov or sech^2 [7]. This project will use the Gaussian envelope in Cadence Virtuoso. The user can substitute the Gaussian envelope for other ones and the amplitude of the waveform must be normalized to 1 V for the testbench to operate properly, the desired amplitude of the drive pulse can be set through the parameter 'V_o' of the control block which will serve as a scaling factor. The code for the Verilog-A model creating the Gaussian envelope is available in Appendix D. Choosing different envelope shapes depends on factors like the characteristics of the qubit, the noise sources and desired trade-offs between fidelity and speed. The Gaussian envelope has been chosen due to it being an often used pulse in signal processing.

The circuit will consist of two Verilog-A blocks, the first one will output the Gaussian envelope which will then be mixed with a sinusoidal voltage source. The modulated signal will then serve as the input for the main Verilog-A block, called control block from now on. This control block will output the angle of rotation. An ideal sinusoidal voltage source and a mixer will also be used.

Figure 13 depicts the circuit schematic, where 'pulseGaussian' is the Verilog-A block for the Gaussian envelope, 'Qubit_Control' is the control block, 'V1' is the sinusoidal voltage source and 'E0'. Both 'V1' and 'E0' are components available in Cadence in the library *analogLib*.

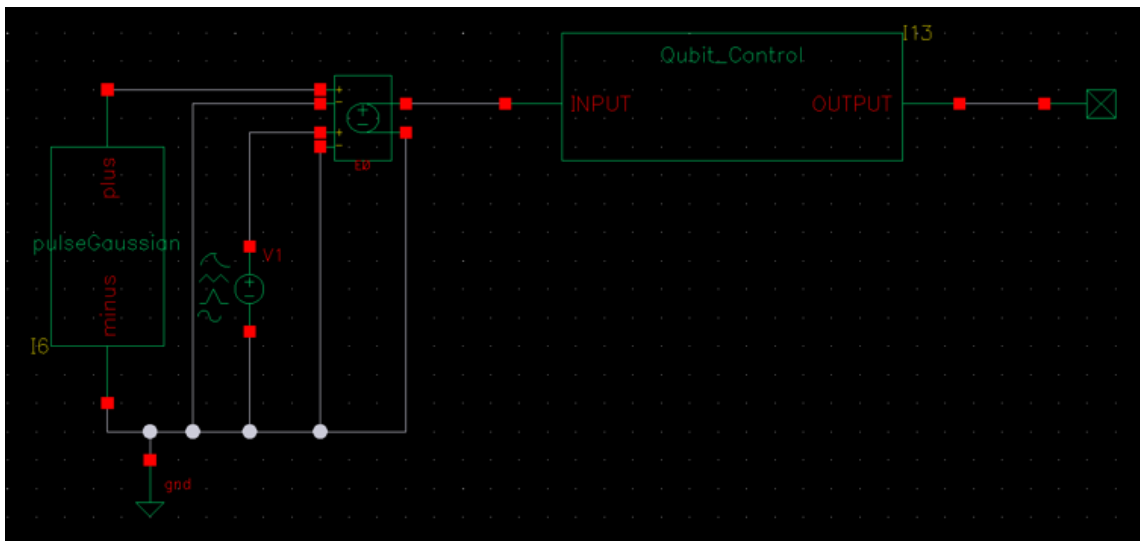


Figure 13: Circuit schematic for the control of the transmon qubit.

The angle by which a state is rotated given the capacitive couplings, the impedance of the circuit, the magnitude V_o of the carrier, and the waveform envelope $s(t)$ is given by

$$\theta(t) = -\Omega V_o \int_0^t s(t') dt'$$

In order to obtain this equation Ref. 5 has been followed because solving the Hamiltonian of a transmon falls outside the scope of this project. The parameter Ω , drive coupling strength, can be calculated as [5]:

$$\Omega = (C_d/C_\Sigma)Q_{zpf}$$

where $Q_{zpf} = \sqrt{\hbar/2Z}$ is the zero-point charge fluctuations and $Z = \sqrt{L/C}$ is the qubit impedance. $C_\Sigma = C + C_d$ is the total capacitance to ground. Figure 14 shows a circuit diagram of how a generic transmon couples to a drive line [5].

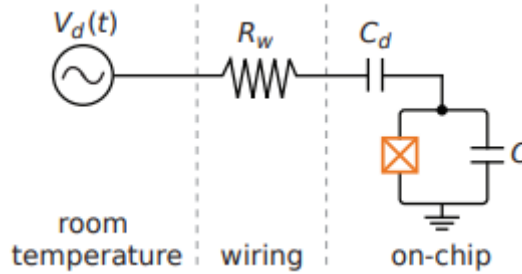


Figure 14: Circuit diagram of the capacitive coupling of a microwave drive line to a generic transmon [5].

Figure 15 illustrates the main functions performed by the control block as well as how the user shall operate the testbench.

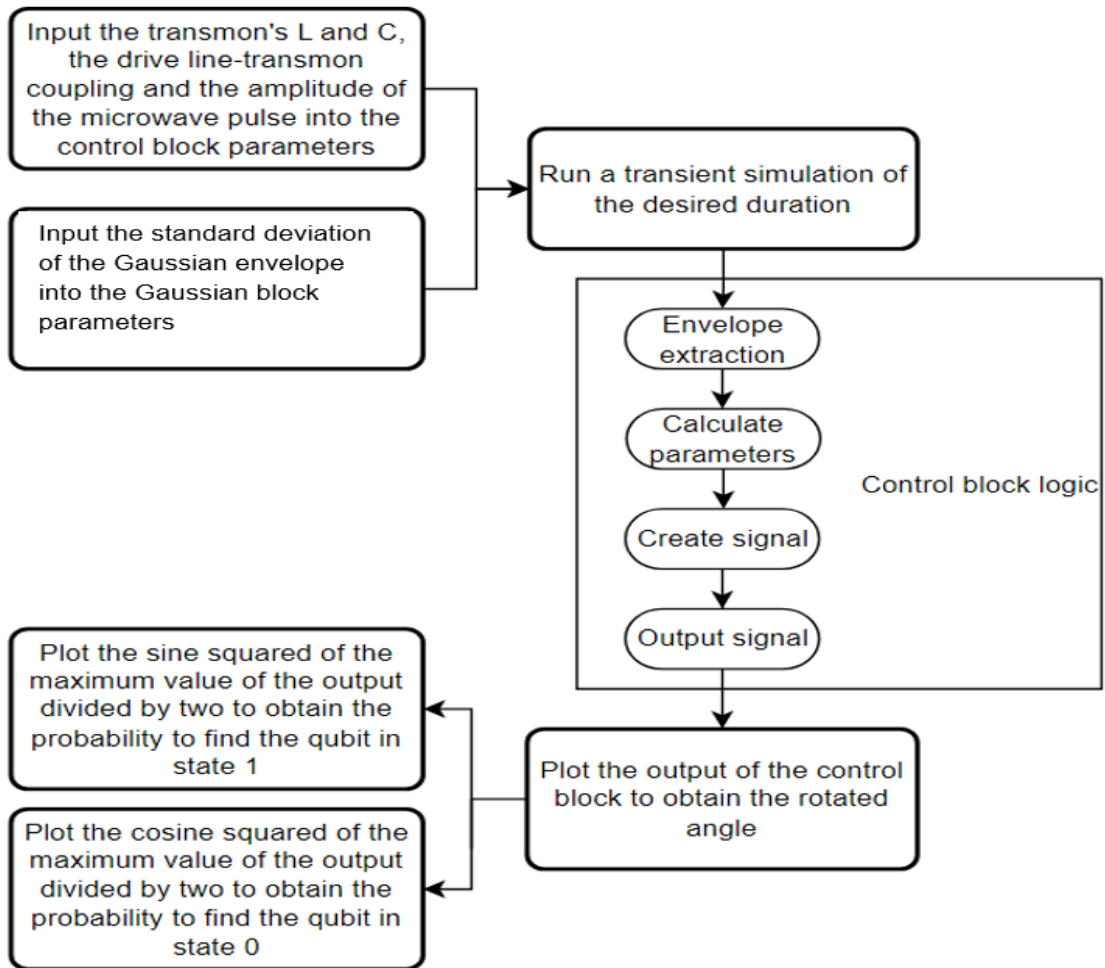


Figure 15: Flow diagram of the control testbench

The control block will extract the envelope of the input in order to allow for different waveforms other than the Gaussian envelope. Once the envelope has been obtained, the values of Z , Q_{zpf} , C_Σ and Ω are calculated. Finally, the signal $\theta(t)$ is generated and assigned as the output of the block. The code used for the Verilog-A model of the control block can be read in Appendix A. The results obtained from this simulation are provided in section 5.1.

4.2 Qubit readout

One of the most used readout techniques is the dispersive readout. The qubit, which is detuned from the resonator's resonant frequency, produces a state-dependent frequency shift of the resonant peak of the resonator due to the coupling between them. Since the frequency of the qubit changes depending on the state this shift is different for either state. Figure 16 illustrates how the resonant peak of the resonator shifts. The output voltage V_o obtained at the end of the readout process will be different depending on the qubit state.

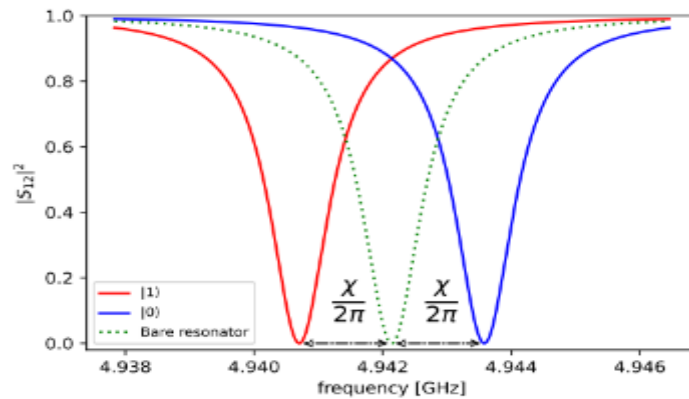


Figure 16: Feedline transmission as a function of frequency when the qubit is in the ground (blue, right) or excited (red, left) state. Along with the bare resonator frequency response (green, dotted) [10].

To emulate the transmon's behavior during readout two different circuits will be designed. The first one will analyze the readout in the steady-state and the second one in the transient state.

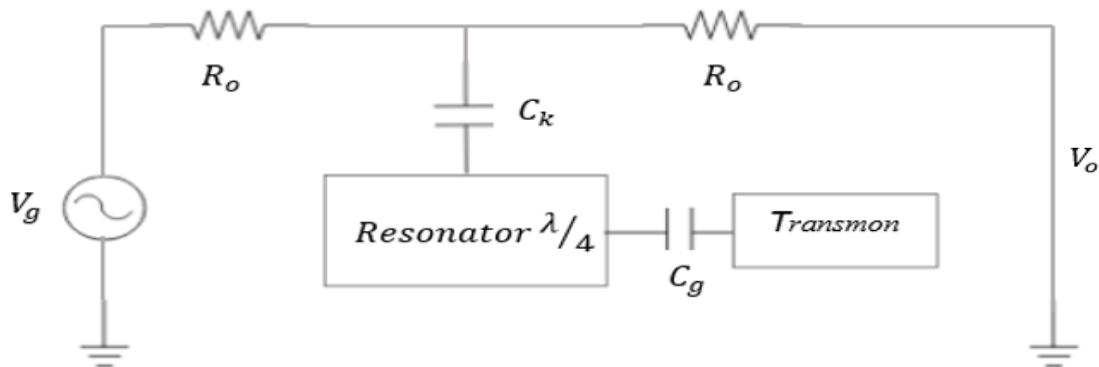


Figure 17: Simplified circuit diagram of the readout. The transmon is capacitively coupled to the resonator and this one to the readout transmission line. The readout pulse is driven by a voltage source through the transmission line.

4.2.1 Steady-state readout

The design process will begin by deciding which parameters the user can input into the testbench. These parameters have been summarized in Table 1. Under ‘*Other parameters*’ are included the output impedance of the signal generator, the input impedance of the detector, the characteristic impedance of the transmission lines, the noise figure of the amplification chain, duration of the readout pulse and the dispersive shift range allows the user to move the frequency of the voltage source between the frequency shifts provoked by the qubit in order to be able to probe the resonator at any desired frequency inside that range.

| Transmon parameters | | Resonator parameters | | Coupling parameters | | Other parameters | |
|----------------------------|-----|---|--------------|----------------------------|-----|--|--------------|
| Inductance (L_j) | [H] | Loss factor (α) | [1/m] | TxLine-Resonator (C_k) | [F] | Resistor's impedance (R_o) | [Ω] |
| | | Length (len) | [mm] | | | Characteristic impedance of TxLine (Z_o) | [Ω] |
| Capacitance (C_s) | [F] | Characteristic impedance (Z_o') | [Ω] | | | Resonator-Transmon (C_g) | [F] |
| Qubit state (Choose_State) | [-] | Angular frequency of resonance (ω_r) | [rad/s] | | | Duration of the pulse (pulseTime) | [s] |
| | | | | | | Dispersive shift range (range) | [-] |

Table 1: Summary of editable parameters of the steady-state testbench with the variable names used in the code inside round brackets.

The circuit will consist of a single component which is the Verilog-A model of the steady-state readout testbench and its outputs, Figure 19 illustrates it. No input signals are needed because all the calculations are made internally using the input parameters set up by the user and the output signal is created inside the model.

The first step the model takes is to calculate the inductance and the capacitance values of the resonator [10]:

$$C_r = \frac{\pi}{4\omega_r Z_o'} \quad L_r = \frac{1}{C_r \omega_r^2}$$

as well as the capacitive and inductive energies of the transmon [8]:

$$E_c = \frac{e^2}{2(C_g + C_s)} \quad E_j = \frac{\hbar^2}{4e^2 L_j}$$

With these energies it is possible to obtain the transmon's transition angular frequencies ω_{ij} . The transition angular frequencies needed to calculate the dispersive shift are the transitions $0 \rightarrow 1$ and $1 \rightarrow 2$ [8]:

$$\omega_{01} = \frac{\sqrt{8E_c E_j} - E_c}{\hbar} \qquad \omega_{12} = \frac{\sqrt{8E_c E_j} - 2E_c}{\hbar}$$

In parallel with the transition angular frequencies the quality factor of the resonator can be calculated. First the resonant angular frequency of the resonator must be recalculated due to the effects of the coupling to the readout transmission line:

$$\omega_o = 1 / \sqrt{L_r (C_r + C_k)}$$

To calculate the quality factor the resonator, the RLC parallel circuit model when side-coupled to a transmission line will be considered, like the one seen in Figure 18. This approach changes the C and R values of the resonator which will be recalculated in order to obtain the quality factor [9]:

$$R = \frac{Z'_o}{\text{length} * \alpha} \qquad R^* = \frac{1 + \frac{\omega_o^2 C_k^2 R_o^2}{4}}{\frac{\omega_o^2 C_k^2 R_o}{2}} \qquad C^* = \frac{C_k}{1 + \frac{\omega_o^2 C_k^2 R_o^2}{4}}$$

$$Q_l = \frac{\omega_o (C^* + C_r)}{\frac{1}{R^*} + \frac{1}{R}}$$

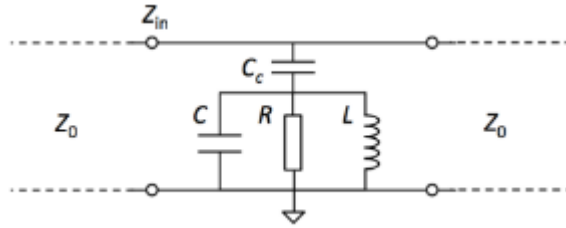


Figure 18: RLC circuit side-coupled to a transmission line [9]

The next step is to calculate the total dispersive shift as [10]:

$$\chi = \chi_{01} - \frac{\chi_{12}}{2} \text{ where } \chi_{ij} = g_{ij}^2 / (\omega_{ij} - \omega_r)$$

The term g_{ij} is known as the qubit-resonator coupling strength and can be calculated as [10]:

$$g_{n,n+1} = \sqrt{n+1} \frac{2\beta e V_{rms}}{\hbar} \left(\frac{E_j}{32E_c} \right)^{\frac{1}{4}}$$

With $\beta = C_g/(C_g + C_s)$, $V_{rms} = \sqrt{\hbar\omega_r/2C_r}$ and e elementary charge. Next, the resonator resonance width is calculated as [8]:

$$k = \frac{\omega_o}{Q_l}$$

The dispersive shift must be larger than the resonance width to allow the readout of the qubit state [10]. The next step is to locate the angular frequencies of the resonator when it has been shifted by the qubit, these are given as $\omega_o^{(0)} = \omega_o - \chi$ when the qubit state is $|0\rangle$ and $\omega_o^{(1)} = \omega_o + \chi$ when the qubit state is $|1\rangle$. At the same time the voltage source's angular frequency is calculated using the parameter 'range' of the model. This parameter can take any value between [-1, 1] and allows the user to set the voltage source angular frequency between $\omega_o^{(0)}$ and $\omega_o^{(1)}$ with ω_g being the angular frequency of the voltage source. Since the steady-state readout does not have inputs this angular frequency is calculated by the block's logic but in the transient state readout the user will need to input it at the voltage source generator:

$$\omega_g = \omega_o - range \cdot \chi$$

Now two angular frequency differences are created between ω_g and $\omega_o^{(1)}$ and between ω_g and $\omega_o^{(0)}$. These are defined as [8] and represent the distance from the resonant angular frequency at which the user is performing the readout to each of the resonant angular frequency peaks shifted by the dispersive shift effect as seen in Figure 16:

$$\Delta\omega^{(1)} = \omega_g - \omega_o^{(1)} \quad \Delta\omega^{(0)} = \omega_g - \omega_o^{(0)}$$

In order to calculate the maximum voltage allowed at the voltage source the critical photon number in the cavity is needed and is given by [8]:

$$n_{crit} = \frac{\Delta^2}{4g^2}$$

Where $\Delta = \omega_o - \omega_{01}$ is the resonator-qubit detuning. This number serves as a threshold which when approached, quantum effects become more prominent. Now it is possible to find the maximum value of V_g allowed at the voltage source [8]. The steady-state readout model will calculate this value for both states and depending on the state chosen by the user one of them will be assigned as the output 'Vgmax' seen in Figure 19:

$$\begin{aligned} |V_g^{(0)}|^2 &= 4\hbar\omega_g k R_o n_{crit} [1 + (2\Delta\omega^{(0)}/k)^2] \\ |V_g^{(1)}|^2 &= 4\hbar\omega_g k R_o n_{crit} [1 + (2\Delta\omega^{(1)}/k)^2] \end{aligned}$$

With these values at hand it is possible now to find the magnitude of the output voltage $|V_o|$ and its phase θ [8] as:

$$|V_o^{(i)}| = |V_g^{(i)}| \left| \frac{\Delta\omega^{(i)}}{k} \right| (1 + (2\Delta\omega^{(i)}/k)^2)^{-1/2} \quad \tan \theta^{(i)} = \frac{k}{2\Delta\omega^{(i)}}$$

As has been done for V_g , the model will differentiate internally between $|V_o^{(0)}|$ and $|V_o^{(1)}|$ and between $\tan \theta^{(0)}$ and $\tan \theta^{(1)}$ by replacing the state-dependent parameters in the equation.

Once these voltages are known it is possible to calculate the S_{21} parameter between the input and output ports of the readout for both states as [8]:

$$S_{21}^{(0)} = \frac{2 |V_o^{(0)}|^2}{|V_g|} \quad S_{21}^{(1)} = \frac{2 |V_o^{(1)}|^2}{|V_g|}$$

These values are assigned as the outputs 'S21_0' and 'S21_1' in Figure 19. Now it is time to study the single-to-noise ratio of the steady state which is defined as the voltage difference between the two states $|V_{sig}|$ divided by two times the standard deviation of the thermal noise added by the amplification chain [8]:

$$SNR = \frac{|V_{sig}|}{2\sigma}$$

$$|V_{sig}| = \sqrt{\left(|V_o^{(0)}| \sin \theta^{(0)} - |V_o^{(1)}| \sin \theta^{(1)}\right)^2 + \left(|V_o^{(0)}| \cos \theta^{(0)} - |V_o^{(1)}| \cos \theta^{(1)}\right)^2}$$

The standard deviation of the measured voltage distribution σ is equal to the RMS voltage of the system noise of the amplification chain [8]:

$$\sigma = V_{RMS} = \sqrt{R_o \hbar \omega_g (1/2 + n_{sys}) B}$$

Where n_{sys} is the noise added by the amplification chain. For a 50 Ω system $n_{sys} = k_B T_{eff} / \omega_g \hbar$ with k_B being the Boltzmann constant and $T_{eff} = T_{ref} [10^{NF/10} - 1]$. This project will set the temperature of reference at 290 K. NF is the total noise figure of the amplification chain and it's a parameter editable by the user. The parameter B is the equivalent noise bandwidth of the measurement chain due to the duration of the readout pulse τ_m and is given by [8]:

$$B = 1/\tau_m$$

Once the SNR has been obtained its value in decibels is assigned to the output called 'SNR_dB' seen in Figure 19.

To represent the noise of the amplification chain a pseudo-random sequence of numbers is generated following a normal distribution with mean 0 and standard deviation σ . After having obtained all needed parameters it is possible now to create the signal V_o , depending on the state chosen by the user. This signal takes the following form:

$$V_o = |V_o| \cos(\omega_g t + \theta) + \sigma$$

The signal will be assigned to the output 'Vo_Noisy' seen in Figure 19 and its noiseless variant will be assigned to the output 'Vo' also seen in Figure 19. Finally the fidelity of the readout is calculated as [8]:

$$F \approx \exp\left(-\frac{\tau_m}{2T_1}\right) \text{erf}(SNR/\sqrt{2})$$

where T_1 is the relaxation time of the qubit and erf is the error function. Finally, the fidelity is assigned to the output 'Fidelity' seen in Figure 19. Outputs 'Vo_0' and 'Vo_1' will show $|V_0^{(0)}|$ and $|V_0^{(1)}|$ respectively. It is important to know the fidelity of a qubit because it represents how well the actual quantum computation matches the ideal computation. With this parameter the accuracy of a quantum operation transforming an initial qubit state to the desired final state is measured.

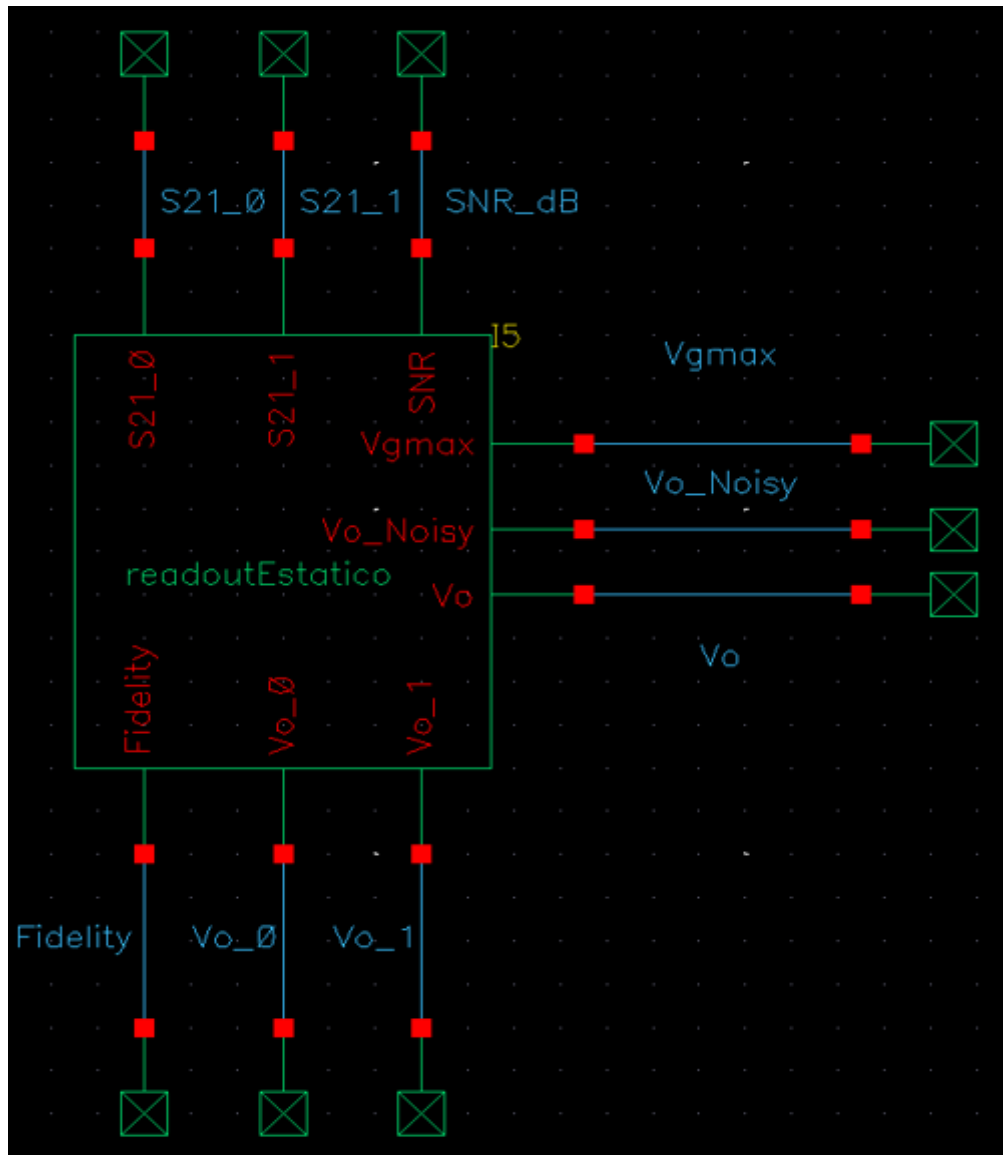


Figure 19: Circuit schematic for the steady-state readout of the transmon qubit.

Figure 20 illustrates the main functions performed by the control block as well as how the user shall operate the testbench.

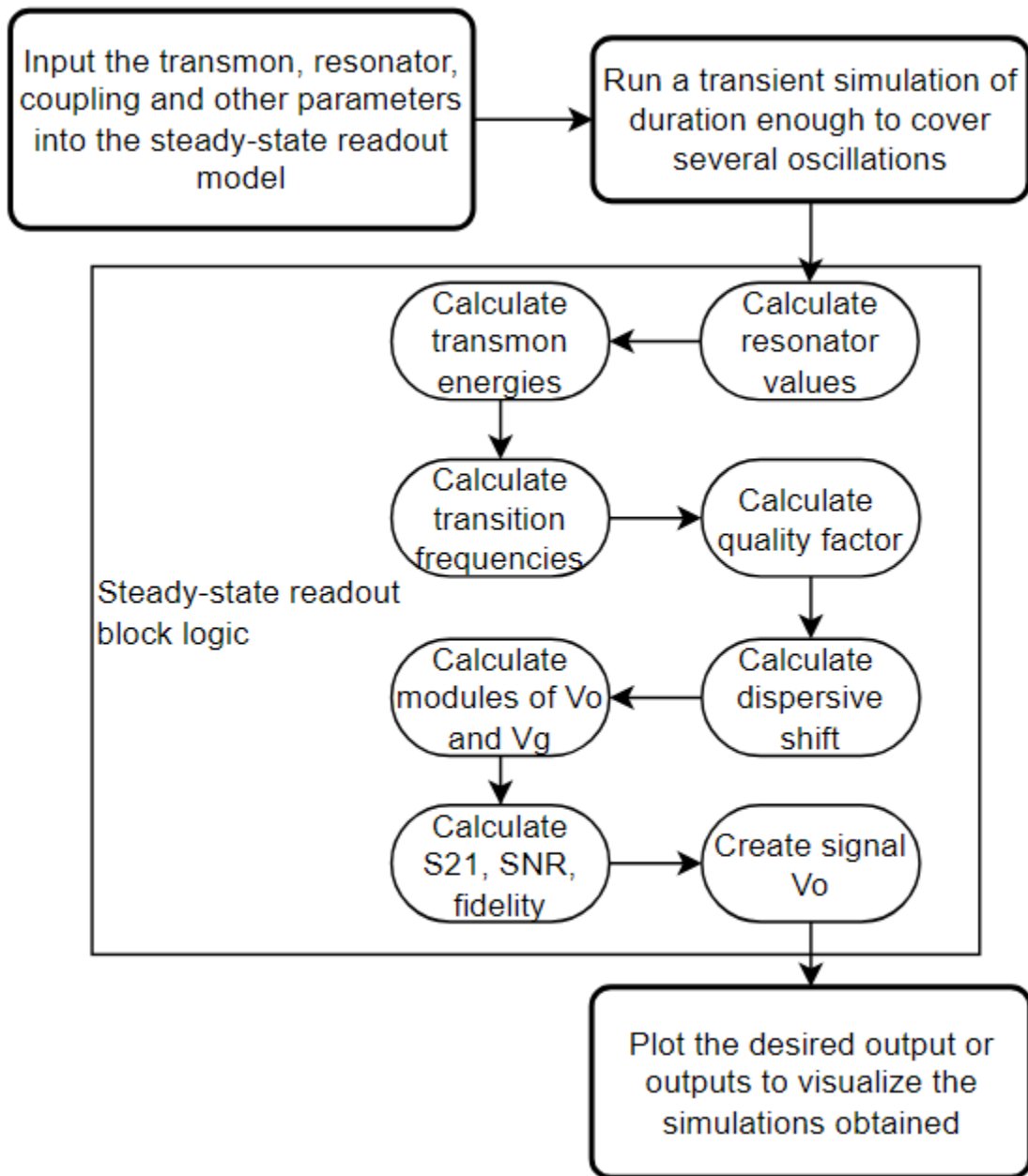


Figure 20: Flow diagram of the steady-state readout testbench.

The steady-state readout model will be performing several calculations which are needed to create the signal V_0 , both noisy and noiseless. If the reader is interested in the Verilog-A model the code is available in its entirety in Appendix B. Results can be found in section 5.2.

4.2.2 Transient readout

This design differentiates from the previous one in three aspects. The first one is that the noise of the amplification chain will be calculated and added using a separate Verilog-A model. The second difference is how the readout signals are going to be treated. The transient state readout model will use as input a signal driven by V_g with frequency and amplitude set up by the user. Finally, this testbench offers more precise simulations because the readout is usually performed before the steady-state has been reached. The main disadvantage is the long simulation times because the tolerances of the simulator need to be reduced from the default ones.

The transient readout block will implement the RLC model of the resonator with its couplings to the qubit and the transmission line as well as an extra inductance in series with its own that will emulate the dispersive shift effect as can be seen in Figure 21. Choosing the state of the qubit will vary the value of this inductance.

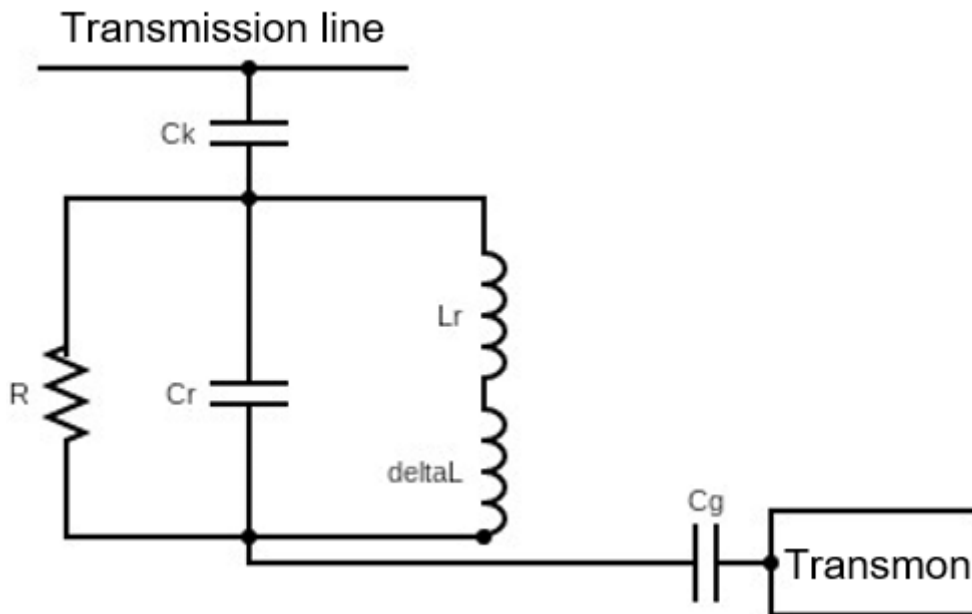


Figure 21: Simplified circuit schematic of the transient state readout block

As has been done in the previous design, the first step involves deciding which parameters the user may input. These are summarized in Table 2.

| Transmon parameters | | Resonator parameters | | Coupling parameters | | Other parameters | | Noise block parameters | |
|----------------------------|-----|-------------------------------------|--------------|-----------------------------------|-----|--|--------------|---|---------|
| Inductance (L_j) | [H] | Loss factor (α) | [1/m] | TxLine-Resonator (C_k) | [F] | Resistor's impedance (R_o) | [Ω] | Source's angular frequency (ω_g) | [rad/s] |
| | | Length (len) | [mm] | | | Characteristic impedance of TxLine (Z_o) | [Ω] | | |
| Capacitance (C_s) | [F] | Characteristic impedance (Z_o') | [Ω] | | | Resonator-Transmon (C_g) | [F] | Noise figure (NF) | [dB] |
| Qubit state (Choose_State) | [-] | Angular frequency (ω_r) | [rad/s] | Duration of the pulse (pulseTime) | [s] | | | Duration of the pulse (pulseTime) | [s] |

Table 2: Summary of editable parameters of the transient state testbench with the variable names used in the code inside round brackets.

By looking at Figure 22 it is noticeable that more components are now present in the circuit schematic compared to the steady-state readout. The sinusoidal voltage source 'V5' will play the role of V_g in Figure 17, the component called 'readoutDinamico' is the main Verilog-A model that will work as the pair resonator-transmon and the component called 'noiseBlock' will introduce the noise effects of an amplification chain at room temperature. To properly utilize this testbench the user will need to perform two different transient simulations and change the tolerances of the Cadence Virtuoso simulator. The tolerances changed for this project were *reltol* (from 1e-3 to 1e-9), *vabstol* (from 1e-6 to 1e-12) and *ibstol* (from 1e-12 to 1e-17).

The first simulation should have a short duration (1 ns is enough) to obtain the ω_o , χ and V_g values valid for the chosen state, both ω_o and χ will be presented in Hertz for an easier conversion for the user. These can be read by plotting the outputs 'W_o', 'Disp_Shift' and 'Vgmax' seen in Figure 22 and are calculated following the implementation explained in section 4.2.1. Once these values are available the user can set V_g with the amplitude obtained from 'Vgmax' and frequency $f_g = \omega_g/2\pi = (\omega_o - range \cdot \chi)/2\pi$. To set the frequency the user will have to operate the values obtained from ω_o and χ . The dispersive shift multiplied by the chosen range has to be added (or subtracted) to the value obtained in ω_o and the result obtained will be the frequency to use for the voltage source. The angular frequency ω_g is transformed into frequency since the voltage source component available in Cadence works using frequencies as input, both ω_o and χ are transformed into frequencies by the block to make it easier for the user to calculate f_g . This first short simulation is run so the user can find f_g and V_g and then use these values for the second simulation. The 'range' parameter is only used as a reminder for the user to scale the dispersive shift before adding or subtracting it to ω_o .

In order to know how these parameters are calculated refer to section 4.2.1 of this document. It is also important to note that ω_o and χ are already presented to the user in frequency and not angular frequency for easier usage.

The second simulation should have a duration of several microseconds depending on the quality factor of the resonator if the user is interested to study how the system behaves until it reaches the steady-state. If not, the user may simulate for the duration of the readout pulse in order to avoid long simulation times.

The way the Verilog-A model of the transmon-resonator pair is represented is by the resonator's RLC parallel configuration coupled to the transmon. This coupling is calculated as the inductance needed in series with the resonator inductance to shift its resonant frequency. This is done knowing that the resonator's inductance is [10]:

$$L_r = \frac{1}{C_r \omega_r^2}$$

Then the difference between the inductance of the resonator only coupled to the transmission line and the inductance of the resonator coupled both to the qubit and the transmission line:

$$\Delta L = \frac{1}{(C_r + C_k)(\omega_o \pm \chi)^2} - L_r$$

When this value is obtained it is then added to the inductive branch of the resonator with a conditional statement depending on the state chosen by the user. This way it is possible to simulate three different situation: the resonator coupled to the qubit in either state and the transmission line and, if the user sets the 'Choose_State' parameter to any other value different than 1 or 0, the resonator only coupled to the transmission line

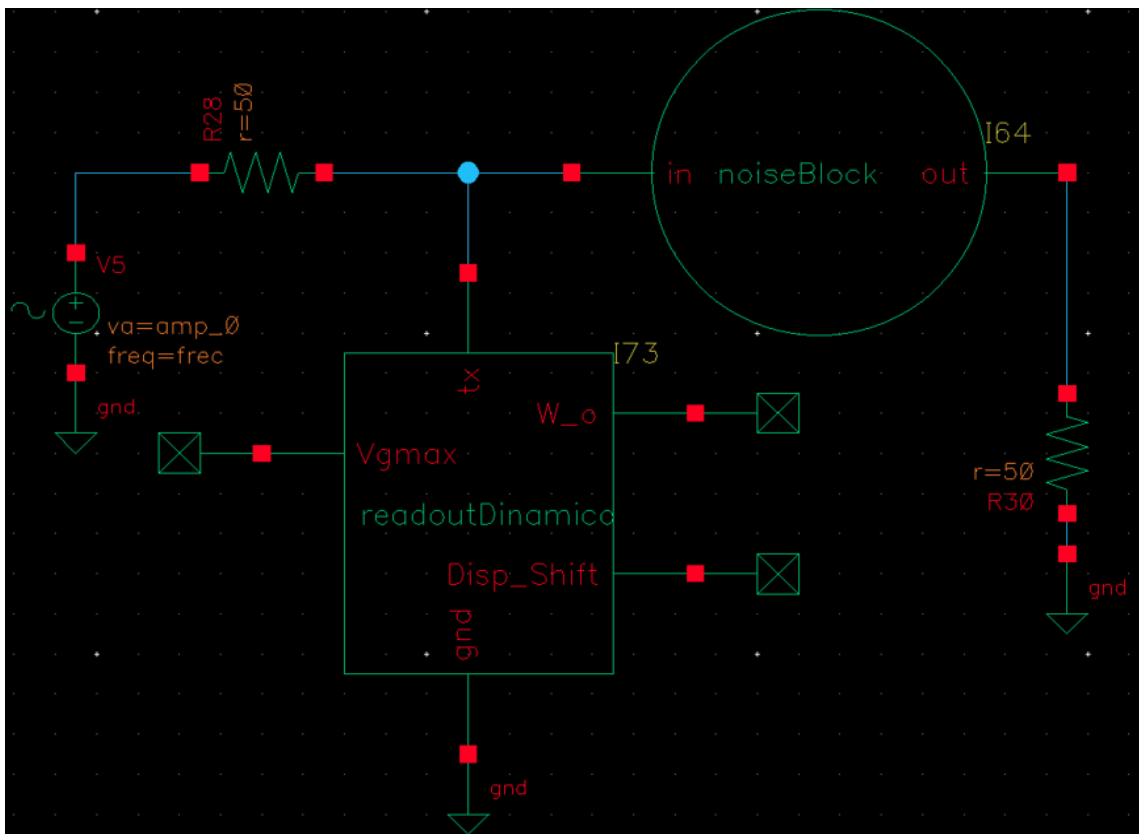


Figure 22: Circuit schematic for the transient state readout of the transmon qubit.

Since the signal is weak in amplitude an amplification chain is needed which will add noise to the system. The way the noise is emulated in this testbench is by adding it to the signal V_o , which is the response of the resonator to the input V_g . The noise is then calculated as has been done in section 4.2.1 for the steady-state readout and the output is the same signal that has been taken at the input adding to it the pseudo-random sequence of numbers created following a normal distribution with mean 0 and standard deviation σ .

This chapter will be closed showing in Figure 23 the main functions performed by the transient state and noise blocks as well as how the user shall operate the testbench. Results will be provided in section 5.3

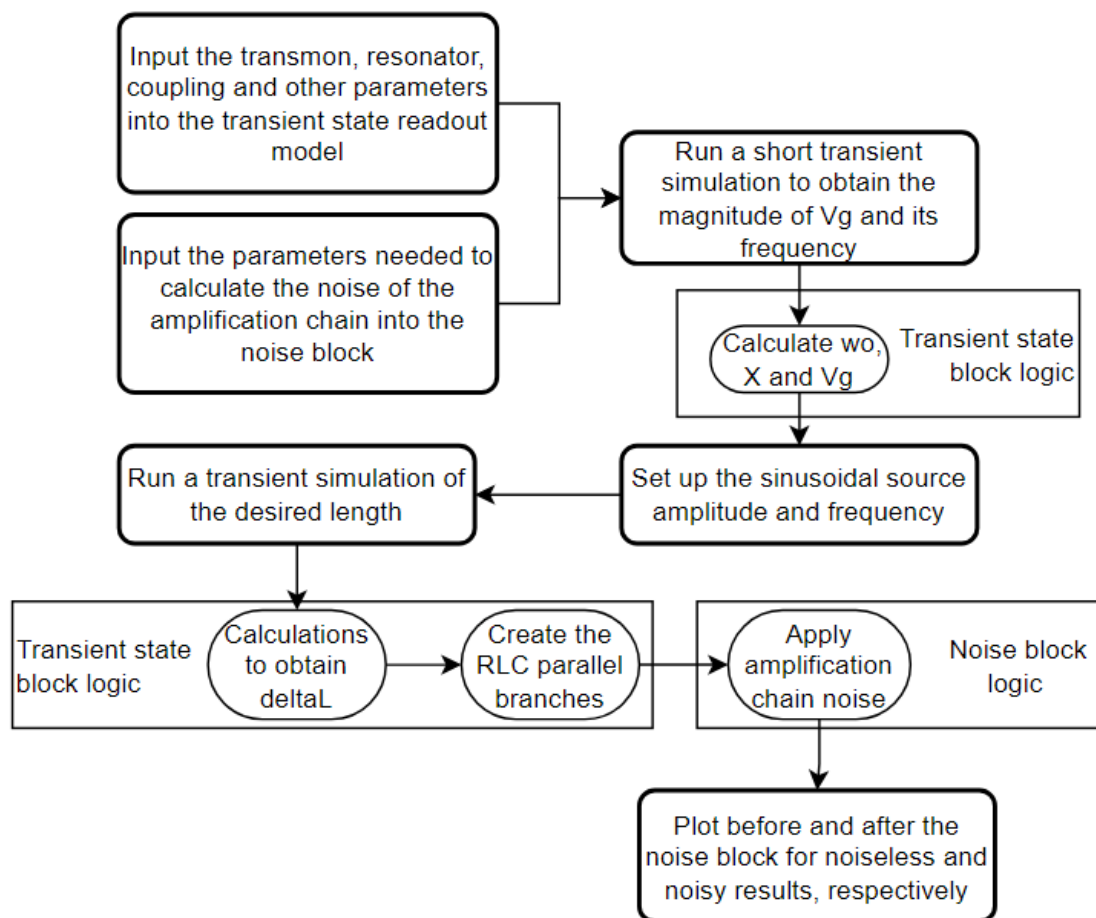


Figure 23: Flow diagram of the transient state readout testbench.

5 RESULTS

In this chapter the signals involved in the control and readout processes will be studied while varying parameters like the state of the qubit, the frequency at which the resonator is probed or the amplitudes of control and readout pulses. First, the control testbench will be analyzed, followed by the steady-state readout testbench and finally the transient state testbench.

5.1 Qubit control testbench

To test the functionality of this testbench the following experiment will be set up. The control pulse used to drive the qubit will be fixed at 600 ns and 16.4 μV , then a set of simulations sweeping the amplitude parameter of the pulse will be run. Then, this set will be plotted and discussed. An example of the Gaussian pulse used as input for the control block can be seen in Figure 24, normalized to 1 V before scaling it to the desired amplitude.

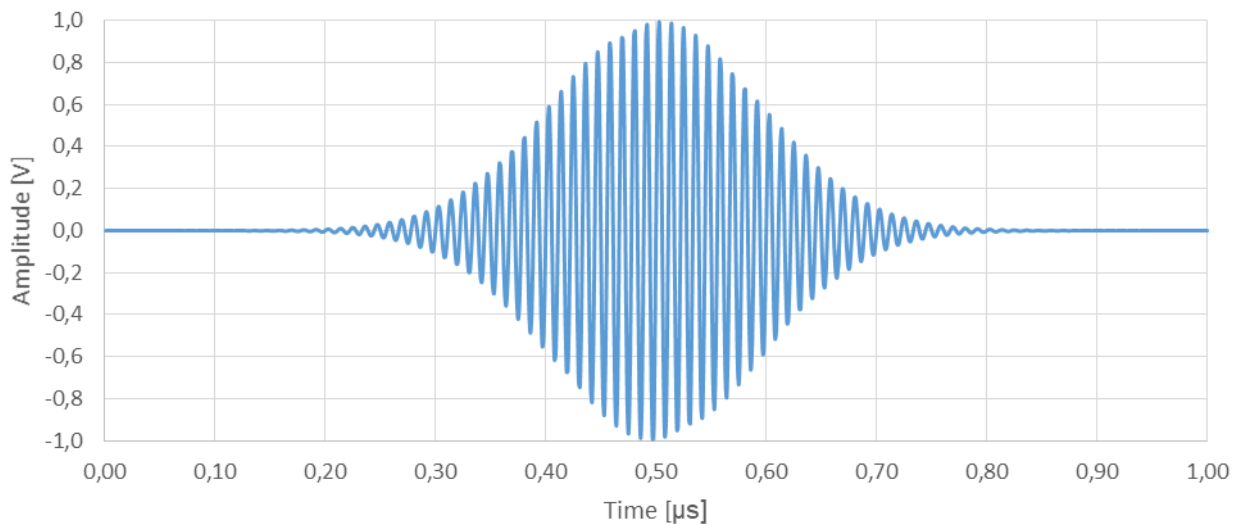


Figure 24: Gaussian pulse obtained after mixing a sinusoidal voltage source with a Gaussian envelope of duration 600 ns and normalized amplitude to 1 V.

The control block will extract the envelope of the input signal and perform its time integral. Once this integration has been done the accumulated rotation angle $\theta(t)$ is calculated as explained in section 4.1 and the probabilities to find the qubit in state $|0\rangle$ or $|1\rangle$ are $\cos(\theta(t)/2)^2$ and $\sin(\theta(t)/2)^2$ respectively. Figure 25 shows the accumulated rotation angle of the qubit when driven by a pulse of amplitude 16.4 μV and duration 600 nanoseconds.

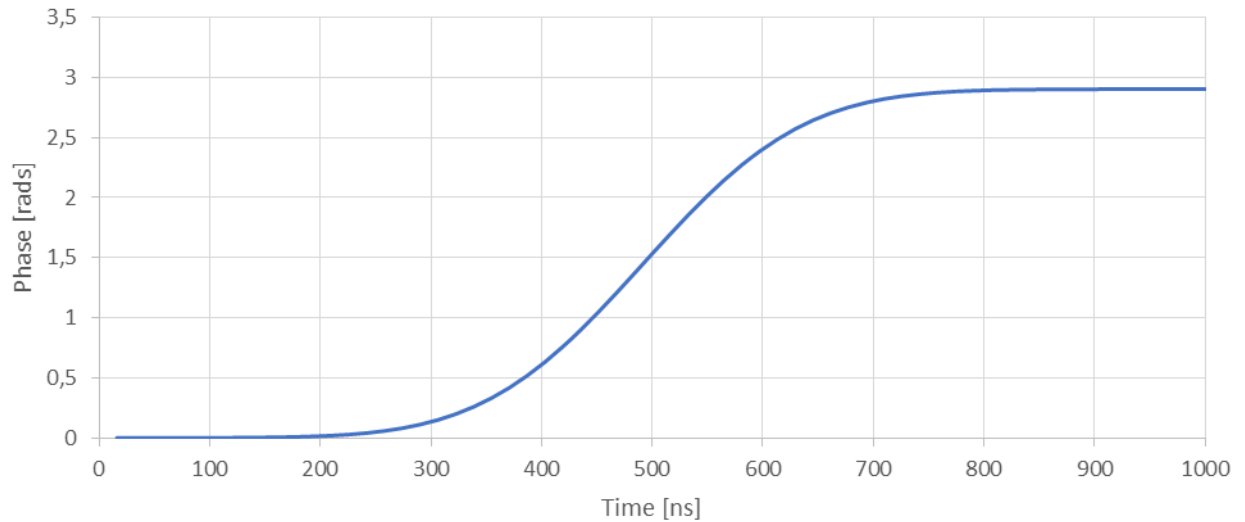
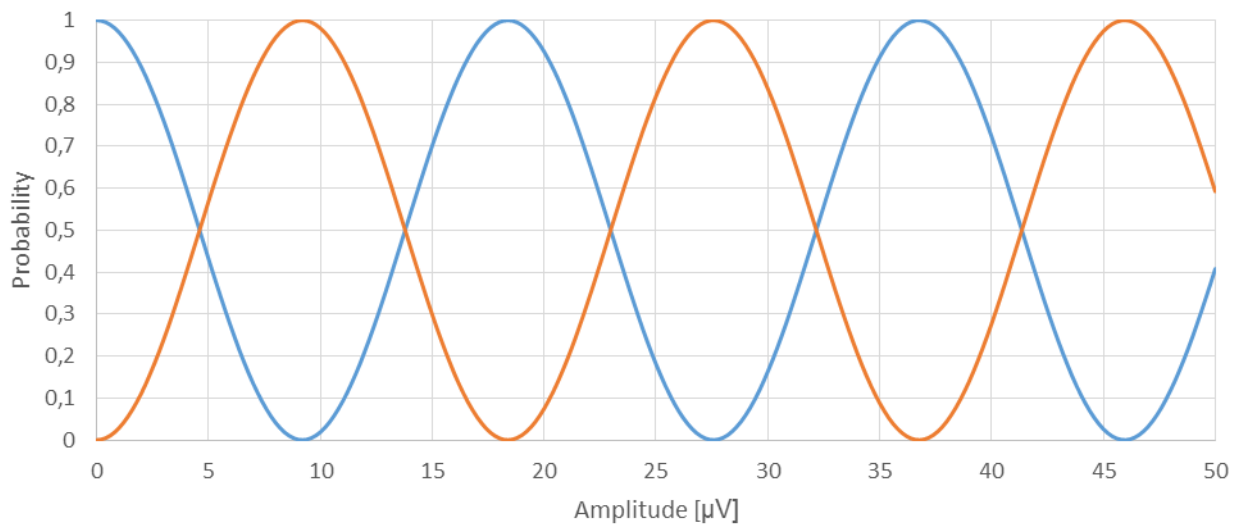


Figure 25: Accumulated rotation angle as a function of time

Figure 26 shows how sweeping the amplitude of the control pulse changes the probability of finding the qubit in state $|0\rangle$ or $|1\rangle$. An oscillation of the probability appears when the amplitude keeps increasing because these pulses provoke rotations of the qubit around the Bloch sphere. By looking at the graph it can be seen that to perform a π -pulse on the qubit, a drive signal of duration 600 ns and amplitude $9.4 \mu\text{V}$ is needed. π -pulses are specially interesting because they allow transitioning the qubit from state $|0\rangle$ to state $|1\rangle$, or vice versa, by performing a 180° rotation. As seen in the plot, around $19.5 \mu\text{V}$ the qubit state has done a full rotation along the Bloch sphere and came back to its original position.


 Figure 26: Probabilities of finding the qubit in state $|0\rangle$ (blue curve) or state $|1\rangle$ (orange curve) depending on the amplitude used for the control pulse with the qubit starting in state $|0\rangle$.

5.2 Qubit steady-state readout testbench

To obtain the results that validate this testbench several transient simulations are going to be run using the qubit described in section 4 as the model. These simulations will vary the qubit state as well as the angular frequency of the readout pulse. Both states will be studied and for each of them the angular frequency of the readout pulse ω_g will take three different values, these angular frequencies will be $\omega_o^{|0\rangle}$, $\omega_o^{|1\rangle}$ and ω_o .

The input parameters mentioned in section 4.2.1 will be fixed for all simulations except 'Chosen_State' and 'range'. The first parameter will simulate which state the qubit is in and the second parameter will shift the angular frequency ω_g to the desired value. Since the resonator experiences a noticeable drop in its S_{21} parameter at a frequency that depends on the state of the coupled qubit (seen in Figure 40) the readout pulse will undergo changes at the end of the readout chain, reflected as different attenuations and phase changes. Both noisy and noiseless simulations will be run to study how an amplification chain with noise figure $NF = 0.02 \text{ dB}$ affects the readout process. The simulation time will be kept at 2 nanoseconds to show several cycles of the signal.

The reader may notice symmetry around ω_o in the results obtained when the state of the qubit is changed due to the shift induced by the qubit state-dependent coupling. The shift produced is $\chi = -8.79 \cdot 10^6 \text{ rad/s}$ to either side of ω_o . The maximum allowed amplitude for each pair of simulations is going to be upper-bounded by the lowest V_g of the two states allowed. It is expected to obtain very noisy outputs when the voltage source frequency is located in the boundaries of the dispersive regime and the signal-to-noise ratio will be maxed when the angular frequency of the probing signal is ω_o .

5.2.1 Simulations when $\omega_g = \omega_o^{|0\rangle}$

The first situation to be studied is with the qubit in state $|0\rangle$ and $\omega_g = \omega_o^{|0\rangle} = \omega_o - \chi$. The angular frequency of the readout pulse is shifted to $\omega_g = 3.12400181 \cdot 10^{10} \text{ rad/s}$ with an allowed amplitude at the voltage source $V_g = 0.2172 \mu\text{V}$. The expected amplitude at the end of the probing is $V_o = 0 \text{ V}$. Figure 27 shows the noiseless result of the simulation while Figure 28 shows the noisy result, these graphs show how all the signal has been absorbed by the resonator and only the noise added by the amplification chain is present at the end of the readout chain.

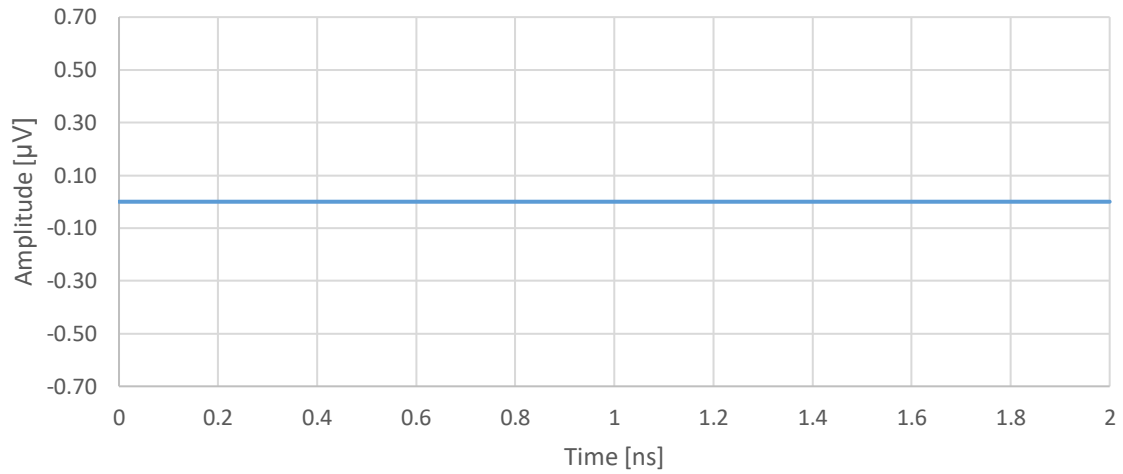


Figure 27: Noiseless output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

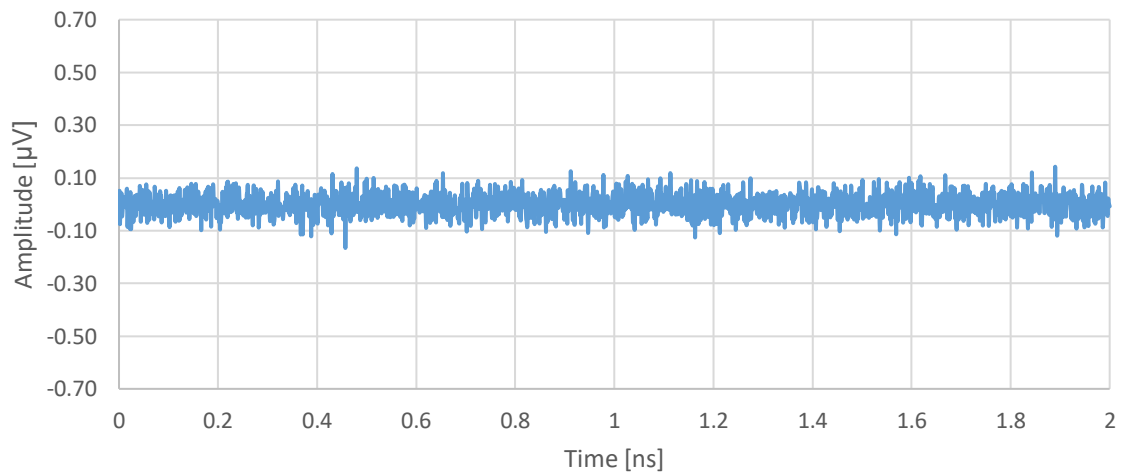


Figure 28: Noisy output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

The second simulation will set the qubit in state $|1\rangle$ and $\omega_g = \omega_o^{(0)} = \omega_o - \chi$. The angular frequency of the readout pulse is shifted to $\omega_g = 3.12400181 \cdot 10^{10} \text{ rad/s}$ with an allowed amplitude at the voltage source $V_g = 0.2172 \mu\text{V}$. The expected amplitude at the end of the probing is $V_o = 0.10803 \mu\text{V}$. Figure 29 shows the noiseless result of the simulation while Figure 30 shows the noisy result.

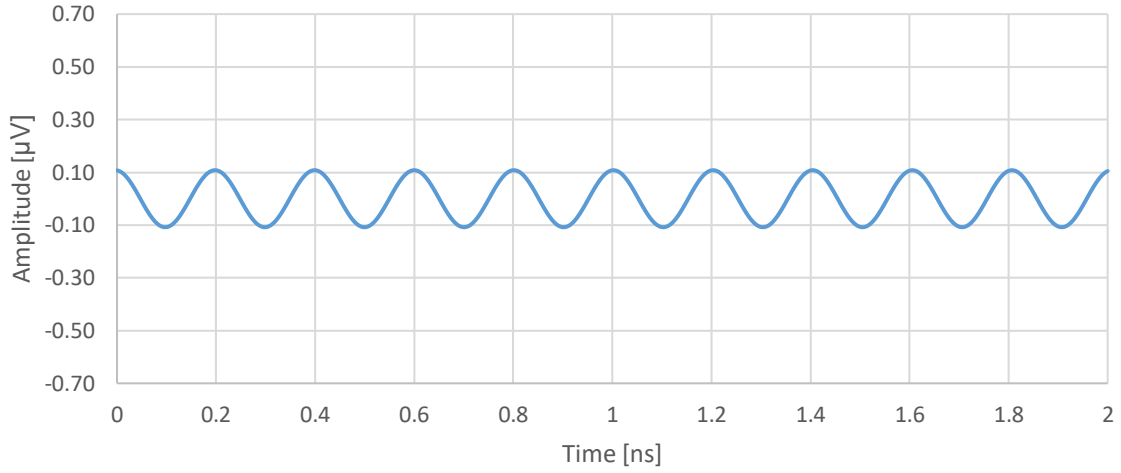


Figure 29: Noiseless output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

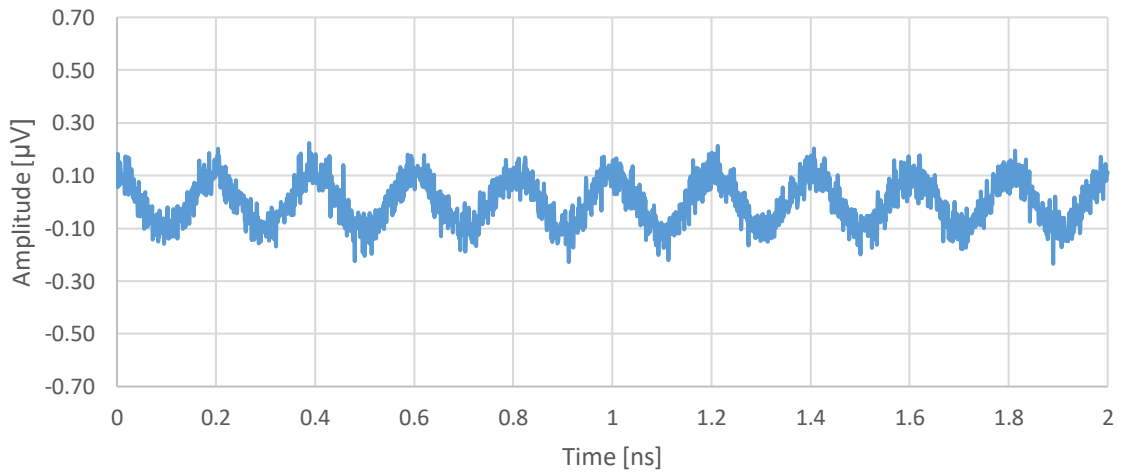


Figure 30: Noisy output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

The S_{21} parameters for the state $|1\rangle$ and the state $|0\rangle$ are $S_{21}^{(1)} = -0.045 \text{ dB}$ and $S_{21}^{(0)} = -\infty \text{ dB}$. Since the state of the qubit cannot be known before performing the readout, the maximum amplitude allowed for the probing signal is $V_g = 0.2172 \mu\text{V}$ for $\omega_o^{(0)}$. The added noise which emulates the amplification chain is too big in comparison to the output signals making the readout not viable.

5.2.2 Simulations when $\omega_g = \omega_o^{1\rangle}$

The first situation to be studied is with the qubit in state $|1\rangle$ and $\omega_g = \omega_o^{1\rangle} = \omega_o + \chi$. The angular frequency of the readout pulse is shifted to $\omega_g = 3.12224277 \cdot 10^{10} \text{ rad/s}$ with an allowed amplitude at the voltage source $V_g = 0.2172 \mu\text{V}$. The expected amplitude at the end of the probing is $V_o = 0 \text{ V}$. Figure 31 shows the noiseless result of the simulation while Figure 32 shows the noisy result, these graphs show how all the signal has been absorbed by the resonator and only the noise added by the amplification chain is present at the end of the readout chain.

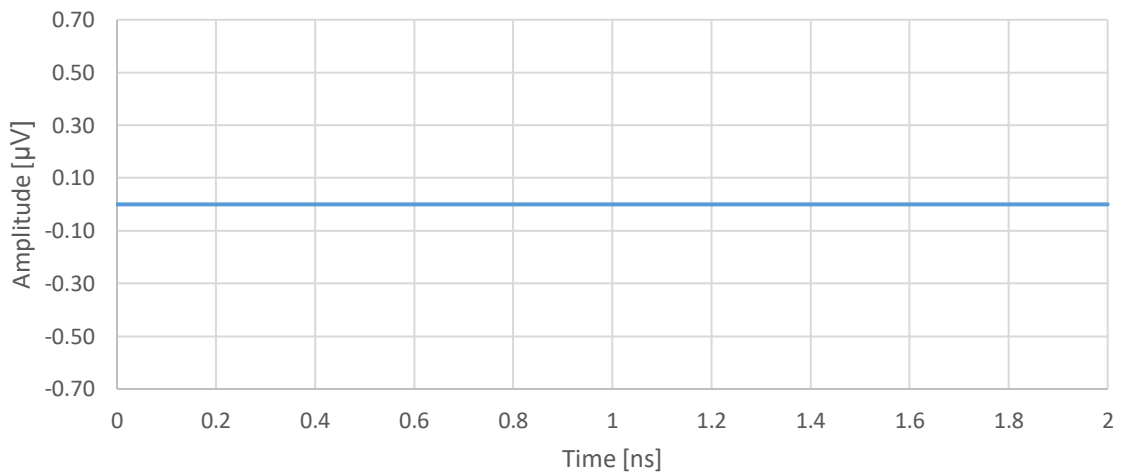


Figure 31: Noiseless output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{1\rangle}$

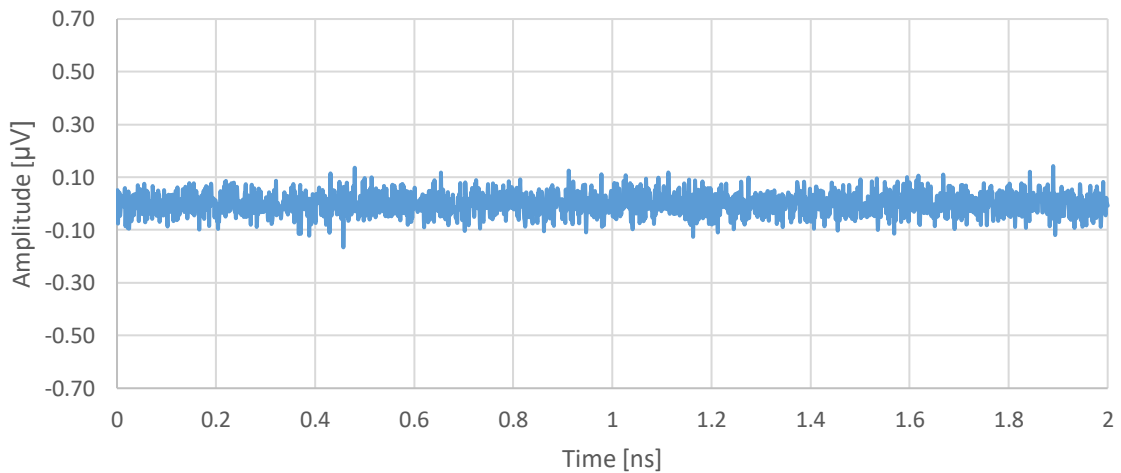


Figure 32: Noisy output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{1\rangle}$

The second simulation will set the qubit in state $|0\rangle$ and $\omega_g = \omega_o^{(1)} = \omega_o + \chi$. The angular frequency of the readout pulse is shifted to $\omega_g = 3.12224277 \cdot 10^{10} \text{ rad/s}$ with an allowed amplitude at the voltage source $V_g = 0.2172 \mu\text{V}$. The expected amplitude at the end of the probing is $V_o = 0.10803 \mu\text{V}$. Figure 33 shows the noiseless result of the simulation while Figure 34 shows the noisy result.

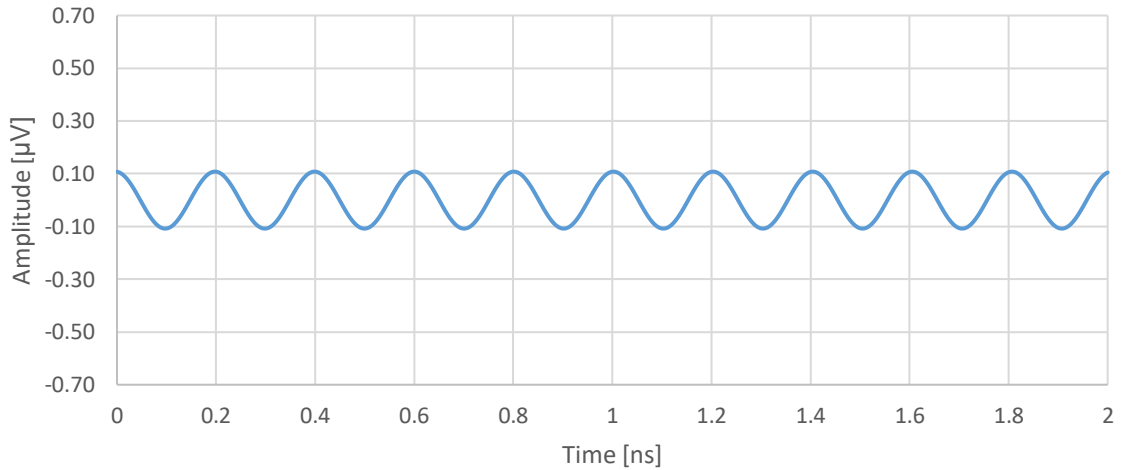


Figure 33: Noiseless output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

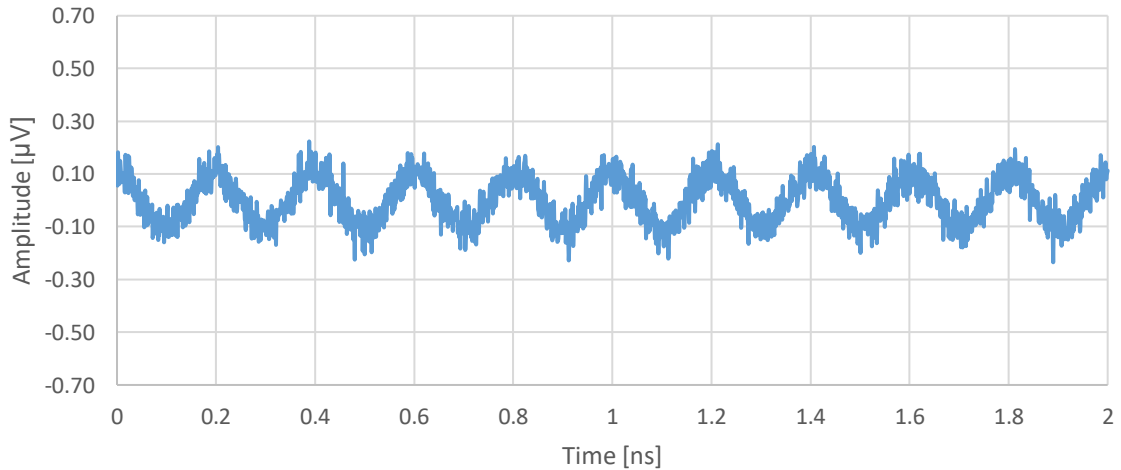


Figure 34: Noisy output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

The S_{21} parameters for the state $|1\rangle$ and the state $|0\rangle$ are $S_{21}^{(1)} = -\infty \text{ dB}$ and $S_{21}^{(0)} = -0.045 \text{ dB}$. Since the state of the qubit cannot be known before performing the readout, the maximum amplitude allowed for the probing signal is $V_g = 0.2172 \mu\text{V}$ for $\omega_o^{(0)}$. The results of this section are symmetrical to section 5.2.1 because the shifts of the angular frequency are symmetrical with respect to ω_o .

5.2.3 Simulations when $\omega_g = \omega_o$

The first situation to be studied is with the qubit in state $|1\rangle$ and $\omega_g = \omega_o$. The angular frequency of the readout pulse is shifted to $\omega_g = 3.12312230 \cdot 10^{10} \text{ rad/s}$ with an allowed amplitude at the voltage source $V_g = 1.0760 \text{ } \mu\text{V}$. The expected amplitude at the end of the probing is $V_o = 0.52693 \text{ } \mu\text{V}$. Figure 35 shows the noiseless result of the simulation while Figure 36 shows the noisy result.

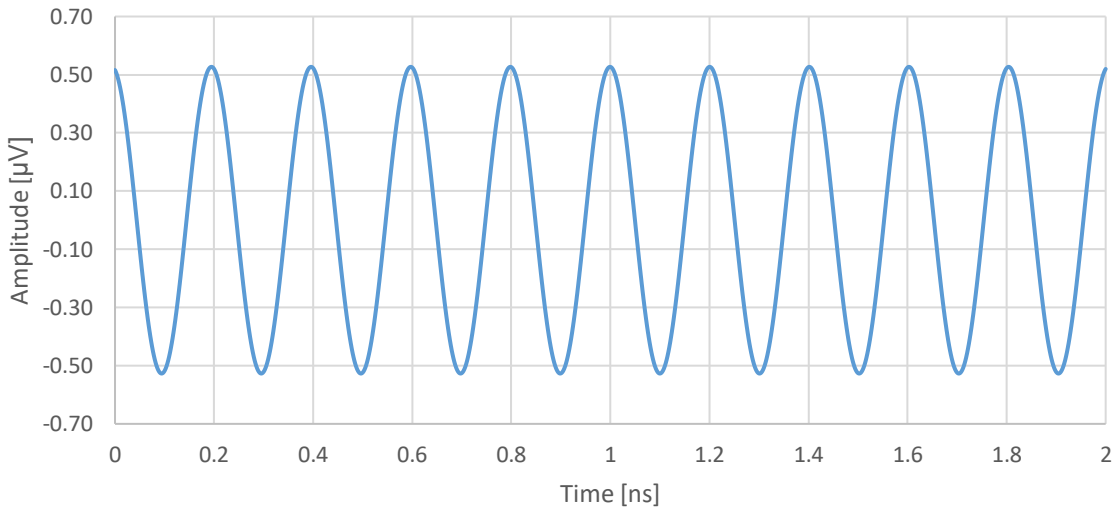


Figure 35: Noiseless output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o

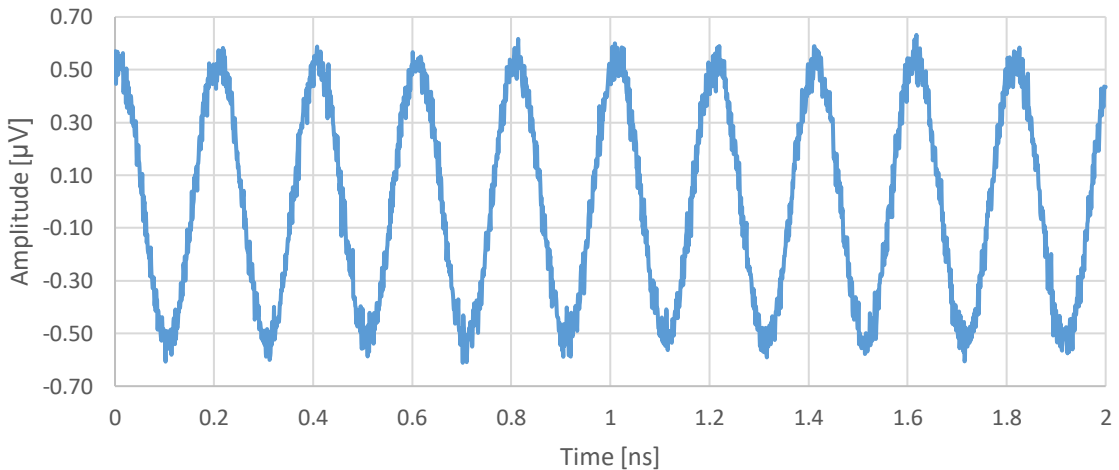


Figure 36: Noisy output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o

The second simulation will set the qubit in state $|0\rangle$ and $\omega_g = \omega_o$. The angular frequency of the readout pulse is shifted to $\omega_g = 3.12312230 \cdot 10^{10} \text{ rad/s}$ with an allowed amplitude at the voltage source $V_g = 1.0760 \text{ } \mu\text{V}$. The expected amplitude at the end of the probing is $V_o =$

$0.52693 \mu V$. Figure 37 shows the noiseless result of the simulation while Figure 38 shows the noisy result.

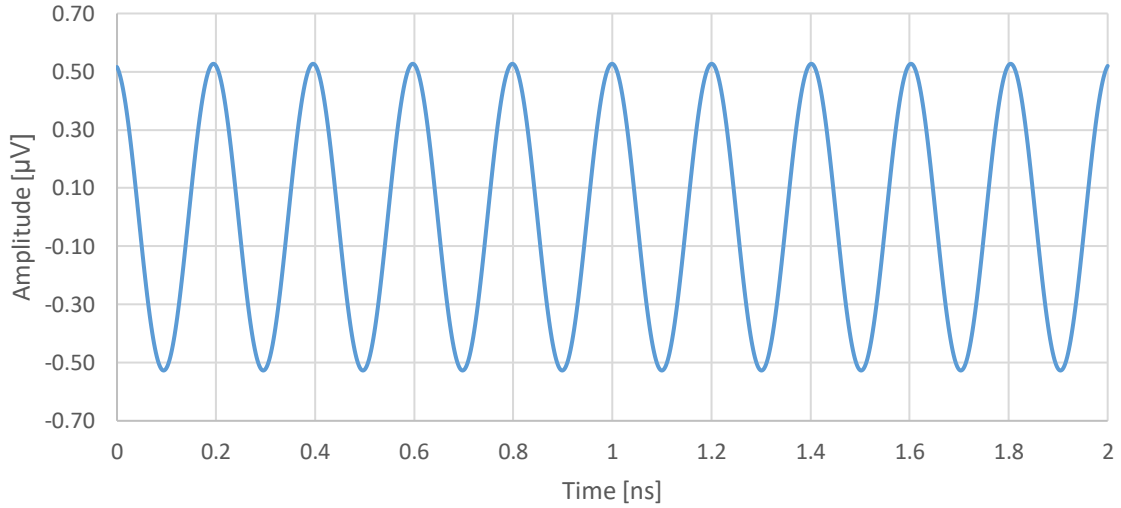


Figure 37: Noiseless output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is ω_o

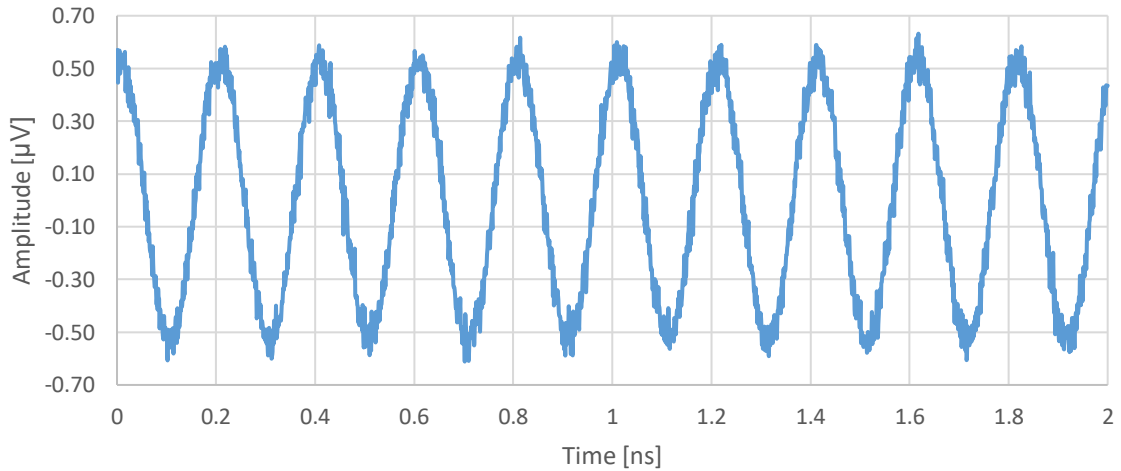


Figure 38: Noisy output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is ω_o

The signal-to-noise ratio obtained after running these simulations is $SNR = 8.291 \text{ dB}$. The S_{21} parameters for the state $|1\rangle$ and the state $|0\rangle$ are $S_{21}^{|1\rangle} = -0.18 \text{ dB}$ and $S_{21}^{|0\rangle} = -0.18 \text{ dB}$. Since the distance in frequency from the frequency of the probing signal to both shifts is the same in magnitude, the maximum allowed amplitude for the readout is the same for both states ($V_g = 1.0760 \mu V$). This is why ω_o is the best angular frequency for the readout of the qubit.

Even though the output obtained when the probing pulse is set at $\omega_g = \omega_o$ has the same amplitude when the qubit is in either state, there exists a phase difference as seen in Figure 39. If this phase difference were not present, the parameter $|V_{sig}|$ would be zero because of the amplitudes of the signals making the signal-to-noise ratio zero as well.

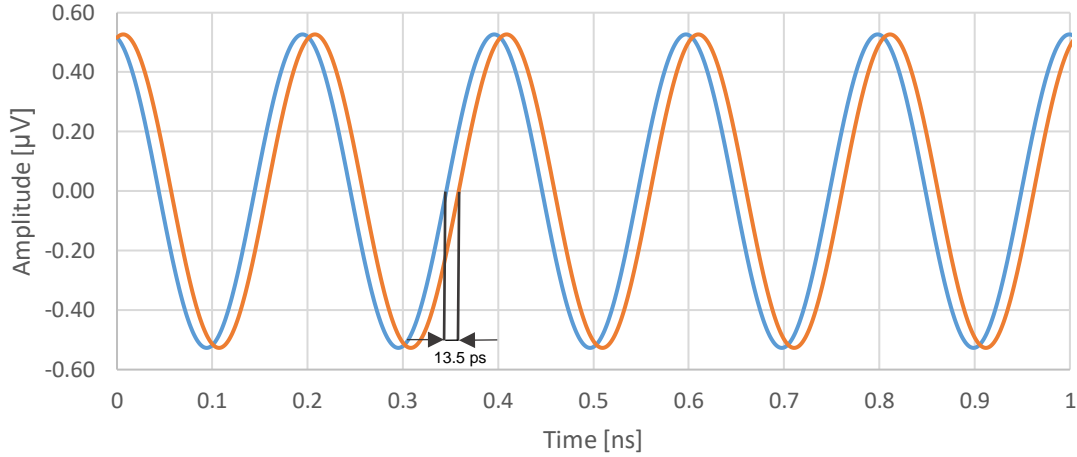


Figure 39: Noiseless output V_o when the angular frequency of the voltage source is ω_o and the qubit is in state $|1\rangle$ (blue, left) and state $|0\rangle$ (orange, right)

This phase difference at the output allows differentiating which state the qubit is in while allowing for maximum amplitude at the voltage source by probing the signal between the shifted angular frequencies.

5.3 Qubit transient state readout

The transient simulations run for this section will be 4 microseconds long, providing overall and zoomed in graphs. The parameters employed for the transient state readout will be the same as for the steady-state readout and fixed for all the simulations, only varying the state of the qubit as well as the amplitude and frequency of the voltage source seen in Figure 22. Before each simulation the user will have to run short simulations (around 1 nanosecond to avoid long wait times) in order to obtain the allowed amplitude and frequency at the voltage source as explained in section 4.2.2.

The S_{21} parameter of the circuit has been simulated (Figure 40) three times by varying the state of the qubit, the results obtained show the resonant frequencies of the resonator only coupled to the readout transmission line (ω_o), the resonator coupled to the transmission line and to the qubit in state $|0\rangle$ ($\omega_o^{|0\rangle}$) and the resonator coupled to the transmission line and to the qubit in state $|1\rangle$ ($\omega_o^{|1\rangle}$). The simulations are done using the Verilog-A block for the transient readout which emulates the RLC model of the resonator including the inductance of value ΔL . This inductance in series with the resonator's inductance is in charge of shifting the resonant peaks to either side depending on the state chosen by the user.

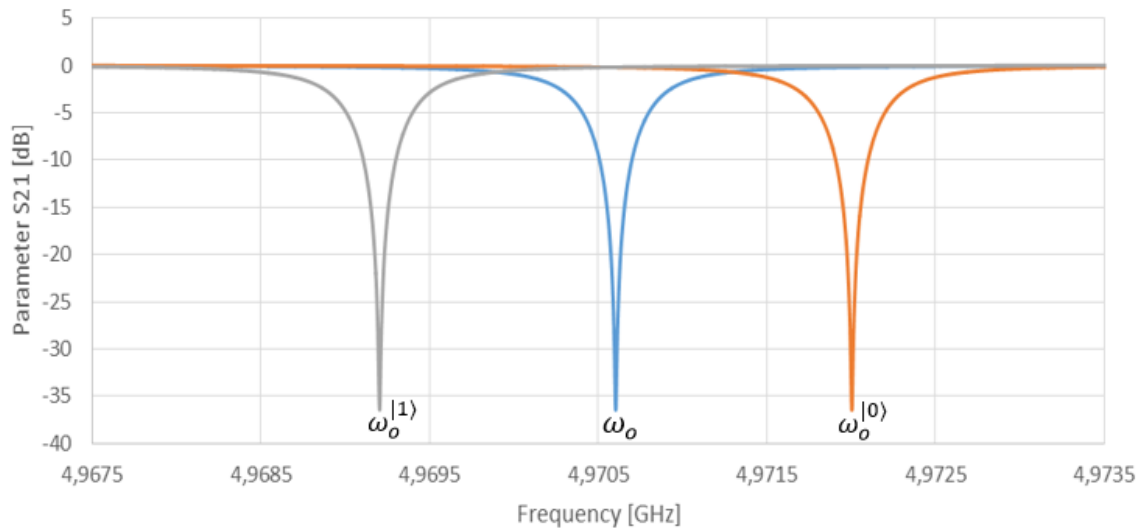


Figure 40: S_{21} parameters of the bare resonator (blue, middle), resonator coupled to the qubit in state $|1\rangle$ (grey, left) and resonator coupled to the qubit in state $|0\rangle$ (orange, right).

After validating the resonating frequencies the same situations as in the steady-state readout are going to be simulated. This means that the frequency of the voltage source ω_g will be varying between $\omega_0^{(1)}$, $\omega_0^{(0)}$ and ω_0 for both states of the qubit. The amplitude of the voltage source will also be adjusted accordingly to the maximum allowed. As has been explained in section 5.2, the user knows beforehand the state of the qubit so the allowed amplitude for each simulation will change.

The results are expected to show a ring-up effect due to the resonator. A detailed image of the signal at 600 nanoseconds will be studied as well as a detailed image once the resonator has stabilized in addition to the full simulation graphs of duration 4 microseconds. Finally, the time delay between qubit states when the voltage source is at the same frequency for both of them will be compared.

5.3.1 Simulations when $\omega_g = \omega_o^{|0\rangle}$

The first simulation will set the amplitude to $V_g = 0.2172 \mu V$ and $f_g \approx 4.9720 GHz$. Which is the amplitude and frequency correspondent to $\omega_g = \omega_o^{|0\rangle} = \omega_o - \chi$. The state of the qubit will be set to $|0\rangle$. Figure 41 shows how the resonator absorbs all the signal once the steady-state has been reached at around 3 microseconds. This result is expected due to the S_{21} parameter of the circuit when the qubit is in state $|0\rangle$ (orange curve in Figure 40). Once the signal has gone through the amplification chain only the added noise is present as can be seen in Figure 42.

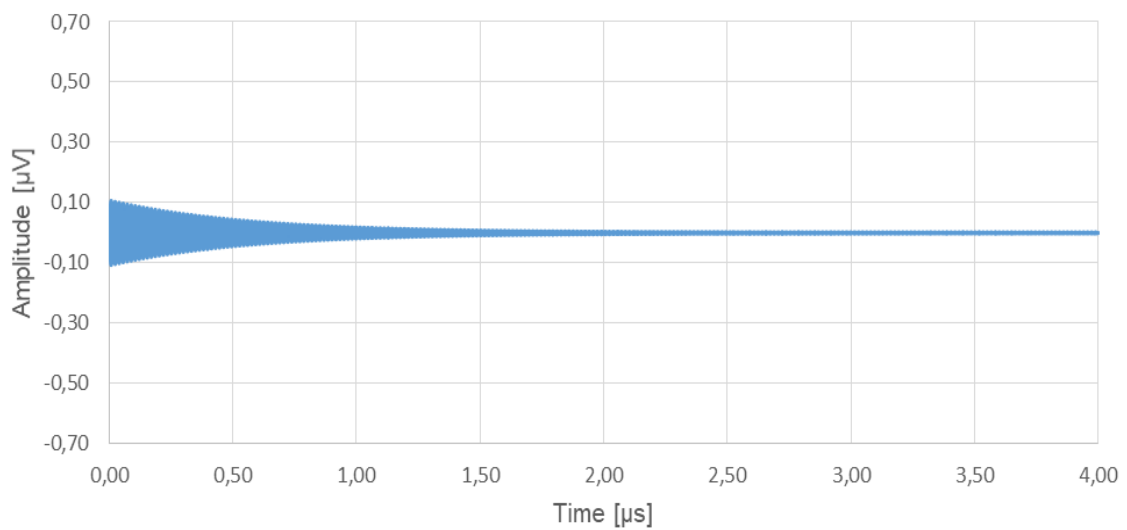


Figure 41: Noiseless output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{|0\rangle}$

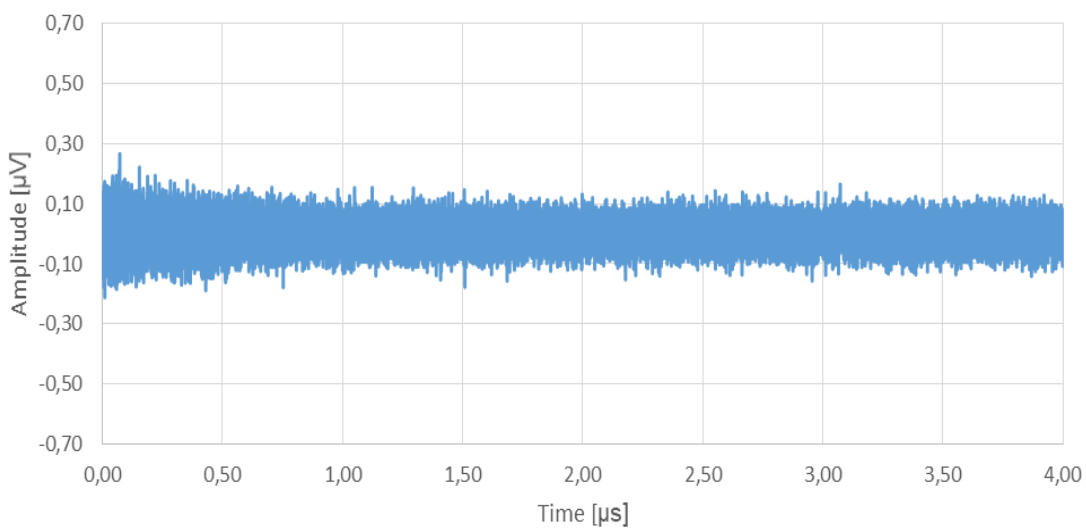


Figure 42: Noisy output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{|0\rangle}$

Now a zoomed in graph of the signals between 3.988 and 3.99 microseconds is provided in Figure 43. The orange curve is the output before noise and the blue curve is the output after noise. By comparing the orange curve to Figure 27 it is possible to appreciate that both the transient state readout and the steady-state readout testbenches show the same behavior when approaching the steady-state of the system.

Since readout pulses are shorter in duration than 4 microseconds it is also of interest to study how the signal behaves at those time stamps. The duration of the readout pulse for this project is set to 600 nanoseconds and Figure 44 shows what happens between 599 and 601 nanoseconds.

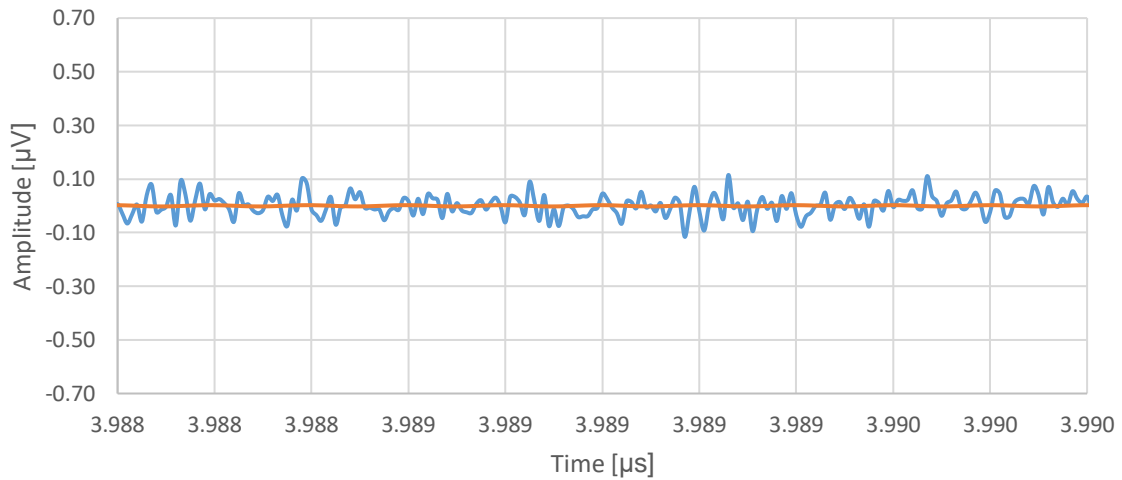


Figure 43: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

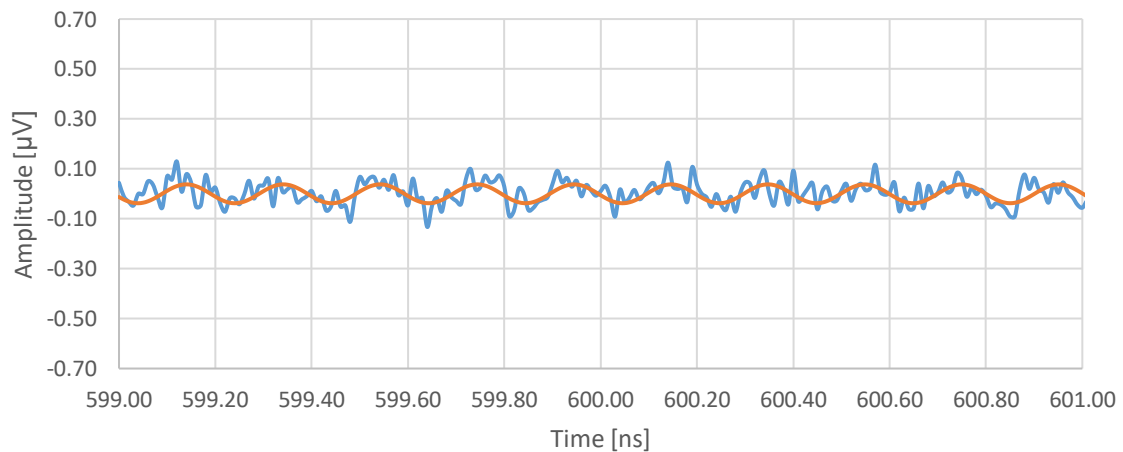


Figure 44: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 and 601 ns when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

As it can be seen the energy of the signal has not been completely absorbed by the resonator but it is very low compared to the noise to obtain any valid reading using this setup.

The second simulation will set the amplitude to $V_g = 0.2172 \mu V$ and $f_g \approx 4.9720 GHz$. Which is the amplitude and frequency correspondent to $\omega_g = \omega_o^{|0\rangle} = \omega_o - \chi$. The state of the qubit will be set to $|1\rangle$. Figure 45 shows that the time needed for the resonator to reach the steady-state is around 2 microseconds. Since the frequency of the probing signal far away from the resonant peak of the resonator, only a small portion of the signal gets absorbed. The amount of added noise can go up to 100 nanoseconds as can be seen in Figure 46 when compared to Figure 45.

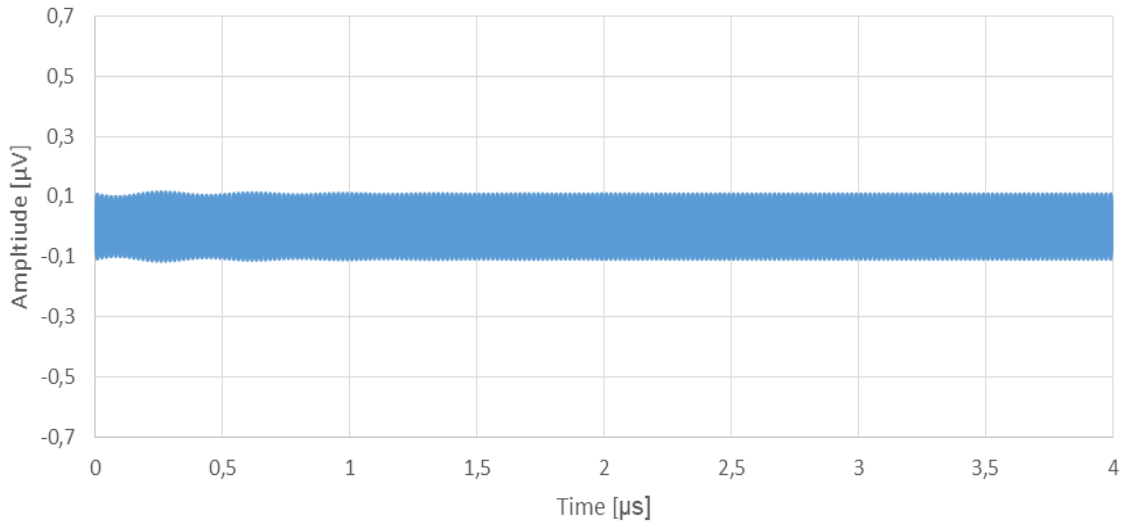


Figure 45: Noiseless output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{|0\rangle}$

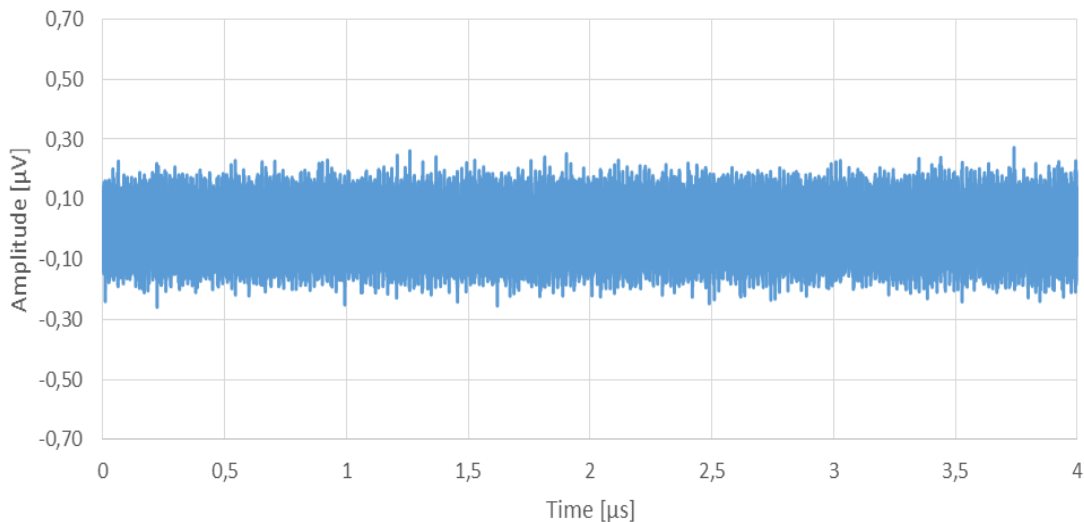


Figure 46: Noisy output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{|0\rangle}$

Analyzing in Figure 46 the noisy output of the readout, the emulated added noise that would be produced by the amplification chain distorts the output making it impossible to appreciate the ring-up effect of the resonator, the simulation used a noise figure of $NF = 0.02 dB$.

The effect of the emulated noise can be appreciated when performing a temporal zoom. Figure 47 shows the both noiseless and noisy outputs between 3.988 and 3.990 microseconds. As happened in the previous simulation, near the steady-state of the system the signals behave as expected. Comparing them to Figures 29 and 30 the results match accordingly.

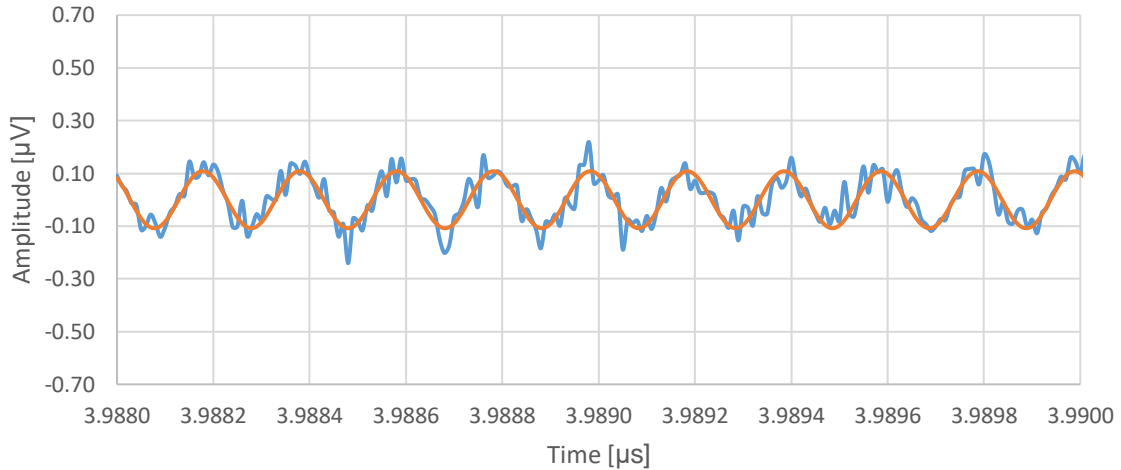


Figure 47: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

In Figure 45 it can be seen that the ring-up effect at 600 nanoseconds has similar amplitude levels as when the system approaches the steady-state. Figure 48 shows a detailed image between 599 and 601 nanoseconds which is similar to Figure 29, in the steady-state.

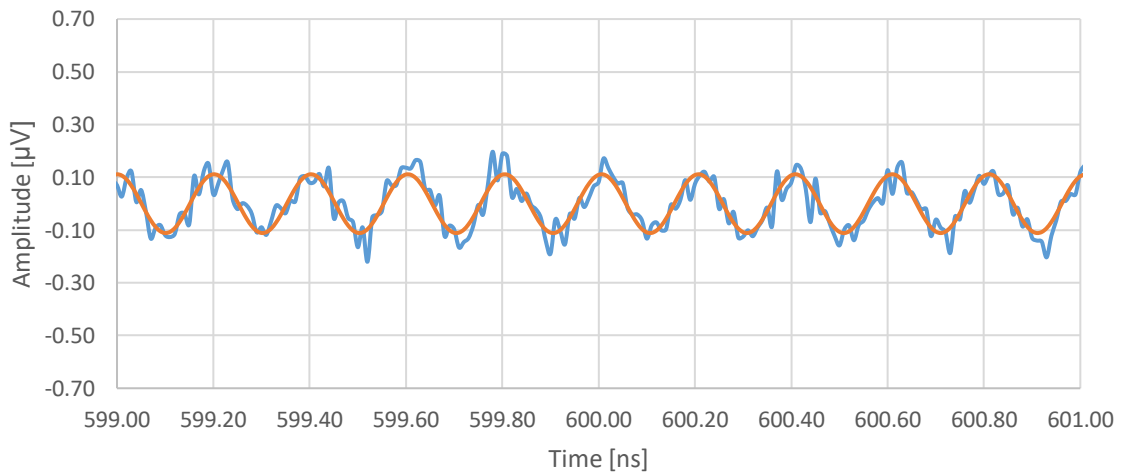


Figure 48: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 ns and 601 ns when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(0)}$

The effect of the added noise produced when emulating the amplification chain has a significant impact in the readout process, since the maximum allowed amplitude is $V_g = 0.2172 \mu V$. This translates to a low signal-to-noise ratio.

5.3.2 Simulations when $\omega_g = \omega_o$

The first simulation will set the amplitude to $V_g = 1.076 \mu V$ and $f_g \approx 4.9706 GHz$. Which is the amplitude and frequency correspondent to $\omega_g = \omega_o$. The state of the qubit will be set to $|0\rangle$. Figure 49 shows that the time needed for the resonator to reach the steady-state is around 2 microseconds. The amount of added noise can go up to 100 nanoseconds as can be seen in Figure 50 when compared to Figure 49.

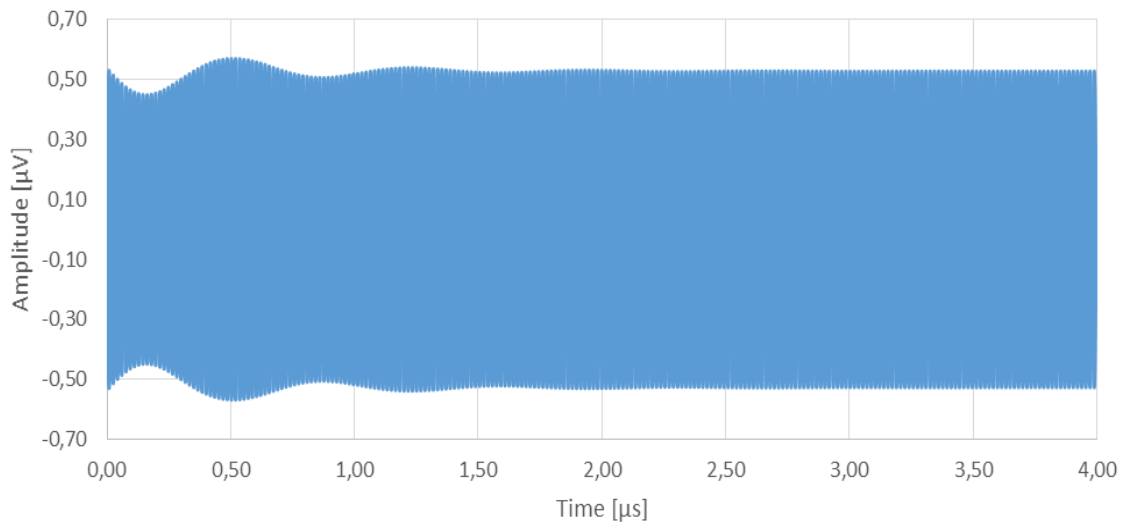


Figure 49: Noiseless output V_g when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is ω_o

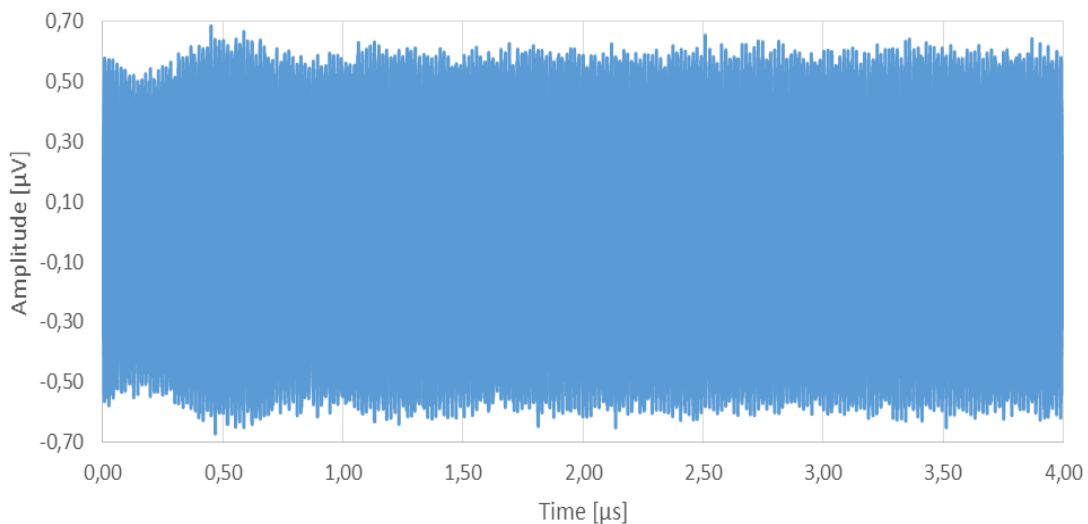


Figure 50: Noisy output V_g when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is ω_o

Now a zoomed in graph of the signals between 3.988 and 3.99 microseconds is provided in Figure 51. The orange curve is the output before noise and the blue curve is the output after noise. By comparing the orange curve to Figure 37 it is possible to appreciate that both the transient state readout and the steady-state readout testbenches show the same behavior when approaching the steady-state of the system.

Since readout pulses are shorter in duration than 4 microseconds it is also of interest to study how the signal behaves at those time stamps. The duration of the readout pulse for this project is set to 600 nanoseconds and Figure 52 shows what happens between 599 and 601 nanoseconds.

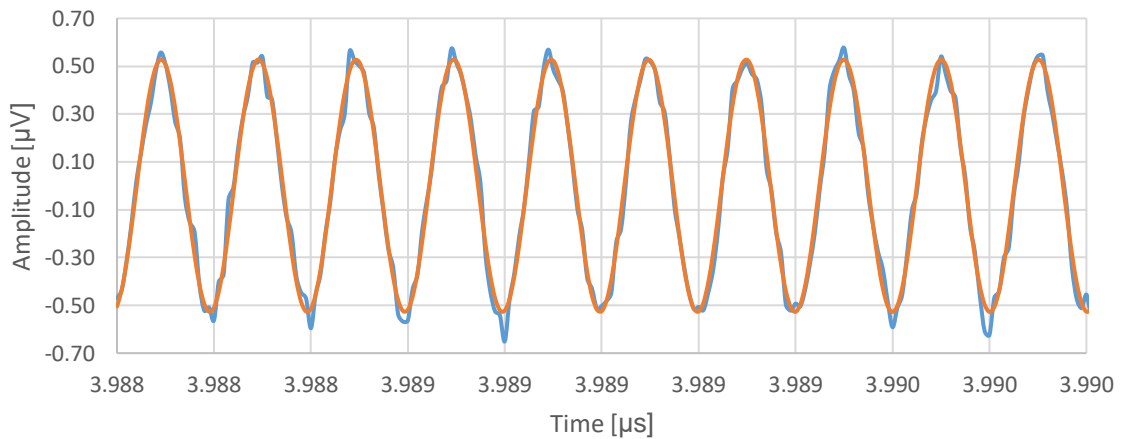


Figure 51: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is ω_o

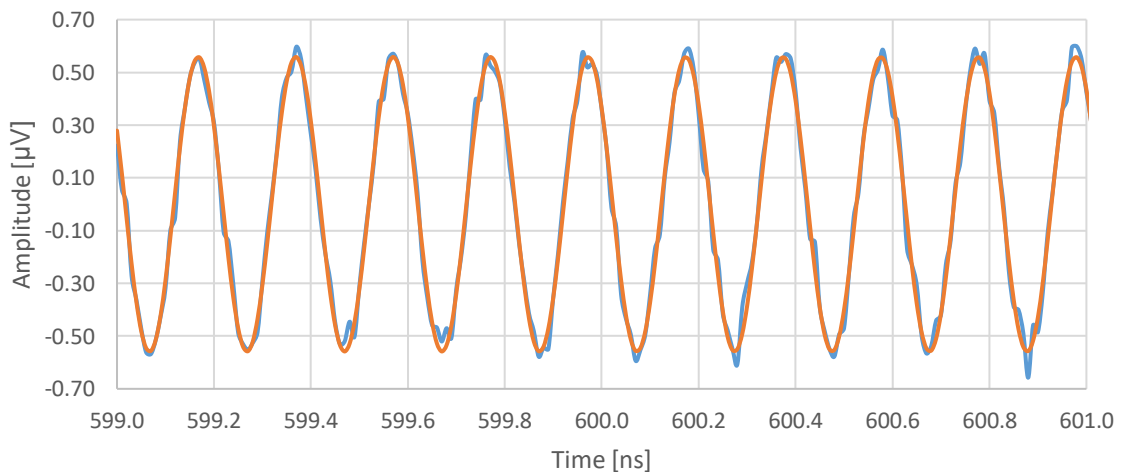


Figure 52: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 and 601 ns when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is ω_o

The effect of the added noise can still be faintly appreciated but compared to the noiseless output signal it is low. This simulation results in a signal-to-noise ratio of $SNR = 8.291 \text{ dB}$.

The second simulation will set the amplitude to $V_g = 1.076 \mu V$ and $f_g \approx 4.9706 GHz$. Which is the amplitude and frequency correspondent to $\omega_g = \omega_o$. The state of the qubit will be set to $|1\rangle$. Figure 53 shows that the time needed for the resonator to reach the steady-state is around 2 microseconds. The amount of added noise can go up to 100 nanoseconds as can be seen in Figure 54 when compared to Figure 53.

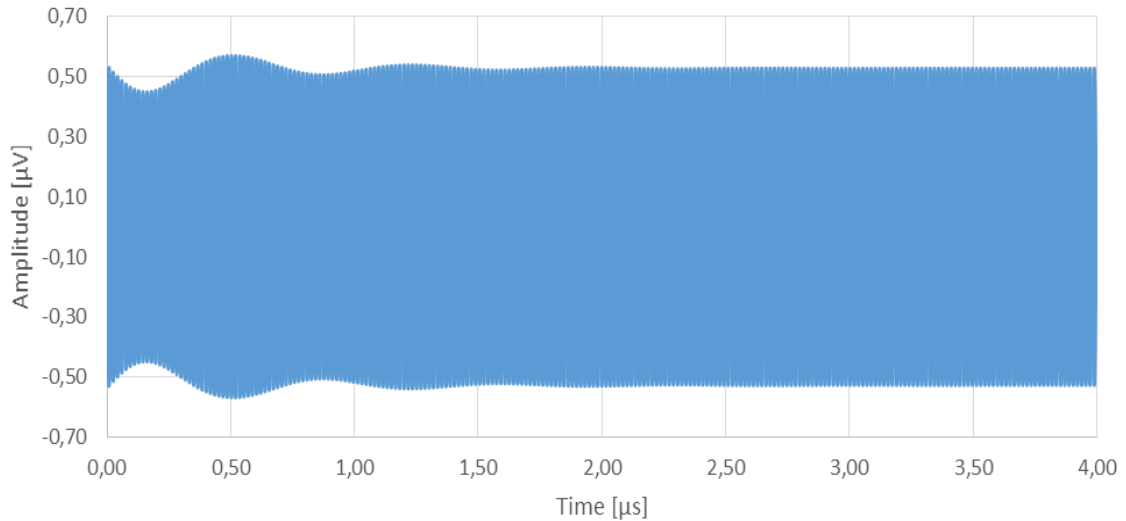


Figure 53: Noiseless output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o

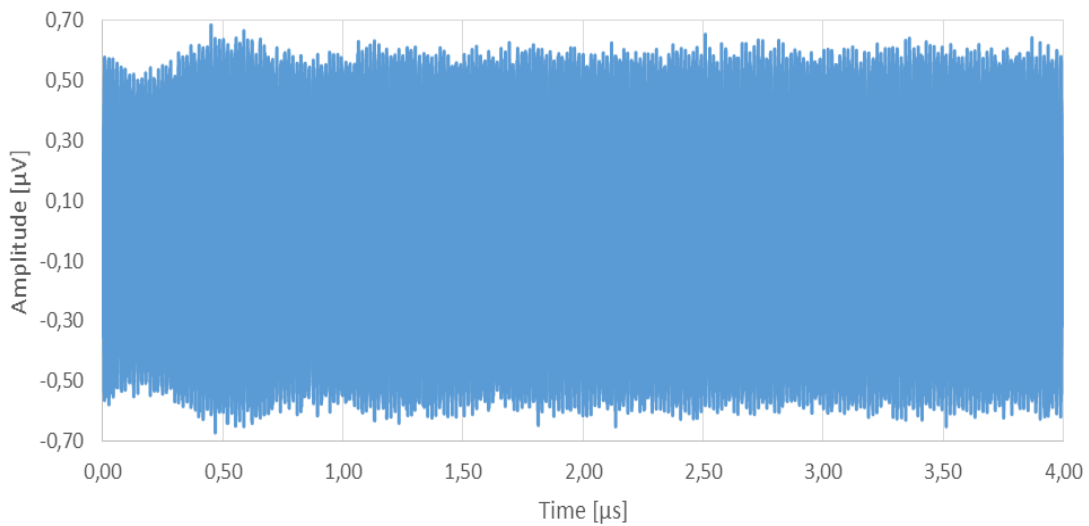


Figure 54: Noisy output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o

Now a zoomed in graph of the signals between 3.988 and 3.99 microseconds is provided in Figure 55. The orange curve is the output before noise and the blue curve is the output after noise. By comparing the orange curve to Figure 35 it is possible to appreciate that both the transient state readout and the steady-state readout testbenches show the same behavior when approaching the steady-state of the system.

Since readout pulses are shorter in duration than 4 microseconds it is also of interest to study how the signal behaves at those time stamps. The duration of the readout pulse for this project is set to 600 nanoseconds and Figure 56 shows what happens between 599 and 601 nanoseconds.

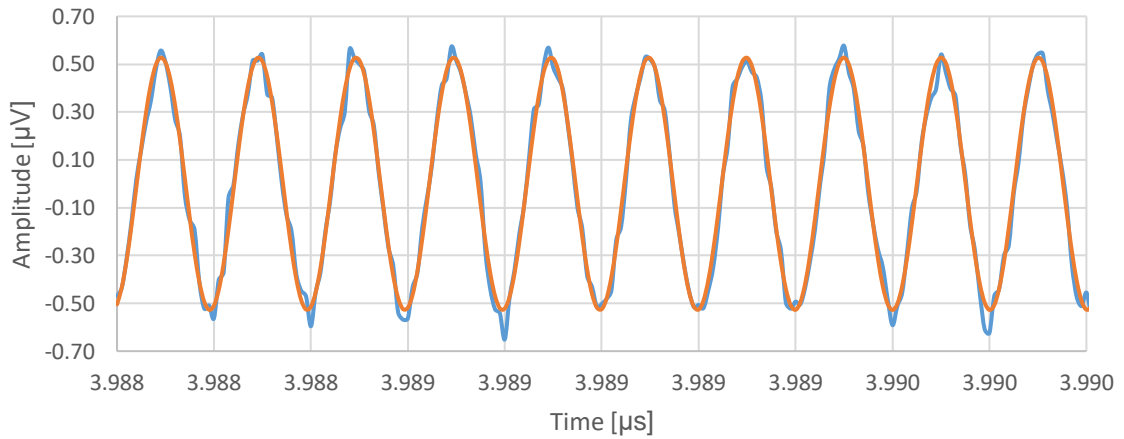


Figure 55: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o

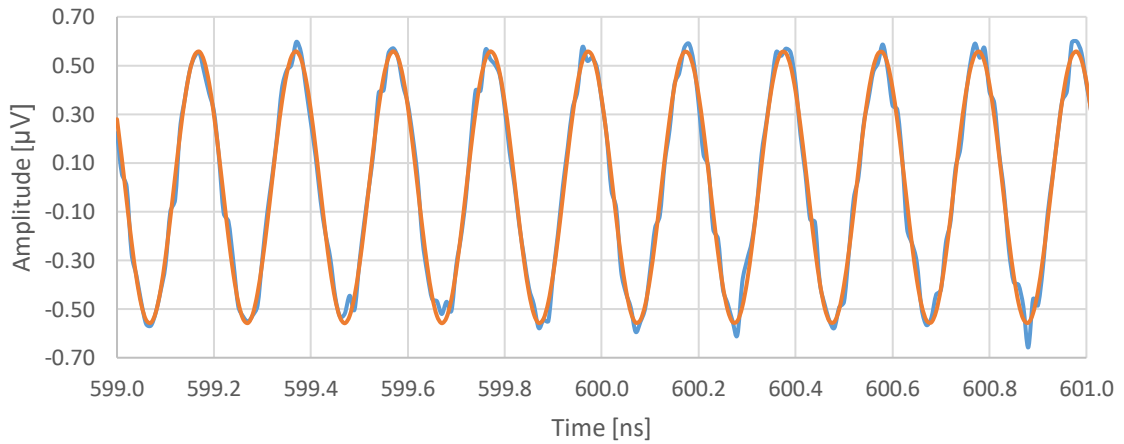


Figure 56: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 ns and 601 ns when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o

As it was expected, states $|1\rangle$ and $|0\rangle$ display symmetrical respect to ω_o since the dispersive shift is equal to either side. The maximum allowed amplitude at the voltage source for this pair of simulations is $V_g = 1.076 \mu V$, which is the maximum allowed for either state due to the mentioned symmetry. Performing the readout at this ω_g maximizes the SNR of the system.

5.3.3 Simulations when $\omega_g = \omega_o^{11}$

The first simulation will set the amplitude to $V_g = 0.2172 \mu V$ and $f_g \approx 4.96920 GHz$. Which is the amplitude and frequency correspondent to $\omega_g = \omega_o^{11} = \omega_o - \chi$. The state of the qubit will be set to $|1\rangle$. Figure 57 shows how the resonator absorbs all the signal once the steady-state has been reached at around 3 microseconds. This result is expected due to the S_{21} parameter of the circuit when the qubit is in state $|1\rangle$ (grey curve in Figure 40). Once the signal has gone through the amplification chain only the added noise is present as can be seen in Figure 58.

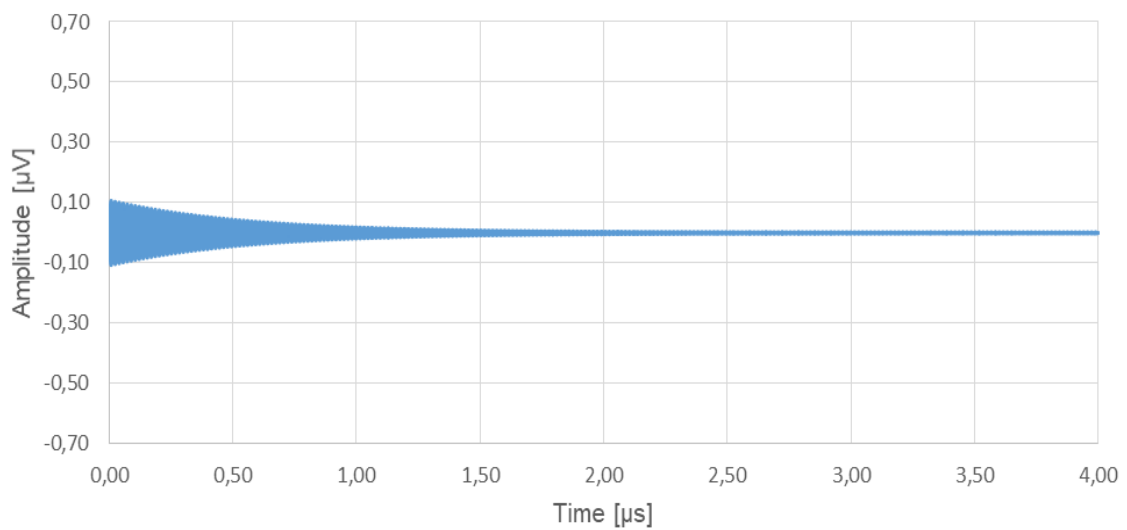


Figure 57: Noiseless output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o^{11}

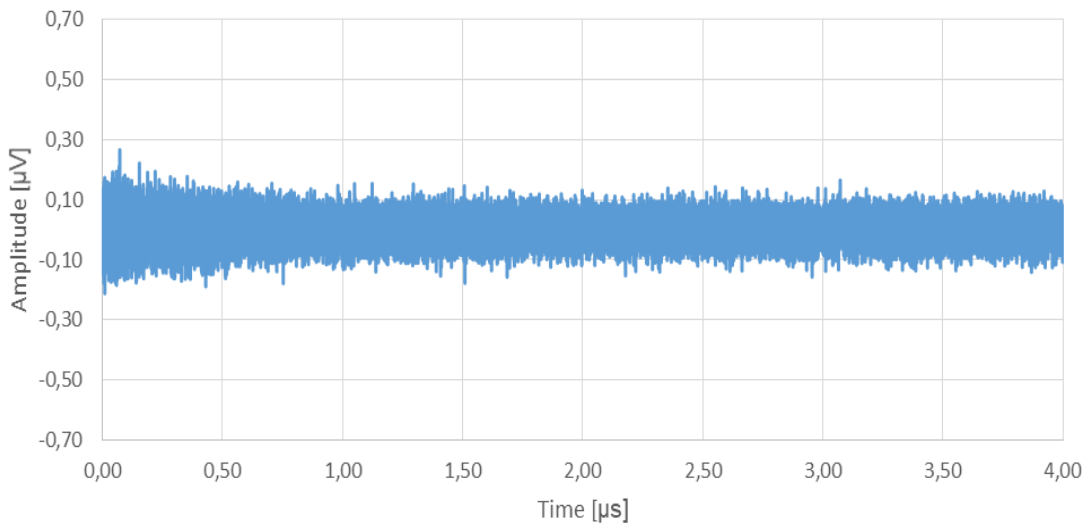


Figure 58: Noisy output V_o when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is ω_o^{11}

Now a zoomed in graph of the signals between 3.988 and 3.99 microseconds is provided in Figure 59. The orange curve is the output before noise and the blue curve is the output after noise. By comparing the orange curve to Figure 31 it is possible to appreciate that both the transient state readout and the steady-state readout testbenches show the same behavior when approaching the steady-state of the system.

Since readout pulses are shorter in duration than 4 microseconds it is also of interest to study how the signal behaves at those time stamps. The duration of the readout pulse for this project is set to 600 nanoseconds and Figure 60 shows what happens between 599 and 601 nanoseconds.

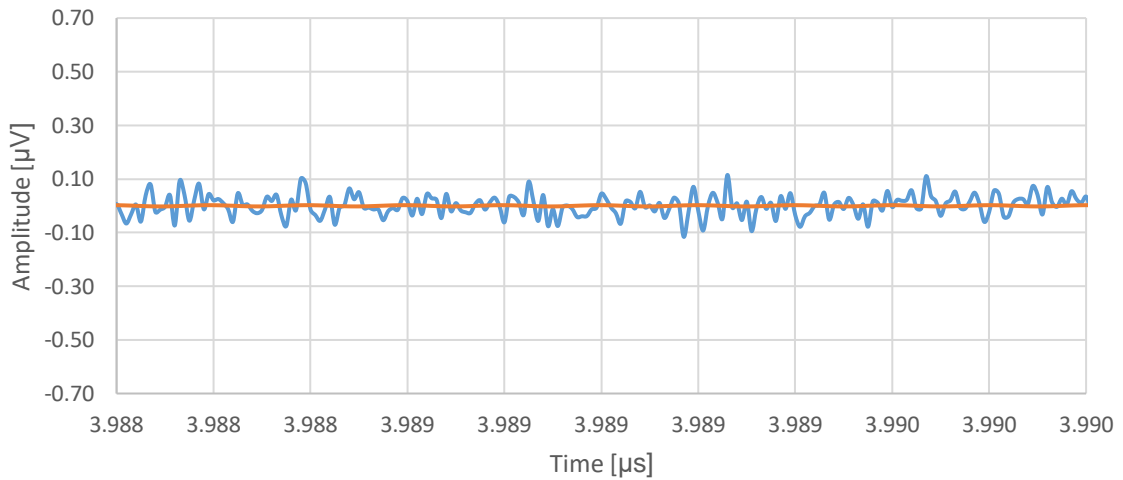


Figure 59: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

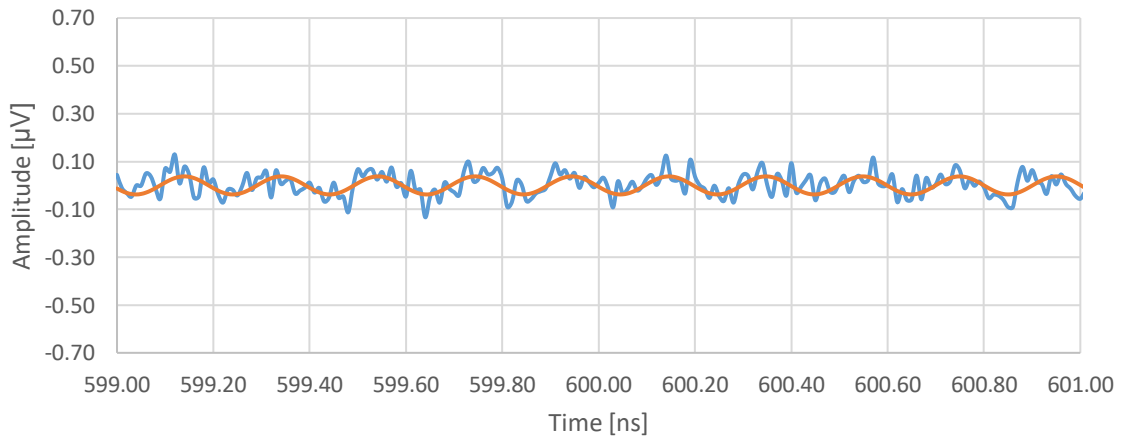


Figure 60: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 and 601 ns when the qubit is in state $|1\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

As it can be seen the energy of the signal has not been completely absorbed by the resonator but it is very low compared to the noise to obtain any valid reading using this setup.

The second simulation will set the amplitude to $V_g = 0.2172 \mu V$ and $f_g \approx 4.96920 GHz$. Which is the amplitude and frequency correspondent to $\omega_g = \omega_o^{(1)} = \omega_o - \chi$. The state of the qubit will be set to $|0\rangle$. Figure 61 shows that the time needed for the resonator to reach the steady-state is around 2 microseconds. Since the frequency of the probing signal is far away from the resonant peak of the resonator, only a small portion of the signal gets absorbed. The amount of added noise can go up to 100 nanoseconds as can be seen in Figure 62 when compared to Figure 61.

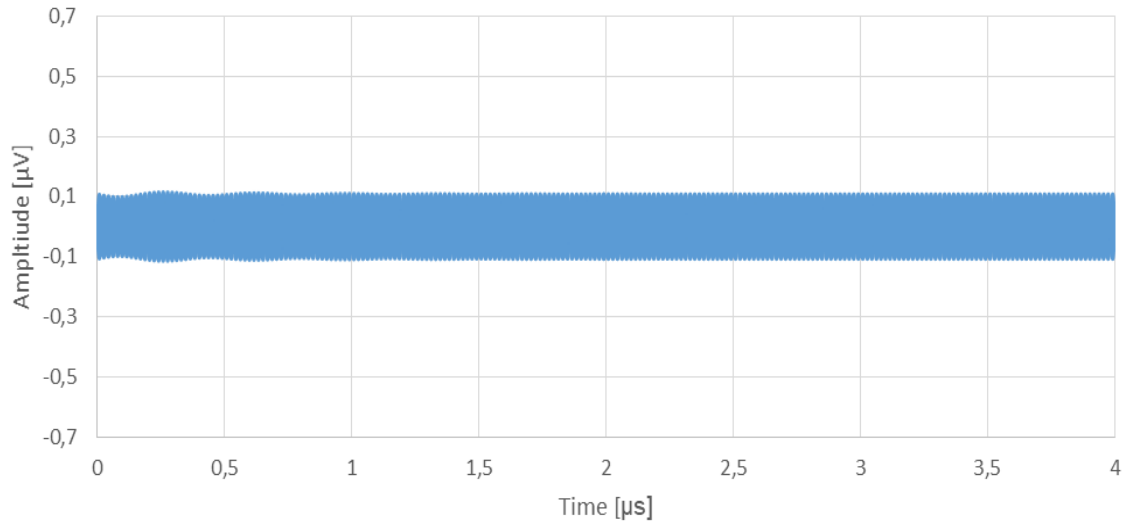


Figure 61: Noiseless output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

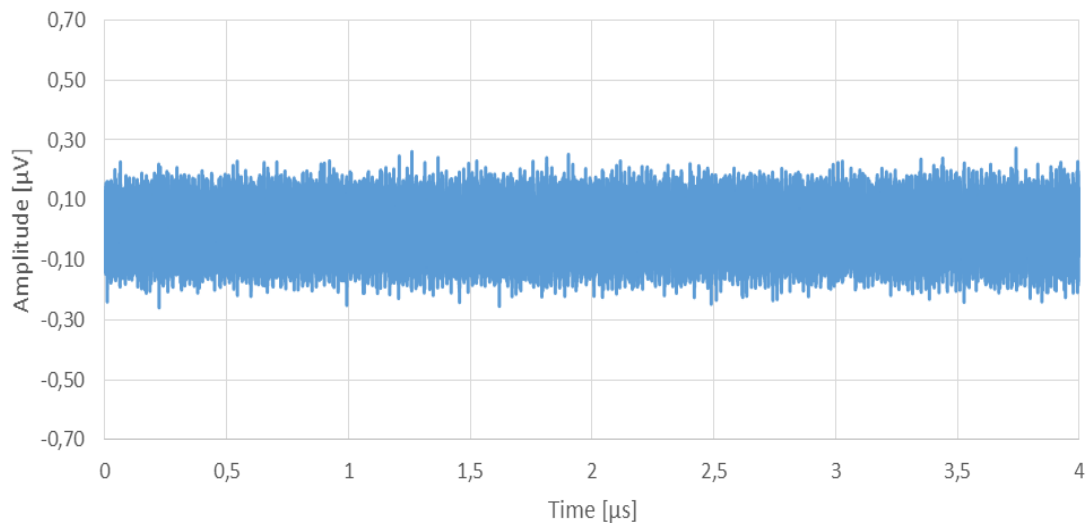


Figure 62: Noisy output V_o when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

Analyzing in Figure 62 the noisy output of the readout, the emulated added noise that would be produced by the amplification chain distorts the output making it difficult to appreciate the ring-up effect of the resonator, the simulation used a noise figure of $NF = 0.02 dB$.

The effect of the emulated noise can be appreciated when performing a temporal zoom. Figure 63 shows the both noiseless and noisy outputs between 3.988 and 3.990 microseconds. As happened in the previous simulation, near the steady-state of the system the signals behave as expected. Comparing them to Figures 33 and 34 the results match accordingly.

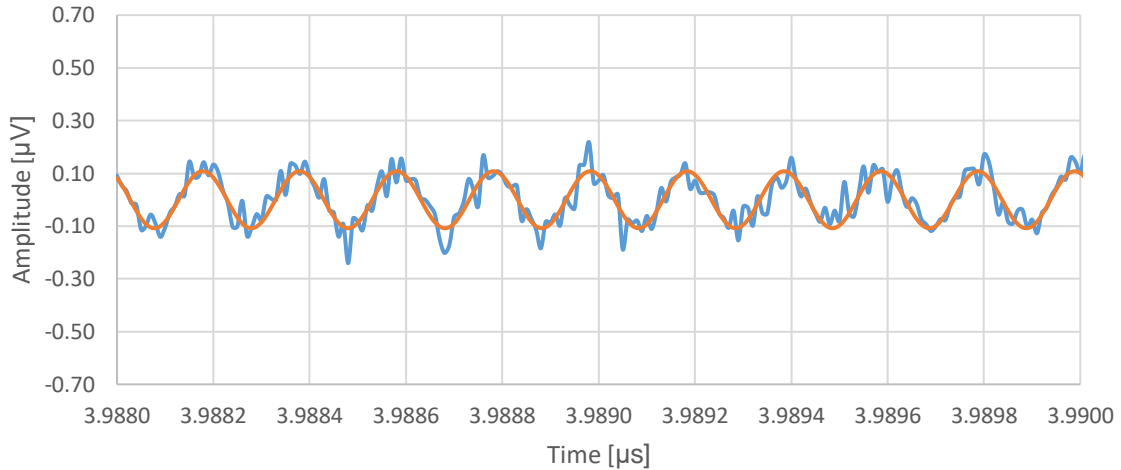


Figure 63: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o near the steady-state when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

In Figure 64 it can be seen that the ring-up effect at 600 nanoseconds has similar amplitude levels as when the system approaches the steady-state. Figure 64 shows a detailed image between 599 and 601 nanoseconds.

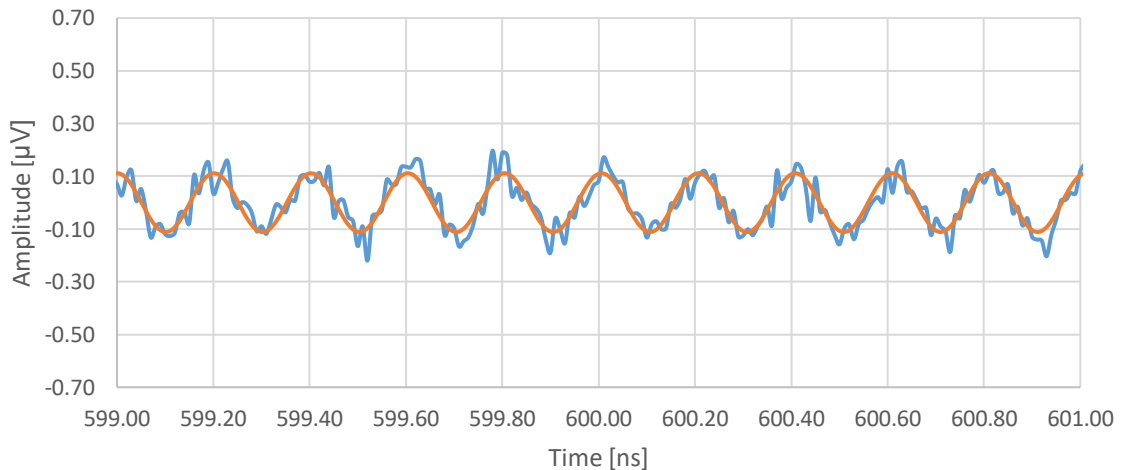


Figure 64: Zoomed-in detail of the noisy (blue) and noiseless (orange) output V_o between 599 ns and 601 ns when the qubit is in state $|0\rangle$ and the angular frequency of the voltage source is $\omega_o^{(1)}$

The effect of the added noise produced when emulating the amplification chain has a significant impact in the readout process, since the maximum allowed amplitude is $V_g = 0.2172 \mu V$. This translates to a low signal-to-noise ratio and the results obtained are similar to the ones obtained in section 5.3.1.

5.3.4 Time delay between states

This section will study the time delay between both states of the qubit, this will be done for the simulations in section 5.3.1, section 5.3.2 and section 5.3.3. All three situations are represented in Figure 65.

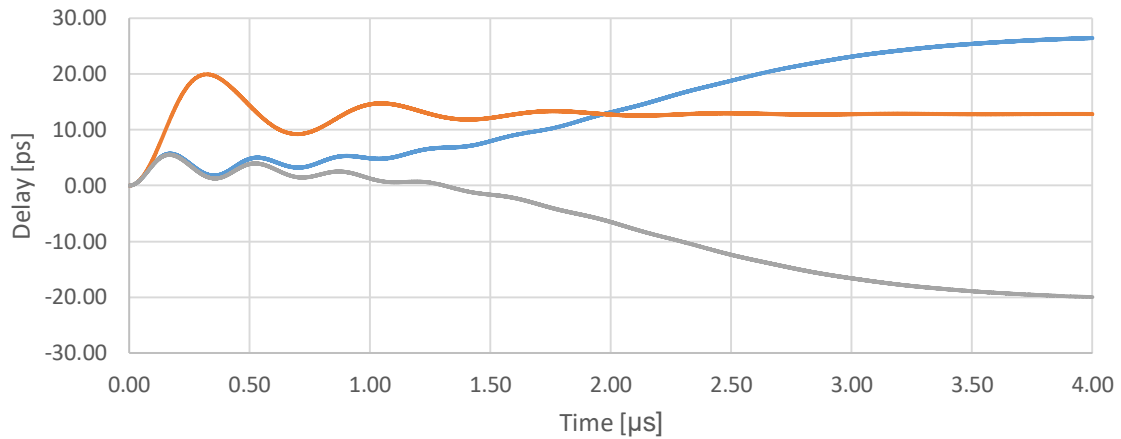


Figure 65: Time delay between the probing signals when the state of the qubit is changed while maintaining the angular frequency of the voltage source: ω_o^{11} (blue curve), ω_o^{00} (grey curve) and ω_o (orange curve).

By analyzing these curves the reader may notice that the ring-up effect of the resonator is present. As the simulations done until this point, Figure 65 shows that the resonator reaches a steady-state at around 2 microseconds. At this time stamp, the ω_o curve (orange) stabilizes and the time delay between states stays stable at around 13.5 picoseconds. The ω_o^{11} and ω_o^{00} time delay curves (blue and grey, respectively) keep increasing in magnitude due to how one of the signals tends to zero since the angular frequency of the voltage source is located at the resonant peak of the qubit in state $|1\rangle$ and state $|0\rangle$ respectively.

6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

At the end of this project, all the proposed objectives have been achieved. A series of testbenches in Cadence Virtuoso for the verification of the readout and control analog electronics involving transmon qubits have been developed. Understanding and some degree of expertise of the Verilog-A programming language has been obtained in order to build the custom analog blocks needed to perform the logic behind the simulations and the arithmetic operations.

These testbenches will provide future users and third parties information about how the control and readout signals behave while allowing them to modify parameters of the qubit, the resonator or the noise and voltage sources. Aside of the signals additional parameters such as the signal-to-noise ratio of the system, the S_{21} parameter of the circuit or the maximum allowed amplitude at the voltage source are also available.

The testbench for the control of the qubit is flexible as it permits the user to change between different waveforms for the input as well as making it possible to perform parametric analyses sweeping variables to study how the state of the qubit changes depending on the amplitude and the duration of the driving pulses.

As has been described in the design of the project, the testbench for the readout of the qubit has been divided into two different models, one of them studying what happens when the system reaches the steady-state and the second one studying what happens when the system reaches the transient state.

The steady-state readout works without any signal inputs, all the parameters needed for the outputs and the processing are set by the user. Once simulated the user can obtain the expected output signal both in its noiseless and noisy versions. The signal-to-noise ratio, fidelity of the readout, S_{21} parameter and allowed voltage source amplitude are also simulated.

The transient readout simulates the RLC parallel representation of the resonator and needs an input signal. This signal is created by a sinusoidal voltage source whose parameters can be modified by the user. To obtain the parameters for the voltage source the user has to run a simulation of short duration setting the transmon's and the resonator's parameters. Once this simulation has been run the user obtains the allowed amplitude at the voltage source as well as the angular resonant frequency of the resonator when coupled to the readout transmission line and the shift induced by the dispersive shift effect produced by the qubit state. When all the parameters have been updated a second simulation of the user's desired duration can be run to obtain both the noisy and noiseless versions of the output.

6.2 Future work

Even though the objectives of the project have been accomplished there is room for improvement within the solutions provided.

- The testbench for the control of the qubit uses a custom Verilog-A model for the input signal using a Gaussian envelope. If a future user would like to change this waveform they would need to design or find a way to implement the waveform they want to use.
- The transient state readout requires of two different simulations to obtain the desired results. The user needs to be informed beforehand of how this testbench needs to be operated. Finding a way to improve this should be possible.
- The default tolerances of Cadence Virtuoso need to be stricter in order for the transient state testbench to work correctly. If these tolerance are not changed the results obtained are not correct. Changing these tolerances increase the simulation time drastically.
- Since the design implemented consists of a single qubit it would be wise to escalate the blocks in such a way that it would allow for simulations consisting of several qubits.
- With these testbenches operating properly it is interesting to try to design actual electronics for readout and control of qubits at cryogenic temperatures like the ones seen in Figures 10 and 11.

7 BUDGET

In this section the total cost of the implementation of the project will be broken down. These costs will be divided in: hardware costs, software costs, human resources and total budget.

| Hardware costs | | | | | |
|-----------------------|-----------|---------------------------|------------------------------|-------------------|------------------|
| Equipment | Price (€) | Amortization time (years) | Amortization installment (€) | Use time (months) | Amortization (€) |
| Computer | 1200 | 2 | 50 | 3 | 150 |
| Total | | | | | 150 € |

Table 3: Hardware costs

| Software costs | | | | | |
|--------------------------|-----------|---------------------------|------------------------------|-------------------|------------------|
| Software | Price (€) | Amortization time (years) | Amortization installment (€) | Use time (months) | Amortization (€) |
| Cadence Virtuoso | 12.000 | 1 | 1.000 | 3 | 3.000 |
| Microsoft Office 2013 | 300 | 1 | 25 | 3 | 75 |
| Linux CentOS 7 | - | - | - | 3 | - |
| Microsoft Windows 10 Pro | 120 | 1 | 12 | 3 | 36 |
| Total | | | | | 3.111 € |

Table 4: Software costs

| Human resources | | | |
|------------------------|----------------|---------------------|-------------------|
| Personnel | Work hours (h) | Cost per hour (€/h) | Total cost (€) |
| Junior engineer | 300 | 40 | 12.000 |
| Senior engineer | 50 | 90 | 4.500 |
| Total | | | 16.500 (€) |

Table 5: Human resources

| Total costs | |
|--------------------|-----------------|
| Hardware | 150 € |
| Software | 3.111 € |
| Human resources | 16.500 € |
| Total | 19.761 € |

Table 6: Total costs

8 REFERENCES

- [1] Cadence. *Standards and Languages*. [Online; accessed 27-August-2023]. URL: https://www.cadence.com/en_US/home/alliances/standards-and-languages.html#Verilog
- [2] I. Chuang, M. Nielsen. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 2010. URL: <https://www.cambridge.org/highereducation/books/quantum-computation-and-quantum-information/01E10196D0A682A6AEFFEA52D53BE9AE#overview>
- [3] TUDelft. *4.2.1 The transmon qubit - TU Delft OCW*. [Online; accessed 27-August-2023]. URL: <https://ocw.tudelft.nl/course-lectures/4-2-1-the-transmon-qubit/>
- [4] Thomas E. Roth, Ruichao Ma, Weng C. Chew. *An Introduction to the Transmon Qubit for Electromagnetic Engineers*. 2021. URL: <https://arxiv.org/abs/2106.11352>
- [5] Philip Krantz. *A Quantum's Engineer's Guide to Superconducting Qubits*. 2021. URL: <https://arxiv.org/abs/1904.06560>
- [6] Yvonne Y. Gao, Chen Wang. *A Practical Guide for Building Superconducting Quantum Devices*. 2021. URL: <https://arxiv.org/abs/2106.06173>
- [7] Ivo S. Mihov, Nikolay V. Vitanov. *Pulse shape effects in qubit dynamics demonstrated on an IBM quantum computer*. 2023. URL: <https://arxiv.org/abs/2301.10004>
- [8] David J. Hover. *Dispersive Readout of a Superconducting Qubit Using a SLUG Amplifier*. 2013. URL: <https://mcdermottgroup.physics.wisc.edu/pdfs/Hover.pdf>
- [9] Alexander Cronheim. *A Circuit Lagrangian Formulation of Opto-mechanical Coupling between two Electrical Resonators mediated by a SQUID*. 2018. URL: <https://repository.tudelft.nl/islandora/object/uuid%3Aa4c72663-65c9-4857-8ffa-ebaf2cbc9782>
- [10] Danilo Labranca, Leonardo Bianchi. *First design of a superconducting qubit for the QUB-IT experiment*. 2022. URL: <https://arxiv.org/abs/2207.09290>

[11] Gabriel Popkin. *Quest for qubits*. [Online; accessed 29-August-2023] URL: <https://www.science.org/doi/10.1126/science.354.6316.1090>

[12] Scientific American. *What are Josephson junctions? How do they work?*. [Online; accessed 29-August-2023] URL: <https://www.scientificamerican.com/article/what-are-josephson-juncti/>

[13] Joseph C. Bardin. *Design and Characterization of a 28-nm Bulk-CMOS Cryogenic Quantum Controller Dissipating Less Than 2 mW at 3 K*. 2019. URL: https://www.researchgate.net/publication/336454314_Design_and_Characterization_of_a_28-nm_Bulk-CMOS_Cryogenic_Quantum_Controller_Dissipating_Less_Than_2_mW_at_3_K

9 APPENDIXES

A. Qubit control code

```

1 // VerilogA for Prueba_1, Qubit_Control, veriloga
2 `include "constants.vams"
3 `include "disciplines.vams"
4 nature dimensionless
5     units = " ";
6     access = DS;
7     abstol = lp;
8     idt_nature = Charge;
9 endnature
10 discipline adimensional
11     domain continuous;
12     potential dimensionless;
13 enddiscipline
14
15 module Qubit_Control(INPUT,OUTPUT);
16     input INPUT; electrical INPUT;
17     output OUTPUT; adimensional OUTPUT;
18     real sigma;
19     real q_zpf; //Zero-point charge fluctuations
20     parameter real L = 8e-9; //Qubit's inductance
21     parameter real C = 80e-15; //Qubit's capacitance
22     parameter real Cd = 30e-18; //TxLine capacitance
23     real Ce; //Total capacitance to ground
24     real Z; //Impedance of the circuit to ground
25     real barred_planck = 1.054e-34; //Reduced planck constant
26     //Variables used to obtain the envelope
27     real old = 0.000, med = 0.000, new = 0.000, maximo = 0.000;
28     //Amplitude of the baseband envelope
29     parameter real V_o = 0;
30
31
32     analog begin
33         //Obtaining the envelope
34         old = med;
35         med = new;
36         new = V(INPUT);
37         if (med > new && med > old) begin
38             maximo = med;
39         end
40         //Obtaining the total capacitance
41         Z = sqrt(L/C);
42         //Calculation of sigma
43         q_zpf = sqrt(barred_planck/(2*Z));
44         Ce = C + Cd;
45         sigma = (Cd/Ce)*q_zpf;
46         //Assignment of the signal generated to the output
47         DS(OUTPUT) <+ sigma/barred_planck*V_o*idt(maximo,0);
48     end
49 endmodule

```

B. Qubit steady-state readout code

```
1 // VerilogA for ReadoutDeQubit, Qubit_Readout, veriloga
2 `include "constants.vams"
3 `include "disciplines.vams"
4 nature decibel
5     units = "dB";
6     access = dB;
7     abstol = 1u;
8     idt_nature = Charge;
9 endnature
10 nature dimensionless
11     units = " ";
12     access = DS;
13     abstol = 1p;
14     idt_nature = Charge;
15 endnature
16 discipline decibelios
17     domain continuous;
18     potential decibel;
19 enddiscipline
20 discipline adimensional
21     domain continuous;
22     potential dimensionless;
23 enddiscipline
24
25 module readoutEstatico( Vo_0, Vo_1, SNR, Vgmax,
26 S21_0, S21_1, Vo, Fidelity, Vo_Noisy);
27
28     output Vo_0, Vo_1, SNR, Vgmax, S21_0, S21_1,
29     Vo, Fidelity, Vo_Noisy;
30     electrical Vo_0, Vo_1, Vgmax, Vo, Vo_Noisy;
31     decibelios SNR;
32     adimensional S21_0, S21_1, Fidelity;
33
34     // Transmon parameters
35     parameter real Lj = 1.10e-8; //JJ Inductance
36     parameter real Cs = 98.18e-15; //Capacitance
37     real Ec, Ej; //Capacitive and inductive energies
38     //0->1 and 1->2 transition angular frequencies
39     real w_01, w_12;
40     //Ratio of gate capacitance to total capacitance
41     //and RMS zero-point voltage of the resonator
42     real beta, Vrms;
43     //Qubit-resonator 0->1 and 1->2 coupling strengths
44     real g_01, g_12;
45     //Dispersive shifts (total, 0->1 and 1->2)
46     real X, X_01, X_12;
47     //Qubit state chosen by the user
48     parameter real Choose_State = 0;
49
50     // Coupling parameters
```

```

51 //TxLine-Resonator coupling
52 parameter real Ck = 8.62e-15;
53 //Resonator-Transmon coupling
54 parameter real Cg = 4.4e-15;
55
56 // Input parameters
57 parameter real Ro = 50; //Resistors' impedance
58 //Characteristic impedance of the TxLine
59 parameter real Zo = 50;
60 parameter real NF = 0.1; //Noise figure [dB]
61 //Duration of the pulse in seconds
62 parameter real pulseTime = 600e-9;
63 //User input to move w_g between the w of state 1 or 0
64 parameter real range = 0 from [-1:1];
65
66 // Resonator parameters
67 parameter real w_r = 31.5e9; //Resonant angular frequency
68 parameter real len = 5.76; //Physical length
69 parameter real Zo_prime = 50; //Characteristic impedance
70 parameter real alpha = 2.4e-4; //Loss factor
71 //Capacitance, inductance and resistance
72 //of the RLC representation
73 real Cr, Lr, R;
74 real k; //Resonance width
75 real Ql; //Quality factor
76 // Constants
77 real e = 1.60217663e-19; //Electron charge
78 real h_bar = 1.054571817e-34; //Reduced planck
79 real boltzmann = 1.380649e-23; //Boltzmann's
80 // Private parameters
81 real w_o; //Resonator's adjusted angular frequency
82 real detuning; //Cavity-Qubit detuning
83 //Critical number of photons in the resonator
84 real ncrit;
85 //Voltage in module necessary at the
86 //generator depending on the qubit's state
87 real modVg_sq_0, modVg_sq_1;
88 real modVg; //Square root of Vg's module
89 //Vo voltage in module depending on the qubit's state
90 real modVo_0, modVo_1;
91 //Phases of Vo depending on the qubit's state
92 real ph_0, ph_1;
93 real B; //Bandwidth for system noise
94 real nsys; //System noise
95 real w_g; //Source's angular frequency
96 //Difference between w|0/1> and w_o
97 real delta_omega_0, delta_omega_1, delta_omega;
98 //Effective and reference temperatures
99 real Teff, Tref = 290;
100 //Difference in module between Vo|0> and Vo|1>
101 real Vsig;
102 real sigma; //Noise's standard deviation
103 real snr; //Signal to Noise ratio

```

```

104 //Pseudorandom sequence generated to represent noise
105 real randnum;
106 //Seed needed to generate the pseudorandom sequence
107 integer seed;
108 real R_star, C_star; //Norton equivalents
109 //Qubit's state |0> and |1> angular frequencies
110 real w_dirac0, w_dirac1;
111 //Qubit's angular frequency based on chosen state
112 real w_dirac;
113 real s21_0, s21_1; //S21 parameters of states |0> and |1>
114 //Variables used to calculate the fidelity
115 real erf, senal = 0, fidelity, j;
116 //Variables used to calculate the fidelity
117 real relaxationTime = 2.62e-5;
118 integer flag = 0;
119 //Variables used to calculate the fidelity
120 real salida; //Output signal Vo
121
122 analog begin
123 // Calculation of the resonator's LC
124 Cr = `M_PI / (4 * w_r * Zo_prime);
125 Lr = 1 / (Cr * pow(w_r,2));
126 // Calculation of the transmon's energies
127 Ec = pow(e,2) / (2 * (Cg + Cs));
128 Ej = pow(h_bar,2) / (4 * pow(e,2) * Lj);
129 // Calculation of the transmon's transition
130 // frequencies [w_ij / (2*pi*1e9)]
131 w_01 = (sqrt(8*Ec*Ej) - Ec) /
132 (h_bar * 2 * `M_PI * 1e9);
133 w_12 = (sqrt(8*Ec*Ej) - 2*Ec) /
134 (h_bar * 2 * `M_PI * 1e9);
135 // Calculation of resonator's quality factor
136 w_o = 1/sqrt(Lr*(Cr+Ck));
137 R = Zo_prime*1000/len/alpha;
138 R_star=(1+w_o*w_o*Ck*Ck*Ro*Ro*0.5*0.5)
139 / (w_o*w_o * Ck*Ck * Ro * 0.5);
140 C_star = Ck/(1+w_o*w_o*Ck*Ck*Ro*Ro*0.5*0.5);
141 Ql = w_o*(C_star+Cr)/(1/R_star + 1/R);
142 // Detuning
143 detuning = (w_o - (w_01 * 1e9 * `M_TWO_PI));
144 // Calculation of coupling strength
145 beta = Cg / (Cg + Cs);
146 Vrms = sqrt(h_bar*w_r/(2*Cr));
147 g_01 = 2*beta*e*Vrms*sqrt(sqrt(Ej/
148 (32*Ec)))/h_bar;
149 g_12 = sqrt(2)*2*beta*e*
150 Vrms*sqrt(sqrt(Ej/(32*Ec)))/h_bar;
151 // Calculation of dispersive shifts
152 X_01 = pow(g_01,2) /
153 (w_01* 2 * `M_PI * 1e9 - w_r);
154 X_12 = pow(g_12,2) /
155 (w_12* 2 * `M_PI * 1e9 - w_r);
156 X = X_01 - X_12/2;

```

```

157 // Calculation of resonance width
158 k = w_o / Ql;
159 // Selection of w_dirac and
160 //calculation of delta omega
161 w_dirac0 = w_o-X;
162 w_dirac1 = w_o+X;
163 w_g = w_o + range*X;
164 if (Choose_State == 0)
165 w_dirac = w_dirac0;
166 else
167 w_dirac = w_dirac1;
168 delta_omega = w_g - w_dirac;
169 delta_omega_0 = w_g - w_dirac0;
170 delta_omega_1 = w_g - w_dirac1;
171 // Calculation of the amplitude of Vg
172 ncrit = pow(detuning,2) /4 / pow(g_01,2);
173 modVg_sq_0 = abs(4*h_bar*w_g*k*Ro*ncrit*
174 (1 + pow(2*delta_omega_0/k,2)));
175 modVg_sq_1 = abs(4*h_bar*w_g*k*Ro*ncrit*
176 (1 + pow(2*delta_omega_1/k,2)));
177 modVg = min(sqrt(modVg_sq_0)
178 , sqrt(modVg_sq_1));
179 // Calculation of the amplitude
180 //of Vo and its phases
181 modVo_0 = abs(modVg)*abs(delta_omega_0/k)/
182 sqrt(1+pow(2*delta_omega_0/k,2));
183 modVo_1 = abs(modVg)*abs(delta_omega_1/k)/
184 sqrt(1+pow(2*delta_omega_1/k,2));
185 ph_0 = $atan(k/2/delta_omega_0);
186 ph_1 = $atan(k/2/delta_omega_1);
187 // Calculation of the parameter S21
188 s21_0 = pow(2*modVo_0/modVg,2);
189 s21_1 = pow(2*modVo_1/modVg,2);
190 // Calculation of noise
191 Vsig = sqrt(pow(modVo_0*sin(ph_0)-
192 modVo_1*sin(ph_1),2) + pow(modVo_0*cos(ph_0)-
193 modVo_1*cos(ph_1),2) );
194 B = 1/pulseTime;
195 Teff = Tref*(10**(NF/10)-1);
196 nsys = boltzmann*Teff/h_bar/w_g;
197 sigma = sqrt(Ro*h_bar*w_g*(0.5+nsys)*B);
198 snr = Vsig/2/sigma;
199 // Fidelity
200 if (flag == 0)
201 for (j=0;j<=snr/sqrt(2);j=j+0.0001)
202 senal = senal + exp((-1)*pow(j,2))*0.0001;
203 flag = 1;
204 erf = (2 / sqrt(`M_PI)) * senal;
205 fidelity = exp((-1)*pulseTime/
206 (2*relaxationTime))*erf*100;
207 // Random noise generation
208 @(initial_step) begin
209 seed = 23;

```

```
210     end
211     randnum = $rdist_normal(seed,0,sigma);
212     // Output signal based on chosen state
213     if (Choose_State == 0)
214         salida = modVo_0*cos(w_g*$abstime + ph_0);
215     else
216         salida = modVo_1*cos(w_g*$abstime + ph_1);
217     // Asignation of values/signals
218     // to the respective nodes
219     V(Vo) <+ salida;
220     dB(SNR) <+ 20*log(snr);
221     V(Vgmax) <+ modVg;
222     DS(S21_0) <+ s21_0;
223     DS(S21_1) <+ s21_1;
224     V(Vo_0) <+ modVo_0;
225     V(Vo_1) <+ modVo_1;
226     DS(Fidelity) <+ fidelity;
227     V(Vo_Noisy) <+ salida+randnum;
228     end
229 endmodule
```


C. Qubit transient state readout code

```

1 // VerilogA for ReadoutDeQubit, transientSNR_Step2, veriloga
2 `include "constants.vams"
3 `include "disciplines.vams"
4 nature gigas
5     units = "GHz";
6     access = Freq;
7     abstol = 1u;
8     idt_nature = Charge;
9 endnature
10 discipline giga
11     domain continuous;
12     potential gigas;
13 enddiscipline
14 nature megas
15     units = "MHz";
16     access = Freq;
17     abstol = 1u;
18     idt_nature = Charge;
19 endnature
20 discipline mega
21     domain continuous;
22     potential megas;
23 enddiscipline
24 module readoutDinamico( tx, gnd, W_o, Disp_Shift, Vgmax);
25
26     inout tx, Vgmax, gnd; electrical tx, Vgmax, gnd;
27     output W_o, Disp_Shift; giga W_o; mega Disp_Shift;
28     electrical n1;
29     branch (tx,n1) ccap;
30     branch (n1,gnd) res, ind, cap;
31     // Transmon parameters
32     parameter real Lj = 1.10e-8; //JJ Inductance
33     parameter real Cs = 98.18e-15; //Capacitance
34     real Ec, Ej; //Capacitive and inductive energies
35     //Ratio of gate capacitance to total
36     //capacitance and RMS zero-point voltage of the resonator
37     real beta, Vrms;
38     //Qubit-resonator 0->1 and 1->2 coupling strengths
39     real g_01, g_12;
40     real X_01, X_12, X; //Dispersive shifts
41     //0->1 and 1->2 transition angular frequencies
42     real w_01, w_12;
43     //Qubit state chosen by user
44     parameter real Choose_State = 0;
45     // Coupling parameters
46     parameter real Ck = 8.62e-15; //TxLine-Resonator coupling
47     parameter real Cg = 4.4e-15; //Resonator-Transmon coupling
48     // Resonator parameters
49     parameter real w_r = 3e9; //Resonating angular frequency
50     parameter real alpha = 2.4e-4; //Loss factor

```

```

51     parameter real length = 5.76e-3; //Physical length
52     //Capacitance, inductance and resistance values
53     //of the RLC representation
54     real Cr,Lr,R;
55     real Ql; //Quality factor
56     real k; //Resonance width
57     parameter real Zo = 50; //Characteristic impedance
58     // Constants
59     real e = 1.60217663e-19; //Electron charge
60     real h_bar = 1.054571817e-34; //Reduced planck
61     real boltzmann = 1.380649e-23; //Boltzmann's
62     // Inputs
63     parameter real Ro = 50; //Generator and load impedance
64     //User input to move w_g between the w of state 1 or 0
65     parameter real range = 0 from [-1:1];
66     //Characteristic impedance of the TxLine
67     parameter real Zo_prie = 50;
68     // Other variables
69     //New angular frequency after coupling
70     //the resonator to the TxLine
71     real w_o;
72     //Inductive addition to the resonator that
73     //shifts w_o to +X or -X
74     real deltaL;
75     real w_g; //Vg's angular frequency
76     //Norton equivalents for quality factor calculation
77     real R_star, C_star;
78     real detuning; //Cavity-Qubit detuning
79     //Qubit's state |0> and |1> angular frequencies
80     real w_dirac0, w_dirac1;
81     //Qubit's angular frequency based on chosen state
82     real w_dirac;
83     //Difference between w_g and each w_dirac
84     real delta_omega_0, delta_omega, delta_omega_1;
85     real ncrit; //Critical number of photons in the resonator
86     //Voltage in module necessary at the generator for each state
87     real modVg_sq_0, modVg_sq_1;
88     real modVg; //Square root of Vg's module
89
90     analog begin
91         // Calculation of the resonator's LC values
92         Cr = `M_PI / (4 * w_r * Zo);
93         Lr = 1 / (Cr * pow(w_r,2));
94         // Calculation of w_o
95         //(w_r plus coupling effects to the TxLine)
96         w_o = 1/sqrt(Lr*(Cr+Ck));
97         // Calculation of the transmon's
98         //capacitive and inductive energies
99         Ec = pow(e,2) / (2 * (Cg + Cs));
100        Ej = pow(h_bar,2) / (4 * pow(e,2) * Lj);
101        // Calculation of the transmon's
102        //transition angular frequencies
103        w_01 = (sqrt(8*Ec*Ej) - Ec) /

```

```

104      (h_bar * 2 * `M_PI * 1e9);
105      w_12 = (sqrt(8*Ec*Ej) - 2*Ec) /
106      (h_bar * 2 * `M_PI * 1e9);
107      // Calculation of resonator's quality factor
108      R = Zo/(length*alpha);
109      R_star = (1 + w_o*w_o * Ck*Ck * Ro*Ro * 0.5*0.5)/
110      (w_o*w_o * Ck*Ck * Ro * 0.5);
111      C_star = Ck /
112      (1 + w_o*w_o * Ck*Ck * Ro*Ro * 0.5*0.5);
113      Ql = w_o*(C_star+Cr)/(1/R_star + 1/R);
114      // Detuning
115      detuning = (w_o - (w_01 * 1e9 * `M_TWO_PI)); // [GHz]
116      // Calculation of coupling strength
117      beta = Cg / (Cg + Cs);
118      Vrms = sqrt(h_bar*w_r/(2*Cr));
119      g_01 = 2*beta*e*Vrms*sqrt(sqrt(Ej/(32*Ec)))/h_bar;
120      g_12 = sqrt(2)*2*beta*e*Vrms*
121      sqrt(sqrt(Ej/(32*Ec)))/h_bar;
122
123      // Calculation of dispersive shift
124      X_01 = pow(g_01,2) / (w_01* 2 * `M_PI * 1e9 - w_r);
125      X_12 = pow(g_12,2) / (w_12* 2 * `M_PI * 1e9 - w_r);
126      X = X_01 - X_12/2;
127      // Calculation of resonance width
128      k = w_o / Ql;
129      // Selection of w_dirac and calculation of
130      //delta omega based on chosen state
131      w_dirac0 = w_o-X;
132      w_dirac1 = w_o+X;
133      w_g = w_o + range*X;
134      if (Choose_State == 0)
135      w_dirac = w_dirac0;
136      else
137      w_dirac = w_dirac1;
138      delta_omega = w_g - w_dirac;
139      delta_omega_0 = w_g - w_dirac0;
140      delta_omega_1 = w_g - w_dirac1;
141      // Calculation of the amplitude of Vg
142      ncrit = pow(detuning,2) /4 / pow(g_01,2);
143      modVg_sq_0 = abs(4*h_bar*w_g*k*Ro*ncrit*
144      (1 + pow(2*delta_omega_0/k,2)));
145      modVg_sq_1 = abs(4*h_bar*w_g*k*Ro*ncrit*
146      (1 + pow(2*delta_omega_1/k,2)));
147      modVg = min(sqrt(modVg_sq_0),sqrt(modVg_sq_1));
148
149      // Calculation of deltaL based on chosen state
150      if (Choose_State == 0)
151      deltaL = 1 / (pow(w_o-X,2)*(Cr+Ck)) - Lr;
152      else if (Choose_State == 1)
153      deltaL = 1 / (pow(w_o+X,2)*(Cr+Ck)) - Lr;
154      else
155      deltaL = 0;
156      // Asignation of values to the

```

```
157          //respective branches and nodes
158          V(res) <+ R*I(res);
159          I(cap) <+ Cr*ddt(V(cap));
160          V(ind) <+ (Lr+deltaL)*ddt(I(ind));
161          I(ccap) <+ Ck*ddt(V(ccap));
162          Freq(W_o) <+ w_o/2/`M_PI/1e9;
163          Freq(Disp_Shift) <+ X/2/`M_PI/1e6;
164          V(Vgmax) <+ modVg;
165          end
166 endmodule
```

D. Gaussian envelope code

```
1 `include "constants.vams"
2 `include "disciplines.vams"
3 module pulseGaussian (plus,minus);
4
5     inout plus,minus;
6     electrical plus,minus;
7     parameter real sigma=1/2; //Standard deviation
8     parameter real tcentre=50n; //Time shift
9     parameter real amplitude=10; //Pulse's amplitude
10
11     analog begin
12         V(plus,minus) <+ amplitude*
13             exp(-pow($abstime-tcentre)/sigma,2)/2);
14     end
15 endmodule
```

E. Noise source code

```

1 // VerilogA for ReadoutDeQubit, noiseBlock, veriloga
2 `include "constants.vams"
3 `include "disciplines.vams"
4 module noiseBlock(in, out);
5
6     input in; electrical in;
7     output out; electrical out;
8     // Constants
9     real e = 1.60217663e-19; //Electron charge
10    real h_bar = 1.054571817e-34; //Reduced planck
11    real boltzmann = 1.380649e-23; //Boltzmann's
12    // Inputs
13    parameter real NF = 0.02; //Noise figure [dB]
14    //Duration of the pulse in seconds
15    parameter real pulseTime = 600e-9;
16    //Source's angular frequency
17    parameter real w_g = 3.1213e10;
18    parameter real Ro = 50; //TxLine impedance
19    // Other variables
20    real B; //System noise bandwidth
21    real nsys; //System noise
22    //Effective and reference temperatures
23    real Teff, Tref = 290;
24    real sigma; //Noise's standard deviation
25    //Seed needed to generate a pseudorandom sequence
26    integer seed;
27    //Pseudorandom sequence that represents noise
28    real randnum;
29
30    analog begin
31        // Standard deviation calculation
32        B = 1/pulseTime;
33        Teff = Tref*(10**(NF/10)-1);
34        nsys = boltzmann*Teff/h_bar/w_g;
35        sigma = sqrt(Ro*h_bar*w_g*(0.5+nsys)*B);
36        // Pseudorandom sequence generation based
37        // on the standard deviation obtained
38        @(initial_step) begin
39            seed = 23;
40            end
41            randnum = $rdist_normal(seed,0,sigma);
42            // Asignation of values to the
43            // respective branches and nodes
44            I(in) <+ V(in)/Ro;
45            V(out) <+ V(in) + randnum;
46        end
47    endmodule

```