DISS. ETH NO. 23051

Transport Quantum Logic Gates for Trapped Ions

A thesis submitted to attain the degree of

DOCTOR OF SCIENCE of ETH ZÜRICH

(Dr. sc. ETH ZÜRICH)

presented by

LUDWIG ERASMUS DE CLERCQ

MSc. Phys., Univerity of Stellenbosch

born on 15.01.1986

citizen of
Republic of South Africa

Accepted on the recommendation of

Prof. Dr. J. P. Home
Prof. Dr. T. Schätz

2015

Abstract

One of the most promising methods for scaling up quantum information processing with trapped ions is the quantum CCD architecture [Wineland 98, Kielpinski 02], in which ions are shuttled between many zones of a multiplexed ion trap processor. A primary element of this design is the parallel operation of gates in multiple regions, presenting a formidable challenge for scaling of optical control. In this thesis, I demonstrate a new route which could dramatically reduce these requirements, by transporting ions through laser beams [D. Leibfried 07]. The thesis covers the hardware, software and experiments which were required in order to achieve this goal.

In order to perform these experiments I have developed several hardware and software solutions which are more widely applicable. Firstly, the Electronically Variable Interactive Lockbox (EVIL) which is widely used in our laboratory as a PI-controller. Secondly, the Direct Ethernet Adjustable Transport Hardware (DEATH) developed specifically for the transport experiments. Finally, I devised a simple method to create complex transport experiments which require control over multiple independent potential wells.

The main achievement of this thesis is the demonstration of parallel quantum logic gates involving transport of ions. I have also demonstrated that these gates can be performed sequentially in order to create more complex operations by performing a Ramsey experiment. Additionally I have developed a novel method to estimate time-dependent coefficients of a Hamiltonian containing two non-commuting terms. The problem naturally arises when transporting an ion through a laser beam. The method is sensitive enough that we can detect the curvature of the wavefronts of the Gaussian laser beam.

Zusammenfassung

Eine der vielversprechendsten Methoden, die Quanteninformationsverarbeitung mit gefangenen Ionen zu skalieren, ist die quantenladungsgekopplete Architektur („quantum charge-coupled device") [Wineland 98, Kielpinski 02], in welcher Ionen zwischen verschiedenen Zonen eines gemultiplexten Ionenfallen-Prozessors hin und her bewegt werden. Ein wichtiges Element dieses Designs ist die gleichzeitige Anwendung von Gatteroperationen in verschiedenen Regionen des Chips, welches erhebliche Schwierigkeiten bei der Skalierung des optischen Aufbaus zur Steuerung der Ionen bereitet. In der vorliegenden Arbeit beschreibe ich einen neuen Weg, welcher diese Beschränkungen stark vereinfacht, in dem Ionen durch statische Laserstrahlen transportiert werden [D. Leibfried 07]. Diese Dissertation beschreibt die Hard- und Software, sowie Experimente, welche benötigt werden, um dieses Ziel zu erreichen.

Um diese Experimente durchführen zu können, habe ich im Verlauf dieser Arbeit verschiedene Hard- und Softwarelösungen entwickelt, welche auch anderweitig anwendbar sind. Nennenswert sind hier die „elektronisch verstellbare, interaktive Lockbox"(EVIL), welche in unserem Labor als PI-Regler genutzt wird, sowie die „direkt über Ethernet anpassbare Transport-Hardware"(DEATH), welche speziell für die Transport-Experimente entwickelt wurde. Schliesslich habe ich eine einfache Methode entwickelt, um komplexe Transport-Experimente zu realisieren, die Kontrolle über mehrere, voneinander unabhängige Potentialtöpfe benötigen.

Die hauptsächliche Leistung dieser Arbeit ist die Demonstration paralleler quantenlogischer Gatter, welche den Transport von Ionen beinhalten. In dem ich diese Methode genutzt habe, um ein Ramsey-Experiment durchzuführen, konnte ich ausserdem zeigen, dass Gatter sequenziell ausgeführt werden können, um komplexere Operationen zu realisieren. Schliesslich habe ich eine neue Technik entwickelt, die es ermöglicht, die zeitabhängigen Koeffizienten eines Hamiltonians, welcher zwei nicht-kommutierenden Terme beinhaltet, abzuschätzen. Eine solche Problemstellung taucht auf, wenn ein Ion durch einen Laserstrahl transportiert wird. Die Methode ist so empfindlich, dass die Krümmung der Wellenfront eines Gauss-Strahls nahe des Fokuses gemessen werden kann.

Acknowledgments

It has been very fun and educational working with everyone in the TIQI group. You all have been an inspiration to me and have driven me to perform cool experiments. Listening to progress of everyone on a weekly basis was so enlightening that I know everyone is always motivated to do their best to improve things in the laboratory.

I wish to thank Prof. Dr. Jonathan Home, who has given me the opportunity to carry out the research I have performed in his group. He has supervised this work and provided meaningful suggestions and discussions on my topic of research. He always pushed me and motivated me to improve on data I have taken to achieve the best possible results for my experiments. The words "good progress" I will always associate with his name.

Ben Keitch for always being willing to help and provide technical know-how, especially when I started in the group and needed it most.

Vlad Negnevitsky who has truly helped me the most in becoming an engineer and together building the EVIL systems everyone is using in the laboratory. Apart from the engineering we had many fruitful discussions on physics. His willingness to help everyone who asks stands out in our group and he has made a difference in most projects.

Hsiang-Yu Lo for getting me up and running in the laboratory and helping me with experiments whenever I needed it. Discussing basic questions we both had in ion trapping/physics in general and using our hands on occasion as vectors on an imagined Bloch-sphere has certainly been fun. You did an amazing job with the beryllium system, sorry that the parallel gates experiment have been running both times the lasers have broken...

Daniel Kienzler, his designed and manufactured 3D Paul trap which all of the experimentalists in B20 are using on a daily basis.

Matteo Marinelli for taking most of the data with me for the experiments contained in this thesis and walking home with me when we have missed the last bus. You have made great additions to *Ionizer* and without those my fish, fins and flare would never have existed. I have to admit everything was awesome.

David Nadlinger, his work on the EVILs, DEATHs and control system which in some cases were in dire need of change. His work has made it much simpler to obtain good data for my experiments. We have worked a lot together throughout all your projects, or rather, it was great having to see you just excellently completing task without too much input. It has been fun working with you.

Robin Oswald for making excellent upgrades to the way I have solved waveforms and for speeding up the extraction process we both have worked on for quite some time. You have a unique way of approaching problems which is refreshing and I believe you are an equally strong physicist and engineer.

Christa Flühmann for her work on the 729 nm and the magnetic field stabilization which all of the users in B20 can only appreciate the amount of effort she has put in getting it running so well, thank you. I hope you do not play music through those sounds cards while performing experiments by accident.

I also want to thank the experimentalists from our other laboratories in B25, Florian Leupold, Frieder Lindenfelser and Joseba Alonso for always being willing to listen and being able to lift up my mood. You guys have been great.

Lastly I would like to thank my parents Wessel and Hilda de Clercq for always encouraging and supporting me trough all my studies and personal life. They have given and done more for me than any other and I am proud to be their son.

Contents

Chapter 1

Introduction

Quantum mechanics is well established and accepted, but still one of the least intuitive ways of looking at nature for our classically calibrated human senses. From understanding black-body radiation to using the quantum properties of nature in striving to build a quantum computer, it is clear that technology and our understanding of nature has grown in the last odd 100 years.

Yuri Manin [Manin 80] and Richard Feynman [Feynman 82, Feynman 86] independently developed the notion that simulating physics on a quantum computer would be advantageous over using the digital counterpart. Other than simulating physics the question arose whether these computers could outperform classical ones on other tasks too. From the work done on laying out a framework for quantum computation by Paul Benioff, David Deutsch later described the first universal quantum computer [Deutsch 85]. A few years after that people like P. Shor and L. Grover have developed the first algorithms to run on such a quantum device in which it will run more efficiently than on a classical computer [Shor 94, Grover 97]. Within this thesis emphasis is on ion trapping for building a scalable quantum computer. There are several other architectures being investigated in order to build a quantum computer, examples being: neutral atoms, NV-centres, circuit QED, cavity QED and others [Roa 04, Roa].

Initial experiments on Doppler cooling of magnesium ions in a Penning trap [**?**] and simultaneous experiments on Doppler and sideband cooling of barium ions in a Paul trap [Neuhauser 78] were achieved in the late 1970s. These experiments were performed with many ions and not long after the group of Hans Dehmelt demonstrated the first singly trapped and cooled barium ion [Neuhauser 80]. It was only in the late 1980s that David Wineland group successfully cooled a single mercury ion to the ground state [Diedrich 89]. From the studies and proposal by Barenco *et al.* [Barenco 95] in 1995 it has been shown theoretically we only require to implement a controlled not gate (CNOT) and single qubit rotations to be able to perform any operation

required by a quantum computer. Within the same year Cirac and Zoller proposed how trapped cold ions could be used to perform quantum computation [Cirac 95]. It was not long after that the first coherent manipulations in order to implement a CNOT (controlled-NOT) gate on a single beryllium ion were demonstrated [Monroe 95].

The demonstration of an universal set of gates is an important step. Nevertheless the requirements for scaling up a quantum information processor (QIP) are stringent. Requirements on the physical implementation of the gates themselves are quite stringent and it is commonly accepted that an error of the probability of 10^{-4} on a single qubit operation would be acceptable for tolerant quantum computing using error correcting schemes [Preskill 98, Knill 05, Knill 10]. An acceptable range would be between 10^{-6} and 10^{-3} [Steane 03]. Experimentally the best achieved single qubit operation has been performed by the Oxford group on a calcium ion in a surface ion trap in which they achieved average single qubit gate fidelities of 99.9999% [Harty 14]. NIST has also achieved fault tolerant fidelities (error below 10^{-4}) using a beryllium ion in a surface trap [Brown 11]. The best achieved fidelities for two qubit gates have been achieved by the group at Oxford where they have obtained a fidelity of 99.9(1)% in which they have improved the previous best of 99.3(1)% which was reached at the group at Innsbruck [Benhelm 08].

Ion-trapping has made great advancements toward quantum computing by achieving high fidelities on quantum operations. The biggest obstacle which is faced not only in ion trapping quantum computing, but for quantum computing in general is how to scale up our systems to allow for multiple coherent operations on qubits and finally working with 100s to 1000s of qubits in order to build quantum computers. David Wineland and his group at NIST has set out a proposal for scaling up even before the achievements of the aforementioned high fidelities have been achieved, foreseeing the need for scalability in ion trapping [Wineland 98, Kielpinski 02]. The idea is relatively simple and elegant. If we can build a segmented ion trap to act as a sub-unit of a quantum processor reliably we would only need to duplicate the sub-unit's structure in order to scale up and build a quantum computer. In their proposal the carriers of information are the ions and they are required to be physically moved around after coherent operations have been performed on them. Thus laser pulses are applied to various static ions trapped in the quantum charge-coupled device (QCCD) in order to implement gate operations. After the gate operations the ions are transported to a designated zones of the trap which would lead to temporary storage or for further computation. The process can be pictured as shown in Fig. 1.1 for the two graphs on the left.

Within the QCCD architecture various experiments has been performed. Initially transport of single ions was investigated in a linear Paul trap as well as the splitting of beryllium ion pairs [Rowe 02]. They transported an ion over a

distance of 1.2 mm in 50 µs (velocity of 24 m/s) with negligible motional heating. The group at NIST then further increased the complexity of the ion trap to include a X-junction through which they successfully transported an ion with a low heating rate [Blakestad 09, Blakestad 11, Blakestad 10]. Later the ion trapping group at Georgia Tech has demonstrated similar results[Wright 13].

After transport and ion separation has become reliable more complex experiments using combinations of transport, splitting and combining of ions together with well controlled pulsed laser beam sequences were then performed in the QCCD architecture. Chiaverini *et al.* [Chiaverini 04] performed a quantum error correcting algorithm using three beryllium ions, one logical qubit was formed which was protected from spin-flip errors. One splitting and three transport operations were required as part of the encoding scheme in order to address the primary qubit ion and the ancilla ions. In [Barrett 04] conditional laser pulse sequences and several splitting and combining operations of three beryllium ions to demonstrate quantum state teleportation. In [Jost 09] the entanglement of two mechanical oscillators have been demonstrated. In 2009 NIST has implemented within the QCCD architecture all required operations to build a quantum computer in which transport and splitting was again necessary [Home 09, Hanneke 09]. It has become common in the QCCD architecture to perform complex routines which require transport, splitting of ions and the control of laser beams for qubit manipulation. More recently [Gaebler 12] has performed randomized benchmarking in a segmented ion trap using two beryllium ions and two magnesium ions to sympathetically cool the beryllium ions during transport. The aforementioned complex experiments all had one thing is common, the transport, splitting and combination of ions were done slowly enough to avoid motional excitation. More recently diabatic transport of ions was successfully demonstrated as well as how to selectively split up a string of up to 9 beryllium ions [Bowler 12, Bow 15]. The diabatic transport was carried out with a single beryllium ion over a distance of 370 µm within 8 µs (velocity of 46.25 m/s). The fastest splitting of two beryllium ions within the experiment [Bowler 12, Bow 15] was carried out in 55 µs. Within the same period the group at Mainz achieved similar results for diabatic transport of calcium over 280 µm in 3.6 µs (velocity of 77.77 m/s) [Walther 12]. The group at Mainz additionally achieved similar results to that of NIST [Bowler 12, Bow 15] in spitting two calcium ions [Ruster 14] based on theoretical considerations that can be found in [Kaufmann 14]. The method in which they have achieved it was building on one of the very first suggested methods for splitting two ions proposed by Jonathan Home and Andrew Steane [Home 06].

D. Leibfried from NIST has proposed combing the gate operations and transport by implementing gates in which the ions move through laser beams, referred to as transport gates [D. Leibfried 07]. The proposal is depicted in Fig. 1.1 as the combination of separate manipulation and transportation into one

operation where qubit manipulation and transportation occurs simultaneously. The requirements to perform transport gates are for the beams to be on while an ion(s) is(are) transported through a qubit addressing laser beam at constant velocity in order to implement qubit manipulations [D. Leibfried 07]. Implementing gates this way would lead to a significant reduction in the number of optical elements required in order to obtain gate operations. Experimentally we make use of acoustic optic modulars (AOM) to produce laser pulses and the setups in which they appear are typically bulky. Having multiple regions within an ion trap quantum computer in which laser pulses would be required for each qubit manipulation an additional AOM setup would be needed. Thus even though the ion-trap is on the millimeter scale, the optical setups are not and this does not scale well.

The topic of this thesis is the investigation of transport gates. During my research I have come across complications in performing transport gates which were not considered in the original proposal . This was due to our using a varying velocity. This has led me to develop methods to investigate how the velocity of the ion experimentally changes as it travels through the laser beam. It turned out that the velocity of the ion during transport is not constant throughout the process, but repeatable. In particular we wanted a diagnostic and the study of time-dependent Hamiltonians were required to find the form of the velocity during transport.

Theoretical studies such as those carried out by E. Barnes [Barnes 12, Barnes 13] investigated how analytical solutions can be found for solving the two level system when there are more than one Pauli operator acting on it. Quantum system identification has been proposed to study time-independent Hamiltonians experimentally from time traces [Zhang 14]. In the field of femtochemistry "Hamiltonian encoding observation decoding" (HE-OD) has been proposed to study Hamiltonians in an experimental setup in order to obtain constant branching ratios between different excitation pathways in molecules [Mitra 03, de Castro 10, de Castro 13].

1.1 Thesis layout

The experiments I performed all relate to these gates and how to analyze them. I experimented initially with a single calcium ion which was transported through a laser beam. Complications arose, but from the complication interesting physics arose as well. Having created new methods to study the transport gate process in time I have developed a deeper understanding of the physics involved. Simply having the choice of working with beryllium I later on realized it is the simpler ion species to work with regarding the implementation of transport gates. I then created an experiment in which I performed two simultaneous transport gates using a reflected beam and two beryllium

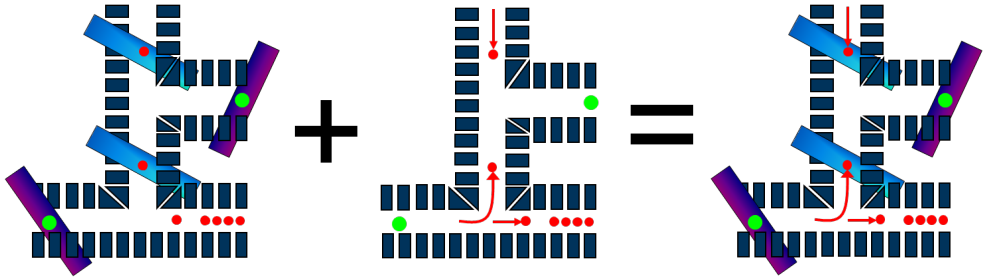


Figure 1.1: The left two proposal are the original from [Wineland 98, Kielpinski 02]. The final proposal that combines transport and qubit manipulations are from D. Leibfried [D. Leibfried 07]

ions. Within the experiment the gate implemented on each ion is tunable by varying the velocity at which it travels through the beam.

In chapter 2 I formulate how ions are trapped in a 3D Paul trap. The Jaynes Cumming Hamiltonian is introduced with a discussion of heating rate measurements by performing blue-sideband flopping.

In chapter 3 the general experimental setup as was used for my experiments with the various laser configurations are described as well as the level structure of calcium and beryllium.

In chapter 4 I describe the development and use of a multi-purpose digital PI controller. Some of the projects that has been carried out on it are briefly discussed and potential future upgrades and projects are mentioned. This is a heavily used device in our laboratory and I can not cover everything in detail, many details can be found in [Hungenberg 13, Nadlinger 13, Fischer 15a].

In chapter 5 I introduce the voltage sources we use for our shim electrodes. More detail is given on the design and creation of the fast voltage sources referred to as DEATH (Direct Ethernet Adjustable Transport Hardware) used in the transport experiments. The design and operation of the DEATHs on hardware as well as on software level are discussed herein.

Chapter 6 discuses adiabatic and diabatic transport of ions. I describe the methods I have used in order to obtain time varying voltages which are applied to the electrodes of the ion trap in order to transport ions around, referred to as waveforms. The framework I have developed on how to solve for multi-ion experimental procedures in our 3D Paul trap is described.

The study of time-dependent Hamiltonians and the extraction of their time-dependent operators from experimental data is formulated and applied to the case of ion transport at a velocity that is varying over time in chapter 7. How the wavefronts of a Gaussian beam influences the observations of the extracted time-dependent detuning and obtaining the velocity of the ion as a function

of time is discussed as well. This chapter is based on a recent publication [de Clercq 15b].

In chapter 8 I present the data obtained from the simultaneous implementation of two transport quantum logic gates using beryllium ions. This chapter is based on a recent publication [de Clercq 15a].

My conclusions and various improvements and possible next experiments are described finally in chapter 9.

Chapter 2

The physics of ion trapping

Within the experiments we have performed we have used calcium and beryllium ions. Within this chapter the theory behind the ideal linear Paul trap which can be used to confine single ions in space is presented. Once ions are confined we would like to perform quantum operations with them and in order to do this we manipulate their quantum behaviour by shining laser light on them. I will discuss the physics required to describe coherent operations on ions using laser light which is the main topic of concern in later chapters.

2.1 Linear Paul trap

In order to trap ions spatially the requirement is to obtain a confining potential in all spatial directions. The electrostatic potential $U(x, y, z)$ to achieve this must satisfy the Laplace equation

$$\nabla^2 U(x, y, z) = 0 \tag{2.1}$$

The equation is impossible to satisfy with purely an electrostatic potential as stated by Earnshaw's theorem. To see why this is consider Eq. 2.1 written as

$$\nabla \cdot \nabla U = \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0. \tag{2.2}$$

Thus in order to satisfy Eq. 2.1 there should be a 2nd order curvature along one of the directions, otherwise the equation cannot be fulfilled. We are therefore left with a saddle point in the electrostatic field where one direction will always be anti-confining. In order to overcome the anti-confinement we can use changing fields by applying a Radio Frequency (RF) potential along with the static potentials, such that Eq. 2.1 is satisfied on average. One such device with which this is done is a linear Paul trap.

A linear Paul trap is a configuration which consists out of four metal rods with end caps on either end as can be seen in Fig. 2.1. An oscillating potential

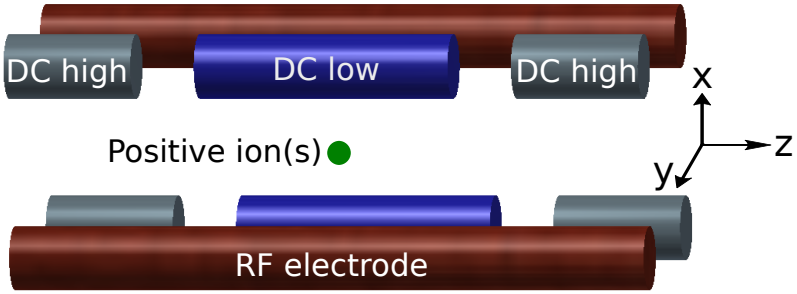


Figure 2.1: An ideal linear Paul with DC- and RF-voltages applied as shown. The z-axis is chosen in this thesis to present the axial confinement and will often be referred to as the transport axis.

$\phi_{\mathrm{rf}} = U_r + V_0 \cos(\Omega_{\mathrm{rf}} t)$ with frequency Ω_{rf}, amplitude V_0 and offset U_r is applied to two of the electrodes while grounding the other two with respect to the RF ground. Additionally a static potential Φ_{st} is applied to the end caps (DC high). In this thesis the direction of static confinement will be referred to as the z-direction (or transport axis) and the x- and y- direction as the radial direction. The total potential can be written as a combination of an oscillating and static potential [Wineland 98, Bow 15]

$$\begin{aligned} U(x,y,z) &= \Phi_{\mathrm{rf}} + \Phi_{\mathrm{st}} \\ U(x,y,z) &= \frac{U_r + V_0 \cos(\Omega_{\mathrm{rf}} t)}{2R^2}(\alpha_r x^2 + \beta_r y^2 + \gamma_r z^2) \\ &\quad + \kappa U_0(\alpha_s x^2 + \beta_s y^2 + \gamma_s z^2) \end{aligned} \tag{2.3}$$

where the coefficients has to satisfy the condition (Eq. 2.1)

$$\alpha_r + \beta_r + \gamma_r = 0 \tag{2.4}$$

$$\alpha_s + \beta_s + \gamma_s = 0 \tag{2.5}$$

In general the coefficients are geometry dependent and require numerical simulation to obtain precise values. For the ideal Paul trap $\gamma_r = 0$ and with the remaining condition given by Eq. 2.2 gives $\alpha_r = -\beta_r$. For the static potential the ideal Paul trap has $\alpha_s = \beta_s$ and for confinement along the z-axis

$$\Phi_{\mathrm{st}} = \kappa U_0 (z^2 - \frac{1}{2}(x^2 + y^2)). \tag{2.6}$$

Where we have introduced the geometrical constant κ which can have values from 0 to 1. Casting this equation in the form of a harmonic oscillator [Wineland 98]

$$\Phi_{\mathrm{st}} = \frac{1}{2q} m \omega_z^2 (z^2 - \frac{1}{2}(x^2 + y^2)). \tag{2.7}$$

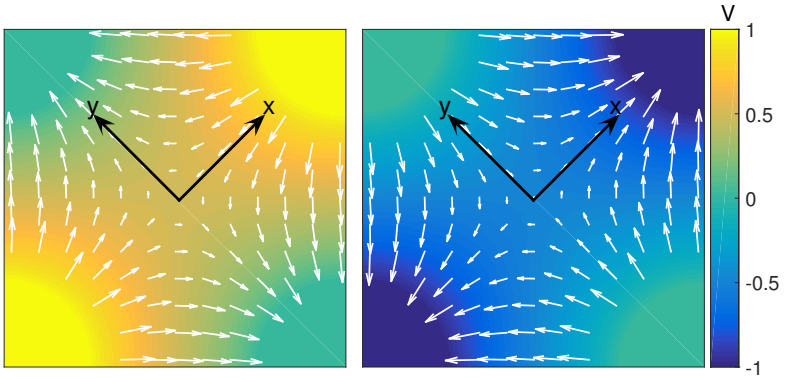


Figure 2.2: Illustration of the switching of an ideal linear Paul trap in the radial direction. In the first stage when positive RF-voltages are applies the trap is confining along the x-direction and anti-confining in y. When the RF signal changes phase and is negative the confinement is changed to be along the y-direction.

where m is the mass of an ion with charge q oscillating in the z-direction with a (motional/secular) frequency of $\omega_z = (2\kappa q U_0/m)^{1/2}$. From the form of Eq. 2.7 we can see that it is only confining in the z-direction and anti-confining in the x- and y-directions. The radial potential given by

$$\Phi_{\text{rf}} = \frac{U_r + V_0 \cos(\Omega_r t)}{2R^2}(x^2 - y^2) \tag{2.8}$$

changes from being confining in the x-direction and anti-confining along the y-direction to being confining in the y-direction and anti-confining along the x-direction depending on the phase of the RF signal. A simple illustration is shown in Fig. 2.2 which illustrates this behaviour. The electric field has been solved in two-dimensions and displayed by the white arrows for two different phases of the RF potential applied to the electrodes (top right, bottom left) while maintaining ground at the remaining electrodes. Equidistant spacings have been chosen between electrodes, distances and sizes are not shown for the illustrative purpose.

The equations of motion for an ion with mass m and charge q placed in the RF potential Φ_{rf} are

$$m\ddot{\vec{r}} = q\vec{E} = -q\nabla\Phi_{\text{rf}} \tag{2.9}$$

which are the Mathieu equations [Wineland 98]

$$\begin{aligned} \frac{dx^2}{d^2\zeta} + \left[a_x + 2q_x \cos\left(2\zeta\right)\right] x = 0 \\ \frac{dy^2}{d^2\zeta} + \left[a_y + 2q_y \cos\left(2\zeta\right)\right] y = 0 \end{aligned} \tag{2.10}$$

with $\zeta = \Omega_{\mathrm{rf}} t/2$ and

$$a_x = \frac{4q}{m\Omega^2}\left(\frac{U_r}{R^2} - \kappa U_0\right) \tag{2.11}$$

$$a_y = -\frac{4q}{m\Omega^2}\left(\frac{U_r}{R^2} + \kappa U_0\right) \tag{2.12}$$

$$q_x = -q_y = \frac{2qV_0}{\Omega^2 m R^2}. \tag{2.13}$$

The Floquet solutions are used to find solutions to the Mathieu equations in general. Under typical operating conditions we have $a \ll b^2 \ll 1$ in which the solutions to Eq. 2.10 are [Sasura 02]

$$x(t) \approx x_0 \left[1 + \frac{q_x}{2}\cos\left(\Omega t\right)\right] \cos\left(\omega_x t + \varphi_x\right) \tag{2.14}$$

$$y(t) \approx y_0 \left[1 - \frac{q_x}{2}\cos\left(\Omega t\right)\right] \cos\left(\omega_y t + \varphi_y\right) \tag{2.15}$$

with

$$\omega_x = \frac{\Omega}{2}\sqrt{\frac{q_x^2}{2} + a_x} \tag{2.16}$$

$$\omega_y = \frac{\Omega}{2}\sqrt{\frac{q_x^2}{2} + a_y}. \tag{2.17}$$

What we notice here is that only when $U_r \neq 0$ do we get distinct secular frequencies ω_x and ω_y. Even though our trap geometry is different to the ideal Paul trap, one will approximately obtain similar results. In the trap the motion that is associated with the frequency Ω is called the micro-motion. Within an experimental setup there are typically stray fields that push the ion away from the ideal conditions and in our trap we minimize micro-motion typically by using an extra set of electrodes, the shim electrodes. See section 3.1 for details the Paul trap design we use. Typically in operation of the trap there is no offset on the RF electrodes ($U_r = 0$) and we can write the pseudo-potential $q\phi_{\mathrm{rad}}$ as experienced by the ions in the radial direction as [Wineland 98, Sasura 02]

$$q\phi_{\mathrm{rad}} = \frac{m\omega_r^2}{2}\left(x^2 + y^2\right) \tag{2.18}$$

with the radial frequency given by

$$\omega_r = \frac{\Omega q_x}{2\sqrt{2}} = \frac{qV_0}{mR^2\omega\sqrt{2}}. \tag{2.19}$$

The pseudo-potential $q\phi_{3D}$ in all directions is then (also assuming the DC-electrodes are set at 0 V) [Sasura 02]

$$q\phi_{3D} = \frac{m\omega_z^2 z^2}{2} + \frac{m\omega_r^2}{2}\left(x^2 + y^2\right) \tag{2.20}$$

Due to the symmetric geometry of the trap and to simplify transport considerations, in this thesis I consider only the potential formed along the z-direction and how to manipulate it to achieve transport. For a single harmonic oscillator along the z-axis we have

$$\Phi_{st}(z,t) = \frac{1}{2q} m\omega_z(t)^2 (z - s(t))^2 + C(t) \tag{2.21}$$

where $s(t)$ represents the lowest potential energy point of the harmonic well which in the case of transport is time-dependent. The oscillator frequency $\omega(t)$ as well as a constant offset term $C(t)$ could also be time-dependent.

2.2 Hamiltonian

In this thesis I will mainly consider the transport of an ion along the transport axis of a 3D Paul trap which will be referred to as the z-axis in this thesis. The experimental depiction of a laser beam and ion within the laser beam is shown in Fig. 2.3. As can be seen in the figure, the z-axis makes an angle θ_{exp} with the laser beam and is displaced $\vec{\xi}$ along the beam axis. In this section I will consider the Hamiltonian for the system and assume that the laser beam can be described by plane waves. Deviation from the plane wave description and how to correct for the curvature of wavefronts is discussed in chapter 7.

The motion of an ion trapped in a harmonic well along the z-axis is described by the Hamiltonian of a harmonic oscillator

$$\hat{H}^{(m)} = \hbar\omega_z \left(\hat{a}_z^\dagger \hat{a}_z + \frac{1}{2}\right). \tag{2.22}$$

Here ω_z is the vibrational frequency along the z-axis and $\hat{a}_z^\dagger$ the creation and $\hat{a}_z$ the annihilation operators of motion in this mode. The eigenvectors of $\hat{H}^{(m)}$ are the Fock states which I will represent with $|n\rangle$ with n the quantum number associated with the energy $\hbar\omega_z\,(n + 1/2)$. The ion's position operator for this mode is $\hat{z} = z_0\left(\hat{a}_z^\dagger + \hat{a}_z\right)$ with $z_0 = \sqrt{\hbar/2m_{ion}\omega_z}$ the standard deviation of the ground state wave function and m_{ion} is the ion mass. The internal electronic

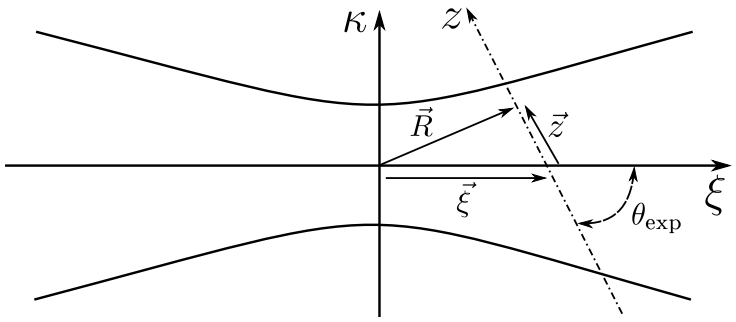


Figure 2.3: Diagram of experimental setup. The ion is transported along the transport z-axis and the light comes in at an angle $\theta_{\text{exp}} = 45^0$ with respect to the z-axis and points in the ξ-direction.

states with the ground state $|g\rangle$ and excited state $|e\rangle$ of the ion, which are separated by an energy $\hbar\omega_0$, are used to form the qubit is described by the Hamiltonian with $\hat{\sigma}_z = |e\rangle\langle e| - |g\rangle\langle g|$

$$\hat{H}^{(e)} = \frac{\hbar\omega_0}{2}\hat{\sigma}_z. \tag{2.23}$$

If we describe laser light of frequency ω_L and phase φ as planes waves traveling in the direction of the wave vector $\vec{k}$ the electric field with polarization $\vec{\epsilon}$ at any position $\vec{R}$ is given by [Sasura 02]

$$\vec{E}(\vec{R},t) = \vec{\epsilon}E(\vec{R},t) = \vec{\epsilon}\cos\left(\omega_L t - \vec{k}\cdot\vec{R} + \varphi\right). \tag{2.24}$$

The Hamiltonian describing the interaction between the electric field and the ion to second order is given by [Sasura 02]

$$\begin{aligned}\hat{H}^{(i)} &= \hat{V}^{\text{DP}} + \hat{V}^{\text{QP}} \\ \hat{H}^{(i)} &= e\hat{\vec{r}}_{\text{ion}}\cdot\vec{\epsilon}E(\hat{\vec{R}},t) + \frac{e}{2}\left(\hat{\vec{r}}_{\text{ion}}\cdot\vec{\epsilon}\right)\left(\hat{\vec{r}}_{\text{ion}}\cdot\nabla E(\hat{\vec{R}},t)\right)\end{aligned} \tag{2.25}$$

where $\hat{\vec{r}}_{\text{ion}} = \{\hat{x},\hat{y},\hat{z}\}$ is the internal operator relating the interacting electron's position relative to the minimum of the confining potential. $\hat{\vec{R}}$ is the external position operator corresponding to the location of the ion within the laser beam. Here $\hat{\vec{R}} = \vec{\xi} + \hat{\vec{z}}$ with $\vec{\xi}$ the position vector indicating where along the beam axis the ion cuts across as shown in Fig. 2.3 and $\hat{\vec{z}} = \vec{z} + \{0,0,\hat{z}\}$ the position operator corresponding to the location of the ion along the z-axis and $\hat{z}$ the centre of mass mode in the z-direction. We can define dipole (DP) and quadrupole (QP) coupling constants as

$$\Omega^{\text{DP}} = \frac{eE_0}{\hbar}|\langle e|\hat{\vec{r}}_{\text{ion}}\cdot\vec{\epsilon}|g\rangle \tag{2.26}$$

$$\Omega^{\text{QP}} = \frac{eE_0|\vec{k}|}{2\hbar}|\langle e|\left(\hat{\vec{r}}_{\text{ion}}\cdot\vec{\kappa}\right)\left(\hat{\vec{r}}_{\text{ion}}\cdot\vec{\epsilon}\right)|g\rangle|. \tag{2.27}$$

If the electronic states are dipole allowed then Ω^{DP} will dominate and if not Ω^{QP} will. Generalizing the interaction Hamiltonian in the case of either dipole or quadrupole coupling dominating results in

$$\hat{H}^{(i)} = \hbar\Omega_R \cos\left(\omega_L t - \vec{k}\cdot\hat{\vec{R}} + \varphi\right)(\hat{\sigma}_+ + \hat{\sigma}_-) \tag{2.28}$$

assuming plane waves interact with the ion. The operators $\hat{\sigma}_+ = |e\rangle\langle g|$ and $\hat{\sigma}_- = |g\rangle\langle e|$ are the raising and lowering operators acting on the internal states of the ion and Ω_R is the on resonant Rabi frequency giving the coupling strength of the interaction be it dipole or quadrupole transition. Evaluating the dot product with the external position operator and writing the resulting interaction Hamiltonian using complex exponentials gives

$$\hat{H}^{(i)} = \frac{\hbar\Omega_R}{2} e^{i\left(\omega_L t - \vec{k}\cdot\vec{z} - \eta\left(\hat{a}_z^\dagger + \hat{a}_z\right) + \varphi\right)}\hat{\sigma}_+ + \text{h.c.} \tag{2.29}$$

since the term $\vec{k}\cdot\vec{\xi}$ is constant the phase was redefined as $\varphi = \varphi - \vec{k}\cdot\vec{\xi}$ and additionally the Lamb-Dicke parameter $\eta = |\vec{k}|z_0$ was introduced. The product term $\vec{k}\cdot\vec{z}$ is not absorbed by the phase φ, since in transport the position varies with time $\vec{z}(t)$ and the product is not constant. This is a core result to transport gates which should be noted that it differs to the static case where the term can absorbed by the phase. The total Hamiltonian for the system is given by

$$\hat{H} = \hat{H}^{(m)} + \hat{H}^{(e)} + \hat{H}^{(i)}. \tag{2.30}$$

Using the unitary operator $U = e^{-\frac{i}{\hbar}\left(\hat{H}^{(m)} + \hat{H}^{(e)}\right)t}$ to transform the Hamiltonian to the interaction picture of the Hamiltonian using the well known transformation relation (see Appendix B)

$$\hat{H}_I = U^\dagger H U - i\hbar U^\dagger \frac{d}{dt} U \tag{2.31}$$

and furthermore neglecting fast oscillating terms $e^{\pm i(\omega_L + \omega_0)t}$ by using the rotating wave approximation (RWA) results in the interaction Hamiltonian taking the following form [Leibfried 03]

$$\hat{H}_I = \frac{\hbar\Omega_R}{2} e^{i\eta\left(\hat{a}e^{-i\omega_z t} + \hat{a}^\dagger e^{i\omega_z t}\right)} e^{-i\left(\delta_L t - \vec{k}\cdot\vec{z} + \varphi\right)}\hat{\sigma}_+ + \text{h.c.} \tag{2.32}$$

When the ion changes its internal and motional states from $|g, n\rangle$ and $|e, n'\rangle$ the Rabi frequency for the transition is given by

$$\Omega_{n',n} = \Omega_R |\langle n'| e^{i\eta(\hat{a} + \hat{a}^\dagger)} |n\rangle| \tag{2.33}$$

$$\Omega_{n',n} = \Omega_R e^{-\eta^2/2} \sqrt{\frac{n_<!}{n_>!}} \eta^{|n'-n|} L_{n_<}^{(|n'-n|)}(\eta^2) \tag{2.34}$$

where $n_{>(<)}$ being the greater(lower) of the two values n and n' and the generalized polynomial $L_n^{(\alpha)}$. The experiments reported in this thesis were performed in the Lamb-Dicke regime which requires that the inequality $\eta\sqrt{(\hat{a}+\hat{a}^\dagger)} \ll 1$ hold for all time during the interaction [Leibfried 03]. The Lamb-Dicke parameters in our experiments are $\eta_{\mathrm{Ca}} \approx 0.05$ for calcium and $\eta_{\mathrm{Be}} \approx 0.24$ for beryllium. Within the Lamb-Dicke regime the interaction Hamiltonian in Eq. 2.32 is approximated as

$$\hat{H}_I = \frac{\hbar\Omega_R}{2}\left(1 + i\eta\left(\hat{a}e^{-i\omega_z t} + \hat{a}^\dagger e^{i\omega_z t}\right)\right)e^{-i\left(\delta_L t - \vec{k}\cdot\vec{z}+\varphi\right)}\hat{\sigma}_+ + \text{h.c.}. \tag{2.35}$$

Next I will consider the conditions for the laser beam under which we have carried out the experiments and an analysis of the temperature of the ion.

2.2.1 This work and analysis of motion

In the work that will be presented on transport gates later in the thesis (chapters 7 and 8) we are mainly interested in the carrier transition. For the carrier transition the detuning of the laser is such that ($\delta_L = \Delta \ll \omega_z$, $n' = n$) of which the interaction Hamiltonian in Eq. 2.35 reduces to

$$\hat{H}_I^{(\mathrm{car})} = \frac{\hbar\Omega_R}{2}e^{-i\left(\Delta t - \vec{k}\cdot\vec{z}+\varphi\right)}\hat{\sigma}_+ + \text{h.c.} \tag{2.36}$$

The interaction Hamiltonian in the Schrödinger picture is

$$\hat{H}_S^{(\mathrm{car})} = \frac{\hbar\Omega_R}{2}e^{i\left(\omega_L t - \vec{k}\cdot\vec{z}+\varphi\right)}\hat{\sigma}_+ + \text{h.c.} \tag{2.37}$$

Tuning the laser such that $\delta \approx \Delta + \omega_z$ results in resonant driving of a motional adding sideband, often referred to as the blue sideband (bsb)

$$\hat{H}^{(\mathrm{bsb})} = \hbar\eta\Omega_R\left(-ie^{-i\left(\Delta t - \vec{k}\cdot\vec{z}+\varphi\right)}\hat{a}\hat{\sigma}_- + ie^{+i\left(\Delta t - \vec{k}\cdot\vec{z}+\varphi\right)}\hat{a}^\dagger\hat{\sigma}_+\right). \tag{2.38}$$

Driving the blue sideband shows sensitivity on the Fock state number distribution and we can therefore use it as a diagnosis tool for the Fock states. Preparing the ion in the ground state we can find the probability of finding the ion in the excited state by driving the blue sideband transition for a time t is given by [Leibfried 03]

$$P^{(\mathrm{bsb})}(|e\rangle) = \frac{1}{2}\sum_{n=0}^{\infty} P(n)\left(1 - \cos\left(\Omega_{n,n+1}t\right)\right). \tag{2.39}$$

For the blue sideband we have $\Omega_{n,n+1} = \eta\sqrt{n+1}\Omega_R$ and $P(n)$ the probability that the initial state is $|g,n\rangle = |g\rangle \otimes |n\rangle$. For a thermal state distribution over the Fock states with an average vibrational number $\bar{n}$ is given by

$$P(n) = \frac{\bar{n}^n}{(\bar{n}+1)^{n+1}}. \tag{2.40}$$

Assuming a thermal state distribution we can fit the probability for the blue sideband in Eq. 2.39 to find the mean vibrational number $\bar{n}$. This method has been used to determine experimentally whether transport in my experiments increases the motional state compared to the case where the ion is kept fixed in position, see chapter 6 for the results.

For completeness the motional subtracting sideband which is typically refered to as the red sideband (rsb) is obtained when the laser detuning is set to $\delta \approx \Delta - \omega_z$.

$$\hat{H}^{(\mathrm{rsb})} = \hbar\eta\Omega_R \left(-ie^{-i\left(\Delta t - \vec{k}\cdot\vec{z} + \varphi\right)} \hat{a}^\dagger \hat{\sigma}_- + ie^{+i\left(\Delta t - \vec{k}\cdot\vec{z} + \varphi\right)} \hat{a}\hat{\sigma}_+ \right). \tag{2.41}$$

For $\Delta = 0$ $\hat{H}^{(\mathrm{rsb})}$ is equivalent to the Jaynes Cummings Hamiltonian used in quantum optics.

2.3 Two-photon stimulated Raman transitions

In our beryllium ion we address two energy levels (see chapter 3) via a two-photon stimulated Raman transition. For the coherent manipulation of the two states we make use of a three level Λ-system two which we apply two off-resonant laser fields $\vec{E}_{\mathrm{r,b}} = \vec{\epsilon}_{\mathrm{r,b}} E_{\mathrm{r,b}} \cos\left(\vec{k}_{\mathrm{r,b}} \cdot \vec{R} - \omega_{\mathrm{r,b}} + \phi_{\mathrm{r,b}}\right)$ with the wave vectors and frequencies indicated in Fig. 2.4. and $\phi_{\mathrm{r,b}}$ the phase of each beam respectively and "r" and "b" indicating the red and blue Raman beams. The Hamiltonian for describing the system is given by

$$\hat{H} = \sum_{n=2,3} \hbar\omega_{n,1} + \vec{d} \cdot \sum_{i=r,b} \vec{E}_i. \tag{2.42}$$

From the theory covered in [Wineland 98] we find that when we apply both lasers simultaneously the three level Λ-system under adiabatic elimination of state $|3\rangle$ can be well describe by a two-level system of states $|1\rangle$ and $|2\rangle$. The effective interaction Hamiltonian is given by [Lo 15]

$$\hat{H}_I = \frac{\hbar\Omega_0}{2} \left(|2\rangle\langle 1| e^{i\left(\Delta\vec{k}\cdot\vec{R} + \Delta\phi\right)} e^{-i\delta' t} + |1\rangle\langle 2| e^{-i\left(\Delta\vec{k}\cdot\vec{R} + \Delta\phi\right)} e^{i\delta' t} \right) \tag{2.43}$$

with $\Delta\vec{k} = \vec{k}_b - \vec{k}_r$, $\Delta\phi = \phi_b - \phi_r$ and $\Omega_0 = |\Omega| = g_{\mathrm{b}} g_{\mathrm{r}}^* / 2\Delta$ with $g_{\mathrm{r,b}}$ the coupling coefficients between levels $|3\rangle$ and $|1\rangle$ given by $g_b = E_b \langle 3|\vec{d}\cdot\vec{\epsilon}_b|1\rangle/\hbar$ and levels $|3\rangle$ and $|2\rangle$ given by $g_r = E_r\langle 3|\vec{d}\cdot\vec{\epsilon}_b|2\rangle/\hbar$. The detuning term $\delta' = \delta - \delta_{\mathrm{sr}} + \delta_{\mathrm{sb}}$ comes from the choice of level reference with δ the difference between the detunings of the red and blue Raman beams. The last two terms δ_{sr} and δ_{sb} are the AC-stark shifts caused by the off-resonant driving of each beam.

Considering two choices of the setup of the two Raman beams as shown in Fig. 2.5, which indicated a co-propagating and a 90 degree setup. The difference

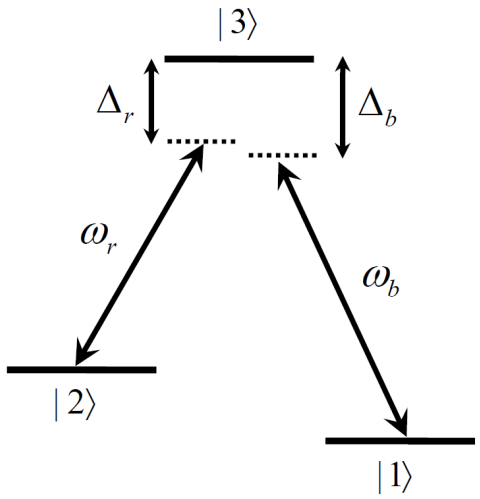


Figure 2.4: Λ-system energy level configuration in which two-photon stimulated Raman transitions are driven. Figure from Hsiang-Yu Lo's thesis [Lo 15].

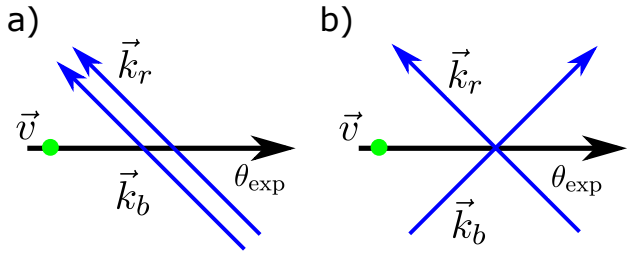


Figure 2.5: Two possible choices for the setup of the Raman laser beams in our experiments a) Co-propogating and b) 90 degree angle with each other. The relative setup of the Raman beams are shown with respect to the transport direction of a $^9Be^+$ ion indicated by the velocity vector $\vec{v}$. The angle θ_{exp} is 45 degrees in our setup.

wave vector is given by,

$$\Delta\vec{k} = \vec{k}_b - \vec{k}_r \tag{2.44}$$

$$\vec{k}_r = \frac{2\pi c}{f_r}\left[\cos\left(\theta_1 \times \frac{\pi}{180}\right), \sin\left(\theta_1 \times \frac{\pi}{180}\right)\right] \tag{2.45}$$

$$\vec{k}_b = \frac{2\pi c}{f_b}\left[\cos\left(\theta_2 \times \frac{\pi}{180}\right), \sin\left(\theta_2 \times \frac{\pi}{180}\right)\right] \tag{2.46}$$

with the experimental relevant frequencies being $f_r = \frac{c}{313.2727\,\text{nm}}$ and $f_b = f_r + 1.25\,\text{MHz}$, where 1.25 MHz is the frequency splitting between states $|1,0\rangle$ and $|2,0\rangle$ in $^9Be^+$ (see chapter 3 for an energy level diagram of $^9Be^+$). The angles θ_1 and θ_2 should be close to -45 degrees and in chapter 8 a difference

between of the two beam angles of 10^{-3} degrees have been observed. For the co-propagating beams assuming both angles are at -45 degrees results in $\Delta\vec{k} = 18.53 \times [1, -1]\,1/\mathrm{m}$. For $\vec{k}_b$ at -135 degrees and $\vec{k}_r$ at -45 degrees corresponding to the 90 degree setup in Fig. 2.5b we have $\Delta\vec{k} = 2.84 \times 10^7 \times [-1, 0]\,1/\mathrm{m}$. The effect of the Doppler shift $\Delta\vec{k} \cdot \vec{v}$ is therefore much reduced in the case of a co-propagating arrangement of Raman laser beams compared to the 90 degree setup.

2.4 Final remarks

I have covered the basic background in ion trapping physics which is required to follow the physics which will be presented in later chapters. Take note that each of my following chapters is written with self contained theory required to understand the topics that are presented within.

Chapter 3

General experimental overview

In this chapter I will briefly give an overview of the relevant apparatus which have been used within the experiments. This chapter serves only as a guide to understand the general apparatus that were used and the setup in which they were performed. Detailed descriptions of the experimental setup can be found in Daniel Kienzler's and Hsiang-Yu Lo's theses [Kienzler 15, Lo 15].

3.1 Trap

The trap shown in Fig. 3.1 consists of 60 electrodes in total with two carrying radio frequency (RF) voltages. It is built by stacking up gold coated wafers which were laser cut as depicted in Fig. 3.2. The trap was specifically designed and optimized for working with both $^{40}Ca^{+}$ and $^{9}Be^{+}$ ions. The top and bottom masks are there to prevent the coating of the electrodes (shims-, DC- and RF-electrodes) with neutral calcium and beryllium atoms when loading. There are 28 electrodes used for micro-motion compensation indicated as shim-electrodes in Fig. 3.2. They have not to date been varied during experimental sequences. We typically change the applied voltages to compensate the micro-motion once before taking data and do not change it until the experiment has completed. For the experiments a simplified sketch of the trap is shown in Fig 3.3 which shows 2×15 DC-electrodes opposing each other on different wafers in the trap. These electrodes are used in transport and their sizes are summarized in Table 3.1. Even though there are in total 30 DC-electrodes which could be controlled individually due to the symmetry of our system the opposing electrodes could be controlled in common mode by shorting them together, thus requiring only 15 voltages to be applied. The middle of electrode 8 is where the origin of the z-axis is located and this zone will be referred to as the experimental zone in this thesis. The negative direction is chosen to lie on the side of electrode 1, therefore on the side where we typically load our ions. The trap has been developed by Daniel Kienzler

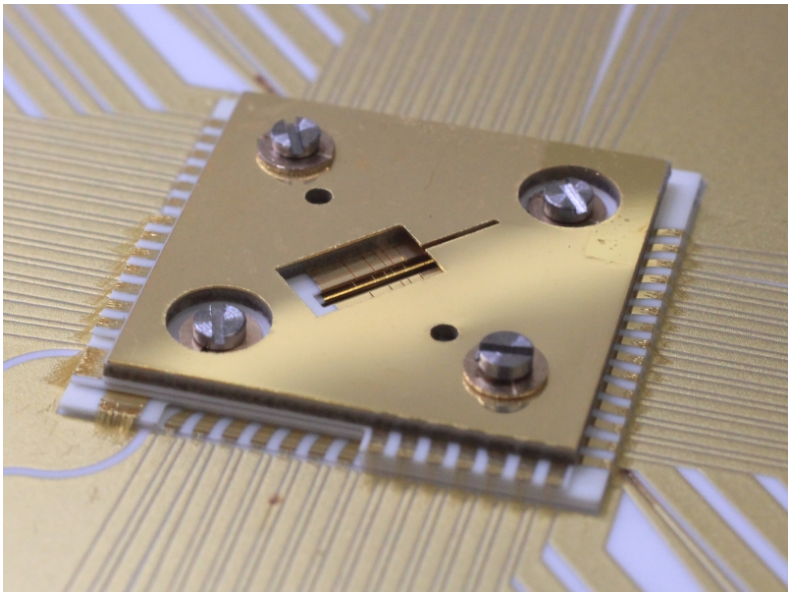


Figure 3.1: 3D Paul trap used in the experiments, built by Daniel Kienzler [Kienzler 15].

e1	e2	e3	e4	e5	e6	e7	e8
1000 µm	500 µm	500 µm	300 µm	155 µm	155 µm	155 µm	300 µm
e9	e10	e11	e12	e13	e14	e15	-
155 µm	155 µm	155 µm	300 µm	500 µm	500 µm	1000 µm	-

Table 3.1: Sizes of the DC-electrodes, all separated by 20 µm.

and more detail regarding to manufacturing the 3D Paul trap can found in his thesis [Kienzler 15].

3.2 Experimental setup overview

Fig. 3.4 shows the periphery of the vacuum system, including laser beams, magnetic field coils and electrical connections. The ion trap is in reality only a few millimeters and is enlarged here to orient the trap clearly with respect to the laser beams.

A day of experiments starts by loading ions from a neutral atom source by either electron bombardment or photoionization [Lucas 04]. Photo-ionization loading is convenient, requiring low oven flux and avoiding electrode charging associated with electron bombardment. During this process both the photoionization and the Doppler cooling beam are turned on simultaneously with the latter to cool captured ion(s). Ionized atoms are thus trapped and Doppler

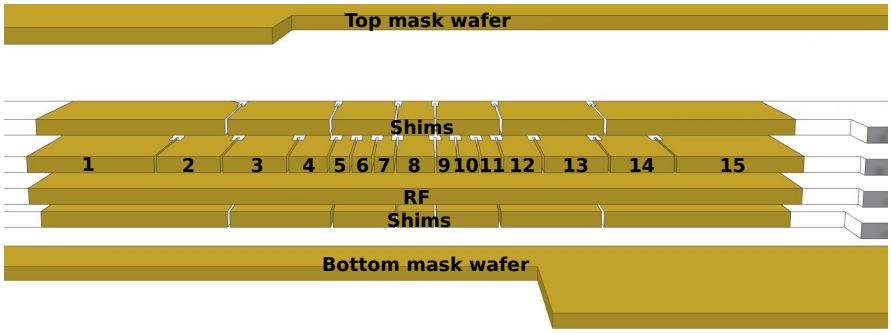


Figure 3.2: Wafer stacks showing the top and bottom mask wafers and the electrodes used for the shims, RF and the DC-electrodes used in confining the ion in the z-direction. Final manufactured design is shown in Fig. 3.1.

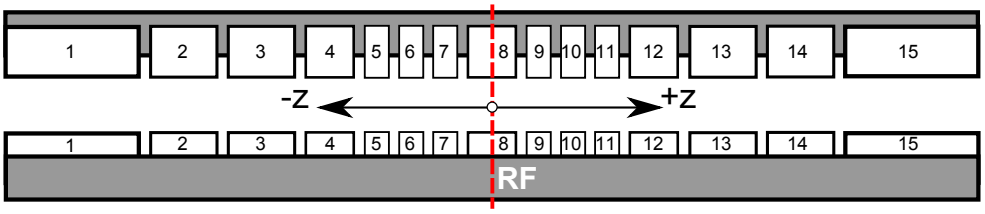


Figure 3.3: Representation of the dc-electrodes used in transport as well as the reference frame for the z-axis with the origin placed in the middle of electrode 8.

cooled. After loading a generic run of an experiment uses a sequence similar to that shown in Fig. 3.5. The experiment starts with Doppler cooling followed by manipulating the ions in order to produce quantum state(s) of interest. After performing quantum state manipulations a readout is done by state dependent florescence. The experiment has to be performed several hundred times in order to build up enough statistics in order to characterize the quantum state that has been created. For details on the optical setup and how to work with $^{40}Ca^{+}$ see Daniel Kienzler's thesis [Kienzler 15] and for an in depth discussion as well as simulations on the various polarization considerations of 866 nm and 397 nm see Hsiang-Yu Lo's thesis along with details on how to work with $^{9}Be^{+}$ [Lo 15].

3.3 Calcium and beryllium as qubits

For the experiments performed with transport we have used both calcium and beryllium, the two ion species we can work within our 3D Paul trap.

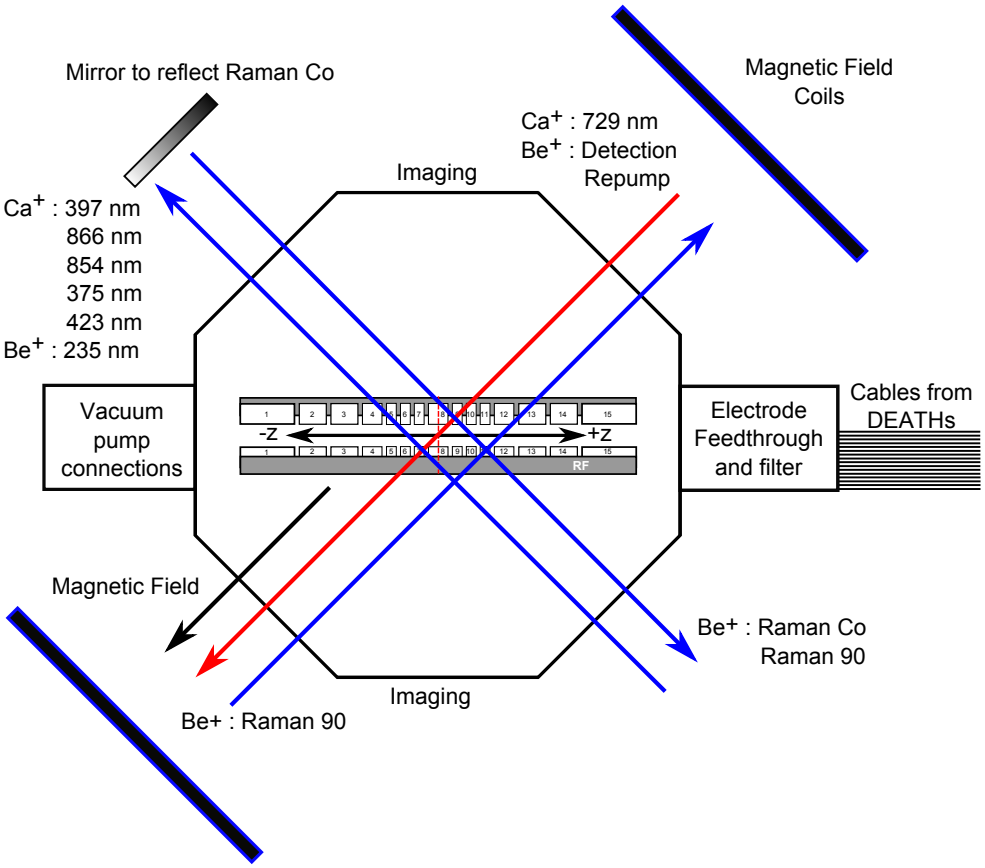


Figure 3.4: Overview of the laser beam configuration showing the all the relevant laser wavelengths used and from where they enter the vacuum can. The location of the magnetic field coils, the vacuum pump connections and the electrical feedthroughs are shown as well as the mirror placement used to reflect the Raman Co-propagating beam used in my experiments.

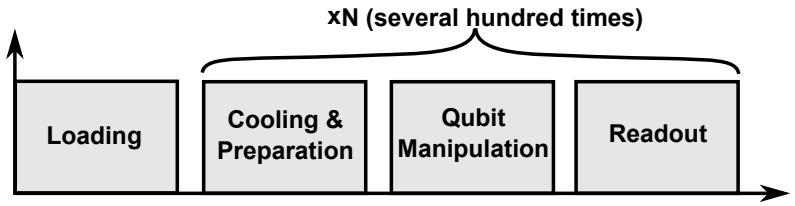


Figure 3.5: The typical procedure in ion trapping experiments. After having trapped an ion, typical experiments involve Doppler cooling, qubit manipulation and the readout of the state. These experiments has to be carried out multiple times in order to obtain enough statistics to discern the quantum state with high accuracy.

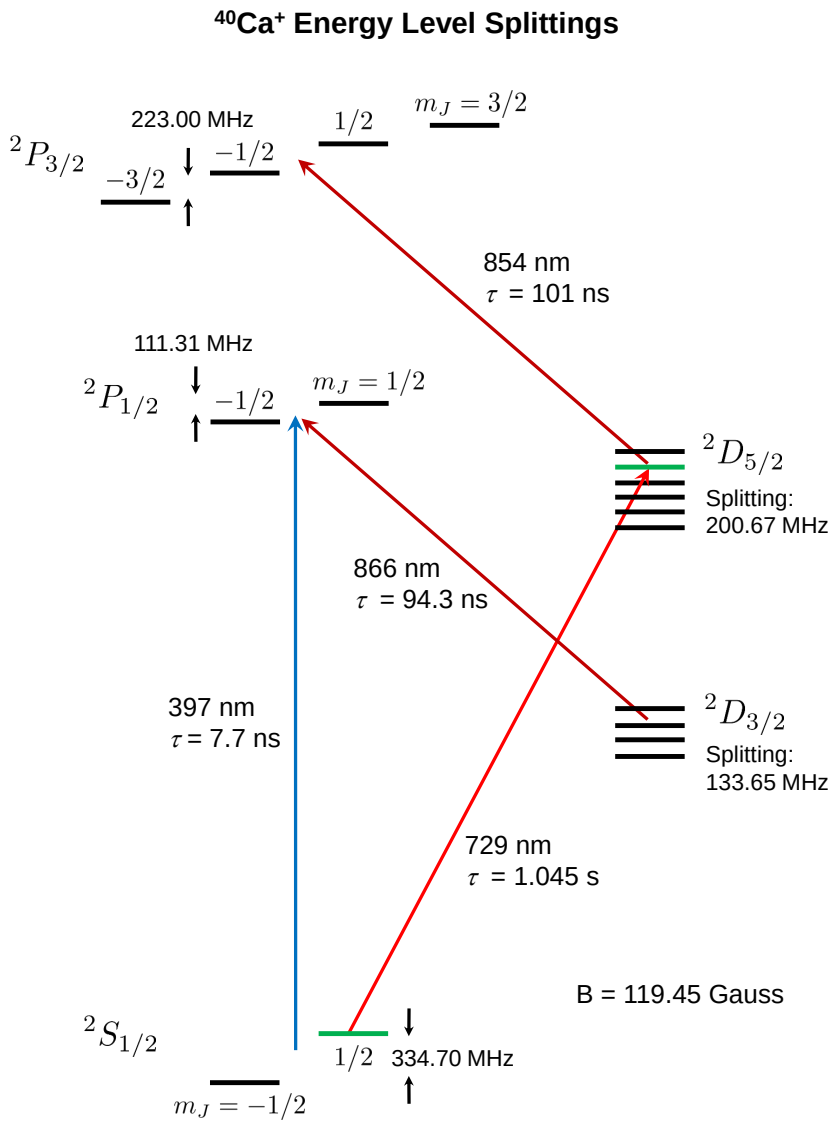


Figure 3.6: $^{40}Ca^{+}$ ion energy level diagram indicating the splittings resulting from the application of a 119.45 G external magnetic field. Additionally the laser wavelengths used in our experimental setup to work with $^{40}Ca^{+}$ ion are indicated along with the decay rates. Figure from Hsiang-Yu Lo's thesis [Lo 15].

3.3.1 Calcium ion

To obtain calcium ions we start with a two-photon photoionization process to ionize neutral calcium atoms by using 423 nm and 375 nm laser light. The 423 nm excites the atom from the ground state $4s^1S_0$ to an excited state $4p^1P_1$ and the 375 nm ionizes the atom by driving the excited state $4p^1P_1$ to the continuum. The energy level diagram for the calcium ion along with the level splittings caused by an external magnetic field of 119.45 G is shown in Fig. 3.6. We use the 397 nm and 866 nm for Doppler cooling and for the quantum state readout. Our qubit state of choice is encoded in the $|S_{1/2}, m_J = 1/2\rangle$ and $|D_{5/2}, m_J = 3/2\rangle$ states. The $|S_{1/2}, m_J = 1/2\rangle \leftrightarrow |D_{5/2}, m_J = 3/2\rangle$ is a quadrupole transition ($\Delta J = \pm 2$) which we drive with a 729 nm laser. Due to the longevity of the D-level we require a way to reset a qubit. The transition $P \leftrightarrow D$ ($\Delta J = \pm 1$) is an electric dipole allowed transition which allows for easy reset of the qubit by applying the appropriate laser light frequencies. The P- and D-levels both have fine-structure and in order to clear the D-level we apply the 866 nm and 854 nm laser light to the $D_{3/2} \leftrightarrow P_{3/2}$ and $D_{5/2} \leftrightarrow P_{3/2}$ transitions. The P-level has a very short lifetime (few ns) and decays quickly to $S_{1/2}$, $D_{3/2}$ and $D_{5/2}$. Using optical pumping the 397 nm, 866 nm and 854 nm light is used to initiate the state in $|S_{1/2}, m_J = 1/2\rangle$.

3.3.2 Beryllium ion

Unlike calcium, beryllium does not have an easily accessible D-level and the qubit states are encoded in the hyperfine structure levels of $S_{1/2}$ manifold. By using a magnetic field, the hyperfine structure splits up and transitions between these states can be driven by using detuned Raman beams. An energy level diagram with the splittings caused due to the application of a 119.45 G external magnetic field is shown in Fig. 3.7. The laser wavelengths used in all transitions are 313 nm with only sub-nanometer differences used to address specific transitions. Since all hyperfine levels are long-lived, we may choose convenient pairs of levels as qubits. The simplest is to encode a qubit which will be referred to as a field-dependent qubit (FDQ) on the $|F = 2, m_f = 2\rangle$ and $|F = 1, m_F = 1\rangle$[1]. The energy splitting between the states of the FDQ is sensitive to magnetic field fluctuations to first order. A better choice for a memory qubit is the $|1, 1\rangle \leftrightarrow |2, 0\rangle$ transition which will be referred to within this thesis as the field-independent qubit (FIQ), which has zero first order field dependence at a magnetic field strength of 119.45 G. By operating close to this point long-lived coherence is readily achieved [Langer 05b, Lo 15]. We have therefore chosen to work at this field strength and the frequency splitting for $S_{1/2}$ is 1.207 GHz.

Initialization of the $|2, 2\rangle$ state in beryllium is performed by optical pumping. This requires simultaneous application of different frequencies of 313 nm laser

[1]From here on I will make used of the following notation $|F = n, m_F = k\rangle = |n, k\rangle$

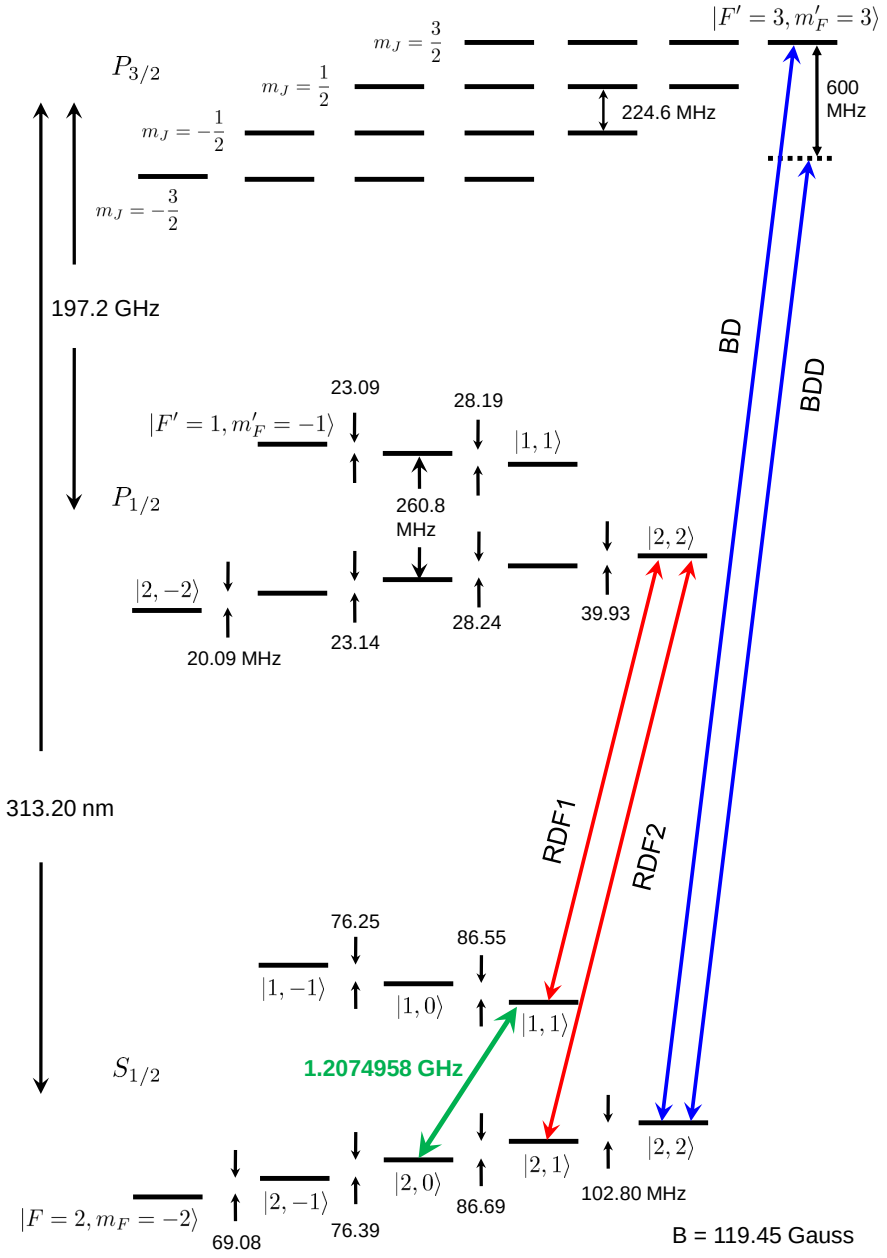


Figure 3.7: $^9Be^+$ energy level diagram along with the splittings as caused by an application of a 119.45 G external magnetic field. The laser beams labelled as BD, BDD, RDF1 and RDF2 are used for state initilization. The frequency splitting for the FIQ states are indicated in green. Figure from Hsiang-Yu Lo's thesis [Lo 15].

beam. Two different frequencies, the red-detuned Doppler F2 (RDF2) and red-detuned Doppler F1 RDF1 are required in order to drive the transitions $|S_{1/2}, F = 2, m_F = 1\rangle \leftrightarrow |P_{1/2}, F = 2, m_F = 2\rangle$, $|S_{1/2}, F = 1, m_F = 1\rangle \leftrightarrow |P_{1/2}, F = 2, m_F = 2\rangle$. Another two frequencies the blue BD and blue Doppler detuned (BDD) are used to drive the transition $|S_{1/2}, F = 2, m_F = 2\rangle \leftrightarrow |P_{3/2}, F = 3, m_F = 3\rangle$ and to off-resonantly drive the same transition with the frequency BDD red detuned by 600 MHz. The excited P states are short lived (8.2 ns) and the $P_{3/2}$ decays to $|S_{1/2}, F = 2, m_F = 2\rangle$ and $P_{1/2}$ to $|S_{1/2}, F = 2, m_F = 1\rangle$, $|S_{1/2}, F = 1, m_F = 1\rangle$ and $|S_{1/2}, F = 2, m_F = 2\rangle$. By using optical pumping we therefore prepare the state in $|S_{1/2}, F = 2, m_F = 2\rangle$ relying on the fact that the excited states are short lived.

State-dependent florescence is used to readout the beryllium ion and is carried out on the closed transition $S_{1/2} \leftrightarrow P_{3/2}$ as indicated in Fig. 3.7 by the applied blue Doppler (BD) laser beam. When working with the FIQ the population should be transferred from $|1, 1\rangle \rightarrow |2, 2\rangle$ with the FDQ Raman beams before readout. To ensure that when working with the FIQ that the state $|2, 0\rangle$ is not off-resonantly depopulated by the readout process we shelve the population to a darker state $|1, -1\rangle$. Using the same pair of laser Raman beams which is further detuned, we can drive the transition $|2, 0\rangle \leftrightarrow |1, -1\rangle$. Here shelving of the population implies driving a transition such that all the population is transferred from one state to another and in our case the shelving state is $|1, -1\rangle$. More details on the beryllium qubit and setup can be found in Hsiang-Yu's thesis [Lo 15].

3.4 Laser setups for transport experiments

Within this thesis experiments are reported in which the orientation of the transport direction with the respect to the beam's wave vector has significant impact on observed results. For calcium the qubit control field is the 729 nm laser. When transporting an ion the laser field is Doppler shifted in the rest frame of the ion by an amount $\vec{k} \cdot \vec{v}$, where $\vec{k}$ is the laser beam wave vector and $\vec{v}$ the velocity of the ion in transit. The Doppler shift has significant observable impact and is reported on in chapter 7 in more detail. In our experiments the laser beam access is limited to 45 degrees with respect to the z-axis as shown in Fig. 3.4. Considering only the trap and the 729 nm laser beam the setup as was used in the experiments is shown in Fig. 3.8. For the given setup the Doppler shift $\vec{k} \cdot \vec{v}$ is negative for an ion moving in the positive z-axis direction and positive if the ion moves into the negative z-axis direction. For the experiment reported in chapter 7 we have always transported into the positive z-direction.

In the experiments using $^9Be^+$ ions, we address the qubit states with two co-propagating Raman beams. In the co-propagating beam configuration, the

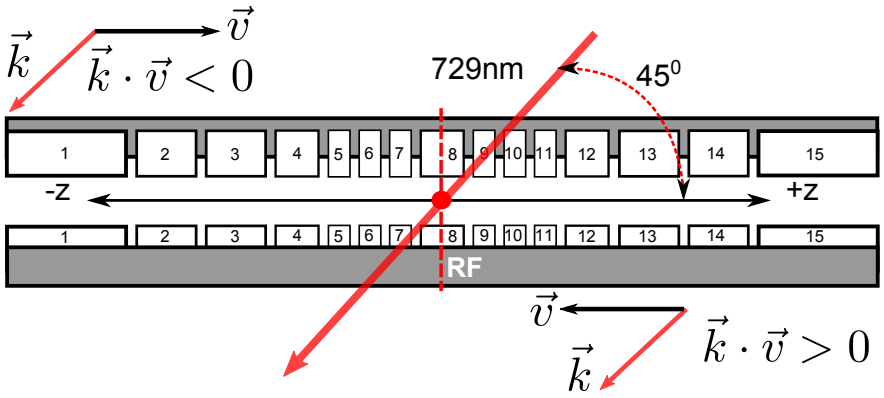


Figure 3.8: Calcium laser incoming direction with respect to the trap orientation. The Doppler shift $\vec{k} \cdot \vec{v}$ is negative for transport along the positive z-axis direction and positive for transport in the negative z-axis direction.

effective wave vector $\Delta\vec{k} = \vec{k}_2 - \vec{k}_1$ is the difference of the wave vectors $\vec{k}_1$ and $\vec{k}_2$ of each beam, which leads to negligible $\Delta\vec{k} \cdot \vec{v}$ terms. The setup of the Raman co-propagating beams with respect to the trap axis is shown in Fig. 3.9. We have performed a multi-process experiment as well as a transport Ramsey experiment, both are detailed in chapter 8. For the Ramsey experiment only a single laser beam in the experimental zone was required. The beam comes in at an angle of -45 degrees with respect to the z-axis. For the multi-process experiment two ions are trapped in different zones of the trap were used as depicted in Fig. 3.9. In order to demonstrate the scalability of transport gates we then retro-reflected the Raman beams back into the trap and made it cross the z-axis at a different location.

3.5 External Filters

To prevent high frequency noise components getting through to the electrodes there are filters both inside and outside of vacuum. Within the vacuum can on the filter board holding the trap there are RC-filters on the DC-lines with a corner frequency at $f_c = 809\,\text{kHz}$. The componant values are shown in Fig. 3.10.

The out-of-vacuum filter boards are designed to work together with the in-vacuum stage in order to realize a second-order Butterworth filter with a cut-off frequency of 250 kHz. The components used are shown in Fig. 3.10. The Bode plot of the filters are shown in Fig. 3.11. For faster transport experiments, the transfer function of the filters must be taken into account in designing the applied waveforms. The transport experiments that were carried out were slow enough not to be influenced by the linear response of the filters. Recently we have found that the filter we have used distorted our

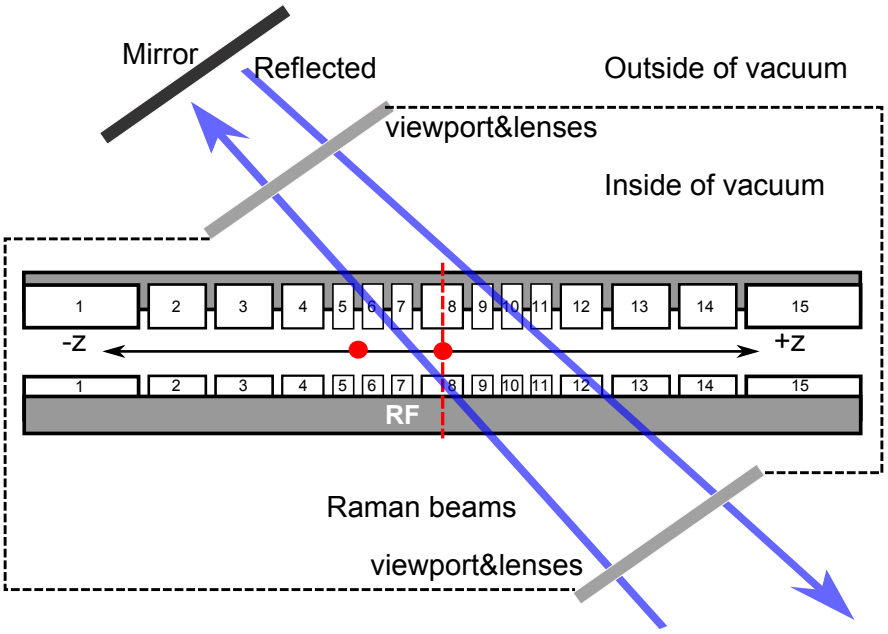


Figure 3.9: Illustration of the setup as was used to perform the multi-process experiment detailed in chapter 8. Co-propagating Raman beams enter the vacuum can via a viewport and then passed through the chamber to be reflected back through in order to address two locations along the ion trap used in the experiment.

waveforms severely due to the amplifiers in the DAC system not being able to provide enough current to the filter to work in the linear regime. Redesigning the filters to use lower capacitor values, higher resistance and inductor values has solved the problem. An in depth discussion will follow in Robin Oswald's masters thesis. The main effect of the filters is to delay the input by approximately 1.8 µs as obtained from the LTspice simulation if the signal contains frequency components below the cutoff. We have also accounted for the filters on occasion during the experiments and found no difference when transporting an ion at 5-8 m/s over 1 mm. For much faster transport the filters will play a more significant role. Details about the external filter design will follow in Vlad Negnevitsky's thesis.

3.6 Experimental control

An experiment consists typically of sequences of laser pulses and control over the triggering of experimental equipment with Transistor-Transistor Logic (TTL) outputs. Fig. 3.12 shows a simple diagram for our control system and the units controlled. Experimental control is carried out by a master

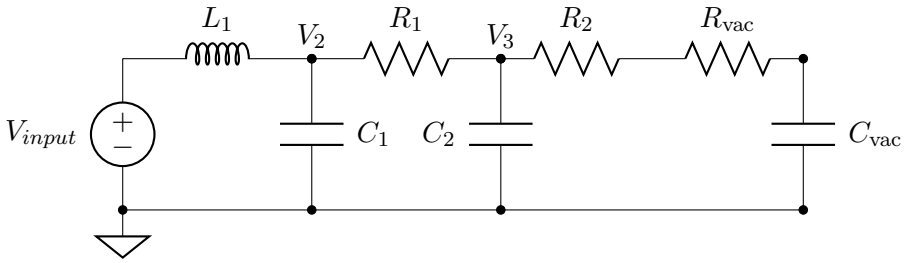


Figure 3.10: Circuit analysis of trap with external and internal filters included. External/internal referring to either being outside/inside of vacuum. Outside of vacuum component values are $L_1 = 4.7\,\mu\text{H}$, $R_1 = 9.1\ \Omega$,$R_2 = 536\ \Omega$, $C_1 = 39\,\text{nF}$ and $C_2 = 120\,\text{nF}$. Inside of vacuum components are $R_{\text{vac}} = 240\,\Omega$ and $C_{\text{vac}} = 820\,\text{pF}$.

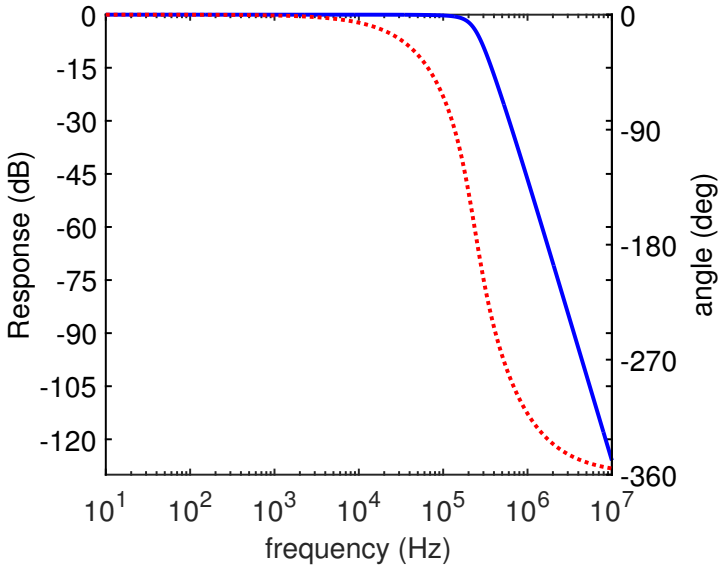


Figure 3.11: AC-analysis for the filters displayed in Fig. 3.10 used in the experiments.

FPGA and a control computer that are relying information to one another. The FPGA is a Xilinx Zynq FPGA which houses two ARM Cortex-A9 predecessors which communicates with the programmable logic (PL) via the AXI bus. We use the Zedboard development kit which is commercially available[2]. Experiments are written in C++ and are run on one of the A9 processors. The system is setup such that we can change variables of a given experiments using the control software running on the PC, which serves as a graphical user interface (GUI) containing multiple pages with user interfaces (UI) to each experiment. It is a further development of NIST's *Ionizer* software created by Til Rosenband, but it has been heavily modified and further developed by Ben Keitch, Matteo Marinelli and David Nadlinger. Each of the UI to the experiments are automatically generated. The experimental sequencing is handled by a pulse sequencer that had been implemented on the PL of the FPGA. The experiments running on the A9 processor relate the information to the PL via the AXI bus. The Zedboard has 32 TTL lines which we can relay to various equipments like shutters, RF-switches and the Direct Ethernet Adjustable Transport Hardware (DEATHs, see chapter 5). Additionally direct digital synthesizer (DDS) boards are connected and controlled from the Zedboard via a backplane. Each of the DDS boards holds four DDS channels and are controlled from a Spartan 6 FPGA. The DDS boards and the backplane in which they are held were designed and manufactured by Enterpoint[3]. Vlad Negnevitsky programmed them such that they can perform pulse shaping, retain phase coherence between channels and build up complex frequency pulses, details will follow in his PhD thesis. A breakout board was designed and manufactured to allow the Zedboard to run on the same backplane as the DDSs. The overall system allows for quick experimental sequence development in an intuitive way.

3.7 Final remarks

The equipments used in our laboratory have been built and improved upon by most members of our research group and more details of many of these elements can be found elsewhere (see www.tiqi.ethz.ch). My principal contribution has been to build up a digital PI controller (EVIL, which it only is on the rare occasion when it goes out of lock) and the multi-channel arbitrary waveform generator (AWG) supplying the voltage waveforms to the trap electrodes (which go by the acronym DEATH, though they aren't particularly dangerous).

[2]ZedBoard development kit: http://zedboard.org/product/zedboard

[3]Milldown DDS channel-card from Enterpoint : http://www.enterpoint.co.uk/ in collaboration with Ben Keitch from our group

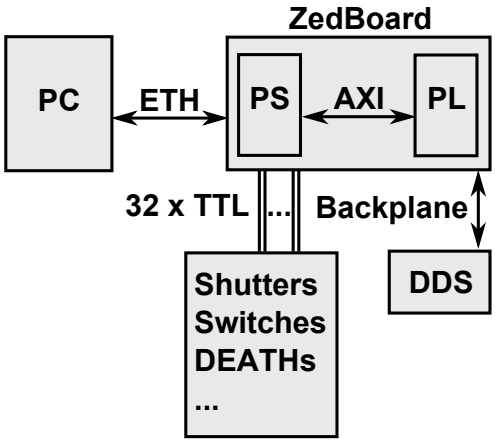


Figure 3.12: Diagram depicting connectivity and control of various devices via TTLs and DDS with our control software *Ionizer*.

Chapter 4

Electronically Variable Interactive Lock-box

The Electronically Variable Interactive Lock-box (EVIL) is a digital PI-controller that has been developed in our group by myself and Vlad Negnevitsky. Alexander Hungenberg, David Nadlinger and Christoph Fischer have done projects on it and have made various improvements over time to the software and firmware. The EVIL is control unit and often referred to as a "lockbox" similar to the typical used devices like the Digilock and the FALC units with the latter units being much more expensive (approx. 2500 CHF). We were originally looking into building a cheap solution PID controller, since we would require a fairly large amount of PID type controllers. Theses units were aimed at typical use in our laboratory in order to lock lasers to cavities, laser to reference cell (Iodine), noise reduction for a tapered amplifier and fiber noise cancellation and many other applications. After making the EVILs available for purchase they cost 900 CHF for two control channels. Additionally the EVILs are general purpose digital signal processing units of which the signal processing part can be modified to an user's liking.

4.1 Overview

In Fig. 4.1 a simplified schematic is given of the EVILs. The device accepts two analog input signals of which both inputs can be varied with potentiometers. The signals are then amplified. The gain of the amplifiers stage can be set using potentiometers. Two similar Analog-to-Digital Converters (ADCs) were used in the design, namely 14bit AD9215 from Analog Devices. The output of the ADCs are digital signals that are routed to a commercial FPGA (Field-Programmable Gate Array), the Papilio One 500K[1]. It is the FPGA that contains all the custom Verilog code which handles communication and runs

[1]Designed by Jack Gasset: http://papilio.cc/

our digital PI-controller design. After signal processing the FPGA outputs digital signals to two different Digital-to-Analog Converters (DACs). One of the these are the DAC904 (section 1 Fig. 4.2) and the other a DAC7811 (section 2 Fig. 4.2). The DAC904 receives communication in parallel from the FPGA and is associated with the fast output and the DAC7811 receives SPI signal from the FPGA and is used for the slow output, due to SPI typically being much slower than parallel communication. The difference between the fast- and slow output is that there are different latencies from input to the ADCs until eventual output from the DACs, because of the communication protocols required by each chip. The slow-out channel has longer latency and with the PI controller in the loop we can only achieve a controllable bandwidth of 70 kHz on this channel, whereas on the fast-channel we have 500 kHz. The board as shown in Fig. 4.2 is our own design which contains the ADCs, DACs and power supply for itself and the FPGA, see Fig. 4.6d as an example when the FPGA is mounted.

For new users of the EVILs the input offset, gain and output gain can be changed via potentiometers on the PCB. They are indicated by the comments in Fig. 4.2 *Gain Out A*, *Gain Out B*, *Offset B*, *Gain B*, *Gain In A*, *Offset In A*. They match the diagram parameters in Fig. 4.1. The potentiometer *Gain Out A* and *Gain Out B* changes the output amplification of the respective fast and slow paths. The potentiometers *Gain A* and *Gain B* change the input gain of the signal to the ADCs for channel A and B and the *Offset A* and *Offset B* changes the input offset. For those interested in the output noise from the fast and slow outputs, the measurements made for each channel are shown in Fig. 4.3 and were made using a spectrum analyzer. The slight difference between the two measurement level offsets is believed to come from the spectrum analyzer (Fieldfox N9912A) having different specifications over the two different frequency ranges. The slow and fast out (Sout,Fout) show the same behaviour in the measurements, some frequency components around 2 MHz which are higher than that observed only for the spectrum analyzer measurement. These features are not simply explained by the components on on the EVIL and might even be picked from other devices. There is a clear 96 MHz peak as shown in Fig. 4.3b, this is the clock frequency the FPGA use to drive both the ADCs and the fast DAC. The frequency range of {10,100} MHz is clean except for the peak at 96 MHz, which is rather good for a device carrying a FPGA and which has various frequency signals coming out. A list of properties for the latest version of the EVILs are displayed in Table 4.1.

4.2 Development history

Ben Keitch and myself started with simple schematics we obtained from Thaned (Hong) Pruttivarasin who was in Hartmut Häffner group in Berke-

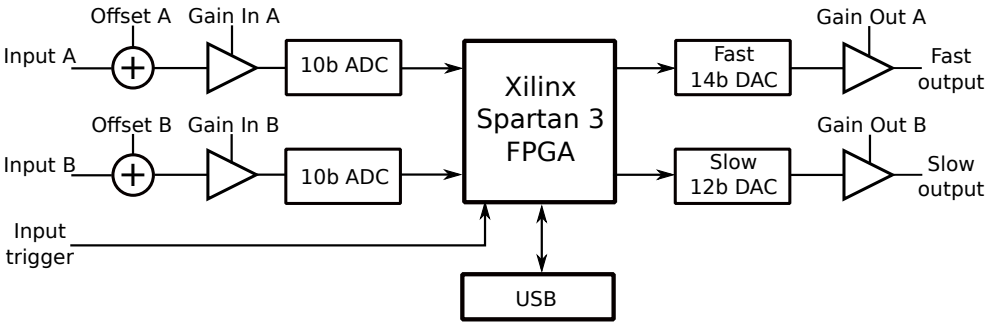


Figure 4.1: Flowchart showing the operation of the EVIL.

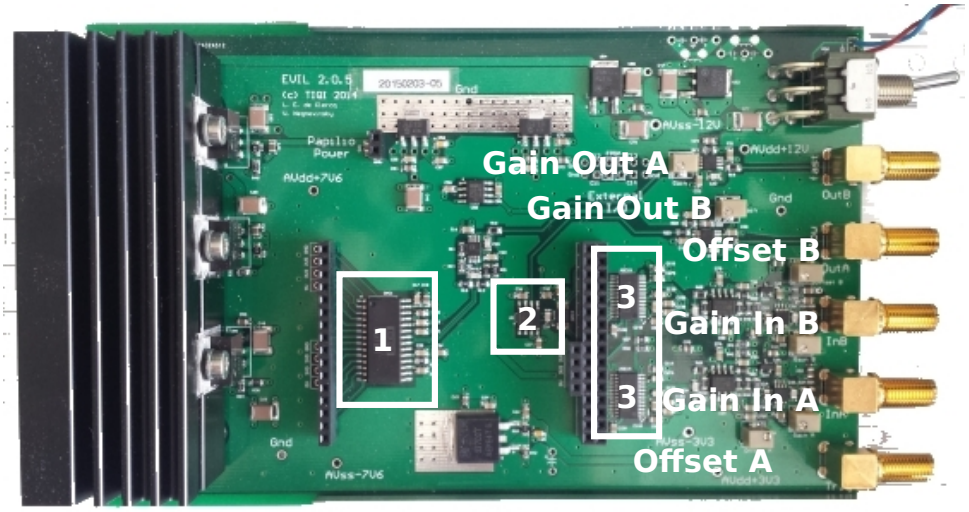
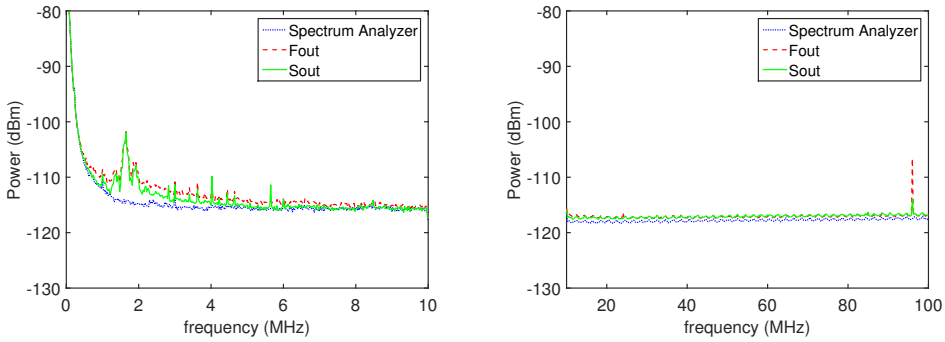


Figure 4.2: Top of EVIL PCB with some core sections highlighted.

Property	In implementation
PC-communication	USB
Data update rate	96 MHz
Output voltages	±10 V (adjustable down to ±1 V)
Input voltages	±10 V (adjustable down to ±100 mV)
Input input offset	±10 V (adjustable on channel A and B in new series)
TTL input	Yes
4x Digital lines	On back connector.
PSU	+15 V @750 mA and -15 V @200 mA and GND
System latency	250 ns (on fast path)
Fast PI controller	500 kHz
Slow PI controller	70 kHz

Table 4.1: Overview of the latest (2015) properties achieved by the EVIL.



(a) Noise measurement taken over the frequency range (0,10 MHz)

(b) Noise measurement taken over the frequency range (10,100 MHz)

Figure 4.3: Noise measurements made of the EVILs. The meaurements have been done in the laboratory which is typically considered a noisy environment. Both measurements have been carried out using a BW = 100 Hz and VBW = 30 Hz.

ley and with a few modifications built the first hardware for a PI-controller. We had two DSP (Digital Signal Processing) lines which each contained an ADC and a DAC connected via the FPGA. In the first design the input to the ADC , but only buffered with an op-amp and the output stage had a fixed gain of two. In the initial tests we could get some sort of lock, but it was unstable and we eventually realized a few mistakes in the design as well that we were limited by the fixed gain stage on the outputs. The PID controller implemented at the time was also not optimized and overall the system was difficult to debug.

Initially we implemented a Xilinx Microblaze processor core who's only purpose was to handle communication with a control computer via USB. The PL (Programmable Logic) was accessed by the processor and allowed changes to internal register in the PL from a PC via the processor. One could then set the control parameters via a serial communication. Getting the hardware and software to work together took quite a long time, starting from using Putty to control the internal parameters to using LabView and then Visual Studio. This required lots of code and still development was very slow and the GUI (Graphical User Interface) was slow and sometime unresponsive.

The project was initially created in Eagle and due to lack of control on the layout within the software we had to port the project over to Target3001! in order to correct for certain layout mistakes that could not be addressed in Eagle at that time. After Vlad Negnevitsky joined our group who had more experience in electronics he took the hardware design we had and with my inputs of what was lacking in the design re-created the project in Altium.

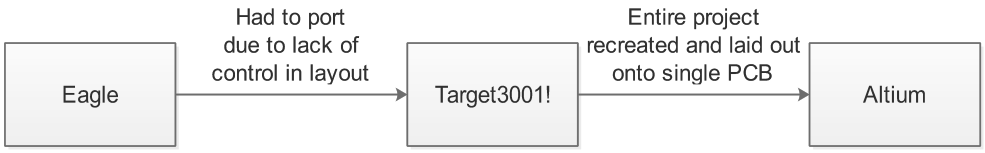


Figure 4.4: Flowchart of software that has been used within the EVIL project in order to design the hardware along with the reasons why we moved on to others.

The programs used from start to final are shown in Fig. 4.4. He kept most of the components we have used in the design and redesigned the input and output stages that would allow for input voltages in the range of ± 10 V as well as outputs within the same range. This solved the issues we had with fixed gains on the output and only buffering the inputs to the ADCs. The ADCs (section 3 shown in Fig. 4.2) and DACs (sections 1 and 2 shown in Fig. 4.2) used in our first design have been kept.

The DACs used are the DAC904 as indicated by section 1 in Fig. 4.2 for the fast output channel and the serial DAC DAC7811 shown in section 2 in Fig. 4.2. The DAC904 is a 14 bit DAC which can run at 165 MHz and the DAC7811 is a 12-bit DAC with a maximum serial clock speed of 50 MHz. The ADCs used are the AD9215 which is a 10 bit converter which can run at a maximum rate of 105 MHz. The Spartan 500E on the Papilio board is clocked by a 33 MHz clock and using a DCM (Digital Clock Manager) the frequency can be increased by making use of a PLL (Phase-Locked loop) to 96 MHz. This is the current clock we are using within the EVIL project.

After having redesigned the EVIL we took out the Microblaze processor and implemented custom logic to handle communication, this freed up a lot of space in the FPGA and compilation was much faster as well. The PID controller we have also worked on for a rather long time and have finally settled on implementing a simple PI-controller and spent most of the time making sure we have good enough control over the P and I parameters within the controller.

A crude but fair comparison between initial and final project is summarized in Table 4.2. The testing of the first board never got to the stage of checking trigger of a TTL. For both versions no single PCB design was used, for the initial design construction of a single unit could not be done by outsourcing the manufacturing due to the process required to build them not being reliable The initial unit consisted of three PCBs with their tasks being; supplying power, performing DSP and signal conversion. The current design has finally been outsourced where the devices are built and tested. For the EVIL project it would be advantageous to have in the future a single PCB design. This should reduce the overall cost of manufacturing since the production will consume less

Property	Initial design	Final design
PC-communication	USB via Microblaze	USB and custom core
Output voltages	±3 V	±10 V
Input voltages	±10 V	±10 V
Input Offset Correction	No	Yes
TTL input	Yes (never tested)	Yes
Single PCB design	No	No
Controller	Attempted PID	PI controller
Complexity to make	Hard	Simple

Table 4.2: Comparison between initial design and final design of the EVIL.

time with constructing and testing. Additionally not having to buy a separate commercial FPGA to mount should make the construction faster and avoid any unnecessary shipping costs and delays.

The main work that has been done after changing the PCB design and the Firmware that runs on the FPGA was in creating the software to control it. The initial control of board was performed via the command line for testing. Getting everything to work we moved the project to LabView and found that it was difficult to have the controller respond in real time to a sliding bar, which had to set the control parameters. we needed this to be as smooth as possible to allow an user to simply "lock" a given setup. Control parameters have to be changed often during initial operation of many locking circuits. Another problem we experienced in using LabView is that programming is not intuitive and structuring it in such a way that others can work on it easily is tough. Reprogramming the initial GUI (Graphical User Interface) from LabView in Visual Studio C++ allowed for the seamless change of control parameters without difficulty. The problem we ran into using Visual Studio C++ is that it did not allow for simple quick modifications to the GUI, the code excluding the gui was over 2500 lines of code. This was still not the best way to have others work on it and quickly get to grasp something that should be simple. Having seen the simplicity in PyQt, we yet again reprogrammed the entire GUI in PyQT. This allowed for simple modifications to the GUI and more complex features like an embedded display to allow for streaming of the error signal to the GUI on the control PC. Python is also a language that is often used in science, so we have seen it as only a win-win situation. With the total amount of lines of codes reaching around 450 with various complex features, this is still the current preference to simple user control we have found. A summary is shown in Fig. 4.5.

I would strongly recommend any newcomer to electronics design for lab use to consider this history and learn from our mistakes and successes. The basic take away points I have are the following

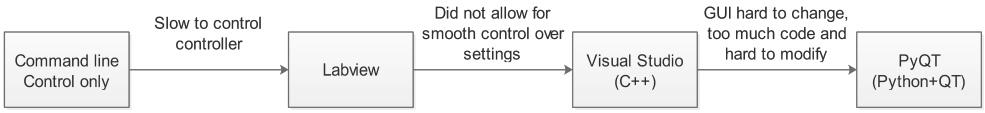


Figure 4.5: Flowchart representing the list of software that has been used within the EVIL projects and why we have moved on to others.

- In designing electronics, try and always have a simulation of sub-circuits of the bigger design. This helps in debugging.
- In designing electronics, use professional software like Altium. Features and functionalities found in such software save a lot of time (no need to source components with every new batch of PCBs being made).
- Use software like PyQT or QT in general. It allows for very fast development of GUIs and easy changes. Simplicity allows others to learn quickly and make their own changes.
- Use Verilog instead of VHDL (I am very biased here) it allows for fewer lines and again allows others to read and understand the code faster and make their own changes.

After Vlad and myself have spent a considerable amount of time to make the EVIL project simple, many students has worked in the EVILs and made major improvements to the software and firmware. Alexander Hungenberg who has implemented the display of the control signals within the GUI and communication improvements in the code [Hungenberg 13]. David Nadlinger who has made the EVIL run two independent PI controllers and several upgrades to the control code such that several can be run from one PC, see [Nadlinger 13]. Christoph Fischer who has worked on a Lock-in-Amplifier feature.

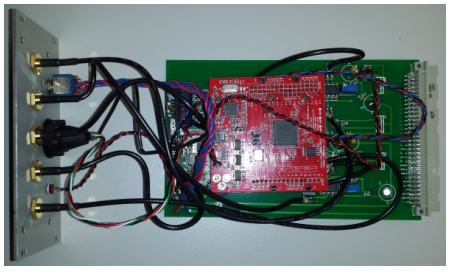
In Fig. 4.6 the first and the latest version (2015) are shown for comparison.

4.3 Projects on the EVIL

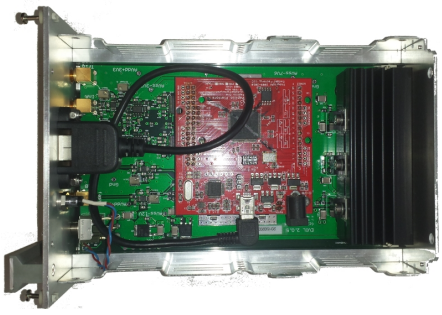
In this section two (out of several) of the projects that have been performed using the EVIL will be discussed. The PI controller was the first project and currently in the laboratory the mostly used. Pulse shaping required a TTL pulse as input and a pulse shape defined by the user to be output. This was the first investigation to see how to produce arbitary voltage shapes over time using fewer information than storing all steps that makes up a given shape in hardware.

(a) The first generation of the EVIL, front view.

(b) The latest used version, front view.



(c) The first generation of the EVIL, top view.



(d) The latest used version, top view.

Figure 4.6: The old and latest version (2015) of the EVIL. The old versions were hard to built compared to the latest version.

4.3.1 PI controller

Some of the most commonly used devices in most laboratories are so called "lock boxes". These can vary in shapes, sizes, connections to and from it and price. They all have one thing in common and that is that they act on feedback from a system it is controlling and that the device itself has a limited adjustable transfer function. The price of these devices are very strongly correlated to the properties of this transfer function as well as how it has been implemented and controlled. This function of the device is most commonly a PID controller which can be viewed as an adjustable low-pass filter. We have implemented a digital PI controller on the EVILs and in this section a description is given of PID control and the digital controller that has been implemented.

A PID controller can be described in time as

$$u(t) = K_p\epsilon(t) + K_i \int_0^t \epsilon(t')dt' + K_d \frac{d}{dt}\epsilon(t) \tag{4.1}$$

$$\epsilon(t) = V_{\text{set}} - V(t) \tag{4.2}$$

where $\epsilon(t)$ is the error signal given by the difference between the set value V_{set} and the current output $V(t)$. The parameters K_p, K_i and K_d are constants that can be set to change the influence of the various terms within the equation as well as in a controller. The first term represents the proportional part which is a fast acting term, since it acts on the current error signal. From the second term there is an accumulative effect based on some of the history of the error which is helpful to cancel drifts in the control. The final term is most often left out of most simple controllers, since it acts on the change of the error signal and can be tricky to deal with experimentally where signals tend to have some noise on them. In cases where it is required to act on the changes of the error signal, a low pass filter can be incorporated in the third term to avoid noise making the controller unstable [Mathworks].

To see how this controller can be implemented digitally, it is instructive to consider each term individually and how they can be approximated numerically which gives some intuition on how each term acts and how it is obtained. Within our setup an analog error signal as in Eq.4.2 is fed into the EVILs where it is digitized at the sampling rate T_s of the ADC. The sampled signal can be represented by $e[k]$ where each k represent a time t_k at which the signal was sampled. The time from one sample to the next is given by the sampling time $t_{k+1} - t_k = T_s$. A discrete output signal $u_p[k]$ proportional to the error signal $e[k]$ at time t_k is given by

$$u_p[k] = K_p e[k]. \tag{4.3}$$

The parameter K_p is known as the proportional coefficient. If we are interested in the scaled behavior of the error signal over time $u_i[k]$ we can integrate the

error signal which when sampling could be approximated by the Trapezoidal rule

$$u_i[k] = u_i[k-1] + K_I T_s \left(\frac{e[k] + e[k-1]}{2} \right) \tag{4.4}$$

The parameter K_i is known as the integral coefficient. This is also known in digital signal processing as the bilinear transformation or Tustin's Method. We may also be interested in how the signal changes between successive steps and we can therefore look at the derivative of the error signal $e[k]$ which can be written in the discrete form using

$$u_d[k] = K_D \left(\frac{e[k] - e[k-1]}{T_s} \right) \tag{4.5}$$

Here the parameter K_D derivative coefficient. The discrete equivalent of the PID action in Eq. 4.2 is the sum of Eqns. 4.3, 4.4 and 4.5.

$$u[k] = u_p[k] + u_i[k] + u_d[k] \tag{4.6}$$

Grouping the constants together and making the following assignments

$$\begin{aligned} K_1 &= K_p + K_i \frac{T_s}{2} + \frac{K_d}{T_s} \\ K_2 &= -K_p + K_i \frac{T_s}{2} - \frac{2K_d}{T_s} \\ K_3 &= \frac{K_d}{T_s} \end{aligned} \tag{4.7}$$

simplifies Eq. 4.6 upon substituting in $u_p[k]$,$u_i[k]$ and $u_d[k]$ to

$$u[k] = u[k-1] + K_1 e[k] + K_2 e[k-1] + K_3 e[k-2]. \tag{4.8}$$

This is the discrete equivalent of the continuous case in Eq. 4.2 when a real signal is sampled at a fixed time T_s. The derivation of the difference equation for implementation of a PID was done for completeness and it should be noted that which we have implemented in the EVILs are PI-controllers, thus $K_d = 0$ and the last term falls away.

Implementing the difference equation in digital logic on a FPGA (Field Programmable Gate Array) was a relatively easy task. It requires only the use of a few adders and registers to accumulate the sum. A challenging task we had was to find the ranges we wanted the proportional and integral term to be varied over. We wanted a produce an user interface in which someone does not run into a situation where they need more or less gain on these parameters than allowed by the controlling software. From us and others experimenting in the laboratory with the PI controller implemented in the EVILs we adjusted the ranges over time to allow for the most common ranges used in our laboratory such that each EVIL is not specifically programmed to each experiment.

This allows users to replace an EVIL simply and using their same settings as before.

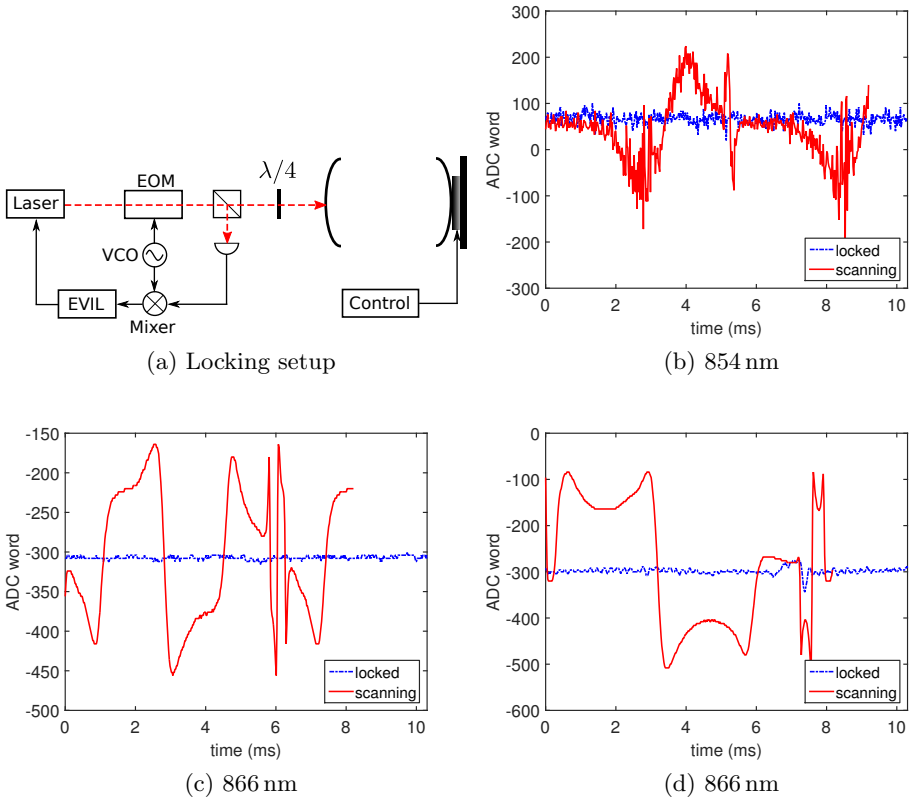
Typically in our laboratory the EVILs are used to either lock lasers to cavities or cavities to lasers. In order to do this reflected light from a cavity is measured on a photo-diode (PD). The output from the PD is mixed with a voltage controlled oscillator (VCO) and then fed into the EVIL to feedback on. The EVIL outputs a control signal to the laser and the control loop is then closed. What we typically then do experimentally is to change the cavity length slowly in order to change the frequency of the laser as indicated by *Control* in Fig. 4.7a. Examples of error signals as obtained from the Pound-Drever-Hall (PDH) technique [Keitch 07] are shown along with the corresponding locked signals in Fig. 4.7. The *ADC word* as indicated on the y-axis is the digital word sent by the 10-bit ADC in 2s compliment. In 2s compliment representation the numbers spanned by N-bits are $[-2^{N-1}; 2^{N-1} - 1]$, thus on the EVILs the range on the y-axis is always limited to $[-512; 511]$.

For locking a cavity to a laser the setup is as shown in Fig. 4.8 and is used typically in our laboratory as a doubling cavity. The reflected light from the cavity is measured with a photo-diode (PD) and the response of the PD is mixed with a voltage controlled oscillator (VCO) and is fed into the EVIL. The EVIL applies a control signal to a piezo which controls the length of the cavity and in the process we stabilize the cavity, assuming that the laser reference is stable.

4.3.2 Pulse shaping

Having worked with the EVILs we have come to realize that the FPGA we use has limited resources for storing information. This would be a major limitation for generating waveforms (voltage sequences to produce transport of ions, see chapter 6). In order to investigate the effect of shaped qubit pulses on the ions, we decided to design some digital logic that would allow for smooth output of voltages over time with a rather low resource consumption. For the problem we would not mind latency (input-to-output delay), as long as it is always fixed such that it does not influence our pulse sequences to address the ions. The idea was then to upload a small amount of information to represent a shaped pulse and then have that pulse triggered from the experimental controller sequence.

In order to implement this we decided to use Matlab's spline interpolation to represent my overall pulse shape. We chose to divide the signal in equal sections in time, where based on the signal we can increase the number of sections in which we divided the time to more accurately represent the signal. We chose the maximum order of the spline used in interpolation to be cubic. After running the spline interpolation for a given pulse shape the coefficients a_0, a_1, a_2 and a_3 are returned for each time section. The cubic function for



(a) Locking setup (b) 854 nm

(c) 866 nm (d) 866 nm

Figure 4.7: The error signal as obtained from the Pound-Drever-Hall (PDH) technique when scanning the laser current and the resulting signal when locked.

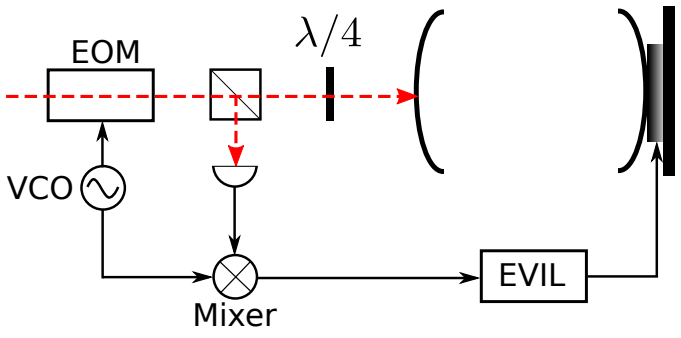


Figure 4.8: Setup for the Pound-Drever-Hall (PDH) technique for locking a cavity to a laser.

s	0	1	2	3	4	5	6	7
t_s	0	1	2	3	4	5	6	7
$f[s] = f(t_s)$	0	1	4	9	16	25	36	49
Δ^1	1	3	5	7	9	11	13	15
Δ^2	2	2	2	2	2	2	2	2
Δ^3	0	0	0	0	0	0	0	0

Table 4.3: Difference table for the function $f(t) = t^2$.

each chosen time segment can be represented as

$$f(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3. \tag{4.9}$$

This is the function that is then required to be implemented on the FPGA with arbitrary values for the coefficients a_n. There are two way to do this digitally, by direct multiplication or by finding an appropriate difference equation. Since each order in Eq. 4.9 requires multiplication of the value with itself, multiple multipliers would have to be used. Considering that in a digital circuit we are limited to fixed-length operations, meaning a limited number of hardware multipliers of a fixed width are available within a given FPGA. For the Spartan 500E which is used in the EVIL project there are 20 [Xilinx 13]. As a result, the difference equation approach is more attractive.

Given that we want to produce a function $f(t)$ on the output of the EVIL, we are limited to outputting only discrete numbers at the fixed rate given by the DACs update rate T. To do this we want to generate an array of output data $\{f[0],f[1],...,f[n]\}$ at times $\{t_0,t_1,...,t_n\}$ which to the bitwidths of the DACs represent the function $f(t)$ at the designated times, thus $f[s] = f(t_s)$ for $s\epsilon[0, n]$. For this Newton's forward difference formula [Mathworld] can be used to represent the function $f(x)$ using finite differences

$$\begin{aligned} f\left(t_0 + sh\right) &= f(t_0) + \sum_{k=1}^{\infty} \binom{s}{k} \Delta^k f(t) \\ \Delta^k f(t) &= \Delta^{k-1} f(t+h) - \Delta^{k-1} f(t) \\ \Delta^1 f(t) &= f(t+h) - f(t) \end{aligned} \tag{4.10}$$

An example using Eq. 4.3 is shown in the difference Table. 4.3 for the quadratic function $f(t) = t^2$. A useful condition on polynomial $f(t)$ of order d given as

$$f(t) = \sum_{k=0}^{d} a_k t^k \tag{4.11}$$

can be obtained from finite differences considerations on Eq. 4.10 when constant spaced outputs (h fixed) is considered. The condition is

$$\sum_{k=0}^{d+1}(-1)^{d-k+1}\left(\begin{array}{c} d+1 \\ k \end{array}\right) f(x+k) = 0 \tag{4.12}$$

Given the cubic-spline section as defined in Eq. 4.9 we can use Eq. 4.12 to obtain a discrete recurrence relation

$$f[n] = -f[n-4] + 4f[n-3] - 6f[n-2] + 4f[n-1]. \tag{4.13}$$

What we have learned now from Eq.4.12 and 4.13 is that you require to take as many consecutive points as the order of the polynomial which you would like to represent plus one additional point. The consecutive points could be at the update rate of the DAC that will have to do the conversion or less. Depending on the DAC that should perform the task, it will set the upper bound on the accuracy you need to achieve on the output. Given a DAC that can only output 12-bit numbers, we would like to achieve this precision within the recurrence relation. For determining the width of the registers we need to use in the FPGA implementation of the recurrence relation given by Eq. 4.13.

What we gain from the recurrence equation is that we only have to use accumulators and registers in logic if we are willing to supply the values $f[n-4], 4f[n-3], f[n-2]$ and $4f[n-1]$. Within a FPGA we have more gates than we have dedicated multipliers and we can perform addition in a pipe-lined way in order to speed-up a process. The downside is that the latency increases, because the propagation delay between registers adds up. The initial values for $f[n-4], 4f[n-3], f[n-2]$ and $4f[n-1]$ are calculated on a computer and are then stored in the FPGA within the bRAM (block Random Access Memory). New values can be sent over USB to the FPGA to update these values used in the implementation of the recurrence relation. The results for an arbitrary shaped test sequence to shape laser pulses is shown in Fig. 4.9. The setup used as shown in Fig. 4.10 has the EVIL with the recurrence relation of Eq. 4.13 implemented and is connected to the PI controller in order to feed back on the pulse shape. The output of the EVIL is connected to a mixer with a 80 MHz local oscillator and then amplified to drive the AOM. The AOM was setup in a double pass configuration for the laser light (397 nm). The photo-diode (PD) detects the laser light and output a voltage signal which is fed back to the EVIL. The data shown in Fig.4.9a had higher frequency components on which we believe was due to impedance mismatching. To see the effect of pulse shaping more clearly the data has been smoothed out data as shown in Fig. 4.9b and we can see that the measured laser light is indeed shaped as we expected.

We have also modified the code such that a predefined voltage sequence is output on the rising edge of a TTL and another voltage sequence on the falling

edge of a TTL. We have chosen to output a Gaussian voltage shaped from the EVILs over a period of $t_{switch} = 3\,\mu s$, see Fig. 4.12 for reference. The output was used to shape 729 nm laser light which drives the $S_{1/2} \leftrightarrow D_{5/2}$ quadrupole transition in calcium. The results we have obtained from a frequency scan of an unshaped and shaped pulses is shown in Fig. 4.11. From the data with an unshaped pulse which has a boxcar shape we can observe more off-resonant behaviour as with the a pulse that has a smooth Gaussian ramp at the beginning and the end of the pulse. More tests were performed by David Nadlinger [Nadlinger 13].

4.4 Bitstream

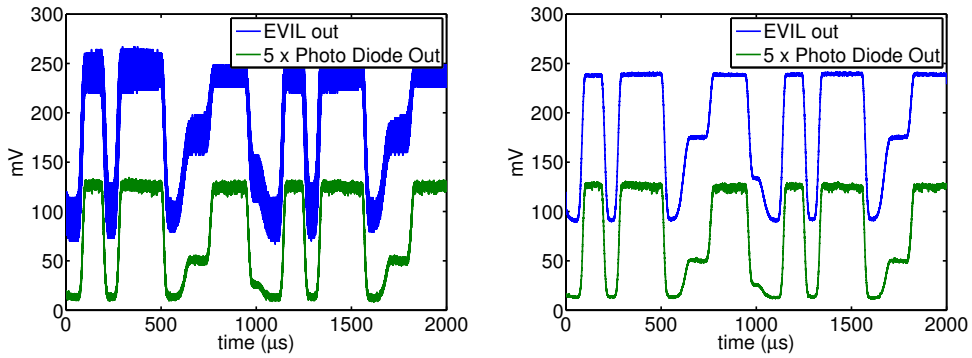
The code that runs on the FPGA is referred to as the bitstream. In Fig. 4.13 a basic overview is given of how the firmware works. The names in the figure are similar to that used in the code. We initially used a Microblaze processor to handle the communication. It was simple to use, but it took up a lot of the FPGA fabric. This is not true for all FPGA's, some have more space. Vlad Negnevitsky has replaced the Microblaze we have used with custom communication core (EVIL decoder). In the figure the USB port is represented by the TX (transmit) and RX lines and it allows for streaming of data to the PC and sending commands to the FPGA. We initially had a single PI controller which David Nadlinger extended to two independent PI-controllers, since we had the extra slow DSP channel. Each of the two channels calls on a concurrent implementation of the original digital PI controller for control and output. Each channel also has the capability of running a sweep also known as a ramp, a required feature in locking laser to cavities in search of the resonances.

The bitstream is relatively simple and has also allowed various students to do projects on and implement their own digital logic and test with the EVIL's. For various projects which has been done to improve or try new bitstreams and upgrades to the software for the EVIL project see the following projects:

- Alex Hungenberg, streaming data to the EVILs and low pass filter for auto-relocking [Hungenberg 13] as well as an EVIL server.
- David Nadlinger, pulse shaping and dual-pi controller implementation [Nadlinger 13].
- Christoph Fischer, lock-in amplifier attempt [Fischer 15a].

4.5 Software

The final software to control the EVILs were written in python in combination with PyQT. We have found that this way it is the fastest to make changes to



(a) Result from the EVIL out and the beam measured on a photo-diode.

(b) Smoothing data out.

Figure 4.9: Initial results obtained in shaping light pulses.

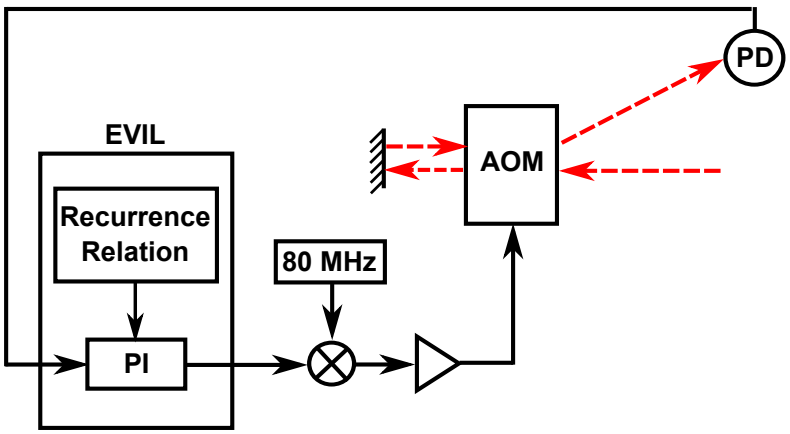


Figure 4.10: Setup to test pulse shaping in the laboratory. The output control signal from the EVIL is output and mixed with a 80 MHz signal amplified and relayed to an AOM. The AOM is in a retro-reflected setup in which a photo-diode is used to measure the signal. The signal acts as reference into the EVIL.

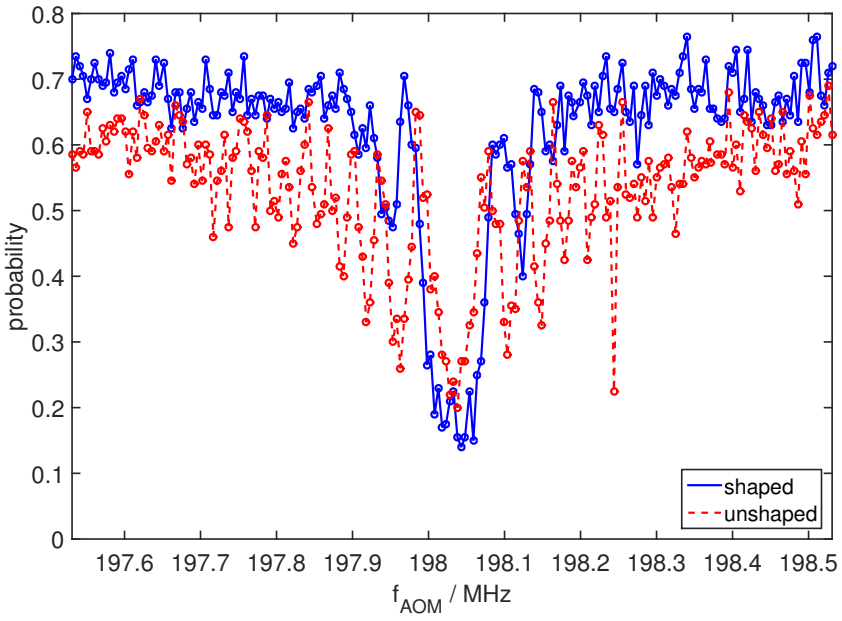


Figure 4.11: Spectrum of an unshaped and shaped 729 nm pulse on calcium.

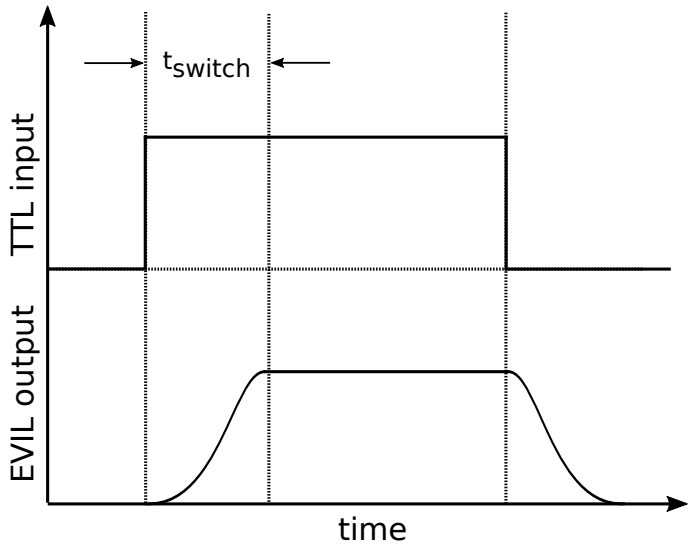


Figure 4.12: TTL input and output from the EVIL.

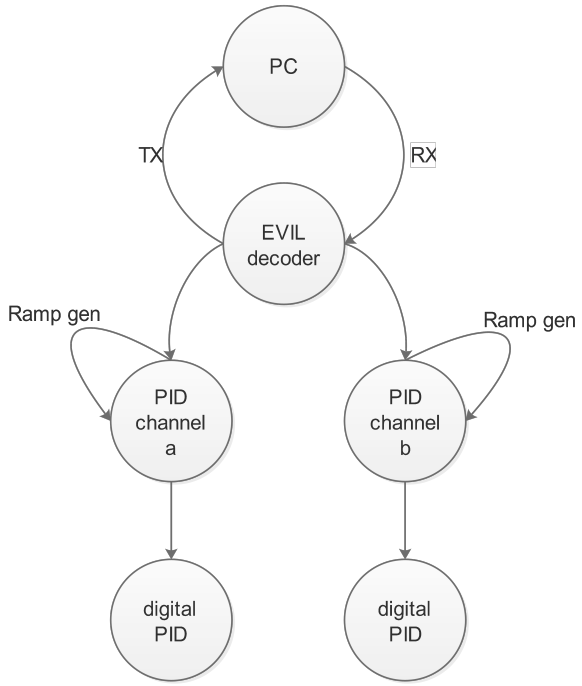


Figure 4.13: Diagram of the EVIL2 firmware connection.

existing code and have new users modify and add to the code without too much effort. PyQT allows for very quick and simple changes to the GUI (graphical user interface) and if no additional new functionality is added there is no need to update the code. This simple design has led to as few as approximately 500 lines of code to have our main control of the EVILs. A screenshot shown of the recent version of software being used in the lab is shown in Fig. 4.14.

Currently there are two independent PI controllers running in parallel on the EVILs. Each has their independent control parameters shown in the fast and the slow PI controller tabs. Typical operation of the PI controller starts with a sweep whereby the output voltage from the EVIL is scanned in a saw-tooth of which the amplitude and frequency can be set by adjusting the *Range* and *Frequency* setting. The *Center* setting allows to adjust the DC-offset of the sweep signal. When running the sweep the user will typically check the error signal being streamed to the GUI.

In stopping the sweep the control is automatically started by using the middle value of the sweep as a starting output to initiate control. By adjusting the control parameters P and I locking can be achieved. Locking here refers to having a stable feedback actively being acted on by the controller. Once the locking has been achieved automatic re-locking can be activated. Acting as a low pass filter the EVIL detects when there is a drift in the output within the parameters set by the user and will output the previous used starting value

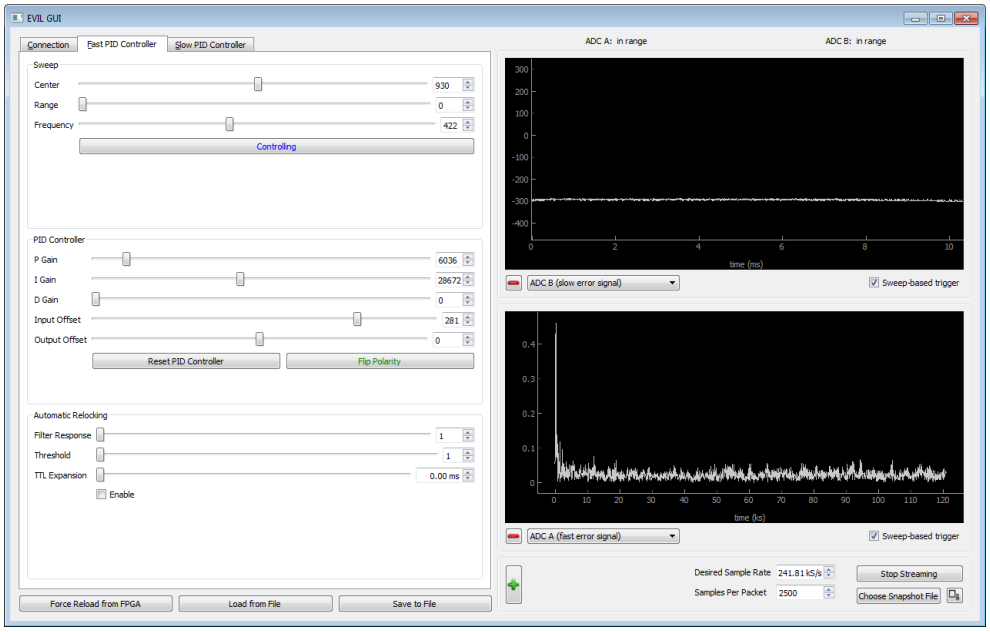


Figure 4.14: EVIL control software as used in our laboratory.

of the sweep to re-initialize locking.

4.6 Future improvements and projects

A list of some brief comments on possible improvements and future projects for the EVIL are

- Design with FPGA on the custom board.
- Using a FPGA with a few more pin-outs to allow the replacements of the slow channel with another fast channel.
- It should be possible to combine the KILL (Keitch Intergrate Locked Loop) board with the EVIL.
- With a new FPGA, move away from using USB to Ethernet.
- Unique identification in the software to allow to recognize which EVIL is which. Ethernet should easily solve this problem
- See chapter 5 for the DEATHs, it would be worth replacing one DAC of the DEATH swith two ADCs and have an upgraded EVIL based on the DEATH design.

Potential future projects

- Transfer function measurement from GUI of device under control and or other units and then export the data. This can then also be used as a

simple function generator for testing a piece of equipment when there is a spare channel. Typically there is not always a spare signal generator, so this would speed-up testing when someone can just go to the nearest EVIL with an unused channel and use the signal generator feature.

- Connecting up LEDs to the frontpanel that indicates whether a controller is out of lock with an indication for which one of the channel. This could also used to indicate overflow and underflow of the signal going to the ADCs. This could help the initial setup of an EVIL. In general there could be various uses.
- Connecting up switches to the frontpanel and make that the bitstream allow for re-locking when pressed. Another example could be toggle between "sweeping mode" and "locking". Since this is what is typically done by people in the laboratory when they re-lock. The coefficients are not typically changed. This will allow for re-locking without using a laptop or PC.

Chapter 5

Voltage sources

Within segmented ion-traps several independent voltages have to be applied to the various electrodes in order to confine ions. In the case of transport these voltages have to vary over time. The collection of all voltages needed to be output by each electrode at each time step in order to perform a transport operation on an ion is referred to as a waveform.

This chapter contains discussions on two hardware solutions which we have that can output waveforms. The first is based on the Analog AD5371 development kit which is slow, but offers a large number of output voltage channels. It is a commercial device and only relevant information to the typical used will be covered. The second is a custom built solution, DEATH (Direct Ethernet Adjustable Transport Hardware). They offer much faster concurrent outputs compared to the AD5371 and they are used in all transporting experiments reported on in this thesis. It will be covered in detail due to it being a custom solution we have developed in our group.

5.1 Slow multi-channel device - AD5371

The Analog devices AD5371[1] provides 40 channels of output voltages with 14-bit resolution and is controlled over SPI (Serial Peripheral Interface) in our experiments. The maximum output voltage span that can be achieved is 20V, limited by the on chip amplifiers. The output voltage range can be varied slightly and can be set between $V_{\mathrm{SS}} + 1.4\,\mathrm{V}$ and $V_{\mathrm{DD}} - 1.4\,\mathrm{V}$. V_{DD} can be set between the range -16.4 V to -4.5 V and V_{DD} from 9 V to 16.5 V without damaging the chips. In order to have a voltage change we need to consider how fast we can transfer data to the device. For the AD5371 this can be done in two ways; either we update a single channel that requires less data to be transferred or we update all the channels simultaneously by first passing all

[1] 40-Channel, 14-Bit, Serial Input, Voltage Output DAC : AD5371 from Analog devices.

the data to the various registers associated with each and then finally give it the update signal ($\overline{LDAC}$) to output the voltages to the electrodes. Both of these methods have advantages and disadvantages. Updating a single channel fast increases the rate at which you can update any given voltage, but does not allow for simultaneous updating of all the channels. The subsequent changes to the potential could complicate transport schemes in which a well controlled transport routine is required. Updating all the channels at the same time decreases the overall rate at which you can change the channels and would make it harder to do fast transport. Given the maximum SPI clock rate of 50 MHz that is achievable and a SPI word length of 24 bits, in order to set a defined voltage to a channel, either all voltages can be updated simultaneously at 52 kHz or an individual channel can be updated at 2.1 MHz.

The communication to these devices has been implemented by Ben Keitch and I have helped debug and assemble the devices. As shown in the diagram in Fig. 5.1 the master FPGA (Field-Programmable Gate Array) directly controls the AD5371 via an intermediate solution of buffer boards. These boards, designed and built by Ben Keitch are necessary, since SPI is not made for long range communication which is required. They essentially buffer the incoming SPI signal from the FPGA and then transmit it over Ethernet cable (RJ45). This is done such that SPI signal carrying the commands to the AD5371 is not degraded beyond the point where they can not drive logical 1's and therefore not be interpreted. On the receiving side these signals are routed to the AD5371 via a ribbon cable. Within the laboratory we do not run the communications at the maximum rate, since there was a worry about long range SPI transmission. The AD5371 is placed on the optics table and connected with approximately 50 cm of ribbon cable to a filter board box. The filter board box houses custom filters with a cut-off frequency of 250 kHz. The filters are in turn connected to the trap via D-sub connectors. It should be noted that the cut-off frequency can be changed by replacing the external filters. For the experiments performed and described within this thesis, filter boards approximating a 2nd order Butterworth filter with cut-off frequency of 250 kHz were used.

Having worked with this solution and the EVILs in our experiments we have come to the following conclusions;

- If possible avoid SPI over long distances. Signals degrade and you are severely limited in your throughput of data since update rates are slow and data is sent bit by bit.
- Avoid USB communication, for it also is prone to failure over larger than typically 5 m distances. It also assigns a COM (communication port) port in Windows which might change after reboot and can lead to confusion.
- When updating voltages to electrodes it would be preferable to update

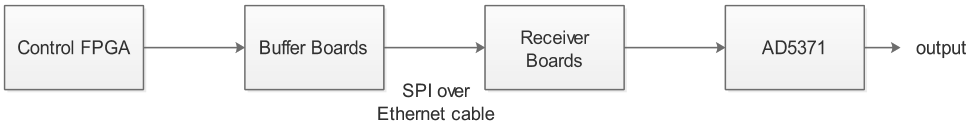


Figure 5.1: Diagram representation of how the AD5371 is controlled in our setup.

synchronously, for you could potentially excite the ion in an unwanted way when the electrodes you are updating are close to the ion. Synchronous updates will allow for well behaved transport.

- If it all possible avoid building a setup where the units that produce the voltages have to rely on multiple different hardware solutions to all work together. This has been the cause of many hours spent in debugging and redoing many designs.
- When possible, simulate circuits thoroughly before having boards built, this is the fastest and best way to avoid long hours in debugging a physical device.

5.2 Other Technologies

Groups at NIST [Bowler 13], Siegen [Baig 13], Ulm [Poschinger 14] that do transport of ions have high speed AWG (Arbitrary Waveform Generation) hardware solutions. Some of the key features of these various technologies as well as that which we had developed in our group are shown in table 5.1. The possibility existed to create collaborations with either of these groups in order to quickly obtain a solution which we could use for transport. At the time there were various reasons of which a critical one has been we would like to develop our own critical technology such that a full understanding of the system within our group exist in order to fix problems when and if they occur. Another key point was based on the experiences with electronics in our lab which led us to want to avoid USB based communications.

The solution used by NIST [Bowler 13] has two different boards, a master and a slave board (waveform generation board). Each waveform generation board produces three voltage output channels ($\pm$10V) with 16-bit resolution. Up to two slaves can be connected to a master board, via board interconnects, resulting in a total of 9 channels. The master board is connected to the computer via USB and sends the data on to the slaves. There is no external clocking solution, thus only up to 9 voltages can be made to run synchronously. A TTL (Transistor-Transistor Logic) input to all the devices is used to trigger the start of a waveform in order to synchronize the outputs. We wanted to stay away from USB based communications and have the option to run from

Group	Comms.	Update rate	Output rate	Noise
NIST	USB	50 MHz	50 MHz	$120\,\mathrm{nV}/\sqrt{\mathrm{Hz}}$
Siegen	USB	25 MHz	25 MHz	$250\,\mu\mathrm{V_{RMS}}$ (100kHz)
Ulm	Ethernet	2.5 MHz	2.5 MHz	0.3 quanta/ms
ETH	Ethernet	400 MHz	50 MHz	$63\,\mathrm{nV}/\sqrt{\mathrm{Hz}}$

Table 5.1: Various high speed waveform generation hardware by ion-trapping groups with some key properties shown.

an external clock when necessary. We would also like to synchronously update multiple channels and including TTL based trigger is a strong requirement.

The group at Siegen has developed a 19-inch rack solution with careful thought being put into the design [Baig 13]. Within this solution there is a main control board with a FPGA that controls up to 12 separate daughter boards (waveform generation boards) via a back-plane solution. These are all mounted in the same 19-inch rack. Each waveform generation board produces two voltage channel outputs (±9V) with 16-bit resolution. The control board (one with FPGA) receives commands via USB and based on these command controls the slaves in a specific way. Each waveform generation board can store waveform information data up to a depth of 128 kB. These boards can be run from an external clock to have them operate in a synchronous way. The control board also provides an output TTL to trigger other parts in the experiments. We wanted to again avoid communications via USB and in this case three boards in total are required, a master board housing the FPGA to control slave boards via a backplane board. We wanted a simple solution that can be modular and does not rely on a backplane type of solution. This would allow for placing the DACs (digital-to-analog converter) closer to the trap as well as reduce the complexity of integrating with our control system.

In Ulm a similar solution as in Siegen is used. A Virtex-V FPGA main control board interfaces with 4 separate daughter boards. Each daughter board has 3 serial quad DAC8814 16-bit DACs and thus deliver 12 individually controlled voltages [Poschinger 14]. They have a maximum update rate of 2.5 MHz when a single voltage is updated and the FPGA produces the clock for the daughter boards.

A summary of the technologies used by varies groups doing ion transport is displayed in Table 5.1 as well as our own custom solution. Regarding the noise measurements of Siegen, the $250\,\mu\mathrm{V_{RMS}}$ (100kHz) was measured over 1-10^5 kHz [Baig 13]. For the DEATHs the average noise floor within the typical motional frequency range of $2\pi \times (1-3)$ MHz is shown. The noise measurements made by the group of Ulm is shown in the table as a heating rate [Poschinger 14].

5.3 Considerations and requirements

From our own experiences with electronics in our experiments, mainly from the EVIL project as well as discussions with people from groups with the technologies listed in Table. 5.1, have led us to a summarized list of properties we would want to achieve with our own custom solution in Table 5.2.

The requirement that communication should be over Ethernet was a rather strong design requirement. In our experience we found some annoyances when working with equipment that work over USB in our laboratory. Adding multiple devices (in this case EVILs) to a single computer became a challenge to identify which ones are which and we could only get to identify them via the COM port itself. Restarting a computer would also sometimes lead to different COM ports being assigned to various EVILs and have been the cause of confusion on several occasions. With the EVILs we also had some initial issues with virtual COM ports not working or not going away which could only be fix manually or by re-installing the drivers. Communicating over long distances (greater than 5 m) became a problem. The signal carried by USB would sometimes seem to be attenuated over these distances and we would not be able to establish communication with our devices. The attenuation might not be the main culprit in the failure, in the USB specification [Compaq 00] a maximum cable delay of 26 ns is allowed. Assuming a typical velocity factor of around 0.7 results in a cable length of approximately 5.5 m. We had to (and still do) use USB repeaters (powered USB hubs) to strengthen the signals. With Ethernet we know we can go other longer distances without a problem and connecting multiple devices requires one only to know the IP-addresses of each device.

Regarding the data-update rate, noise and output voltages we wanted to have comparable specifications to other groups and for our 3D-trap the voltage range requirement is suitable, based on our experience with the AD5371. The option of updating the voltages synchronously would be preferred. The reasoning behind this is that if there are more channels added to the trap from other waveform generation cards (DEATHs) and they are not synchronous, the relative phase jitter between the various voltages might cause some unwanted motional excitation. Here I had no experience, but we decided to err on the side of caution and would rather over engineer and avoid potential pitfalls.

Another strong requirement is to be able to store a large number of waveforms, thus a large amount of data on the individual devices. This should enable quick selection of various waveforms and potentially allow for branching. Where branching of a waveform implies a selection of the next executed waveform based on measurements returned from the ion. NIST's solution can branch in 40 ns [Bowler 13] via TTL, and a simmilar solution is desired. A storage amount similar to that of the Siegen group's solution would suffice, anything more would be a bonus. We did not have a good estimate for this

requirement. It depends on the amount of electrodes to be controlled, the number of voltages to define a given waveform and the total number of waveforms required in an experiment. We choose to over engineer the system and avoid future experimental difficulties due to custom electronics. It should be noted that the synchronous update of multiple voltages sources and therefore a smooth change of the potential seen by the ions was the main goal of the design.

A TTL output from the boards would be an extra feature we might use in the future for other possible applications, like triggering other experiments, but this is not a strong design requirement, since we could circumvent this using the control FPGA in our experiments. A TTL input on the input side is a strong requirement, since we planned to use a waveform generation card that can be triggered to start outputting its voltages based on this TTL input. Having a TTL input allows the triggering of multiple devices to start outputting data on the same time. The TTL-input with the Ethernet communication requirement allows us to have a modular design that does not have to be in the same 19-inch rack. Having the option to move the device simply to another 19-inch rack, would be preferable in case we have some noise-related problems based on being too close to other devices. These requirements exclude the solutions of having board interconnects or a backplane, like the solutions used by NIST, Siegen and ULM.

Another requirement is to have the device be compatible with other Eurocard sized solutions in our lab, regarding supplying of the power to the device and their pin out. This was unfortunately not possible. We use a male DIN 41612 connector to power our Eurocard sized devices, and the **DEATH pinouts are different**. Thus it is important to note that for the DEATHs the voltage pinout on the connector is different to that of the EVILs or KILLs. For the DEATHs {A1-6,B1-6,C1-6} is connected to -15 V, {A9-12,B9-12,C9-12} is connected to GND, {A13-24,B13-24,C13-24} to +5 V and {A26-32,B26-32,C26-32} to +15 V. See the DEATH schematics in Appendix D for more clarification.

Having a single modular design would also be preferable. Relying on too many components to work well together typically creates problems. We have good experience with using a market based FPGA solution with connectors for which we then design a daughter-board. One of the most heavily used electronic devices in our lab, the EVIL (4) was the predecessor of sorts to the DEATH boards using this type of design. As has been discussed in chapter 4, we could not upgrade the EVILs to be a voltage source for transport. Two DACs are used that run at different rates and improving the design would have required a complete redesign since we have designed it for a FPGA which would not be suitable for the amount of data we would like to store on the chip.

Property	Requirement	Achieved
PC-communication	Ethernet	Yes
Data update rate	25-50 MHz	Yes, 50 MHz
Noise	120 nV/$\sqrt{\text{Hz}}$	Yes, 63 nV/$\sqrt{\text{Hz}}$
Output voltages	± 9-10 V	Yes
TTL input	Yes	Yes
TTL output	Maybe	Yes (untested)
Ext. Clocking	Yes	Yes
Waveform Depth	128 KB	Yes
Single [1]PCB design	Preferably	No, custom+commercial
[2]PSU Compatibility	Lab compatible	No

Table 5.2: Preferences and design goals. [1]Printed Circuit Board. [2]Power Supply Unit.

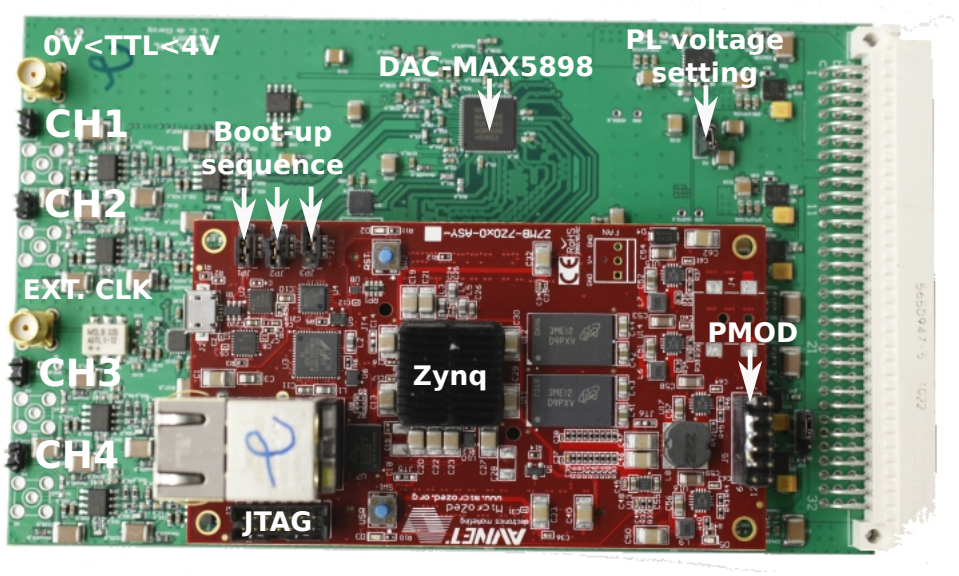


Figure 5.2: The DEATH viewed from atop with the MicroZed attached. The second DAC is located under the MicroZed board.

5.4 Hardware design - DEATH

In the custom design of the hardware there are various choices to be made of the physical components that will populate the board. In Fig. 5.2 a top view is shown of the custom designed board for the DEATH. The number of output channels are shown as well as the locations for the PMOD (Peripheral Module) jumper which are used to give an unique ID via an Ethernet address. The JTAG used to re-program the FPGA as well as the TTL input to sync and

the EXT clock input are indicated which would be of use to a more advanced user of the board.

As a simple rule for all the resistors we have used on the board we have made use of 0603 (size) Philips resistors with accuracy better than 5 %. The brand is not as important as that all of the resistors come from the same company. Different companies have slight changes in the physical sizes of their resistors even though there are standard sizes they should adhere to. For capacitors we tried to get the cheapest ones, for they are only used as bypass capacitors close to chips and filtering. It should be noted that we have used a rather large number of capacitors for a design like this, but we were trying to avoid having on-board chips pulling their current from the main supply directly. The latter would likely result in increased noise, because the frequency at which different chips pull their power from the source will couple to other chips as well. To prevent this to a higher degree we have added ferrite beads close to each active element which is a passive component that blocks high frequency noise in both directions of current on the power lines with additional capacitors to suppress the higher frequency noise.

Figure 5.3 gives a brief overview of how the DEATHs fit into our system and the functions associated with each electronics board. The DEATH consists of a custom build board and a commercial FPGA (MicroZed). Control over the DEATHs is executed from a control PC interfacing with the FPGA via Ethernet. The custom board provides power to the FPGA and there are shared digital lines between the two in order to use the DSP (digital signal processing) components on the custom board. The rest of this section will have a brief overview and discussion of how the main signal processing components were chosen.

5.4.1 DACs

A wide variety of DACs are available on the market. My primary consideration was to find a DAC offering a high update rate, low noise and which is able to output constant voltages (high-speed DACs can not necessarily do this). We chose to use the MAX5898[2] by Maxim as the DAC for the design. This dual channel DAC can take a maximum clock frequency of 500 MHz. For the implemented design 400 MHz is used due to timing difficulties which arose when trying to clock it at the maximum rate. When considering the spectrum of the outputted signal from the DAC, images of the signal will form at multiples of the input data rate due to sampling. Having a different output rate from the DAC, allows the use of the interpolation feature of the device of which the DAC is capable of performing 1x, 2x, 4x and 8x interpolation on the output data. What happens is that when a DAC is required to output at

[2] 16-Bit, 500Msps, Interpolating and Modulating Dual DAC with Interleaved LVDS Inputs : MAX5898, from Maxim.

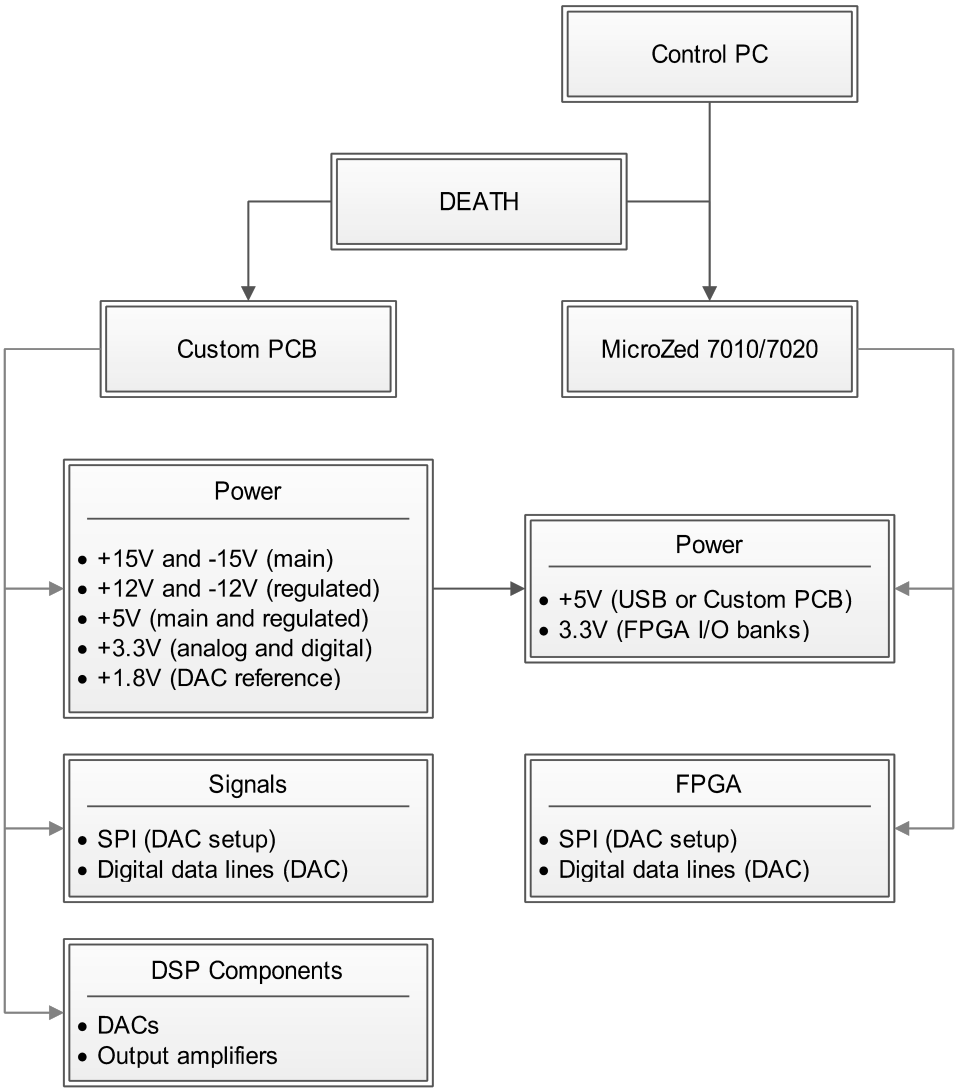


Figure 5.3: Overview of the DEATH and how it is controlled from a PC.

a faster rate than it receives data it can upsample the incoming data and then apply a low-pass filter to it in order to operate under these conditions. This is known as an interpolation filter. The higher the interpolation number in the setting the more interpolating filters are applied. Using the interpolating setting then result in attenuation of the spectral images at the various input data rate frequency components. Within the design we have chosen to use 8x interpolation, which then results in an effective update rate of 50 MHz and attenuation of all images from 1-7 times the input data rate. The DACs used are indicated by 2 in Fig. 5.5. In hindsight we would have chosen a DAC with lower update rate and higher output voltage if possible to avoid amplification on the final stage minimizing amplification of drifts and pre-amplifier noise. This could have made the output more stable over time with respect to voltage drifts as well as noise levels. Both of these will be discussed in the section covering the DEATHs performance.

5.4.2 Amplifiers

The AD8021 is a high speed voltage feedback amplifier that we have used in the design of the EVILs. At the time of design we have looked into various other possible solutions and have found only a few that came on par with the performance with respect to noise and gain bandwidth product offered by this device. Due to the high speed nature and design of the chip special layout considerations had to be taken. It is suggested that the output signal form a guard ring around the output pin of the device in order to shield the compensation pin from local noise while reducing stray capacitance[3]. Thus the risk is reduced for an unstable circuit when designing the final output stage. This final stage is indicated by section 3 of Fig. 5.5. Unlike the EVIL we have decided to make use of a cascaded amplifier circuit. This reduces the gain required to be achieved with one single amplifier. This in turn should lead to faster output frequencies of the device due to using lower gains. Between the two amplifiers we have added a 4-th order Butterworth filter with a cut-off frequency at 50 MHz. This has been done to avoid possible digital noise at the various clock rates being used on the board coupling to the output. The final filtering of the signal to the electrodes is performed near the trap. There we have a filter approximating a 2nd order Butterworth filter with cut-off frequency at 250 kHz. The filter can be easily changed if needed for higher transport speeds, but is kept low to reduce possible noise at higher frequencies which might couple in from other laboratory devices.

5.4.3 FPGA

Based on the experiences we had with the EVIL project, we wanted to use a commercial FPGA again and design a custom extension board. Having

[3] see AD8021 datasheet, available online: http://www.analog.com/

searched a rather long time for a suitable FPGA, we decided on the MicroZed 7010[4]. The FPGA on board is a Xilinx Zynq 7010 chip, which houses two ARM core A9 processors and programmable logic. There is 240KB of bRAM (block Random Access Memory) on the chip which allows to store over 16 waveforms with each having around a 1000 steps for four independent channels which requires $4 \times 2\text{Bytes} \times 2^{14}\text{Addresses} = 128\,\text{KB}$ of bRAM. The RAM to structure the sequence takes up 256 B of bRAM. The bRAM is the RAM that is physically built into the chip itself and if this would be a limitation there is 1GB of RAM on the board which would allow for storage of up to 2^{27} voltages ($\frac{1024^3}{2\times4}$). At that time the structure RAM will have to be modified slightly to allow for the much greater number of waveforms, for it allows currently only 2^{17} voltages to be stored per channel of a DEATH unit.

The custom board acts as a daughter-board which supplies power to the MicroZed board and acts as the carrier for the signals from the FPGA to the DACs. The ARM processors interact via an AXI bus with the PL (Programmable Logic) and therefore allows for complex design architectures where the processor could handle communications with a computer which is not as timing critical as the control over the DACs. The latter can be handled by the PL. The MicroZed board plugs into the headers indicated by 8 on Fig. 5.5.

5.4.4 Power

The main power to the DEATHs is provided by commercial linear power supplies that provides $\pm15\,\text{V}$ and $5\,\text{V}$. These are situated in the back of the DEATH box as shown in Fig. 5.4. Also shown in this figure are four installed DC-powered fans for cooling the DEATH boards that are situated in the front of the DEATH box. The $\pm15\,\text{V}$ is used to drive voltage regulators to produce clean and steady $\pm12\,\text{V}$ sources that in turn supply the amplifiers on the board. The $+5\,\text{V}$ drives various voltage regulators to produce $+3.3\,\text{V}$, $+2.5\,\text{V}$, $+5\,\text{V}$. The regulators we have used are: LP38789,TPS7A4700 and TPS7A3301. These regulators we have found quite useful, for they can supply a wide range of constant voltages by intelligent choice of feedback resistors. They are also some of the lowest noise linear amplifiers we could find with relatively high output power. The first has been used to provide all the positive supplies lower than $+5\,\text{V}$ and the last two have been use to regulate to $+12\,\text{V}$. We have decided to keep voltage regulators as close as possible to the where the power is supplied to the board to avoid possible noise pick-up from digital sources. The section where they are located are shown by section 1 in Fig. 5.5. Section 7 of the same figure indicates the circuit that controls the timings of the voltages that are supplied to the MicroZed board as required per datasheet. An accurate voltage reference, MAX6161, suggested within

[4]MicroZed development kit: http://zedboard.org/product/microzed

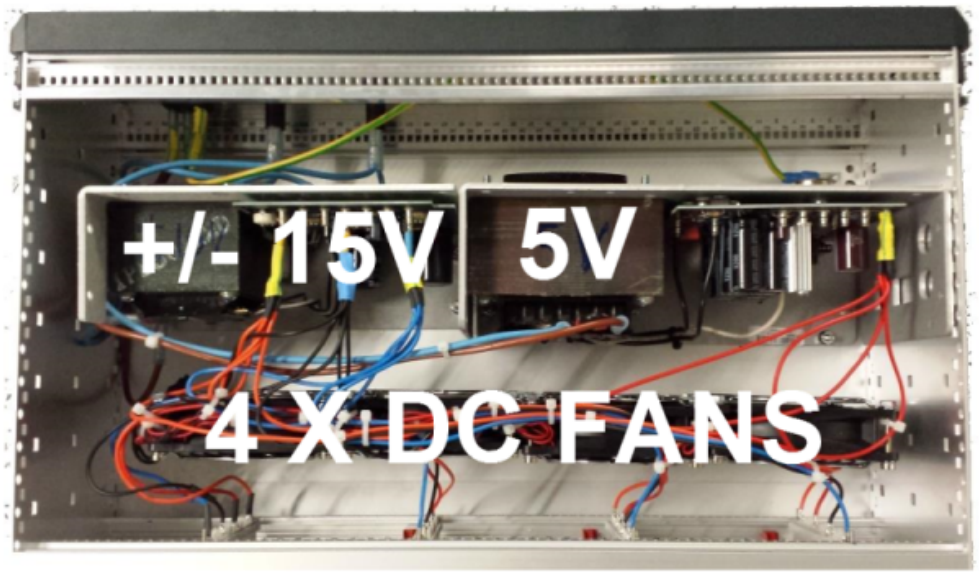


Figure 5.4: Open DEATH box with the power supplies and DC-fans shown.

the MAX5898 datasheet have been used to supply the 1.8 V reference to these DACs.

5.4.5 External Clocking

It is possible to clock the DEATHs from an external clock source in order to sync various units together with the TTL input. Section 4 in Fig. 5.5 indicates where components can be added to allow for external clocking. Whether or not this feature is being used, a setting has to be in place in section 5 (shown in Fig. 5.5) to choose whether the clock source should be external or obtained locally from the FPGA. For the setup used in all my experiments external clocking was not used. The box containing four DEATHs delivers 16 voltages that were(are) sufficient for the experiments we have performed.

5.4.6 PCB

In order to keep the layout as simple as possible and potentially lower the risk of failure the use of a commercial board was used to provide the FPGA the MicroZed 7010 that plugs into the board that have been designed. The MicroZed should be able to plug into the custom board via specialized connectors that allow for differential signaling. Power to the MicroZed should be provided and care should be taken regarding the sequence in which power has to be supplied. Having the custom board supply power to the MicroZed and to the various on-board components and carrying high speed signals (∼500 MHz), makes laying out a board like this fairly complex.

The board is a four layer board, two layers for signals, one for ground and one for power. The top layer is a signal layer followed by the ground plane, then the power plane and the bottom layer again a signal layer. This order is the default order in which four layer boards are layed out, also known as the standard four layer stackup.

Having read many sources to decide on whether to section the ground plane (layer) or not for the analog and digital signals we have decided to keep it a continuous plane. Cut outs from the ground plane were made only under the amplifiers to reduce capacitive coupling to the ground plane as is typical in high-speed designs. This makes some of the layout simpler w.r.t. grounding. The power plane was sectioned such that as far as possible all power sections within the plane could be as big as allowed with no thin connections.

Routing the signals that should carry frequencies up to 500 MHz, requires that tracks that carry information to e.g. the DACs should be length matched to avoid potential relative phase delays and result in signal to be pushed at much slower rates down these lines. The DAC chosen has differential signaling for all but the SPI and power lines to the chip. Thus noise common to both lines of the differential lines should not result in the DACs output to be noisy. The DACs themselves only output $\pm 0.5\,\mathrm{V}$ and therefore amplification is required to achieve the desired output voltage range of $\pm 10\,\mathrm{V}$. Here we have made use of a dual stage amplification circuit which is impedance matched with the DAC to avoid any reflection and possible instabilities that might occur [Zumbahlen 07].

All other signal carrying lines on the board have been length matched in order to run the components at maximum frequency and where possible with differential lines. When we had to pick between a chip that has or does not have differential signaling, we always picked the former, because it should be more resilient to noise.

A DEATH which is the custom board together with the MicroZed finally slots into a box from the front and a panel with the various connections closes it up. From Fig. 5.6 the four output channels: CH1, CH2, CH3 and CH4 can be seen. The TTL input connection and the external clocking input are shown as well as the Ethernet connection to the board.

5.5 Digital design - Bit stream

5.5.1 Overview

In order to have the hardware produce waveforms digital logic has to be programmed into the FPGA. Each part of the digital design that has a specific task will be referred to as a core. The bit stream is the digital control implemented in the PL (Programmable Logic) part of the Zynq7010 Xilinx chip.

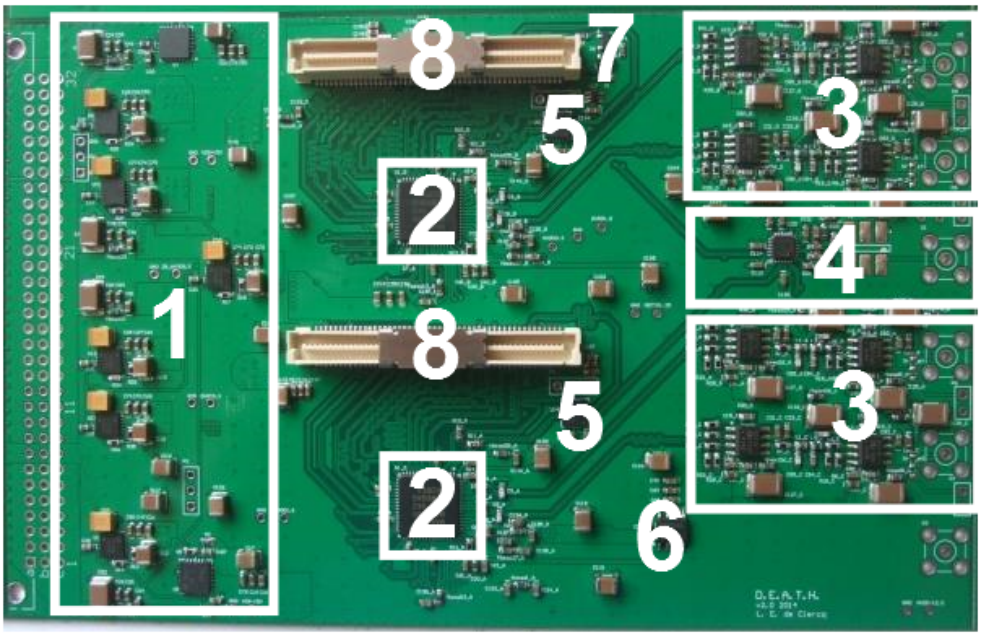


Figure 5.5: The various sections of the custom PCB in the DEATH design.

Some of the key parts of the control which are contained within the bit stream are

1. Setting up the processing system and accessing the hard-wired GPIOs (General Purpose Input/Output) to determine the Ethernet settings.
2. The transport core that takes care of the storage, recalling and outputting of waveforms to the DACs.
3. A bridge core which allows to direct the control over the voltages to either come from the transport core or the processor system (PS, the CPU on chip).
4. The MAX5898 parallel driver, a core that interleaves the data going to the two channels of DAC chip.

.

5.5.2 Transport core

Within this core is custom HDL code that implements two FSM(Finite State Machines) that control the receiving of data from the ARM processor sequencing the output of voltages to be used by the DACs.

All incoming data is structured in such a way that the voltages for each of the four voltage outputs are stored each in its own RAM by a FSM (RAM0 to

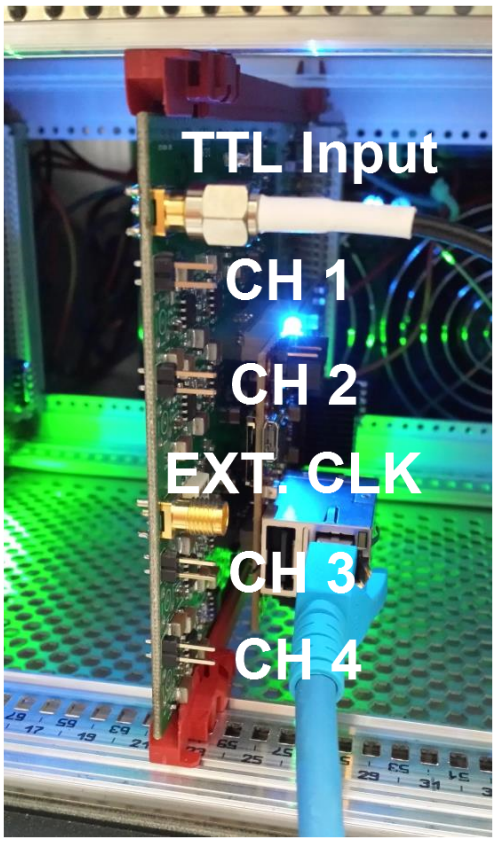


Figure 5.6: View of a single DEATH in the DEATH box with front panel and most of the connection removed.

RAM0) as illustrated in Fig. 5.7. Special instructions for how to access these stored values are written to the RAM as well and is referred to as *Sequence RAM* in Fig. 5.7.

The *Sequence RAM* is built up out of one experimental 16 bit word and the rest sequence words. The experimental word contains only the last address within the *Sequence RAM* that will be accessed by a FSM. The FSM work its way through the sequences and keep track of the progress of execution. After running all desired sequences it will reset to the initial address associated with the first sequence. In this context a sequence is a function that acts on a waveform. A sequence word contains 48 bits and is built up out of 3×16 bit words, such that it fits into the RAM structure as shown in Fig. 5.7. In a sequence you can specify which waveform to work with, how many elements that waveform has whether to run the waveform in reverse and the total number of times it should be output from the DEATHs. One might question why the sequence contains the number of elements within the waveform it should execute. The reason we have done it this way is to allow for variable length of

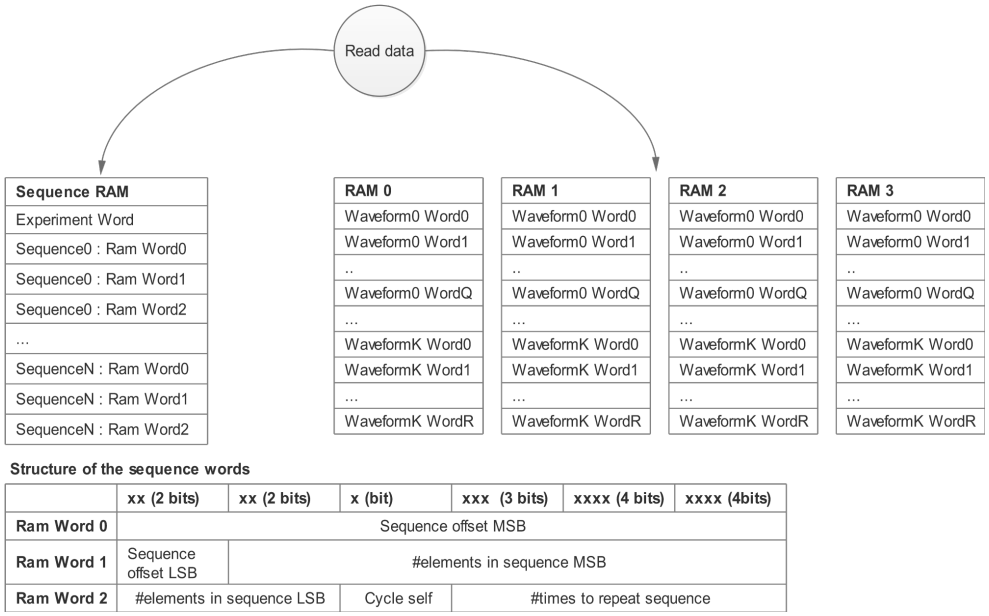


Figure 5.7: The FSM that reads data into the various RAMs and how the structure of the these RAMs look like.

waveforms. You could have a waveform that contains 1 set of voltages which will represent a constant setting you would like to work as well as a waveform that has thousands of voltage steps used in transport. The *Sequence RAM* is filled with sequences that should be run in succession and the experimental word to indicate to which sequence the FSM should execute to. This structure allows for the build up of very complex transport sequences. A TTL is send to all the DEATHs and on each trigger of the TTL one waveform will be output.

For the more advance user the following might be of interest. From the table in Fig. 5.7 the structure of a 48 bit sequence word is built up out of the three parts as stated previously. The first 18 bits (locations 0:17) give the location in RAM where the start of a waveform is located. For each of the RAMs (0,1,2 and 3) the location is the same, since they each have to output a DAC word that converts into a voltage and is sent to its connected electrode in the end. The next 18 bits (locations 18:35) contains the number of elements that build of the waveform the starting at the address specified. This is how the FSM as shown in Fig. 5.8 knows when the waveform has finished and what to do next. Bit number 36 when high indicates to the FSM that once the waveform has been executed it has to execute it directly afterward once in reverse. The final 11 bits (locations 37:48) contains the number of times the FSM should perform the procedure of outputting the waveform including to do it in reverse when set.

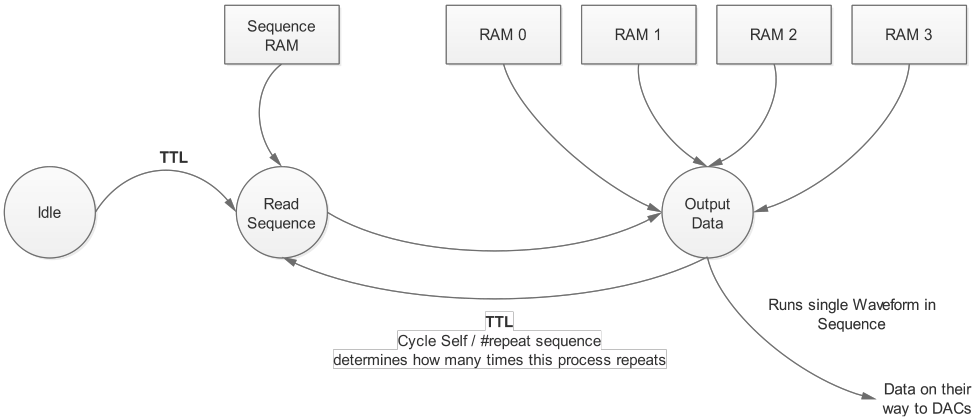


Figure 5.8: The FSM that reads data from the RAMs and then outputs.

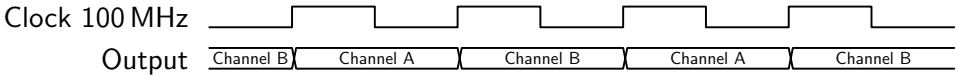


Figure 5.9: The interleaving of the data to the two channels of the DAC w.r.t. the input clock.

5.5.3 Bridge core

This core allows you to set via the processing system (PS, CPU on chip) whether the voltages to be output to the DACs should come from the transport core or from the processing core.

5.5.4 DAC controller core

The MAX5898 requires data to be sent to it in an interleaved fashion, since there are only 16 data lines and it is a dual channel 16-bit DAC. This has been achieved by using a custom core, the "max5898_parallel_driver", which takes two inputs and produces a single output of interleaved data. A simple timing diagram is shown in Fig. 5.9. As can be seen from the figure the interleaved data changes at half the clock frequency to the core.

The implemented timings for experimentally functioning design is shown in Fig. 5.10. The phase difference between the data clock and the change of data has been exaggerated in the timing diagram, the time difference is 0.56 ns. It should be noted that the frequencies should be phase locked otherwise the channel onto which the data is send on the DAC might change order. This sounds simple, but we ran into this problem thinking that we have phase locked the frequencies, but this was not the case. This is what have led to the value for the phase difference found as well as a proper implementation to avoid this with clocking cores. The values displayed for the timing indicates

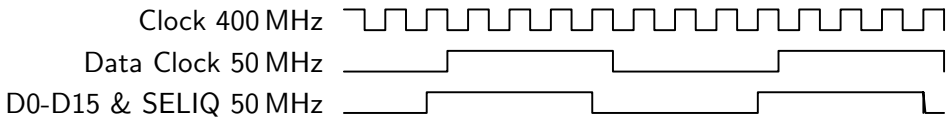


Figure 5.10: Implemented timing diagram for data transfer of the MAX5898.

Interpolation Rate	f_{DATA}:f_{CLK}	f_{DAC}:f_{CLK}
1×	1:1	1:2
2×	1:1	1:1
4×	1:2	1:1
8×	1:4	1:1

Table 5.3: Preferences and design goals.

the use of the 8× interpolation which we intended for the design. The other possible settings for interpolation is listed in Table 5.3.

5.6 Software design - Firmware

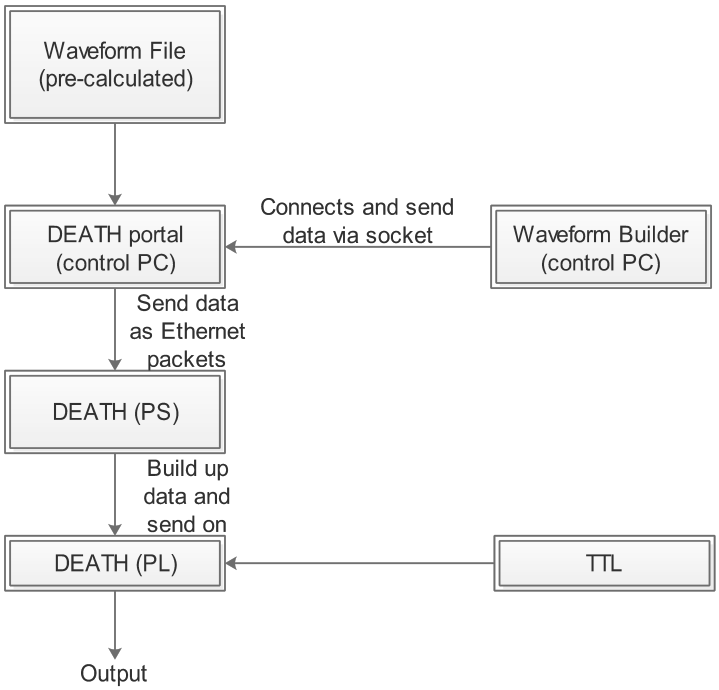
In the previous section we have discussed the implemented custom cores that run on the programmable logic (PL) of the Zynq FPGA. This section gives a brief overview of the tasks of one of the CPU's on the FPGA. The software that runs on the CPU will be referred to as firmware, not to be confused with the logic that has been created for the PL, namely the bit-stream.

5.6.1 On startup

When the DEATHs are powered on for the first time the CPU carries out a routine to setup the DACs in the correct mode. It sends SPI signals via the EMIO (Extended Multiple Input-Output) to the PL. First the DACs are reset and then SPI commands follow to allow for the DAC to run on 8x interpolation mode. This is all done by setting up some registers within the DACs, which can be read back and compared to what was sent. When they agree the DAC is in the correct state and working. One thing to note about first design of the boards, is that we forgot to add level shifters on the SPI lines and this sometimes requires the commands to be sent more than once to have them operate correctly. In the CPU the problem was corrected by sending the command a few times and reading back the registers.

5.6.2 Communications

Communication to a control PC on the DEATHS are handled by the processor without making use of an operating system. A bare metal (no operating system) PS (Processing System) with a lwIP (Lightweight TCP/IP stack)



Data build up from Ethernet packets

Trigger Word	Ram Width	Data Width	RAM Addr 0	RAM Data 0	RAM Addr 1	RAM Data 1	...	RAM Addr N	RAM Data N
2bits	8 bits	8 bits	32 bits	32 bits	32 bits	32 bits	...	32 bits	32 bits

Figure 5.11: Software overview and connectivity. Additionally the build-up of the Ethernet packets are displayed.

core provided by Xilinx in a demo was used and modified to receive data over Ethernet. The CPU essentially receives Ethernet packets and builds up all the required data in the PS. This is finally sent on to the custom cores running on the PL. The data that is sent to the DEATHs is shown in figure 5.11. With the trigger word, a process within the PS is started such that it knows how to initialize building up the data in a sequential fashion and in a specific manner. Care had to be taken, since Ethernet packets do not necessarily have a fixed length. This could make received data seem corrupt upon evaluation.

5.6.3 Control

The control of one or more DEATHs are carried out by the PC control software which was written in Python. This takes care of connecting to the DEATHs via an Ethernet connection. It allows the user to upload a waveform file which is in JSON[5] format to the devices. A single file can contain one or multiple

[5]Details of the format can be found online: http://www.json.org/

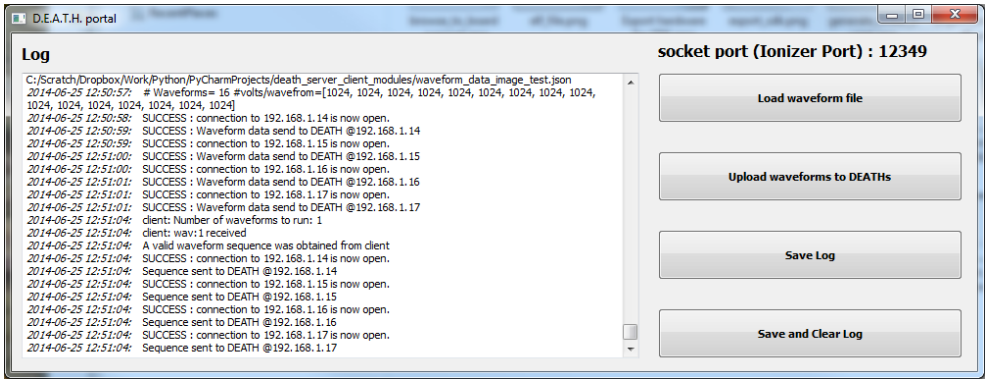


Figure 5.12: The screen shot of the software that controls and communicate with the DEATHs.

waveforms with various numbers of data points. We have initially tried just adding all the data to a text file, but it was a very inefficient way to store the data and just loading it would crash many programs. The JSON file format made storing large amount of data within a single file plausible for quick loading from programs. The Python software has a built-in library to read JSON format and is thus fast to read. The software also allows other programs to connect to it via a socket and send waveform sequences. The speed at which waveforms should be executed is also set via the software, which allows for the output rate of the waveforms to be changed in a given experiment. A screen shot of the program being used in the laboratory is shown in Fig. 5.12. A test program that was used to build up simple sequences of waveforms is shown in Fig. 5.13. It should be noted that during writing up this thesis the software was successfully incorporated into our main control system by David Nadlinger. The programs described in this chapter can still be used in order to run a single DEATH from a regular PC and have been successfully used in the laboratory as well.

5.6.4 Setup in Laboratory

After careful design of the electronics to deliver our waveforms, the physical connections to the trap had to be considered carefully as well. Shielded 110 Ω differential digital cables with mini XLR connectors have been used to connect the DEATHs. The decision on the cables was rather tedious and having too little experience, we have made use of the same ones that the group in Siegen use. The cable are standard audio cables used in the music industry, thus we found the connectors typically used with them, namely XLR connectors. These cables are all cut to the same length and are bunched together to reduce noise. They plug into a custom made box as can be seen in Fig. 5.14. What is not shown in the figure is that short twisted pair cables plug into the filter

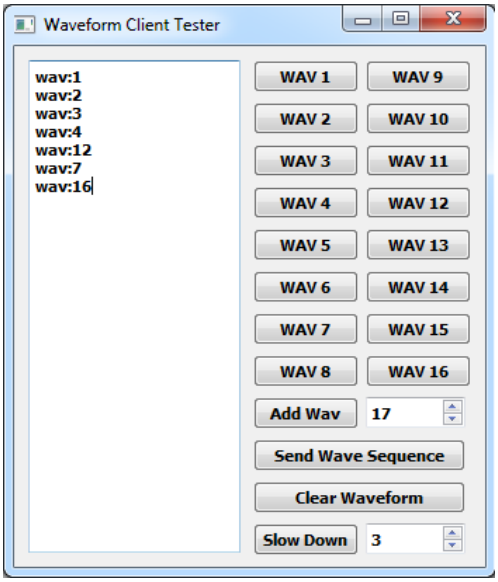


Figure 5.13: A temporary solution to build-up waveform sequences for execution on the DEATHs.

board box using crimp connectors..

5.6.5 Calibration and results

In order to have reliable outputs from the DEATHs the ouput of each channel was calibrated with a Keithley 2100 digital multimeter. From the datasheet the DACs are expected to be linear in their response and given the latest firmware a voltage can be converted into a DAC word that has to be sent with the following equation.

$$W = \text{Round}\left[\left(2^{16}-1\right)\frac{V - V_{\text{min}}}{V_{\text{max}} - V_{\text{min}}}\right]. \tag{5.1}$$

The results for requesting {-7,-3,0,+3,+7}V are shown for channel 1 and 16 in Fig. 5.15. Only two channels are shown for simplicity. We have used these measurements in order to calibrate the DEATHs as well as to determine the linearity. A straight line was fitted to the data and the results are shown in Fig. 5.15. The variance of a set voltage has been measured after calibration on a Keithley 2100 digital multimeter to be typically within 10uV of the requested value.

Any resonances that would occur at a motional frequency could be devastating, resulting in an unwanted heating of the ion. In Fig. 5.16 the frequency spectrum over typical motional frequencies we typically care for is shown and the level is low with no clear resonances. The noise floor within this range

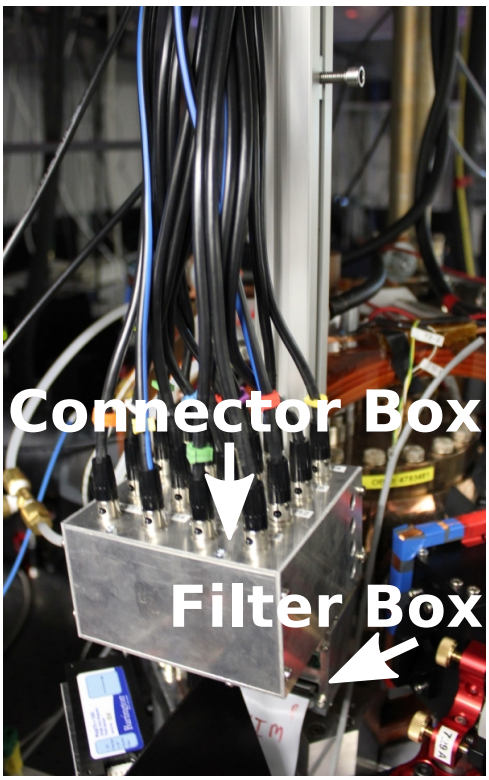
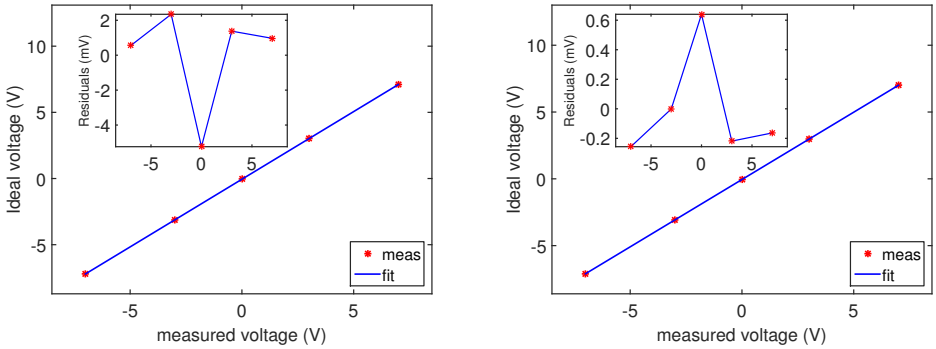


Figure 5.14: DC electrode cables, connector box and hidden filter box.



(a) Linear fit to the output of channel 1.

(b) Linear fit to the output of channel 16.

Figure 5.15: A check of the linearity of the voltage response of the DEATHs 10 m away from the outputs.

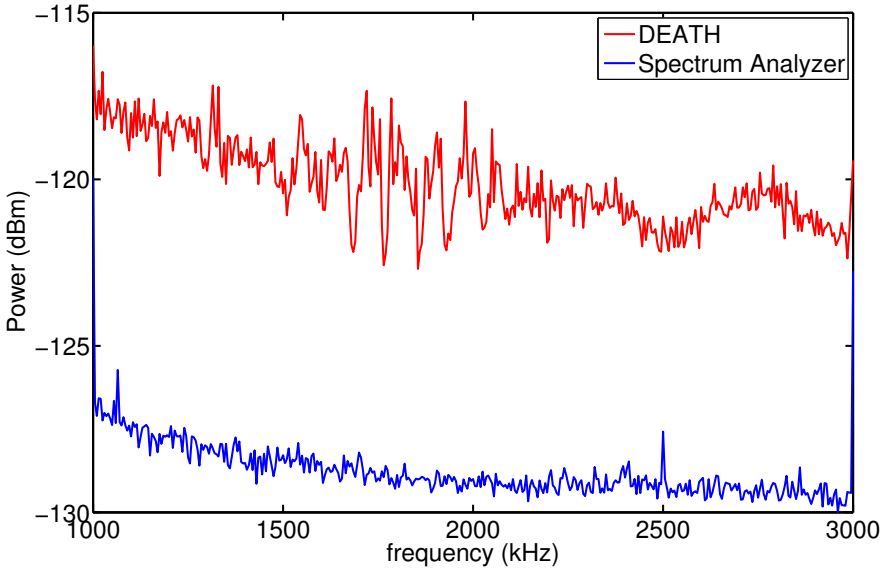
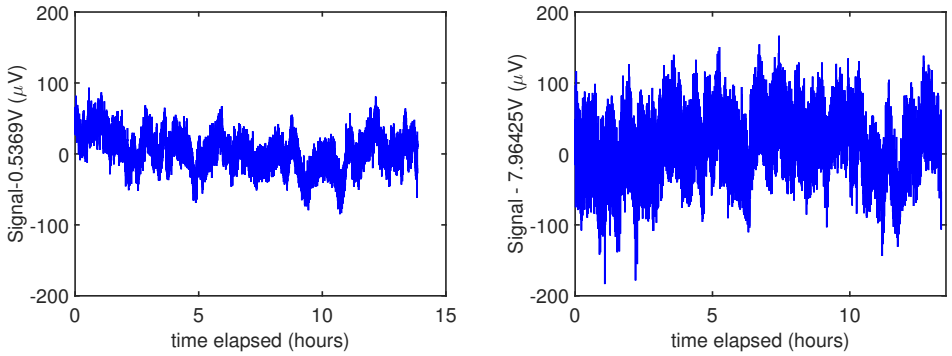


Figure 5.16: Frequency spectrum taken from one channel of the DEATH over the frequency range of typical motional frequencies with a bandwidth (BW) and video bandwidth (VBW) of 10 Hz.

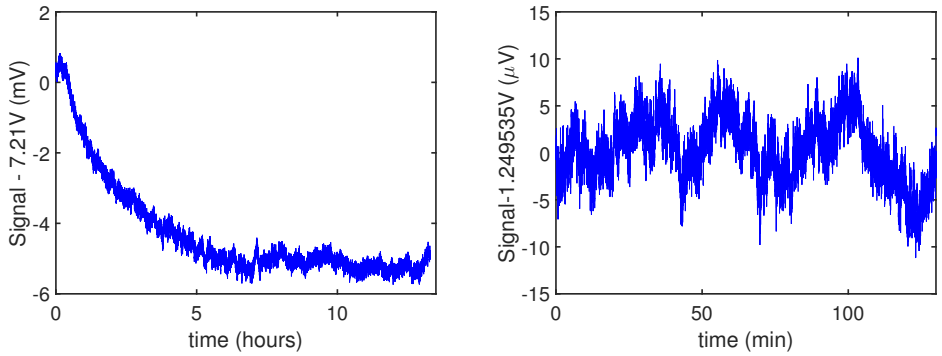
is around -131 dBm/Hz which translates into $63\,\mathrm{nV}/\sqrt{\mathrm{Hz}}$. This compares well with voltage solutions from other groups like NIST [Bowler 13] which has $120\,\mathrm{nV}/\sqrt{\mathrm{Hz}}$ (no frequency specified). The group at Siegen [Baig 13] has $250\,\mu\mathrm{V_{RMS}}$ over a span of 100 kHz at ± 9 V. In comparison the DEATHs exhibit $20\,\mu\mathrm{V_{RMS}}$ over the same span where the average noise floor within the range shown in Fig. 5.16 has been used.

Another important issue is drift over the timescales required to perform experiments. Fig. 5.17 shows the drift of a constant set voltage of both the DEATH and the AD5371 as it changes over a period of roughly 14 hours. The data has been measured over night for both the slow DAC solution (AD5371) and the DEATHs. For the DEATHs the standard deviation over 14 hours is 24 µV and the maximum change in the voltage is 178 µV. In comparison the slow DACs has a standard deviation over a similar time scale of 38 µV and the largest change in voltage is 350 µV. It should be noted that both figures were taken with two different sampling rates and thus the slow DAC solution seems noisier, due to a higher sampling rate. In this comparison the DEATH custom PCBs were fully covered with thermal pads and each with at least 3 heat sinks with the box opened at the top. In Fig. 5.18a the drift and eventual settling of the set constant voltage can be observed when the custom PCBs are only partially covered (sections 3 and 4 in Fig. 5.5) with no heat sinks. Fig. 5.18b shows the drift over several minutes once all the thermal



(a) DEATHs drifts, sampling rate of 2000 ms. (b) AD5371 drifts, sampling rate of 500 ms.

Figure 5.17: Drifts measured over several hours for both a single DEATH and AD5371 voltage channel. Measurement were made with a Keithley 2100 digital multimeter.



(a) DEATHs drifts, sampling rate of 500 ms partially coverred with thermal pads.

(b) AD5371 drifts, sampling rate of 500 ms. DEATHs fully coverred with thermal pads and heat sinks.

Figure 5.18: Drifts measured with and without thermal pads to cover the entire custom PCB of the DEATHS. Measurement were made with a Keithley 2100 digital multimeter.

pads and sinks are in placed. Lastly the box we have used as shown in Fig. 5.4 when close leads to an increase of about 10 degrees measured at the heat sinks of the Zynq 7010 chips. We measure the temperature with the lid off to be 27 compared to 37 degrees when the lid is closed. Having achieved drift performance similar to that of a commercial device we have not investigated the topic much further. It might be worth further investigation when better performance would be required.

5.7 Future improvements and projects

As with the EVIL the DEATHs are also undergoing continuous evolution in both the bit-stream and software design. For the latest improvements and additions see David Nadlinger's future thesis/documentation. Since the boards that are in the laboratory currently are still the first version of the DEATHs, it is remarkable that they operate as well as they do, speaking from an engineering point of view. Typically a few iterations are required to get to a stable solution. Though the DEATHs are in general behaving, we have a few improvements we would strongly recommend.

- The need for level shifters on the SPI lines going to the DAC chips are necessary to increase the reliability of the SPI communication. We have neglected this in the initial design.
- Addition of a thermalization solution to the entire board. Currently they are cooled with fans sitting in the back and heat dissipating material is stuck on the custom pcb to evenly distribute heat. We would suggest heating up the boards to around 40 to 50 degrees and enclosing them. We have seen that temperature changes on the board can lead to drifts in voltages. Due to the mother-daughter board design, a cooling/heating solution is not trivial, but should be possible with a controlled heating element as designed by Peter Märki in our department. If heating turns out to be unreliable another casing with better airflow and shielding the whole of the DEATHs with thermal pads should improve performance.
- For better performance a second opamp can be added to the output stages to correct for drifts. Another possiblility would be to replace the DACs with DACs that output $\pm 10\,\mathrm{V}$ already. With such DACs, no need for a two stage gain chain would be required for amplification which magnifies noise and drifts when they are coming from the DAC. The DEATH project would be simple enough to change to allow for different DACs. Slight changes to the firmware would be required afterwards, but the main system would still function as before.
- It would be advantages to add some pinouts to a dedicated PMOD on the custom board. In a current project we do fast branching of waveforms, but we have to go through the rather slow interface with PMOD on the commercial FPGA. They are slow, for you can only access them through the PS first to set them in the PL. We did not initially notice this.
- There are still enough space on the custom board, and switching to the MicroZed 7020 will give even more pin-outs which can be use and then adding a single AD5371 chip on the back would allow the DEATHs to also control slow voltages over ETH to set the shim electrodes.

Chapter 6

Transport

Transport of ions are required in the original proposals for the QCCD architecture [Wineland 98, Kielpinski 02] as well as the extension to using transport gates as proposed by D. Leibfried [D. Leibfried 07]. This chapter starts with a discussion on adiabatic and diabatic transport of ions. Performing transport, combining and splitting operations of ions, multiple time varying voltages has to be applied to electrodes. In this chapter I will discuss how to obtain the potentials to apply to the electrodes and then on how to obtain waveforms. Additionally I will introduce a formalism for creating complex transport experiments within ion traps controlling multiple potential wells.

6.1 Adiabatic and diabatic transport

The first requirement for transporting ions reliably is to be able to place ions at arbitrary locations along the ions trap axis [D. Hucul 08]. A second aim is that a prepared quantum state does not undergo a large motional change $\langle n \rangle \ll 1$ during transport where $\langle n \rangle$ is the average motional excitation to allow for multi-qubit gate operations. This is not strict condition since the ion can be sympathetically cooled as has been done in [Jost 09, Hanneke 09]. However cooling requires extra time and complexity, so a low temperature starting point is desirable

Various theoretical approaches have been taken to designing waveforms which produce low excitations of the ion. At NIST they considered the amount of energy transferred to the ion during transport. Here we will consider the approach taken in [Reichle 06], since it not only gives insight into what is considered adiabatic transport, but similar arguments and equations have been studied by others using the approach of invariant engineering [Palmero 14, Guéry-Odelin 14, Lu 14]. When we want to transport an ion, we would ideally want to displace a potential $\Phi_{\text{ideal}}(z)$ from its original location, say at position $z = 0$ to the position $z = z_0(T)$ within a time T ideally by perfectly

translating it along the axis. Experimentally we do not have limitless control over our system and we can therefore not construct the exact potential we desire at each point in time. Therefore, during transport the resulting potential may be described as the linear combination of the ideal electric potential $\Phi_{\text{ideal}}(z - z_0(T))$ at time t and position $z_0(t)$ and a residual electric potential $\Phi_{\text{res}}(z)$. In our experiments we typically add an offset which is not dependent on the axial position to our potential in order to ensure stable trapping conditions and here we also allow this to vary in time according to $\Phi_0(t)$. It will be shown that when considering motion along transport axis it can be neglected. The total electric potential the ion experiences during the process is then given by

$$\Phi(z,t) = \Phi_{\text{ideal}}\left(z - z_0(t)\right) + \Phi_{\text{res}}\left(z\right) + \Phi_0(t) \tag{6.1}$$

The desired electric potential is harmonic with respect to position, and thus given by the functional form

$$\Phi_{\text{ideal}}(z) = \frac{1}{2}\frac{m\omega_z^2}{Q}z^2. \tag{6.2}$$

In the laboratory frame we can write the equation of motion of a particle with charge Q and mass m and trapping frequency ω_z in a potential $\Phi(z,t)$ as [Reichle 06]

$$a = \ddot{z}(t) = -\frac{Q}{m}\frac{\partial}{\partial z}\Phi(z,t) \tag{6.3}$$

From here we note that the time-dependent offset $\Phi_0(t)$ would not contribute to the dynamics and may be neglected. The equations of motion are obtained by substituting the potential Eq. 6.1 into Eq. 6.3 and then obtaining

$$\ddot{z}(t) + \frac{Q}{m}\frac{\partial}{\partial z}\Phi_{\text{ideal}}(z - z_0(t)) + \frac{Q}{m}\frac{\partial}{\partial z}\Phi_{\text{res}}(z) = 0. \tag{6.4}$$

The second term is described by ideal potential form given by Eq. 6.2 and the third term we identify with the residual potential giving rise to a residual acceleration $a_{\text{res}}(z) = -(Q/m)\partial\Phi_{\text{res}}(z)/\partial z$. The equations of motion in the laboratory frame is

$$\ddot{z}(t) + \omega_z^2\left(z - z_0(t)\right) - a_{\text{res}}(z) = 0. \tag{6.5}$$

Moving to a frame in which we follow the minimum of the potential we may use a change of variables $u(t) = z - z_0(t)$ and write the differential equations as

$$\ddot{u}(t) + \omega_z^2 u(t) = -\ddot{z}_0(t) + a_{\text{res}}(u(t) + z_0(t)) \tag{6.6}$$

In [Reichle 06] they go on to show that if you expand the residual acceleration around the potential minimum $u(t) = 0$ to second order

$$a_{\mathrm{res}}(u(t) + z_0(t)) \approx \; a_{\mathrm{res}}(z_0(t)) + \frac{\partial}{\partial u} a_{\mathrm{res}}(z_0(t))u(t) + \frac{\partial^2}{\partial^2 u} a_{\mathrm{res}}(z_0(t))u^2(t) \tag{6.7}$$

you can write the resultant equations of motion [Reichle 06]

$$\ddot{u}(t) + \omega_z^2 \left(1 + \frac{a'_{\mathrm{res}}(z_0(t))}{\omega_z^2}\right) u(t) - \frac{a''_{\mathrm{res}}(z_0(t))}{2} u(t)^2 ... = -\ddot{z}_0(t) + a_{\mathrm{res}}(z_0(t)) \tag{6.8}$$

where $a'_{\mathrm{res}}(z_0(t))$ and $a''_{\mathrm{res}}(z_0(t))$ represent the first and second order derivative with respect to u. Making the associations $f(t) = -\ddot{z}_0(t) + a_{\mathrm{res}}(z_0(t))$ and $\omega^2(t) = \omega_z^2 \left(1 + a'_{\mathrm{res}}(z_0(t))/\omega_z^2\right)$. Following the derivation of R. Reichle *et al.* closely we can disregard the second and higher derivatives for certain trap geometries where the width of the electrodes used in transport with respect to the ion distance results in ratios between 0.5 and 1. In our trap this is true for all the electrodes typically used in transport, but not the central electrode number 8 shown in Fig. 3.3. Under these conditions Eq. 6.8 is given by [Reichle 06]

$$\ddot{u}(t) + \omega(t)u(t) = f(t) \tag{6.9}$$

This differential equation describes a parametrically driven harmonic oscillator with energy given by the Hamiltonian

$$H(t) = \frac{p^2}{2m} + \frac{1}{2} m\omega^2(t)u^2 - mf(t)u(t) \tag{6.10}$$

with $p = m\dot{u}$. From the definition of an adiabatic process a system is adiabatic if at all times the system stays in an instantaneous eigenstate of the Hamiltonian. A constraint we pose on the transport is that $\omega(t) \approx \omega_z$ such that there is minimal parametric driving involved due to the residual acceleration. For this to hold we require

$$\frac{a'_{\mathrm{res}}(z_0(t))}{\omega_z^2} \ll 1 \tag{6.11}$$

$$|\ddot{z}_0(t)| \gg |a_{\mathrm{res}}(z_0(t))|. \tag{6.12}$$

From these constraints we additionally find that on this type of transport the time T of displacing the potential should be much larger compared to the period of oscillation the ion experiences in a potential with angular frequency ω_z. This introduces the adiabatic time constraint as

$$T \gg \frac{1}{\omega_z}. \tag{6.13}$$

Under the above constraints the aim is to investigate the ideal trajectory over time that the ion has to follow to minimize the energy transfered to the oscillator after transport is complete. Under the approximations made above the energy transferred to the oscillator is given by $m|\Xi(t)|^2/2$ with

$$\Xi(t) = -e^{i\omega_z t} \int_0^t \ddot{z}_0(t') e^{-i\omega_z t'} dt' \tag{6.14}$$

when the transport is complete and the integral is evaluated over all of time [Reichle 06]. For a trajectory which has an error function profile the transferred energy is minimized. Another useful adiabatic condition on transport found in [Blakestad 10, Hucul 06] is that

$$\omega_z \gg \frac{1}{a(T)} \frac{da(t)}{dt} \tag{6.15}$$

with $a(t) = \ddot{z}_0(t)$ gives some information on how the transport should be carried out. The faster we would want to transport the higher ion oscillation frequencies should be used.

A useful approach that is equivalent to the consideration given above is to consider the size of the coherent state that is generated by a given transport trajectory $z_0(t)$ [Lau 11]

$$\alpha(t) = \sqrt{\frac{m\omega_z}{2\hbar}} \left(-e^{-i\omega_z t} \int_0^t \dot{z}_0(t') e^{i\omega_z t'} dt' \right). \tag{6.16}$$

We can see the equivalence when we consider the amount of energy transferred $\hbar\omega_z|\alpha(t)|$ in terms of the coherent state in that it is equivalent to the energy transferred $m|\Xi(t_1)|^2/2$ using Eq. 6.14 when the transport starts and ends without motion $\dot{z}(t=0) = 0$ and $\dot{z}(t) = 0$. We typically only care about the state of the ion after transport and it is easy to see from Eq. 6.16 that if we generate a trajectory $z_0(t)$ such that the velocity $\dot{z}_0(t)$ does not contain the trap frequency components the ion will end in the initial state after transport [Bowler 12]. Nevertheless even though we have no final motional excitation left, during transport we could have generated large coherent states. This has been experimentally demonstrated in diabatic transport experiments using linear Paul traps [Walther 12, Bowler 12]. The timing constraint is shown in Walther *et al.*. This is analogous to moving a bowl with a marble in between two locations such that at the end the marble is still at the middle of the bowl after transport. You could do this really fast, but then you would have to be intelligent and have good timing at the end such the marble ends up without any oscillations [Alonso 13].

Other more recent theoretical investigations of fast transport have considered the possibility to use invariant engineering in order to achieve low motional excitations after transport [Palmero 14, Guéry-Odelin 14, Lu 14]. Within these recent theoretical investigations using invariant engineering the same dynamics as describe by the parametrically driven oscillator Eq. 6.10 is studied as in [Reichle 06]. The difference being that in [Palmero 14, Guéry-Odelin 14, Lu 14] they find an invariant of the system and expand the dynamics in terms of the invariant. Depending on the requirements of the transport scheme one sets up constraints on the ion trajectory itself. As with most of the experiments with ion transport the goal is to end in the same state as in which was started. The problem is then solved using the invariant formalism. The difference to the work done at NIST [Reichle 06] is that during the transport the system stays in the instantaneous eigenstate of the invariant and not of the Hamiltonian. Invariant engineering can be used to consider how to generate trajectories given specific constraints. In [Lu 14] they investigated how to obtain a trajectory when there is variation of the potential curvature during the transport. Recently a paper has been published showing which trajectory would be necessary to transport pairs of ions with different masses using invariant engineering [Palmero 14]. The invariants based approach to transport is a more general method than the methods proposed by NIST, but at the cost of added complexity.

6.2 Implementation of transport

In order to generate waveforms for the transport of ions we require a model of how the potential of each of the electrodes within our segmented ion trap contribute over the desired region in which we would like to perform control. Within our 3D Paul trap we can restrict ourselves to the manipulations of potentials along the z-axis, assuming that the radial confinement is strong enough that we are always close to this axis. With this simplification we only require to the contribution of each electrode along the transport axis. In the following, we consider how to obtain such information from simulations of the trap. It should be noted that our trap is symmetric and thus opposing DC-electrodes contribute the same potential along the z-axis and we can therefore apply the same potential to each. This reduces the number of voltages sources required in order to perform the transport operations.

Once we have the contributions of each of the electrodes we can find the combinations that are required to produce a given harmonic potential with specific properties: frequency, position and offset. How to specify and implement this within a waveform is a routine task in ion transport and the approach we have taken will be discussed within the sections that follow.

6.2.1 Electrode potentials

As soon as an ion trap does not resemble the simple geometry of a Paul trap having circular rods as been discussed in chapter 2 the potentials produced by the electrodes do not have a simple analytical solution. When this is the case a boundary element method (BEM) or finite element method (FEM) package solver is then used in order to investigate the potentials produced by the geometry of the trap.

Since the electric potential is a scalar it adheres to the superposition principle. Given that our ion trap consist out of many electrodes this mean that we can simulate the effect of each electrode within the geometry by setting its potential to 1 V and ground (0 V) all other electrodes and numerical find its potential contribution throughout the trap. In the case of transport we can record the potential from each of the electrodes as a function of the position and form a basis with which we can build up arbitrary shaped potentials.

In our laboratory we have adopted BEM (Boundary Element Method) code that had been developed at NIST. This code allows you to import your trap design from software like Inventor. There is a limiting condition when using the NIST BEM package version we have and that is the simulation can not use more than 2 GB of RAM[1]. Thus in order to simulate large structure we chose to split the simulation region up in parts and simulate each part individually and afterwards concatenate the solutions.

Another alternative is to use COMSOL. COMSOL also has an option to import geometries from Inventor. Within COMSOL is a multi-physics software and the setup of the physics of interest that needs to be considered is easily done. It takes manual selection of the electrodes in order to specify each within the simulation interface. Having set up COMSOL such that we can solve for the potentials of the DC electrodes we have written a Matlab script to take the various electrodes to iteratively set each electrode to 1V and the rest to ground (0 V). After iterating through all the electrodes and simulating the potential contributions in space we extract the solutions of the potentials of each electrode along the z-axis. Having done this using both NIST's software package and COMSOL we have found a good agreement between the two packages see Fig. 6.1 for examples. In this figure the blue solid and red dashed curves show the potential along the trap axis when 1 V is applied to each of the inner electrodes in turn and all others are grounded. COMSOL seems to find values that are consistently higher. This can be seen from the residual between the COMSOL and NIST's software package solutions as shown over the same relevant section in Fig 6.1. The largest difference between the two sets are 2.9mV. Recently Christoph Fischer has investigated the differences

[1]The latest version has been updated by Robert Jordens at NIST and no longer exhibits this behavior.

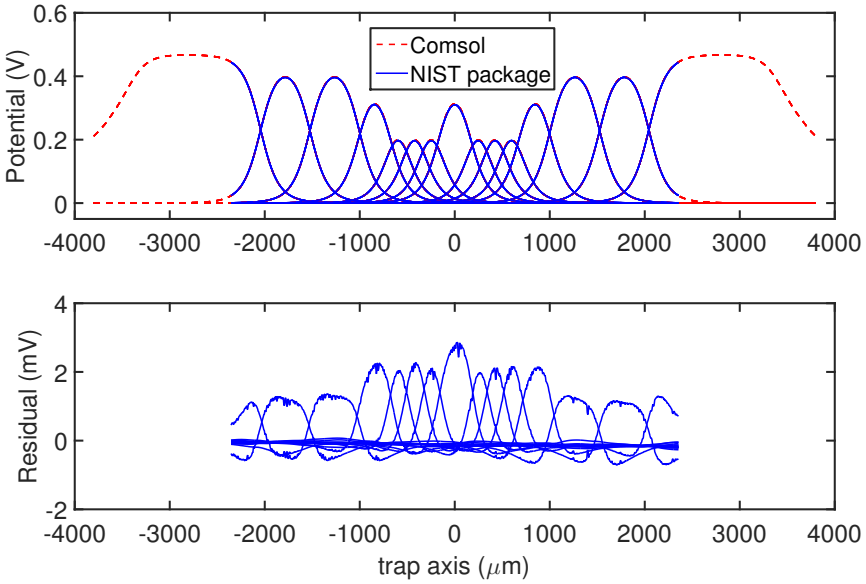


Figure 6.1: Comparison between the NIST software package and COMSOL solutions.

between the NIST software package and COMSOL in more detail and found COMSOL to be more accurate and faster [Fischer 15b].

6.2.2 Waveform generation

From the NIST BEM package and COMSOL simulations we have obtained the potential Φ_n contributions of each electrode along the z-axis as shown in Fig 6.1. The electric potentials adhere to the superposition principle, thus if we apply a set of voltages $\{e_n\}$ to all the electrodes we can write the resulting potential as

$$\Phi(z_i) = \sum_{n=1}^{N} \Phi_n(z_i) e_n \tag{6.17}$$

where $\Phi_n(z_i)$ is the contribution of each electrode at a the point z_i on the z-axis when 1 V was applied to a specific electrode and all others grounded as obtained from the simulation. By defining a vector $\vec{\Phi}_n = [\Phi_n(z_1), \Phi_n(z_2), ...\Phi_n(z_k)]$ which contains the potential contribution of the n-th electrode at all discrete values of z for which the potential was simulated and creating a voltage column vector $\vec{e} = [e_1; e_2; ...; e_n]$ we can write Eq. 6.17 more compactly as

$$\vec{\Phi} = \boldsymbol{M}\vec{e}. \tag{6.18}$$

Here the column vectors of M are formed by the vectors $\vec{\Phi}_n$ and therefore looks like $\boldsymbol{M} = [\vec{\Phi}_1, \vec{\Phi}_2, ...\vec{\Phi}_n]$. The vector $\vec{\Phi}$ therefore represent the formed potential by applying the electrode voltages $\{e_n\}$ to all the various electrodes with associated potentials $\vec{\Phi}_n$. To find an electrode configuration that results in a given requested potential we can use SVD (Singular Value Decomposition) of $\boldsymbol{M}$

$$\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* \tag{6.19}$$

with $\boldsymbol{U}$ and $\boldsymbol{V}$ unitary matrices and $\boldsymbol{\Sigma}$ containing the singular values of $\boldsymbol{M}$ on its diagonal and all other entries 0. The electrode configuration found using SVD is given by

$$\vec{e} = \boldsymbol{V}\mathrm{diag}(\boldsymbol{\Sigma}^{-1})\boldsymbol{U}^T\vec{\Phi}. \tag{6.20}$$

In using SVD to find a solution in the case of an overdetermined system it will find the solution which results in a least square fit to requested potential $\vec{\Phi}$. The cost function which is minimized for each output step of the waveform is

$$J = |\boldsymbol{M}\vec{e} - \vec{\Phi}|^2 \tag{6.21}$$

In Matlab one can solve Eq. 6.18 in one step by calling the command $\vec{e} = \boldsymbol{M}\backslash\vec{\Phi}$, otherwise using SVD command $[\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}] = \mathrm{svd}(\boldsymbol{M})$ and then using equation 6.20 to solve for the electrode configuration.

It is therefore a rather simple task to find a given electrode configuration for a fixed requested potential. The goal is to construct a waveform which is a collection of electrode configurations which when executed in succession results in potential changes that can then be used for transport or modulation of a given potential. The logical first approach to solve for a waveform would be for each time step to solve the potential as it is required in that instant in time and finally concatenate the results for the required voltages. This requires invoking Eq. 6.20 for each time step. In practice it leads to solution jumping [Blakestad 10], since the SVD technique does not take into account the previous time step voltage configuration and only tries to minimize the least squares problem. Another problem comes in constraining voltages within bounds, which typically can not be done without manual checks. One way to resolve this is to take into account the voltage configuration of the previous step and aim to minimize the cost function

$$J = |\boldsymbol{M}\vec{e}_q - \vec{\Phi}_q|^2 + \alpha|\vec{e}_q - \bar{e}_{q-1}|^2 \tag{6.22}$$

where α is an user controlled parameter which can be set to either favour the term on the left or the term on the right. The term on the left represents the least squares problem and the one on the right limits the amount by

which voltages may change. This problem is classically know as Tikhonov regularization which has the form

$$J = |\boldsymbol{M}\vec{e} - \vec{\Phi}|^2 + |\Gamma\vec{e}|^2. \tag{6.23}$$

where $\boldsymbol{\Gamma}$ is the Tikhonov matrix, which can be viewed as an operator acting on $\vec{e}$. This problem has been well studied and tools are readily available to solve it [Hansen 98]. Both NIST and Mainz have used this technique [Reichle 06, Singer 10] and we have initially performed transport in our trap using it as well. The difficulty with this technique is to find the appropriate α parameter from Eq.6.22, which we had to find experimentally by trial and error. It is possible and one can obtain smooth electrode functions over time as well as deep wells, but it can be very time consuming.

We have recently started to investigate quadratic programming similar to that used in [Blakestad 10, Bow 15]. The typical quadratic problem is stated as follow [Boyd 04]

$$\begin{aligned} J &= \frac{1}{2}\vec{x}^T\boldsymbol{P}\vec{x} + \vec{q}^T\vec{x} + r \\ \boldsymbol{G}\vec{x} &\leq \vec{h} \\ \boldsymbol{A}\vec{x} &= \vec{b}. \end{aligned} \tag{6.24}$$

Here $\vec{x}$ is the vector of parameters $\{x_0,x_1,...,x_n\}$ we are solving for subjected to inequality constraints $\boldsymbol{G}\vec{x} \leq \vec{h}$ and equality constraints $\boldsymbol{A}\vec{x} = \vec{b}$. A linear least squares problem as stated in Eq. 6.21 is a subset of this type of problem without constraints. This can be seen as

$$J = |\boldsymbol{M}\vec{e} - \vec{\Phi}|^2 = \vec{e}^T\boldsymbol{M}^T\boldsymbol{M}\vec{e} - 2\vec{\Phi}^T\boldsymbol{M}\vec{e} + \vec{\Phi}^T\vec{\Phi} \tag{6.25}$$

which can be written in the form of the cost function of Eq.6.24 as

$$\begin{aligned} J &= \frac{1}{2}\vec{x}^T\boldsymbol{P}\vec{x} + \vec{q}^T\vec{x} + r \\ P &= 2\boldsymbol{M}^T\boldsymbol{M} \\ q^T &= -2\vec{\Phi}^T\boldsymbol{M} \\ r &= \vec{\Phi}^T\vec{\Phi}. \end{aligned} \tag{6.26}$$

There is no analytic solution to the general quadratic programming problem as stated in Eq. 6.24. Even though this is the case the problem has been well studied and many numerical solvers exist to solve it, for example, see Gurobi [Gurobi Optimization 15]. Robin Oswald has recently looked in to how to setup the quadratic programming problem to solve for transport waveform and have Gurobi as solver. The latest implementation and the considerations involved will be found in his Masters thesis.

6.2.3 Transport profile creation

In order to generate waveforms that are smoothly varying in time we must specify the required potential along the z-axis for each time step. The approach taken by R.B. Blakestad is to specify the potentials that should be formed at each point along the trap which included both the radial and transport direction confinement requirements [Blakestad 10]. This he had to be done, since his transport was carried out in a X-junction ion trap and the transport was thus not along a single straight line. When we shuttle ions along one direction (z-direction in our case) the method employed by Mainz is useful. Their idea for transport of this type is to consider that the ion is well confined in the radial direction and specify that a harmonic potential should be formed at each point along the z-axis [Singer 10, Schulz 06]. The transport we perform in our trap is along one direction and we have decided to build on the approach employed by Mainz. We have extended the method to make it possible to include multiple harmonic wells to allow for individual ion control. This was required in order to perform the parallel transport gate operations discussed in chapter 8. The resulting electric potential using the same form as in Eq. 2.21 is given by

$$\Phi_{\mathrm{st}}(t) = \sum_{n=1}^{N} \frac{1}{2q} m\omega_{z,n}(t)^2 (z - z_n(t))^2 + C_n(t). \tag{6.27}$$

The equation is a sum of N potentials along the z-axis, where the index n represents a potential with motional frequency $\omega_{z,n}(t)$ and the associated minimum position of the potential $z_n(t)$. Both parameters are allowed to be time dependent in the case of ion transport. The constant $C_n(t)$ offset is included to allow for an offset to the potential which is included in order to simplify experimental realization, for we have found it allows the manipulation of the radial confinement properties, in particular the rotation of the radial eigenmodes. This seems to be important for the long term stability of the ions, which we think is related to Doppler cooling efficiency. Within the experiments we would find a constant value for the offset which ensures stable trapping conditions and use this within transport experiments, thus all $C_n(t)$ for the different potentials are constant and equal. In the experiments performed N was maximally two, but it should be noted that the techniques and procedures to find the waveforms are not necessarily limited to this value. The final constraint will be the geometry of the trap and the potentials that can be formed with the electrodes.

In order to manipulate multiple ions within an ion trap we have developed Matlab code in which the user can specify various harmonic potentials $\Phi_i(t)$ that vary simultaneously in time. The user has to make sure that the various potentials that are specified are not out of reach of the electrode voltages which can be applied. For our current voltage sources, the DEATHs (see chapter 5),

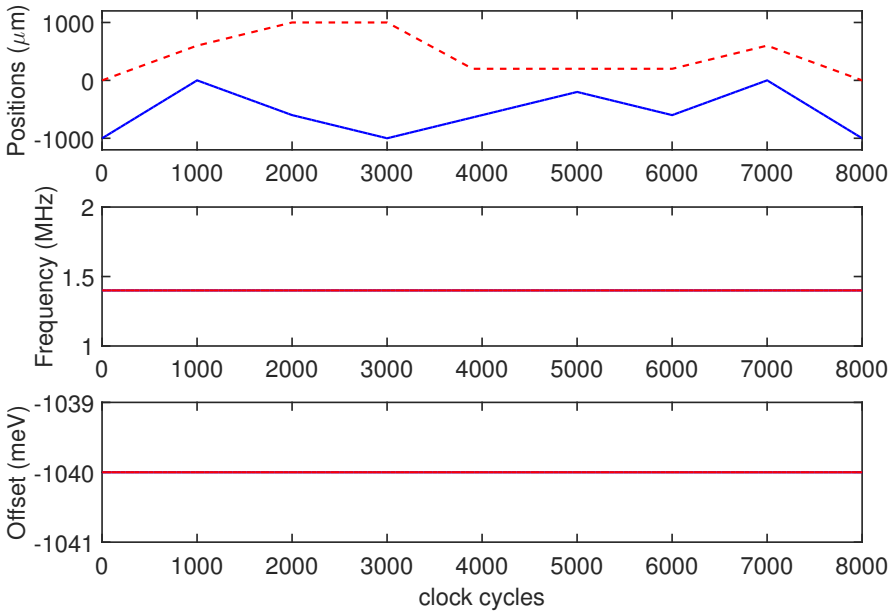


Figure 6.2: Specification of the properties that waveforms should exhibit at each update of the electrodes. The example shown here is what was used in the parallel transport quantum logic gates experiment.

this is limited to be within ± 10 V. It would for example not make sense to specify two harmonic potential at the exact location with different offsets and frequencies. The code is written such that it is simple to specify the frequencies, positions and offsets of each of the independent harmonic potentials over time. An example is shown in Fig 6.2, which is the experimental description of the parallel process which is discussed in chapter 8. The reason why clock cycles are shown is such that the user can determine what velocities to expect for the various transport routines, since the clock rate can be changed in the experiments. Within the parallel process experiment two wells have to move to the various locations and this was encoded in what is shown in Fig. 6.2. The axial frequency was to be set to 1.4 MHz for both potentials and they have a constant offset of -1040 meV. The frequency has been chosen from experience in working with our trap and the DEATH system and it was chosen such that the frequencies are simpler to maintain for the given value during transport.

6.2.4 SVD, Tikhonov regularization and quadratic programming

To demonstrate some of the differences we have found when using SVD, Tikhonov regularization and quadratic programming to solve for a given transport sequence, consider transport from -1000 µm to +1000 µm with an axial frequency

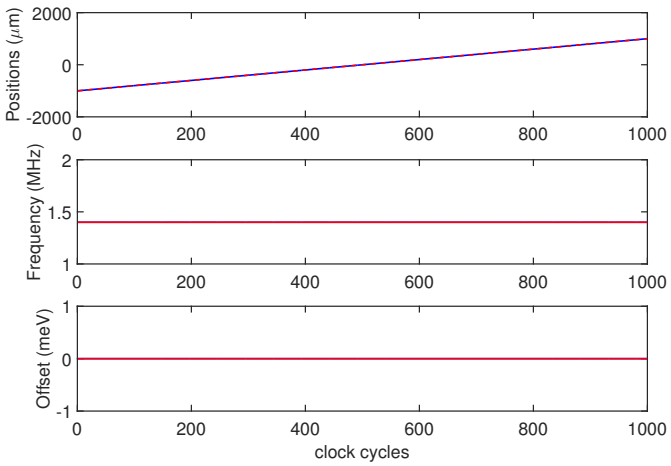
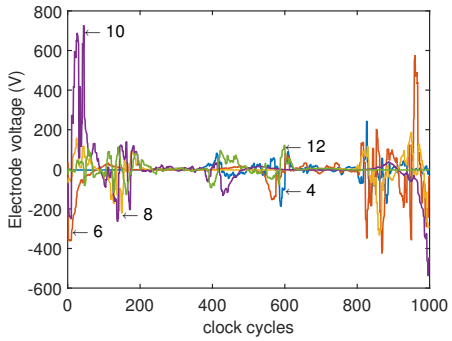


Figure 6.3: Example experiment transporting from -1000 µm to +1000 µm.

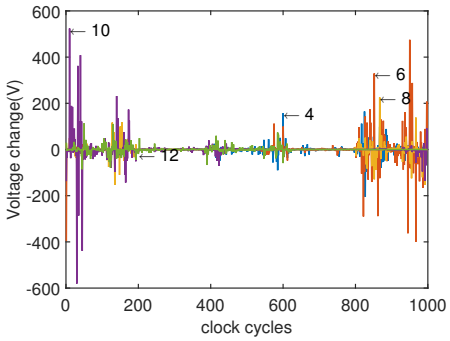
of 1.4 MHz and no DC-offset. The sequence is shown in Fig. 6.3 using my formalism for representing a transport sequence.

As have been mentioned, SVD does not allow for constraint of the electrode voltages within the waveform, or the change in the voltages from step to step. In Fig.6.4a the voltages as have been found by using SVD are shown for electrodes 4,6,8,10 and 12. It is clear that the voltages found are not limited at all between ±10 V as is required for using the DEATHs. The changes of these voltages as shown in Fig.6.4b also have no constraint and the peak like structure is indicative of solution jumping. From Fig. 6.7a an instant during the waveform the potential is evaluated when the harmonic potential which has to be formed is at the origin. We can see that the overall achieved potential has some complicated structure. Given this structure the solution in the vicinity of the required potential is near perfect match. From using the SVD technique it is clear that limitations on the voltage ranges has to be applied and that from step to step for this example you need at one point a maximum of 500 V to change in one clock cycle. High speed DACs can reach slew-rates of 100 V/µs as in the case with the DEATHs. This is assuming only the fast DACs are connected and no filters are in the path to the electrodes which is not the case and limits the maximum slew rate. Clearly another approach has to taken to avoid the shortcomings of SVD.

Making use of Tikhonov regularization we can limit the change of the voltages from one time step to the next via the α parameter in Eq. 6.22. Solving the exact same problem, but manually finding a suitable α=0.008 and setting the starting voltages for solving high the voltages for electrode 4,6,8,10 and 12 are shown in Fig. 6.5a. The voltages are limited to the output range ±10 V for the entire waveform, thus a clear improvement over using only SVD. The voltages

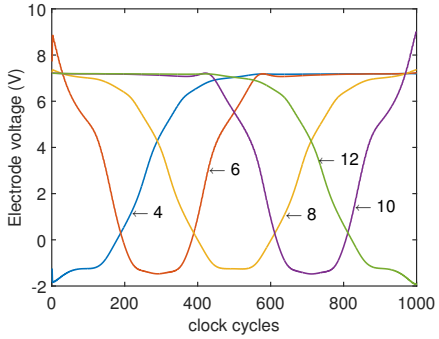


(a) Voltages of electrodes 4,6,8,10 and 12.

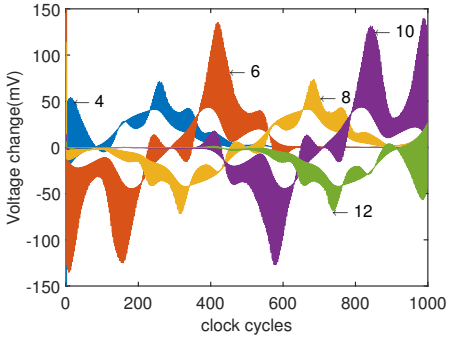


(b) Change of the voltages of electrodes 4,6,8,10 and 12.

Figure 6.4: Selected electrode voltages and the changes from the solved waveform of the transport sequence using SVD.

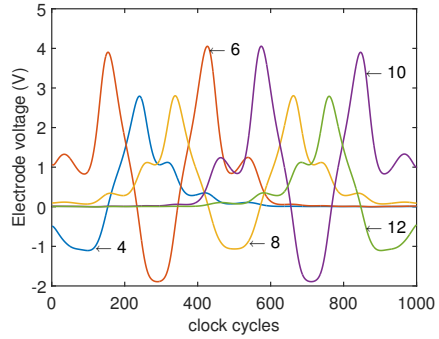


(a) Voltages of electrodes 4,6,8,10 and 12.

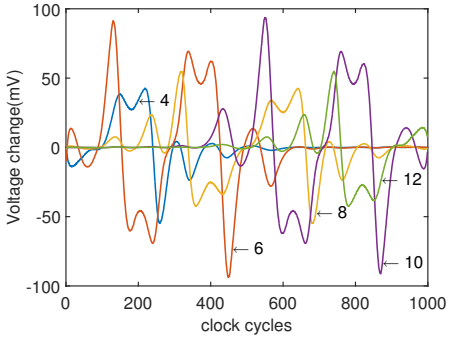


(b) Change of the voltages of electrodes 4,6,8,10 and 12.

Figure 6.5: Selected electrode voltages and the changes from the solved waveform of the transport sequence using Tikhonov regularization.

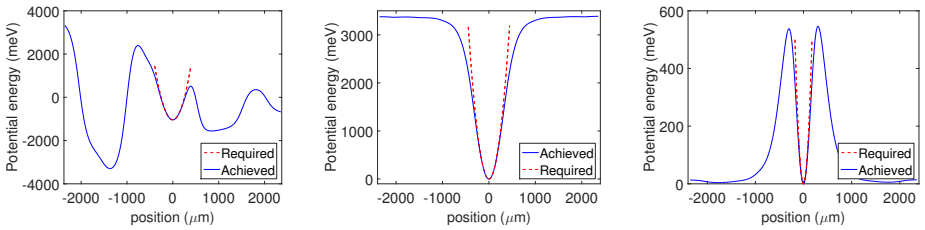


(a) Voltages of electrodes 4,6,8,10 and 12.



(b) Change of the voltages of electrodes 4,6,8,10 and 12.

Figure 6.6: Selected electrode voltages and the changes from the solved waveform of the transport sequence using quadratic programming.



(a) Singular value decompisition (b) Tikhonov regularization (c) Quadratic programming

Figure 6.7: Instants of the potentials captured when the potential is at the origin when solving with SVD, Tikhonov regularization and quadratic programming.

seem to be smooth, but considering the differences in the voltages from step to step as displayed in Fig. 6.5b solution jumping still occurs. The jumps are in the range of 100 mV. Knowing that we have low pass filters in the path to the electrodes, the jumps will be filtered out. When manually changing the α parameter and the initial conditions we have found that this behavior can be reduced to range 10 mV and below. This can be very time consuming in order to find the correct value for each experiment. The solution for a given instance when the harmonic potential solved for is at the origin is shown in Fig. 6.7b. The potential does not have such complicated structure as with using SVD to solve the problem.

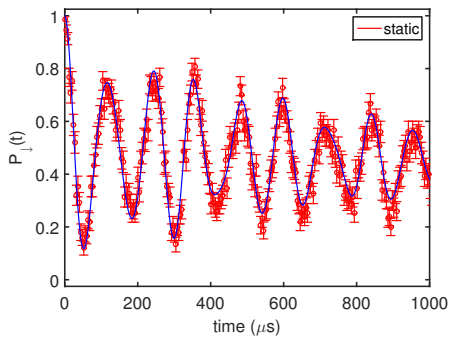
Using quadratic programming to solve for the same problem as well, the found solutions for the voltages of electrodes 4,6,8,10 and 12 are shown in Fig. 6.6a. The solutions are bounded as with using Tikhonov regularization. Considering the changes of these voltages as shown in Fig. 6.6b we see no solution jumping, but only continuous changes from each output of the voltages. An instant of the formed potential is shown when the harmonic potential specified during transport is at the origin is shown in Fig. 6.7c. Using quadratic programming it should be noted that solving for waveforms is not completely automated and still requires user input on occasion, though this occurs seldom.

Both Tikhonov regularization and quadratic programming methods are quick to solve, but neither is at the stage where there is no user input required. For the Tikhonov method the initial voltages should be chosen wisely by the user as well as the α parameter and can lead to solutions that are not viable at all. In the quadratic programming there are less parameters that need to be changed, but on occasion it requires the user to modify some cost coefficients, see Robin Oswald future thesis. One of the main differences in the Tikhonov method and the quadratic programming is that Tikhonov regularizations is an one step estimator, meaning that in the cost function as shown in Eq. 6.23 we

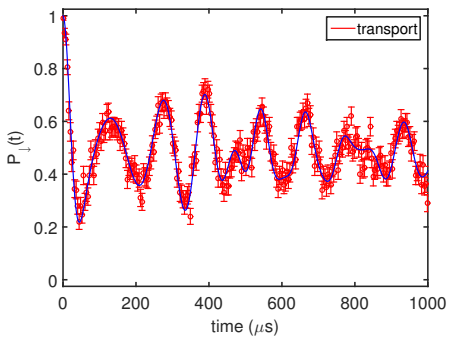
consider only one step into the future how the voltage should be limited. In the quadratic programming the entire evolution of the problem is considered and solved together with equality and inequality constraints.

6.3 Heating due to transport

In order to check if transport significantly heats up the ion we have prepared a $^{40}Ca^{+}$ ion in the ground state with an average $\bar{n}_{\text{static}}$=0.36±0.02. The measurement was made using a blue-sideband time scan in which the motional adding side-band was driven and the probability of finding the ion in the ground state was recorded as a function of the time the side-band was driven. The results for the ion kept statically directly after cooling is shown in Fig. 6.8a. Under a typical transport experiment as will be describe in the next section (see Fig. 7.5) an ion was prepared in the ground state and afterwards the ion was transported a distance of 500 µm along the transport axis, followed by a transit in the opposite direction by 1000 µm and finally back to its original location. The entire transport sequence took about 1 ms. After this the blue side-band is driven as discussed above with the result shown in Fig. 6.8b. From a fit for a thermal state to the data an $\bar{n}_{\text{static}}$=1.26±0.06 qaunta/s was obtained. Given that heating rate in our trap is 10 quanta/s we can conclude that the procedure added a maximum of around 1 quanta. We have not yet studied how to optimize with respect to heating, but this will become important for multi-qubit transport gates, which are planned in our future experiments.



(a) Fitting to the blue-sideband floping for a static ion.



(b) Fitting to the blue-sideband floping for an ion being transported.

Figure 6.8: Blue side-band flopping for in the cases where the ion is static and being transported.

6.4 Final remarks

There is currently no optimal way in which one can solve for potentials within ions traps over time to generate waveforms. There are always some tweaks that have to be made to a given method to optimize its outcome. In SVD one has to manually set electrodes to the bounding constraints after a solution has been found, which then can influence the potential which one is trying to achieve. Tikhonov regularization reduces this process and can give fast and good results, nevertheless it still requires occasional modification of the user controlled α parameter as in Eq. 6.22 in order to achieve the best possible outcomes. There are simple ways in which one can take bounding conditions on maximum and minimum voltages by adapting the size of the α parameter during the solution process in an intelligent way. What we have done was to put in a check and see if the bounds are surpassed and then try to solve again with an increased strength on the α parameter and finally if it does not work, clip the voltage at the extremes. The most convenient way so far is working with quadratic programming, there are still parameters to tweak, but there is no manual clipping necessary on the extremes of electrode voltages. It should be noted that Georgia Tech Research institute seems to have a code for waveform generation. No specifics are published with some information available online [Institute].

Chapter 7

Hamiltonian estimation

The time evolution of a closed quantum system is connected to its Hamiltonian through Schrödinger's equation. The ability to estimate the Hamiltonian is critical to our understanding of quantum systems, and allows optimization of control. Though spectroscopic methods allow time-independent Hamiltonians to be recovered, for time-dependent Hamiltonians this task is more challenging [Zhang 14, Mitra 03, de Castro 10, de Castro 13, Devitt 06, Shabani 11]. Here, using a single trapped ion, we experimentally demonstrate a method for estimating a time-dependent Hamiltonian of a single qubit. The method involves measuring the time evolution of the qubit in a fixed basis as a function of a time-independent offset term added to the Hamiltonian. In our system the initially unknown Hamiltonian arises from transporting an ion through a static, near-resonant laser beam [D. Leibfried 07]. Hamiltonian estimation allows us to estimate the spatial dependence of the laser beam intensity and the ion's velocity as a function of time. This work is of direct value in optimizing transport operations and transport-based gates in scalable trapped ion quantum information processing [Wineland 98, Bowler 12, Walther 12], while the estimation technique is general enough that it can be applied to other quantum systems, aiding the pursuit of high operational fidelities in quantum control [Martinis 14, Schutjens 13].

The chapter starts with an overview of the experimental sequence and following it the theory to describe ion transport with a varying velocity is discussed. The theory is derived in different frames of rotation, the first of which is the standard interaction picture which is followed by the more general time varying rotating frame. A new technique which we call Hamiltonian estimation is introduced within this chapter that allows the study of time-dependent systems experimentally. The technique is described in a general way with the hope that fields other than ion trapping could benefit from its use. Experimental results we have obtained and the application of Hamiltonian estimation to the data follow the theoretical discussion of the estimation technique. The

influence that the wavefronts of a Gaussian beam have on ion transport is derived and used to obtain the velocity of the ion over time. The results shown in this chapter are covered in our recent publication [de Clercq 15b].

7.1 Hamiltonian estimation method

Here we propose and demonstrate a method for reconstructing a general time-dependent Hamiltonian with two non-commuting terms which drives the evolution of a single qubit. The method works with any single-qubit Hamiltonian $\hat{H} = \sum_i f_i(t)\hat{\sigma}_i$, where the $f_i(t)$ are arbitrary time-dependent functions and $\hat{\sigma}_i$ are the Pauli operators. In our experiments, a Hamiltonian with two non-commuting time-dependent terms arises when we try to perform quantum logic gates by transporting an ion through a static laser beam [D. Leibfried 07, de Clercq 15a]. In this case, the Hamiltonian describing the interaction between the ion and the laser can be written in an appropriate rotating frame as

$$\hat{H}_I(t) = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \delta(t)\hat{\sigma}_z\right) \tag{7.1}$$

which includes a time-varying Rabi frequency $\Omega(t)$, and an effective detuning $\delta(t)$ which is related to the first-order Doppler shift of the laser in the rest frame of the moving ion (see next section for details). For a Hamiltonian of this type with unspecified time-dependent coefficients, no analytical solution to Schrödinger's equation exists [Barnes 12, Barnes 13]. In order to reconstruct the Hamiltonian we make use of two additional features of our experiment. The first is that we can switch off the Hamiltonian at time t_{off} on a timescale which is fast compared to the evolution of the qubit. Secondly we are able to offset one of the terms in the Hamiltonian, in our case by adding a static detuning term $\hat{H}_{\text{s}} = \hbar\delta_L\hat{\sigma}_z/2$ such that the total Hamiltonian is $\hat{H}_I(t) + \hat{H}_s$. We then measure the expectation value of the qubit in the $\hat{\sigma}_z$ basis as a function of δ_L and t_{off}. Repeating the experiment with identical settings many times, we obtain an estimate of the expectation value which we denote as $\langle\hat{\sigma}_z^{\text{meas}}(t_{\text{off}}, \delta_L)\rangle$.

Hamiltonian extraction involves theoretically generating the qubit populations $\langle\hat{\sigma}_z^{\text{sim}}(t_{\text{off}}, \delta_L)\rangle$, and attempting to find the Hamiltonian for which this most closely matches the data. In order to provide a simple parameterization, we represent $\delta(t)$ and $\Omega(t)$ as a linear weighted combination of basis splines [Bartels 95, de Boor]. $\langle\hat{\sigma}_z^{\text{sim}}(t_{\text{off}}, \delta_L)\rangle$ is compared to the measured data using a weighted least-squares cost function, which we optimize with respect to the weights of the basis-splines used to parameterize $\delta(t)$ and $\Omega(t)$ (section C.1). Solving this optimization problem in general is hard because the cost function is subject to strong constraints imposed by quantum mechanics, producing a non-trivial relation between the weights and the spin populations.

We overcome this problem by making use of the inherent causality of the quantum-mechanical evolution, and by assuming that the parameters of the Hamiltonian vary smoothly. We call our technique "Extending the Horizon Estimation", in analogy to established methods in engineering [Muske 95] (a detailed description of our method can be found in section C.2). Rather than optimizing over the whole data set at once, we build up the solution by initially fitting the data over a limited region of time $0 < t_{\text{off}} < T_0$. The solution obtained over this first region can be extrapolated over a larger time span $0 < t < T_1$ where $T_1 = T_0 + \tau$, which we use as a starting point to find an optimal solution for this extended region. This procedure is iterated until $T_{n_{\max}} = \max(t_{\text{off}})$. The method allows us to choose a reduced number of basis spline functions to represent $\delta(t)$ and $\Omega(t)$, and also reduces the amount of data considered in the early stages of the fit, when the least is known about the parameters. This facilitates the use of non-linear minimization routines, which are based on local linearization of the problem and converge faster near the optimum. More details regarding the optimization routine can be found in section C.2.

7.2 Transport dynamics

The interaction of a laser beam with frequency ω_L and wave vector $\vec{k}(\vec{z}(t))$ with a two-level atom with resonant frequency ω_0 and time-dependent position of the ion $\vec{z}(t) = (0, 0, z(t))$ can be described in the Schrödinger picture by the Hamiltonian

$$\hat{H}_S = -\frac{\hbar\omega_0}{2}\hat{\sigma}_z - \hbar\Omega(z(t))\cos\left(\vec{k}(\vec{z}(t))\cdot\vec{z}(t) - \omega_L t\right)\hat{\sigma}_x\,, \tag{7.2}$$

where the Rabi frequency $\Omega(z(t))$ gives the interaction strength between the laser and the two atomic levels. We can define the laser phase at the position of the ion as $\Phi(t) = \phi(t) - \omega_L t$ with $\phi(t) = \vec{k}(\vec{z}(t))\cdot\vec{z}(t) = k_z(z(t))z(t)$ and $k_z(z(t)) = |\vec{k}|\cos(\theta(t))$ being the projection of the laser beam onto the z-axis along which the ion is transported. Here $\theta(t)$ is the angle between the wave-vector $\vec{k}(z(t))$ and the transport axis evaluated at position $z(t)$. Moving to a rotating frame using the unitary transformation $U = e^{-i\frac{\Phi(t)}{2}}$ we find

$$\begin{aligned}\hat{H}_I = \frac{\hbar}{2}\left(-\omega_0 - \dot{\Phi}(t)\right)\hat{\sigma}_z &- \frac{\hbar\Omega(t)}{2}\left(e^{+2i\Phi(t)} + 1\right)\hat{\sigma}_+ \\ &- \frac{\hbar\Omega(t)}{2}\left(1 + e^{-2i\Phi(t)}\right)\hat{\sigma}_-\,.\end{aligned} \tag{7.3}$$

Applying the RWA (Rotating Wave Approximation) with respect to optical frequencies, we find that

$$\hat{H}_I = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \left(-\omega_0 - \dot{\Phi}(t)\right)\hat{\sigma}_z\right). \tag{7.4}$$

Defining a static laser detuning $\delta_L = \omega_L - \omega_0$ we obtain

$$\hat{H}_I = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \left(\delta_L - \dot{\phi}(t)\right)\hat{\sigma}_z\right). \tag{7.5}$$

Defining an instantaneous detuning as

$$\delta(t) = \delta_L - \dot{\phi}(t) \tag{7.6}$$

and substituting it into Eq. 7.5 results in

$$H_I = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \delta(t)\hat{\sigma}_z\right). \tag{7.7}$$

If the wave vector of the beam were to be constant the $\phi(t) = \vec{k}\cdot\dot{\vec{z}}(t)$ the familiar expression for the first order Doppler shift and $\dot{\vec{z}}(t)$ the velocity of the ion. Furthermore if we detune the laser such that we cancel this Doppler shift we obtain

$$H_I = -\frac{\hbar}{2}\Omega(t)\hat{\sigma}_x. \tag{7.8}$$

This is the time-dependent case for Rabi oscillations.

7.2.1 Example dynamics

I will present a few cases which are relevant to the experimental observations we have made. The first case will be for ideal transport gates which should be performed at constant velocity [D. Leibfried 07]. The other cases will deal with non constant velocities where there are some time varying profiles we suspected to arise experimentally and are shown in Fig. 7.1.

In the examples dynamics a Gaussian beam with a waist of $W_0 = 50\,\mu\text{m}$ will be used and the intensity chosen such that it gives rise to a peak Rabi frequency of $\Omega_{\max}/2\pi$=50 kHz. The transport is simulated as an ion starting from rest 500 µm away from the beam and then shuttled at a velocity $v = 4.4311\,\text{m/s}$ through the laser beam making such that the angle between the laser wave vector $\vec{k}$ and the velocity $\vec{v}$ is $3\pi/2$. Fig. 7.2 shows the both the velocity and the Rabi frequency as a function of time. The time axis starts at a later time which indicates that it takes the ion some time to reach the beam. In Fig. 7.3a the probability for the ion to be in the excited state given a laser detuning and time the ion has spent in the beam is illustrated as a 2D example data set $\langle\hat{\sigma}_z\left(t_{\text{off}}, \delta_L\right)\rangle$. Considering an exert from the data for a given detuning $\delta_L = 0$ over all time, as shown by the line in Fig. 7.3a, we obtain the on resonance behavior as shown in Fig.7.3b.

The angle of rotation of the state as a function of time for the case of on resonance $\delta_L = 0$ and constant velocity is given by integrating the $\hat{\sigma}_x$ term in

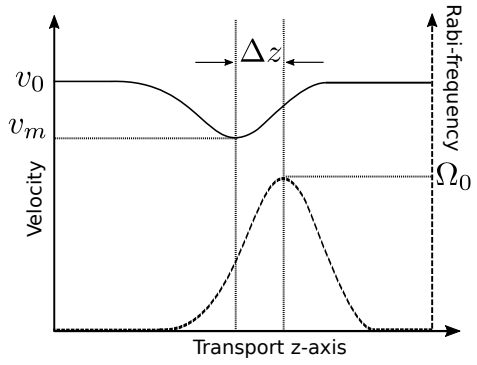


Figure 7.1: Representation of the essential physics that is involved with transport gates.

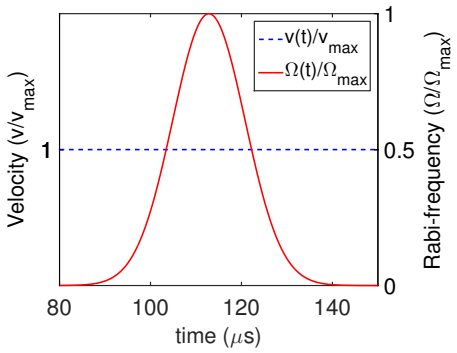
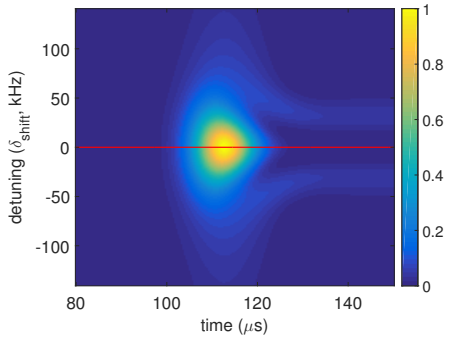
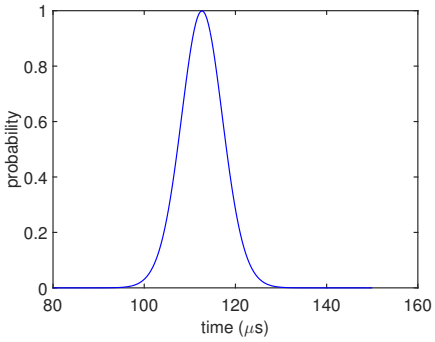


Figure 7.2: The velocity and Rabi frequency.



(a) Simulated dynamics.



(b) Time trace for the line cut-out of the dynamics.

Figure 7.3: The resulting dynamics coming from the simulation and a single trace taken from the dynamics as indicated by the red line.

Eq. 7.8 over $(-\infty, t]$. This is equivalent to stating that we should start far away enough from the beam when transporting such that we do not drive any dynamics initially. The rotation angle of the state at time t is then

$$\phi(t) = \frac{\hbar}{2} \int_{-\infty}^{t} \Omega(t')dt'. \tag{7.9}$$

The time-dependent Rabi frequency here is given by

$$\Omega(t) = \Omega_{\max} e^{-2\left(\frac{vt\sin(\alpha)}{W_0}\right)^2}. \tag{7.10}$$

The exponential term reflects the fact that the transport is carried out at an

angle α with respect to the beam. Performing the integration in Eq.7.9 gives

$$\phi(t) = \sqrt{\frac{\pi}{2}} \frac{W_0 \Omega_{\max}}{2v \sin(\alpha)} \left(1 + \operatorname{erf}\left(\frac{\sqrt{2} v \sin(\alpha)\, t}{W_0}\right)\right). \tag{7.11}$$

Extending the time to infinity $t \to \infty$ is equivalent to the gate operations being finished and the final rotation angle is then given by [D. Leibfried 07]

$$\phi_{\text{end}} = \sqrt{\frac{\pi}{2}} \frac{W_0 \Omega_{\max}}{v \sin(\alpha)}. \tag{7.12}$$

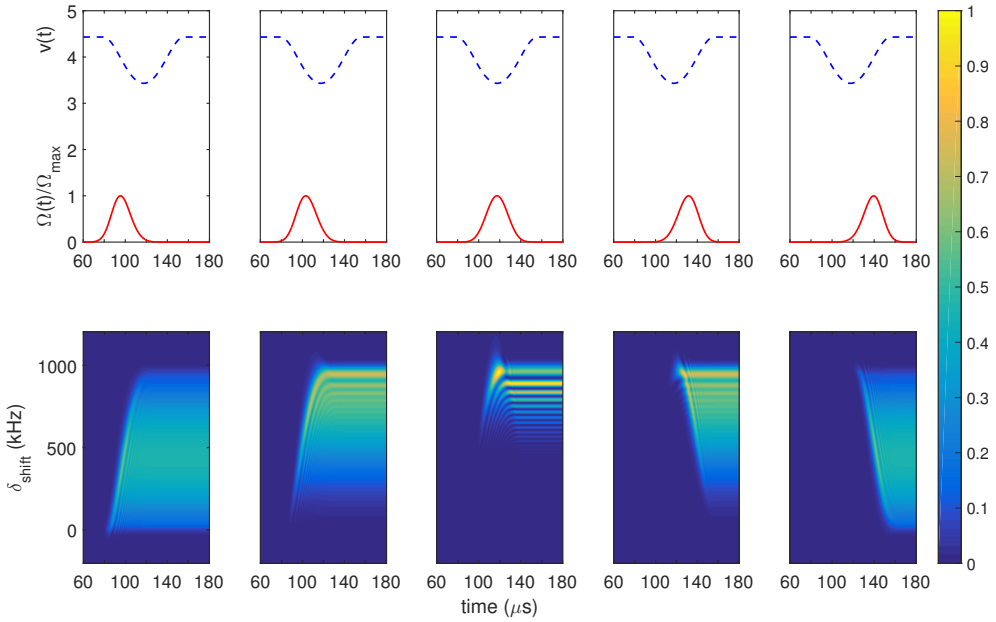
For the beam waist, the angle the ion is transported through the beam and maximum Rabi frequency for the laser beam in the example shown, the velocity required to perform a 2π rotation is $v = 4.4311\,\text{m/s}$ using Eq. 7.12.

The possibility that the velocity varies during the transport adds an additional complication. To illustrate these effects, in Fig. 7.1 I have plotted simulated results from a series of example cases in which the velocity makes a deviation from a fixed value which varies in time as a Gaussian. Note that it is not necessarily true that this Gaussian velocity profile has its peak at the same time as the intensity profile. The parameters used are given in the figure captions of Fig. 7.4. From Fig. 7.4a where the beam moves from the left to the right it samples less of the velocity profile and the off resonant driven dynamics becomes minimum when the beam is aligned with the velocity minimum. When the beam moves through the velocity profile to the right, the dynamics which results are slightly different due to a different velocity being sampled. In Fig. 7.4b the maximum of the beam intensity and hence the maximum of the Rabi frequency is aligned and only the velocity profile width is changed. As can be seen from an increasing FWHM of the velocity profile the dynamics over a broad range of detunings are driven initially and becomes less spread out as the velocity profile is broadened with respect to the beam width. When the profile is very broad compare to the beam's spread the velocity seems almost constant and shifted with respect to the velocity being sampled by the beam.

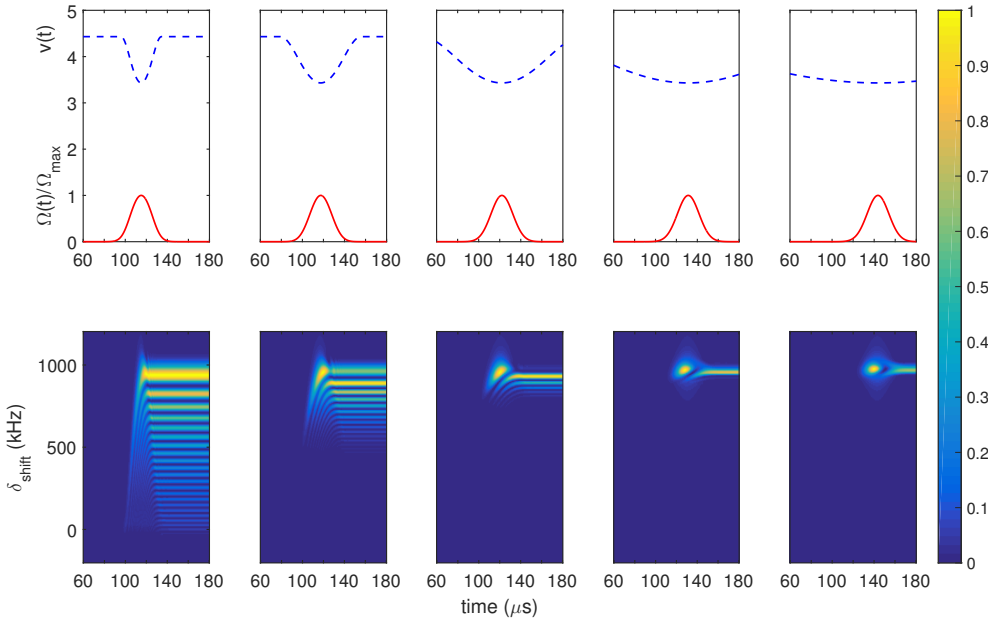
In practice, due to experimental deviations from the ideal case, both the intensity and velocity profiles are unlikely to be Gaussian. This requires methods for extracting these time-dependent parameters with no assumptions about their form.

7.3 Experimental sequence

In the experimental work, we demonstrate reconstruction of the spin Hamiltonian for an ion transported through a near-resonant laser beam. Our qubit is encoded in the electronic states of a trapped calcium ion, which is defined by $|0\rangle \equiv \left|{}^2S_{1/2}, M_J = 1/2\right\rangle$ and $|1\rangle \equiv \left|{}^2D_{5/2}, M_J = 3/2\right\rangle$. This transition is well resolved from all other transitions, and has an optical frequency



(a) The velocity profile does not change, but the beam positioning changes from the left to right as -80 µm, -50 µm, 0 µm, +50 µm and +80 µm with respect to the velocity profile minimum.



(b) The beam maximum is aligned to the velocity minimum. Form left to right, the velocity's profile FWHM is changed according to 75 µm,150 µm,300 µm,600 µm and 1000 µm.

Figure 7.4: Changes in dynamics due to varying the beam position and the velocity profile width. Maximum Rabi frequency of 50 kHz and beam waist of 50 µm is use. Ion's peak velocity is $v = 4.33$m/s and minimum $v = 3.33$m/s.

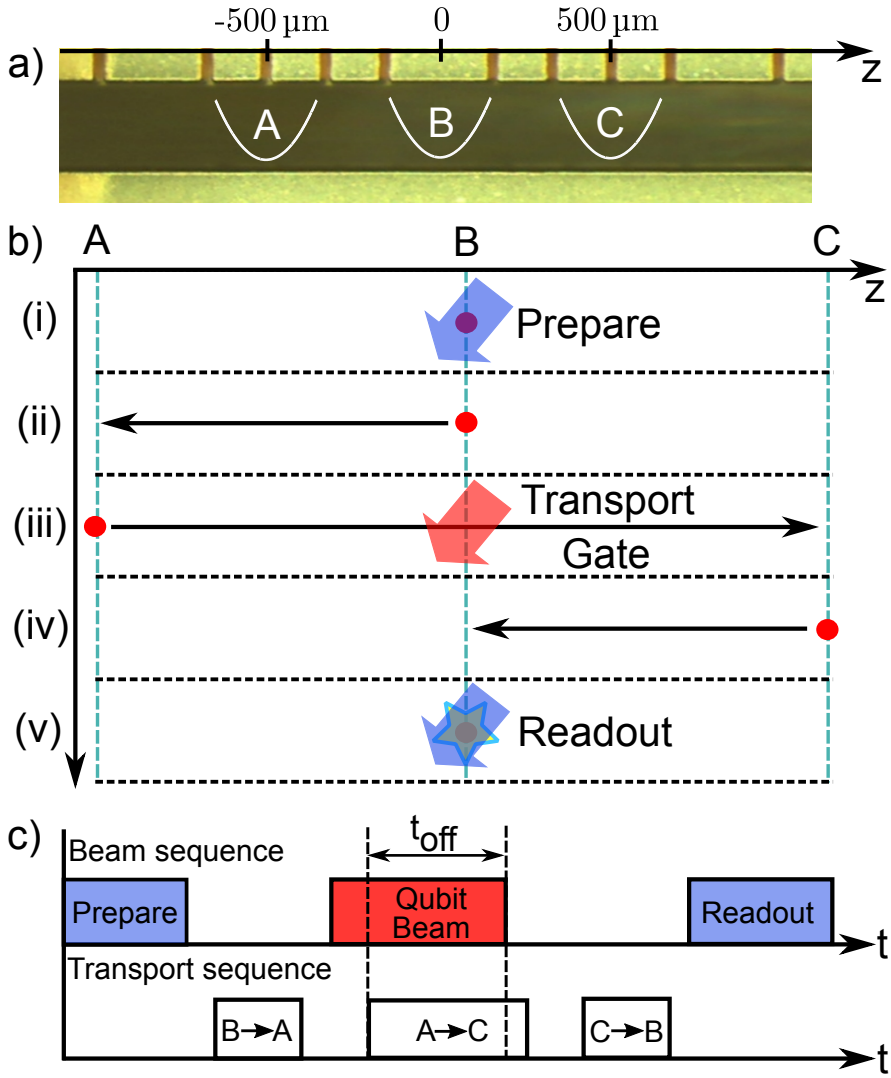


Figure 7.5: Experimental sequence and timing: a) The experiment is carried out in three zones of the trap indicated by A, B and C. b) The experimental sequence involves steps (i) through (v). Preparation and readout are carried out on the static ion in zone B. The qubit evolves while the ion is transported from zone A to zone C, via the laser beam in zone B. c) Experimental sequence showing the timing of applied laser beams and ion transport, including shutting off the laser beam during transport.

$\omega_0/(2\pi) \simeq 411.0420$ THz. The laser beam points at 45 degrees to the transport axis, and has an approximately Gaussian spatial intensity distribution. The time-dependent velocity $\dot{z}(t)$ of the ion is controlled by adiabatic translation of the potential well in which the ion is trapped. This is implemented by applying time-varying potentials to multiple electrodes of a segmented ion trap, which are generated using a multi-channel arbitrary waveform generator, each output of which is connected to a pair of electrodes via a passive third order low-pass Butterworth filter. The result is that the ion experiences a time-varying Rabi frequency $\Omega(t)$ and a laser phase which varies with time as $\Phi(t) = \phi(z(t)) - \omega_L t$, where $\phi(z(t)) = k_z(z(t))z(t)$ with $k_z(z(t))$ the laser wavevector projected onto the transport axis at position $z(t)$ and ω_L the laser frequency. The spatial variation of $k_z(z(t))$ accounts for the curvature of the wavefronts of the Gaussian laser beam. In order to create a Hamiltonian of the form given in Eq. 7.1 we work with the differential of the phase, which gives a detuning $\delta(t) = \delta_L - \dot{\phi} = \delta_L - (k_z'(z)z + k_z(z))\,\dot{z}$ with $\delta_L = \omega_L - \omega_0$ the laser detuning from resonance. For planar wavefronts $k_z'(z) = 0$, and $\delta(t)$ corresponds to the familiar expression for the first-order Doppler shift (see section C.3 details).

The experimental sequence is depicted in Fig. 7.5. We start by cooling all motional modes of the ion to $\bar{n} < 3$ using a combination of Doppler and electromagnetically-induced-transparency cooling [Roos 00], and then initialize the internal state by optical pumping into $|0\rangle$. The ion is then transported to zone A, and the laser beam used to implement the Hamiltonian is turned on in zone B. The ion is then transported through this laser beam to zone C. During the passage through the laser beam, we rapidly turn the beam off at time t_{off} and thus stop the qubit dynamics. The ion is then returned to the central zone B in order to perform state readout, which measures the qubit in the computational basis. The additional Hamiltonian $\hat{H}_{\text{s}}$ is implemented by offsetting the laser frequency used in the experiment by a detuning δ_L. For each setting of t_{off} and δ_L the experiment is repeated 100 times, allowing us to obtain an estimate for the qubit populations $\langle \hat{\sigma}_z^{\text{meas}}(t_{\text{off}}, \delta_L)\rangle$.

7.4 Results

Experimental data is shown in Fig. 7.6 for two different beam positions, alongside the results of fitting performed using our iterative method. The beam positions used for each data set differ by around 64 µm, but the transport waveform used was identical. The reconstructed velocities should therefore agree in the region where the data overlap. It can be seen from the residuals that the estimation is able to find a Hamiltonian which results in a close match to the data. In order to get an estimate of the relevant error bars for our reconstruction, we have performed non-parametric resampling with replacement, optimizing for the solution using the same set of basis spline

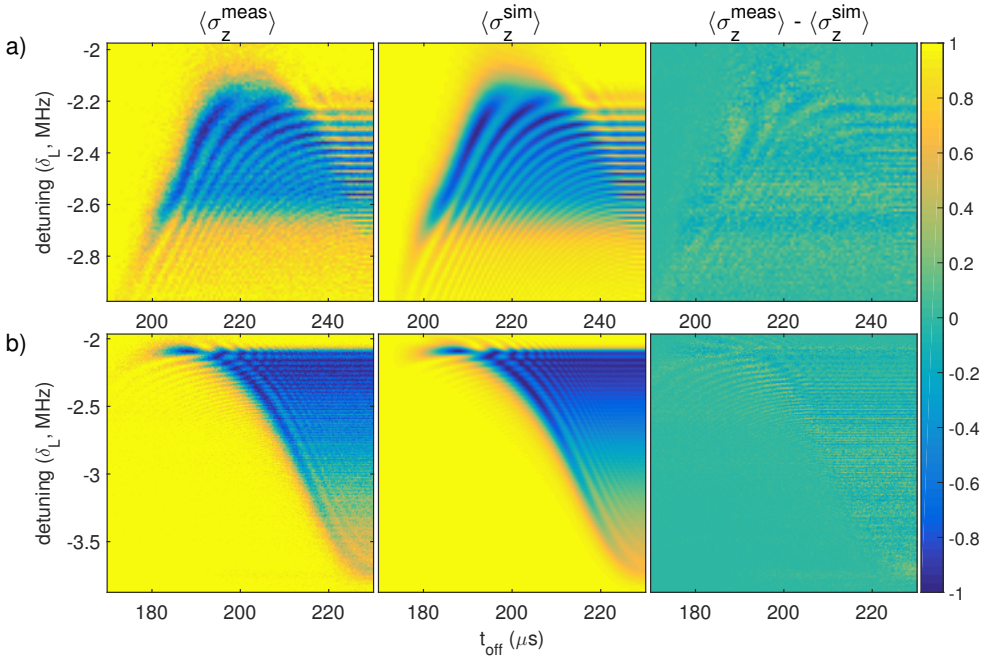


Figure 7.6: Spin population as a function of detuning and switch-off time of the laser beam. a) is for a laser beam centered in zone B, while for b) the beam was displaced towards zone C by 64 μm. From left to right are plotted the experimental data, the populations generated from the best fit Hamiltonian, and the residuals. Each data point results from 100 repetitions of the experimental sequence. The data in a) consist of an array of 100×101 experimental settings, while that shown in b) consists of an array of 201×201 settings. This leads to smaller error bars in the reconstructed Hamiltonian for the latter. For the Hamiltonian estimation the data was weighted according to quantum projection noise.

functions as was used for the experimental data to provide a new estimate for the Hamiltonian. This is repeated for a large number of samples, resulting in a distribution for the estimated values of $\delta(t)$ and $\Omega(t)$ from which we extract statistical properties such as the standard error. The error bounds shown in figures 7.7 correspond to the standard error on the mean obtained from these distributions (see section C.4 for further details).

The estimated coefficients of the Hamiltonian extracted from the two data sets are shown in Fig. 7.7a and 7.7b. It can be seen that the values of $\delta(t)$ for the two different beam positions differ for the region where the reconstructions overlap. We think that this effect arises from the non-planar wavefronts of the laser beam. Inverting the expression for $\delta(t)$ to obtain the velocity of the ion, we find $\dot{z}(t) = (\delta_L - \delta(t))/(k_z'(z)z + k_z(z))$. Using this correction, we find that the two velocity profiles agree if we assume that the ion passes through

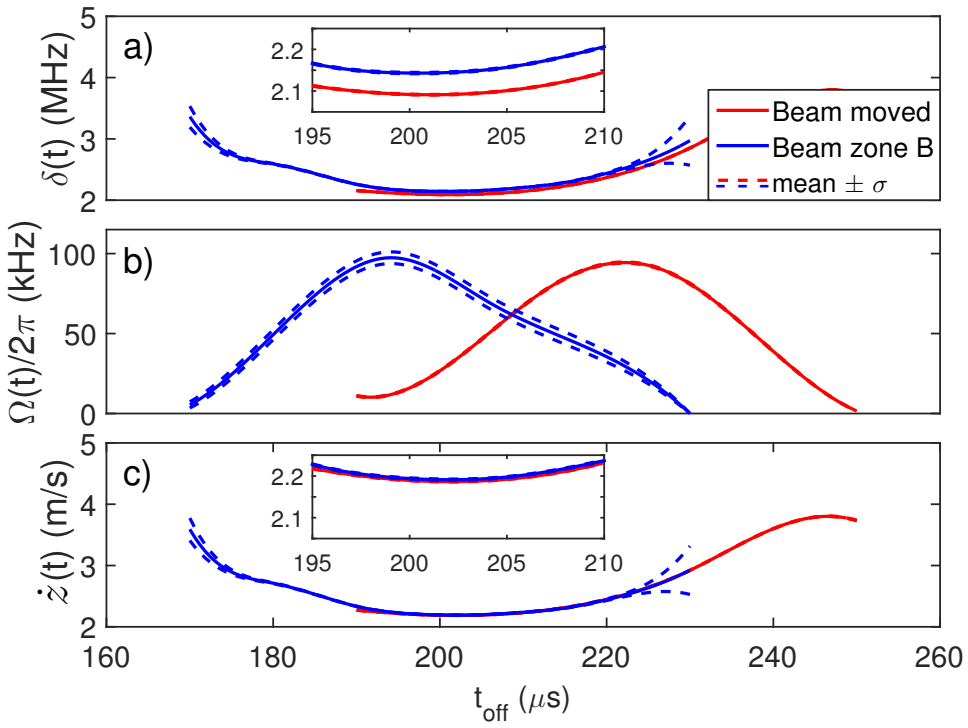


Figure 7.7: Estimates of time-dependent co-efficients: a) The effective detuning $\delta(t)$ and b) Rabi frequency $\Omega(t)$ obtained from the two data sets, along with dashed lines indicating the standard error on the mean of these estimates obtained from resampling. For a), the inset shows a close up of the estimated $\delta(t)$ in the regions where the estimates overlap, showing that these do not give the same value. c) The estimated velocity $\dot{z}(t)$ of the ion obtained after applying wavefront correction. The inset shows that this can produce consistent results.

the center of the beam at a distance of 2.27 mm before the minimum beam waist, a value which is consistent with experimental uncertainties due to beam propagation and possible mis-positioning of the ion trap with respect to the fixed final focusing lens. The velocity estimates taking account of this effect are shown in Fig. 7.7c.

Fig. 7.8 shows the results of a reconstruction for a second pair of data sets taken using two different velocity profiles but with a common beam position. The resolution in both time and detuning were lower in this case than for the data shown in Fig. 7.6. We observe that the estimated Rabi frequency profiles agree to within the error bars of the reconstruction. One interesting feature of this plot is that the error bars produced from the resampled data sets are notably higher at the peak than on the sides of the beam. We think

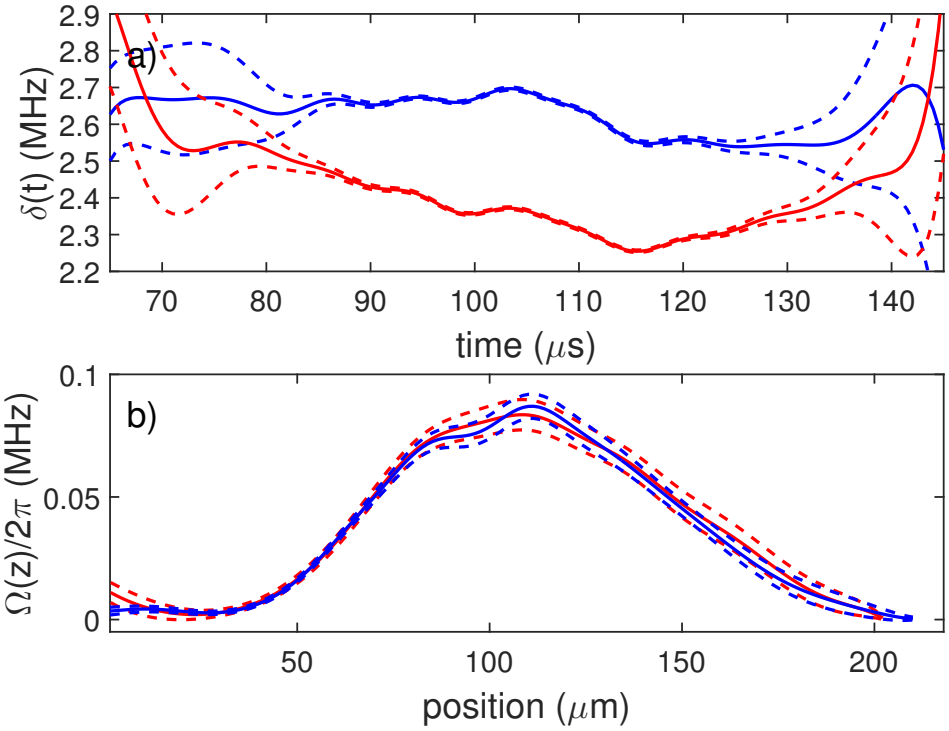


Figure 7.8: Spatial Rabi frequency: a) The estimated $\delta(t)$ obtained from the second pair of data sets (Fig. 7.9). b) The estimated Rabi frequency $\Omega(t)$ for the same two data sets.

that this happens because the sampling time of the data is 0.5 µs, which is not high enough to accurately resolve the fast population dynamics resulting from the high Rabi frequency (the Nyquist frequency is 1 MHz). In order to optimize the efficiency of our method, it would be advantageous to run the reconstruction method in parallel with data taking, thus allowing updating of the sampling time and frequency resolution of points based on the current estimates of parameter values.

7.4.1 Single beam profile with two different velocity profiles

As a check that our method is also able to produce consistent results for the Rabi frequency profile, we measured a second pair of data sets in which we take two different velocity profiles using the same beam position. This data is shown in Fig. 7.9. Also shown are the best-fits obtained from the reconstructed Hamiltonians. The parameter variations obtained from the reconstructed Hamiltonians for these data sets can be found in the main text in Fig. 7.8. The sampling rate of the data in these data sets was 2 MHz, resulting in a Nyquist frequency of 1 MHz.

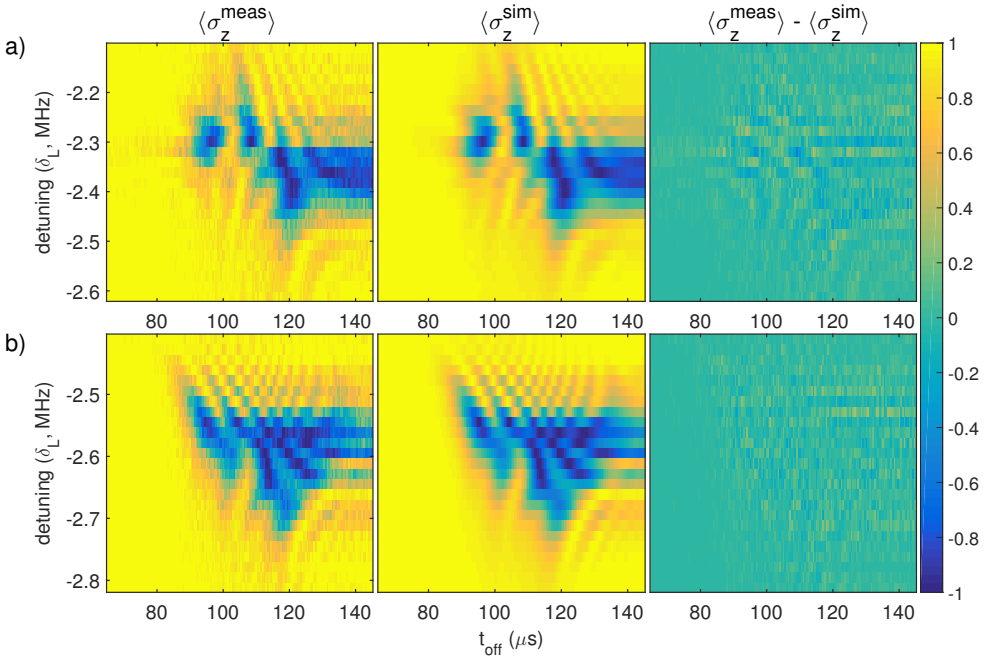


Figure 7.9: Spin population as a function of detuning and switch-off time of the laser beam, for the data sets used to obtain the reconstructed parameters shown in Fig. 7.8. a) uses a velocity profile with only small variations. b) A second data run in which large variations in the velocity profile were used. Each data point results from 100 repetitions of the experimental sequence. For the Hamiltonian estimation the data was weighted according to quantum projection noise.

7.5 Conclusion

Our method for directly obtaining a non-commuting time-dependent Hamiltonian uses straightforward measurements of the qubit state in a fixed basis as a function of time and a controlled offset to the Hamiltonian. This simplicity means that the method should be applicable in a wide range of physical systems where such control is available, including many technologies considered for quantum computation [Devitt 06, Shabani 11, Zhang 14, Barends 15, Bonato 15]. A process-tomography based approach would require that for every time step multiple input states be introduced, and a measurement made in multiple bases [Chuang 97, Poyatos 97, Riebe 06]. An effective modulation of the measurement basis arises in our approach due to the additional detuning δ_L. It is worth noting that tomography provides more information than our method: it makes no assumptions about the dynamics aside from that of a completely positive map while we require coherent dynamics. Extensions to our work are required in order to provide a rigorous estimation of the ef-

ficiency of the method in terms of the precision obtained for a given number of measurements, and to see whether a similar approach could be taken to non-unitary dynamics. Using this method on considerably lower resolution data sets, we have recently been able to improve the control over the velocity, which will be necessary in order to realize multi-qubit transport gates in our current setup [D. Leibfried 07].

Chapter 8

Parallel transport logic gates

Quantum information processing will require combinations of gate operations and communication, with each applied in parallel to large numbers of quantum systems [Kielpinski 02, Steane 07]. These tasks are often performed sequentially, with gates implemented by pulsed fields and information transported either by moving the physical qubits or using photonic links [Home 09, Moehring 07]. For trapped ions, an alternative approach is to implement quantum logic gates by transporting the ions through static laser beams, combining qubit operations with transport [D. Leibfried 07]. This has significant advantages for scalability since the voltage waveforms required for transport can potentially be generated using micro-electronics integrated into the trap structure itself, while both optical and microwave control elements are significantly more bulky. Using a multi-zone ion trap, we demonstrate transport gates on a qubit encoded in the hyperfine structure of a beryllium ion. We show the ability to perform sequences of operations, and to perform parallel gates on two ions transported through separate trap locations using a single recycled laser beam. For the latter, we demonstrate independent quantum gates by controlling the speed of each of the ions. This work provides a scalable path to ion trap quantum computing without a dramatic increase in optical control complexity. The results shown in this chapter are covered in our recent publication [de Clercq 15a].

8.1 General experimental implementation

We work with qubits stored in the hyperfine levels of the ground state of beryllium ions. Working at a magnetic field of 0.011945 T, the $\left|\mathrm{S}_{1/2}, F=1, m_F=1\right\rangle \leftrightarrow \left|\mathrm{S}_{1/2}, F=2, m_F=0\right\rangle$ transition energy is first-order independent on the magnetic field, making this an excellent memory qubit which is robust against spatial magnetic field variations during transport [Langer 05a]. In order to initialize the ion in one of these levels, we start by optically pumping to

$\left|S_{1/2}, F = 2, m_F = 2\right\rangle$ followed by a resonant stimulated Raman transition pulse which coherently transfers the population to $\left|S_{1/2}, F = 1, m_F = 1\right\rangle$ (referred as $|\uparrow\rangle$ in the text). All the coherent operations used in this experiment are performed using co-propagating Raman beams, using a laser red-detuned by $\Delta_R/(2\pi) \approx 230$ GHz from the $^2S_{1/2} \leftrightarrow ^2P_{1/2}$ transition.For the transport gates, we shuttle the ion between two different regions of the trap, passing through the laser beam during the transport.

Following the quantum logic gate, we read out the qubit state by first performing two coherent Raman transfer pulses, the first of which shelves the population from state $\left|S_{1/2}, F = 2, m_F = 0\right\rangle$ to $\left|S_{1/2}, F = 1, m_F = -1\right\rangle$, while the second transfers the population from $\left|S_{1/2}, F = 1, m_F = 1\right\rangle$ to $\left|S_{1/2}, F = 2, m_F = 2\right\rangle$. Subsequent readout of the quantum state is performed using state-dependent fluorescence [Wineland 98].

We transport ions by applying time-dependent potentials to the trap electrodes, which are generated using a home-built 16-channel Arbitrary Waveform Generator (AWG), the DEATHs (see chapter 5). We calculate the required waveforms using a constrained optimization method, allowing control of multiple potential wells with constrained position, curvature and depth simultaneously.

8.2 Rabi oscillations and Ramsey experiment

For initial experiments with a single ion, we prepare and read out the state in zone B, as depicted in Fig. 8.1. The transport gate is implemented by first transporting the ion to zone S, which is outside the intensity profile of the laser beams used to perform the logic gate. We then turn on the gate Raman beam pair, which cross the trap axis near the center of zone B, with the beams making an angle of 45 degrees to the trap axis. We measure the beam in independent experiments to have a Gaussian Rabi frequency profile along the axis with a waist of 40±3 µm. After the Raman beam pair has been turned on, we transport the ion from zone S to zone E, passing through zone B. All transport sequences are adiabatic, and thus the ion position closely follows the potential well minimum [Bowler 12, Walther 12]. We aim to achieve a constant velocity during transport, but have observed in separate experiments with calcium ions that this is not achieved due to an interplay between the outputs of our AWG and passive noise filters which connect these to the ion trap [de Clercq 15b]. Nevertheless, by tuning the velocity we are able to control the gate operation performed on the ions.

The Rabi oscillations shown in Fig. 8.2 are obtained by recording the population of $|\uparrow\rangle$ as a function of the transit time through the beam. This is achieved by switching off the laser beams rapidly during the transport. For an ion transported at a constant velocity v through a Gaussian beam profile with waist

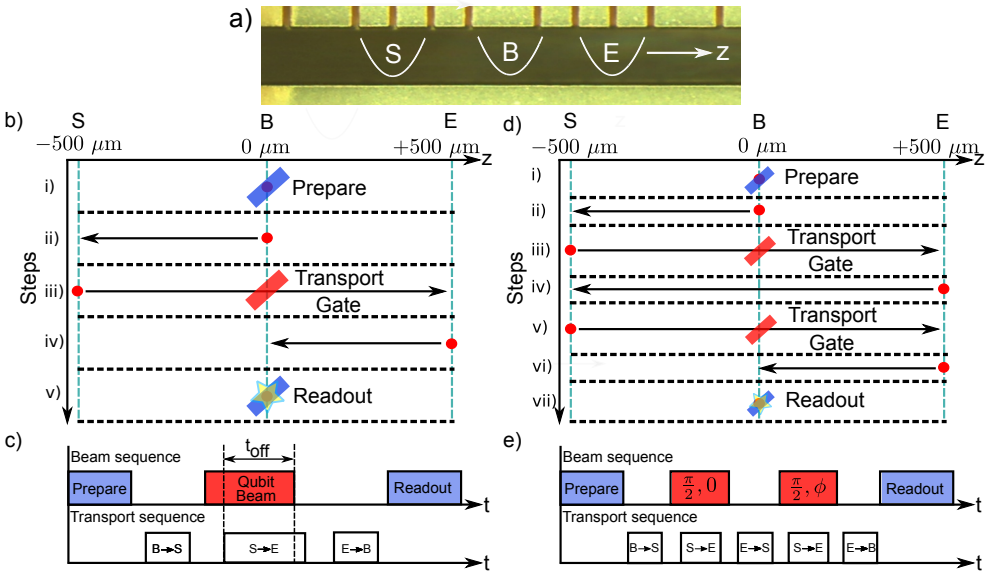


Figure 8.1: a) Image of the different trap zones along the transport axis z. b) The transport sequence for single ion transport Rabi oscillations. The experiment is carried out as a sequence of operations which occur in numerical order from i) to v). c) The pulse scheme used to probe Rabi oscillations. The qubit beams are switched on before the start of the transport gate and rapidly switched off at a time t_{off} during the transport in order to probe the dynamics of the spin. d) The sequence used for the Ramsey sequence, using a similar notation as in b). e) The pulse sequence used for the Ramsey experiment. We concatenate two transport $\pi/2$ gates in which the phase of the laser is shifted by ϕ relative to the first. In this case the qubit beams remain on until the ion has passed through to zone E.

w_0 and a peak Raman transition Rabi frequency Ω_0, the probability to find the ion in the $|\uparrow\rangle$ state as a function of time is given by [D. Leibfried 07]

$$P(\uparrow, t) = \cos^2(\zeta(t)/2),\ \zeta(t) = \frac{\Omega_0}{\chi}\sqrt{\pi}\ \text{erf}(\chi t) \tag{8.1}$$

with $\chi = \sqrt{2}v/2w_0$. We fit the data with a form similar to equation 8.1 but including offset and amplitude parameters which account for imperfections in state preparation, readout and the Rabi oscillation itself. We obtain $\Omega_0/(2\pi) = 5.669 \pm 0.001$ kHz and $\chi = 10965 \pm 32$ s^{-1}. Using the relationship between velocity v and beam waist ω_0 given in equation 8.1, we use the beam waist measurement to deduce a transport velocity of 0.62 ± 0.05 ms^{-1}. To further improve the quality of the fit, equation 8.1 could be modified to take into account changes in the ion velocity during transport, and the deviations of the intensity of the laser beams from an ideal Gaussian [de Clercq 15b].

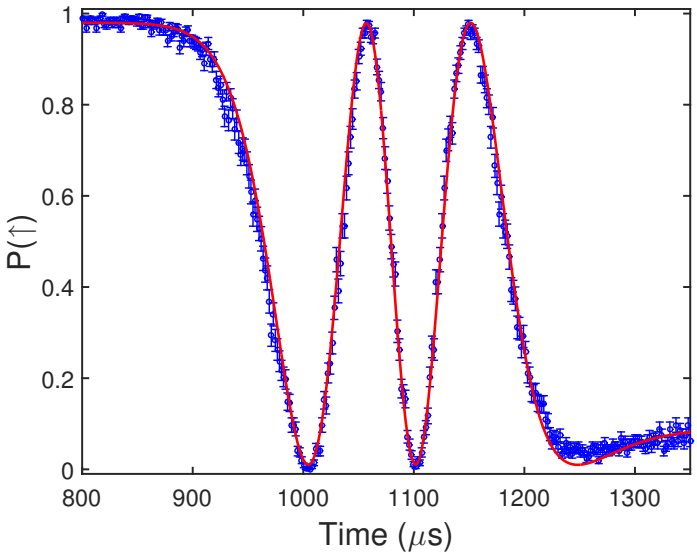


Figure 8.2: Rabi oscillations as an ion is transported through a two-frequency Raman laser beam pair. The x-axis is the time at which the beam was rapidly switched off during transport. The fitted curve follows the theoretical prediction from equation 8.1, which assumes a Gaussian beam profile. Discrepancies between the form of the fit and the data likely result from velocity deviations or deviations between the experimental beam shape and a Gaussian profile [de Clercq 15b]. Each datapoint represents the average of 350 repeats of the experimental sequence, with error bars estimated assuming quantum projection noise.

Having verified that the dynamics are close to those expected, we calibrate the laser intensity and the ion velocity in order to apply a single qubit rotation $\hat{R}(\pi/2)$, where we use a definition $\hat{R}(\theta) \equiv \cos(\theta/2)\hat{I} + \sin(\theta/2)\hat{\sigma}_x$. To perform this operation we speed up the ions to a target velocity of 7 ms^{-1}. Using two sequential transports (Fig. 8.1d and e), we perform a sequence of two gates on the ion, implementing a Ramsey experiment. Both gates produce the rotation $\hat{R}(\pi/2)$, however we shift the relative phase of the two Raman beams before applying the second transport gate. A scan of this phase is shown in Fig. 8.3, which we fit with a sinusoid to obtain an amplitude of 0.985 ± 0.004. This is consistent with the amplitude we observe for Rabi oscillations performed with a static ion and pulsed laser fields.

8.3 Parallel transport quantum logic gates

Fig. 8.4 shows a second set of experiments in which we work with two ions in parallel. In order to do this, we shift the beams coming from the laser to pass

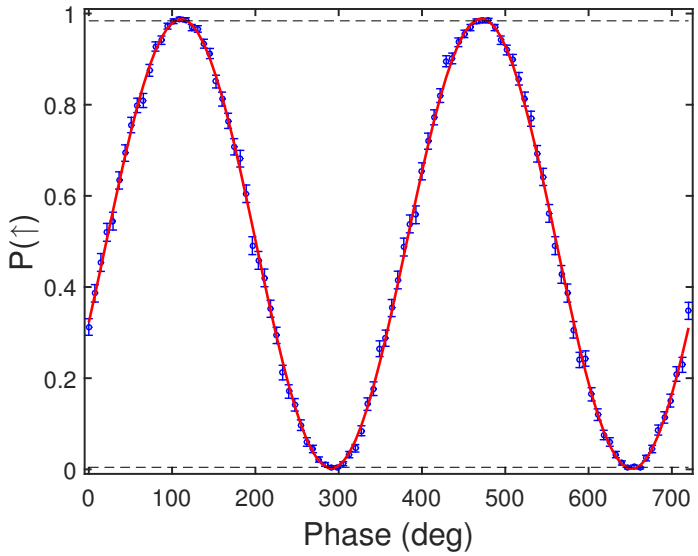


Figure 8.3: Experimental results for a Ramsey experiment performed using a sequence of two transport gates. The phase of the laser for the second pulse is scanned relative to the first using a shift in the relative phase of the two Raman beams applied using an acousto-optic modulator. The amplitude of the oscillations obtained from the fit is 0.985 ± 0.004. For comparison, we also plot horizontal dashed lines which indicate the maximum and minimum of Rabi oscillations performed using standard static-ion gates. Each datapoint is the average of 600 repeats of the experimental sequence, with error bars estimated assuming quantum projection noise.

through the trap in zone B1 (600 µm away from the trap center), and then reflect these laser fields back through the vacuum setup, with the reflected beams passing through a focus close to the axis of the trap in zone B2. The experimental sequence of operations is shown in Fig. 8.4. Preparation and measurement steps are performed sequentially for the two ions using the same methods as in the single ion experiments. For the transport gate operation, ion 1 is transported from zone S1 to zone E1 passing through the beams centered in B1, while ion 2 is transported from zone S2 to zone E2 through the beams centered in B2. Readout is then performed sequentially. Data for two different transport velocities of ion 2 are shown in Fig. 8.5, resulting in a $\hat{R}(\pi)$ or $\hat{R}(\pi/2)$ pulse being implemented. The transport velocity of ion 1 is 10.7 ms^{-1} and remains the same for both datasets. It is calibrated to produce a rotation $\hat{R}(\pi)$. For ion 2, the velocity is calibrated to either produce $\hat{R}(\pi)$ (requiring $v \approx 4.7$ ms^{-1}) or $\hat{R}(\pi/2)$ (requiring $v \approx 8.9$ ms^{-1}). We fit the data using simple error functions of the form $a + b\mathrm{erf}(s(t - t_c))$ where a is the offset, b the amplitude, s the steepness and t_c the time shift of the error

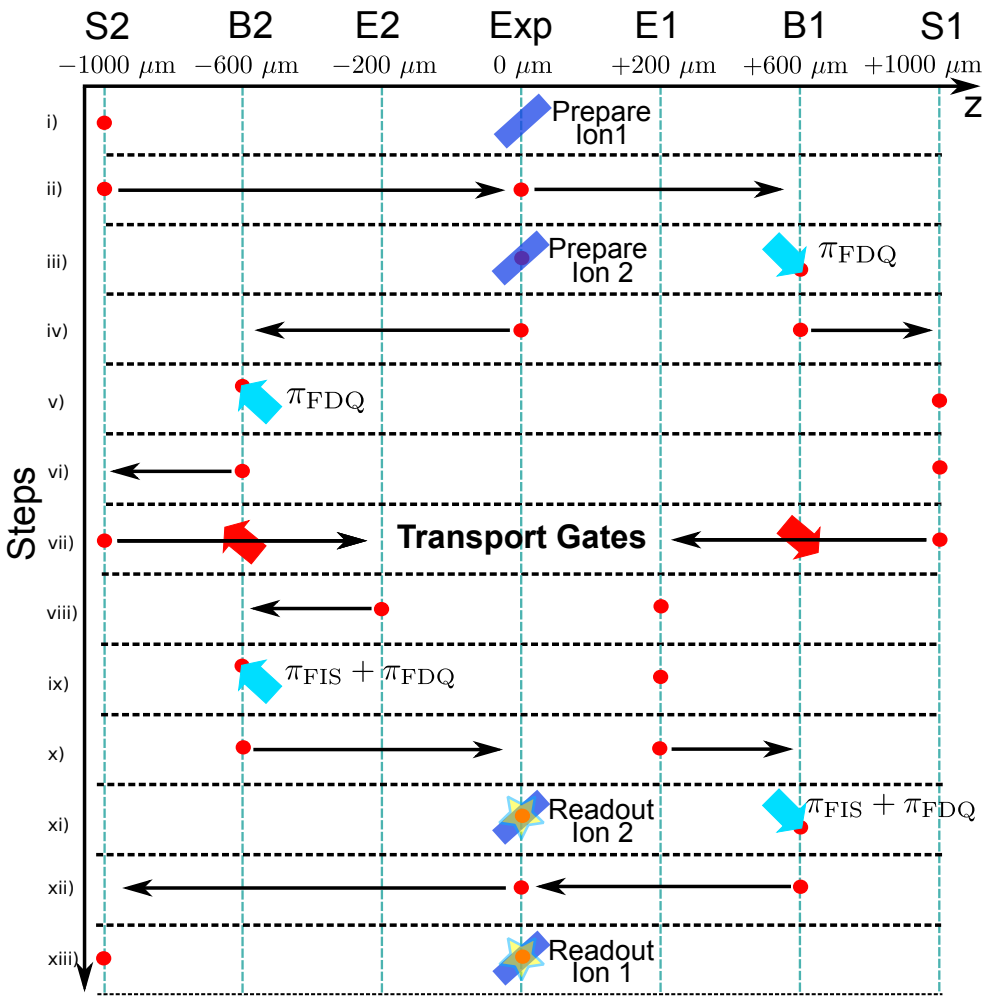


Figure 8.4: Sequence of operations performed for the demonstration of parallel transport gates on two ions. The steps i) through to xiii) are carried out sequentially. State preparation and readout is performed sequentially on the two ions in zones B1 and B2. The transport gates are applied simultaneously, by passing the ions through qubit beams located ± 600 μm from the trap center.

function. We obtain a visibility of 0.985 ± 0.002 for the $\hat{R}(\pi)$ rotations and a final population of 0.502 ± 0.004 for the $\hat{R}(\pi/2)$ rotation. These results therefore demonstrate the individual addressing capability in the transport gates while simultaneously performing operations with a single pair of Raman laser beams.

In the experiments performed above, the Raman laser beams are reflected back into the trap after passing out of the vacuum system. This produces two significant effects. Firstly there is a loss in optical power due to imperfect

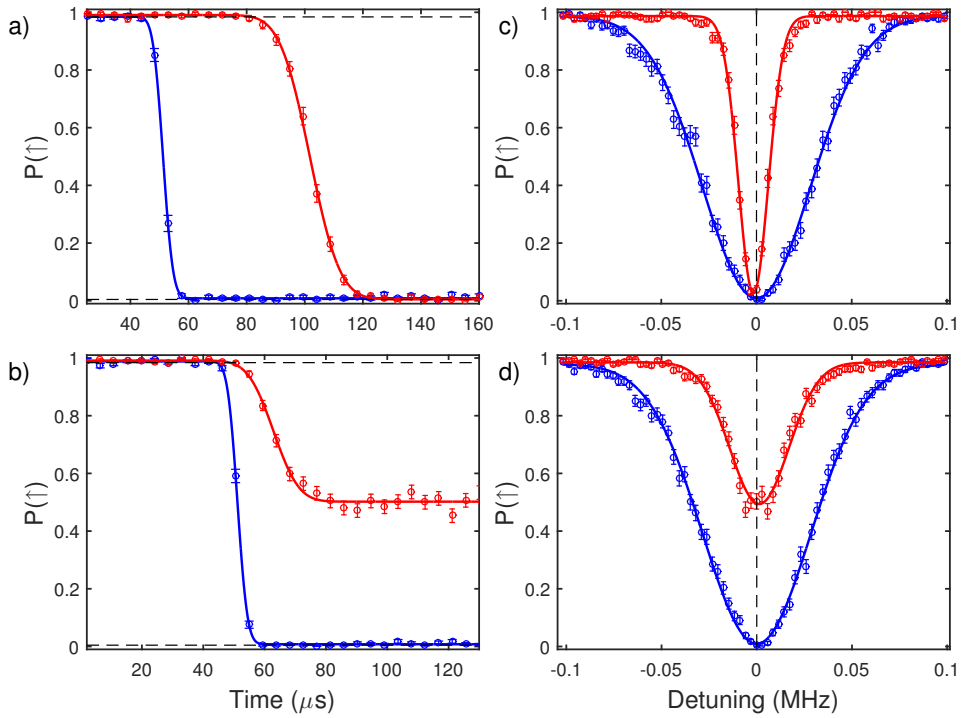


Figure 8.5: The figures on the left show time scans for the simultaneous rotation of ion 1 (blue) and ion 2 (red) with two different velocities for ion 2 resulting in a rotation $\hat{R}(\pi)$ (top) and $\hat{R}(\pi/2)$ (bottom). The velocity of ion 1 is kept constant at 10.68 ms^{-1} for the two scans while the velocity of ion 2 is increased from 4.68 ms^{-1} to 8.91 ms^{-1} to obtain a $\hat{R}(\pi/2)$ pulse. The data is fitted using error functions. We measure the visibility of the $\hat{R}(\pi)$ rotation to be 0.985 ± 0.002 while for the $\hat{R}(\pi/2)$ rotation we observe a final population of the upper state $P(\uparrow) = 0.502 \pm 0.004$. For comparison, we also plot horizontal dashed lines which indicate the maximum and minimum in Rabi oscillations performed using static-ion gates. The frequency scans on the right are taken after the transport gate has finished to avoid potential Doppler shifts. By fitting Gaussian functions to the data we measure the difference in the center frequency of the two ions to be 1.7 ± 0.3 kHz in c) and -1.3 ± 0.4 kHz in d). Vertical dashed lines indicate the centre of the fit to the B1 data. Each datapoint is the average of 250 repeats of the experimental sequence, with error bars estimated assuming quantum projection noise.

transmission of the vacuum windows and other optical components. Secondly, we observe an increase of the beam size in B2 relative to B1. Experimentally we measure the waist diameter of the beam in B1 to be 73 ± 2 µm and a peak Rabi frequency of $2\pi \times (47.9 \pm 0.5$ kHz) while the retro-reflected beams in B2 have a waist diameter of 160 ± 4 µm and a peak Rabi frequency of $2\pi \times (16.1 \pm 0.5$ kHz). The significant difference in intensities leads to a differential AC Stark shift between the two ions which we measure using static Rabi oscillations to be 1.3 ± 0.2 kHz, agreeing with the predicted value [Wineland 03]. By tuning the difference frequency of the Raman light fields close to the resonance of the less intense beams, the average operation fidelity in zone B1 (the location of the more intense beams) due to this residual detuning is $F = 0.9998$ [Pedersen 07]. Since both the Stark shift and the Rabi frequency scale proportionally, this error remains constant for faster gates. The differential AC Stark shift could be reduced by better equalization of the beam intensity, both through improved focussing and by the use of in-vacuum mirrors.

The use of co-propagating beams has the significant advantage that the difference wave vector is close to zero, which minimizes the first-order Doppler shift. In our current setup the two Raman beams are generated by passing each through a separate optical path, and combining them on a beamsplitter. We notice that even small misalignments lead to a differential Doppler shift which shifts the effective resonant frequencies of the two ions (a relative angle of 10^{-3} degrees gives a Doppler shift of around 400 Hz for a velocity of 10 ms^{-1}). We attempt to calibrate this out by performing two frequency scans with both ions simultaneously, where the transport direction of one of the ions is reversed between each scan. We then null out the corresponding change in the resonance frequency for this ion by changing the angle of one of the beams. However we note that the frequency scan data in Fig. 8.5 is slightly skewed, which we think might be due to residual Doppler shifts. The use of common optical fibers for both Raman beams would alleviate this problem [Gebert 14, Colombe 14].

8.4 Conclusion

The contrast of operations performed using transport gates is the same as for our standard pulsed static gates to within experimental error, which indicates that the transport itself does not producing a detectable additional error. In our current setup, the population transfer from $\left|\mathrm{S}_{1/2}, F = 2, m_F = 2\right\rangle$ to $\left|\mathrm{S}_{1/2}, F = 1, m_F = 1\right\rangle$ and vice versa produce the biggest source of error. From calculations using our experimental detection time, we estimate the static-ion transfer pulses to have a fidelity of around 99.2%. Better characterization of the transport gates independent of state preparation and measurement errors could be performed using randomized benchmarking [Knill 08, Gaebler 12].

Transport-based quantum logic gates provide significant advantages for scaling up ion trap quantum processors. Waveforms applied to the trap electrodes could in future be performed using in-vacuum electronics, and possibly integrated into the trap structure itself through the use of CMOS or other well established technologies [Mehta 14]. This would shift the control challenge from bulky optical setups to the well-developed field of micro-electronics, which has proven scalability. The use of recycled laser beams to address ions in many parts of a trap array would optimize the use of laser power, which is a precious resource since lasers are expensive and bulky.

Chapter 9

Summary and outlook

The QCCD architecture as proposed D. Wineland and D. Kielpinksi for scaling up ion trap quantum computing [Wineland 98, Kielpinski 02] has been successfully used in this work to demonstrate the first experimental realization of transport gates as proposed by D. Leibfried [D. Leibfried 07]. In this thesis the work which was required to experimentally perform transport gates were covered. The work done included the design and implementation of the EVILs and DEATHs. The EVIL has inspired various projects which summer students has performed and is still the most widely used PI controller in our laboratory. For transport we have used the DEATHs and they have made all the transport experiments possible together with the work we have carried out on generating waveforms to perform complex experiments.

From the experiments we have performed on Hamiltonian extraction with the time varying velocity of calcium a new method for studying time-dependent Hamiltonians was developed. Additionally the experiments have showed sensitivity to the wavefronts of the Gaussian beam which was used in the experiment and it was possible to tell on which side of the focus the ion has passed through the laser beam. The Hamiltonian extraction experiment has thus given rise to a technique for measuring the velocity of an ion in vacuum as well as method to characterize the interacting laser beam of which both have never before been possible before.

The multi-process experiment performed with two beryllium ions transport gates have proved to be simple to extend from one operation through a single beam to two operations using a reflected beam. With fidelities on both processes of 98.5% for single qubit rotations and independently being able to vary the operation through varying the speed of the ion through the beam we are of strong opinion that transport gates in the QCCD architecture is a viable way for scaling up quantum processing while reducing the optical overhead required. With reflected beams we would require them to reflect close to the ion location to avoid the beams attenuating too much. For at-

tenuated beams we would be required to adapt the velocity of each operation individually. This would make the overall architecture more complex and concatenating sequences of operations can only be as fast as the slowest transport gate operation.

The experiments were made possible by the accurate design and implementation of a high speed DAC system we have developed, DEATH. Having investigated various methods to generate waveforms it turned out that Tikhonov regularization and quadratic programming were the two methods most likely to produce reliable transport waveforms. Being able to effortlessly create complex experiments with my formalism for creating multiple ion transport routines have reduced the experimental overhead of generating new waveforms during experimentation in the lab.

Some possible ideas for future experiments will now be discussed. Some of these are already at the time of writing the thesis being investigated other may simply be potential useful procedures that can be combined in other experiments.

9.1 Transport Molmer-Sorensen gate

A transport Molmer-Sorensen gate as with two beryllium ions would the logical next step to be implemented in our lab. Using two frequencies of light and the transport of two ions within one controlled harmonic potential, a two-qubit gate could be performed using transport. We have all the relevant equipment in our laboratory in order to perform such a gate. Matteo Marinelli has performed some of the calculations necessary to investigate how well controlled the velocity should be in order to perform these two-qubit gates. It would be great to see in the future an universal set of transport quantum gates.

9.2 Randomized benchmarking of transport gates and more

This is currently being pursued by David Nadlinger in our laboratory with the end goal to determine the fidelities of transport gates in our experiments and compare this to the normal static gates we can perform in our trap. Initial experiments have been performed with randomized benchmarking of static gates. The DEATHs hardware has been upgraded in order to implement randomized transport gates. Furthermore the hardware allows for branching on transport routines and conditional quantum operations using transport, splitting and combinations of ions will be possible with the new hardware.

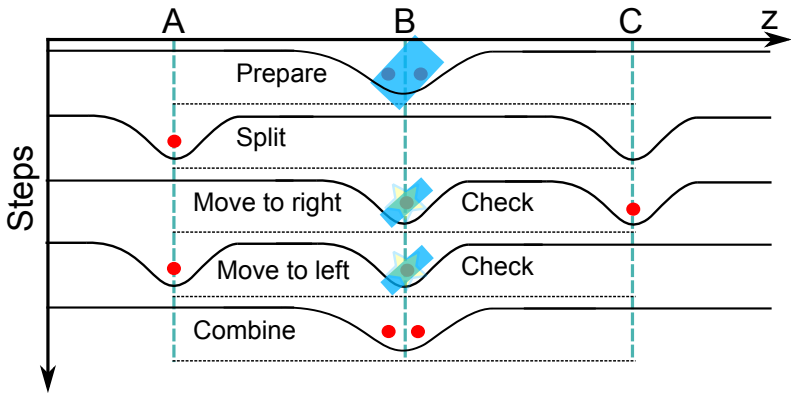
9.3 Iterative learning control and Hamiltonian engineering

The Hamiltonian extraction experiment can be used in order to extract the velocity as have been done in this experiment. We have seen that the velocity is not constant and at the stage of writing this thesis Robin Oswald is performing experiments to control the velocity profile. In the first iteration we are aiming to obtain a constant time-dependent detuning $\delta(t)$ and afterward arbitrarily shaped velocity profiles by using iterative learning control (ILC). With well controlled velocity profiles it would open up various fields we could study. One application would be time-dependent Hamiltonian engineering through the velocity profile. This clearly has to be though about in more detail, but one can imagine designing a time-dependent Hamiltonian such that it simulates the behaviour of the evolutions of other time-independent Hamiltonians. Another would be to characterize the beam profile in vacuum and feed back in order to adjust the beam properties. This can be done in various way by either a digital micromirror array (DMA) or when one would simply like to adjust the focus. The combination of Hamiltonian extraction, ILC and DMA may also help the future of scaling up the QCCD architecture by measuring the velocity and beam profiles using Hamiltonian extraction in different zones of the trap and then perform corrections and calibrations using ILC.

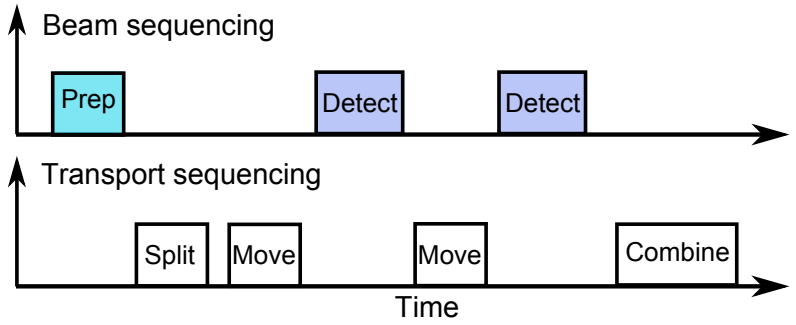
9.4 Ion conveyor, splitting and multi species transport

The idea behind the ion conveyor is simple and is only intended as a tool in working with multiple ions within our trap. Starting with a single ion specie like calcium or beryllium in a given experimental zone of the ion trap one can add additional species selectively by loading a single ion and transporting it and then combine with the ion/ions in an experimental zone. This will then act as an ion conveyor, combining it with splitting of ions reliably as R. Bowler [Bow 15] will enable more possibilities for multi-specie type of experiments.

We have performed splitting of ions using a waveforms designed to do so, but we used the shim electrodes to adjust the experiment to go from having two ions in one well to splitting. With the latest firmware on the DEATHs we should now be able to use a single splitting waveform and adjust any of the DC-electrodes offsets within *Ionizer* until we obtain reliable splitting. Before either the entire waveform would have had to be re-created with the offset manually added and by running many different settings for the same splitting waveform try to find the correct setting to achieve reliable splitting. Now by potentially scanning a parameter from *Ionizer* will allow an experimenter to easily find the correct setting. The experiment we have created in order to check reliable transport is shown in Fig.9.1. The experiment start of by preparing two ions, in this experiment cooling would suffice, since we are



(a) Overview of sequence. Sequence is carried out from top to bottom.



(b) Timing of the laser pulses as well as triggering of the transport.

Figure 9.1: Experimental sequence is carried out from top to bottom. The experiment is independent of ion species.

only interested in splitting two ions reliably. After some cooling the splitting procedure is performed and moving the two wells to the left and right of the splitting zone. The ions should be split, to check this the well on the left is moved to the splitting zone and a detection pulse is used to detect if the their is an ion in the well. The same is done to check whether there is an ion on the right hand side. Afterwards the two ions are combined and the process start over again. This experiment can be run with different offsets applied/scanned additionally to a chosen DC-electrode until the experiment can be run continuously and the ion pair is always split.

Our trap is designed to work with two species and it would be logical to try and perform transport and even fast transport of two ion species in the manner proposed in [Palmero 14]. This will serve two purposes, first the obvious fast transport of two ions species that has not been done before. Secondly using ILC and Hamiltonian extraction we would be able to not only calculate an optimal trajectory using Invariant based techniques as in [Palmero 14] $\vec{z}(t)$ for the ion over time, but also confirm via Hamiltonian extraction that the

trajectory is indeed what we are aiming to achieve. Using ILC then would allow us the correct the trajectory in an iterative manner in order to achieve the specified target $\vec{z}(t)$.

Appendix A

Abbreviations

AOM	Acousto-Pptic Modulator
DDS	Direct Digital Synthesizer
EOM	Electro-Optic Modulator
FWHM	Full Width Half Maximum
qccd	Quantum charge-coupled device
rms	Root mean square
rf	Radio frequency
FPGA	Field-Programmable Gate array
DAC	Digital-to-Analog Converter
ADC	Analog-to-Digital Converter
PMOD	Peripheral Module interface
PL	Programmable Logic
PS	Processing System
FSM	Finite State Machine
TTL	Transistor-Transistor Logic
EVIL	Electronically Variable Interactive Lockbox
DEATH	Direct Ethernet Adjustable Transport Hardware
SPI	Serial Peripheral Interface
COM	Communications Port
DSP	Digital Signal Processing
QCCD	Quantum Charge-Coupled Device
BW	Bandwidth
VBW	Video Bandwidth
AWG	Arbitrary Waveform Generator

Appendix B

Rotating frame and matrix exponential

Here I give a derivation changing to a general rotating frame. This is often done in order to simplify problems. Consider moving to an interaction like frame in which the associated Hamiltonian is H_R using a transformation matrix $H_r(t)$. We assume that this transformation matrix gives rise to an unitary operator U given by

$$U = e^{-\frac{i}{\hbar}\int_0^t H_r(t')dt'} \tag{B.1}$$

$$U^\dagger = e^{+\frac{i}{\hbar}\int_0^t H_r^\dagger(t')dt'}. \tag{B.2}$$

Unitary matrices have the following properties

$$U^\dagger U = UU^\dagger = I \tag{B.3}$$

$$\frac{dU}{dt}U^\dagger = -\frac{dU^\dagger}{dt}U \tag{B.4}$$

Given that the Hamiltonian H_S in the Schrödinger has the associated equation

$$i\hbar\frac{\partial}{\partial t}\left|\psi\right\rangle_S = H_S\left|\psi\right\rangle_S \tag{B.5}$$

We can now define the wave-function in transformed frame associated with the index R as

$$\left|\psi\right\rangle_S = U\left|\psi\right\rangle_R \tag{B.6}$$

$$\left|\psi\right\rangle_R = U^\dagger\left|\psi\right\rangle_S \tag{B.7}$$

The Schrödinger equation within this transformed frame is written as

$$\begin{aligned} i\hbar\frac{\partial}{\partial t}\left|\psi\right\rangle_R &= i\hbar\frac{\partial}{\partial t}\left(U^\dagger\left|\psi\right\rangle_S\right) \\ &= i\hbar\left(\frac{d}{dt}U^\dagger\left|\psi\right\rangle_S + U^\dagger\frac{d}{dt}\left|\psi\right\rangle_S\right) \\ &= i\hbar\left(\frac{i}{\hbar}H_r^\dagger(t)U^\dagger\left|\psi\right\rangle_S + \frac{1}{i\hbar}U^\dagger H_S\left|\psi\right\rangle_S\right) \\ &= i\hbar\left(\frac{i}{\hbar}H_r^\dagger(t)U^\dagger U\left|\psi\right\rangle_R + \frac{1}{i\hbar}U^\dagger H_S U\left|\psi\right\rangle_R\right) \\ &= \left(U^\dagger H_S U - H_r^\dagger(t)\right)\left|\psi\right\rangle_R. \end{aligned} \tag{B.8}$$

From the second property of unitary matrices listed $H_r(t)$ must be Hermitian $H_r(t) = H_r^\dagger(t)$. We can now define the Hamiltonian associated with the transform matrix $H_r(t)$ as

$$\hat{H}_R = U^\dagger H_S U - H_r(t). \tag{B.9}$$

Without the explicit differentiation as shown in the above derivation one can obtain the following useful results

$$\hat{H}_R = U^\dagger H_S U - i\hbar U^\dagger\frac{d}{dt}U. \tag{B.10}$$

Appendix C

Methods and tools for Hamiltonian estimation

C.1 Basis spline curves

One challenge in obtaining an estimate for the Hamiltonian is that we must optimize over continuous functions $\delta(t)$ and $\Omega(t)$. To address this, we represent $\delta(t)$ and $\Omega(t)$ with basis spline curves. Basis spline curves allow the construction of smooth functions using only a few parameters. This is achieved by introducing a set of polynomial Basis (B)-spline functions $B_{i,k}(t)$ of order k [de Boor]. A smooth curve $S(t)$ can then be represented as a linear combination of these B-spline basis functions [Bartels 95]

$$S(t) = \sum_{i=0}^{n} \alpha_i B_{i,k}(t). \tag{C.1}$$

The B-splines $B_{i,k}(t)$ of order k are recursively defined over the index i over a set of points $\vec{K} = \{t_0, t_1, ..., t_{n+k}\}$ which is referred to as the knot vector [de Boor].

$$\begin{aligned} B_{i,1}(t) &= \begin{cases} 1 & t_i \leq t \leq t_{i+1} \\ 0 & \text{otherwise} \end{cases} \\ B_{i,k}(t) &= \omega_{i,k}(t) B_{i,k-1}(t) + (1 - \omega_{i+1,k}(t))\, B_{i+1,k-1}(t). \\ \omega_{i,k}(t) &= \begin{cases} \frac{t - t_i}{t_{i+k-1} - t_i} & \text{if } t_i \neq t_{i+k-1} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{C.2}$$

Figure C.1 gives a visualization of the B-splines $B_{i,k}(t)$ and a basis spline curve. The B-spline construction ensures that any linear combination of the B-splines is continuous and has $(k-2)$ continuous derivatives. The knot vector $\vec{K}$ determines how the basis functions are positioned within the interval $[t_0, t_{n+k}]$. We notice that for our Hamiltonian the spacing of the B-splines is

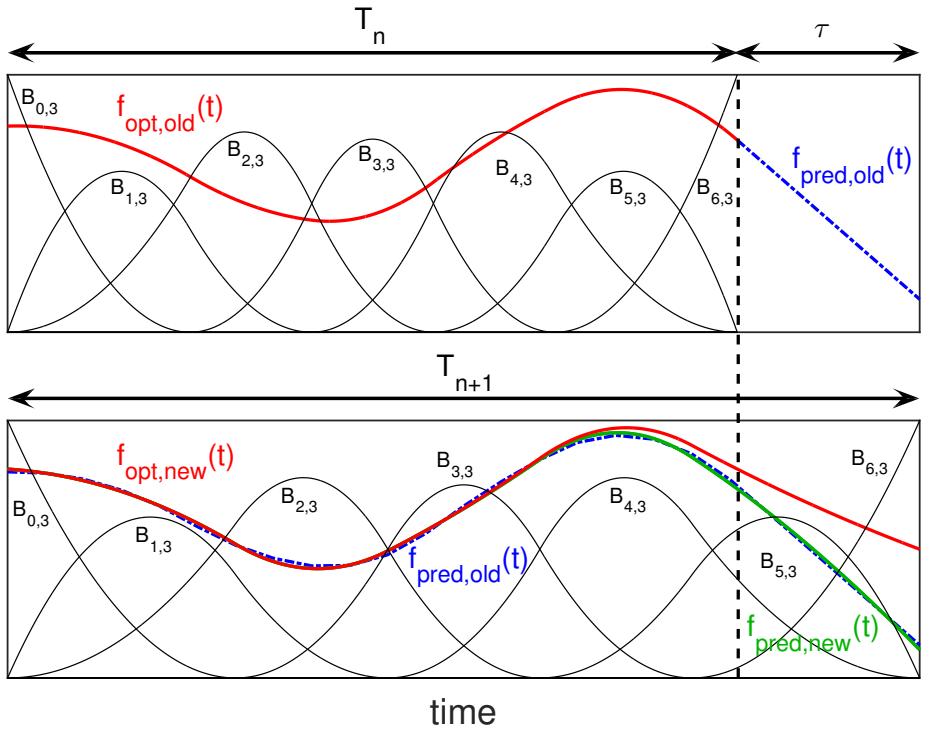


Figure C.1: Extending the Horizon Estimation: The steps performed when extending the time horizon from T_n to $T_{\mathrm{n}+1}$ are illustrated. We first predict in the old basis, then move to the new basis, and finally optimize again. The figure also shows the basis splines $B_{i,k}(t)$.

not critical, which we think is due to the smoothness of the variations in our Hamiltonian parameters $\delta(t)$ and $\Omega(t)$. We therefore used the Matlab function *spap2* to automatically choose a suitable knot vector and restricted ourselves to optimizing the coefficients α_i. We collect all coefficients α_i for $\delta(t)$ and $\Omega(t)$ and store them in a single vector $\vec{\alpha}$.

C.2 Extending the Horizon Estimation

The task of inferring the time-dependent Hamiltonian of the form (7.1) from the measured data can be cast into an optimization problem for which we use a reduced chi-squared cost function

$$J = \frac{1}{\nu} \sum_{t_\mathrm{off}} \sum_{\delta_L} \left[\frac{\langle \hat{\sigma}_z^\mathrm{meas}\,(t_\mathrm{off}, \delta_L)\rangle - \langle \hat{\sigma}_z^\mathrm{sim}\,(t_\mathrm{off}, \delta_L)\rangle}{\sigma^\mathrm{meas}(t_\mathrm{off}, \delta_L)} \right]^2 \tag{C.3}$$

where $\nu = N-n-1$ is the degrees of freedom with N the number of data points and n the number of fitting parameters, and $\sigma^\mathrm{meas}(t_\mathrm{off}, \delta_L)$ is the standard

error on the estimated $\langle \hat{\sigma}_z^{\text{meas}}(t_{\text{off}}, \delta_L)\rangle$ which we obtain assuming quantum projection noise. In our case the Hamiltonian $\hat{H}_I$ (Eq. 7.1) with offset $\hat{H}_{\text{s}} = \hbar\delta_L\hat{\sigma}_z/2$ is parametrized by $\Omega(t)$ and $\delta(t)$. We can thus write the problem as

$$\min_{\delta(t),\,\Omega(t)} J(\delta(t), \Omega(t)) \tag{C.4}$$

subject to

$$\begin{aligned} i\hbar\frac{\partial}{\partial t}|\Psi(t,\delta_L)\rangle &= \left(\hat{H}_I(t) + \hat{H}_s\right)|\Psi(t,\delta_L)\rangle\,, \\ |\Psi(t=0,\delta_L)\rangle &= |0\rangle\,, \\ \langle\hat{\sigma}_z^{\text{sim}}(t,\delta_L)\rangle &= \langle\Psi(t,\delta_L)|\,\hat{\sigma}_z\,|\Psi(t,\delta_L)\rangle \end{aligned} \tag{C.5}$$

for all δ_L.

This optimization problem is hard to efficiently solve in general, because it is nonlinear and non-convex due to the nature of Schrödinger's equation and the use of projective measurements. In order to overcome this challenge, we have implemented a method which we call "Extending the Horizon Estimation" (EHE) in analogy to a well-established technique called "Moving Horizon Estimation" (MHE) [Muske 95].

The key idea is that because our measurement data arises from a causal evolution, we can also estimate the Hamiltonian in a causal way. We define a time span ranging from the initial time to some later time which we call the time horizon. Instead of optimizing J over the complete time span at once, we first restrict ourselves to a small, initial time horizon reaching only up to the start of the qubit dynamics. Optimizing J over this short time horizon requires fewer optimization parameters and is simpler than attempting to optimize over the full data set. Once we have solved this small sub-problem, we extend the time horizon and re-run the optimization, extrapolating the results of the initial time window into the extended window in order to provide good starting conditions for the subsequent optimization. This is greatly advantageous for the use of non-linear least squares optimization, which typically works by linearizing the problem and converges much faster near the optimum. The extension of the horizon is used repeatedly until the time window covers the full data set.

Conceptually EHE is very similar to MHE. The main difference is that in MHE the time span has a fixed length and thus its origin gets shifted forwards in time along with the horizon. In EHE the origin stays fixed at the expense of having to increase the time span under consideration. MHE avoids this by introducing a so-called arrival cost to approximate the previous costs incurred before the start of the time span. This keeps the computational burden fixed over time, which is very important as MHE is usually used to estimate the state of a system in real-time, often on severely constrained embedded platforms.

Since neither constraint applies to our problem, we decided to extend the horizon rather than finding an approximate arrival cost. This is advantageous since finding the arrival cost in the general case is still an open problem. Due to the similarity between MHE and EHE, we anticipate future improvements by adapting techniques used in MHE to EHE.

Next, we present a more detailed algorithmic summary of our implementation of the method outlined above.

1. **Searching for a starting point.** Here we reconstruct the Hamiltonian for a first, minimal time horizon such that we can then use this as a starting point to iteratively extend the horizon as described in step 2.

   a) Choose an initial time horizon such that it contains the region where the first discernible qubit dynamics occur.

   b) Cut down the number of fitting parameters as much as possible, e.g. by using few Basis splines of low order. This amounts to choosing empirically a low number of basis splines (and thus the length of $\vec{\alpha}_0$) which might represent $\delta(t)$ and $\Omega(t)$ over the given region.

   c) Use a nonlinear least squares fitting routine to minimize J by varying the parameters $\vec{\alpha}_0$. In the case that the initial fit is not good or no minimum is found, try new initial conditions, change the number of B-spline functions, or manually adjust the function using prior knowledge of the physical system under consideration.

   This procedure is used to provide a starting point for the optimization over the initially chosen window, which is typically performed with a higher order set of B-splines. From this starting point, we iteratively extend the fitting method to the full data set as follows.

2. **Extend the horizon** This step is repeated until the whole time horizon is covered. It consists of the following sequence, which is illustrated in figure C.1.

   a) Extend the time horizon by τ from T_{n} to $T_{\mathrm{n+1}} = T_{\mathrm{n}} + \tau$.

   b) Extrapolate $f_{\mathrm{opt,old}}(t)$ within τ, e.g. using *fnxtr* in Matlab.

   c) Adapt the Basis splines to the new time horizon $T_{\mathrm{n+1}}$ and represent $f_{\mathrm{pred,old}}(t)$ in the new basis, giving $f_{\mathrm{pred,new}}(t)$. In Matlab one can use *spap2* to do this.

   d) Use $f_{\mathrm{pred,new}}(t)$ as the initial guess for a weighted nonlinear least squares fit over the extended time span up to $T_{\mathrm{n+1}}$.

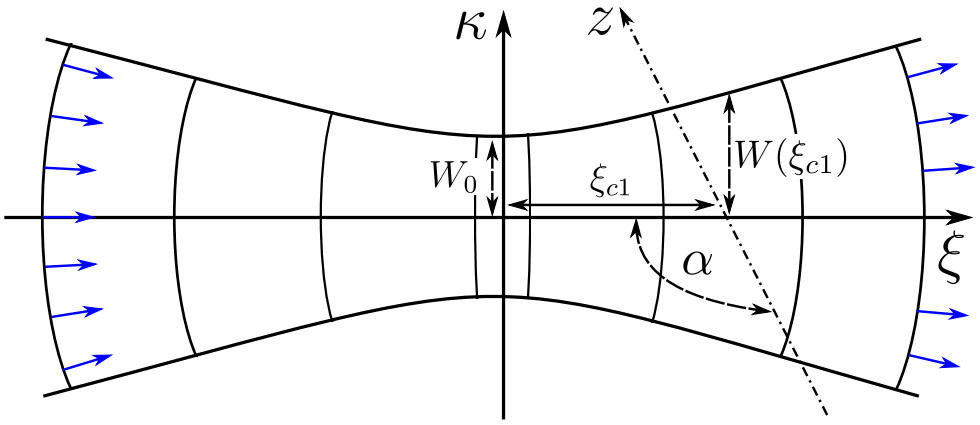


Figure C.2: The beam propagation direction lies along the ξ-axis and the ion is transported along the z-axis lying the $\kappa\xi$-plane as indicated. Normalized vectors representing $\vec{e}_l(\kappa, \xi)$ lying perpendicular to the wavefronts are indicated by the blue arrows.

e) Judge the results of the fit based on its reduced chi-squared value χ^2_{red}. If it is below a specified bound, continue with an additional iteration of steps a)-d), repeating until the full region of the data is covered. Otherwise, try the following fall-back procedures:

   i. Reduce τ, the time by which the time horizon is extended, and try again
   ii. Increase the number of Basis splines and try again
   iii. Try again using a different starting point.

   If all of those fail, we have to resort to increasing the bound on χ^2_{red}.

3. **Post-processing**. The following steps are optional and were performed manually in cases where we desired to improve the fit, or examine its behaviour.

   a) The optimization over the whole time horizon was re-run using different numbers of Basis splines for $\delta(t)$ and $\Omega(t)$. This served as an useful check on the sensitivity of the fit.

   b) The optimization over the whole time horizon was re-run using a starting point based on the previously found optimum plus randomized deviations. This examines robustness of the final fit.

C.3 Wavefront correction

For plane waves we find that $\dot{\phi}(t) = \vec{k} \cdot \vec{v}(t)$ which is the well-known expression for the first-order Doppler shift. For transport through a real Gaussian beam, the wave-vector direction changes with position. Taking this into account, the derivative of $\phi(t)$ becomes

$$\dot{\phi}(t) = \left[k_z'(z(t))z(t) + k_z(z(t))\right] \dot{z}(t) \tag{C.6}$$

where $k_z' = dk_z/dz$ and $\dot{z}(t)$ is the component of the ion's velocity which lies along the z-axis. We extract $\delta(t)$ using our Hamiltonian estimation procedure, thus to obtain the velocity of the ion we use

$$\dot{z}(t) = \frac{-\delta(t) + \delta_L}{k_z'(z(t))z(t) + k_z(z(t))} \ . \tag{C.7}$$

As the ion moves through the beam it experiences the same magnitude of the wave vector $|\vec{k}| = 2\pi/\lambda$, but the angle θ between the ion's direction and the wave vector changes. Written as a function of this angle, the velocity becomes

$$\dot{z}(t) = \frac{-\delta(t) + \delta_L}{-|\vec{k}| \sin\left(\theta(z(t))\right) \theta'(z(t))z(t) + |\vec{k}| \cos(\theta(z(t)))} \tag{C.8}$$

where $\theta'(z(t)) = d\theta(z(t))/dz(t)$. We parameterize our Gaussian beam according to figure C.2. The phase is given as a function of both the position along the beam axis ξ and the perpendicular distance from this axis κ by [Saleh 07]

$$\varphi(\kappa, \xi) = |\vec{k}|\xi - \zeta(\xi) + \frac{|\vec{k}|\kappa^2}{2R(\xi)} \ . \tag{C.9}$$

where the Gaussian beam parameters include the beam waist $W(\xi)$, the radius of curvature $R(\xi)$ the Rayleigh range ξ_R and the Guoy phase shift $\zeta(\xi)$. These are given by the expressions

$$\begin{aligned} W(\xi) &= W_0 \sqrt{1 + \left(\frac{\xi}{\xi_R}\right)^2} \\ R(\xi) &= \xi \left(1 + \left(\frac{\xi_R}{\xi}\right)^2\right) \\ \zeta(\xi) &= \tan^{-1}\left(\frac{\xi}{\xi_R}\right) \\ \xi_R &= \frac{\pi W_0^2}{\lambda} \\ k &= \frac{2\pi}{\lambda} \end{aligned} \tag{C.10}$$

where W_0 is the minimum beam waist and λ the laser wavelength. The ion moves along the z-axis shown in figure C.2. In the $\kappa\xi$-plane an unit vector $\vec{e}_l(\kappa,\xi)$ perpendicular to the wavefronts is given by

$$\vec{e}_l(\kappa,\xi) = \frac{\nabla\varphi(\kappa,\xi)}{||\nabla\varphi(\kappa,\xi)||} \tag{C.11}$$

and the unit vector $\vec{e}_v$ pointing along the direction of transport is given by

$$\vec{e}_v = \begin{bmatrix}\cos(\alpha)\\ \sin(\alpha)\end{bmatrix} \tag{C.12}$$

The angle $\theta(\xi)$ between the wave and position vector is then given by the dot product

$$\theta(\kappa) = \cos^{-1}\left(\vec{e}_n \cdot \vec{e}_v\right). \tag{C.13}$$

which can be written in terms of the full set of parameters above as

$$\begin{aligned}
\theta(\kappa) &= \cos^{-1}\left(\gamma_1 + \gamma_2\right)\\
\gamma_1 &= \frac{\cos(\alpha)\left(-2\xi_R\left(\xi^2+\xi_R^2\right) + k\kappa^2\left(\xi_R^2-\xi^2\right) + 2k\left(\xi^2+\xi_R^2\right)^2\right)}{\eta(\kappa)}\\
\gamma_2 &= \frac{\sin(\alpha)2k\kappa\xi\left(\xi^2+\xi_R^2\right)}{\eta(\kappa)}\\
\eta(\kappa,\xi) &= \left(\xi^2+\xi_R^2\right)\left[4\left(\frac{k\kappa\xi}{\xi^2+\xi_R^2}\right)^2 + \left(-\frac{2\xi_R}{\xi^2+\xi_R^2} + k\left(2 + \frac{\kappa^2\left(\xi_R^2-\xi^2\right)}{\left(\xi^2+\xi_R^2\right)^2}\right)\right)^2\right]\\
\kappa(t) &= z(t)\sin(\alpha)
\end{aligned} \tag{C.14}$$

where in our experiments $\alpha = 3\pi/4$.

Using Eq. C.8 and C.14 we examined the value of ξ_{cl} required for the velocity to match for our two beam positions. We find that they agree for $\xi_{\mathrm{cl}} = -2.27$ mm, which is within the experimental uncertainties for our setup.

C.4 Error estimation

To obtain error bars of the time-dependent functions we use non-parametric bootstrapping [Murphy 12]. The process is summarized as follow:

1. **Estimate initial solution** Estimate the time dependent functions from the original data using Hamiltonian estimation.
2. **Resampling** Create N_{s} sample solutions for all time-dependent functions in the following way:

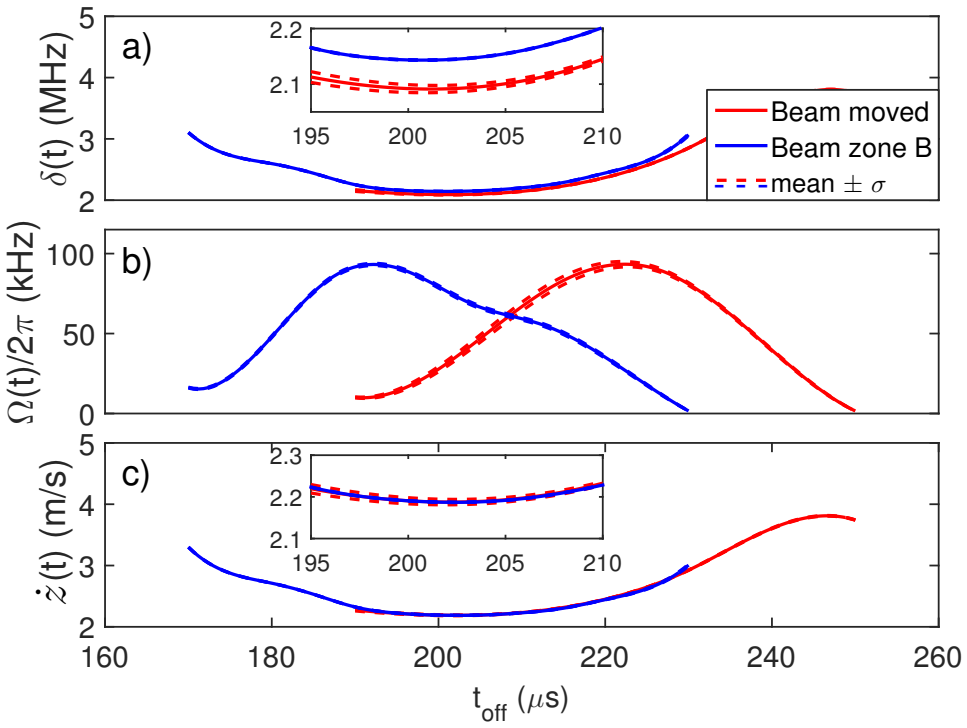


Figure C.3: Parametric bootstrap resampling: Predictions for $\delta(t)$, $\Omega(t)$ and $\dot{z}(t)$ with error bounds obtained using parametric bootstrap resampling, assuming quantum projection noise. This can be compared with the error bounds obtained from the non-parametric method which are shown in Figure 7.7 in the main text. The bounds are tighter for the parametric bootstrapping.

a) Form a sample set by randomly picking with replacement from the photon count data used in qubit detection.

b) Re-estimate new time-dependent functions by optimizing over the full time span, using the solution found in (1) as a starting point.

c) Record the reduced chi-squared values $\chi^2_{\mathrm{red,r}}$ for each sample r along with the B-spline curve coefficients $\vec{\alpha}_{\mathrm{r}}$

3. **Post-process samples**

a) Form a histogram of the chi-squared values $\chi^2_{\mathrm{red,r}}$.

b) Find and fit a normal-like distribution to the histogram with preference to the spread with lowest lying $\chi^2_{\mathrm{red,r}}$ in the case of a multimodal distribution. From the fit obtain the mean reduced-chi squared value $\langle\chi^2_{\mathrm{red,r}}\rangle$ as well as the standard deviation σ_χ.

c) Eliminate the outlier samples by removing all $\vec{\alpha}_r$ with $\chi^2_{\text{red,r}}$ values that are 3-5σ_χ from the mean $\langle\chi^2_{\text{red,r}}\rangle$.

d) Form a matrix $\boldsymbol{Y}$ where each row vector is a sample set of coefficients $\vec{\alpha}_r$ that remained after step 3(c).

4. **Obtain statistics**

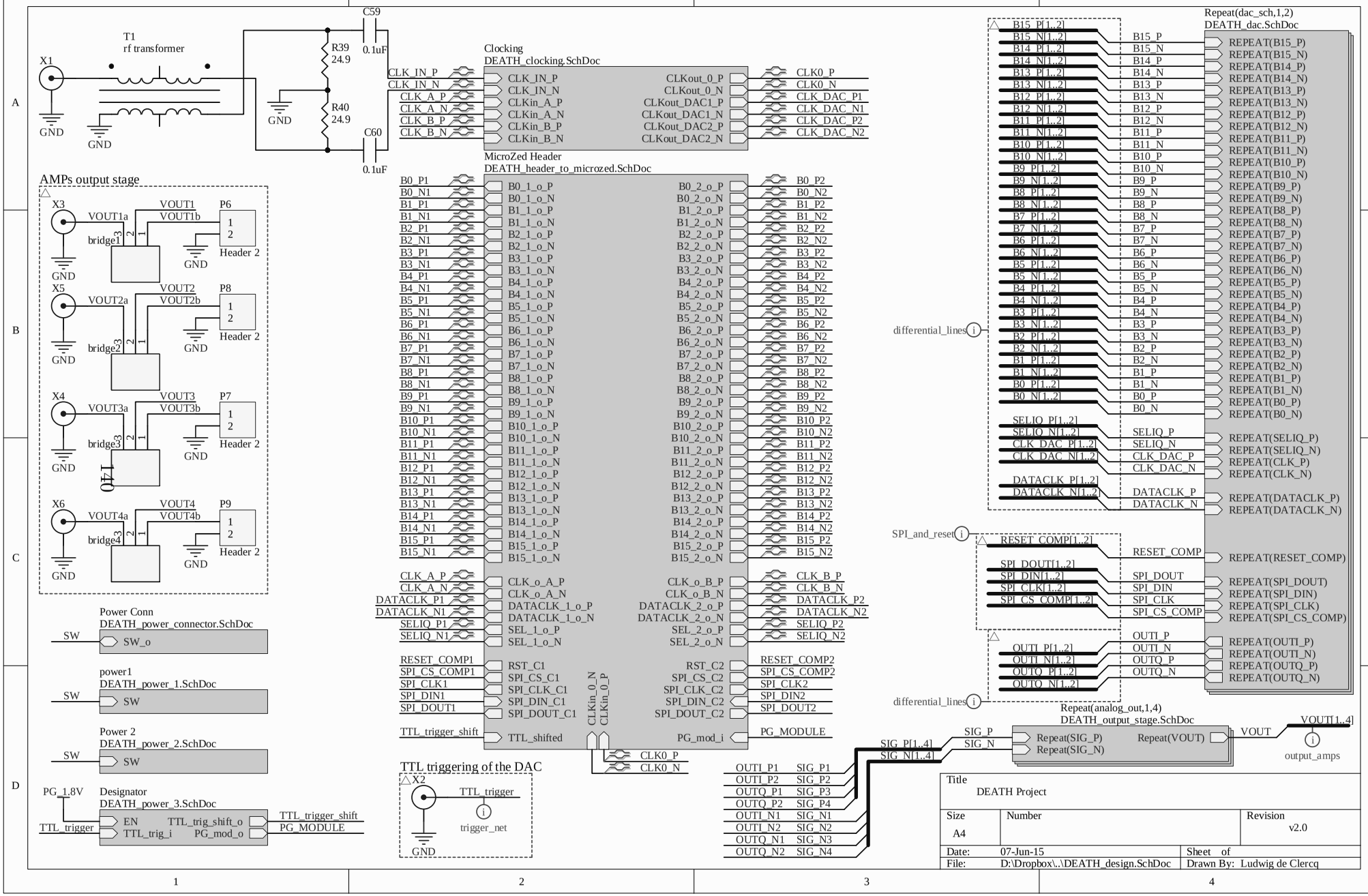
a) Find the mean B-spline coefficients $\langle\vec{\alpha}\rangle$ of equation C.1 by taking the mean over the column vectors of $\boldsymbol{Y}$ with each element of the mean given by $\langle\vec{\alpha}\rangle_i = \langle\alpha_i\rangle$.

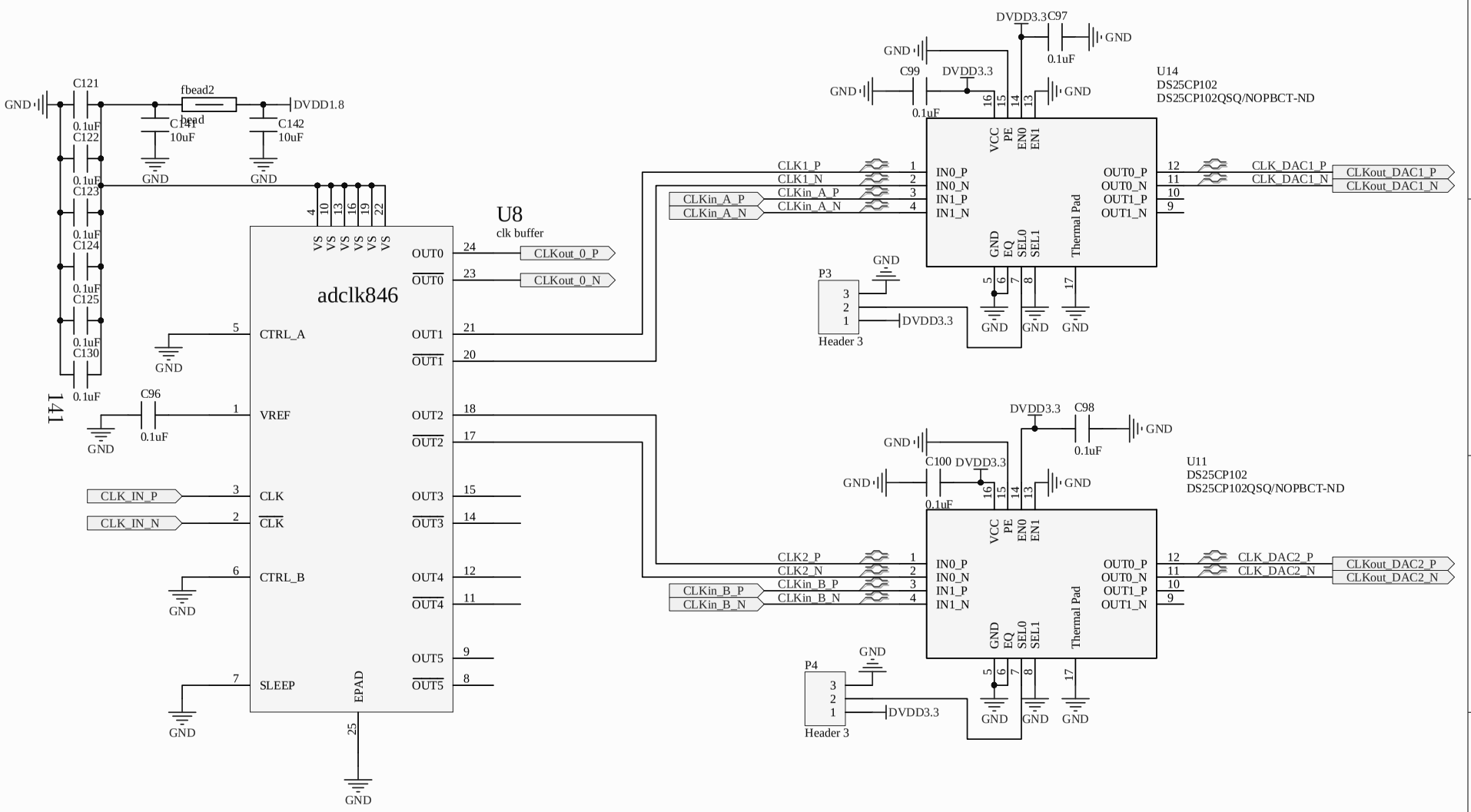
b) Find the covariance matrix $\boldsymbol{\Sigma} = \text{cov}(\boldsymbol{Y}^\alpha)$ with the elements $\boldsymbol{\Sigma}_{ij} = \hat{E}\left[(\alpha_i - \langle\alpha_i\rangle)(\alpha_j - \langle\alpha_j\rangle)\right]$ and $\hat{E}$ the expectation operator. The standard deviations of each of the mean coefficients $\langle\alpha_i\rangle$ is given by $\sigma_{\langle\alpha_i\rangle} = \sqrt{\boldsymbol{\Sigma}_{ii}}$. We record these values in a row vector $\vec{\sigma}_{\langle\alpha_i\rangle}$.

We have also applied parametric bootstrapping in order to obtain the error bounds shown in figure C.3. The difference to the non-parametric case is that in point (2) the samples are created using the solutions obtained from (1) and adding quantum projection noise. For each sample the Hamiltonian is estimated. The estimates from multiple samples are used to construct error bounds in the same manner as for the non-parametric resampling. We have found that the error bounds obtained from parametric bootstrapping are lower compared to that of the non-parametric case as shown in figure 7.7. We think this is due to the latter exploring deviations around a single minimum in the optimization landscape, while the case resampling arrives at different local minima which are spread over a wider region.

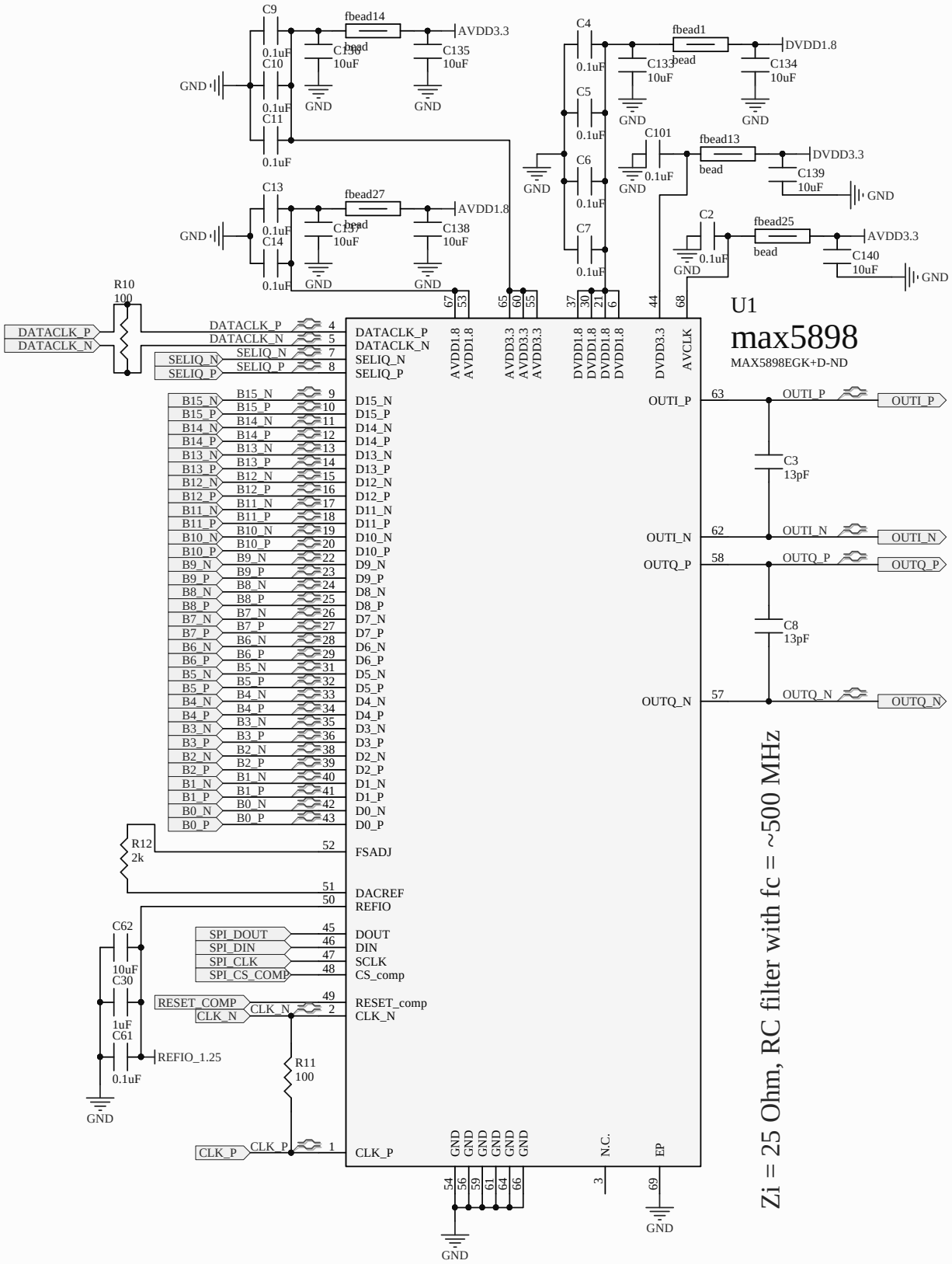
Appendix D

DEATH Schematics

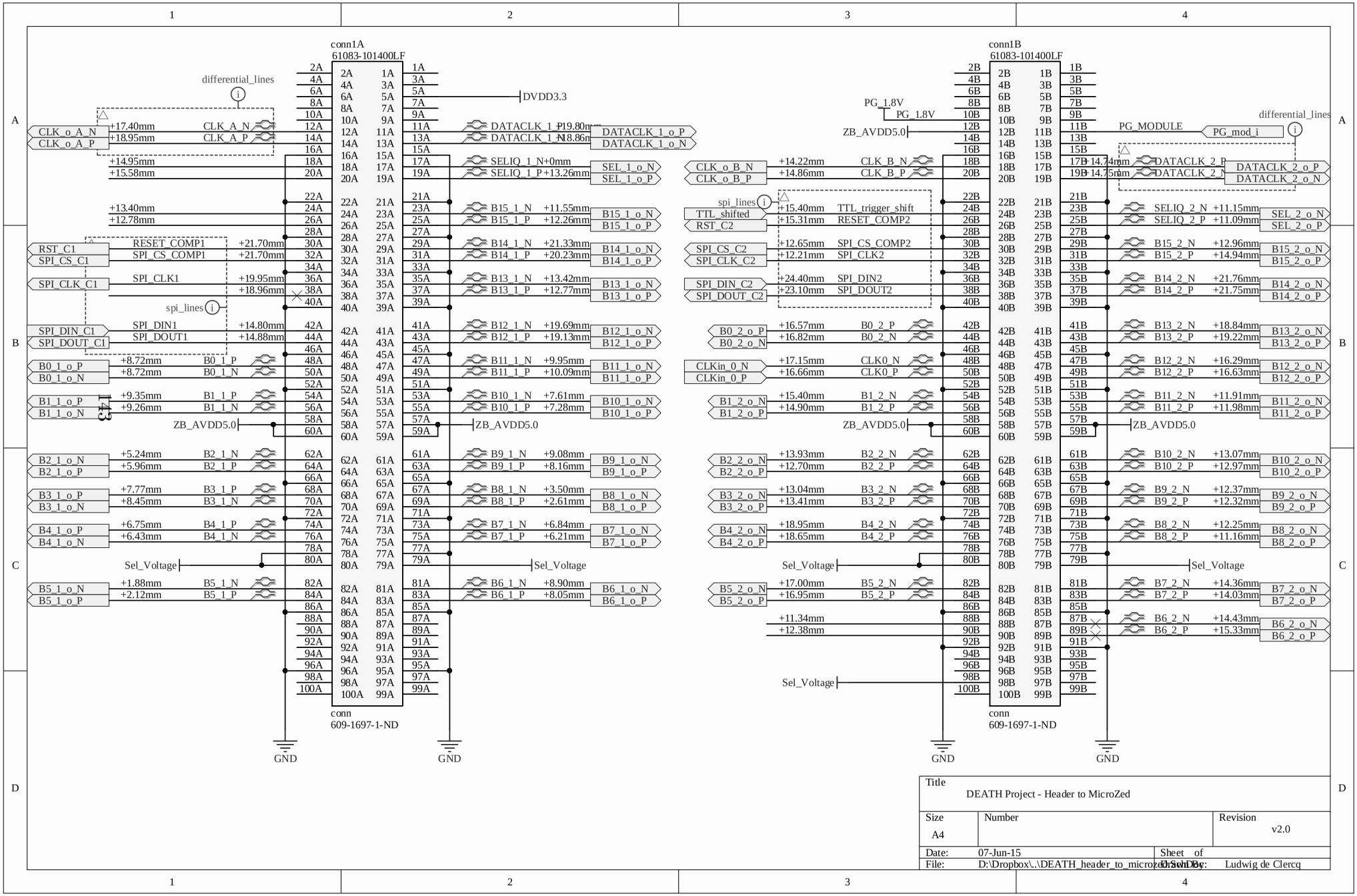

T1
rf transformer
X1
GND
C59
0.1uF
R39
24.9
R40
24.9
C60
0.1uF
CLK_IN_P
CLK_IN_N
CLK_A_P
CLK_A_N
CLK_B_P
CLK_B_N
Clocking
DEATH_clocking.SchDoc
CLKout_0_P
CLKout_0_N
CLKout_DAC1_P
CLKout_DAC1_N
CLKout_DAC2_P
CLKout_DAC2_N
CLK0_P
CLK0_N
CLK_DAC_P1
CLK_DAC_N1
CLK_DAC_P2
CLK_DAC_N2
MicroZed Header
DEATH_header_to_microzed.SchDoc
AMPs output stage
X3
X5
X4
X6
VOUT1a
VOUT1b
VOUT2a
VOUT2b
VOUT3a
VOUT3b
VOUT4a
VOUT4b
bridge1
bridge2
bridge3
bridge4
P6
P8
P7
P9
Header 2
140
Power Conn
DEATH_power_connector.SchDoc
power1
DEATH_power_1.SchDoc
Power 2
DEATH_power_2.SchDoc
PG_1.8V
Designator
DEATH_power_3.SchDoc
EN
TTL_trig_i
TTL_trig_shift_o
PG_mod_o
TTL_trigger_shift
PG_MODULE
TTL triggering of the DAC
X2
TTL_trigger
trigger_net
CLK_o_A_P
CLK_o_A_N
CLK_o_B_P
CLK_o_B_N
DATACLK_1_o_P
DATACLK_1_o_N
DATACLK_2_o_P
DATACLK_2_o_N
SEL_1_o_P
SEL_1_o_N
SEL_2_o_P
SEL_2_o_N
RST_C1
SPI_CS_C1
SPI_CLK_C1
SPI_DIN_C1
SPI_DOUT_C1
RST_C2
SPI_CS_C2
SPI_CLK_C2
SPI_DIN_C2
SPI_DOUT_C2
CLKin_0_N
CLKin_0_P
TTL_shifted
PG_mod_i
PG_MODULE
OUTI_P1
OUTI_P2
OUTQ_P1
OUTQ_P2
OUTI_N1
OUTI_N2
OUTQ_N1
OUTQ_N2
SIG_P1
SIG_P2
SIG_P3
SIG_P4
SIG_N1
SIG_N2
SIG_N3
SIG_N4
differential_lines
SPI_and_reset
Repeat(dac_sch,1,2)
DEATH_dac.SchDoc
REPEAT(B15_P)
REPEAT(B15_N)
REPEAT(B14_P)
REPEAT(B14_N)
REPEAT(B13_P)
REPEAT(B13_N)
REPEAT(B12_P)
REPEAT(B12_N)
REPEAT(B11_P)
REPEAT(B11_N)
REPEAT(B10_P)
REPEAT(B10_N)
REPEAT(B9_P)
REPEAT(B9_N)
REPEAT(B8_P)
REPEAT(B8_N)
REPEAT(B7_P)
REPEAT(B7_N)
REPEAT(B6_P)
REPEAT(B6_N)
REPEAT(B5_P)
REPEAT(B5_N)
REPEAT(B4_P)
REPEAT(B4_N)
REPEAT(B3_P)
REPEAT(B3_N)
REPEAT(B2_P)
REPEAT(B2_N)
REPEAT(B1_P)
REPEAT(B1_N)
REPEAT(B0_P)
REPEAT(B0_N)
REPEAT(SELIQ_P)
REPEAT(SELIQ_N)
REPEAT(CLK_P)
REPEAT(CLK_N)
REPEAT(DATACLK_P)
REPEAT(DATACLK_N)
REPEAT(RESET_COMP)
REPEAT(SPI_DOUT)
REPEAT(SPI_DIN)
REPEAT(SPI_CLK)
REPEAT(SPI_CS_COMP)
REPEAT(OUTI_P)
REPEAT(OUTI_N)
REPEAT(OUTQ_P)
REPEAT(OUTQ_N)
Repeat(analog_out,1,4)
DEATH_output_stage.SchDoc
Repeat(SIG_P)
Repeat(SIG_N)
Repeat(VOUT)
VOUT
VOUT[1..4]
output_amps
Title
DEATH Project
Size
A4
Number
Revision
v2.0
Date:
07-Jun-15
Sheet of
File:
D:\Dropbox\..\DEATH_design.SchDoc
Drawn By: Ludwig de Clercq



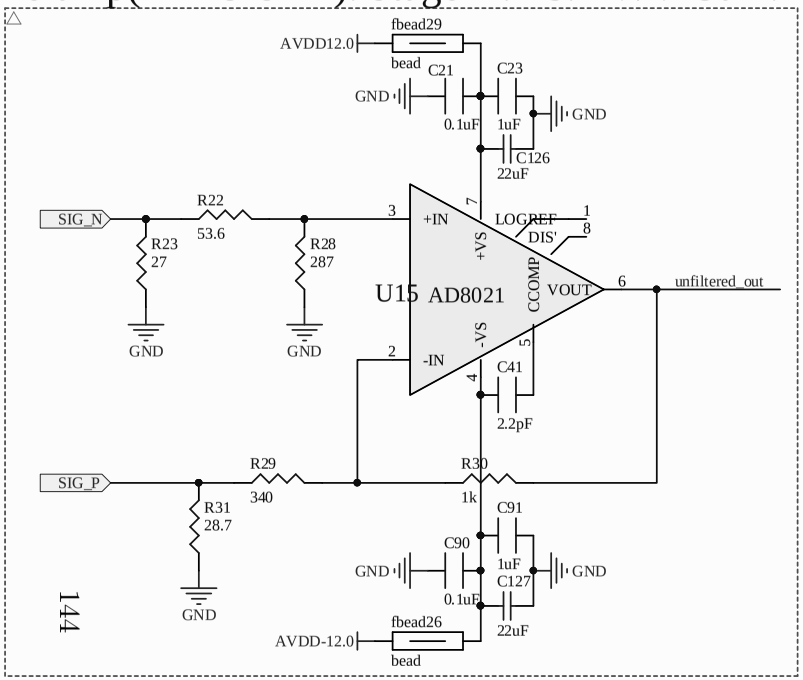
Title	DEATH Project - Clocking	
Size: A4	Number:	Revision: v2.0
Date: 07-Jun-15	Sheet of	
File: D:\Dropbox\..\DEATH_clocking.SchDoc	Drawn By: Ludwig de Clercq	



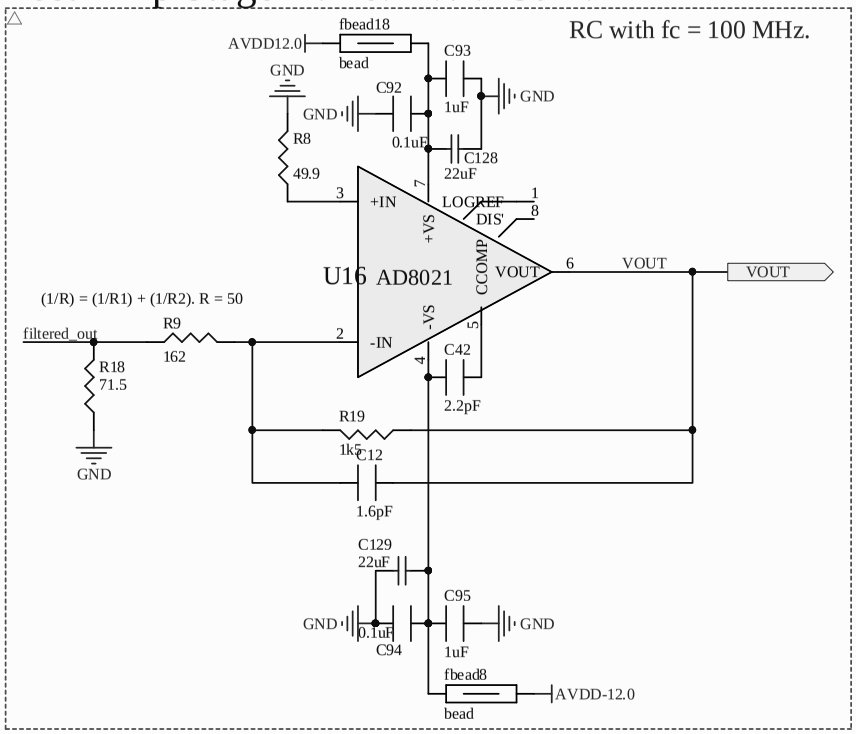
Title: DEATH Project - DAC		
Size: A4	Number: 142	Revision: v2.0
Date: 07-Jun-15		Sheet of
File: D:\Dropbox\..\DEATH_dac.SchDoc		Drawn By: Ludwig de Clercq


conn1A
61083-101400LF
conn1B
61083-101400LF
differential_lines
spi_lines
DVDD3.3
ZB_AVDD5.0
Sel_Voltage
PG_1.8V
PG_MODULE
PG_mod_i
GND
conn
609-1697-1-ND
CLK_o_A_N
CLK_o_A_P
CLK_A_N
CLK_A_P
DATACLK_1_o_P
DATACLK_1_o_N
SEL_1_o_N
SEL_1_o_P
B15_1_o_N
B15_1_o_P
B14_1_o_N
B14_1_o_P
B13_1_o_N
B13_1_o_P
B12_1_o_N
B12_1_o_P
B11_1_o_N
B11_1_o_P
B10_1_o_N
B10_1_o_P
B9_1_o_N
B9_1_o_P
B8_1_o_N
B8_1_o_P
B7_1_o_N
B7_1_o_P
B6_1_o_N
B6_1_o_P
RST_C1
SPI_CS_C1
SPI_CLK_C1
SPI_DIN_C1
SPI_DOUT_C1
B0_1_o_P
B0_1_o_N
B1_1_o_P
B1_1_o_N
B2_1_o_N
B2_1_o_P
B3_1_o_P
B3_1_o_N
B4_1_o_P
B4_1_o_N
B5_1_o_N
B5_1_o_P
RESET_COMP1
SPI_CS_COMP1
SPI_CLK1
SPI_DIN1
SPI_DOUT1
B0_1_P
B0_1_N
B1_1_P
B1_1_N
B2_1_N
B2_1_P
B3_1_P
B3_1_N
B4_1_P
B4_1_N
B5_1_N
B5_1_P
CLK_o_B_N
CLK_o_B_P
CLK_B_N
CLK_B_P
DATACLK_2_o_P
DATACLK_2_o_N
SEL_2_o_N
SEL_2_o_P
B15_2_o_N
B15_2_o_P
B14_2_o_N
B14_2_o_P
B13_2_o_N
B13_2_o_P
B12_2_o_N
B12_2_o_P
B11_2_o_N
B11_2_o_P
B10_2_o_N
B10_2_o_P
B9_2_o_N
B9_2_o_P
B8_2_o_N
B8_2_o_P
B7_2_o_N
B7_2_o_P
B6_2_o_N
B6_2_o_P
TTL_shifted
RST_C2
SPI_CS_C2
SPI_CLK_C2
SPI_DIN_C2
SPI_DOUT_C2
B0_2_o_P
B0_2_o_N
CLKin_0_N
CLKin_0_P
B1_2_o_N
B1_2_o_P
B2_2_o_N
B2_2_o_P
B3_2_o_N
B3_2_o_P
B4_2_o_N
B4_2_o_P
B5_2_o_N
B5_2_o_P
TTL_trigger_shift
RESET_COMP2
SPI_CS_COMP2
SPI_CLK2
SPI_DIN2
SPI_DOUT2
B0_2_P
B0_2_N
CLK0_N
CLK0_P
B1_2_N
B1_2_P
B2_2_N
B2_2_P
B3_2_N
B3_2_P
B4_2_N
B4_2_P
B5_2_N
B5_2_P
Title
DEATH Project - Header to MicroZed
Size
A4
Number
Revision
v2.0
Date:
07-Jun-15
Sheet of
File:
D:\Dropbox\..\DEATH_header_to_microzed.SchDoc
Drawn By:
Ludwig de Clercq

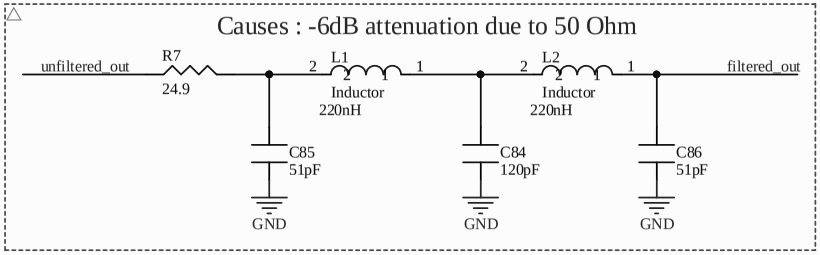
Pre-amp(Zi=25 Ohm). Stage 1 : ~3.1 V/V Gain.



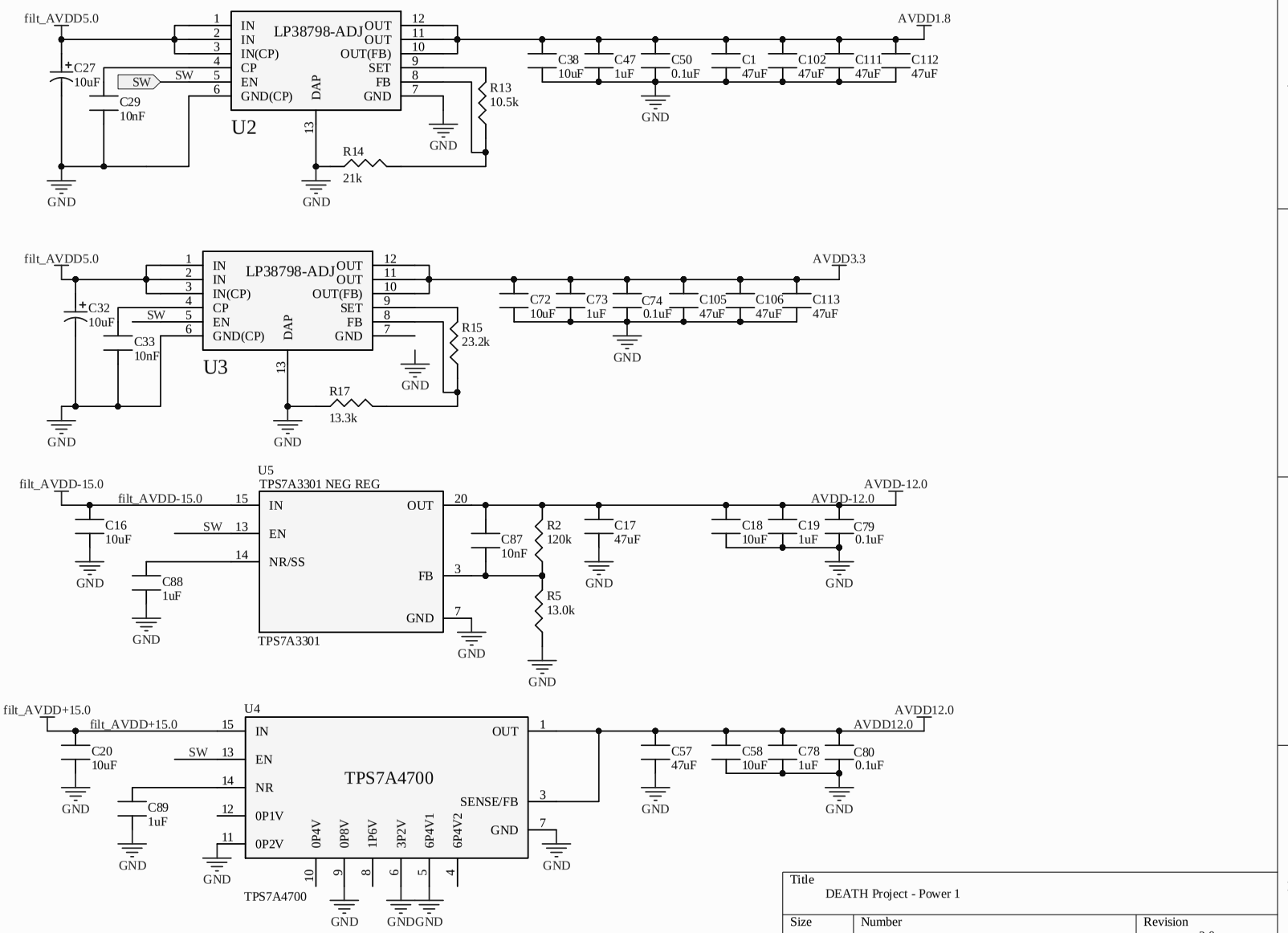
Post-Amp Stage 1 : ~6.2 V/V Gain.



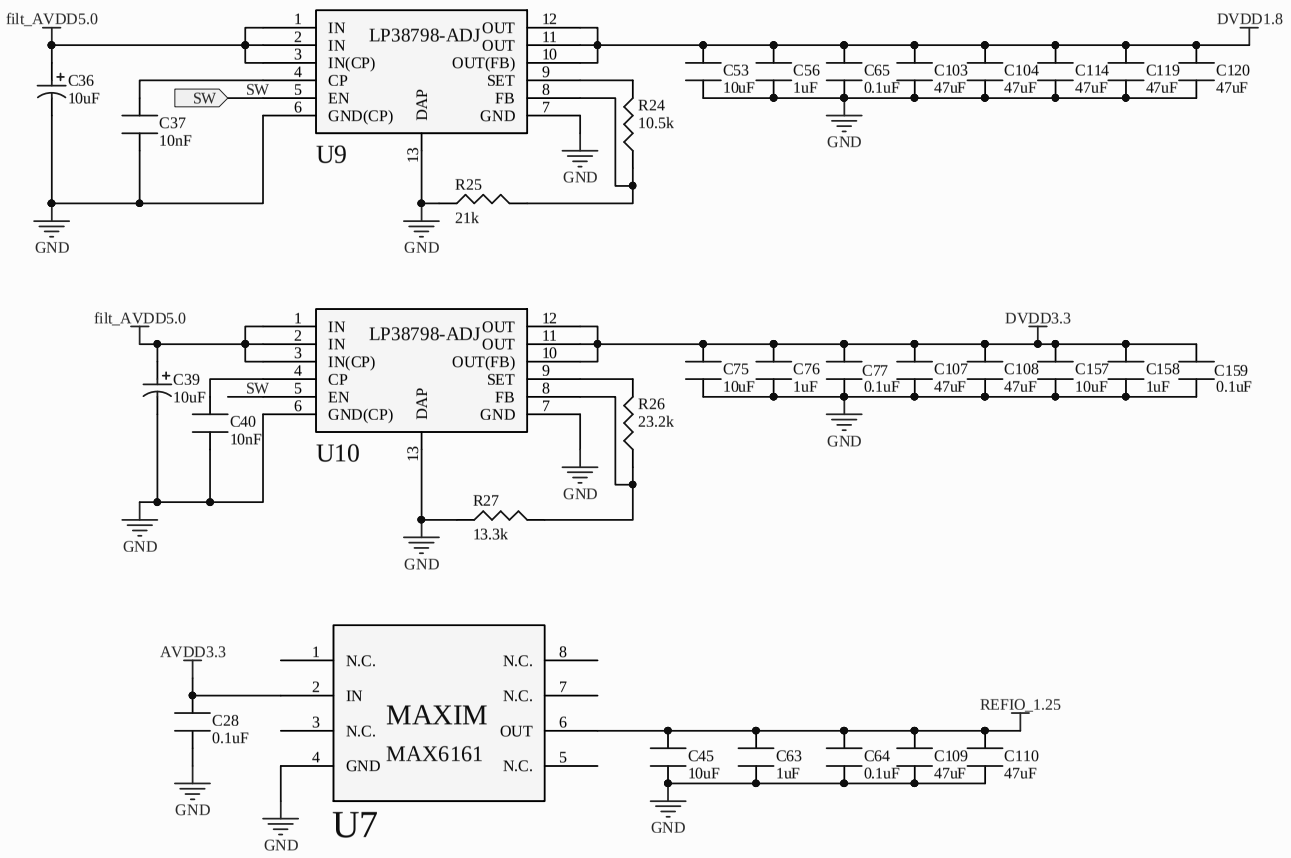
4th Order Butterworth : fc ~= 50 MHz



Title: DEATH Project - Output Stage		
Size: A4	Number:	Revision: v2.0
Date: 07-Jun-15	Sheet of	
File: D:\Dropbox\..\DEATH_output_stage.SchDoc	Drawn By: Ludwig de Clercq	



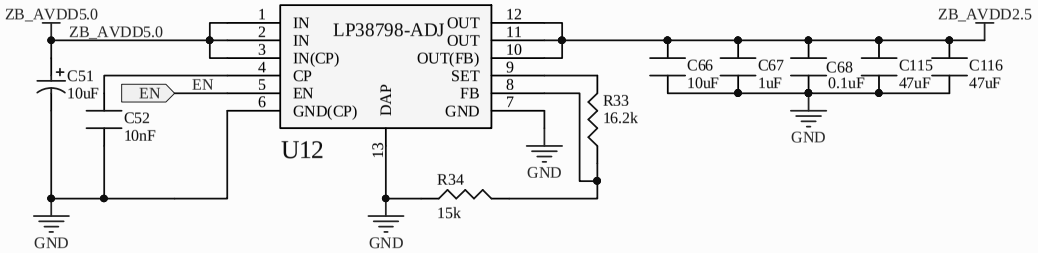
Title: DEATH Project - Power 1		
Size: A4	Number:	Revision: v2.0
Date: 07-Jun-15	Sheet of	
File: D:\Dropbox\..\DEATH_power_1.SchDoc	Drawn By: Ludwig de Clercq	

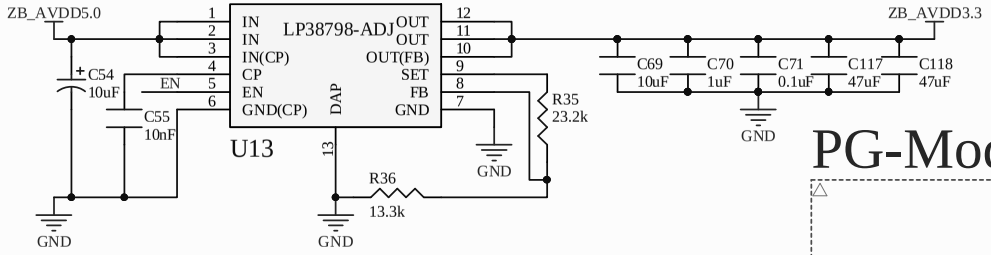


146

Title: DEATH Project - Power 2		
Size: A4	Number:	Revision: v2.0
Date: 07-Jun-15	Sheet of	
File: D:\Dropbox\..\DEATH_power_2.SchDoc	Drawn By: Ludwig de Clercq	

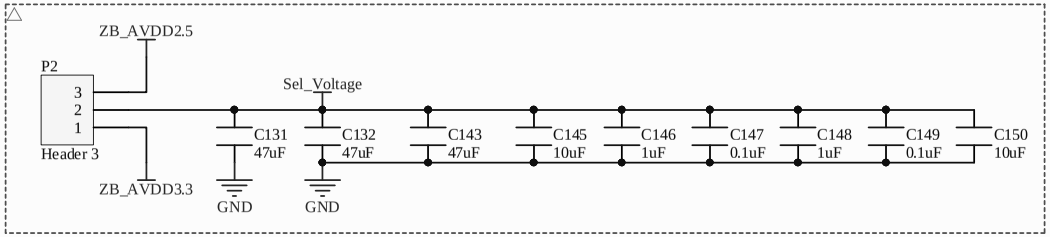
Dedicated PSU to microzed



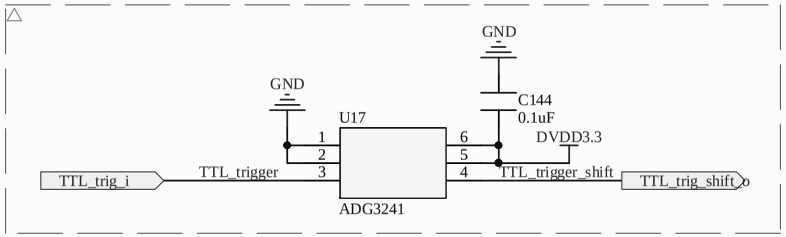


147

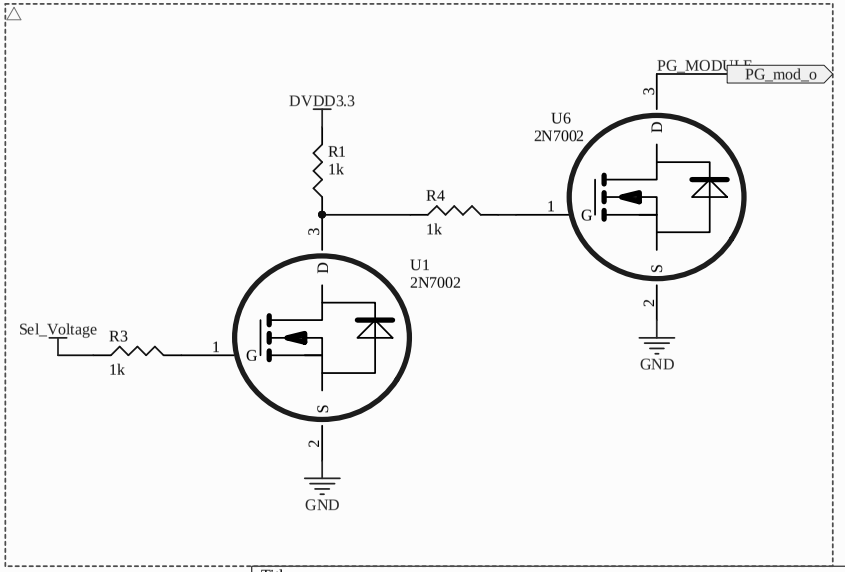
Voltage select ZeBoard



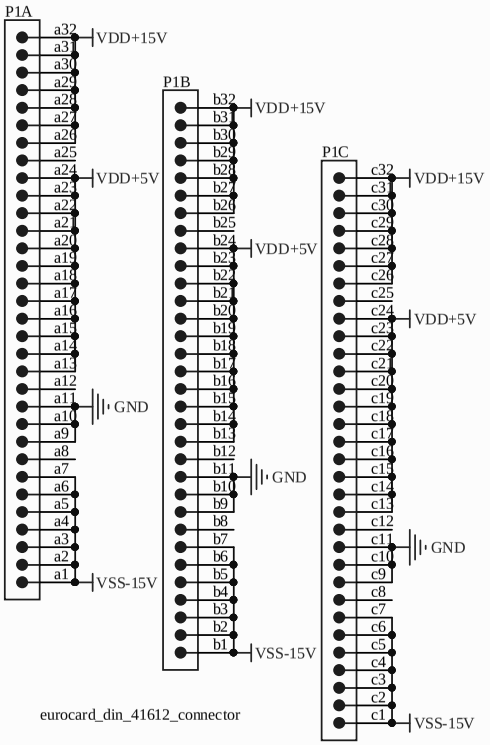
Level translator TTL

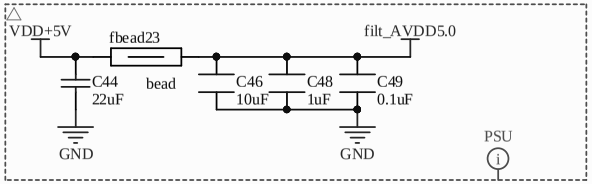


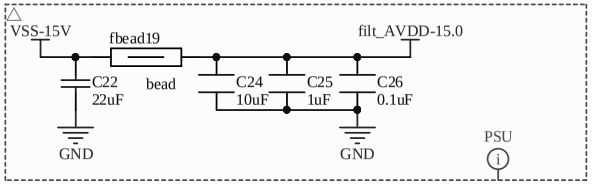
PG-Module

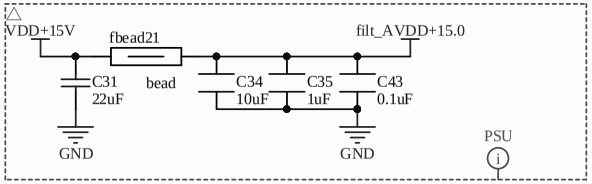


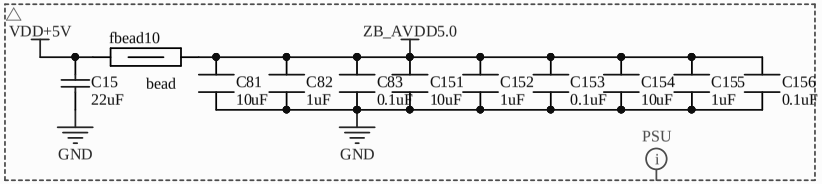
Title: DEATH Project - Power 3		
Size: A4	Number:	Revision: v2.0
Date: 07-Jun-15	Sheet of	
File: D:\Dropbox\..\DEATH_power_3.SchDoc	Drawn By: Ludwig de Clercq	



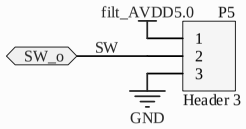








Switch for all power



Title: DEATH Project - Power Connector		
Size: A4	Number:	Revision: v2.0
Date: 07-Jun-15	Sheet of	
File: D:\Dropbox\..\DEATH_power_connector.SchDoc	Drawn By: Ludwig de Clercq	

Bibliography

[Alonso 13] J. Alonso, F.M. Leupold, B.C. Keitch & J.P. Home. *Quantum control of the motional states of trapped ions through fast switching of trapping potentials.* New J. Phys., vol. 15, no. 023001, 2013.

[Baig 13] M. T. Baig, M. Johanning, A. Wiese, S. Heidbrink, M. Ziolkowski & C. Wunderlich. *A scalable, fast and multichannel arbitrary waveform generator.* Rev. Sci. Instrum, 2013.

[Barenco 95] A. Barenco, C. Bennett, R. Cleve, D. DiVincenzoand N. Margolus, P. Shor, T. Sleator, J. Smolin & H. Weinfurter. *Elementary gates for quantum computation.* PRA, vol. 52, page 3457, 1995.

[Barends 15] R. Barends, L. Lamata, J. Kelly, L. Garcia-Alvarez, A.G. Fowler, A. Megrant, E. Jeffrey, T.C. White, D. Sank, J.Y. Mutus, B. Campbell, Yu Chen, Z. Chen, B. Chiaro, A. Dunsworth, I.-C. Hoi, C. Neill, P.J.J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, E. Solano & John M. Martinis. *Digital quantum simulation of fermionic models with a superconducting circuit.* Nat. Commun., vol. 6, no. 7654, 2015.

[Barnes 12] E. Barnes & S. Das Sarma. *Analytically Solvable Driven Time-Dependent Two-Level Quantum Systems.* Phys. Rev. Lett., vol. 109, no. 060401, 2012.

[Barnes 13] E. Barnes. *Analytically solvable two-level quantum systems and Landau-Zener interferometry.* Phys. Rev. A, vol. 88, no. 013818, 2013.

[Barrett 04] M. D. Barrett, J. Chiaverini, T. Schaetz, J. Britton, W. M. Itano, J. D. Jost, E. Knill, C. Langer, D. Leibfried, R. Ozeri & D. J. Wineland. *Deterministic quantum teleportation of atomic qubits.* Nature, vol. 429, pages 737–739, 2004.

[Bartels 95] R. H. Bartels, J. C. Beatty & B. Barsky. An introduction to splines for use in computer graphics and geometric modeling. Elsevier, 1995.

[Benhelm 08] J. Benhelm, G. Kirchmair, C. F. Roos & R. Blatt. *Towards fault-tolerant quantum computing with trapped ions.* Nature Physics, vol. 4, pages 463 – 466, 2008.

[Blakestad 09] R. B. Blakestad, C. Ospelkaus, A. P. VanDevender, J. M. Amini, J. Britton, D. Leibfried & D. J. Wineland. *High-Fidelity Transport of Trapped-Ion Qubits through an X-Junction Trap Array.* Phys. Rev. Lett., vol. 102, page 153002, 2009.

[Blakestad 10] R. B. Blakestad. *Transport of Trapped-Ion Qubits within a Scalable Quantum Processor.* PhD thesis, University of Colorado, 2010.

[Blakestad 11] R. B Blakestad, C. Ospelkaus, A. P VanDevender, J. H Wesenberg, M. J Biercuk, D. Leibfried & D. J Wineland. *Near-ground-state transport of trapped-ion qubits through a multidimensional array.* Phys. Rev. A, vol. 84, no. 032314, 2011.

[Bonato 15] C. Bonato, M. S. Blok, H. T. Dinani, D. W. Berry, M. L. Markham, D. J. Twitchen & R. Hanson. *Optimized quantum sensing with a single electron spin using real-time adaptive measurements.* arXiv preprint arXiv:1508.03983, 2015.

[Bow 15] R. Bow. *Coherent Ion Transport in a Multi-electrode Trap Array.* PhD thesis, University of Colorado, 2015.

[Bowler 12] R. Bowler, J. Gaebler, Y. Lin, T. R. Tan, D. Hanneke, J. D. Jost, J. P. Home, D. Leibfried & D. J. Wineland. *Coherent Diabatic Ion Transport and Separation in a Multizone Trap Array.* Phys. Rev. Lett., vol. 109, page 080502, Aug 2012.

[Bowler 13] R. Bowler, U. Warring, J. W. Britton, B. C. Sawyer & J. Amini. *Arbitrary waveform generator for quantum information processing with trapped ions.* Rev. Sci. Instrum, 2013.

[Boyd 04] S. Boyd & L. Vandenberghe. Convex optimization. Cambridge University Press, 2004.

[Brown 11] K. R. Brown, A. C. Wilson, Y. Colombe, C. Ospelkaus, A. M. Meier, E. Knill, D. Leibfried & D. J. Wineland. *Single-qubit-gate error below* $\mathbf{10^{-4}}$ *in a trapped ion.* Phys. Rev. A, vol. 84, page 030303, Sep 2011.

[Chiaverini 04] J. Chiaverini, D. Leibfried, T. Schaetz, M.D. Barrett, R.B. Blakestad, J. Britton, W.M. Itano, J.D. Jost, E. Knill, C. Langer, R. Ozeri & D.J. Wineland. *Realization of quantum error correction.* Nature, vol. 432, pages 602–605, 2004.

[Chuang 97] I. L. Chuang & M. A. Nielsen. *Prescription for experimental determination of the dynamics of a quantum black box.* J. Mod. Opt., vol. 44, pages 2455–2467, 1997.

[Cirac 95] J. I. Cirac & P. Zoller. *Quantum computations with cold trapped ions.* Phys. Rev. Lett., vol. 74, no. 20, pages 4091–4094, 1995.

[Colombe 14] Y. Colombe, D.H. Slichter, A.C. Wilson, D. Leibfried & D.J. Wineland. *Single-mode optical fiber for high-power, low-loss UV transmission.* Opt. Express, vol. 22, no. 16, pages 19783–19793, Aug 2014.

[Compaq 00] Intel Lucent Microsoft NEC Compaq Hewlett-Packark & Philips. *Universal serial bus specification.* http://sdpha2.ucsd.edu/Lab_Equip_Manuals/usb_20.pdf, 2000.

[D. Hucul 08] S. M. Olmschenk C. R. Monroe W. K. Hensinger D. Hucul M. Yeo & J. Rabchuk. *On the transport of atomic ions in linear and multidimensional ion trap arrays.* Quantum Inf. Comput., vol. 8, pages 501–578, 2008.

[D. Leibfried 07] C. Ospelkaus D. Leibfried E. Knill & D. J. Wineland. *Transport quantum logic gates for trapped ions.* Phys. Rev. A, 2007.

[de Boor] Carl de Boor. *B(asic)-Spline Basics.* http://www.researchgate.net/publication/228398058_B_(asic)-spline_basics. Notes that are kept up to date by the author.

[de Castro 10] R. Rey de Castro & H. Rabitz. *Laboratory implementation of quantum-control-mechanism identification through Hamiltonian encoding and observable decoding.* Phys. Rev A., vol. 81, no. 063422, 2010.

[de Castro 13] R. Rey de Castro, R. Cabrera, D. I Bondar & H. Rabitz. *Time-resolved quantum process tomography using Hamiltonian-encoding and observable-decoding.* New J. Phys., vol. 15, no. 025032, 2013.

[de Clercq 15a] L. de Clercq, H.-Y. Lo, M. Marinelli, D. Nadlinger, R. Oswald, V. Negnevitsky, D. Kienzler, B. Keitch & J.P. Home. *Parallel transport quantum logic gates with trapped ions.* arXiv:1509.06624, 2015.

[de Clercq 15b] L.E. de Clercq, R. Oswald, C. Flühmann, B. Keitch, D. Kienzler, H.-Y. Lo, M. Marinelli, D. Nadlinger, V. Negnevitsky & J.P. Home. *Time-dependent Hamiltonian estimation for Doppler velocimetry of trapped ions.* arXiv:1509.07083, 2015.

[Deutsch 85] D. Deutsch. *Quantum theory and the Church-Turing principle and the universal quantum computer.* Proc. R. Soc. Lond. A, vol. 400, pages 97–117, 1985.

[Devitt 06] S. J. Devitt, J. H. Cole & L. C. L. Hollenberg. *Scheme for direct measurement of a general two-qubit Hamiltonian.* Phys. Rev A., vol. 73, no. 052317, pages 1–5, 2006.

[Diedrich 89] F. Diedrich, J. C. Bergquist, W. M. Itano & D. J. Wineland. *Laser cooling to the zero-point energy of motion.* Phys. Rev. Lett., vol. 62, no. 4, pages 403–406, 1989.

[Feynman 82] R. P. Feynman. *Simulating physics with computers*. International Journal of Theoretical Physics, vol. 21, page 467, 1982.

[Feynman 86] R. P. Feynman. *Quantum mechanical computers*. Found. Phys., vol. 16, pages 507–531, 1986.

[Fischer 15a] Christoph Fischer. *Implementation of a digital lock-in amplifier on a field programmable gate array and its remote control in a local Area*, 2015. Semester project.

[Fischer 15b] Christoph Fischer. Optical trapping in a build-up cavity. Master's thesis, ETH Zürich, 2015.

[Gaebler 12] J. P. Gaebler, A. M. Meier, T. R. Tan, R. Bowler, Y. Lin, D. Hanneke, J. D. Jost, J. P. Home, E. Knill, D. Leibfried & D. J. Wineland. *Randomized Benchmarking of Multiqubit Gates*. Phys. Rev. Lett., vol. 108, page 260503, 2012.

[Gebert 14] F. Gebert, M. H. Frosz, T. Weiss, Y. Wan, A. Ermolov, N. Y. Joly, P. O. Schmidt & P. St. J. Russell. *Damage-free single-mode transmission of deep-UV light in hollow-core PCF*. Opt. Express, vol. 22, no. 13, pages 15388–15396, Jun 2014.

[Grover 97] L. K. Grover. *Quantum mechanics helps in searching for a needle in a haystack*. Phys. Rev. Lett., vol. 79, pages 325–328, 1997.

[Guéry-Odelin 14] D. Guéry-Odelin & J. G. Muga. *Transport in a harmonic trap: Shortcuts to adiabaticity and robust protocols*. Phys. Rev. A, vol. 90, no. 063425, 2014.

[Gurobi Optimization 15] Inc. Gurobi Optimization. *Gurobi optimizer reference manual*. http://www.gurobi.com, 2015.

[Hanneke 09] D. Hanneke, J. P. Home, J. D. Jost, J. M. Amini, D. Leibfried & D. J. Wineland. *Realization of a programmable two-qubit quantum processor*. Nature Physics, vol. 6, pages 13–16, 2009.

[Hansen 98] Christian Hansen. *Regularization Tools: A MATLAB package for analysis and solution of discrete ill-posed problems. Version 4.1.*

http://ch.mathworks.com/matlabcentral/fileexchange/52-regtools, 1998. Last updated Feb 2015.

[Harty 14] T.P. Harty, D.T.C Allcock, C.J. Ballance, L. Guidoni, H.A. Janacek, N.M. Linke, D.N. Stacey & D.M. Lucas. *High-Fidelity Preparation, Gates, Memory, and Readout of a Trapped-Ion Quantum Bit.* Phys. Rev. Lett., vol. 113, page 220501, Nov 2014.

[Home 06] J. P. Home & A. Steane. *Electrode Configurations for Fast Separation of Trapped Ions.* Quantum Information and Computation, vol. 6, page 289, 2006.

[Home 09] J. P. Home, D. Hanneke, J. D. Jost, J. M. Amini, D. Leibfried & D. J. Wineland. *Complete Methods Set for Scalable Ion Trap Quantum Information Processing.* Science, vol. 325, page 1227, 2009.

[Hucul 06] D. A. Hucul. Operation of a two- dimensional ion trap array for scalable quantum computation. Master's thesis, University of Michigan, 2006.

[Hungenberg 13] Alexander Hungenberg. *Automatic relocking of an FPGA-based PID controller using a bandpass-ltering approach.* Semester Thesis, 2013.

[Institute] Georgia Tech Research Institute. *Automated waveform generator.* http://www.quantum.gatech.edu/automatedWaveformGenerator.shtml.

[Jost 09] J. D. Jost, J. P. Home, J. M. Amini, D. Hanneke, R. Ozeri, C. Langer, J. J. Bollinger, D. Leibfried & D. J. Wineland. *Entangled Mechanical Oscillators.* Nature, vol. 459, pages 683–685, 2009.

[Kaufmann 14] H. Kaufmann, T. Ruster, C.T. Schmiegelow, F. Schmidt-Kaler & U.G. Poschinger. *Dynamics and control of fast ion crystal splitting in segmented Paul traps.* New Journal of Physics, vol. 16, no. 7, page 073012, 2014.

[Keitch 07] Ben Keitch. *A quantum memory qubit in calcium-43 - Experimental Systems and Control.* PhD thesis, Oxford, 2007.

[Kielpinski 02] D. Kielpinski, C.Monroe & D. Wineland. *Architecture for a large-scale ion-trap quantum computer.* Nature, vol. 417, pages 709–711, 2002.

[Kienzler 15] Daniel Kienzler. *Quantum harmonic oscillator state synthesis by reservoir engineering.* PhD thesis, ETH Zürich, 2015.

[Knill 05] E. Knill. *Quantum Computation with realistically noisy devices.* Nature, vol. 434, pages 39–44, 2005.

[Knill 08] E. Knill, D. Leibfried, R. Reichle, J. Britton, R. B. Blakestad, J. D. Jost, C. Langer, R. Ozeri, S. Seidelin & D. J. Wineland. *Randomized benchmarking of quantum gates.* Phys. Rev. A, vol. 77, page 012307, Jan 2008.

[Knill 10] Emanuel Knill. *Quantum computing.* Nature, vol. 463, pages 441–443, 2010.

[Langer 05a] C. Langer, R. Ozeri, J.D. Jost, J. Chiaverini, B.L. DeMarco, A. Ben-Kish, R.B. Blakestad, J. Britton, D. Hume, W.M. Itano, D. Leibfried, R. Reichle, T. Rosenband, T. Schaetz, P.O. Schmidt & D.J. Wineland. *Long-Lived Qubit Memory Using Atomic Ions.* Phys. Rev. Lett., vol. 95, page 060502(4), 2005.

[Langer 05b] C. E. Langer. *High Fidelity Quantum Information Processing with Trapped Ions.* PhD thesis, University of Colorado, 2005.

[Lau 11] Hoi-Kwan Lau & Daniel F. V. James. *Decoherence and dephasing errors caused by the dc Stark effect in rapid ion transport.* Phys. Rev. A, vol. 83, page 062330, Jun 2011.

[Leibfried 03] D. Leibfried, R. Blatt, C. Monroe & D. Wineland. *Quantum dynamics of single trapped ions.* Rev. Mod. Phys., vol. 75, pages 281–324, 2003.

[Lo 15] Hsiang-Yu Lo. *Creation of squeezed schrödinger's cat states in a mixed-species ion trap.* PhD thesis, ETH Zürich, 2015.

[Lu 14] Xiao-Jing Lu, J. G. Muga, Xi Chen, . G. Poschinger, F. Schmidt-Kaler & A. Ruschhaupt. *Fast shuttling of a trapped ion in the presence of noise.* Phys. Rev. A, vol. 89, no. 063414, 2014.

[Lucas 04] D.M. Lucas, A. Ramos, J.P. Home, M.J. McDonnell, S. Nakayama, J.-P. Stacey, S.C. Webster, D.N. Stacey & A.M. Steane. *Isotope-selective photoionisation for calcium ion trapping.* Phys. Rev. A, vol. 69, page 012711, 2004.

[Manin 80] Yuri Manin. *Computable and Uncomputable (in Russian).* Sovetskoye Radio, 1980.

[Martinis 14] J.M. Martinis & M.R. Geller. *Fast adiabatic gates using only σ_z control.* Phys. Rev A., vol. 90, no. 022307, 2014.

[Mathworks] Mathworks. http://uk.mathworks.com/help/control/ref/pid.html?refresh=true.

[Mathworld] Wolfram Mathworld. *Newton's forward difference formula.* http://mathworld.wolfram.com/NewtonsForwardDifferenceFormula.html. Online notes from Wolfram Mathworld.

[Mehta 14] K. K. Mehta, A. M. Eltony, C. D. Bruzewicz, I. L. Chuang, R. J. Ram, J. M. Sage & J. Chiaverini. *Ion traps fabricated in a CMOS foundry.* Applied Physics Letters, vol. 105, no. 4, pages –, 2014.

[Mitra 03] A. Mitra & H. Rabitz. *Identifying mechanisms in the control of quantum dynamics through Hamiltonian encoding.* Phys. Rev A., vol. 67, no. 033407, 2003.

[Moehring 07] D. L. Moehring, P. Maunz, S. Olmschenk, K. C. Younge, D. N. Matsukevich, L.-M. Duan & C. Monroe. *Entanglement of single-atom quantum bits at a distance.* Nature, vol. 449, page 68, 2007.

[Monroe 95] C. Monroe, D. M. Meekhof, B. E. King, W. M. Itano & D. J. Wineland. *Demonstration of a fundamental quantum logic gate.* Phys. Rev. Lett., vol. 75, no. 25, pages 4714–4717, 1995.

[Murphy 12] K. P. Murphy. Machine learning: a probabilistic perspective. MIT press, 2012.

[Muske 95] K. R. Muske & J. B. Rawlings, editeurs. Methods of model based process control. Springer Netherlands, 1995.

[Nadlinger 13] David Nadlinger. Laser intensity stabilization and pulse shaping for trapped-ion experiments using acousto-optic modulators. Master's thesis, ETH Zürich, 2013.

[Neuhauser 78] W. Neuhauser, M. Hohenstatt, P. Toschek & H. Dehmelt. *Optical-sideband cooling of visible atom cloud confined in parabolic well.* Phys. Rev. Lett., vol. 41, pages 233–236, 1978.

[Neuhauser 80] W. Neuhauser, M. Hohenstatt, P. E. Toschek & H. Dehmelt. *Localized visible Ba^+ mono-ion oscillator.* hys. Rev. A, vol. 22, page 1137, 1980.

[Palmero 14] M. Palmero, R. Bowler, J. P. Gaebler, D. Leibfried & J. G. Muga. *Fast transport of mixed-species ion chains within a Paul trap.* Phys. Rev. A, vol. 90, no. 053408, 2014.

[Pedersen 07] Line Hjortshoej Pedersen, Niels Martin Moeller & Klaus Moelmer. *Fidelity of quantum operations.* Physics Letters A, vol. 367, pages 47 – 51, 2007.

[Poschinger 14] U. Poschinger & V. Kaushal. Private communication, 2014.

[Poyatos 97] J. F. Poyatos, J. I. Cirac & P. Zoller. *Complete characterization of a quantum process: the two-bit quantum gate.* Phys. Rev. Lett., vol. 78, no. 2, pages 390–393, 1997.

[Preskill 98] John Preskill. *Reliable quantum computing.* Proc. R. Soc. London, Ser. A, pages 385–410, 1998.

[Reichle 06] R. Reichle, D. Leibfried, R. B. Blakestad, J. Britton, J. D. Jost, E. Knill, C. Langer, R. Ozeri, S. Seidelin & D. J. Wineland. *Transport dynamics of single ions in segmented microstructured Paul trap arrays.* Fortschr. Phys., 2006.

[Riebe 06] M. Riebe, K. Kim, P. Schindler, T. Monz, P. O. Schmidt, T. K. Körber, W. Hänsel, H. Häffner, C. F. Roos & R. Blatt. *Process Tomography of Ion Trap Quantum Gates.* Phys. Rev. Lett., vol. 97, page 220407, 2006.

[Roa] http://qurope.eu/content/Roadmap.

[Roa 04] http://qist.lanl.gov/, 2004.

[Roos 00] C. F. Roos, D. Leibfried, A. Mundt, F. Schmidt-Kaler, J. Eschner & R. Blatt. *Experimental demonstration of ground state laser cooling with electromagnetically induced transparency.* Phys. Rev. Lett., vol. 85, no. 26, pages 5547–5550, 2000.

[Rowe 02] M. A. Rowe, A. Ben-Kish, B. DeMarco, D. Leibfried, V. Meyer, J. Beall, J. Britton, J. Hughes, W. M. Itano, B. Jelenković, C. Langer, T. Rosenband & D. J. Wineland. *Transport of quantum states and separation of ions in a dual rf ion trap.* Quantum Information and Computation, vol. 2, page 257, 2002.

[Ruster 14] T. Ruster, C. Warschburger, H. Kaufmann, C. T. Schmiegelow, M. Hettrich A. Walther, A. Pfister, V. Kaushal, F. Schmidt-Kaler & U. G. Poschinger. *Experimental realization of fast ion separation in segmented Paul traps.* Phys. Rev. A, 2014.

[Saleh 07] B.E.A. Saleh & M.C. Teich. Fundamentals of photonics. Wiley, 2007.

[Sasura 02] M. Sasura & V. Buzek. *Cold trapped ions as quantum information processors.* J. Mod. Opt., vol. 49, no. 10, 2002.

[Schulz 06] S. Schulz, U. Poschinger, K. Singer & F. Schmidt-Kaler. *Optimization of segmented linear Paul traps and transport of stored particles.* Fortschritte der Physik, vol. 54, no. 8-10, pages 648–665, 2006.

[Schutjens 13] R. Schutjens, F. Abu Dagga, D.J. Egger & F.K. Wilhelm. *Single-qubit gates in frequency-crowded transmon systems.* Phys. Rev. A, vol. 88, page 052330, Nov 2013.

[Shabani 11] A. Shabani, M. Mohseni, S. Lloyd, R. L. Kosut & H. Rabitz. *Estimation of many-body quantum Hamiltonians via compressive sensing.* Phys. Rev A., vol. 84, no. 012107, pages 1–8, 2011.

[Shor 94] P. W. Shor. *Polynomial-time algorithms for prime factorisation and discrete logarithms on a quantum computer.* In Proceedings of the 35th annual symposium on foundations of computer science, pages 124–134. IEEE Computer Society Press, 1994.

[Singer 10] K. Singer, U. Poschinger, M. Murphy, P. Ivanov, F. Ziesel, T. Calarco & F. Schmidt-Kaler. *Colloquium: Trapped ions as quantum bits: Essential numerical tools.* Rev. Mod. Phys., vol. 82, pages 2609–2632, Sep 2010.

[Steane 03] Andrew M. Steane. *Overhead and noise threshold of fault-tolerant quantum error correction.* Phys. Rev. A, vol. 68, page 042322, 2003.

[Steane 07] A. Steane. *How to build a 300 bit and 1 Gigaop quantum computer.* Quant. Inf. and Comp., vol. 7, page 171, 2007.

[Walther 12] A. Walther, F. Ziesel, T. Ruster, S. T. Dawkins, K. Ott, M. Hettrich, K. Singer, F. Schmidt-Kaler & U. Poschinger. *Controlling Fast Transport of Cold Trapped Ions.* Phys. Rev. Lett., vol. 109, page 080501, Aug 2012.

[Wineland 78] D. J. Wineland, R. E. Drullinger & F. L. Walls. *Radiation-Pressure Cooling of Bound Resonant Absorbers.* Phys. Rev. Lett., vol. 40, pages 1639–1642, 1978.

[Wineland 98] D. J. Wineland, C. Monroe, W. M. Itano, D. Leibfried, B. E. King & D. M. Meekhof. *Experimental issues in coherent quantum-state manipulation of trapped atomic ions.* J. Res. Natl. Inst. Stand. Technol., vol. 103, pages 259–328, 1998.

[Wineland 03] D. J. Wineland, M. Barrett, J. Britton, J. Chiaverini, B. DeMarcoand W. M. Itano, B. Jelenkovic, C. Langer, D. Leibfried, V. Meyer & T. Rosenbandand T. Schätz. *Quantum information processing*

with trapped ions. Phil. Trans. R. Soc. Lond. A, vol. 361, pages 1349–1361, 2003.

[Wright 13] K. Wright, J. M Amini, D.L. Faircloth, C. Volin, S.C. Doret, H. Hayden, C-S Pai, D.W. Landgren, D. Denison, T. Killian, R.E. Slusher & A.W. Harter. *Reliable transport through a microfabricated X-junction surface-electrode ion trap.* New J. Phys., vol. 15, no. 033004, 2013.

[Xilinx 13] Xilinx. *Spartan-3E FPGA Family Data Sheet*, 2013.

[Zhang 14] J. Zhang & M.Sarovar. *Quantum Hamiltonian identification from measurement time traces.* Phys. Rev. Lett., vol. 113, no. 080401, 2014.

[Zumbahlen 07] Hank Zumbahlen, editeur. Basic linear design. Analog Devices, 2007.