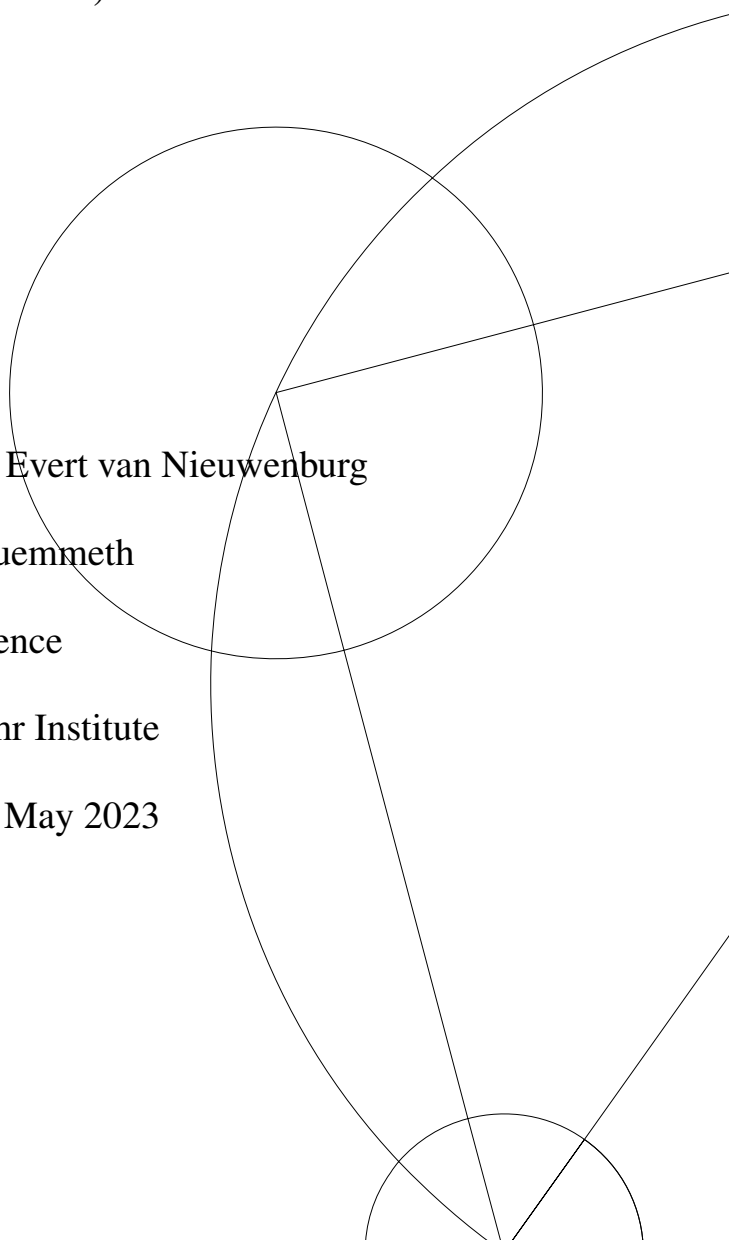
UNIVERSITY OF COPENHAGEN

Automation of tuning strategies for spin qubits

from 2DEG to quantum dot

Lara Lausen (tmn232)



Supervisors: Anasua Chatterjee, Evert van Nieuwenburg

and Ferdinand Kuemmeth

Faculty: Science

Institute: Niels Bohr Institute

Submitted 22nd of May 2023

ACKNOWLEDGEMENTS

To begin with, I would like to thank those who have supported me during the writing of this master's thesis.

First of all, I would like to thank Evert van Nieuwenburg for taking on the role of my supervisor, despite knowing that you would be moving to Leiden halfway through the project and having busy times ahead. Thank you for always being ready with advice and finding a spot for me in your busy schedule. Thank you as well for making sure I would be in good hands at Qdev after your move. I realized how important it is to have helpful colleagues around to discuss when working on a long-term project.

When I started on my master's thesis, my plan was to do a theoretical project, and I didn't expect that I would be doing experiments and hands-on measurements on a real quantum device. I would like to thank Anasua Chatterjee for taking over the role as my supervisor at Qdev and making sure that I would be part of a project in the spin qubit group. It turned out that I enjoy experimental work just as much as theoretical work. You have supported me not only on the educational side and matters related to the project but also made me think about my future and career by asking the right questions and showing the possibilities that exist. For me, you have not only been a supervisor but also a role model and source of inspiration for women in science.

Ferdinand Kuemmeth, thank you for making it possible to write my master thesis as a part of the spin qubit group.

Torbjørn Rasmussen, thank you for spending so much of your time teaching me so much about the tuning of quantum devices, for always being patient and willing to assist whenever I encountered difficulties. I have learned a lot from you and was impressed so many times by your great knowledge of qubits, tuning strategies, and the laboratory equipment.

Harry Lampadaris, thank you for teaching me about Silicon Germanium and for giving me valuable insight into your own project. I would also like to thank you for the great

discussions and debates on spin qubits and tuning strategies, and most importantly for encouraging me.

To the remaining members of the spin qubit group, it has been a pleasure to be a part of this group and to learn about spin qubits, gaining insight into the world of experimental research.

I would also like to extend my thanks to Justyna Zwolak for allowing me to participate in the cold start project, and to Danielle Middlebrooks for the great collaboration on this project.

ABSTRACT

Spin qubits are a promising candidate for quantum information processing. They are implemented using semiconductor-based quantum dots and hence successful implementation requires the fine-tuning of voltages to the quantum dot regime to ensure an optimal performance. Adjusting and finding the optimal voltage configuration for a quantum device is a challenging task that requires skilled specialists and can consume a considerable amount of time. With the need for scalable quantum technologies it is not efficient to rely solely on human experts, and automation techniques are vital for the further development. This work investigates tuning strategies for spin qubits to operate in the quantum dot regime and how to automate these. The work is divided into four main parts. It starts with providing a comprehensive theoretical background, including the principles of quantum computing and the use of spin qubits as a platform for quantum information processing. Secondly, manual tuning strategies for a specific device are developed and optimized to achieve efficient and reliable operation of the specific spin qubit. Thirdly, an automated tuning algorithm based on the manual tuning strategy is developed and implemented, enabling fast and accurate tuning of the spin qubits, using analysis and fitting techniques. Finally, the thesis concludes with an outlook on future developments in the field.

CONTENTS

Part I

INTRODUCTION

1

INTRODUCTION

Quantum computing has the potential to change computation as we know it today fundamentally.

Full-scale realization hasn't been achieved yet, however, there are different suggestions towards universal quantum computing.

One approach to it, is the use of spin qubits based on gate-defined quantum dots. Spin qubits are quantum bits that store information in the spin state of an electron. Gate-defined quantum dots are created and operated by applying a voltage to metallic gates on a semiconductor surface, such that electrons are confined in small regions. The voltages that have to be applied to the gates need to be identified such that individual electrons are captured and tunnel-coupled, this is referred to as the tuning process. As in all research areas of quantum computation, the challenge is to maintain the coherence of the quantum state which can be disturbed by any interactions with the environment and which will lead to computation errors. Spin qubits are highly sensitive to their tuning parameters, such as the magnetic field and gate voltages applied to the device. Currently, human interaction with the tuning process is still common practice, introducing errors and taking up a lot of time and effort. One of the key factors for achieving the high fidelity required for quantum computation, is to make tuning strategies reliable and scalable by minimizing the need of human interaction. This makes it furthermore possible for researchers and experimenters to use their research efforts in the improvement of qubit quality.

In this thesis, I focus on general tuning strategies for spin qubits based on gate-defined quantum dots to get from the two-dimensional electron gas to a tuned quantum dot and the automation of such a process. I aim to develop an automation code that can streamline the process of tuning and help researchers to get their device into the double dot regime

without or as little as possible human intervention and in much shorter time than it is currently done.

To achieve this, I first give an overview of the theory of spin qubits and the relevant concepts to understand electron transport in semiconductor heterostructures. I will then go into depth with the manual tuning strategies as currently performed in the laboratory at The Center of Quantum Devices in Copenhagen, demonstrated on a specific device that can be used to optimize the performance of spin qubits. Finally, I will show a way of automating the proposed tuning strategies which can be used by researchers to easily tune a quantum device into quantum dot regime.

I demonstrate the effectiveness of the automation code by applying it to the experimental setup that was also used for the manual tuning. This makes it possible to compare the automated results to those obtained using manual tuning. The aim is to provide a more efficient and automated way of tuning spin qubits and thereby accelerate the development of the promising technology of spin qubits.

1.1 CURRENT DEVELOPMENTS IN THE AUTOMATION OF QUBIT TUNING STRATEGIES

As per today it has not been possible to fully remove human input and intervention in the tuning process of a quantum device [1]. As long as the tuning still relies on intuition and individual assessment of researchers and experimenters, errors can be made and computing is not scalable. The automation of the tuning process is therefore crucial for further advances in the field. Numerous efforts have already been made in developing such automation strategies, the following will give an overview of research ongoing in the field of automation an optimization and put the work done in this thesis into perspective.

Baart et al. were one of the first to address the importance of auto-tuning. They developed an algorithm that is able find and set the correct gate voltages in gate-defined semiconductor double quantum dots in GaAs hetero structures in order to tune a device into double quantum dot regime. To set the correct voltage ranges, they use 2D scans of the current-voltage space and use fitting techniques. Their constraint is however, the requirement of prior input of the devices' geometry and pinch-off values of each gate [2].

Several researchers make use of image analysis which has proven to be a very powerful tool in the auto-tuning field [3], [4]. Lapointe-Major for example developed an algorithm that can tune a single quantum dot to the single-electron regime by analyzing a series of images of a charge-stability diagram. Their algorithm first remove the physical background of the charge sensor and then uses image analysis to detect charge transition lines [3].

Additionally, van Diepen et al. focus on the automated tuning of the inter-dot tunnel coupling in a linear array of gate-defined semiconductor quantum dots. They do this by using image processing methods to automatically fit the shape of the capacitive anti-crossings between charge states that can be observed in charge stability diagrams. They then tune the tunnel-coupling by using a feed-back loop adjusting gate voltages until the tunnel coupling converges to a predefined target value [4]. Their algorithm requires a device tuned near an inter-dot charge transition regime, for example obtained by the automation method in [2].

As a natural next step to further optimize and speed up the tuning process, machine learning has become an important topic of interest in the field. Different areas of machine learning have been explored, such as deep reinforcement learning [5], supervised [6], [7] and unsupervised machine learning [8] mainly looking for transport features in two-dimensional current-voltage maps/charge stability diagrams and detecting charge transitions: Moon et al. succeeded in completely automating the tuning of a quantum device with multiple gate electrodes, even without human input or prior knowledge of the device faster than a human expert [9].

Darulová et al. implemented a tuning process for an unknown device as a two-step process. First, they characterize the gates of the device by taking 1D measurements, then they attempt to tune the device into single or double quantum dot regime by using 2D maps and charge stability diagrams. Finally, they assess the quality of the measurement results and charge states by four binary classifiers trained with experimental data. The only input needed for their algorithm is the devices' gate layout, bonding scheme, line mappings, safe gate voltage ranges and the setup specific noise floor [10].

More advances in the use of machine learning for tuning quantum dots were made by Kalantre et al. who developed a model reproducing current vs. gate voltage characteristics observed in experiments which was then used to train DNNs to learn charge states of single quantum dots. Here the trained DNNs were tested on previously taken experimental data

in order to show how the auto-tune technique could be implemented in an experimental setup [11].

Another application of machine learning for tuning double qauntum dots into a desired charge occupation was demonstrated by Durrer et al.. Using a neural network trained by supervised learning, their algorithm detects transition lines in coarsely measured charge stability diagrams. They first determine the voltage configuration for the (0,0) state. From there, the algorithm can find any desired charge-state for the quantum dot [12].

A challenge faced by machine learning approaches, however, is the need for larger datasets of high quality to train an ML agent and such datasets still have to be developed [13], [6].

Furthermore, it can be challenging to obtain reliable data. Low-quality or noisy data can lead to failure of algorithms that otherwise have been proven successful.

Ziegler et al. therefore suggest a robust framework for the auto-tuning of quantum dots that combines a machine learning state classifier with a data quality control module, making sure that only reliable data are processed by the state classifier. Otherwise the device should be re-tuned and analyzed or in the case of very low quality data the tuning process is terminated [14].

In order to face the challenge of making quantum computing scalable, researchers recently have turned their efforts towards the automation of larger systems and multi-dot devices: For example by systematically adding qauntum dots to a system [15] or by dividing a larger system into smaller subsystems that can be controlled individually and combined again to larger quantum dot arrays [16].

Ziegler et al. emphasize the importance of orthogonal control of the chemical potentials of quantum dots in order to tune larger quantum dot arrays reliably and resiliently to noise. They establish control by combining machine learning-based image classification and curve fitting in order to identify the capacitive coupling between dots [17].

Another aspect on the way to scalable quantum computation is the focus on making the measuring process more efficient. Lennon et al. address this by introducing a machine learning algorithm to decide which measurements in a quantum dot to perform next based on information theory [18]. Additionally, Teske et al. demonstrated and algorithm for the automated fine-tuning of quantum dots, where a Kalman filter based on Bayesian statistics

was used to estimate the gradients of the parameters in the multidimensional voltage space, reducing the number of requirement measurements and saving lab time [19].

The localization and analysis of Coulomb peaks is an essential step in the tuning process as it gives information about if and where electron transport occurs. An automated Coulomb peak detection therefore is a vital part of every automation technique. Successful approaches have been made for example by Baart et al., as part of their automated tuning algorithm described above, using image detection. They localize Coulomb oscillations by applying a Gabor filter to a 2d-scan of two gates and then evaluate each Coulomb peak based on its height and slope to find the most sensitive peak [2]. Additionally, Coulomb peak detection plays a vital role in the the algorithm of Moon et al.. It is the starting point of their investigation phase, where they take two gates and chose a random candidate coordinate to define a 2d plane. In this plane they take a diagonal trace of the current measurement which is then examined for Coulomb peaks. Based on if Coulomb peaks are observed, the plane is chosen to be further investigated or not [9].

Also Lennon et al. use object detection with deep learning to find Coulomb diamonds in a 2d map, as these include the relevant features to measure in a charge-stability diagram [18].

It is obvious that the auto-tuning of qubits is an essential aspect of quantum computing with important advances in the recent years.

Currently there are different approaches to an efficient and automated qubit tuning process, including machine learning-based methods and optimization algorithms. There is no question that these advances pave the way toward scalable and reliable quantum computing.

However, there are still many challenges to overcome, including the need for more robust and accurate tuning techniques applicable to multiple qubits. Further research is needed to address these challenges.

Part II

THEORY

2

QUANTUM INFORMATION

2.1 QUBITS

In classical computing the basic unit of information is a bit which can be either 0 or 1, the equivalent in quantum computing is a quantum bit, short qubit. Exploiting the characteristics of quantum computing a qubit can be in any state between 0 and 1 which allows a much greater computational power than a classical computer. Qubits are represented using 2 basis states $|0\rangle$ and $|1\rangle$ which can be identified with the two orthogonal vectors $\binom{1}{0}$ and $\binom{0}{1}$ respectively. The most general qubit state $|\psi\rangle$ can be in any superposition

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \text{where} |\alpha|^2 + |\beta|^2 = 1 \tag{1}$$

By parametrizing this state by angles θ and ϕ, it can be geometrically visualized as a point on the surface of the Bloch sphere, a sphere with radius 1 (unit sphere). Here α and β are given by $\alpha = \cos\frac{\theta}{2}$ and $\beta = e^{i\phi}\sin\frac{\theta}{2}$ where $e^{i\phi}$ is the global phase and $\theta \in [0, \pi]$ and $\phi \in [0, \frac{\pi}{2}]$ [20].

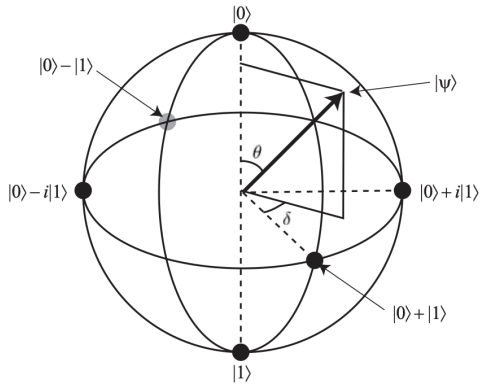


Figure 1: Bloch sphere representation of a qubit. The two qubit basis states are denoted by $|0\rangle$ and $|1\rangle$. Any point on the Bloch sphere can be represented by the state $|\Psi\rangle$ as a superposition of the two states. Taken from [21].

We can now represent all superpositions of $|\psi\rangle$ as a point on the Bloch sphere.

Qubits have been realized in different ways, in the following section the two types important for this thesis will be described.

2.2 SINGLE SPIN QUBITS/LOSS-DIVINCENZO SPIN QUBIT

The first person to specify requirements for a viable implementation of quantum computation was David DiVincenzo [22].

His original criteria were the following:

1. *A scalable physical system with well-characterized qubits:* The physical system must be scalable, meaning that it can be expanded to include many qubits. The qubits must also be well-characterized and easily controllable.

2. *The ability to initialize the qubits to a known state:* The qubits must be initialized to a known state before performing any quantum computation

3. *Long coherence times:* The qubits must have long coherence times, meaning that they can maintain their quantum state for a sufficient amount of time to perform quantum operations.

4. *A universal set of quantum gates:* A universal set of quantum gates must be available for performing any quantum computation.

5. *The ability to measure individual qubits:* The qubits must be able to be measured individually to obtain information about the quantum state.

6. *The ability to interconvert stationary and flying qubits:* The ability to transfer quantum information between stationary and flying qubits is necessary for error correction and communication.

7. *A scalable architecture for connecting qubits:* The qubits must be connected in a scalable architecture to allow for efficient communication between them.

These criteria are considered essential for building a viable quantum computer and achieving all the criteria simultaneously is still challenging.

A two state-system that can be used for implementing such a qubit is a spin 1/2 particle, such as an electron. The two states are described by the electrons spin orientation $|\uparrow\rangle$ and $|\downarrow\rangle$. The simple configuration of having one electron in one dot was first introduced by Loss and DiVincenzo in 1998 [23].

Having a tight electronic confinement with one electron per dot, the Hamiltonian describing the system is given by the Heisenberg exchange Hamiltonian and the single-electron Zeeman Hamiltonian

$$H(t) = \frac{1}{4}\sum_{\langle i,j\rangle} J_{ij}(t)\boldsymbol{\sigma}_i\boldsymbol{\sigma}_j + \frac{1}{2}\sum_i g_i\mu_b \boldsymbol{B}_i\boldsymbol{\sigma}_i \tag{2}$$

where the spin-operators are defined by the Pauli operators $\boldsymbol{S_i} = \frac{\boldsymbol{\sigma_i}}{2}$, $\boldsymbol{B}_i$ and g_i are the magnetic field and g factor at site i [24].

For the initialization of the qubit, the electron needs to first be trapped in either the up or down state. This can be done by applying a large static magnetic field $B >> kB_T/g\mu B$ which only allows the spin down state to tunnel off the reservoir whereas the spin up state will stay on the dot. It can then be measured if there is a tunneling event or not using sensitive charge detectors which will tell the spin orientation of the electron. The spin of a single electron can be manipulated by applying an external static magnetic field B_0 in the z-direction which splits the energy levels of spin $\uparrow$ and $\downarrow$ by the Zeeman energy $\Delta E_Z > kB_T$. Further, an oscillating magnetic field can be applied perpendicular to B_0 at a frequency close to qubit splitting in order to drive oscillations around the Bloch sphere resulting into the following Hamiltonian

$$H = g\mu_B\vec{B}\cdot\hat{S} = g\mu_B(B_z\sigma_z + B_x\sigma_x cos(\omega t + \phi)) \tag{3}$$

[25]. The large static magnetic field is usually applied globally to the whole qubit device, whereas there are different methods for applying the oscillating magnetic field which usually is much smaller than the static one (frequencies ranging from approximately 5 to 50 GHz). One method is to fabricate a coplanar waveguide (CPW) near the qubit. [26] Others generate the magnetic fields by applying a voltage with frequency ω to nearby gates, which then derived the electron within the dot, this method is used in GaAs/AlGaAs and relies on the spin–orbit interaction [27]. Additionally, another method involves fabricating a micromagnet adjacent to the quantum dot to convert electron motion into magnetic fields [28].

2.3 SINGLET - TRIPLET SPIN QUBITS

Singlet-triplet $(S - T_0)$ qubits are formed from two spin-coupled electrons in a double quantum dot (DQD) and store information in the joint spin state of these two electron. The singlet-triplet qubit states are defined as:

$|S\rangle = \frac{1}{\sqrt{2}} |\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle$
$|T_0\rangle = \frac{1}{\sqrt{2}} |\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle$
$|T_+\rangle = |\uparrow\uparrow\rangle$
$|T_-\rangle = |\downarrow\downarrow\rangle$

which come from the eigenstates of a single electron spin $|\uparrow\rangle$ and $|\downarrow\rangle$. For spin 1/2 particle, we have $|0\rangle = |\uparrow\rangle$ and $|1\rangle = |\downarrow\rangle$ rotations on the Bloch sphere can then be built form the computational basis $|\pm\rangle = \frac{1}{\sqrt{2}} |0\rangle \pm |1\rangle$, as it can be seen on figure 1).

The electron occupation on the dot is denoted by (N,M), where (N) = number of electrons on the left dot and (M) = number of electrons on the right dot.

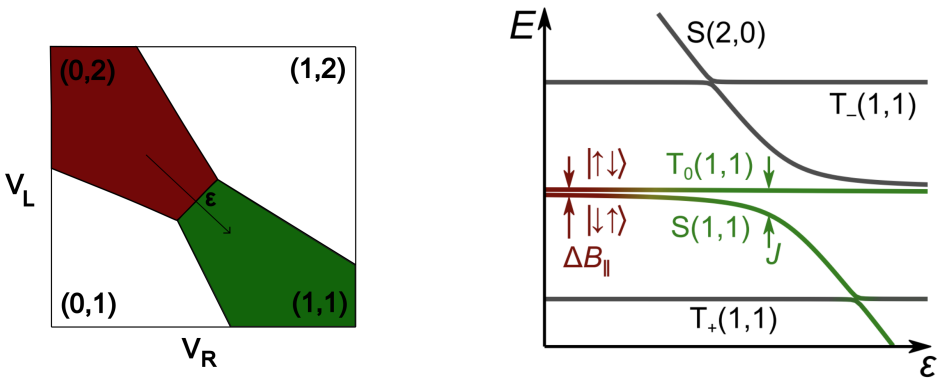
Opposite to the single electron spin qubit, the $S - T_0$ qubit is unresponsive to large global magnetic fields as $m = 0$ for both $|S\rangle$ and $|T_0\rangle$. The Hamiltonian with respect to the Pauli spin-operators for the $S - T_0$ system is given by

$$H = J_{ij}(\epsilon)\frac{\sigma_z}{2} + \mu_B \Delta \boldsymbol{E}_z \frac{\sigma_x}{2} \tag{4}$$

where $\Delta \boldsymbol{E}_z$ represents the difference in the Zeeman energy of the two spins and can be caused by a different g-factor $\Delta \boldsymbol{E}_z = \Delta g \mu_B \boldsymbol{B}_z$ or the difference in magnetic field between

the dots along the z-axis $\Delta \boldsymbol{E_z} = g\mu_B \Delta \boldsymbol{B_z}$. The exchange splitting $J_{ij}(\epsilon)$ is given by the external bias that controls the qubit frequency. This term drives transitions between the singlet and triplet states and can be used to perform qubit operations rotating around the z-axis [29]. and is controlled by the relative energy detuning ϵ between the two charge states (2,0)-(1,1) [30].

The ΔB_z term from the Hamiltonian makes the $S - T_0$ qubit sensitive to local magnetic-field fluctuations and drives rotations around the x-axis. For the initialization of a $S - T_0$ qubit Petta et al. have shown that two electrons on the same dot will after a while decay to the ground energy and become a singlet state $|S\rangle$ [31]. Using the above described notation (N,M), the qubit is initialized in the (2,0) charge configuration, where $\epsilon = 0$. Then by changing the detuning, that is the difference in energy between the left and right dot, J is switched on and thereby rotations around the x-axis are driven. Experimentally this is done by applying voltage pulses to the left and right gates (V_L and V_R) in figure 2a.



(a) (b)

Figure 2: (a) Schematic of a double quantum dot charge stability diagram with detuning axis ϵ. V_L and V_R represent the left and right gate voltage. The charge states are denoted (N,M) where N is the number of electrons in the left and M the number of electrons in the right dot. Adapted from [30]. (b) Energy levels of the two-electron double dot as a function of the detuning ϵ. For small ϵ the qubit rotates around the z-axis, $_{||}$, for larger amplitude values it rotates around the y-axis, J Taken from [32]

.

A more intuitive way to see figure 2a is with respect to the detuning axis as depicted in figure 2b. On the left hand side, the exchange interaction dominates over the Zeeman splitting difference and the eigenstates of the system are the common eigenstates of the

total spin operator $|S\rangle$ and $|T_0\rangle$. Moving along the detuning axis further to the right side, the eigenstates continuously change, as the electrons are now decoupled from each other and thereby the eigenstates of the system become $|\uparrow\downarrow\rangle$ and $|\downarrow\uparrow\rangle$. The connection in the middle region is used to incoherently convert the $|S\rangle$ and $|T_0\rangle$ states into $|\uparrow\downarrow\rangle$ and $|\downarrow\uparrow\rangle$ or vice versa. Lastly, on the very right side of the diagram in the (1,1) charge state, the electrons are decoupled from each other. Here the eigenstates become the tensor products of the individual spin states $|\uparrow\rangle$ and $|\downarrow\rangle$[33].

3

ELECTRON TRANSPORT IN SEMICONDUCTOR HETEROSTRUCTURES

3.1 QUANTUM POINT CONTACTS

The simplest form of electron transport can be explained with the example of a quantum point contact (QPC). Experimentally, a QPC can be implemented in a split-gate structure placed on top of a two-dimensional electron gas (2DEG). By applying a negative gate voltage to the gates, the 2DEG underneath the gates is depleted such that only a narrow channel still connects the remaining two electron reservoirs. This was first shown experimentally by van Wees et al. in 1988 [34] where they measured the resistance as a function of the voltage applied to the gate in a GaAs heterostructure as described above. By decreasing the applied voltage, the channel width connecting the two electron reservoirs decreases and thereby the measured resistance increases. The new observation for van Wees et al. was that the resistance increases step-like with plateaus at quantized values $\frac{h}{2Ne^2}$. The conductance is just the inverse of the resistance and thereby the plateau values for the conductance are

$$G = \frac{1}{R} = \frac{2e^2}{h}N \tag{5}$$

where N is an integer number. This is twice the conductance quantum

$$G_0 = \frac{e^2}{h} = 3.8740459 \times 10^{-5}\Omega^{-1} \tag{6}$$

[21]

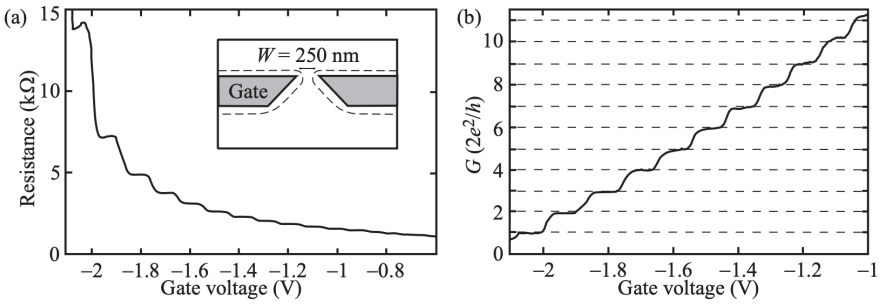


Figure 3: (a) Resistance of a QPC as a function of gate voltage. The inset shoes the split-gate structure consisting of two gates divided by a 250 nm gap. (b) Conductance of the same QPC as a function of gate voltage where the series resistance of the surrounding 2DEG is substracted. Taken from [21].

3.2 QUANTUM DOTS

As compared to a QPC, a quantum dot (QD) is a spacial region in a semiconductor material and the electrons on a quantum dot are confined in all 3 dimensions. To form a quantum dot, electrons are trapped, forming a charge island with discrete energy levels. The size of a QD is typically on the order of tens of nanometers to a few micrometers.

A quantum dot is connected via tunnelling barriers to conducting reservoirs and can be connected purely capacitively to any number of gates.

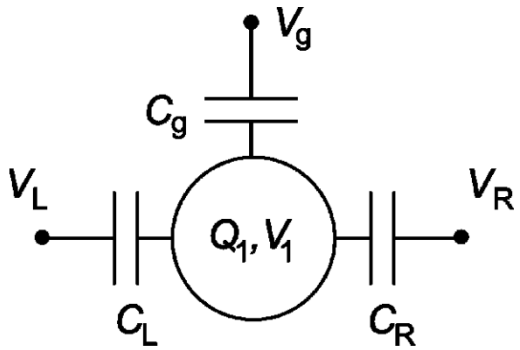


Figure 4: Schematic setup of a qauntum dot with source and drain at V_L, V_R respectively.The dot is connected to a plunger gate controlled by V_g, the dot is coupled via capacitors to source and drain and the plunger gate. Taken from [35]

For the work done in this thesis, we are mostly interested in gate-defined quantum dots, quantum dots where electron transport through the dot is controlled by the connected gate voltages.

3.2.1 *Coulomb blockade*

When explaining transport through quantum dots, the Coulomb blockade phenomenon plays an important role. In quantum dot structures electrons are confined to regions of very small space, therefore it is possible to study the quantum mechanical energy levels of the electrons. The quantum dot typically is coupled to a source and drain contact via tunneling coupling. By applying a small bias voltage electrons are sent through the setup and the source drain current can be measured. Further, the quantum dot is capacitatively coupled to a gate allowing to tune the quantum mechanical energy states in the quantum dot [36]. As explained earlier the quantum dot functions as an island containing N electrons with charge e, thereby the charge of the island is equal to Ne. The quantum dot functions as a conductor with capacitance C, where the energy required to add an extra electron to the dot can be expressed as

$$E_C = \frac{e^2}{C} \tag{7}$$

This becomes important in the limit where

$$\frac{e^2}{C} >> k_B T \tag{8}$$

and can be achieved by making the dot very small. Furthermore, the barriers need to be opaque in a way that the electrons will be either in the source, the drain or on the island, meaning that the amount N of electrons that can tunnel through the barriers is much less than the inverse of the time scale of the measurement. To charge or discharge the island it takes $\Delta t = R_t C$, where R_t is the tunnel resistance of the barriers. The Heisenberg uncertainty relation $\Delta E \Delta t > h$ can then be rewritten to $\frac{e^2}{C} R_t C > h$ which implies that

$$R_t >> \frac{h}{e^2} \tag{9}$$

where $\frac{h}{e^2}$ is the resistance quantum. This can be achieved by weakly coupling the dot to the source and drain leads.

Considering the simple setup of a quantum dot coupled to source and drain via tunneling barriers and capacitively to a plunger gate as schematically depicted in figure 4, we can

regard the quantum dot as an isolated system with an integer number of electrons N. Each N-electron quantum dot has a ground state and a number of excited state, which we label with n, such that the energy of the quantum dot can be written as $E_N^{(n)}$ where n increases with higher energy states. Another factor influencing the energy of the dot is the gate voltage V_g applied to the plunger gate coupled to the dot, such that the energy of the dot can be tuned by changing V_g.

Both source and drain have an electrochemical potential μ_s or μ_d, as well as the dot μ_N. The electrochemical potential of the quantum dot describes the energy required to add or remove an electron from the dot:

$$\mu_N(V_g) = E_N^{(0)}(V_g) - E_{N-1}^{(0)}(V_g) \tag{10}$$

as equation (10) suggests, changing V_g will change the energy of the dot, this can be used to tune the electrochemical potential of the dot such that electron transport through the dot is possible. This is the case when the electrochemical potential of the source contact, the dot and the drain contact are aligned (see figure). In the case of adding one electron to a N-electron dot.

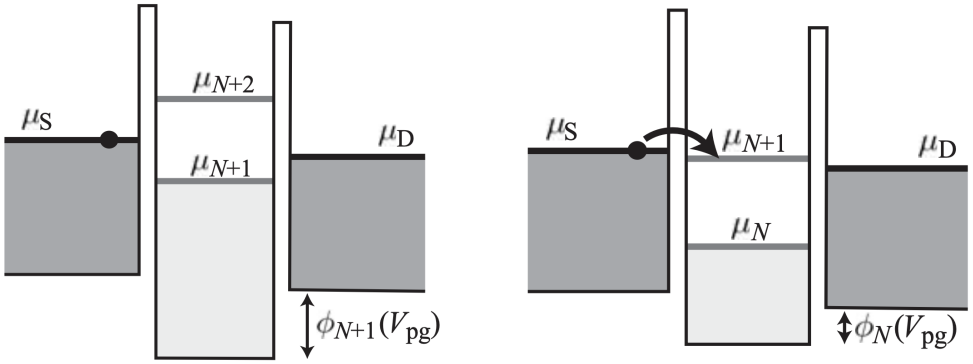
$$\mu_S \approx \mu_{N+1}(V_g) \approx \mu_D \tag{11}$$

In the situation where neither of the electrochemical potentials are aligned, tunneling is not possible as this requires too much energy and we are in the regime of Coulomb blockade. Current does not only flow when the energy levels are perfectly aligned, but also when a small bias window is applied. Here the electrochemical potential of the dot needs to be within the bias window of source and drain:

$$\mu_S - \mu_D = -|e|V_{SD} \tag{12}$$

in order for current to flow, with the condition that:

$$\mu_s \geq \mu_N V_g \geq \mu_D \tag{13}$$



(a) (b)

Figure 5: Electron transport through a quantum dot system. (a) Energy level structure of the system in the Coulomb blockade. (b) Position of the energy levels that allows a current to flow between source and drain if a very small bias voltage is applied. Taken from [21].

In an experimental context Coulomb blockade can be observed in the trace of a current-voltage $(I_{sd} - V_g)$ measurement, showing high conductance in the areas of electron transport and zero conductance at Coulomb blockade. An example can be seen in figure 6 where a simple quantum dot is formed by two quantum point contacts connected to source and drain, with a plunger gate in the middle (see inset 6). The function of the plunger gate is to tune the electron density between the two QPCs. By applying a small voltage between source and drain, a current that can be controlled by the plunger gate, starts flowing through the quantum dot. In this regime the conductance in the dot is given by $G = I/V_{SD}$ and can be plotted as a function of the plunger gate voltage.

Now, when the plunger gate voltage is in the configuration where the energy levels of source and drain are aligned, conductance resonance can be observed in a form of a sharp peak, showing electron transport from source to drain. In the regime where the applied plunger gate voltage tunes the energy levels in such way that they are not aligned, it is not possible for electrons to tunnel through the dot and the system is in the Coulomb blockade.

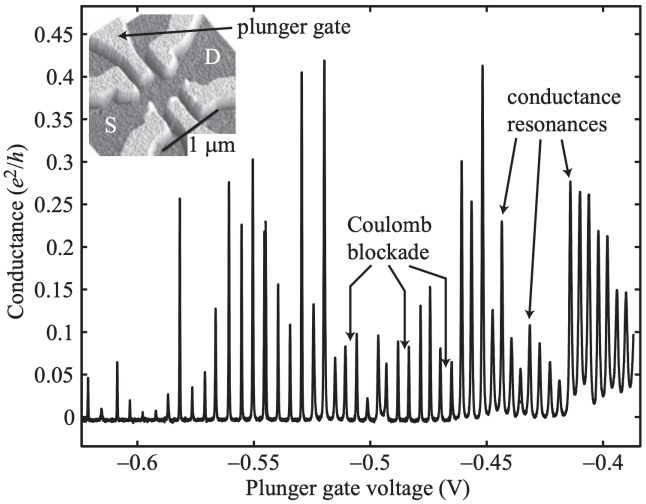


Figure 6: Conductance through a quantum dot as a function of the applied plunger gate voltage showing Coulomb blockade (no conductance) and Coulomb peaks (conductance resonances) Inset: SEM image of a simple quantum dot setup connected to source (S) and drain (D). Taken from [21].

3.3 CONSTANT INTERACTION MODEL

The constant interaction model describes the interaction between the electrons in a system under the assumption that these are constant. The number of electrons on the dot is given by an integer N, thereby the charge of the dot is given by Ne. When an electron tunnels onto or from the dot the charge of the dot changes by the quantized amount e while the Coulomb energy changes by the charging energy $E_C = \frac{e^2}{C}$.

For this we have to work with 2 assumptions:

1. Coulomb interactions between the electrons on the dot with each other and the electrons on the dot with the electrons in the environment can be parameterized with one single capacitance term $C = C_{source} + C_{drain} + C_{gate}$.

2. The energy-level spectrum of the single-particle is independent of the Coulomb interactions.

The ground state energy for a dot with N electrons is given by the electrostatic energy:

$$U(N) = \frac{1}{2C}[-|e|(N - N_0) + C_g V_g + C_S V_S + C_D V_D]^2 + \sum_{n=1}^{N} E_n(B) \qquad (14)$$

with $C_i V_i$ describing the charge induced at the capacitors which will effectively change the electrochemical potential of the dot. N_0 is the amount of electrons in the dot with no voltages applied. The sum in the end of the equation sums over the quantized energy states coming from the confinement potential of the quantum dot multiplied by the external magnetic field.

The electrochemical potential is given by the difference in ground state of two following occupations.

$$\mu(N) = U(N) - U(N-1) = \frac{e^2}{C}(N - \frac{1}{2} - N_0) - \frac{1}{C}(C_g V_g + C_S V_S + C_D V_D) + E_N \tag{15}$$

introducing the known term for the charging energy $E_c = \frac{e^2}{C}$ describing the difference between the energy occupations, we can write the chemical potential as

$$\mu(N) = E_c(N - \frac{1}{2} - N_0) - \frac{E_c}{|e|}(C_g V_g + C_S V_S + C_D V_D) + E_N \tag{16}$$

Now the total energy needed to add an electron to the dot is given by the difference in the chemical potential of two adjacent occupations:

$$E_{add} = \mu(N+1) - \mu(N) = E_c + \Delta E \tag{17}$$

By applying a bias voltage between the source and drain a so called bias window between source and drain opens up. Within this window the electron states in one reservoir are filled while the electron states in the other reservoir are empty. An electron can tunnel through the dot, when the electrochemical potential of one energy level is within this bias window. As shown in figure 5 it is not possible for an electron to tunnel through the dot while there is no alignment within the bias window of the electrochemical potential of the dot, here the system is in the Coulomb blockade. By decreasing the gate voltage, the chemical potential μ_{N+1} is raised and thereby reaches the regime inside the bias window between the chemical potential of the source and drain. Now it is possible for an electron to tunnel onto the dot and subsequently leave the dot to the drain [37].

When looking at an $I_{sd} - V_g$ curve, we can thereby see the tunneling effects as lines of high currents in the regimes where tunneling is made possible due to the correct setting of bias window and applied gate voltage.

3.3.1 *Constant interaction model in matrix form*

The constant interaction model can also be written in matrix form, which will be useful when studying multiple quantum dots coupled to one another[35]. We can write the charge on the *ith* quantum dot as the sum of all charges on the capacitors connected to it:

$$Q_i = \sum_{j=0}^{n} q_{ij} = \sum_{j=0}^{n} C_{ij}(V_i - V_j) \tag{18}$$

where V_i and V_j are the electrostatic potentials on the *ith* and *jth* node respectively. This can be expressed in matrix form as

$$\vec{Q} = \boldsymbol{C}\vec{V} \tag{19}$$

or when

$$\begin{pmatrix} \vec{Q}_c \\ \vec{Q}_v \end{pmatrix} = \begin{pmatrix} \boldsymbol{C}_{cc} & \boldsymbol{C}_{cv} \\ \boldsymbol{C}_{vc} & \boldsymbol{C}_{vv} \end{pmatrix} = \begin{pmatrix} \vec{V}_c \\ \vec{V}_v \end{pmatrix} \tag{20}$$

where c and v denote the charge for the dot and voltage for the gate nodes respectively. The charge matrix is is divided into 4 matrices in order to consider all couplings: charge-charge, charge-voltage, voltage-charge and voltage-voltage coupling.

In order to obtain the voltages on the charge nodes equation 11 gives

$$\vec{V}_c = \boldsymbol{C}_{cc}^{-1}(\vec{Q}_c - \boldsymbol{C}_{cv}\vec{V}_v) \tag{21}$$

where C_{cc} and Ccv can be obtained from the experiment.

The voltages on the charge nodes can then be used to calculate the total energy of the system, using that the total electrostatic potential energy stored in a capacitor is given by

$$U_C = \frac{1}{2}V^2C = \frac{1}{2}Q^2C = \frac{1}{2}QV \tag{22}$$

[38] and it is known from the experimental setup which voltage is applied to the plunger gates.

3.4 DOUBLE QUANTUM DOTS

Two quantum dots can be coupled either in series or in parallel to form a double quantum dot. For a double quantum dot coupled in series, the system can be regarded as two islands coupled to source and drain via tunneling barriers connected with capacitors $C_{L(R)}$ and

tunnel resistors $R_{L(R)}$ in between the dots and source and drain. Furthermore, each dot is capacitively coupled to a gate voltage $V_{g1(2)}$ and they are coupled to each other via a capacitor C_m and a tunnel resistor R_m, which are introduced as the interdot coupling strength (see figure 7).

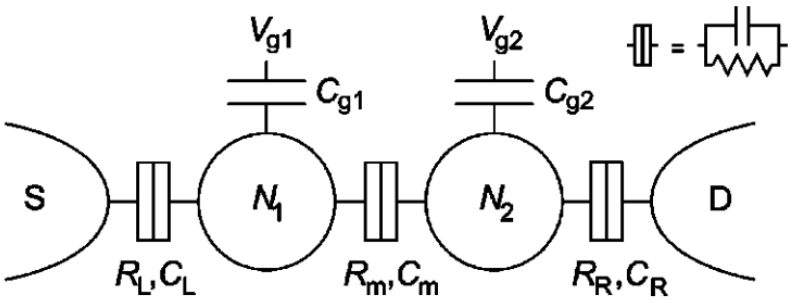


Figure 7: Schematic setup of a double quantum dot with source and drain. The two dots are in the middle with N_1 and N_2 electrons on each dot respectively. Each dot is coupled to a gate via a capacitor and can be controlled with the voltage applied to the gate. Quantum dot 1 is coupled to a source contact and dot 2 to a drain contact. Taken from [35]

The constant interaction model can also be applied to a system of two or more quantum dots as shown above. The energy of the system as per equation (22) is then given by

$$U(N_1,N_2) = \frac{1}{2}N_1^2E_{C1} + \frac{1}{2}N_2^2E_{C2} + N_1N_2E_{Cm} + f(V_{g1},V_{g2}) \tag{23}$$

where

$$\begin{aligned} f(V_{g1},V_{g2}) = \frac{1}{-|e|}\{&C_{g1}V_{g1}(N_1E_{C1} + N_2E_{cm}) \\ &+ C_{g2}V_{g2}(N_1E_{Cm} + N_2E_{C2})\} \\ &+ \frac{1}{e^2}\Big\{\frac{1}{2}C_{g1}^2V_{g1}^2E_{C1} + \frac{1}{2}C_{g2}^2V_{g2}^2E_{C2} \\ &+ C_{g1}V_{g1}C_{g2}V_{g2}E_{Cm}\Big\} \end{aligned} \tag{24}$$

Here $C_{1(2)} = C_{L(R)} + C_{g1(2)} + C_m$ are the total capacitances coupled to dot 1 (2) and

$$E_{C1} = e^2\frac{C_2}{C_1C_2 - C_m^2}, E_{C2} = e^2\frac{C_1}{C_1C_2 - C_m^2}, \tag{25}$$

are the charging energy of dot 1(2) and

$$E_{Cm} = e^2\frac{C_m}{C_1C_2 - C_m^2} \tag{26}$$

is the electrostatic coupling energy, that is the change in the energy of one dot when an electron is added to the other dot.

3.4.1 *charge stability diagram*

To explain electron transport in a double quantum dot, we need to consider the electrochemical potential of the two coupled dots μ_1 and μ_2. As described previously, the electrochemical potential describes the energy needed to add or remove an electron from one quantum dot, combining equation (14) and (23) we can write the electrochemical potentials of dot 1(2) as

$$\begin{aligned}\mu_1(N_1,N_2) &= U(N_1,N_2) - U(N_1-1,N2)\\ &= \left(N_1 - \frac{1}{2}\right)E_{C1} + N_2 E_{Cm}\\ &- \frac{1}{|e|}(C_{g1}V_{g1}E_{C1} + C_{g2}V_{g2}E_{Cm})\end{aligned} \tag{27}$$

and

$$\begin{aligned}\mu_2(N_1,N_2) &= U(N_1,N_2) - U(N_1,N_2-1)\\ &= \left(N_2 - \frac{1}{2}\right)E_{C2} + N_1 E_{Cm}\\ &- \frac{1}{|e|}(C_{g1}V_{g1}E_{Cm} + C_{g2}V_{g2}E_{C2})\end{aligned} \tag{28}$$

The charge stability diagram shows the voltage configuration of V_{g1} and V_{g2}, where the double quantum dot system is in a particular charge state, having a specific number of electrons in each quantum dot. The coupling strength between the dots determines the shape of the charge stability diagram. In the case of $C_m = 0$ the dots are completely decoupled and the charge on the dot is only dependent on the gate voltage of the gate it is coupled to. By increasing the coupling strength to the intermediate regime, where $C_{1(2)} > C_m > 0$ the lines in the charge stability diagram form hexagonal shapes. In the strong regime, where $C_m > C_{1(2)}$, the double dot can be seen as one large dot, with total charge $N = N_1 + N_2$. These three configurations can be seen in figure 8.

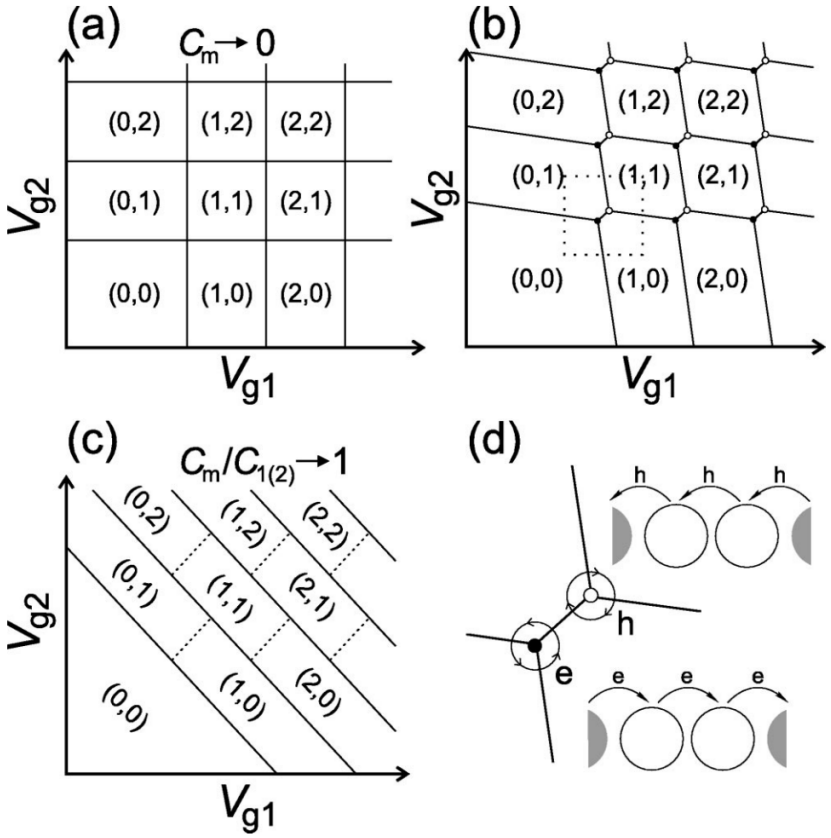


Figure 8: Charge stability diagrams for the different limits for C_m. (a) small interdot coupling (b) intermediate interdot coupling showing the characteristic honeycomb pattern and (c) large interdot coupling. The charge states are denoted by (N_1, N_2). (d) Zooms in to the dashed square, showing the electron and hole triple points, a full square marks an electron tunneling counterclockwise and an empty circle a hole tunneling clockwise. Taken from [35].

The lines between different charge states in the charge stability diagram correspond to points where the electrochemical potential of one or both quantum dots cross the chemical potential of the leads. At a triple point where three lines meet, the electrochemical potential of the right and left quantum dot are both aligned with the electrochemical potential of the electron reservoirs. At these triple points the charge transfer can be described as

$$(N_1, N_2) \to (N_1 + 1, N_2)) \to (N_1, N2 + 1) \to (N_1, N_2) \tag{29}$$

for an electron tunneling from dot 1 to dot 2 and

$$(N_1 + 1, N_2 + 1) \to (N_1 + 1, N_2)) \to (N_1, N2 + 1) \to (N_1 + 1, N_2 + 1) \tag{30}$$

for a hole tunneling in the opposite direction. Triple points are interesting, because at their location, three different charge configurations have the same energy and the system can

transform into any of these charge configurations by adding or removing an electron or hole from one of the quantum dots. At this configuration, the chemical potential of the electron reservoir is aligned with the chemical potentials of dot 1 and 2, making electron tunneling possible in the same manner as for a single quantum dot.

3.5 IMPLEMENTATION OF SPIN QUBITS IN SEMICONDUCTORS

The two-dimensional electron gas (2DEG) is made of a thin layer of electrons that can move in two dimensions and is confined by the surrounding materials in the third dimension. In the context of semiconductors, 2DEGs can be formed at the interface between a semiconducting material and an insulating material or between two different semiconducting materials with different band gaps. Finally, the semiconducting material is overgrown with an oxide and a metal gate. By applying a voltage to the metal gate, the electric field created by the gate is able to control the electrons in the gas from underneath, creating a potential well [10]. Thereby electrons can be localized to a one-dimensional channel underneath the gate and the conduction through the device can be controlled by the metal gate. In an experimental setup, typically there are several gates and the gate voltages can be controlled individually, confining the electrons in the semiconductor such that quantum dots or double dots are created and the number of electrons on each dot can be controlled with the gate voltages. If there is a very narrow confinement, a quantum point contact is created.

One common approach to implement spin qubits is to use a semiconductor material, such as silicon, silicon/germanium (Si/SiGe) or gallium arsenide/aluminum gallium arsenide (GaAs/AlGaAs)[39], and create a structure that can trap a single electron in a well-defined location.

GaAs/AlGaAs is a typical material used for a 2DEG. This is due to the fact that only materials of the same lattice constant and crystal structure can be grown on top of each other without creating strain [21]. However, when a thin layer of AlGaAs is deposited on top of GaAs, there is a difference in the lattice constants, hence the band gap of the two materials, leading to strain at the interface of the materials, creating a quantum well at their interface [30]. A schematic of such a heterostructure can be seen in figure 9:

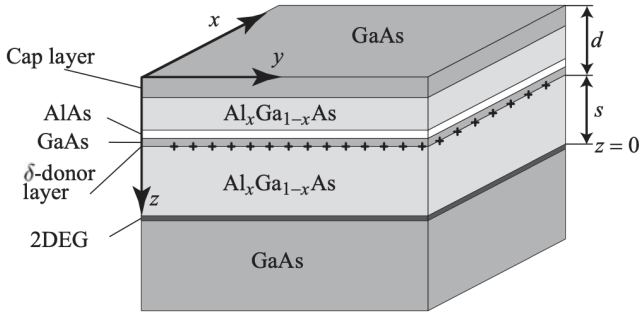


Figure 9: Typical AlGaAs heterostructure taken from [21]. The metal gate structure is placed on top of the cap layer (not depicted in this schematic).

The measurements and tuning described in this thesis, were performed on quantum dots formed in a two-dimensional electron gas in a GaAs heterostructure, a device fabricated by Federico Fedele at Qdev in 2018 [30]. The layer structure of his device is thankfully taken from his Ph.D. thesis:

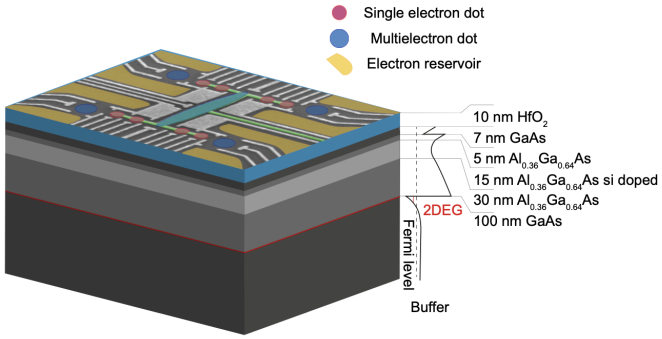


Figure 10: GaAs/AlGaAs hetero structure including metal gate structure on top of the cap layer. Taken from [30]

Compared to other heterostructures as for example silicon/silicon germanium, gallium arsenide has some advantages, such as faster electron mobility and higher electron densities. However, gallium arsenide hasn't been accommodated as much in the modern electronic technologies as silicon/silicon germanium, and there are challenges in terms of fabrication and integration with other components in a quantum computer. Even though the automation code of this thesis is focused on GaAs, equivalent work has been done in silicon/silicon germanium and should be straightforward to transfer the knowledge to different materials.

Part III

TUNING PROCESS MANUALLY

4

TECHNICAL SETUP AND SOFTWARE

This chapter will give a short overview of the experimental setup in the laboratory and the device used for measuring and tuning. It is important to note, that there are several different ways of acquiring data from a quantum device and they can have many different architectures which will all influence the tuning method. This chapter will also briefly introduce the Python package used for data acquisition.

4.1 DC TRANSPORT

A simple setup to acquire a quick understanding of the loaded device is the direct current transport setup. The device is loaded to and cooled down to $\approx$ 30mK by an Oxford instruments Triton dilution refrigerator. For applying a voltage a high-precision low-noise computer-controlled voltage generator (QDAC) is used. The QDAC is connected to the breakout box (Qbox) which is connected to the dilution refrigerator. Both the QDAC and the Qbox have 24 channels, these channels can be connected to each other via BNC coaxial cables. Furthermore, the channels on the Qbox can be individually turned on and off by small switches. As each gate of the quantum device is connected to one of the output channels of the QDAC the turn-on/off of the gates can be controlled by the Qbox. The outgoing current is measured by going through an Ithaco current pre-amplifier and finally measured with an Agilent Keysight digital multimeter.

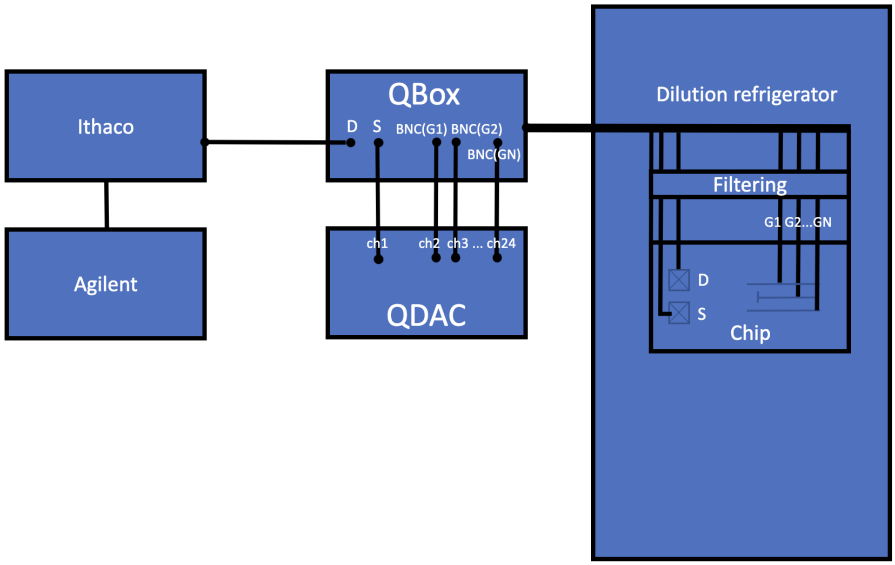


Figure 11: DC transport experimental measurement setup for N connected gates (G), ohmic source and drain. A QDAC has 24 channels, we can use $N-1=23$ gates or we can synchronize a second QDAC and reach 47 channels gate input. The QDAC is connected to the Qbox/breakoutbox which is connected to the dilution refrigerator containing the device via a Fischer cable. The outgoing signal is amplified by the ithaco and measured with the keysight agilent.

4.1.1 *RF reflectometry*

For the radio-frequency reflectometry the experimental setup is more complicated. The in-going signal (green in figure 12) is first sent into an demodulation setup. The first component of the demodulation setup is a DC blocker that eliminates all direct current to proceed through the setup. The rest of the signal is then split into two. One part continues as the in-going signal to a RF mixer which we will come back to later, the other damped output continues to a phase shifter and an high- and low pass filter. Finally the signal goes through a programmable attenuator that functions as a RF switch and turns on the RF excitation by directing the signal to the RF input TxA port of the dilution refrigerator. Before entering the fridge and the sample, the signal is passed through multiple attenuators, reducing its power. The reflected signal is again amplified in the fridge and sent as the output signal (red in figure 12 from the TxB port to an another amplifier in the demodulation setup. The amplified signal is mixed with the original split input signal in the mixer and passed through a low-pass filter and measured with the Agilent keysight digital multimeter.

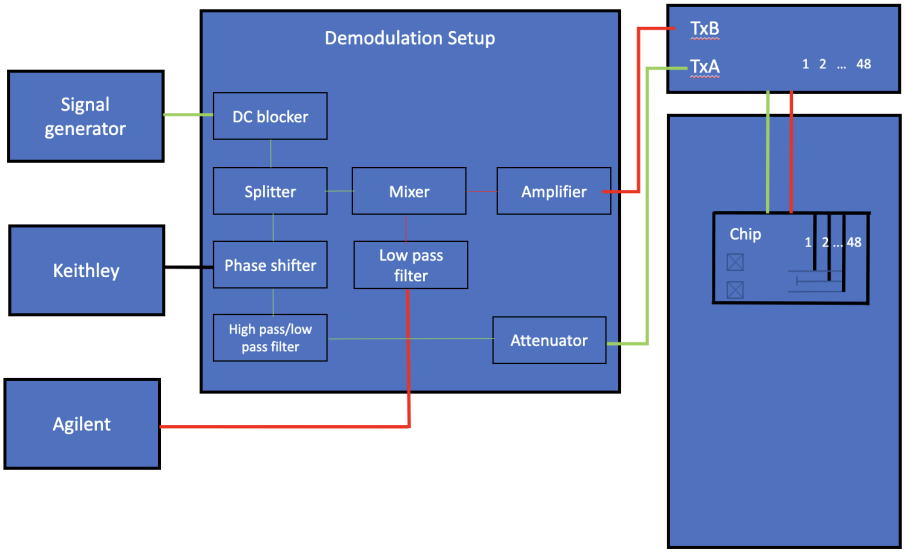
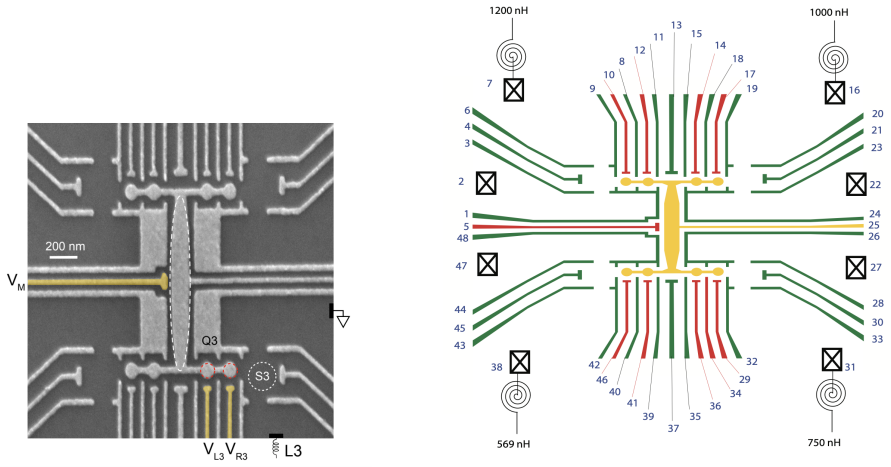


Figure 12: RF Reflectometry setup used for the measurements in this work. The green lines marks the incoming signal and the red lines the outgoing reflected signal.

4.2 DEVICE

The device used in this thesis, is a Al/GaAs device which has been fabricated by Federico Fedele in the course of his Ph.d. thesis in 2018. All measurements and tuning described in the following chapters have been conducted on this device. The geometry of the device plays an important role for the tuning into the regimes where quantum dots are formed and for interpreting the results.



(a) (b)

Figure 13: FF1A device fabricated by Federico Fedele. (a) SEM image of the FF1A device. (b) BNC chart pin out of the same device. Taken from [30].

The split gate structure of the device consists of 4 quadrants. Each quadrant contains two ohmic contacts that can be connected to a current provider or set to a bias. The metallic gate structure contains plunger and barrier gates, also sometimes called finger gates.

A finger gate, is a narrow gate that is located near the edges of the quantum dot, it is used to confine the electrons within the dot. By applying a voltage to the finger gate, an electrostatic potential is created that acts to trap the electrons and confine them to a quantum dot. A special type of finger gate is the barrier gate. These are used to control the flow of electrons between adjacent quantum dots or between a quantum dot and a nearby reservoir.

A plunger gate is located above the quantum dot and is used to change the energy level of the dot. By applying a voltage to the plunger gate, the energy level of the quantum dot can be tuned. This makes it possible for electrons to move into or out of the dot as explained in section 3.2.1.

The yellow gate in the middle is the Jelly Bean gate which can be used to couple the qubits of the device, but will not be used in the course of this thesis.

4.3 QCODES

For taking measurements on the device, the data acquisition framework QCoDeS (Quantum Control and Dynamics Simulation) is used. QCoDeS is a Python-based software suite developed by the Copenhagen / Delft / Sydney / Microsoft quantum computing consortium [40].

QCoDeS is used for scientific experiments in the field of quantum computing and provides a framework for controlling and measuring experimental quantum devices, as well as for data acquisition and analysis.

In this thesis QCoDeS is used to automate the experimental setup and run experiments in a reproducible and scalable way. It is used for data acquisition, data logging, real-time plotting and parameter tuning. The data logging is to ensure the reproducibility of measurement results to facilitate experiments to be repeated in the same manner. Data visualization is important to understand data and measurements taken, crucial for the manual tuning as decisions are based on data plots. For the automated tuning interim data plots are not necessary, but still a useful tool to demonstrate results. The most important, basic functions that are needed to take measurements on a device can be found in appendix 8.3. For more details I refer to the QCoDeS documentation: `https://github.com/QCoDeS/Qcodes`.

5

TUNING AND MEASUREMENT TECHNIQUES

5.1 EXPERIMENTAL MEASURING TECHNIQUES

In this section I will briefly explain the two different measurement techniques which have been used for the experimental tuning of quantum dots in this work.

5.1.1 *DC transport*

Tuning in transport is a straightforward method for tuning a device into quantum dot regime. It is easy to set up and therefore a good start when learning how a quantum device works. For this method, a direct current (DC) is sent past the device's gates while its gate voltages are adjusted such that the 2 DEG is depleted and quantum dots are formed. As much as it is practical to obtain information about a device's functionality, it can be difficult to observe all charge states in the direct current trough the quantum dots. Close to the pinch-off point where the current goes to zero for example it is hard to detect the Coulomb peaks and thereby difficult to find the first charge states such as $(0,0)$ or $(0,1)$ [41]. For this purpose charge sensing is more reliable.

5.1.2 *Charge sensing*

A widely used method to measure the charge of semiconductor gate-defined quantum dots, is through charge sensors. This method can be used to read out the electron spin and charge, but also to read out the charge on self-assembled quantum dots, as first demonstrated by Kiyama et al. [42]. For charge sensing one quantum dot is formed, which is then used

as a sensitive detector for measuring the state of another target dot. With large enough capacitive coupling between the dots, the conductance through the sensor dot changes when an electron is added to or removed from the target dot. Furthermore, real-time single-electron tunneling events can be detected. In practice, the current of the sensor dot is measured as a function of the gate voltage applied to its plunger gates [43]. For a good sensor dot, the aim is to optimize the sensors' sensitivity by tuning its barrier gate voltages while measuring the conductance through the sensor dot. In the right regime, it is expected to observe Coulomb oscillations when scanning the barrier gates and plunger gate. Often the optimal configuration is reached in a position where the slope of the Coulomb peaks is steepest. The experimental implementation and steps to tune a sensor dot are outlined in section 6.3.1.

5.1.3 *RF reflectometry*

A precise method for measuring and tuning quantum devices via charge sensing is radio-frequency (RF) reflectometry. In RF reflectometry, changes in impedance are measured and give information on the state of the system. [44]. This is done by monitoring the reflection coefficient of a microwave signal that is sent through a transmission line to the device with total impedance Z_{load}. The reflection coefficient is a measure of the amount of power that is sent in, $V_{in}(t)$ and reflected back (in reflection configuration) towards the source of the signal, and it depends on the impedance of the device at the point where the signal is reflected. It is defined as the ratio of the amplitude of the reflected wave to the amplitude of the incident wave Z_{load}/Z_0 with $Z_0 = \sqrt{\frac{L_l}{C_l}}$ for a wave propagating in a single direction and typically represented as a complex number with a magnitude and a phase, depending on the impedance of the device and the frequency of the signal.

The outgoing signal is then amplified to boost it above the noise of following electronics used for the analysis. Finally, the signal is demodulated to shift it away from the carrier frequency and to convert it from AC to DC. This is to get to the signal itself, as it is often more convenient to work with a lower frequency.

When the impedance of the device changes, the reflection coefficient also changes, and this can be detected by measuring the amplitude and phase of the reflected/transmitted signal which is first amplified and then demodulate by multiplying it with a demodulation

tone $V_{LO}(t)$ and low-pass filtering the product. The output $V_{IF}(t)$ of the demodulation circuit carries the required information about the device impedance which are detected and converted to a voltage [44].

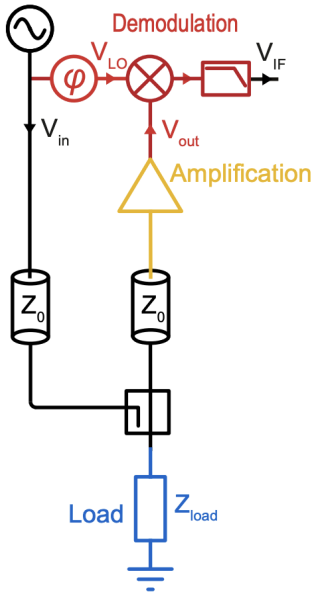


Figure 14: Reflectometry transmission setup. Taken from [44]

By measuring the reflection coefficient as a function of frequency, it is possible to determine the resonant frequencies of the device or circuit, as well as the impedance at these frequencies. This information can be used to tune the resonant frequencies of the device and optimize its performance.

5.2 MEASUREMENTS

While in the previous section I considered the measuring techniques, this section will focus on the methods of how the data is taken and depicted.

5.2.1 *1D and 2D measurements*

Current and voltage data can be acquired for a single gate and plotted as one-dimensional $I_{sd} - V_g$ curves, so-called pinch-off curves, showing the relationship between the current (I_{sd}) flowing from source to drain in a device and the gate voltage (V_g) applied to the gate

or for two gates simultaneously, where the voltage and current values are dependent on each other and plotted as a two-dimensional $V_{g1} - V_{g2}$ current map showing the strength of the current as a colorbar. The data in this work is acquired using QCodes `do1d` or `do2d` functions (see appendix 8.3). Taking one-dimensional measurements takes significantly less time and computational power as they only show data for one single gate, meaning one measurement per data point, whereas the two-dimensional measurements measure two gates simultaneously. Pinch-off curves or current maps are taken for different purposes. To begin with they are used for assessing the current flowing through the whole device by measuring the bias. Later, for gate characterization, that is to check the functionality of a gate and finally for tuning, that is to find the right gate voltage configuration for creating quantum dots.

5.2.2 *Bias sweep*

The bias sweep is a single measurement of the current flowing through the entire device depending on the bias set to an ohmic contact, which depends on the bias applied to an ohmic contact, resulting in a voltage from source to drain. As an ohmic contact follows ohms law $U = RI$ the slope of the $I_{sd} - V_{sd}$ graph, which responds to the resistance, should be constant and we expect a linear line going from a high positive current at the most negative bias voltage to an high negative current at the most positive bias voltage.

5.2.2.1 *Gate characterization*

This section will describe the typical properties of the 1D and 2D measurements plots used for the gate characterization.

The pinch-off curve of a functional gate typically will have a sigmoidal shape, showing voltage (V) on the x-axis and current (I) on the y-axis, with three regions:

- Pinch-off region: In this region, the current is zero or very small. This is due to the voltage being applied here is too negative to allow conductance.
- Linear region: In this region, the current increase proportional to the voltage, either linearly or linearly with small steps/plateaus in between showing QPC behavior.

- Saturation region: In this region, the current stops increasing and becomes constant at the point where the device has reached its maximum conductance.

The two-dimensional map shows the voltage applied to the first gate on the x-axis and the voltage applied to the second gate on the y-axis. The current flowing through the device is shown as a color scale in the case of this thesis ranging from dark blue for no current to yellow for current saturation. Similarly the two-dimensional map can be divided into three regions.

- Pinch-off region: In this region, there is zero or very small current. The voltage applied to the two gates doesn't allow conductance.
- Middle region: In this region, both gate voltages allow small conductance, this region corresponds to the linear region, described for the one-dimensional plots and indicates transitions from one operational state of the device to another.
- Saturation region: In this region, the voltage of one or both gates is increased to the point where the current reaches its maximum value.

In both cases a broken gate will mostly show noise and no regular features.

5.2.3 *Quantum dot and double dot tuning*

When tuning the device into quantum dot or double dot regime we can both use 1D and 2D measurements. The typical approach is to start out with 2D measurements of the outer gates expected to form a quantum dot with the purpose of finding the voltage space showing interesting features that can then be used for further tuning. Areas with no current can directly be excluded for further analysis. Later in this work, I will show that the scope of the interesting voltage space can also be found by 1D measurements.

Once the device is in the quantum dot regime, the desired one-dimensional pinch-off curve exhibits Coulomb oscillations, a series of sharp peaks in the region where a quantum dot is formed. These peaks prove areas of high conductance, followed by Coulomb blockade areas with no or low conductance. Each peak corresponds to a different number of electrons occupying the quantum dot, with the number of electrons increasing as the bias voltage is increased. The physical explanation of Coulomb oscillations is described more

detailed in section 3.2.1. The two-dimensional maps are also desired to show Coulomb oscillations, in the form of Coulomb diamonds. For the double dot we would like to see a honeycomb pattern, showing charge state transitions in the double dot, as explained in more detail in section 3.4.1.

5.2.4 *Optimization*

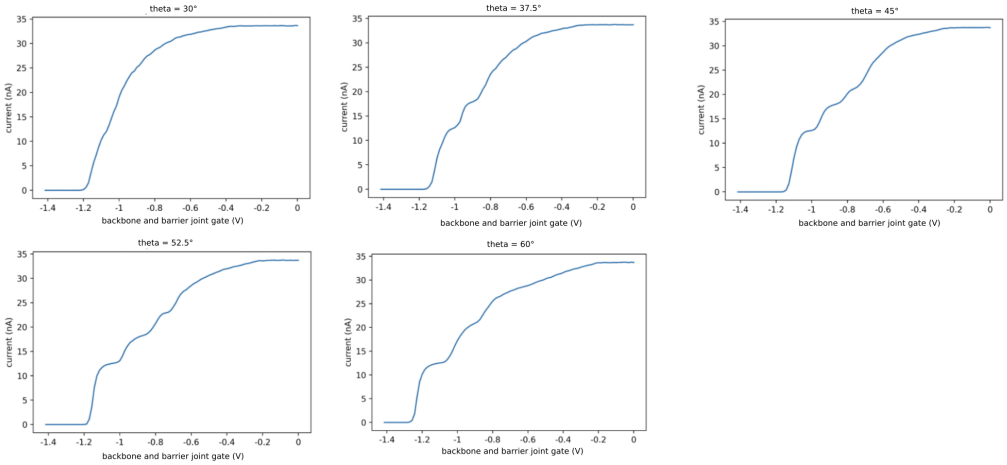
While 1D measurements of current and voltage of a single gate can be taken in a relatively short amount of time, acquiring 2D maps of two gates simultaneously is more time-consuming. This is because data needs to be acquired for two gates at the same time and their interrelation needs to be considered. This can be hindering at times, particularly in the light of developing an automated tuning process where speed also plays a role.

There are several possibilities to speed up the process, I will elaborate on the methods I have implemented in the course of this thesis.

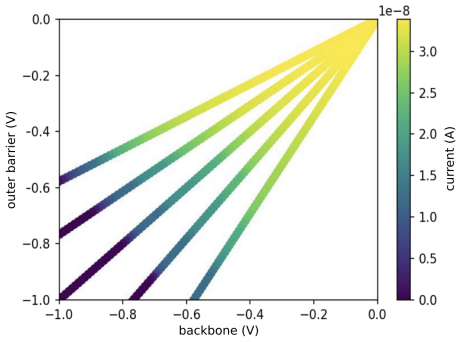
5.2.4.1 *Ray based measurements*

The ray-based measurement method aims at only picking rays in the 2D map and thereby reduces the number of measured points significantly. When sweeping two gates over a large voltage area, as it is typically done in the beginning of the tuning to obtain a comprehensive understanding of the gates pinch-off landscape, the interesting features are usually only present in a smaller, often the middle, area of the 2D map. Consequently, computation time can be saved by only measuring the common gate behavior in rays within the middle region. To achieve this, we define a joint gate of the two gates we want to sweep together. An arbitrary number of evenly spaced angles θ where $\theta = \frac{\pi}{4} rad$ represents a straight diagonal line cutting through the plot from the top right corner to the bottom left corner, is defined to arrange the spread of the rays to be measured. Each joint gate is then formed from the two specified gates based on the particular angle using $\cos\theta\frac{\pi}{180}$ or $\sin\theta\frac{\pi}{180}$ which is multiplied by a predefined value a. The joint gate is returned as a `qcodes.Parameter` which can be swept in a single 1D scan over a large voltage span. In order to achieve a sweep over the typical voltage range $[-1, 0]V$ a is often chosen to be $a = \frac{-1}{\cos(\pi/4)}$and functions of the start value for the `do1d` sweep. These sweeps can then be plotted as single joint gate per angle pinch-off curves or recalculated to rays in the

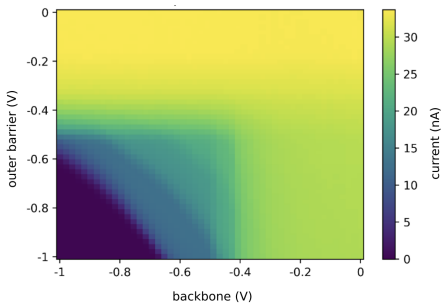
2D plane. Ideally, the joint gate plots can show interesting features in form of Coulomb oscillations (1D) or the stripy pattern (2D). These plots can be used to select the suitable voltage values in the quantum dot regime.



(a)



(b)



(c)

Figure 15: Ray-based measurement method, here used to find the initial gate settings for the backbone and outer barrier gate that allow a small current flowing through the device. (a) Joint gate measurements per angle showing the joint gate voltage as a function of the current through the device. At the angle $\theta = 30°$ the pinch-off curve shows little variation and a quick change from pinch-off to saturation, indicating an unsuitable setting for the voltage. At the angle $\theta = 45° - 52.5°$ the curve shows step like features indicating the desired graduate change in current. A joint gate value yielding a small current $\approx 10 - 15nA$ should be chosen. From these plots it is possible to chose the joint gate value alone, for intuition, we can plot the rays as they would be appear in a 2D map as in (b). For reference (c) full measurement implemented as a 2D sweep, showing a typical corner plot with no current at very negative gate values (pinch-off) and a small current in the turquoise area.

Figure 15 shows the procedure of the ray based sweep. Here it is used in order to set the quadrant defining gates, close to pinch-off, such that there still is a small current flowing, but not with as little noise as possible. Choosing from the plotted rays this means an individual value of $\approx -0.8V$ for the backbone and barrier gate respectively.

5.2.4.2 *Modified* `do1d`

Per default the voltage sweep is executed over the voltage range the user decides in the `do1d` function. Often this causes a too large sweep as it is more beneficial to look at larger range and see the whole picture instead of missing out on interesting features. In order to save computation time, I have implemented a modified version of the `do1d` function. This modified function stops measuring after a specific number of points don't show further change in current, i.e. when the gate is fully pinched off and the current either goes to zero or saturates. This is done by measuring the variance in the current values and stopping the measurement at a user defined threshold which defines the amount of steps that are enough to determine that the current has pinched-off or saturated. The modified `do1d` takes the stop threshold has an additional argument. Although the function was not used for the measurements in this thesis, it can be easily implemented for future measurements and thereby make the measurements more efficient.

6

MANUAL COLDSTART TUNING WORKFLOW

This chapter demonstrates the necessary steps from loading a device into the dilution refrigerator to a tuned double quantum dot and how they are implemented manually by the experimenter. The auto-tuning algorithm suggested in this thesis, is based on the manual steps usually implemented by a human.

The goal of tuning a quantum device is to deplete the 2DEG in a way that single or double quantum dots are formed. Since the geometry of different devices can vary and factors such as fabrication variances, impurities in the material and defect gates can exist, there is no universally applicable gate voltage configuration that works for all devices. Even for the same device, voltage configurations can slightly differ in different tuning cycles. It is therefore necessary to tune the voltages on every single gate of a device after cool down. A common manual strategy and the one used for the later following automation is described in the following.

6.1 PRELIMINARY WORK

In the course of my master thesis, I have not fabricated my own device, but used the previously described GaAs device. Therefore I won't elaborate on fabrication processes. Next steps, that I was lucky to be able to watch on a SiGe device, but will not be covered in detail in this thesis, is the bonding process, where the device is physically connected to a circuit board which then can be loaded to the dilution refrigerator and used to connect the ohmic contacts and gates of device to the rest of the measurement system. These processes can not be automated as they require manual human work.

6.2 BOOTSTRAPPING

Once the device is loaded to the dilution refrigerator, it is cooled down to mK in order to ensure the least thermal fluctuations that could interfere with quantum effects such as electron tunnelling and allow the precise control and manipulation of quantum states. The gates have to be connected via the QDAC and BNC coaxial cables with the breakout box. This will make sure that each gate can be controlled with an individual voltage (see chapter 4). In the following steps the device will be configured and analyzed. This process is often called *bootstrapping* as it starts with little knowledge about the device with the aim to fully characterize it.

6.2.1 *Manual configuration input*

In order to start the tuning process some information about the device is required. This includes information about the geometry of the device to know which gates have which functions. Depending on a gates function different results for the subsequent steps are expected. Furthermore, it is in some cases possible to replace a non-working gate with another gate of the device, meaning the device is not entirely useless even if one or more gates are not operational. Lastly, it is necessary to set up a configuration file that can be loaded to the measuring code in python. This file tells the measuring computer which channel on the QDAC is connected to which gate, the measuring unit (voltage), measuring limits and measuring scale.

6.2.2 *Ohmic contacts check*

The first step is to verify the functionality of the ohmic contact in the device and ensure that a current is flowing through it. In the simplest setup, a current pre-amplifier that amplifies and converts a small electrical current into a voltage signal is connected to one ohmic contact that is then readout by a digital multimeter and a bias is applied to another one, such that a current is flowing from one ohmic contact to the other. At this stage, all gates have zero gate voltage applied, but are already turned on. By sweeping the bias voltage (V) and measuring the resulting current (I), we can determine if the current is running through

the device as expected and if the ohmic contacts are functioning properly. We typically sweep in a voltage range from $[-1, 1]mV$. In an ohmic contact, which follows ohms law, the slope of the I-V graph should be a straight diagonal line. From here the experimenter can set the bias voltage depending on the desired current magnitude and direction through the device.

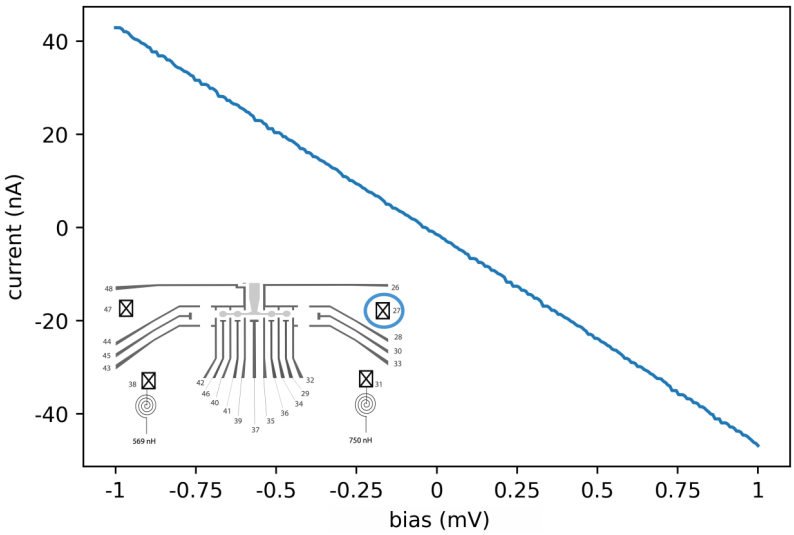


Figure 16: Bias sweep with expected ohmic behavior. Here the bias is set to the ohmic contact 27, circled in blue and the current is sent to the ohmic contact 31, in the lower right while all other gates are set to $0V$. From this plot the bias value used for further tuning can be chosen depending on which current should flow through the device.

6.2.3 *Gate characterization*

The next important step is the gate characterization to test if each individual gate is working or not. Each gate is swept over a large voltage range, typically from $[0, -1]V$ resulting into an $I_{sd} - V_g$ curve showing which gates can pinch off the current flow at which gate voltage, these traces are called pinch-off curves. As each gate is characterized individually, this step is implemented as 1D measurements. From this measurement it is then possible to determine if a gate is functioning or not, the characteristics of a healthy gate vs a broken gate are explained in 5.2.2.1. Furthermore, the shape of the pinch-off curve can show

non-linearity of the gate's behavior. Non-linearities can be caused by quantum effects or electron-electron interactions

Taking the geometry of the device into consideration, it is possible to determine if a device can be used for further tuning or not. The gates of a device can have different functions, if one of the gates that is needed for the further tuning is not showing the expected pinch-off curve, that part of the device can be classified as broken. In cases of less essential gates, it is possible to use another gate to replace the specific gate or to disregard it entirely. The cause for noise in the device can for example be a failed bonding, shorted gates or gate leakage and is often difficult to determine. In the desired case of all gates being functional, one can turn towards the tuning of quantum dots by adjusting the gate voltages individually or in combination.

A full gate characterization for the FF1A device has been carried out for the lower right quadrant in figure 17. For the finger gates/barrier gates the source is set to ohmic contact 27 and the drain to ohmic contact 47, for the sensor dot gates the current is flowing to ohmic contact 31, while the bias remains set to ohmic contact 27. This quadrant is fully functioning and is fit to be used to form a sensor dot. As a comparison the pinch-off curves of two broken gates in the upper left quadrant and upper right quadrant can be seen in figure 18. The two cases have to be treated differently: The gate in figure 18b is not essential for the quadrant to be working as there is the possibility to use the neighboring gate, indicated as gate 13 in figure 13 as a replacement. The gate in figure 18a on the other hand is needed for pinching off the area to form a sensor dot in the upper left quadrant and cannot be replaced by the neighboring plunger gate, indicated as gate 10, as it is built as a plunger gate and has a different position compared to a barrier gate, thereby the plunger gate would not be strong enough. After running the classification, this quadrant can directly be sorted out for further tuning attempts. This example shows the importance of knowing the devices' geometry to make a well-informed decision regarding the further tuning steps.

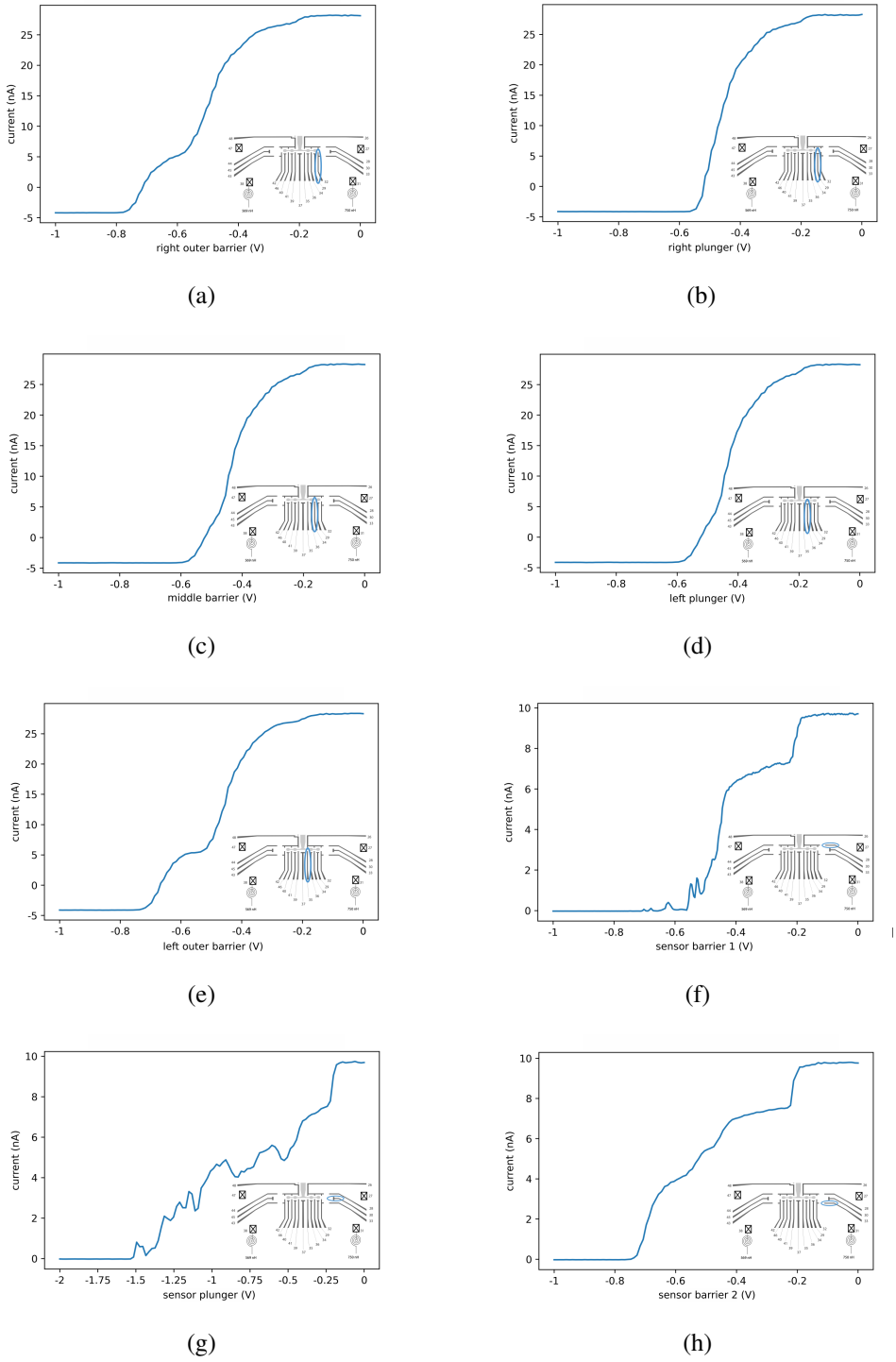
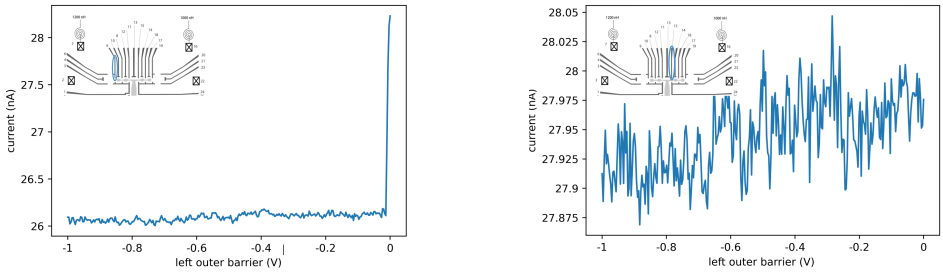


Figure 17: Full gate characterization of the gates in the lower right quadrant of the FF1A device. Here all gates are working and can be used for further dot tuning. Note that for the plunger in (g) the voltage range goes from $[-2, 0]$ as the plunger is more withdrawn from the current and therefore pinches off at more negative voltages.



(a) (b)

Figure 18: Current traces of two broken gates. (a) is an essential gate and cannot be replaced by another gate. (b) can be replaced by the gate next to it (labeled 13).

6.3 QUANTUM DOT TUNING

In this section, I will use the FF1A GaAs device to explain the tuning process, but it can be applied to other materials and various different device geometries. Depending on the device's geometry, the experimenter has to focus on one area of the device to form quantum (double) dots to make a qubit. Here the first step is to set the gates defining the tunable qubit region, the so-called backbone gates to a highly negative value, typically at $-1V$, to make sure that the current runs on the desired path and no current will be going out of that particular area. Another way of setting the backbone gate values is to take the pinch-off point from the gate characterization.

6.3.1 *Sensor dot tuning*

For the sensor dot a single quantum dot is formed, which can later be used for charge sensing, as explained in section 5.1.2. Even with the broken upper ohmic contact, it can still be tuned by to a sensor dot, by allowing a current to flow across the device, as demonstrated in section 6.3.2.1. However, considering the devices' geometry, it is not ideal to send the current this way due to the outer sensor barrier gate being located somewhat away from the current path. As a result, it is more difficult to have an effect on the electrons and effectively form the sensor dot in this region and it doesn't reflect ideal conditions. We will therefore use the lower right quadrant to demonstrate the ideal case of tuning a sensor

dot. However, for the purpose of demonstrating charge sensing using reflectometry, we will use the upper right half with the sensor dot formed as shown in 22. This choice is due to other gates in the lower half, which are needed to form a double dot, not functioning, and better double dot results can be obtained in the upper right quadrant.

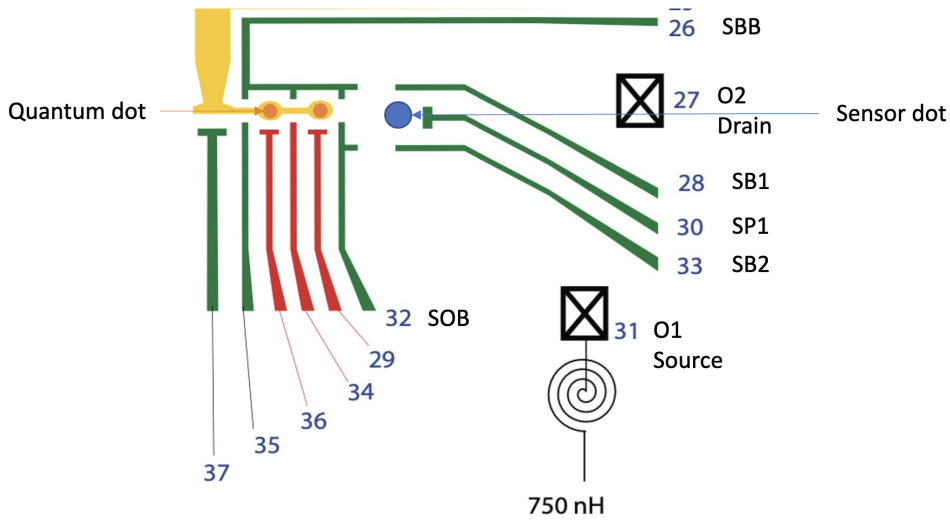


Figure 19: FF1A device lower, right quadrant. Sensor dot barrier gates SB1 and SB2 and sensor dot plunger gate SP1 in the middle which form the sensor dot. The blue dot marks the sensor dot area and the orange dots quantum dot areas.

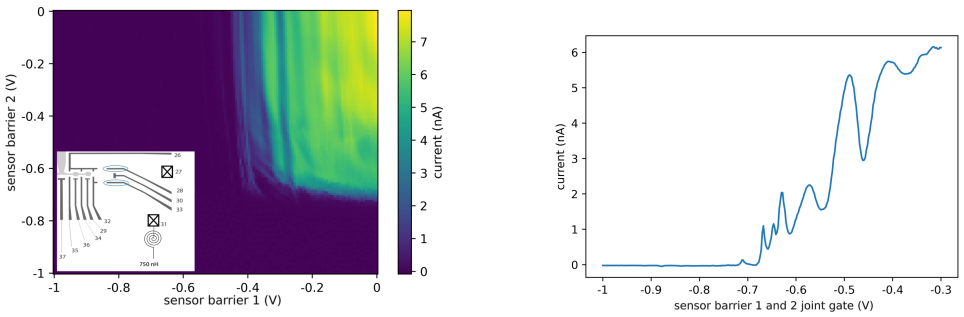
A current is sent from source (O1) to drain (O2). The aim is now to deplete the 2DEG such that a sensor dot is formed. The following steps need to be executed:

1. Set the backbone gate (SBB) to a negative voltage, typically $-1V$, to ensure full pinch-off. Also, set the outer barrier gate (SOB) to a negative value close to the pinch-off value, determined from the individual gate characterization. These settings make sure that the 2 DEG is depleted in the sensor dot region and prevent electrons from getting into other areas of the device.

2. Set the plunger gate (SP1) to a slightly negative value near to its saturation point. This ensures a sufficient effect for the plunger gate, without depleting the 2DEG too much. Setting the plunger gate to a voltage value close to the saturation point can make the sensor dot more stable and thereby less sensitive to fluctuations in the nearby dot. Stability is crucial for step 3 in the sensor dot tuning process to ensure precise and accurate measurement of the conductance in the sensor dot. The

pinch-off and saturation values for step 2 can also be obtained from the individual gate characterization carried out beforehand.

3. Sweep outer barrier gates (SB1 and SB2) either as a two-dimensional sweep or with the ray based approach. The desired outcome is to observe Coulomb oscillations. In a 2D map these can be seen as a stripy pattern of areas of conductance followed by areas of no conductance as depicted in figure 20a. If the gates are swept with the combined gate method, Coulomb oscillations should be observed as shown in figure 20b.



(a) (b)

Figure 20: Barrier barrier sweeps of the two sensor dot barriers with bias value $V_{sd} = -0.25mV$. (a) 2D map showing interesting features for sensor 1 around $-0.4V$ and for sensor 2 at $-0.6V$. (b) 1D sweep of the joint gate at angle $\theta = 50°$, interesting oscillations show at a joint gate value between $-0.7V$ and $-0.6V$.

4. a) If using the 2D scan SB1 and SB2 are to be set to a voltage value in the regime showing Coulomb oscillations in form of the stripy pattern.

   b) If using the optimized ray based approach, the optimal angle is selected based on the number of oscillations observed. The joint gate value is then set to the steepest slope of a Coulomb peak, allowing the extraction of individual gate voltage values for SB1 and SB2.

5. Sweep the plunger gate SP1 alone. Now, if the barrier gates are in the right regime, clear Coulomb oscillations are expected. If this is not the case, other values for SB1 and SB2 (step 3) are to be applied and step 4 is to be repeated. If good Coulomb oscillations are observed the sensor dot tuning was successful and the configuration

can be used for further quantum dot tuning which will be explained in the next section.

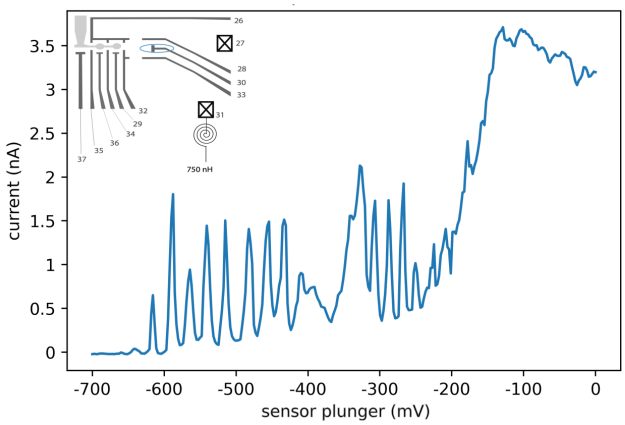


Figure 21: Plunger sweep of the sensor dot, clear Coulomb oscillations are visible in the voltage range $\approx [-600, -400] mV$ confirming the presence of a quantum dot. Here the bias is lowered further and set to $-0.2 mV$

6.3.2 *Double dot tuning*

In this section I will describe the two different methods of tuning the sensor dot: in DC transport and with charge sensing.

6.3.2.1 *transport*

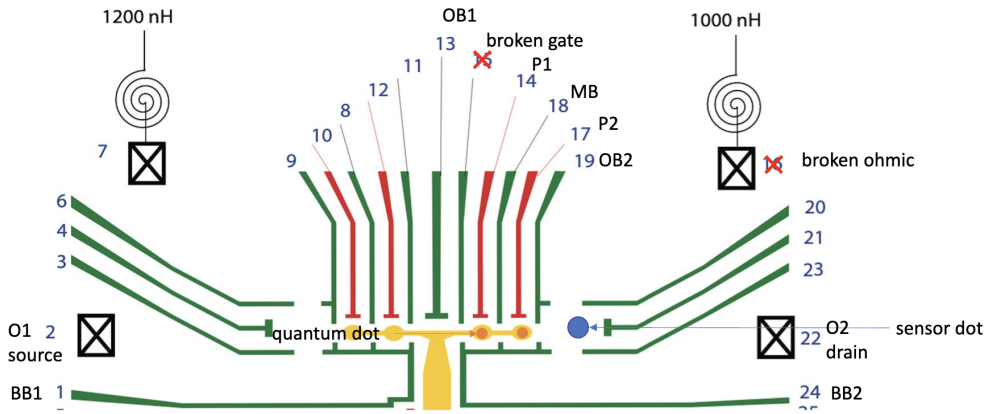


Figure 22: Split-gate structure of the FF1A device upper half. Here the gates and ohmic contacts important for the double dor tuning are labeled. The blue dot marks the sensor dot area and the orange dots quantum dot areas. One of the barrier gates as well as the upper right ohmic contact is not functioning and needs to be taken into consideration when tuning the device.

A current is sent *in transport* from source (O1) to drain (O2) past the gates which are meant to be used to deplete the 2DEG in such way that a double quantum dot is formed. For simple *in transport* tuning, the tuning steps are the following:

1. Take the pinch-off values from one-dimensional individual gate scans and set the backbone values to a negative value such that the 2DEG is depleted in the tuning area.

2. Set the plunger gates P1, P2, P3 to slightly negative values close to their saturation points. Setting the plunger gate voltages to a value in this regime secures that the potential energy of the potential dot is more stable and less sensitive to changes in the barrier gates. This ensures a more accurate result when analyzing the electron transport through the potential dot when sweeping the two outer barriers (see next step).

3. Sweep the outer barrier gates (OB1) and (OB2) together in order to see at which gate voltage combination quantum behavior can be observed. This sweep can also be done as a one-dimensional joined gate sweep.

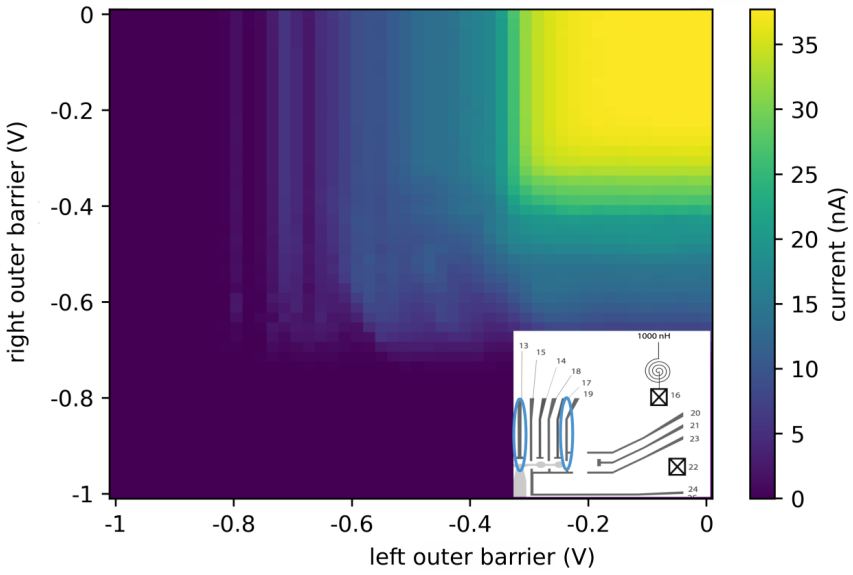


Figure 23: 2D barrier-barrier sweep of the two outer barriers for the double dot. With a bias voltage $V_{sd} = -0.3mV$. Interesting features can be seen in the turquoise area, particularly at a gate voltage in the interval $[-0.5, -0.8]V$ for the right outer barrier and $[-0.6, -0.8]V$ for the left outer barrier. The dark blue area shows gate voltages where the quadrant is fully pinched-off, meaning no current is flowing. The yellow are in the upper right corner shows where the current is fully saturated.

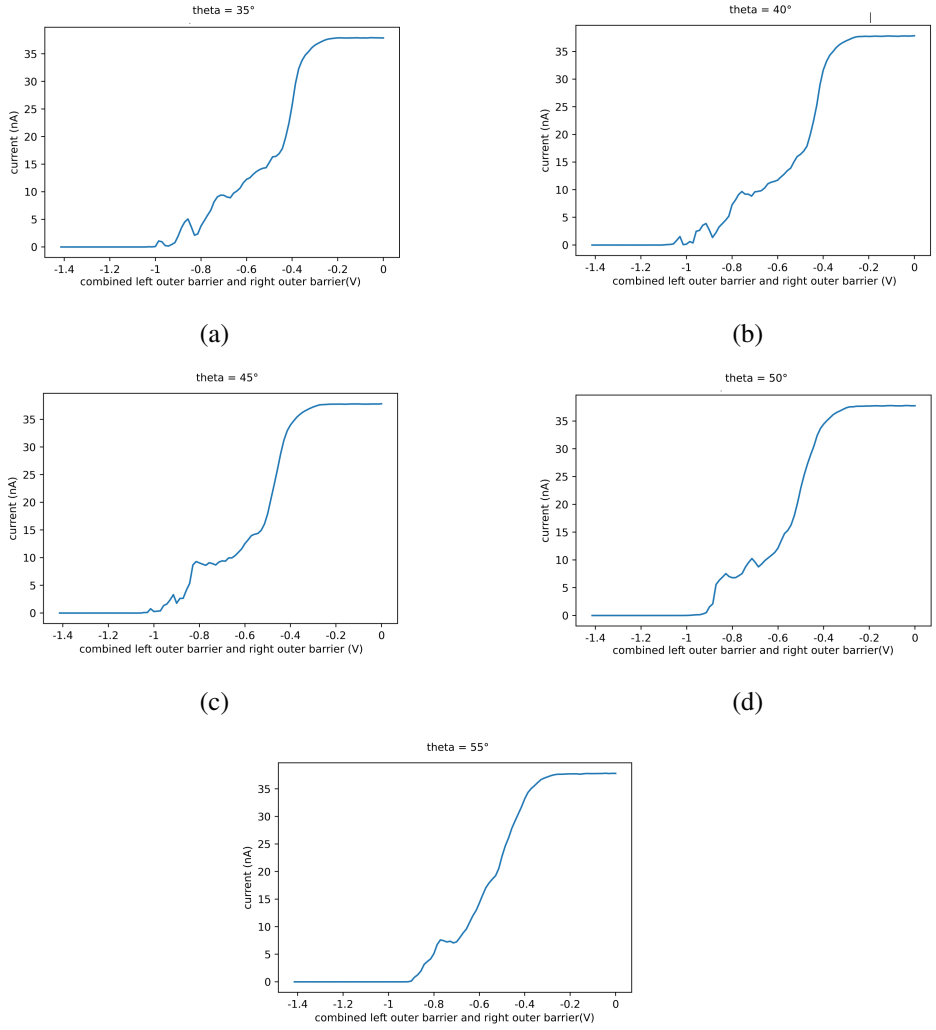


Figure 24: Same barrier-barrier sweep as in 23, but with the ray based method combining the two barrier gates to a single ray gate. The ray angle θ is varying from 35-55°. $\theta = 35°$ shows oscillations that could indicate dot presence.

4. From the above plots, take the voltage values to set the barrier gates. Using the ray based sweep: The gate combination in figure 24b looks to have promising oscillations, choosing this ray and park the joined gate at $-0.9V$, which yields individual values for the right outer barrier gate at $-0.58V$ and the left outer barrier gate at $-0.69V$.

   For analysis using the 2D map: the right outer barrier gate at $-0.58V$ and the left outer barrier gate at $-0.69V$ matches well with the interesting stripy area in this plot.

5. Sweep the two plunger gates to observe quantum dot behavior.

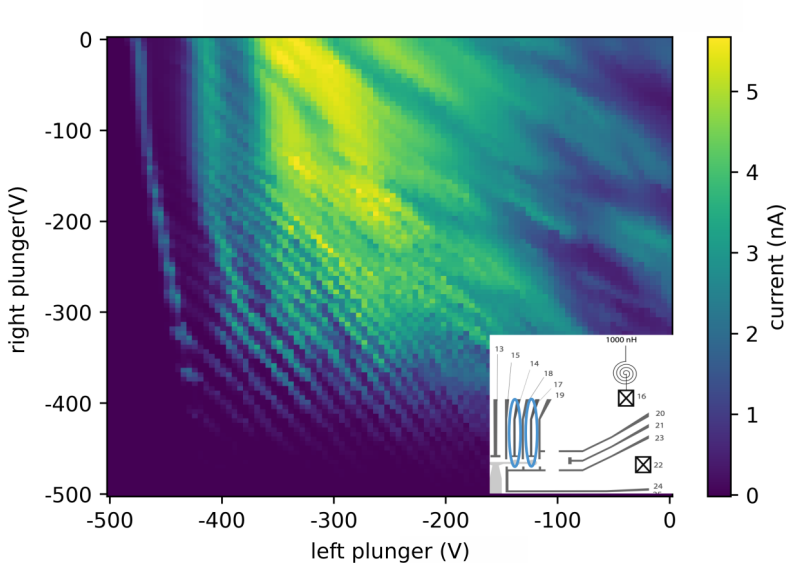


Figure 25: Sweep of the left and right plunger gates tuned to quantum dot regime. The stripy pattern clearly shows Coulomb oscillations and indicates the presence of a single quantum dot in this voltage regime.

6. Tune up middle gate between the two plunger gates to a higher negative value in order to split quantum dot into a double quantum dot. Here it can be useful to try different middle gate voltage values in order to find the best splitting.

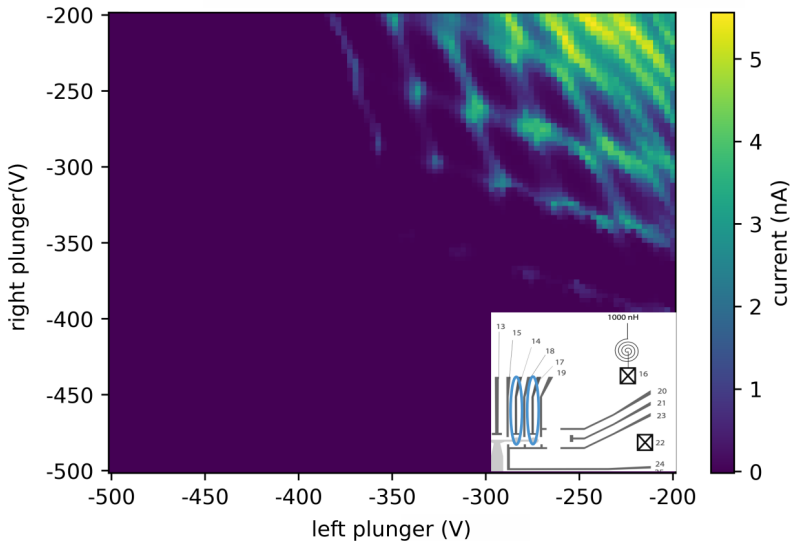


Figure 26: Retake of the plunger-plunger sweep as in figure 25 but with the middle barrier gate tuned up to a more negative value, $V_{MB} = -0.4V$ in this case, resulting in a larger effect for the middle gate that is able to split up the single dot into a double dot. The voltage range for the chosen scan is slightly modified as well to only focus on the area showing interesting features.

6.3.2.2 *Charge sensing with reflectometry*

The method of charge sensing has been explained in section 5.1.3. In section 5.1.2 the implementation and steps that are performed when tuning a sensor dot where explained more detailed. This section will show an applied example of charge sensing at the FF1A device.

Using a sensor dot for charge sensing is the preferred method to detect quantum (double) dots and is therefore the first tuning step in an automation process after the bootstrapping of the device.

1. Tune the sensor dot as described in section 6.3.1 in order to later use it for detecting the charge of the nearby double dot by measuring the current flowing through this dot.

2. Tune the double dot as described in section 6.3.2.1, but this time measuring the the demodulated voltage of the sensor dot. The sensor dot has been slightly retuned for

the reflectometry setup. The voltage settings from the DC transport tuning can be used as a guideline, but not be adopted blindly.

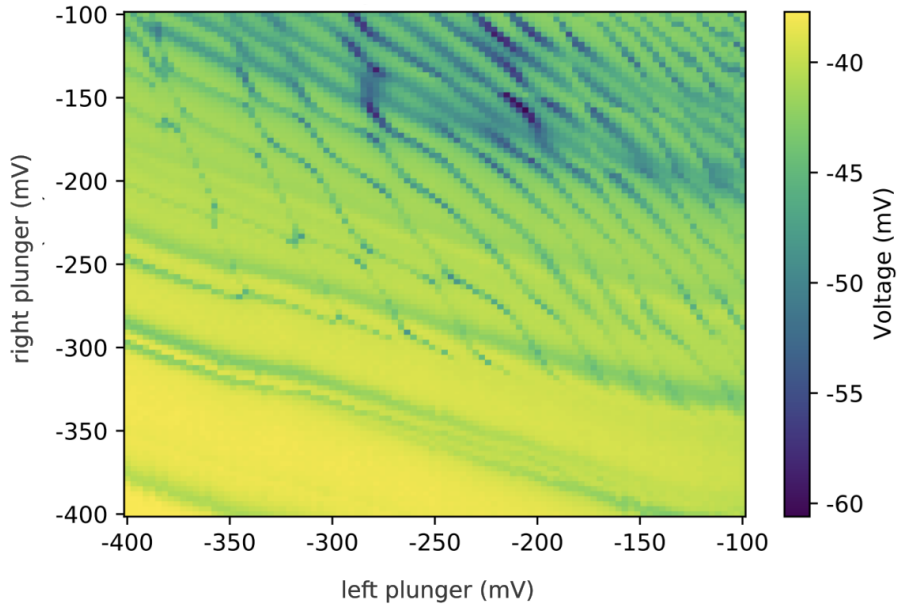


Figure 27: Same plunger-plunger sweep as in 26, but data acquired with RF reflectometry. For the reflectometry measurement the plunger gate of the sensor dot had to be increased slightly to not fully deplete the 2DEG and is at $V_{SP1} = -0.15mV$

Part IV

AUTOMATION

7

AUTOMATION

So far it is not possible to initiate the tuning procedure without any prior knowledge of the device. Basic information about the devices geometry, as explained in chapter 4.2 and 6.3, are necessary for the successful implementation of the analysis presented in this work. The following section relies on the assumption that device geometry is known and can be given as a user input to the automation functions. The full process has been outlined in figure 28 and will be further elaborated upon in the subsequent sections.

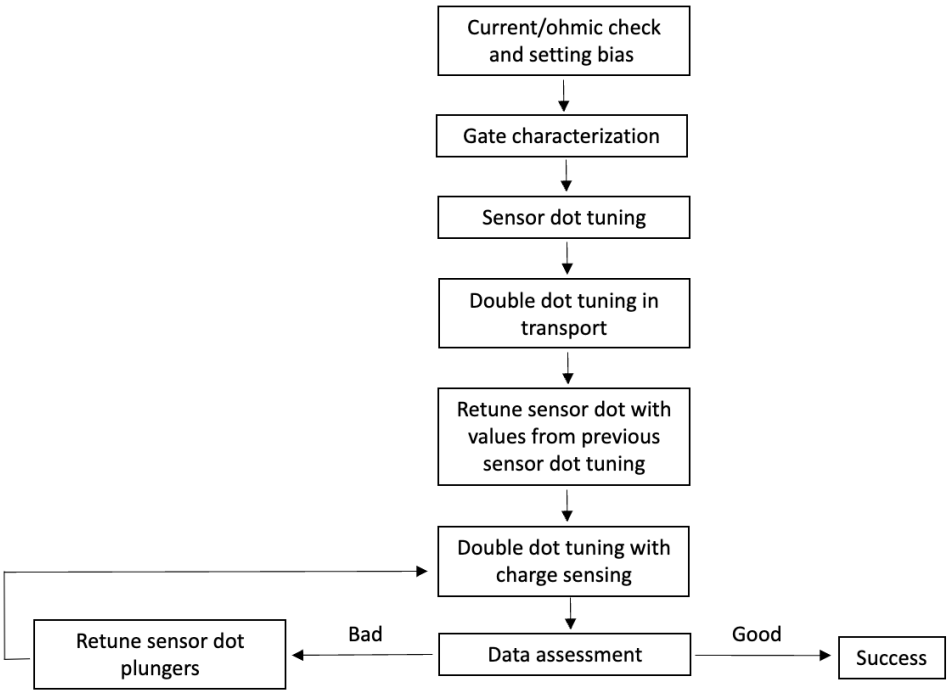


Figure 28: Full automated tuning process after setting up the device for measuring. The first two steps are implemented together in one single function. The output from the gate characterization is then used as input for the separate dot tuning steps. These steps are currently run by the experimental scientist one after the other, but they can also be combined in future into one single function. The data assessment and retuning steps still require further refinement to achieve a fully automated process.

7.1 CURRENT/OHMIC CONTACT VERIFICATION

This step is part of the preliminary work, but can easily be automated to maximize the capabilities of the automated cold start tuning. As the preliminary measurements to define the quantum dots require a current running through the device, the bias for the ohmic contact needs to be set in such way that a small positive current is reached. In this regime, measurements in current can still be performed, but the Coulomb blockade regime is maintained. We take a bias voltage sweep to test for ohmic behavior. We also set a target value for the desired current to flow through the device. The target value can vary and was set to $\approx$ 25nA for this tuning, based on prior measurements. We can determine the bias

value at which this current is reached by simply reading the plot. Finally, we set the bias to exactly this value. Furthermore, at this step it can be determined if the ohmic contacts are working (as in, they conduct current through the device in an "ohmic" fashion). As explained in chapter 6 a straight line is expected as can be seen in figure 29.

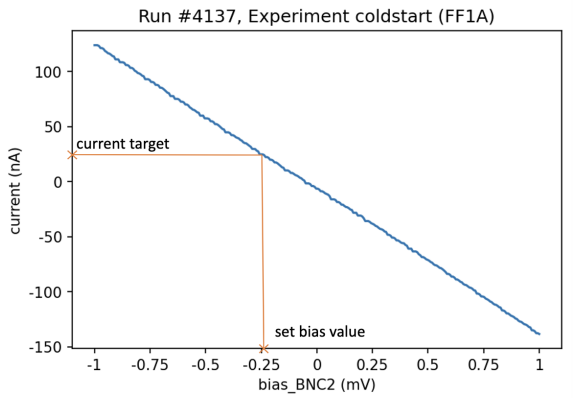


Figure 29: Bias value that will be chosen automatically in order to get a slightly positive current. The desired current can be defined as a target value and the bias will be set to the corresponding voltage value.

It is important to note that the set bias value should not be considered final and may require adjustment during the retuning steps if the final result does not exhibit the desired behavior.

7.2 INDIVIDUAL GATE CHARACTERIZATION

The first step towards an automated tuning is to automate the characterization of gating function of each individual gate. This step allows us to determine if a device will be suitable for the tuning to QD or DQD regime. The current through the device for each gates is measured one-by-one by applying a gate voltage ranging from $[0, -1]V$ and $[-1, 0]V$. The obtained current and voltage values from the $[0, -1]$ measurement are then used for further analysis.

7.2.1 *Gate functionality and classification*

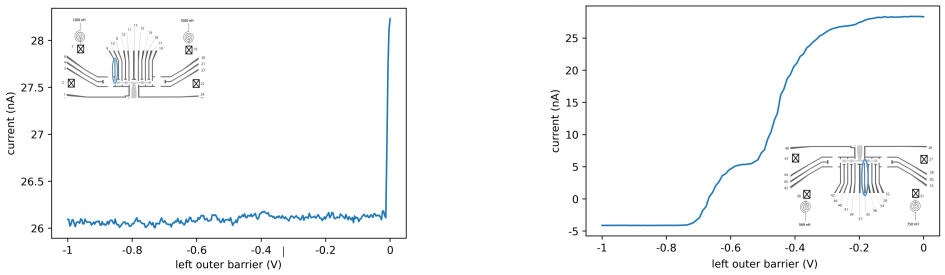
We take the gradient of the current and analyze the peaks of the current curve in order to classify a gate into *working* or *not working*. The gradient of the current is first smoothed with a Gaussian filter and then fit with a cosine hyperbolic function

$$f(x, a, b, x_0, y_0) = \frac{a}{\cosh^2 b(x - x_0)} + y_0 \tag{31}$$

where the amplitude a, slope b, offset in the x-direction x_0 and offset in the y-direction y_0 are extracted and can be used to classify the gate further: The amplitude shows the magnitude of the current variations observed during the gate sweep, where a high amplitude indicates a gate with stronger current response or larger charge fluctuations. The slope can tell us how strongly the current through the device responds to changes in the gate voltage. This may also be useful in future work in data-aware design of devices, as some gates (the plunger gates) need to have a stronger response, or "lever arm", on the quantum dot, while others (the outer barrier gates) do not. Automated characterization, feedback and collection of statistics on gate function can therefore be very useful to refine designs. The offsets in the x-direction are useful to find the locations where charge states or transitions occur. Additionally, the number of the peaks as well as the peak widths are analyzed, which give information about the gate's energy levels. To determine if a gate is working or not, the median absolute deviation of the current signal is taken. The median absolute deviation computes the median of the absolute deviations from the median of the data set

$$MAD = median(|X_i| - \tilde{X}) \tag{32}$$

where $\tilde{X}$ is the median of the data set [45]. If the current has a wide enough distribution, depending on a predefined threshold, the gate is classified as *working*. The threshold is selected by analyzing previous scans of working gates. For a non-working gate, we typically observe only noise, without any characteristics or dispersion, thereby the method can also be used to classify non-working gates.



(a) (b)

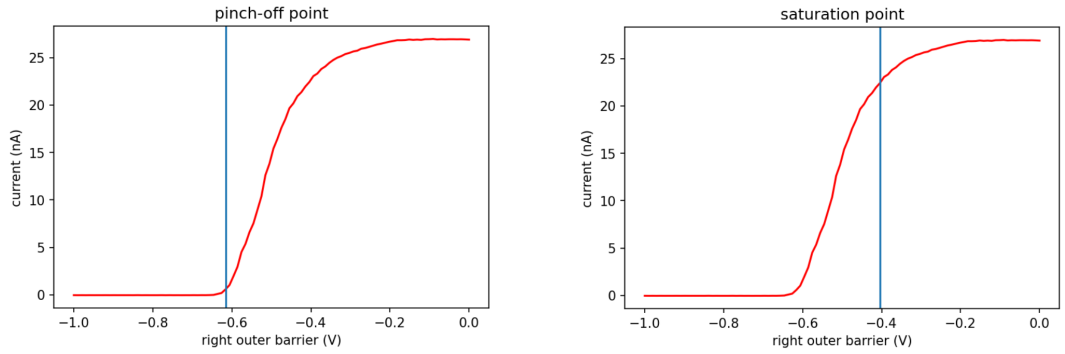
Figure 30: Broken vs. functioning gate. (a) is only showing noise whereas (b) is showing the expected current-voltage behavior for a functioning gate.

7.2.2 *Pinch-off and saturation values*

For further tuning of the device we are interested in the voltage values at the pinch-off and just before saturation point. Qualitatively the pinch-off point can be described as the point where the pinch-off curve goes from nearly flat to a very steep slope, where we are interested in the highest curvature in the graph. The saturation point is the point where the sigmoidal shaped signal first starts to change from increasing slope to a decreasing slope. These values are defined by and can be extracted from the one-dimensional pinch-off curves using the following two methods:

To find the pinch-off point of a gate, we take the second derivative of the current signal and find its maximum. We first compute the prominent peaks above a predefined threshold. Based on the test gates we used in our analysis, a threshold of 1 seems to be sufficient. Once we have found all prominent peaks, we can use the location of the maximum in the second derivative to determine the pinch-off value.

For the saturation point value, we fit the signal to a hyperbolic tangent function $f(x) = a(1 + \tanh(bx + c))$. We then take the second derivative of the fitted function. The point where the second derivative reaches a minimum is where we define the saturation point value. This is the point where the signal's slope transitions from increasing to decreasing.



(a) (b)

Figure 31: Pinch-off and saturation values found by analyzing the pinch-off curves using fitting methods. The pinch-off and saturation points are saved in a python dictionary and can directly be used for the subsequent device tuning.

The section from pinch-off point to saturation point is also the only region that is relevant for further analysis as this is the operating range where the gate voltage controls the electron flow through the device. Therefore, the remaining parts of the curve can be disregarded.


```
{qdac_BNC12': {'gate_working': True,
'num_prom_peaks': 0, 'prom_peaks_widths': array([],
dtype=float64), 'Peak_info': {'Peak0': {'a':
9.097458314439458e-08, 'b': -6.516889094656409,
'x0': -0.5146295532066292, 'y0':
-7.759389118617895e-10}}} 'Pinch-off':
-0.635451505016722, 'Saturation point':
-0.41137123745819404,'Pinch-off':
-0.655518394648829}}},
'qdac_BNC9': {'gate_working': False,
'num_prom_peaks': 0, 'prom_peaks_widths': array([],
dtype=float64), 'Peak_info': {'Peak0': {'a':
1.2224721260006287e-09, 'b': -3.885228737954161,
'x0': -0.318837616020866, 'y0':
-1.1129553646013182e-09}, 'Peak1': {'a':
8.249211659906074e-10, 'b': -9.770043544108585,
'x0': -0.511391937510608, 'y0':
-5.476236426389475e-10}, 'Peak2': {'a':
2.8765823937526878e-09, 'b': -0.4999999977180205,
'x0': -0.5000000408031976, 'y0':
-2.8509495137484434e-09, 'Saturation point':
-0.979933110367893, 'Pinch-off':
-0.892976588628762}}}}}}
```


Figure 32: Example of a working and broken gate output from the analysis. The gate labeled 'qdac_BNC9' corresponds to the broken gate from figure 30. Apart from the 'gate_working': False status, the analyzed values do not add any value as the gate is broken and won't be used for further tuning, but are kept for completeness. Currently, this "True" or "False" output is used by the human experimenter to run the next path of the tuning process; in future, the output could be passed to the functions in the next section directly.

7.3 AUTOMATED SENSOR DOT TUNING

The output from the gate characterization is used as the starting point for the sensor dot tuning, which is the next step in tuning a double quantum dot. The aim of the automated tuning algorithm is to automate the manual steps of the dot tuning, as described in chapter 6, such that a single function outputs a functioning sensor dot. It is still necessary for the user to input manually the geometry of the dot, i.e. which gate has which function (e.g. backbone, barrier, plunger). In future, the design file of the gate geometry could be passed to the tuning module, such that it could make intelligent assumptions about the gate functions. This was however outside the scope of the present thesis.

For the device geometry that we investigate in this thesis, the first stage is to initialize the quadrant of interest and to deplete the 2DEG in the sensor dot region. The backbone gate can simply be set to $-1V$ to ensure full depletion of the 2DEG in the quadrant and isolation of the (nominally) four double quantum dot spin qubits from each other. From the gate characterization dictionary, the pinch-off and close-to-saturation points can be extracted and used to set the outside barrier (close to pinch-off point) as well as plunger (close-to-saturation point) gates.

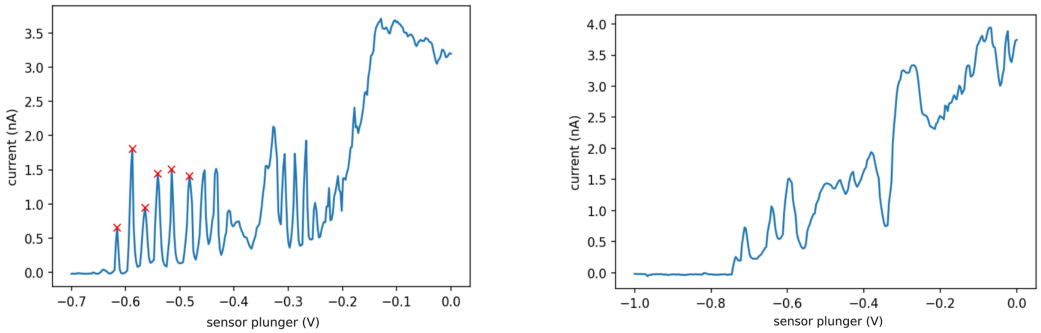
With the initializing conditions, it should be possible to form the sensor dot in the selected quadrant. A barrier-barrier sweep provides information about how the two outside barrier gates function together. We chose to take this measurement with a ray-based approach, as suggested in [46] and described in detail in section 5.2.4.1 in order to make the algorithm faster. This entails a one-dimensional measurement along an axis suggested by the algorithm (i.e. a ray), instead of the traditional scan favored by a human experimenter: a two-dimensional scan in the two barrier gate space (forming the x and y axes). The two-dimensional sweep is more suitable for human intuition, which is not relevant anymore for automated tuning. We take five ray measurements spread in angles $\theta = [35, 40, 45, 50, 55]^\circ$, each measurement is further analyzed for Coulomb oscillations. For this purpose, we take the current signal of each of the joined gates and count the number of peaks present. The best angle is chosen based on which ray shows the most oscillations and taken on for further analysis where the algorithm picks the most prominent peaks and determines the slope for each peak. Finally, the peak with the steepest slope is chosen, the steeper the slope, the greater is the response to small variations in the gate

capacitance, such as the rearrangement of electrons forming the qubit in the quantum dot, allowing precise control of the qubit. The midpoint of this peaks' slope is taken in order to set the common voltage value for the joined gate, which can be translated into two individual voltage gate values for each barrier gate respectively.

The final step of the sensor dot tuning is the sweep of the plunger gate to check if the measurement exhibits Coulomb oscillations. The plunger is swept in the range $[0, -1]V$, again we take the most prominent peaks in the current signal dependent on a user defined threshold. We then use two criteria in order to classify the Coulomb oscillations:

1. *Oscillations start and end at around zero current.* We analyze the peaks in the current signal and determine if the minima to the left and right of the peaks drop close to zero, where we define the close-to-zero value to be $0.2nA$. We further define the number of peaks where we would like this condition to be true, where currently this value is set to be five peaks based on the experiments we have implemented in the lab. This values still has to be defined further based on more statistics and analysis. If there are at least five of such peaks the first criterion for a *good oscillation* is fulfilled.

2. *Regular and equal spacing between the peaks*. To determine this, we look at the distance x between the location of the peaks. The distances should all be about the same or close to the mean distance. To classify as *equally spaced* and *regular*, we define a threshold and check if the standard deviation of the distances is less than this threshold. Here, we use a threshold of 0.02, based on the experience from the lab. For the purpose of developing a versatile and robust algorithm, as for the previous criteria, this threshold value should be based on analysis of more data.



(a) (b)

Figure 33: Classification of Coulomb oscillations of a sensor plunger gate sweep after tuning attempts of the outer barriers to quantum dot regime. (a) Oscillations classified as "good", the peaks are evenly spaced and go nearly down to zero. (b) Oscillations classified as "bad", even though oscillations are present, they are very irregular and don't reach low current values ($< 0.2nA$)

We then classify the signal as *good oscillations* or *bad oscillations*, in the first case the algorithm stops here as the desired outcome is reached (a tuned sensor dot). If the measured current doesn't exhibit Coulomb oscillations, we iterate over the previous steps until Coulomb oscillations are present and we are in the quantum dot regime. In the case that the tuning is unsuccessful just by adjusting the two barrier values and and re-sweeping the plunger, we need to follow a different approach; these strategies are listed in section 7.5.

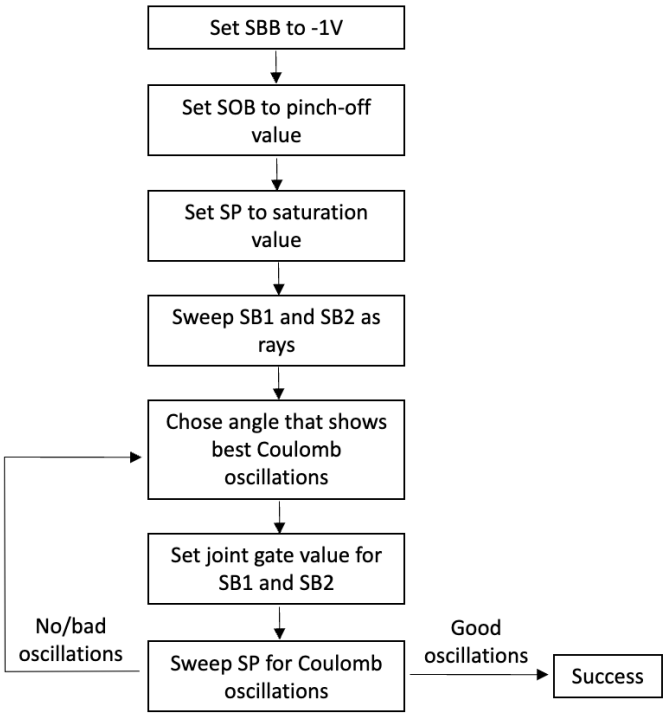


Figure 34: Flowchart for the sensor dot tuning process. The same gate structure as in figure 19 is used.

7.4 AUTOMATED DOUBLE DOT TUNING

The next natural step after automating the sensor dot tuning is the automation of the double dot tuning. This is the final goal of the device tuning: a tuned qubit where the quantum dots act as the two-level system of the qubit.

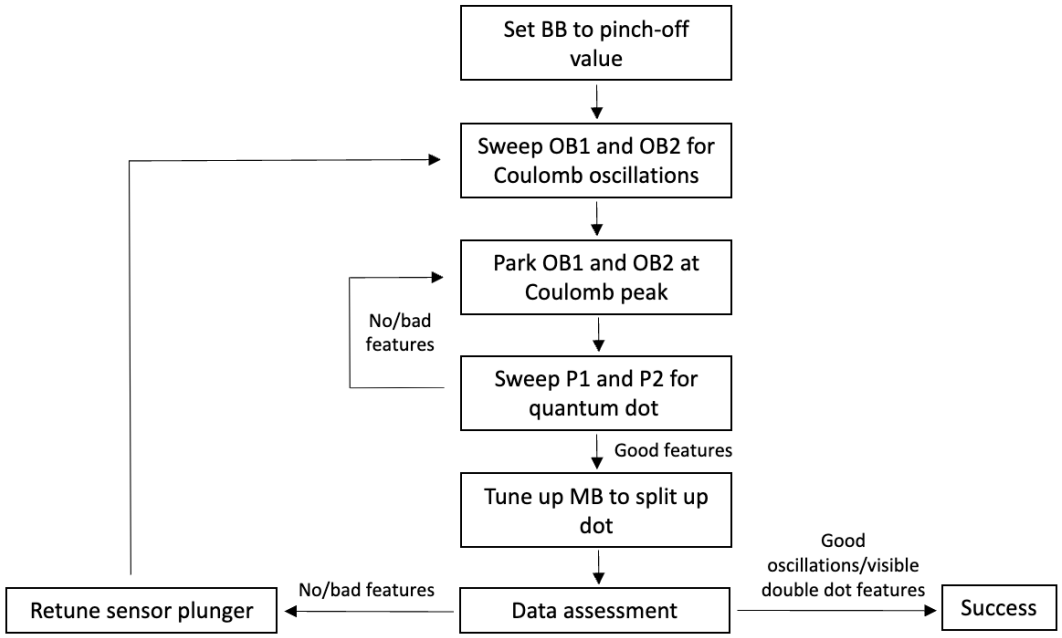


Figure 35: Flowchart for the double dot tuning process.

The above process already is clearly outlined and all functions needed for the individual steps are implemented, but have not yet been fully automated in the course of this work. For the purposes in this work, the steps have still been executed manually.

7.5 RETUNING STEPS

The following steps have been performed only with manual tuning of the device and still need to be automated. They are listed in this section as they contain suggestions for automation that can be implemented using the current automated analysis. Based on the experience from the laboratory there are three suggestions that can be followed and be implemented in the algorithm. All three involve changing one of the fixed parameters that we found during the initial analysis of section 7.3:

1. *Pinch-off point investigation.* By adjusting the outside barrier voltage, it may be possible to find the voltage configuration, where the tunneling rate is just right for the double dot regime. This step can easily be implemented by taking the pinch-off value from the analysis and explore immediate neighboring points.

2. *Saturation point investigation.* Another attempt of the barrier-barrier sweep with a varying plunger gate voltage, with values close to the original saturation-point. This can also be implemented by looping through neighboring points of the original saturation point and analyzing the resulting oscillations.

3. *Bias value investigation.* The applied voltage bias can be increased or decreased having a direct effect on the potential landscape of the qubit. In the case of a double dot, the bias voltage affects the energy difference between the two dots and the barrier between them and thereby can affect its energy states and their spacing.

7.6 CHARGE STATE CLASSIFICATION

Finally, the data quality of the double quantum dot is assessed and evaluated. For this project we started on integrating an already existing automation code as developed and used by Ziegler et al. [17] and redefined by Zwolak et al. [46] with the measurements taken in the Qdev lab. The aim is to achieve a fully automated process from a cooled-down device to a tuned and classified quantum dot. In the following section I will explain what has been developed by Zwolak et al. and show the preliminary results we got on our device using their code integrated with our measurements.

The auto-tuning algorithm developed by Zwolak et al. uses a pretrained convolutional neural network (CNN) to determine the global state of a device. To train the CNN, synthetic, simulated 2D charge-sensor measurement data of two plunger gates forming a double dot is used. The CNN consists of two convolutional layers (each followed by a pooling layer) and four fully connected layers which are used to reduce the size of the feature maps while extracting the most important characteristics of the data. The model is able to identify the state of the device for the taken 2D scan V_R and returns a probability vector

$$\mathbf{p}(V_R) = (p_{ND}, p_{SD}, p_{DD}), \tag{33}$$

containing the probability of the device being in the state of *no dot* (p_{ND}), *single dot* (p_{SD}) or *double dot* (p_{DD}). For the optimization, they first define a target p_{target} and compare the obtained probability vector with this target, also including a penalty for single-dot and no-dot regions. Afterwards they use the Nelder-Mead method to find the minimum of an objective function [47]. To do so, a set of initial points in the 2D map is taken to form the

vertices of a simplex. The simplex is then iteratively modified by changing its vertices until a minimum is found. In the case of Zwolak et al., the initial simplex is defined by the fitness value of the starting region V_R and two additional regions obtained by lowering the voltage on each of the plungers one at a time by 75 mV [46].

The model we integrated to classify the states in our measurement was taken from Ziegler et al. who used a model based on the above described CNN. In their case the state assessment returns the probability vector

$$\mathbf{p}(V_R) = (p_{ND}, p_{LD}, p_{RD}, p_{DD}) \tag{34}$$

additionally classifying left single dot and right single dot.

The data quality is assessed by comparing $\mathbf{p}(V_R)$ to a target value for the double quantum dot state and the algorithm terminates if $\mathbf{p}(V_R)$ is sufficiently close to this value. Otherwise they measure again with a different plunger voltage configuration, determined by an action vector. They define the action vector as

$$\mathbf{v}_{act} = (V_{P_1}^{act}, V_{P_2}^{act}) = \mathbf{p}(x_c)\mathbf{A}^T \tag{35}$$

where

$$\mathbf{A} = \alpha \mathrm{diag}(\mathbf{v}_{E_C}) \begin{pmatrix} 1 & 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 \end{pmatrix} \tag{36}$$

Here $\mathrm{diag}(\mathbf{v}_{E_C})$ is a diagonal matrix containing the approximate charging energies for the plunger gates determined in the gate characterization. The retuning finishes when either the target value for the double quantum dot is reached or an error is raised indicating that no double dot can be formed within this plunger-plunger configuration. In the latter case, a retuning of the middle barrier gate is performed before taking a new plunger-plunger scan.

The implementation on our end so far takes a live measurement on the FF1A device in 2D in the plunger-plunger space. For testing purposes we start out in a regime known to show double quantum dot behavior.

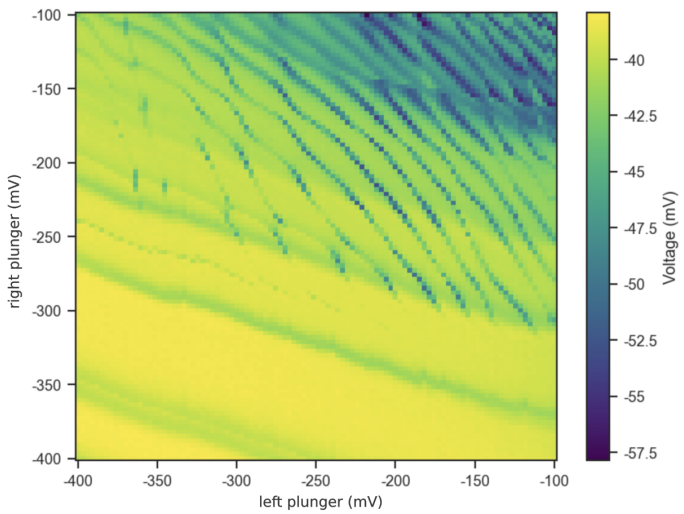


Figure 36: Plunger-plunger scan used for the quantum dot analysis taken with reflectometry.

The above 2D map is used directly for the state evaluation, where we take nine 32x32 pixel sized squares with midpoints distributed evenly in the plunger-plunger space and use the classification model on these squares. For each square, the state assessment returns the probability of each possible state to be contained in the respective measurement square. Here we distinguish between *ND* for no dot, *CD* for central single dot, *LD* for left single dot, *RD* for right single dot and *DD* for double dot. Comparing the results with our personal intuition, the class assessment is accurate and can reliably and consistently identify the states in the squares. Nevertheless it is to be noted that a known double dot configuration was taken for the state assessment and results could be different with less defined charge lines.

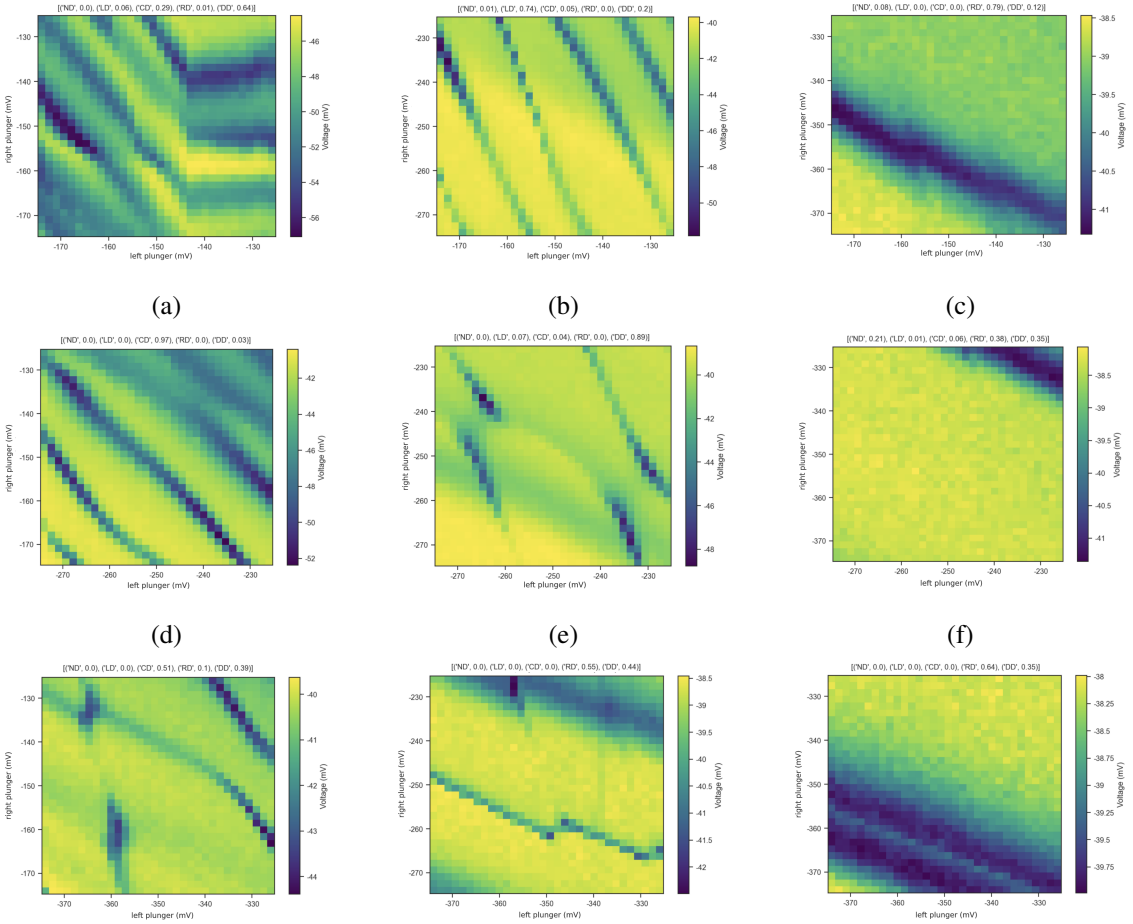


Figure 37: State classification from the evaluator model. Outputs are given in percentages for no dot ND, central single dot CD, left single dot LD, right single dot RD and double dot DD. The results are given in table 1.

Table 1

Range	ND	LD	CD	RD	DD
a	0 %	6 %	29 %	1 %	64 %
b	1 %	74 %	5 %	0 %	20 %
c	8 %	0 %	0 %	79 %	12 %
d	0 %	0 %	97 %	0 %	3 %
e	0 %	7 %	4 %	0 %	89 %
f	21 %	1 %	6 %	38 %	35 %
g	0 %	0 %	51 %	10 %	39 %
h	0 %	0 %	0 %	55 %	44 %
i	0 %	0 %	0 %	64 %	35 %

Table 1: State classification results for the ranges shown in 37 in percentage (rounded numbers). Classifications are for no dot ND, central single dot CD, left single dot LD, right single dot RD and double dot DD.

Part V

OUTLOOK

8

OUTLOOK AND CONCLUSION

8.1 OUTLOOK

In the course of this thesis, we achieved the automation of the initial steps involved when tuning a quantum device into double dot regime. This section explores the future work that can be implemented to realize a fully automated tuning algorithm.

8.1.1 *Noise detection*

The last step in the double dot tuning algorithm, denoted as *data assessment*, is directly connected to the state classification. In a non-ideal case, the state classification returns a state vector that doesn't contain DD features. In this situation, a new measurement will be taken defined by the action vector, as described in section 7.6.

In an alternative setting, the charge-stability diagram can contain noise making it difficult or impossible to classify the charge states. A noise model differentiating between moderate and high noise is used to evaluate the measurement. Based on the noise evaluation, the algorithm will chose to ignore the noise or to retune the device to achieve better results.

8.1.2 *Full automation*

At the current stage, the gate characterization and sensor dot tuning is still implemented in two steps. In the first step, a function analyzing all gates is called and returns a dictionary with the gate characterization. This dictionary is used for the second step as an input. In between these steps, currently, it is still necessary to interfere as the dot tuning requires

information about the devices' geometry, in particular the function of the gate (plunger gate, outer barrier, etc.). To optimize and automate the tuning further, the gate characterization should be able to analyze the pinch-off curve for each particular gate and have an additional entry in the output dictionary containing the gate's function. This will make it possible to directly continue with the tuning without human intervention.

8.2 CONCLUSION

In this thesis I explore the steps for getting from "having no to little knowledge of a quantum device" to "a fully characterized device" and how this knowledge can be used to tune the device into the double quantum dot regime. Furthermore, I suggest an automation technique for the characterization and tuning procedure using python.

Commencing with the necessary theoretical background knowledge about qubits and electron transport in semiconductor heterostructures, this thesis enables the reader to understand the manual tuning process of spin qubits in an experimental context. By explaining the individual steps and methods used - from loading a new device to the tuned double quantum dot - I provide an understanding of the full tuning process. Finally, the manual tuning process functions as the underlying framework for the automatic analysis and tuning, and should guide the reader to understand the logic behind the automation code going from an untuned device, via gate characterization and profound analysis to the tuning of sensor dot, single dot and double dot, to a fully tuned qubit.

My thesis shows the importance of automation in quantum computation for the development of scalable and reliable quantum computing systems and how to reduce the bottleneck in the qubit fabrication process currently present caused by the need of human expert knowledge. Automating the tuning of qubits is also a significant step in the direction of making qubits more accessible for everyone. Even though there still is work to be done for the fully autonomous tuning of qubits, the foundational stones are laid and we can expect further exciting developments in the field in the coming years.

BIBLIOGRAPHY

[1] J. P. Zwolak and J. M. Taylor, "Colloquium: advances in automation of quantum dot devices control", Reviews of Modern Physics **95**, 10.1103/revmodphys.95.011006 (2023).

[2] T. A. Baart, P. T. Eendebak, C. Reichl, W. Wegscheider, and L. M. Vandersypen, "Computer-automated tuning of semiconductor double quantum dots into the single-electron regime", Applied Physics Letters **108**, 213104 (2016).

[3] M. Lapointe-Major, O. Germain, J. Camirand Lemyre, D. Lachance-Quirion, S. Rochette, F. Camirand Lemyre, and M. Pioro-Ladrière, "Algorithm for automated tuning of a quantum dot into the single-electron regime", Physical Review B **102**, 10.1103/physrevb.102.085301 (2020).

[4] C. J. van Diepen, P. T. Eendebak, B. T. Buijtendorp, U. Mukhopadhyay, T. Fujita, C. Reichl, W. Wegscheider, and L. M. Vandersypen, "Automated tuning of inter-dot tunnel coupling in double quantum dots", Applied Physics Letters **113**, 033101 (2018).

[5] V. Nguyen, S. B. Orbell, D. T. Lennon, H. Moon, F. Vigneau, L. C. Camenzind, L. Yu, D. M. Zumbühl, G. A. Briggs, M. A. Osborne, and et al., "Deep reinforcement learning for efficient measurement of quantum devices", npj Quantum Information **7**, 10.1038/s41534-021-00434-x (2021).

[6] J. Darulová, M. Troyer, and M. C. Cassidy, "Evaluation of synthetic and experimental training data in supervised machine learning applied to charge-state detection of quantum dots", Machine Learning: Science and Technology **2**, 045023 (2021).

[7] S. Czischek, V. Yon, M.-A. Genest, M.-A. Roux, S. Rochette, J. Camirand Lemyre, M. Moras, M. Pioro-Ladrière, D. Drouin, Y. Beilliard, and et al., "Miniaturizing neural networks for charge state autotuning in quantum dots", Machine Learning: Science and Technology **3**, 015001 (2021).

[8]N. M. van Esbroeck, D. T. Lennon, H. Moon, V. Nguyen, F. Vigneau, L. C. Camenzind, L. Yu, D. M. Zumbühl, G. A. Briggs, D. Sejdinovic, and et al., "Quantum device fine-tuning using unsupervised embedding learning", New Journal of Physics **22**, 095003 (2020).

[9]H. Moon, D. T. Lennon, J. Kirkpatrick, N. M. van Esbroeck, L. C. Camenzind, L. Yu, F. Vigneau, D. M. Zumbühl, G. A. Briggs, M. A. Osborne, and et al., "Machine learning enables completely automatic tuning of a quantum device faster than human experts", Nature Communications **11**, 10.1038/s41467-020-17835-9 (2020).

[10]J. Darulová, "Automated tuning of gate-defined quantum dots", PhD thesis (2020).

[11]S. S. Kalantre, J. P. Zwolak, S. Ragole, X. Wu, N. M. Zimmerman, M. D. Stewart, and J. M. Taylor, "Machine learning techniques for state recognition and auto-tuning in quantum dots", npj Quantum Information **5**, 10.1038/s41534-018-0118-7 (2019).

[12]R. Durrer, B. Kratochwil, J. Koski, A. Landig, C. Reichl, W. Wegscheider, T. Ihn, and E. Greplova, "Automated tuning of double quantum dots into specific charge states using neural networks", Physical Review Applied **13**, 10.1103/physrevapplied.13.054019 (2020).

[13]J. P. Zwolak, S. S. Kalantre, X. Wu, S. Ragole, and J. M. Taylor, "Qflow lite dataset: a machine-learning approach to the charge states in quantum dot experiments", PLOS ONE **13**, 10.1371/journal.pone.0205844 (2018).

[14]J. Ziegler, T. McJunkin, E. Joseph, S. S. Kalantre, B. Harpt, D. Savage, M. Lagally, M. Eriksson, J. M. Taylor, J. P. Zwolak, and et al., "Toward robust autotuning of noisy quantum dot devices", Physical Review Applied **17**, 10.1103/physrevapplied.17.024069 (2022).

[15]C. Volk, A. M. Zwerver, U. Mukhopadhyay, P. T. Eendebak, C. J. van Diepen, J. P. Dehollain, T. Hensgens, T. Fujita, C. Reichl, W. Wegscheider, and et al., "Loading a quantum-dot based "qubyte" register", npj Quantum Information **5**, 10.1038/s41534-019-0146-y (2019).

[16]H. Liu, B. Wang, N. Wang, Z. Sun, H. Yin, H. Li, G. Cao, and G. Guo, "An automated approach for consecutive tuning of quantum dot arrays", Applied Physics Letters **121**, 084002 (2022).

[17] J. Ziegler, F. Luthi, M. Ramsey, F. Borjans, G. Zheng, and J. P. Zwolak, *Automated extraction of capacitive coupling for quantum dot systems*, 2023.

[18] D. T. Lennon, H. Moon, L. C. Camenzind, L. Yu, D. M. Zumbühl, G. A. Briggs, M. A. Osborne, E. A. Laird, and N. Ares, "Efficiently measuring a quantum device using machine learning", npj Quantum Information **5**, 10.1038/s41534-019-0193-4 (2019).

[19] J. D. Teske, S. S. Humpohl, R. Otten, P. Bethke, P. Cerfontaine, J. Dedden, A. Ludwig, A. D. Wieck, and H. Bluhm, "A machine learning approach for automated fine-tuning of semiconductor spin qubits", Applied Physics Letters **114**, 133102 (2019).

[20] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information* (University Press, Cambridge, 2010).

[21] T. Ihn, *Semiconductor nanostructures: quantum states and electronic transport* (Oxford University Press, 2015).

[22] D. Loss and D. P. DiVincenzo, "Quantum computation with quantum dots", Phys. Rev. A **57**, 120–126 (1998).

[23] D. Loss and D. P. DiVincenzo, "Quantum computation with quantum dots", Physical Review A **57**, 120–126 (1998).

[24] G. Burkard, T. D. Ladd, J. M. Nichol, A. Pan, and J. R. Petta, *Semiconductor spin qubits*, 2021.

[25] S. P. Harvey, "Quantum dots/spin qubits", Oxford Research Encyclopedia of Physics, 10.1093/acrefore/9780190871994.013.83 (2022).

[26] F. H. Koppens, C. Buizert, K. J. Tielrooij, I. T. Vink, K. C. Nowack, T. Meunier, L. P. Kouwenhoven, and L. M. Vandersypen, "Driven coherent oscillations of a single electron spin in a quantum dot", Nature **442**, 766–771 (2006).

[27] K. C. Nowack, F. H. Koppens, Y. V. Nazarov, and L. M. Vandersypen, "Coherent control of a single electron spin with electric fields", Science **318**, 1430–1433 (2007).

[28] M. Pioro-Ladrière, T. Obata, Y. Tokura, Y.-S. Shin, T. Kubo, K. Yoshida, T. Taniyama, and S. Tarucha, "Electrically driven single-electron spin resonance in a slanting zeeman field", Nature Physics **4**, 776–779 (2008).

[29]X. Zhang, H.-O. Li, G. Cao, M. Xiao, G.-C. Guo, and G.-P. Guo, "Semiconductor quantum computation", National Science Review **6**, 32–54 (2018).

[30]F. Fedele, "Spin interactions within a two-dimensional array of gaas double dots", PhD thesis (2020).

[31]J. R. Petta, A. C. Johnson, J. M. Taylor, E. A. Laird, A. Yacoby, M. D. Lukin, C. M. Marcus, M. P. Hanson, and A. C. Gossard, "Coherent manipulation of coupled electron spins in semiconductor quantum dots", Science **309**, 2180–2184 (2005).

[32]F. K. Malinowski, F. Martins, Ł. Cywiński, M. S. Rudner, P. D. Nissen, S. Fallahi, G. C. Gardner, M. J. Manfra, C. M. Marcus, F. Kuemmeth, and et al., "Spectrum of the nuclear environment for gaas spin qubits", Physical Review Letters **118**, 10.1103/physrevlett.118.177702 (2017).

[33]F. K. Malinowski, "Noise suppression and long-range exchange coupling for gallium arsenide spin qubits", PhD thesis (University of Copenhagen, 2017).

[34]B. J. van Wees, H. van Houten, C. W. Beenakker, J. G. Williamson, L. P. Kouwenhoven, D. van der Marel, and C. T. Foxon, "Quantized conductance of point contacts in a two-dimensional electron gas", Physical Review Letters **60**, 848–850 (1988).

[35]W. G. van der Wiel, S. De Franceschi, J. M. Elzerman, T. Fujisawa, S. Tarucha, and L. P. Kouwenhoven, "Electron transport through double quantum dots", Reviews of Modern Physics **75**, 1–22 (2002).

[36]L. P. Kouwenhoven, C. M. Marcus, P. L. McEuen, S. Tarucha, R. M. Westervelt, and N. S. Wingreen, "Electron transport in quantum dots", Mesoscopic Electron Transport, 105–214 (1997).

[37]R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha, and L. M. K. Vandersypen, "Spins in few-electron quantum dots", Rev. Mod. Phys. **79**, 1217–1265 (2007).

[38]S. J. Ling, J. Sanny, and W. Moebs, "8", in *University physics: volume 2* (OpenStax, Rice University, 2016).

[39]A. Chatterjee, P. Stevenson, S. De Franceschi, A. Morello, N. P. de Leon, and F. Kuemmeth, "Semiconductor qubits in practice", Nature Reviews Physics **3**, 157–177 (2021).

[40].

[41] T. Botzem, M. D. Shulman, S. Foletti, S. P. Harvey, O. E. Dial, P. Bethke, P. Cerfontaine, R. P. McNeil, D. Mahalu, V. Umansky, and et al., "Tuning methods for semiconductor spin qubits", Physical Review Applied **10**, 10.1103/physrevapplied.10.054026 (2018).

[42] H. Kiyama, A. Korsch, N. Nagai, Y. Kanai, K. Matsumoto, K. Hirakawa, and A. Oiwa, "Single-electron charge sensing in self-assembled quantum dots", Scientific Reports **8**, 10.1038/s41598-018-31268-x (2018).

[43] F. Hader, J. Vogelbruch, S. Humpohl, T. Hangleiter, C. Eguzo, S. Heinen, S. Meyer, and S. van Waasen, "On noise-sensitive automatic tuning of gate-defined sensor dots", IEEE Transactions on Quantum Engineering, 1–19 (2023).

[44] F. Vigneau, F. Fedele, A. Chatterjee, D. Reilly, F. Kuemmeth, F. Gonzalez-Zalba, E. Laird, and N. Ares, *Probing quantum devices with radio-frequency reflectometry*, Feb. 2022.

[45] $Scipy.stats.median_abs_deviation$.

[46] J. P. Zwolak, T. McJunkin, S. S. Kalantre, J. Dodson, E. MacQuarrie, D. Savage, M. Lagally, S. Coppersmith, M. A. Eriksson, J. M. Taylor, and et al., "Autotuning of double-dot devices in situ with machine learning", Physical Review Applied **13**, 10.1103/physrevapplied.13.034075 (2020).

[47] J. A. Nelder and R. Mead, "A simplex method for function minimization", The Computer Journal **7**, 308–313 (1965).

Part VI

APPENDICES

8.3 QCODES FUNCTIONS

- `doNd`: The doNd functions is used for simple 0d, 1d, and 2d measurements. The following functions are shortened in order to only include the relevant parameters for a simple measurement. For the full definition of the functions, please refer to `https://github.com/QCoDeS/Qcodes/tree/master/qcodes/dataset/dond`

```
1
2  def do1d(
3      param_set: ParameterBase,
4      start: float,
5      stop: float,
6      num_points: int,
7      delay: float,
8      *param_meas: ParamMeasT
9
10     """
11     Perform a 1D scan of ``param_set`` from ``start`` to ``stop``
       in
12     ``num_points`` measuring param_meas at each step. In case
       param_meas is
13     an ArrayParameter this is effectively a 2d scan.
14     Args:
15         param_set: The QCoDeS parameter to sweep over
16         start: Starting point of sweep
17         stop: End point of sweep
18         num_points: Number of points in sweep
19         delay: Delay after setting parameter before measurement is
        performed
20         param_meas: Parameter(s) to measure at each step or
       functions that
21               will be called at each step. The function should take no
        arguments.
22               The parameters and functions are called in the order
       they are
23               supplied.
24     """
```

Listing 8.1: Shortened do1d function including the relevant parameters

Example use:

```
# Running masurement with do1d
do1d(qdac.ch1, 0, -1, 300, 0.001, curr)
```

```
    def do2d(
    param_set1: ParameterBase,
    start1: float,
    stop1: float,
    num_points1: int,
    delay1: float,
    param_set2: ParameterBase,
    start2: float,
    stop2: float,
    num_points2: int,
    delay2: float,
    *param_meas: ParamMeasT,
    """
    Perform a 1D scan of ``param_set1`` from ``start1`` to ``stop1
    `` in
    ``num_points1`` and ``param_set2`` from ``start2`` to ``stop2
    `` in
    ``num_points2`` measuring param_meas at each step.
    Args:
        param_set1: The QCoDeS parameter to sweep over in the
    outer loop
        start1: Starting point of sweep in outer loop
        stop1: End point of sweep in the outer loop
        num_points1: Number of points to measure in the outer loop
        delay1: Delay after setting parameter in the outer loop
        param_set2: The QCoDeS parameter to sweep over in the
    inner loop
        start2: Starting point of sweep in inner loop
        stop2: End point of sweep in the inner loop
        num_points2: Number of points to measure in the inner loop
        delay2: Delay after setting parameter before measurement
    is performed
        param_meas: Parameter(s) to measure at each step or
    functions that
```

```
29          will be called at each step.
```

Listing 8.2: Shortened do2d function including the relevant parameters

Example use

```
1 # Running measurement with do2d
2 do2d(qdac.ch1, 0, -1, 100, 0.001, qdac.ch2, 0, -1, 100, 0.001,
    curr)
```