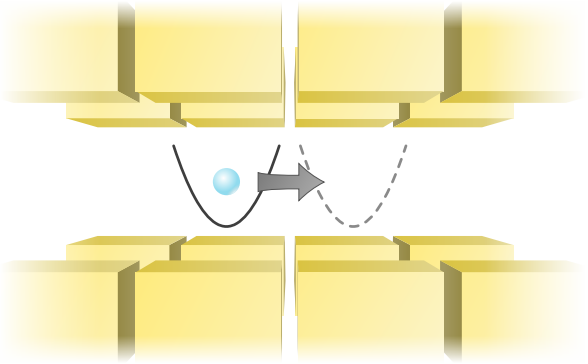
Diplomarbeit

Schneller Transport von kalten gefangenen Ionen

Thomas Ruster

21. Dezember 2012



JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

Erstgutachter: Prof. Dr. Ferdinand Schmidt-Kaler
Zweitgutachter: Prof. Dr. Peter van Loock

Zusammenfassung

Transporte von gefangenen Ionen sind eine Voraussetzung zur Realisierung eines skalierbaren Quantencomputers in einer segmentierten Mikro-Ionenfalle, in der grundzustandsgekühlte Ionen zwischen Speicher- und Prozessorregionen hin- und hertransportiert werden müssen. Um die höchstmögliche Rechenleistung zu erzielen, müssen die Transporte im nichtadiabatischen Regime durchgeführt werden, d.h. mit einer Transportdauer in der Größenordnung der Schwingungsdauer eines gefangenen Ions.

Das ist im Rahmen dieser Diplomarbeit gelungen. In einem Zeitraum von fünf Schwingungszyklen ($\approx 3{,}6\,\mu$s) wurden Ionen nahe am quantenmechanischen Grundzustand um mehr als die 10^4-fache Ausdehnung des Grundzustandswellenpakets ($\approx 280\,\mu$m) verschoben, mit einer Anregung des Bewegungszustandes auf eine mittlere Phononenzahl von weniger als 0,20. Darüber hinaus konnte gezeigt werden, dass die sowohl im Spin- als auch im Bewegungszustand kodierte Quanteninformation nach dem Transport erhalten bleibt. Die Erfahrungen, die bei der präzisen Kontrolle von Ionen während der Transportexperimente gesammelt wurden, konnten genutzt werden, um kohärent verschobene Fock-Zustände mit einer Kombination von optischen und elektrischen Manipulationen herzustellen.

Ein weiteres Kernthema dieser Diplomarbeit war die Weiterentwicklung der zur Steuerung des Experiments genutzten Software. Unter anderem wurden erweiterte experimentelle Pulssequenzen wie die neu entwickelten Pseudo-Energiemessung implementiert. Diese beschleunigte die Messung des Bewegungszustandes von Ionen nach durchgeführten Transporten um einen Faktor 100 und trug somit wesentlich zum Erfolg der Experimente bei.

Durch die stark erweiterte Software wurde weiterhin eine neue Klasse von Messungen ermöglicht, durch die wichtige experimentelle Parameter wie zum Beispiel Laserfrequenzen oder Pulsflächen vollautomatisch kalibriert werden können - eine Investition für zukünftige komplexere Experimente, in die immer mehr Parameter einfließen werden, welche ständig aufgrund von Drifts nachjustiert werden müssen.

Die Ergebnisse dieser Diplomarbeit können ebenfalls in den Veröffentlichungen „Controlling fast transport of cold trapped ions“ [WZR+12] und „Experimental creation and analysis of displaced number states“ [ZRW+12] nachgelesen werden.

Abstract

Transport of trapped ions is a prerequisite for realizing a scalable quantum computer in a segmented microstructured Paul trap. Information processing can be accomplished by moving ions that have been cooled close to the ground state between memory regions and interaction regions, where logic operations are performed. To achieve the best possible performance, one has to carry out transports in a nonadiabatic way, i.e. with a duration that is comparable to the oscillation period of a trapped ion.

Such transports were successfully performed during the work for this thesis. Ions were shuttled over a distance of more than 10^4 times the size of the ground state wave function ($\approx 280\,\mu$m) during only five motional cycles ($\approx 3.6\,\mu$s), with a mean excitation of less than 0.20 motional quanta after transport. Furthermore, it is shown that quantum information stored in both the spin state and in the motional state of ions is preserved throughout the transport. The precise control over the motional state of ions in our experiment additionally enabled us to create displaced number states using a combination of optical and electrical manipulations.

A further goal of this thesis was to enhance the existing experimental control software. The added functionality made the implementation of advanced experimental sequences possible, such as the newly created pseudoenergy measurement, which accelerated measurements of the motional state of ions by a factor of 100 and was thus a key ingredient to the success of the transport measurements.

Using the enhanced software, it was possible to design automated measurement schemes for calibration of experimentally relevant parameters such as laser frequencies or pulse areas. These will be especially important for more complex experiments in the future, which will require a significant overhead of calibration measurements.

The results of this thesis are also presented in the publications "Controlling fast transport of cold trapped ions" [WZR+12] and "Experimental creation and analysis of displaced number states" [ZRW+12].

Inhaltsverzeichnis

1. Einführung **9**

2. Theoretische Grundlagen **13**
2.1. Bewegung eines klassischen gefangenen Teilchens in einer linearen Paulfalle 13
2.2. Licht-Ionen-Wechselwirkung mit gefangenen Ionen 14
2.3. Kalzium-Ionen als Quantenbits 19

3. Experimenteller Aufbau **23**
3.1. Die segmentierte Mikrofalle 23
3.2. Konfiguration der Laserstrahlen 24
3.3. Fallen-Spannungsversorgung 26
3.4. Experimentelles Kontrollsystem 27

4. Experimentelle Sequenzen **31**
4.1. Optimierung der Dopplerkühlung und Ionenfluoreszenz 32
4.2. Messung und Minimierung der Mikrobewegung 34
4.3. Shelving 36
4.4. Seitenbandkühlen 39
4.5. Raman-Kalibrationssequenzen 40
4.6. Detektionsverfahren 47

5. Realisierung von schnellen Ionentransporten **61**
5.1. Theorie der kohärenten Zustände 62
5.2. Bewegungsgleichung eines transportierten Ions 63
5.3. Transport-Trajektorien 65
5.4. Erzeugung von Transportspannungen 67
5.5. Die Pseudo-Energie 69
5.6. Experimentelle Ergebnisse 71
5.7. Erhaltung der Quanteninformation 74
5.8. Präparation kohärent verschobener Fockzustände 78

6. Ausblick **83**

A. Software-Dokumentation **85**
A.1. Das Master-Control-Programm 85
A.2. Dokumentation der Kamera-Software 86
A.3. Struktur der Mikrofallen-Script.Dll 94

B. Heizmessungen an einem akustooptischen Modulator **113**

1

Einführung

In den letzten Jahrzehnten durchlief die Informationstechnologie eine beispiellose Entwicklung. Computerchips werden heute in großen Massen produziert - und Jahr für Jahr werden diese immer leistungsfähiger. Gordon Moore schätzte erstmals 1965 [Moo65] im nach ihm benannten Gesetz, dass sich die Komplexität, d.h. die Anzahl der Transistoren auf einem Computerchip, innerhalb von 12 Monaten verdoppelt. Zwar musste er diese Vorhersage später auf eine Verdopplung alle 24 Monate korrigieren [Moo75], jedoch konnte dieser Trend dann für die letzten 40 Jahre erfüllt werden (Abbildung 1.1). Grund für diese Entwicklung ist die Möglichkeit, Transistoren in immer kleineren Strukturen herzustellen: Heutzutage können Prozessoren in 22 nm-Strukturbreite in Massenfertigung produziert werden. Diese Entwicklung kann jedoch nicht endlos fortgesetzt werden: Je kleiner die produzierten Strukturen werden, desto stärker spielen quantenmechanische Effekte eine Rolle. Spätestens die Größenordnung einzelner Atome bildet eine unüberwindbare Barriere. Moore selbst schätzt, dass sein Gesetz in den Jah-

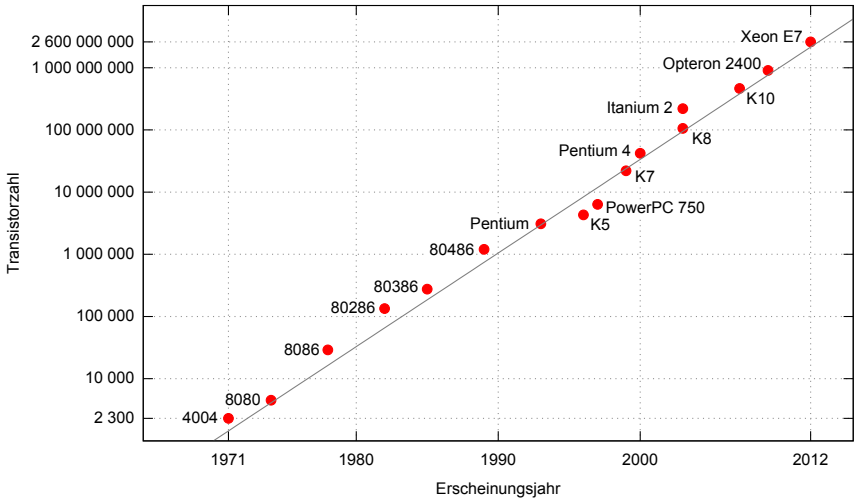


Abbildung 1.1.: Entwicklung der Transistoranzahl von Mikroprozessoren. Die Transistorzahlen sind Angaben der jeweiligen Hersteller. Die schwarze Linie zeigt eine Verdopplung der Transistorzahl alle zwei Jahre.

ren zwischen 2020 und 2030 seine Gültigkeit verlieren wird [Dub05]. Eine Möglichkeit, die Entwicklung in der Informationstechnologie darüber hinaus fortzusetzen, bietet der Quantencomputer. Feynman schlug erstmals 1986 vor, Computer bestehend aus quantenmechanischen Elementen aufzubauen [Fey86]. Die Nutzung der Quantenmechanik erlaubt nicht nur eine Verkleinerung der dem Computer zugrundeliegenden Bauteile, sondern auch die Realisierung neuer Algorithmen, wie z.B. die von Deutsch [DJ92], Grover [Gro96] und Shor [Sho97]. Besonders der von Shor vorgestellte Algorithmus stößt auf ein riesiges öffentliches Interesse - mit ihm lassen sich große Zahlen effizient faktorisieren und somit die heute üblichen Kryptographie-Methoden knacken. Der tiefere Grund der erweiterten Möglichkeiten von Quantenalgorithmen liegt im Superpositionsprinzip der Quantenmechanik: Während ein klassisches Bit genau die Werte 0 und 1 annehmen kann, kann sich ein Quantenbit, oder Qubit, in einem Überlagerungszustand

$$|\Phi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

befinden. Der Zustand eines Systems aus zwei Qubits

$$|\Phi\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle$$

ist im Allgemeinen verschränkt, d.h. er lässt sich nicht mehr in die Einzelzustände der einzelnen Qubits separieren. Für diese Art von Zuständen gibt es in der klassischen Physik kein Analogon, was bedeutet, dass Quantencomputer prinzipiell leistungsfähiger sein können als klassisch arbeitende Computer. DiVincenzo formulierte mehrere Kriterien [Div00], die ein physikalisches System zur konkreten Implementierung eines Quantencomputers erfüllen muss:

1. Eine im Prinzip beliebig hohe Anzahl an wohldefinierten Qubits.
2. Initialisierung in einen bestimmten Anfangszustand, z.B. $|0000\ldots\rangle$.
3. Ein universeller Satz von Quantengattern, d.h. beliebige unitäre Operationen auf N Qubits. Jedes N-Qubit-Gatter lässt sich aus Einzelqubit-Operationen und dem CNOT-Zwei-Qubit-Gatter zusammensetzen [BBC+95].
4. Eine Dekohärenzzeit viel größer als die Dauer einer Gatteroperation.
5. Messung aller Qubits.

Ein vielversprechender Ansatz zur Erfüllung aller Kriterien stellen einzelne gefangene Ionen in linearen Paulfallen dar [CZ95]. Qubits werden hier aus zwei langlebigen internen elektronischen Zuständen der Ionen gebildet, Gatteroperationen und Detektion werden durch Laser-Wechselwirkung realisiert. Bei richtiger Wahl der Fallenspannungen ordnen sich die Ionen entlang einer linearen Kette an und sind über einen gemeinsamen Bewegungszustand miteinander gekoppelt. Dieser lässt sich durch Laserwechselwirkung beeinflussen, was Gatteroperationen mit mehreren Ionen ermöglicht. Ein großes Problem allerdings stellt die Skalierung des Systems hin zu vielen Qubits dar: Je mehr Ionen gefangen werden, desto stärker muss der radiale Einschluss sein, damit sich die Ionen

entlang einer Kette ausrichten. Gleichzeitig rücken die Ionen mit steigender Zahl immer näher zusammen, was die Adressierung und Detektion einzelner Ionen erschwert. Des weiteren nimmt die Anzahl an Schwingungsmoden der Bewegung linear mit der Anzahl der Ionen zu, sodass die Adressierung individueller Schwingungsmoden im Frequenzraum immer schwieriger wird. Zur Lösung dieser Probleme schlugen Kielpinksi, Monroe und Wineland [KMW02] vor, eine lineare Falle in mehrere Segmente zu unterteilen, die jede für sich selbst eine Paulfalle bildet, in der nur einzelne oder zwei Ionen gefangen werden müssen. Die segmentierte Falle kann dann in verschiedene Regionen unterteilt werden: Speicherregionen zur Aufbewahrung von Ionen, und Prozessorregionen, wo Laserwechselwirkungen stattfinden. Durch Änderungen an den Fallenspannungen können Ionen von einer Region in eine andere transportiert werden. Eine solche segmentierte Falle wird auch in unserem Experiment eingesetzt.

Da Transporte wie Gatteroperationen zur Dauer eines Quantenalgorithmus beitragen, müssen sie möglichst schnell durchgeführt werden können - in jedem Falle schnell gegenüber der Dekohärenzzeit des Systems. Als weitere Voraussetzung kommt hinzu, dass Transporte den Bewegungszustand der Ionen nur geringfügig anregen dürfen. Ansonsten können Zwei-Qubit-Gatteroperationen wie z.B. das geometrische Phasengatter nicht mehr durchgeführt werden, da diese Gatter voraussetzen, dass die Ionen nahe des Bewegungsgrundzustandes gekühlt sind [LBD+05].

Die ersten Transporte in einer segmentierten Ionenfalle wurden experimentell von Rowe et al. [RBKD+02] im *adiabatischen Regime* durchgeführt, d.h. mit einer Transportdauer, die lang gegenüber der axialen Schwingungsdauer der gefangenen Ionen ist. Adiabatische Transporte wurden später auch von Home et al. [HHJ+09] und Blakestad et al. [BOV+09, BOV+11] demonstriert. Im Gegensatz zu Transporten im adiabatischen Regime verursachen schnellere Transporte eine signifikante Anregung des Bewegungszustandes der Ionen, in Messungen von Huber et al. [HDS+08] um mehrere Milli-Elektronenvolt.

Die Realisierung von schnellen Transporten, bei denen die Ionen nach Abschluss wieder in den Bewegungsgrundzustand zurückkehren, verlangt eine genaue Wahl der Segmentspannungen während des Transports. Das ist im Verlauf dieser Diplomarbeit in Modellexperimenten mit einem einzelnen Ion gelungen. Entscheidend für den Erfolg war nicht nur leistungsfähige Hardware, vor allem eine schnelle und rauscharme Spannungsversorgung für mehrere Fallenelektroden, sondern auch ein flexibles Softwaresystem auf einem Kontrollrechner zur Steuerung aller Hardwarekomponenten. Auch im Hinblick auf zukünftige Experimente wird einer ausgeklügelten Steuerungssoftware eine zentrale Bedeutung zukommen. Bei Experimenten mit vielen Ionen wird es immer wichtiger werden, verschiedene Einstellungen wie Laserfrequenzen oder Transportparameter automatisch zu kalibiereren. Das Design und der Test von automatischen Messsequenzen und nicht zuletzt die Durchführung von Messungen zu schnellen Ionentransporten waren die Kernthemen dieser Diplomarbeit.

In Kapitel 2 werden zunächst die zum Verständnis notwendigen theoretischen Grundlagen der linearen Paulfalle sowie der Wechselwirkung zwischen Licht und gefangen Ionen besprochen. Kapitel 3 gibt einen Überblick über den Aufbau des Experiments, wobei besonderes Augenmerk auf die für den Ionentransport wichtigen Komponenten gesetzt wird. Daraufhin werden in Kapitel 4 die generelle Struktur von experimentellen Sequenzen und im speziellen die nötigen Schritte zur Vorbereitung eines Transportexperiments erklärt. Kapitel 5 gliedert sich in zwei Teile: Im ersten Teil wird zunächst theoretisch besprochen, wie sich der Bewegungszustand von Ionen während Transportoperationen verhält und auf welche Weise die Anregung des Zustandes möglichst klein gehalten werden kann. Der zweite Teil stellt dann die eigentlichen Messergebnisse vor.

2

Theoretische Grundlagen

Dieses Kapitel gibt einen Überblick über die für unser Experiment relevanten theoretischen Grundlagen der Paufalle und der Licht-Ionen-Wechselwirkung. Auf komplexe Rechnungen wird in dieser Arbeit weitesgehend verzichtet, für tiefere Details sei hier auf die entsprechenden im Text angegebenen Quellen verwiesen. Die Theorie der Licht-Ionen-Wechselwirkung ist an der Ausarbeitung von Leibfried et al. [LBMW03] orientiert, dort finden sich bei Bedarf auch weitere Informationen. Darüber hinaus wird kurz auf die Eigenschaften der in unserem Experiment verwendeten Kalzium-Ionen eingegangen.

Neben der Licht-Ionen-Wechselwirkung können Ionen in unserem Experiment auf eine zusätzliche Art und Weise beeinflusst werden: Durch die Änderung des elektrischen Potentials, in denen sich die Ionen befinden. Das ist vor allem für die Durchführung von Transporten relevant und wird daher in Kapitel 5 behandelt.

2.1. Bewegung eines klassischen gefangenen Teilchens in einer linearen Paulfalle

In der typischen Konfiguration einer linearen Paulfalle werden vier stabförmige Elektroden gemäß Abbildung 2.1 angeordnet. Auf zwei sich gegenüberliegenden Elektroden

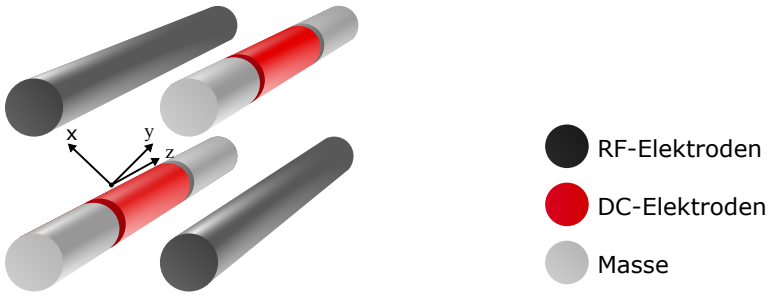


Abbildung 2.1.: Skizze einer linearen Paulfalle

wird eine Wechselspannung mit Radiofrequenz (RF) angelegt, während die anderen beiden Elektroden in Segmente geteilt sind, wobei auf die jeweils mittleren Segmente eine Gleichspannung (DC) und auf die Randsegmente Masse angelegt wird. Mit dieser Elektrodenkonfiguration können Ionen entlang der z-Achse, die auch Fallenachse genannt wird, gespeichert werden. Sie erzeugt in der Nähe der Fallenmitte ein Quadrupolpotential

$$\phi(x,y,z,t) = \frac{U_{\mathrm{dc}}}{2}\left(\alpha_x x^2 + \alpha_y y^2 + \alpha_z z^2\right) + \frac{U_{\mathrm{rf}}}{2}\cos(\omega_{\mathrm{rf}} t)\left(\beta_x x^2 + \beta_y y^2 + \beta_z z^2\right) \quad (2.1)$$

mit U_{dc} bzw. U_{rf} den an den Elektroden angelegten Spannungen, ω_{rf} der Frequenz der Wechselspannung und geometrieabhängigen Konstanten, für die in der linearen Falle aufgrund der Laplace-Gleichung $\Delta\phi = 0$ die Bedingungen

$$\alpha_x + \alpha_y = -\alpha_z \qquad \text{und} \qquad \beta_x = -\beta_y, \beta_z = 0 \quad (2.2)$$

gelten. Die klassische Bewegungsgleichung für ein geladenes Teilchen mit der Ladung $Z\,|e|$ ergibt für jede Raumdimension die Mathieu-Differenzialgleichung

$$\frac{\mathrm{d}^2 u}{\mathrm{d}\tau^2} + (a_u - 2q_u\cos(2\tau))\,u = 0 \qquad \text{mit } u = x, y, z \quad (2.3)$$

mit den Definitionen

$$\tau = \frac{\omega_{\mathrm{rf}} t}{2}\,, \qquad a_u = \frac{4Z|e|U_{\mathrm{dc}}\alpha_u}{m\omega_{\mathrm{rf}}^2}\,, \qquad q_u = \frac{2Z|e|U_{\mathrm{rf}}\beta_u}{m\omega_{\mathrm{rf}}^2} \quad (2.4)$$

Im Grenzfall $a_u < q_u^2 \ll 1$ erhält man die stabile Lösung

$$u(t) \approx u_0 \cos\left(\gamma_u \frac{\omega_{\mathrm{rf}}}{2} t\right)\left(1 + \frac{q_u}{2}\cos(\omega_{\mathrm{rf}} t)\right) \quad (2.5)$$

mit $\gamma_u \approx \sqrt{a_u + q_u^2/2}$. Diese Lösung ist in Abbildung 2.2 dargestellt: Eine langsame Schwingung mit der Frequenz $\gamma_u\omega_{\mathrm{rf}}/2$, die *Säkularbewegung*, ist von einer Schwingung mit kleiner Amplitude und der Frequenz ω_{rf} überlagert, der *Mikrobewegung*. Die Amplitude der Mikrobewegung hängt von der Auslenkung des Teilchens aus der Fallenmitte ab. In axialer Richtung, d.h. entlang der z-Achse, verschwindet die Mikrobewegung, da $\beta_z = 0$. Entlang dieser Achse verhält sich das System wie ein gewöhnlicher harmonischer Oszillator.

2.2. Licht-Ionen-Wechselwirkung mit gefangenen Ionen

Die Wechselwirkung zwischen Licht und gefangenen Ionen ist für unser Experiment von fundamentaler Bedeutung. Laser werden verwendet zur Kühlung, Präparation, Detektion und zur Manipulation der Ionen. Der Unterschied zur Licht-Atom-Wechselwirkung

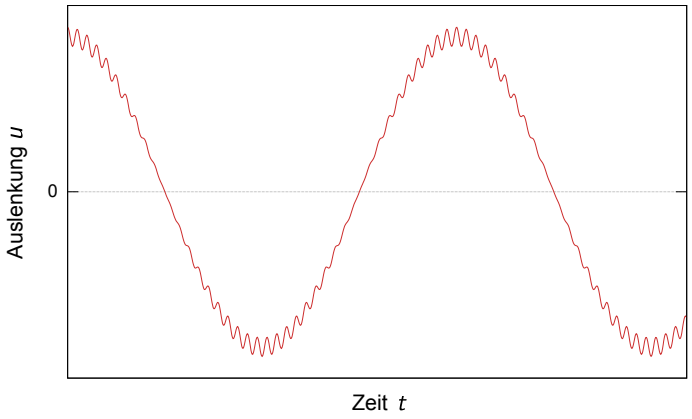


Abbildung 2.2.: Lösung der Mathieu-Differenzialgleichung in einer Raumdimension

mit freien Atomen besteht darin, dass der Bewegungszustand gefangener Ionen einem harmonischen Oszillator entspricht. Dieser wird quantenmechanisch durch den Hamiltonoperator

$$\hat{\mathcal{H}}_{\text{Bewegung}} = \hbar\omega_z \left(\hat{a}^\dagger \hat{a} + \frac{1}{2}\right) \tag{2.6}$$

beschrieben mit $\hat{a}^\dagger$ bzw. $\hat{a}$ den Auf- bzw. Absteigeoperatoren. Der Einfachheit halber wird im Folgenden nur die Bewegung entlang der Fallenachse z betrachtet. Die Frequenz ω_z nennt man axiale Fallenfrequenz.

Die interne elektronische Struktur des Ions wird als Zwei-Niveau-System mit den Niveaus $|g\rangle$ und $|a\rangle$ aufgefasst. Diese Näherung ist zulässig, sofern das eingestrahlte Licht nah an der Resonanz mit genau einem atomaren Übergang ist. Der Energienullpunkt sei so gewählt, dass der Zustand $|g\rangle$ die Energie $-\frac{\omega_0}{2}$ und der Zustand $|a\rangle$ die Energie $+\frac{\omega_0}{2}$ besitzt. Der Hamiltonoperator, der das Zwei-Niveau-System beschreibt, lautet

$$\hat{\mathcal{H}}_{\text{Zwei-Niveaus}} = \hbar\frac{\omega_0}{2}\Big(|a\rangle\langle a| - |g\rangle\langle g|\Big) = \hbar\frac{\omega_0}{2}\hat{\sigma}_z \tag{2.7}$$

mit $\hat{\sigma}_z$ der Pauli-z-Matrix.

Die Energieeigenzustände eines Ions setzen sich unter diesen Voraussetzungen aus den Eigenzuständen des harmonischen Oszillators und des Zwei-Niveau-Systems zusammen:

$$|\Psi\rangle = |g,a\rangle\,|n\rangle \tag{2.8}$$

Man spricht dabei auch von n Phononen im Bewegungszustand.

Nun soll die Wechselwirkung mit Laserlicht hinzugefügt werden. Das Licht wird als ein klassisches elektrisches Feld behandelt, welches sich in z-Richtung ausbreitet:

$$\vec{E}(z,t) = \vec{E}_0\left(\mathrm{e}^{\mathrm{i}(kz-\omega_L t)} + \mathrm{e}^{-\mathrm{i}(kz-\omega_L t)}\right) \tag{2.9}$$

In unserem Experiment können drei verschiedene Typen von Laserwechselwirkungen eingesetzt werden:

- **Dipolwechselwirkung:**
  Das elektrische Feld wechselwirkt mit dem Dipolmoment des Ions. Der Hamiltonoperator lautet
  $$\hat{\mathcal{H}}_{\text{Dipol}} = e\vec{x} \cdot \vec{E}(z,t) \tag{2.10}$$
  mit der Elektronenladung e und der Position des Valenzelektrons $\vec{x}$.

- **Quadrupolwechselwirkung:**
  Ist die Dipolwechselwirkung verboten, so wird die nächstschwächere Wechselwirkung über das Quadrupolmoment vermittelt. Der Hamiltonoperator lautet
  $$\hat{\mathcal{H}}_{\text{Quadrupol}} = \sum_{i,j} eq_{ij} \frac{\partial}{\partial x_i} E_j \qquad \text{mit} \qquad q_{ij} = \frac{1}{2}\left(x_i x_j - \frac{1}{3}\delta_{ij}\vec{x}^2\right) \tag{2.11}$$

- **Stimulierte Ramanübergänge:**
  Stimulierte Ramanübergänge sind Zwei-Photon-Übergänge, bei denen zwei Grundzustandsniveaus $|1\rangle$ und $|2\rangle$ mithilfe von zwei Laserstrahlen gekoppelt werden, deren Frequenzunterschied genau der Übergangsfrequenz der Niveaus entspricht. Jeder der beiden Strahlen ist für sich nah an der Resonanz eines Dipolübergangs zu einem dritten Niveau $|3\rangle$ mit einer Verstimmung Δ, die groß genug ist, damit das dritte Niveau kaum noch bevölkert wird (Abbildung 2.3).

  Ramanübergänge können effektiv als Übergänge zwischen den Niveaus $|1\rangle$ und $|2\rangle$ aufgefasst werden, wobei sich die effektive Frequenz und der effektive Wellenvektor aus den Differenzen der einzelnen Strahlen ergeben [WMI+98]:
  $$\begin{aligned} \omega_{\text{eff}} &= \omega_2 - \omega_1 \\ \vec{k}_{\text{eff}} &= \vec{k}_2 - \vec{k}_1 \end{aligned} \tag{2.12}$$

Definiert man sich für alle drei Übergangstypen eine effektive Rabifrequenz Ω, eine effektive Frequenz ω_L und eine effektive Wellenzahl k, so lassen sie sich alle mit dem selben Formalismus beschreiben [LBMW03]. Der Wechselwirkungsterm des Hamiltonoperators lautet dann

$$\hat{\mathcal{H}}_{\text{WW}} = \frac{\hbar}{2}\Omega\left(\hat{\sigma}_+ + \hat{\sigma}_-\right)\left(\mathrm{e}^{\mathrm{i}(kz-\omega_L t)} + \mathrm{e}^{-\mathrm{i}(kz-\omega_L t)}\right) \tag{2.13}$$

mit den Pauli-Auf- bzw. Absteigeoperatoren des Zwei-Niveau-Systems $\hat{\sigma}_+$ und $\hat{\sigma}_-$. Der Hamiltonoperator des Gesamtsystems ergibt sich dann zu

$$\hat{\mathcal{H}} = \hat{\mathcal{H}}_{\text{Bewegung}} + \hat{\mathcal{H}}_{\text{Zwei-Niveaus}} + \hat{\mathcal{H}}_{\text{WW}} \tag{2.14}$$

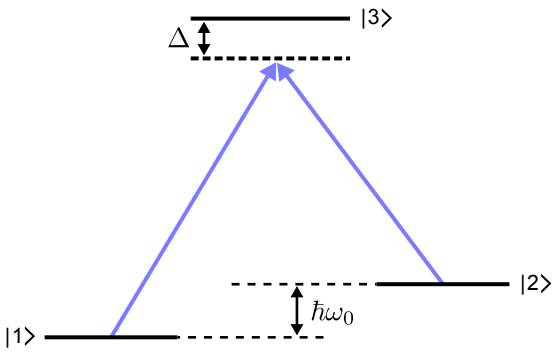


Abbildung 2.3.: Prinzip des stimulierten Raman-Übergangs

Es ist nun nützlich, den Hamiltonoperator in das Interaktionsbild zu transformieren. Dann lässt sich die Drehwellennäherung anwenden, es werden also Terme, die mit der Frequenz $\omega_L + \omega_0$ oszillieren, vernachlässigt. Man erhält

$$\hat{\mathcal{H}}_{\text{int}} = \frac{\hbar}{2}\Omega\left(\hat{\sigma}_+ \mathrm{e}^{\mathrm{i}\eta\left(\hat{a}^\dagger \mathrm{e}^{\mathrm{i}\omega_z t} + \hat{a}\mathrm{e}^{-\mathrm{i}\omega_z t}\right)}\mathrm{e}^{-\mathrm{i}\delta t} + \hat{\sigma}_- \mathrm{e}^{-\mathrm{i}\eta\left(\hat{a}^\dagger \mathrm{e}^{\mathrm{i}\omega_z t} + \hat{a}\mathrm{e}^{-\mathrm{i}\omega_z t}\right)}\mathrm{e}^{\mathrm{i}\delta t}\right) \tag{2.15}$$

mit der Verstimmung $\delta = \omega_L - \omega_0$ und dem *Lamb-Dicke-Parameter*

$$\eta = k\sqrt{\frac{\hbar}{2m\omega_z}} \tag{2.16}$$

Der Lamb-Dicke-Parameter beschreibt das Verhältnis der Ausdehnung des Wellenpakets des Ions in Bezug auf die Wellenlänge des einfallenden Laserlichts. Der Winkel des Lasers in Bezug auf die Fallenachse spielt dabei eine entscheidende Rolle, denn mit der Wellenzahl k ist hier nur der Anteil des Wellenvektors entlang der Fallenachse z gemeint.

Im Falle von $\eta \ll 1$ und für niedrige Phononenzahlen im Bewegungszustand lässt sich die Exponentialfunktion mit den ersten Termen einer Taylorentwicklung nähern und man erhält

$$\mathcal{H}_{\text{int}} \approx \frac{\hbar}{2}\Omega\left[\sigma_+ \mathrm{e}^{-\mathrm{i}\delta t}\left(1 + \mathrm{i}\eta\left(a^\dagger \mathrm{e}^{\mathrm{i}\omega_z t} + a\mathrm{e}^{-\mathrm{i}\omega_z t}\right)\right) + \sigma_- \mathrm{e}^{\mathrm{i}\delta t}\left(1 - \mathrm{i}\eta\left(a^\dagger \mathrm{e}^{\mathrm{i}\omega_z t} + a\mathrm{e}^{-\mathrm{i}\omega_z t}\right)\right)\right] \tag{2.17}$$

Man spricht dann vom *Lamb-Dicke-Regime*. Für bestimmte Wahlen von δ werden bestimmte Terme im Hamiltonoperator dominant, relevant sind hier drei verschiedene Fälle:

- **Der Trägerübergang** ($\delta = 0$):
  Beim Trägerübergang wird der Term $(\hat{\sigma}_+ + \hat{\sigma}_-)$ dominant. Es können dann lediglich Ab- bzw. Abregungen des Zwei-Niveau-Systems stattfinden, während der Bewegungszustand nicht beeinflusst wird.

- **Rotes Seitenband** ($\delta = -\omega_z$):
  Hier wird der Term $(\hat{a}\hat{\sigma}_+ - \hat{a}^\dagger\hat{\sigma}_-)$ dominant. Es können dann Prozesse stattfinden, bei denen das Zwei-Niveau-System bei gleichzeitiger Anregung der Bewegung angeregt wird und umgekehrt.
- **Blaues Seitenband** ($\delta = +\omega_z$):
  In diesem Fall wird der Term $(\hat{a}^\dagger\hat{\sigma}_+ - \hat{a}\hat{\sigma}_-)$ dominant. Es finden dann Prozesse statt, bei denen das Zwei-Niveau-System und der Bewegungszustand gleichzeitig an- bzw. abgeregt werden.

Die verschiedenen Übergänge sind in Abbildung 2.4 veranschaulicht. Außerhalb der Näherung des Lamb-Dicke-Regimes existieren weitere Terme, die bei Verstimmungen $\delta = n\omega_z$ dominant werden, mit n einer beliebigen ganzen Zahl [LBMW03]. Bei solchen Übergängen kann der Bewegungszustand bei Ab- bzw. Anregung des Zwei-Niveau-Systems um mehrere Phononen auf einmal geändert werden. Man spricht dann vom n-ten roten bzw. blauen Seitenband.

Bei den verschiedenen Seitenbändern ergeben sich unterschiedliche effektive Rabifrequenzen, die außerdem noch von der Phononenzahl des Zustandes abhängen. Die effektive Rabifrequenz, mit der ein Zustand mit n Phononen in einen Zustand mit $n+m$ Phononen überführt wird, lautet

$$\Omega_{n,n+m} = M_{n,n+m}\Omega \tag{2.18}$$

mit den Übergangsmatrixelementen [CG69]

$$M_{n,n+m} = \mathrm{e}^{-\eta^2/2}\,(\mathrm{i}\eta)^{|m|}\,\mathcal{L}_n^{|m|}(\eta^2)\left(\frac{n!}{(n+m)!}\right)^{\mathrm{sign}(m)/2} \tag{2.19}$$

Abbildung 2.5 zeigt berechnete Matrixelemente für den Trägerübergang und für die ersten und zweiten Seitenbänder bei einem Lamb-Dicke-Parameter von $\eta = 0{,}23$. Das entspricht dem Wert des Lamb-Dicke-Parameters in unserem Experiment. Deutliche Unterschiede in den Matrixelementen der roten und blauen Seitenbänder gibt es nur für sehr kleine Phononenzahlen.

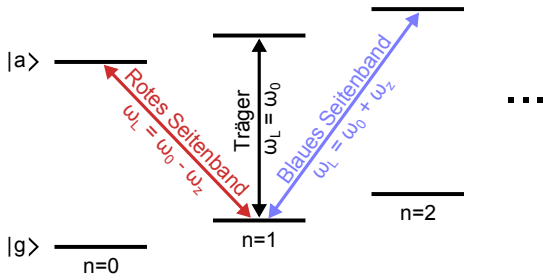


Abbildung 2.4.: Die verschiedenen Übergänge im Lamb-Dicke-Regime

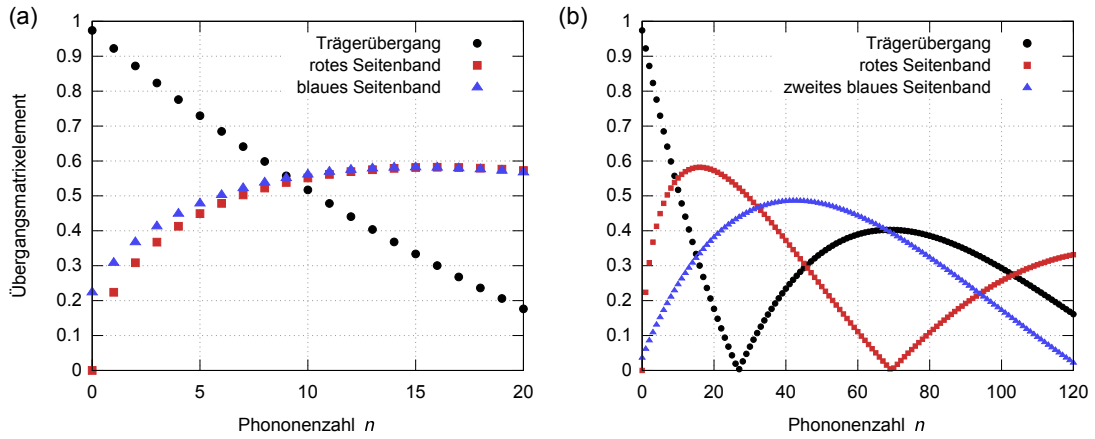


Abbildung 2.5.: Matrixelemente in Abhängigkeit der Phononenzahl bei $\eta = 0{,}23$ (a) für Träger und beide ersten Seitenbänder und (b) für Träger, erstes rotes und zweites blaues Seitenband

Solange die Laserverstimmung δ in der Nähe eines Seitenbandes liegt, führt das System für jede besetzte Phononenzahl Rabioszillationen aus [Pos10]. Für die Population im angeregten Zustand $|a\rangle$ gilt

$$p_a(t) = \sum_{n=0}^{\infty} p_n \sin^2\left(\frac{\Omega_{n,n+m}}{2}t\right) \tag{2.20}$$

mit p_n der Phononen-Verteilungsfunktion im Ausgangszustand. Befindet sich das System in einer Überlagerung von Bewegungszuständen, ob kohärent oder inkohärent, muss also über alle Phononenzahlen summiert werden.

Diese Beziehung kann verwendet werden, um den Bewegungszustand des Ions zu messen: Da die Matrixelemente $\Omega_{n,n+m}$ bekannt sind, kann durch einen Fit der Rabi-Oszillationen die Phononen-Verteilungsfunktion p_n rekonstruiert werden. Da die Zuordnung von Matrixelement zur Phononenzahl nicht eindeutig ist, müssen zur eindeutigen Bestimmung von p_n in der Regel mehrere Seitenbänder gemessen werden. Für mittlere Phononenzahlen $\bar{n} < 5$ ist es sinnvoll, Rabioszillationen des Trägers und beider ersten Seitenbänder aufzunehmen. Für größere Phononenzahlen ist es hingegen sinnvoll, die Rabioszillationen des Trägers, eines ersten Seitenbandes und eines zweiten Seitenbandes zu messen.

2.3. Kalzium-Ionen als Quantenbits

Zur Implementierung eines Quantenbits, bzw. eines Zwei-Niveau-Systems im Sinne des letzten Kapitels, werden in unserem Experiment $^{40}Ca^{+}$-Ionen verwendet. Kalzium-

Ionen besitzen einerseits durch heute verfügbare Laserwellenlängen erreichbare Übergänge, die als langlebiges Zwei-Niveau-System geeignet sind, andererseits aber auch weitere Übergänge, die zur Kontrolle und Detektion der Ionen verwendet werden können. Abbildung 2.6 zeigt die in diesem Experiment relevanten Energieniveaus und Übergänge.

Das Quantenbit wird aus den beiden Spin-Zuständen des $4S_{1/2}$-Niveaus gebildet. Man spricht hierbei vom *Spin-Qubit*. Beide Niveaus sind aufgrund eines äußeren Magnetfeldes um etwa 18 MHz aufgespalten. Sie haben eine praktisch unendliche Lebensdauer, sodass die Kohärenzzeit nicht vom spontanen Zerfall eines Niveaus limitiert wird. Da die Energiedifferenz der Niveaus allerdings ausschließlich von einem Magnetfeld erzeugt wird, dominieren stattdessen Dephasierungsprozesse aufgrund von Magnetfeldfluktuationen. Kohärente Übergänge zwischen den beiden Zuständen im Sinne des letzten Kapitels werden durch stimulierte Ramanübergänge erreicht, wobei der $4P_{1/2}$-Zustand als Zwischenstufe verwendet wird. Da beide Strahlen vom selben Laser erzeugt werden, und bei Ramanübergängen lediglich die Frequenzdifferenz beider Strahlen relevant ist, besitzt der Übergang eine hohe Phasenstabilität, auch wenn der Laser selbst diese nicht besitzt.

Der Übergang $4S_{1/2} \rightarrow 4P_{1/2}$ bei 397 nm ist mit einer Lebensdauer von 7,1 ns sehr kurzlebig, somit können an ihm in kurzer Zeit eine große Menge Photonen gestreut werden. Er eignet sich damit gut zur Doppler-Kühlung, zur Detektion und zur Qubit-Initialisation durch optisches Pumpen. Die während der Doppler-Kühlung vom Ion gestreuten Photonen können zur Ionendetektion von einer Kamera nachgewiesen werden. Beim optischen Pumpen wird ein σ^+ polarisierter Laserstrahl verwendet, sodass in kurzer Zeit viele Übergänge nach Abbildung 2.7a stattfinden. Durch die Polarisation wird der Spin-Up-Zustand des Grundzustandes nicht angeregt.

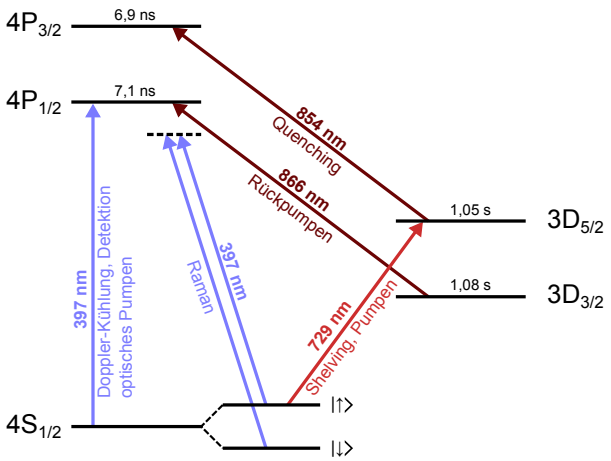


Abbildung 2.6.: Termschema von $^{40}Ca^+$ mit allen relevanten Übergängen. Die Zeeman-Unterzustände sind zur besseren Übersicht nur für den Grundzustand eingezeichnet.

Der $4P_{1/2}$-Zustand zerfällt jedoch mit einer Wahrscheinlichkeit von ca. 6,7% in den metastabilen $3D_{3/2}$-Zustand. In diesem Zustand kann das Ion nicht mehr vom Laser bei 397 nm angeregt werden. Deshalb nutzt man einen Rückpumplaser bei 866 nm, um das Ion wieder in den $4P_{1/2}$-Zustand zu bringen.

Ein Laser bei 729 nm kann das Ion vom $4S_{1/2}$-Zustand in den metastabilen $3D_{5/2}$-Zustand befördern. Im Gegensatz zum Laser bei 397 nm ist die Linienbreite des hierfür verwendeten Lasers schmal genug, um zwischen Zeeman-Unterzuständen zu unterscheiden. Somit kann mit diesem Laser beispielsweise nur der Spin-Up-Zustand $|\uparrow\rangle$ in den $3D_{5/2}$-Zustand gebracht werden. Das wird zur zustandsselektiven Detektion benutzt: Wird Laserlicht bei 397 nm eingestrahlt, so emittiert ein Ion im $3D_{5/2}$-Zustand kein Fluoreszenzlicht. Den Vorgang, einen Spinzustand vor einer Detektion zu leeren, bezeichnet man als *Shelving*.

Damit nach dem Shelving nicht auf den Zerfall des metastabilen Zustandes gewartet werden muss, wird das Ion mit einem Laser bei 854 nm in den kurzlebigen $4P_{3/2}$-Zustand gebracht, von wo aus es aufgrund der geringen Lebensdauer wieder in den Grundzustand zerfällt. Diese Prozedur wird *Quenching* genannt.

Der Übergang bei 729 nm kann ebenfalls als Qubit-Übergang genutzt werden, man spricht dann vom *optischen Qubit*. Damit auf diesem Übergang hohe Kohärenzzeiten erzielt werden können, ist im Gegensatz zum Spin-Qubit aber ein Laser mit einer hohen Kohärenzzeit notwendig.

Die Laser bei 729 nm und 854 nm können nach Abbildung 2.7b auch zur Qubit-Initialisation genutzt werden. Hierbei wird ausgenutzt, dass die Linienbreite des 729 nm-Lasers klein genug ist, um nur den Spin-Down-Zustand des Grundzustandes anzuregen. Im Gegensatz zu den Dipolübergängen ist der 729 nm-Übergang kohärent, sodass hier π-Pulse mit Pulsdauern in der Größenordnung von Mikrosekunden verwendet werden

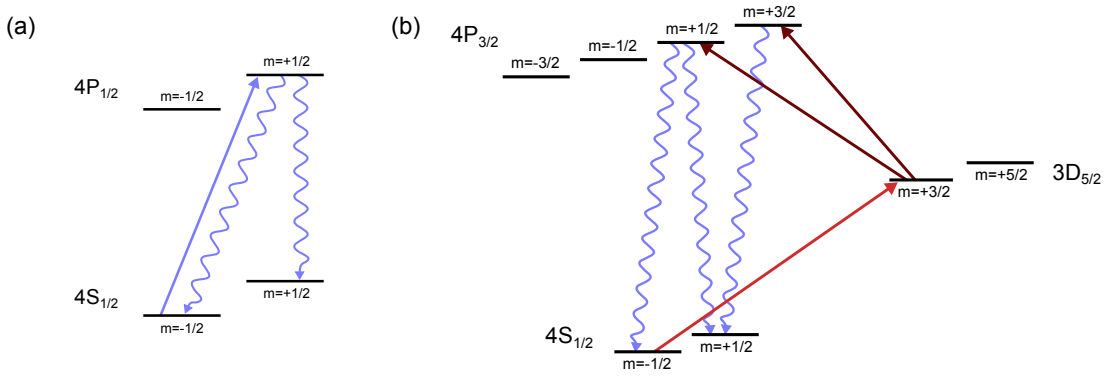


Abbildung 2.7.: Zwei Möglichkeiten zur Qubit-Initialisation: (a) mithilfe des 397 nm-Lasers (b) mithilfe des 729 nm-Lasers

müssen. Das verlängert die Dauer der Qubit-Initialisation, im Gegensatz zum optischen Pumpen mit dem 397 nm-Laser entstehen hier aber keine Fehler aufgrund einer möglicherweise nicht perfekten Polarisation des Laserstrahls. Daher wird diese Methode meistens verwendet, um nach dem optischen Pumpen die Genauigkeit der Qubit-Initialisation zu erhöhen.

3

Experimenteller Aufbau

In diesem Kapitel wird ein kurzer Überblick über den Aufbau des Experiments gegeben. Dabei liegt der Schwerpunkt vor allem auf die für den Ionentransport nötige Spannungsversorgung der einzelnen Segmente der Mikrofalle und auf den Komponenten des experimentellen Kontrollsystems, welche entscheidend für die Durchführung von experimentellen Sequenzen sind. Auf tiefere Details bezüglich der Vakuumapparatur und der Lasersysteme wird in dieser Arbeit nicht eingegangen, dafür wird hier lediglich auf entsprechende Quellen verwiesen.

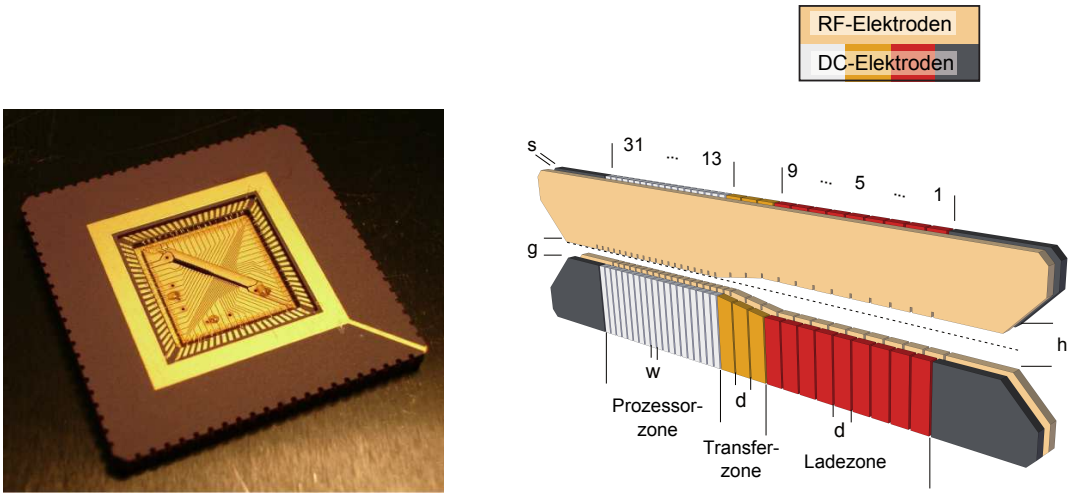
3.1. Die segmentierte Mikrofalle

Die segmentierte Mikrofalle (Abbildung 3.1) ist das Herzstück unseres Experiments. Sie besteht aus zwei RF-Elektroden und zwei DC-Elektroden, wobei letztere jeweils in 31 Segmente aufgeteilt sind, welche unabhängig voneinander mit Spannung versorgt werden können. Jedes Segment bildet für sich eine lineare Paulfalle, wie sie in Kapitel 2 beschreiben ist. Die Falle ist in drei Regionen unterteilt: Eine Ladezone, eine Prozessorzone und eine Transferzone. Die Segmente der Prozessorzone sind kleiner und näher beieinander, um höhere Fallenfrequenzen zu ermöglichen. Details zur Herstellung finden sich in [Sch09].

Alle im Rahmen dieser Arbeit durchgeführten Experimente fanden mit einem gefangenen Ion in der Ladezone statt. Bei einer üblicherweise angelegten Gleichspannung von $-7\,\mathrm{V}$ beträgt die axiale Fallenfrequenz etwa 1,4 MHz. Ionen wurden in Segment 5 gefangen, wobei sie bei Transportmessungen in Segment 4 und zurück transportiert wurden.

Die Mikrofalle befindet sich in einer Ultrahochvakuumkammer, um Stöße der gefangenen Ionen mit Hintergrundgas zu minimieren. Eine Titan-Sublimationspumpe sorgt für einen Druck zwischen 10^{-10} und 10^{-9} mbar. In [Mai06] ist die Optik beschrieben, mit der Bilder von gefangenen Ionen mithilfe einer EMCCD-Kamera aufgenommen werden.

Zwei an der Vakuumkammer angebrachte Spulen sorgen für ein Magnetfeld, das die beiden Grundzustandsniveaus der Kalzium-Ionen um etwa 18 MHz aufspaltet. In der



(a) (b)

Abbildung 3.1.: (a) Foto und (b) Skizze der segmentierten Falle [Sch09]. Die Maße betragen $s = 125\mu$m, $g = 250\mu$m, $w = 125\mu$m und $d = 250\mu$m. Die Länge der Falle beträgt insgesamt 5,8 mm.

Nähe der Falle befinden sich zwei Kalzium-Öfen, die Kalzium-Atome in Fallenrichtung emittieren. Mehrere Sichtfenster in der Vakuumkammer erlauben es, Ionen von verschiedenen Winkeln mit Laserstrahlen zu erreichen.

3.2. Konfiguration der Laserstrahlen

Um festzulegen, welche Übergänge von Lasern getrieben werden können, ist nicht nur die Polarisation der einzelnen Laserstrahlen entscheidend, sondern auch der Winkel der Laserstrahlen relativ zum von außen angelegten Magnetfeld. Des weiteren ist für den Lamb-Dicke-Parameter eines Übergangs der Winkel der Laserstrahlen zur Fallenachse relevant. Die Konfiguration aller verwendeten Laserstrahlen ist in Abbildung 3.2 skizziert.

Zwei Diodenlaser bei 374 nm und 423 nm werden zur Photoionisation von Kalziumatomen eingesetzt. Zum Photoionisationsprozess können nähere Details in [GRB+01] nachgelesen werden.

Die beiden Strahlen „Doppler“ und „Sigma“ werden vom selben 397 nm-Diodenlaser erzeugt. Er wird, wie der Diodenlaser bei 866 nm auch, nach dem Pound-Drever-Hall-Verfahren auf einen externen Resonator stabilisiert [Bla01]. An den Resonatoren angebrachte Piezo-Elemente erlauben es, die Wellenlänge der beiden Laser einzustellen.

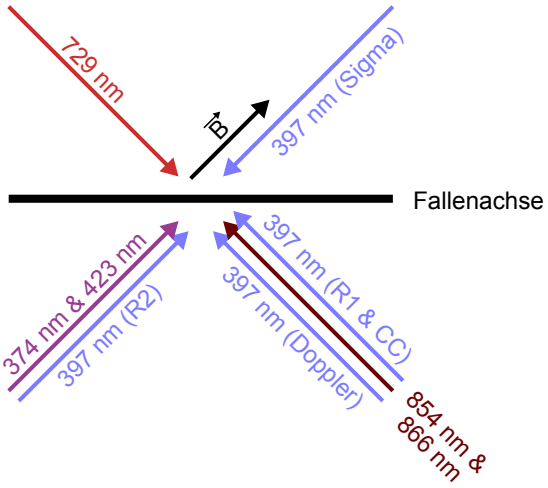


Abbildung 3.2.: Konfiguration der Laserstrahlen relativ zur Fallenachse

Der Strahl „Doppler“ wird zum Dopplerkühlen eingesetzt. Er trifft gefangene Ionen so, dass sowohl π-Übergänge als auch $\sigma^{\pm}$-Übergänge getrieben werden können. Außerdem ist er im 45°-Winkel zur Falle ausgerichtet, sodass er sowohl axiale als auch radiale Schwingungsmoden kühlen kann. Der Strahl „Sigma“ wird zum optischen Pumpen eingesetzt. Er ist entlang der Magnetfeldachse ausgerichtet und zirkular polarisiert, sodass er lediglich σ^{+}-Übergänge treiben kann.

Ein weiterer Diodenlaser bei 397 nm erzeugt die Strahlen „R1“, „R2“ und „CC“. Sie dienen zur Realisierung von stimulierten Raman-Übergängen und können untereinander kombiniert werden, um verschiedene effektive Lamb-Dicke-Parameter zu erhalten. Die Kombination der Strahlen „R1“ und „CC“ wird auch „Cocarrier“-Übergang genannt. Der effektive Wellenvektor, wenn diese beiden Strahlen kombiniert werden, verschwindet, da beide Strahlen entlang der selben Richtung propagieren. Daher koppelt dieser Übergang weder an radiale noch an axiale Schwingungsmoden. Er ist damit komplett unabhängig vom Bewegungszustand gefangener Ionen. Kombiniert man die Strahlen „R1“ und „R2“, so zeigt der effektive Wellenvektor in Richtung der Fallenachse. Übergänge, die mit diesen beiden Strahlen getrieben werden, koppeln mit einem Lamb-Dicke-Faktor von etwa $\eta \approx 0{,}23$ an die axialen Schwingungsmoden der Ionen. Für die radialen Schwingungsmoden ist der Lamb-Dicke-Faktor hier ebenfalls Null.

Für den Übergang bei 729 nm wurde in diesem Jahr ein Titan-Saphir-Laser aufgebaut [Mac12]. Dieser wird nach dem Pound-Drever-Hall-Verfahren auf einen Resonator hoher Güte stabilisiert. Die Linienbreite und die Frequenzdrifts dieses Lasers sind gering genug, dass der Übergang wie bisher nicht nur für das Electron Shelving, sondern ebenfalls als Qubit-Übergang verwendet werden kann, was neue experimentelle Möglichkeiten eröffnet. Der Laserstrahl ist momentan so polarisiert, dass er Übergänge mit $\Delta m = 1$ und $\Delta m = 2$ gleichermaßen treiben kann. Aufgrund des Winkels zum Magnetfeld ist der

Übergang mit $\Delta m = 0$ verboten. Mehr Details zu Quadrupolübergängen finden sich z.B. in [Roo00]. Dieser Laser koppelt sowohl an Radial- als auch an Axialmoden der Bewegung.

Der Laser bei 854 nm ist ebenfalls ein Diodenlaser. Da er lediglich zum Leeren des $D_{5/2}$-Zustandes eingesetzt wird, genügt es, seine Wellenlänge mithilfe des im Labor verwendeten Wavemeters konstant zu halten.

3.3. Fallen-Spannungsversorgung

Die Spannungsversorgung für die verschiedenen Gleichspannunssegmente der Mikrofalle ist eine vollständige Eigenentwicklung unserer Arbeitsgruppe. An die Spannungsversorgung werden hohe Ansprüche gestellt, die von bisherigen kommerziellen Lösungen nicht erfüllt werden konnten, nämlich:

- Zur Realisierung von Transportexperimenten muss die Spannung mindestens einmal pro Mikrosekunde geändert werden können.
- Alle Fallensegmente müssen mit unterschiedlichen Spannungen versorgt werden können.
- Die Spannung sollte möglichst rauscharm sein, um die Kohärenz gefangener Ionen zu bewahren, und um ein Aufheizen des Bewegungszustandes zu vermeiden.
- Die Spannung sollte eine möglichst geringe Driftrate aufweisen.

Bei der Spannungsversorgung handelt es sich um ein modulares System, welches aus potentiell mehreren Platinen mit Digital-Analog-Wandlern (DAC), einer weiteren Platine mit einem *Field Programmable Gate Array* (FPGA), und einer batteriebetriebenen Spannungsversorgung besteht.

Das System wird aus Batterien versorgt, um es von Störungen des Stromnetzes zu entkoppeln. Die Digital-Analog-Wandler haben eine 16 Bit-Auflösung, was bei einer Referenzspannung von +10V kleinstmögliche Spannungsschritte von etwa 0,3 mV zulässt. Der maximale Takt, mit dem die Digital-Analog-Wandler betrieben werden können, beträgt 50 MHz. Das entspricht einem Taktzyklus alle 20 ns. Es werden 20 Taktzyklen zur Ändern der Spannung benötigt, was einer Gesamtdauer von 400 ns entspricht.

Zur Steuerung der Digital-Analog-Wandler wird ein FPGA verwendet. Das ist ein Integrierter Schaltkreis, dessen interne Struktur programmiert werden kann. Er sendet im 100 MHz-Takt, d.h. mit bestenfalls 10 ns Zeitschritten, digitale Signale an die Digital-Analog-Wandler. Außerdem ist ein PowerPC-Prozessor integriert, der eine Gigabit-Netzwerkschnittschnelle zum Kontrollrechner des Experiments verwaltet. Vom Kontrollrechner aus können somit Spannungsdaten an den FPGA gesendet werden.

Der FPGA ist ein mächtiges Werkzeug, welches die in dieser Arbeit durchgeführten Transport-Experimente erst möglich gemacht hat. Um den Bewegungszustand eines Ions zu steuern, müssen die Spannungen schnell gegenüber der Schwingungsdauer des Ions gesteuert werden können, welche etwa 700 ns beträgt. Zwar können die Digital-Analog-Wandler ihre Ausgangsspannung nur nach einer 400 ns-Zeitverzögerung ändern, diese jedoch lässt sich aufgrund der Steuerung durch den FPGA in 10 ns-Schritten erhöhen.

Abbildung 3.3 zeigt ein Spektrum des Rauschens, das von der Spannungsversorgung verursacht wird, während die Ausgangsspannungen aktualisiert werden. Somit beschreibt es das typische Rauschverhalten während eines Transportexperiments. Besonders kritisch ist Rauschen auf Frequenzen nahe der Fallenfrequenz von etwa 1,4 MHz, da dieses den Bewegungszustand von Ionen anregt. In diesem Bereich fällt das Rauschen am geringsten aus - somit ist die Spannungsversorgung sehr gut für unser Experiment geeignet. Um diese und auch höhere Frequenzen weiter zu unterdrücken, wird ein Π-Tiefpass-Filter mit einer Grenzfrequenz von etwa 300 kHz verwendet.

3.4. Experimentelles Kontrollsystem

Um komplexe Experimente durchführen zu können, müssen zahlreiche Hardwarekomponenten synchronisiert werden. Es müssen Laser an- und ausgeschaltet, Fallenspannungen für Transportexperimente kontrolliert sowie Kamerabilder aufgenommen werden. Zur kohärenten Manipulation ist außerdem die genaue Kontrolle der Leistung, Frequenz und Phase der 729 nm- bzw. Raman-Laserstrahlen notwendig. Dies wird im Experiment

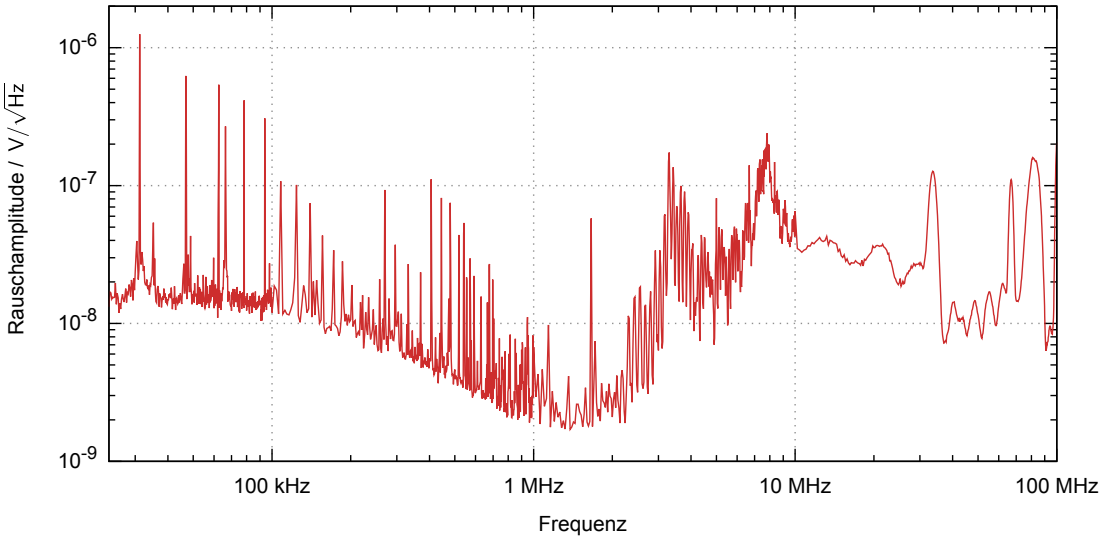


Abbildung 3.3.: Rausch-Spektrum der Spannungsversorgung

durch akustooptische Modulatoren verwirklicht. Diese sind in der Lage, die Eigenschaften von eintreffenden Radiofrequenz-Signalen auf das Laserlicht zu übertragen. Die dafür notwendigen Radiofrequenz-Signale werden von einem *Versatile Function Generator* (VFG) erzeugt, welcher die notwendigen Daten über eine USB-Schnittstelle vom Kontrollrechner erhält.

Die wichtigsten Teile des Kontrollsystems sind in Abbildung 3.4 skizziert. Um eine korrekte und genaue zeitliche Abfolge von Ereignissen in einem Experiment zu ermöglichen, sind im Kontrollrechner mehrere Erweiterungskarten von National Instruments eingebaut. Diese sind in der Lage, synchronisiert digitale TTL-Signale im Mikrosekunden-Takt an andere Hardware-Komponenten zu senden. So können dann mithilfe elektrooptischer oder akustooptischer Modulatoren Laser an- bzw. abgeschaltet werden - darüber hinaus werden Triggersignale an die EMCCD-Kamera, die Fallen-Spannungsversorgung und an den VFG geschickt, um diesen zu signalisieren, dass sie ihre zuvor vom Kontrollrechner festgelegten Aktivitäten abarbeiten sollen.

Die beiden Photoionisationslaser können ebenfalls über ein TTL-Signal an- bzw. abgeschaltet werden. Die Strahlen können hier mechanisch blockiert werden, da die beiden

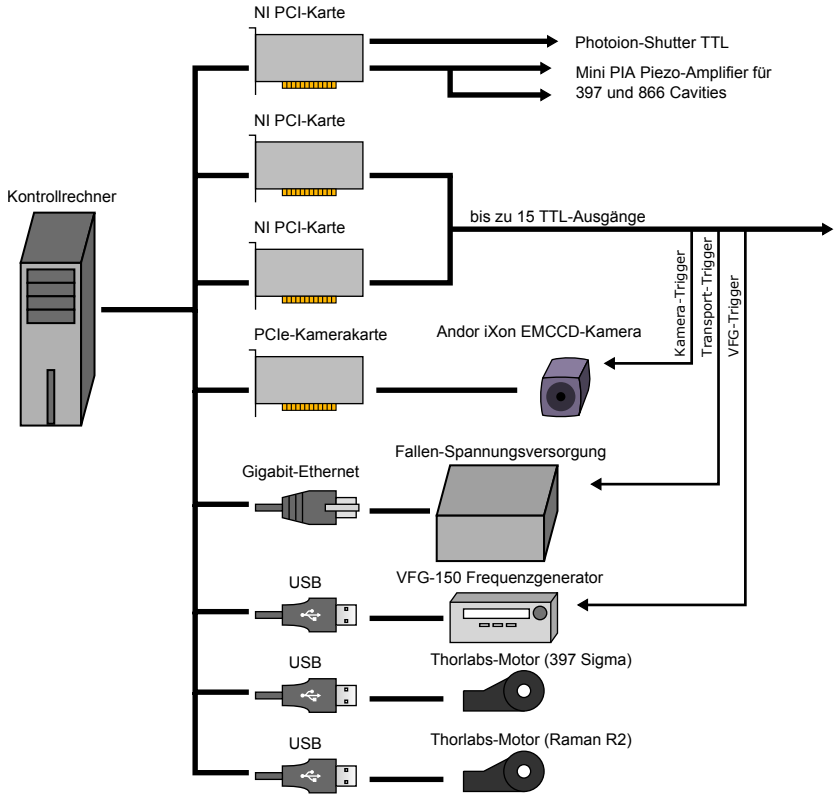


Abbildung 3.4.: Skizze des experimentellen Kontrollsystems

Laser in schnellen Sequenzen nicht benötigt werden. Darüber hinaus besitzen die National Instruments-Erweiterungskarten analoge Ausgänge, mit denen die Piezo-Elemente an den Resonatoren für die 397 nm- und 866 nm-Laser kontrolliert werden können. So kann die Wellenlänge dieser beiden Laser auf einer Millisekunden bis Sekundenzeitskala geändert werden.

Ebenfalls auf einer Sekundenzeitskala lassen sich die Polarisationen der 397 nm-Sigma- und Raman R2-Strahlen ändern. Dazu sind Lambda-Plättchen auf motorisierten Haltern der Firma Thorlabs montiert, die sich über eine USB-Verbindung ebenfalls vom Kontrollrechner steuern lassen. Mit der Polarisation des 397 -Sigma-Strahls lässt sich bestimmen, in welchen der beiden Spin-Zustände zu Beginn einer Sequenz gepumpt werden soll. Das ist z.B. nützlich zum Einstellen der Parameter des 729-Pumpens. Die Polarisation des Raman R2-Strahls muss sehr präzise eingestellt sein, um die durch diesen Strahl verursachte AC-Starkverschiebung zu minimieren [Pos10].

4

Experimentelle Sequenzen

Nachdem nun in den vorangegangenen Kapiteln die grundlegende Theorie und die wichtigsten Teile des experimentellen Aufbaus behandelt wurden, soll es in diesem Kapitel darum gehen, welche Schritte zur Durchführung eines konkreten Experiments notwendig sind. Fast alle Messungen laufen in der Regel nach einem bestimmten Schema ab, welches in Abbildung 4.1 skizziert ist. Gefangene Ionen werden zunächst dopplergekühlt, danach werden sie durch das *Seitenbandkühlen* in den Grundzustand der axialen Bewegung gebracht. Optisches Pumpen initialisiert den Spin der Ionen in den $m=+\frac{1}{2}$-Grundzustand. Falls eine besonders hohe Genauigkeit erforderlich ist, kann optional nach dem optischen Pumpen die Initialisierung durch Pumpen mit dem 729 nm-Laser weiter verbessert werden. Nach diesen Schritten sind die gefangenen Ionen im Zustand $|\uparrow\rangle\,|0\rangle$ präpariert.

In diesem Zustand lassen sich dann Quantengatter oder andere kohärente Operationen mithilfe stimulierter Raman-Übergänge durchführen, oder - besonders für diese Arbeit relevant - Transporte von kalten Ionen gefolgt von Ramanpulsen zur Bestimmung des Bewegungszustandes nach dem Transport.

Nachdem die gewünschten Operationen durchgeführt wurden, muss noch der Spin der Ionen nachgewiesen werden. Dazu wird beim Shelving mit dem 729 nm-Laser der Spin-Up-Zustand geleert, sodass nach dem Einschalten des 397 nm-Lasers beim anschließenden Detektionsfenster im Gegensatz zu Ionen im Spin-Down-Zustand kein Fluoreszenzlicht mehr emittiert wird. Anhand eines während des Detektionsfensters aufgenommenen Kamerabildes kann dann entschieden werden, ob sich Ionen im dunklen oder im hellen Zustand befanden. Beim abschließenden Quenching werden Ionen durch opti-

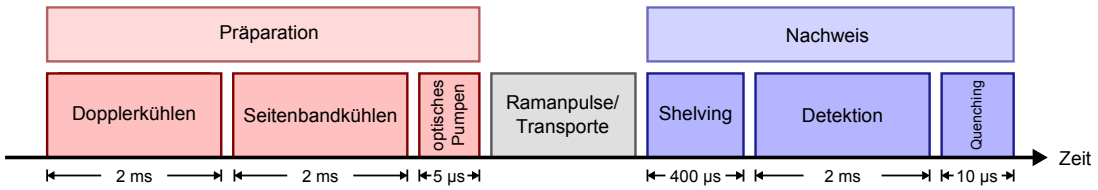


Abbildung 4.1.: Eine typische experimentelle Sequenz

sches Pumpen in das $P_{3/2}$-Niveau angeregt, von wo aus sie wieder in den Grundzustand zerfallen.

Da bei der Detektion der Quantenzustand eines Ions auf einen der beiden Qubit-Zustände projeziert wird, können Superpositionszustände nicht mit einer Einzelmessung charakterisiert werden. Stattdessen muss ein Experiment mehrmals wiederholt werden, um Wahrscheinlichkeiten für ein helles bzw. dunkles Ion zu erhalten. In der Regel werden solche Experimente je nach gewünschter statistischer Genauigkeit 100 oder 200 mal wiederholt.

Bevor allerdings aufwendige Experimente durchgeführt werden können, müssen einige Vorbereitungen getroffen werden. Es müssen zunächst einmal Ionen mithilfe der Photoionisationslaser bei 423 nm und 374 nm gefangen werden. Für jeden der oben aufgeführen Experimentschritte müssen verschiedene Parameter wie Laserfrequenzen, Pulszeiten oder auch Fallenspannungen korrekt gesetzt sein. Das soll in den folgenden Abschnitten genauer behandelt werden. Ein besonderes Augenmerk soll an dieser Stelle auch auf der automatischen Kalibration von Parametern liegen. Da alle Parameter aufgrund von Drifts in regelmäßigen Abständen korrigiert werden müssen, ist so eine enorme Zeitersparnis möglich.

4.1. Optimierung der Dopplerkühlung und Ionenfluoreszenz

Damit gefangene Ionen möglichst effizient dopplergekühlt werden können, muss die Frequenz des 397 nm-Lasers passend eingestellt werden. Da der Laser mithilfe des Pound-Drever-Hall-Verfahrens auf einen externen Resonator stabilisiert ist, lässt sich die Wellenlänge durch Anlegen einer Spannung auf ein Piezoelement, das direkt am Resonator angebracht ist, vom Kontrollrechner aus steuern. Mithilfe der Analog-Ausgänge der im Kontrollrechner eingebauten Ni-Karten wird eine Spannung erzeugt, die dann von einem Verstärker weiter verstärkt wird. Die Werte im folgendenen Abschnitt beziehen sich immer auf die Ausgangsspannung der Ni-Karte. Zur effizienten Dopplerkühlung sollte der Laser nicht genau auf der Resonanz des Übergangs liegen, sondern um eine halbe Linienbreite rotverstimmt sein. Von der Wellenlänge des 397 nm-Lasers hängt aber auch gleichzeitig die Intensität der Ionenfluoreszenz ab. Diese sollte zur schnellen Detektion in Sequenzen möglichst hoch sein, der 397 nm-Laser auf Resonanz wäre daher optimal. Als Kompromiss und guten Erfahrungswert wird deshalb die Wellenlänge so weit rotverstimmt, bis noch etwa 70% der maximalen Fluoreszenz vorhanden ist. Das kann in einer vollautomatischen Messung erfolgen.

Automatische Messung 1:
Bei der automatischen Messung wird zunächst das Fluoreszenzmaximum gesucht. Dazu wird die Frequenz des 397 *nm-Lasers schrittweise erhöht und für jeden Schritt die Ionenfluoreszenz f gemessen. Bei jedem Schritt werden dazu N Kamerabilder aufgenommen*

und der Mittelwert gebildet. Der Fehler der einzelnen Messpunkte ergibt sich dann zu $\Delta f = \frac{\sigma_f}{\sqrt{N}}$ *wobei mit* σ_f *die Standardabweichung gemeint ist.*

Sobald die eingestellte Frequenz die Resonanzfrequenz überschritten hat, verschwindet die Fluoreszenz vollständig, da bei blauverstimmtem Laser Ionen aufgeheizt anstatt gekühlt werden. Tritt dies ein, wird die Frequenz des Lasers wieder auf den Startwert zurückgesetzt und dann schrittweise so lange verringert, bis die Fluoreszenz unter 35% des maximalen Wertes gefallen ist. Diese Frequenz wird für die im nächsten Kapitel vorgestellte Sequenz zur Minimierung der Mikrobewegung zwischengespeichert. Dann wird die Laserfrequenz dauerhaft auf den Messpunkt eingestellt, bei dem die Ionenfluoreszenz gerade unter 70% gefallen ist. Abbildung 4.2a zeigt eine Beispielmessung.

Die Wellenlänge des 397 nm-Lasers driftet im Verlauf eines Tages erheblich (Abbildung 4.2b). Die Frequenzverschiebung durch das Piezo-Element beträgt etwa $48\,\frac{\mathrm{MHz}}{\mathrm{V}}$, was bedeutet, dass an jedem Tag ein Frequenzdrift von über 100 MHz kompensiert werden muss. Um eine konstante Ionenfluoreszenz zu gewährleisten, sollte die automatische Kalibrierung daher mindestens einmal pro halbe Stunde ausgeführt werden. Als Grund der starken Frequenzschwankungen werden thermische Drifts des Resonators, auf den der Laser stabilisiert wird, vermutet - der Resonator wird durch den Laserstrahl selbst aufgewärmt. Da der Laser nachts abgeschaltet ist, ist das Driftverhalten jeden Tag identisch.

Der Rückpump-Laser bei 866 nm hat ebenfalls Auswirkungen auf das Dopplerkühlen und die Ionenfluoreszenz. Da diese Auswirkungen aber klein sind, reicht eine grobe Stimmung dieses Lasers per Hand aus.

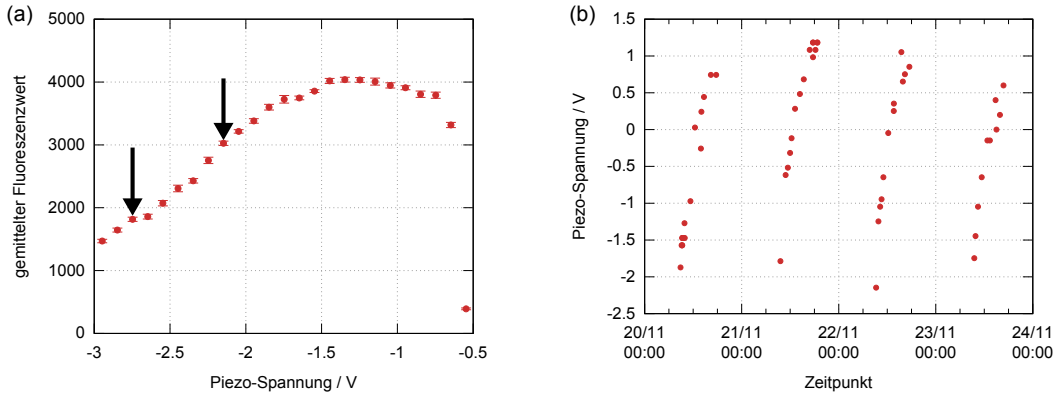


Abbildung 4.2.: (a) Ergebnisse einer Beispielmessung zur Bestimmung der optimalen Wellenlänge des 397 nm-Lasers. Es werden zwei Piezo-Spannungen bestimmt: Der linke Pfeil zeigt die Spannung, die zur Minimierung der Mikrobewegung verwendet wird. Die Spannung des rechten Pfeils wird für alle anderen Experimente verwendet. (b) zeitlicher Verlauf der Piezo-Spannung über mehrere Tage

4.2. Messung und Minimierung der Mikrobewegung

Mikrobewegung, verursacht durch die an der Falle angelegte Radiofrequenz, tritt immer dann auf, wenn ein Ion aus der Fallenmitte ausgelenkt ist. Durch Assymmetrien in der Fallengeometrie oder elektrische Ladungen in der Nähe des Ions verschiebt sich dessen mittlere Position, was zusätzliche Mikrobewegung verursacht. Laserlicht, das entlang einer Achse mit nichtverschwindender Mikrobewegung ausgerichtet ist, wird im Bezugssystem des Ions frequenzmoduliert, was zu zusätzlichen Seitenbändern im Spektrum führt und die Stärke von Trägerübergängen verringert. Bei Dipolübergängen mit hoher Linienbreite führt dies zu einer effektiven Verbreiterung der Linien, und kann z.B. die Effizienz des Dopplerkühlens beeinflussen [BMB+98].

Daher sollte die überschüssige Mikrobewegung so klein wie möglich gehalten werden. Das wird im Experiment durch Anlegen einer zusätzlichen Spannungsdifferenz an sich gegenüberliegenden Gleichspannungselektroden erreicht, mit dem Ziel, Ionen zur Fallenmitte zurückzuschieben. Damit die Kompensationsspannung korrekt gesetzt werden kann, ist ein Messverfahren nötig, um die Amplitude der Mikrobewegung genau zu messen. An unserem Experiment hat es sich bewährt, die Mikrobewegung mithilfe der Stärke der Ionenfluoreszenz zu untersuchen. Dazu wird der Laser bei 397 nm rotverschoben, sodass die gemessene Fluoreszenz empfindlich auf die Linienbreite des Übergangs reagiert. Je größer die Mikrobewegung, desto höher ist die Linienbreite und desto höher ist die Anregungswahrscheinlichkeit bei rotverschobenem Laserlicht. Es geht also darum, ein Minimum im Fluoreszenzsignal zu finden, während die Kompensationsspannung variiert wird. Mehr Details finden sich in [BMB+98] und [Pos10].

Automatische Messung 2:
Da sowohl die Wellenlänge des 397 nm-Lasers als auch die Kompensationsspannung vom Kontrollrechner gesteuert werden können, kann die Kompensation der Mikrobewegung vollautomatisch durchgeführt werden. Zunächst wird die Wellenlänge des 397 nm-Lasers auf den Wert verschoben, der zuvor in der 397 nm-Spektroskopiesequenz gespeichert wurde. Dann wird für verschiedene Kompensationsspannungen um den von der letzten Messung bestimmten Wert die Ionenfluoreszenz gemessen. Die Position des Minimums wird wie folgt bestimmt:

- *Suche ausgehend vom zuletzt bekannten Wert zwei Maxima, und zwar eines links und eines rechts vom Startwert.*
- *Betrachte nur Messpunkte, die sich unterhalb des niedrigeren der beiden Maxima befinden. Der Fluoreszenzwert dieses Maximums betrage m.*
- *Berechne einen gewichteten Mittelwert*

$$\overline{V} = \frac{\sum_{i=1}^{n} g_i^2 \cdot V_i}{\sum_{i=1}^{n} g_i^2} \tag{4.1}$$

mit n der Anzahl der relevanten Messpunkte, V_i der gesetzten Kompensationsspannung des i-ten Messpunktes und den Gewichtungsfaktoren $g_i = m - f_i$, wobei f_i der Fluoreszenzwert des i-ten Messpunktes ist.

- *Der statistische Fehler in den einzelnen Messpunkten wird analog zur Messung der Frequenz des 397 nm-Lasers abgeschätzt. Gaußsche Fehlerfortpflanzung liefert dann für den statistischen Fehler der Kompensationsspannung*

$$\Delta\overline{V} = \frac{2}{\sum_{i=1}^{n} g_i^2} \sqrt{\sum_{i=1}^{n} \left[g_i^2 \left(V_i - \overline{V} \right)^2 \left((\Delta m)^2 + (\Delta f_i)^2 \right) \right]} \tag{4.2}$$

Der gewichtete Mittelwert ist eine gute Schätzung für die optimale Kompensationsspannung, die resistent ist gegen kleine durch Rauschen bedingte Schwankungen der Messwerte.

Abbildung 4.3a zeigt Messwerte einer durchgeführten automatischen Messung. Zur Veranschaulichung der Ergebnisse wurde ein großer Bereich um die vorher bekannte Spannung aufgenommen, zur zuverlässigen automatischen Bestimmung reichen schon deutlich weniger Messwerte aus.

Unter normalen Bedingungen ändert sich die benötigte Kompensationsspannung über einen langen Zeitraum kaum (Abbilding 4.3b). Daher muss die Kalibration im normalen Betrieb nur wenige male am Tag durchgeführt werden. Eine plötzliche Änderung der nötigen Kompensationsspannung deutet normalerweise darauf hin, dass die Ladung der Batterien für die Spannungsversorgung zur Neige geht. Auch hat die Amplitude der an der Falle angelegten Wechselspannung einen Einfluss auf die benötigte Kompensationsspannung.

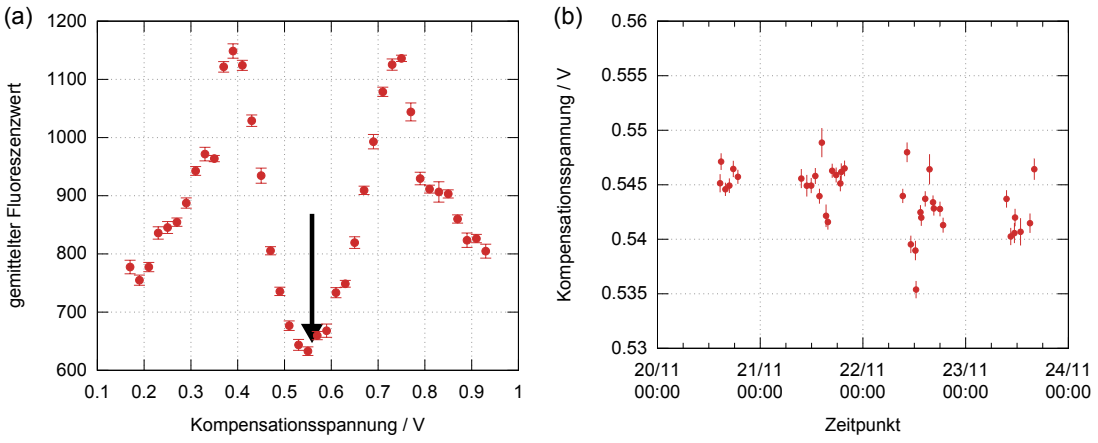


Abbildung 4.3.: (a) Ergebnisse einer Beispielmessung zur Bestimmung der Kompensationsspannung. Der schwarze Pfeil zeigt die neu bestimmte Spannung. (b) Zeitlicher Drift der optimalen Kompensationsspannung

4.3. Shelving

Im nächsten Schritt müssen Parameter für ein optimal funktionierendes Shelving bestimmt werden. Da der Übergang bei 729 nm ein Quadrupolübergang mit geringer Linienbreite ist, muss die korrekte Laserfrequenz zunächst mittels eines akustooptischen Modulators genau eingestellt werden. Dazu muss eine Spektroskopiesequenz ausgeführt werden, bei der zunächst gefangene Ionen durch optisches Pumpen in den $|\uparrow\rangle$-Zustand initialisiert werden, gefolgt von einem Laserpuls mit dem 729 nm-Laser und einer Detektion mit der EMCCD-Kamera. Diese Sequenz wird zunächst mehrmals für eine feste eingestellte Frequenz am akustooptischen Modulator zur Bestimmung einer Anregungswahrscheinlichkeit ausgeführt, was dann nochmals für verschiedene Frequenzen wiederholt wird um ein Spektrum wie in Abbildung 4.4 zu erhalten.

Die Gestalt des Spektrums hängt stark von der verwendeten Pulsdauer und Laserleistung ab. Für jeden Messpunkt werden Rabioszillationen mit anderer Verstimmung getrieben, die Anregungswahrscheinlichkeit p mit Verstimmung lautet [Fox11]

$$p = \frac{1}{2}\frac{\Omega_0^2}{\Omega^2}\left(1 - \cos(\Omega\tau)\right) \tag{4.3}$$

mit τ der Pulsdauer, Ω_0 der resonanten Rabifrequenz und $\Omega = \sqrt{\Omega_0^2 + \delta^2}$ der Rabifrequenz bei der Verstimmung δ. Nehme an, die Parameter seien so gewählt, dass die Anregungswahrscheinlichkeit bei Resonanz genau 1 ist. Die maximal mögliche Anregungswahrscheinlichkeit nimmt dann mit zunehmender Verstimmung ab, sodass ein

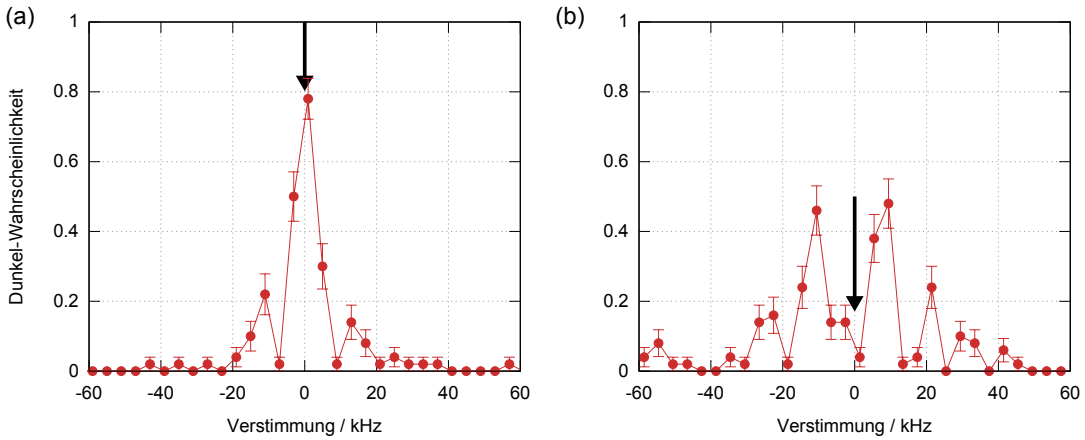


Abbildung 4.4.: Spektroskopieergebnisse um die Resonanzfrequenz eines Übergangs herum (a) bei optimal eingestellter Laserleistung und (b) bei zu hoch eingestellter Leistung. Der schwarze Pfeil zeigt das Ergebnis der automatischen Auswertung. Trotz der falsch eingestellten Leistung in (b) konnte die Resonanzfrequenz korrekt bestimmt werden.

Maximum um die Resonanz herum entsteht. Die Breite des Maximums ist bei langen Pulszeiten geringer, da dann die Verstimmung im Verhältnis einen höheren Beitrag zur effektiven Rabifrequenz liefert und so das Verhältnis $\frac{\Omega_0^2}{\Omega^2}$ schneller abnimmt. Da sich mit der Verstimmung aber auch die effektive Rabifrequenz ändert, können weitere Nebenmaxima auftreten. Bei ungünstig eingestellter Laserleistung kann es passieren, dass bei Resonanz ein Minimum in der Anregung entsteht.

Automatische Messung 3:
Die Spektroskopieergebnisse können durch Bildung eines gewichteten Mittelwertes effektiv automatisch ausgewertet werden. Die Anregungswahrscheinlichkeit übernimmt dabei die Rolle des Gewichtungsfaktors. Die mittlere Frequenz bestimmt sich zu

$$\overline{\nu} = \frac{\sum_{i=1}^{n} p_i^2 \cdot \nu_i}{\sum_{i=1}^{n} p_i^2} \tag{4.4}$$

mit n der Anzahl der Messpunkte. Die statistischen Fehler in den Anregungswahrscheinlichkeiten schlagen sich auch auf die so bestimmte mittlere Frequenz nieder. Gaußsche Fehlerfortpflanzung ergibt

$$\Delta\overline{\nu} = \frac{2}{\sum_{i=1}^{n} p_i^2} \sqrt{\sum_{i=1}^{n} \left(p_i^2 \left(\nu_i - \overline{\nu}\right)^2 \left(\Delta p_i\right)^2 \right)} \tag{4.5}$$

Der gewichtete Mittelwert liefert bei beliebigen symmetrischen Strukturen um die Resonanz herum korrekte Ergebnisse, sodass auch Spektren mit schlecht eingestellter Laserleistung ausgewertet werden können.

Durch die automatische Bestimmung können auf einfache Weise Langzeitdrifts in der Resonanzfrequenz untersucht werden. In unserem Experiment werden die beiden Übergänge in die magnetischen Unterzustände $m = +\frac{5}{2}$ und $m = +\frac{3}{2}$ des $D_{5/2}$-Niveaus verwendet. Da beide Übergänge unterschiedlich stark von Magnetfeldern beeinflusst werden, können Drifts in den Frequenzen beider Übergänge in Drifts im Magnetfeld und in der Laserfrequenz umgerechnet werden. Abbildung 4.5 zeigt die berechneten Drifts über mehrere Tage. Die Laserfrequenz ändert sich gleichmäßig mit konstanter Steigung über einen langen Zeitraum hinweg. Die Steigung wurde aus einem Fit der Daten zu

$$\frac{\Delta\nu}{\mathrm{d}} = (8{,}88 \pm 0{,}03)\,\frac{\mathrm{kHz}}{\mathrm{d}} \tag{4.6}$$

bestimmt. Die Drifts des Magnetfeldes wiederholen sich periodisch von Tag zu Tag. Es liegt also Nahe, dass diese von Geräten verursacht werden, die täglich an- und ausgeschaltet werden. Da die beiden Magnetfeldspulen der Falle rund um die Uhr mit Strom versorgt werden, scheiden diese als Ursache aus. Eine mögliche Ursache könnte der Kalzium-Ofen sein, der direkt neben der Falle angebracht ist. Er wird von einem

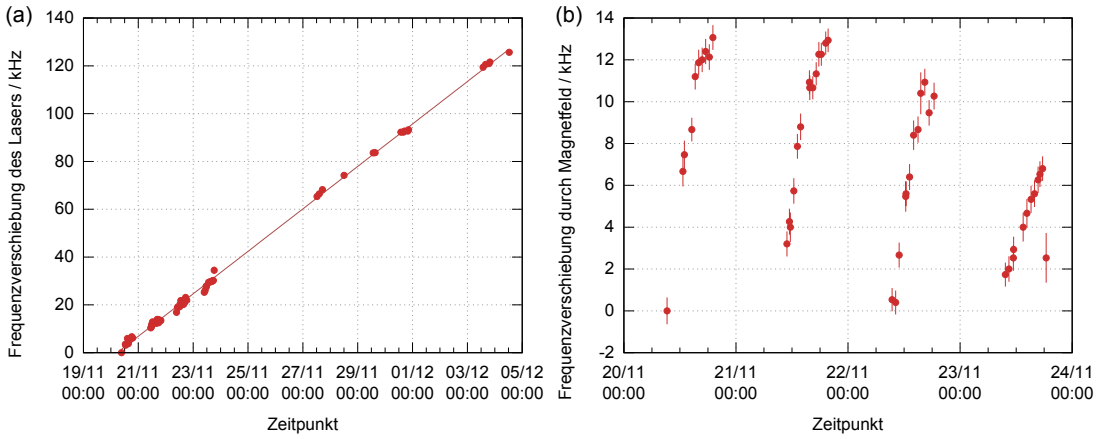


Abbildung 4.5.: Drifts der (a) Laserfrequenz über zwei Wochen mit linearem Fit und (b) der Übergangsfrequenz aufgrund des Magnetfeldes innerhalb vier Tagen

Strom höher als 1 Ampere versorgt, und nach jedem Tag abgeschaltet. Widerstandsänderungen des Ofens über den Tag könnten eine leichte Veränderung des Stroms und damit des erzeugten Magnetfeldes zur Folge haben.

Beim Shelving in experimentellen Sequenzen muss die Population des $|\uparrow\rangle$-Zustandes möglichst effizient in den metastabilen $D_{5/2}$-Zustand gebracht werden. Dazu wird die Technik der *rapid adiabatic passage* (RAP) verwendet. Diese Technik ist robust gegen Fluktuationen in der Laserfrequenz und der Laserleistung, sodass diese Parameter weniger häufig kalibriert werden müssen. Das wird erreicht, indem die Laserfrequenz während des Pulses linear über die Resonanz gefahren wird und gleichzeitig die Laserintensität in Form einer Gaußkurve variiert wird. Abbildung 4.6 veranschaulicht die Wirkung des Laserpulses auf der Blochkugel. Im Idealfall sollte die Verstimmung den

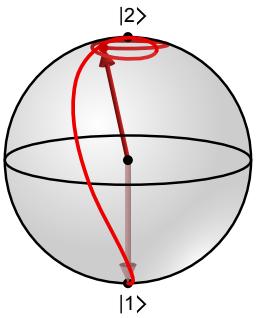


Abbildung 4.6.: Ein RAP-Puls visualisiert auf der Bloch-Kugel. Sind die Frequenz und die Dauer des Pulses nicht perfekt gewählt, so dreht sich der Bloch-Vektor am Ende des Pulses aufgrund der hohen Verstimmung um die z-Achse. Dadurch bleibt die Population in der Nähe der Maximums.

gesamten Bereich von $-\infty$ bis $+\infty$ durchlaufen - im Experiment ist es optimal, eine möglichst große Verstimmung zu wählen, bei der noch keine anderen Übergänge angeregt werden. Um die Effizienz weiter zu erhöhen, werden RAP-Pulse sowohl auf dem $\Delta m = +1$-Übergang als auch auf dem $\Delta m = +2$-Übergang ausgeführt. Genauere Untersuchungen von RAP-Pulsen in unserem Experiment wurden in [Hub10] durchgeführt.

4.4. Seitenbandkühlen

Das Seitenbandkühlen wird unmittelbar nach dem Dopplerkühlen ausgeführt und präpariert gefangene Ionen im Bewegungsgrundzustand. Die Phononenzahl im Bewegungszustand nach dem Dopplerkühlen ist thermisch verteilt [LBMW03], d.h. mit der Verteilungsfunktion

$$p_n = \frac{\overline{n}^n}{(\overline{n}+1)^{n+1}} \tag{4.7}$$

mit $\overline{n}$ dem Erwartungswert der Phononenzahl. Abbildung 4.7a zeigt eine Beispielverteilung mit $\overline{n} = 20$, was ungefähr den Gegebenheiten in unserem Experiment entspricht. Man erkennt, dass Phononenzahlen weit höher als 20 noch signifikant besetzt sind. Die Aufgabe des Seitenbandkühlens ist nun, die Population sämtlicher angeregter Bewegungszustände zu leeren, sodass ausschließlich der Grundzustand besetzt ist. Dazu werden Ramanübergänge auf dem roten Seitenband verwendet. Ein Laserpuls auf dem roten Seitenband führt zu einer Verringerung der Phononenzahl um 1 und einer Umkehr der Ionenspins. Die Ionen werden danach durch optisches Pumpen mit dem 397 nm-Laser wieder im Zustand $|\uparrow\rangle$ initialisiert. Diese Prodezur wird mehrfach wiederholt, bis der Grundzustand erreicht ist (Abbildung 4.7).

Es können keine exakten π-Pulse auf dem roten Seitenband getrieben werden, da die π-Zeit für jede besetzte Phononenzahl unterschiedlich ist. Daher sind die roten Seitenband-Pulse nur begrenzt erfolgreich, was dazu führt, dass eine große Anzahl an Pulsen aus-

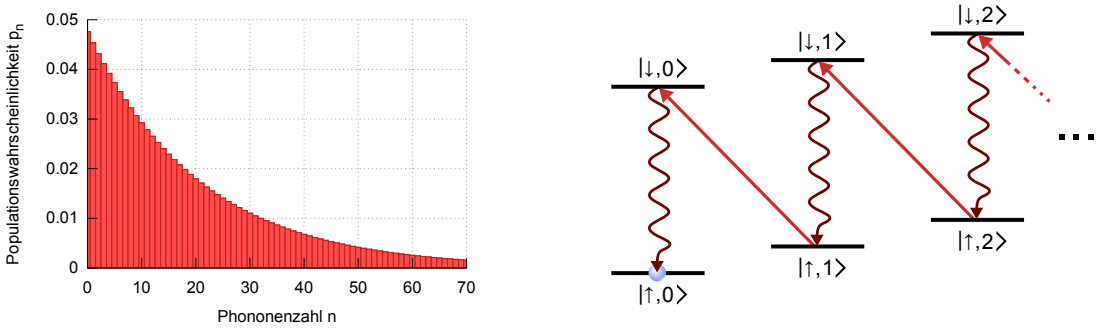


Abbildung 4.7.: (a) Phononenverteilung eines thermischen Zustandes für $\overline{n} = 20$ (b) Prinzip des Seitenbandkühlens

geführt werden muss. In der Praxis hat es sich bewährt, zunächst 30 Laserpulse auf dem zweiten roten Seitenband gefolgt von 30 Laserpulsen auf dem ersten roten Seitenband durchzuführen. Die Übergangsmatrixelemente des zweiten roten Seitenbandes unterscheiden sich für Phononenzahlen zwischen 20 und 40 kaum von denen des ersten Seitenbandes, sodass hier die Sequenz beschleunigt werden kann, da zwei Phononen mit einem Anregungspuls entfernt werden können. Im Laboralltag werden so regelmäßig Zustände mit mittleren Phononenzahlen $\overline{n} < 0.1$ erzeugt.

4.5. Raman-Kalibrationssequenzen

Da der Ramanlaser für kohärente Operationen auf den Qubit-Zuständen zuständig ist, ist eine genaue Kalibration der Parameter dieses Lasers von besonderer Bedeutung. Mit kohärenten Operationen sind einerseits das geometrische Phasengatter zur Verschränkung zweier Ionen [Ott12] gemeint, andererseits aber auch Einzelqubit-Operationen wie π-Pulse und $\frac{\pi}{2}$-Pulse. Die Pulszeiten müssen für beide Pulsarten einzeln bestimmt werden, da die Dauer von π-Pulsen aufgrund der An- und Abschaltvorgänge der Akustooptischen Modulatoren nicht exakt doppelt so lange wie die Dauer von $\frac{\pi}{2}$-Pulsen ist. Messsequenzen zur Bestimmung beider Zeiten, die sich besonders gut zur automatischen Durchführung eignen, werden in den folgenden Abschnitten vorgestellt.

Bevor aber die Pulszeiten kalibriert werden können, müssen zunächst die Frequenzen der Übergänge ermittelt werden. Anders als beim 729 nm-Laser werden hier nicht nur Trägerübergänge, sondern beispielsweise zum Seitenbandkühlen oder zur Bestimmung des Bewegungszustandes von Ionen auch verschiedene Seitenbandübergänge verwendet. Wie beim 729 nm-Laser können die Frequenzen aller Übergänge mittels einer automatischen Spektroskopiesequenz ermittelt werden. Da hier allerdings keine RAP-Pulse zum Einsatz kommen können, ist besonders hohe Präzision gefragt. Um diese zu erreichen, kann nach der gewöhnlichen Spektroskopie eine Ramsey-Spektroskopiesequenz gestartet werden. Diese ermöglicht eine höhere Präzision, und kann außerdem leicht automatisiert werden.

4.5.1. Die Ramsey-Spektroskopiesequenz

Die hier vorgestellte Ramsey-Spektroskopiesequenz wurde nach dem Vorbild von [LGRS04] implementiert. Es werden zwei Ramsey-Experimente ausgeführt, d.h. nach der Präparation gefangener Ionen werden zwei durch eine Wartezeit getrennte $\frac{\pi}{2}$-Pulse ausgeführt und dann der Zustand der Ionen ausgelesen (Abbildung 4.8). Die beiden Experimente unterscheiden sich in der Phase des zweiten Pulses: Beim ersten Sequenzteil beträgt sie $\theta = +90°$, beim zweiten Teil $\theta = -90°$.

Die Wirkung der Sequenz auf die Zustände von Ionen kann man sich auf der Blochkugel visualisieren. Der erste $\frac{\pi}{2}$-Puls dreht den Anfangszustand auf den Äquator der

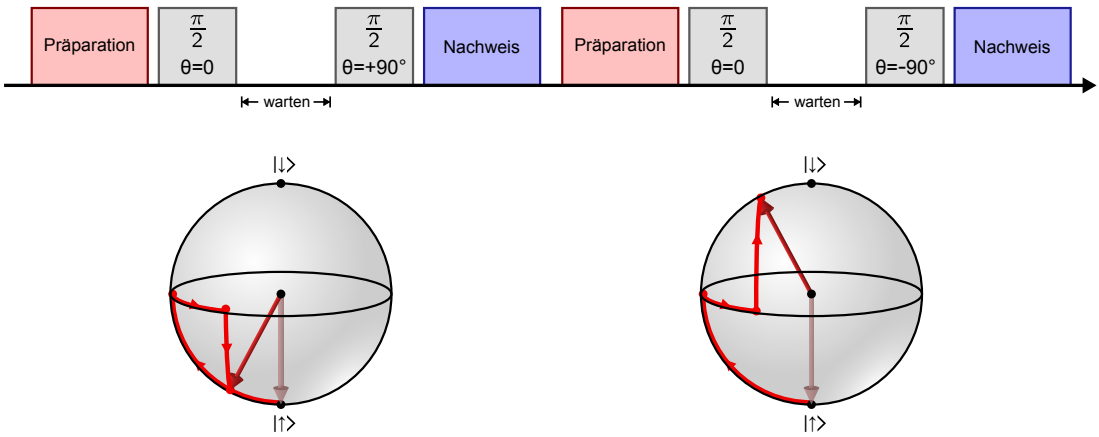


Abbildung 4.8.: Ablauf der Ramsey-Spektroskopiesequenz sowie die Veranschaulichung beider Sequenzteile auf der Blochkugel

Kugel. Während der folgenden Wartezeit dreht sich der Blochvektor auf dem Äquator um die z-Achse, wobei die Winkelgeschwindigkeit proportional zur Verstimmung des Lasers ist. Die unterschiedlichen Phasen der zweiten Pulse bewirken eine Änderung der Drehachse, gegenüber dem ersten Puls genau um den Winkel +90° bzw. -90°, sodass die neue Drehachse genau parallel bzw. antiparallel zum Blockvektor nach dem ersten $\frac{\pi}{2}$-Puls liegt. Variiert man die Verstimmung des Lasers und bestimmt die Population der Zustände nach den Ramsey-Experimentteilen, so erhält man um 180° verschobene Interferenzmuster (Abbildung 4.9). Diese beiden Interferenzmuster werden nun voneinander subtrahiert. Bei Resonanz befindet sich dann ein Nulldurchgang, an den man den Laser leicht automatisch ausrichten kann. Aus diesem Grund wurde diese Sequenz auch zur Verwendung in optischen Atomuhren vorgeschlagen [LGRS04].

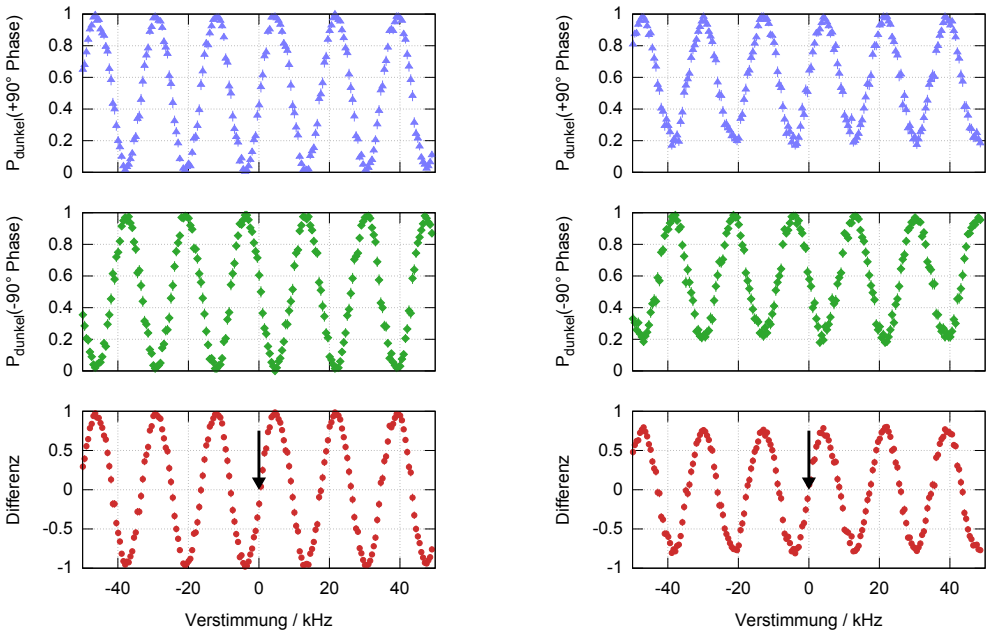
Automatische Messung 4:

Aufgrund der Periodizität des Signals ist diese Sequenz nur anwendbar, wenn die Laserfrequenz bereits mit einer Spektroskopiesequenz gemessen wurde. Es wird zunächst ein Messpunkt bei der momentan eingestellten Laserfrequenz gemessen. Ist der Wert des ersten Punktes größer als Null, so müssen weitere Punkte bei niedrigerer Frequenz, ansonsten bei höherer Frequenz aufgenommen werden, bis der Nulldurchgang gefunden ist. Um zufällige Schwankungen der Werte durch Rauschen zu vermindern, werden die Messpunkte mithilfe eines Binomialfilters geglättet. Die Frequenz ν_{neu} des Nulldurchgang wird durch lineare Interpolation der beiden benachbaren Messpunkte (d_1, ν_1) und (d_2, ν_2) bestimmt. Es ist

$$\nu_{neu} = \nu_1 - d_1 \frac{\nu_2 - \nu_1}{d_2 - d_1} \tag{4.8}$$

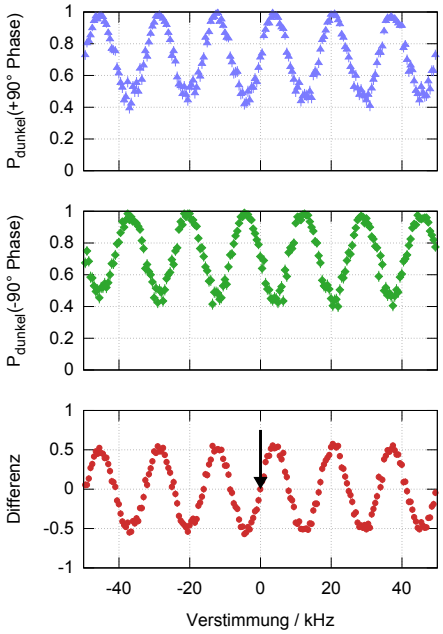
Gaußsche Fehlerfortpflanzung ergibt dann

$$\Delta\nu_{neu} = \frac{\nu_2 - \nu_1}{(d_2 - d_1)^2} \sqrt{(\nu_1 \Delta\nu_2)^2 + (\nu_2 \Delta\nu_1)^2} \tag{4.9}$$



(a) Cocarrier-Übergang

(b) erstes rotes Seitenband



(c) zweites rotes Seitenband

Abbildung 4.9.: Ergebnisse der Ramsey-Spektroskopie für verschiedene Übergänge. Die schwarzen Pfeile zeigen das Ergebnis der automatischen Auswertung.

Der Vorteil der Differenzbildung besteht darin, dass der Nulldurchgang unabhängig von Drifts in der Signalamplitude der beiden Ramsey-Sequenzteile ist. Fehler in der Dauer der $\frac{\pi}{2}$-Pulse, bei der Detektion, beim Shelving und beim optischen Pumpen führen zwar zu einer Reduzierung der Amplitude, der Nulldurchgang bleibt jedoch erhalten. Lediglich die Genauigkeit des Verfahrens nimmt dann aufgrund der geringeren Steigung ab. Das kann man auch eindrucksvoll in Abbildung 4.9 erkennen. Im Gegensatz zum Cocarrier-Übergang, bei dem man die volle Signalamplitude erhalten kann, ist die Amplitude der beiden roten Seitenbänder reduziert. Das liegt an der thermischen Verteilung des Bewegungszustandes nach dem Dopplerkühlen: Da die beiden Laserpulse Rabioszillationen mit vielen verschiedenen Frequenzen treiben, ist die $\frac{\pi}{2}$-Zeit nicht mehr eindeutig definierbar, sondern lediglich eine grobe Abschätzung möglich. Der Cocarrier-Übergang ist hingegen unabhängig vom Bewegungszustand der Ionen. Trotzdem existiert bei den roten Seitenbändern nach der Differenzbildung ein Nulldurchgang, der zur Resonanzbestimmung genutzt werden kann. Die roten Seitenbänder sind für alle Kalibrationssequenzen eine besondere Herausforderung, da diese zur Seitenbandkühlung zwingend schon im dopplergekühlten Zustand verwendet werden müssen. Andere Übergänge können auf Ionen im Grundzustand angewandt werden, wo nur eine Phononenzahl besetzt ist und somit nur noch Rabioszillationen mit genau einer Frequenz getrieben werden. Die roten Seitenbänder hingegen verschwinden im Grundzustand vollständig.

Wie beim 729 nm-Laser auch kann nun die eingestellte Frequenz am akustooptischen Modulator über mehrere Tage untersucht werden (Abbildung 4.10a). Da bei Ramanübergängen die effektive Frequenz unabhängig von der Frequenz der einzelnen Laserstrahlen ist, wird der Drift ausschließlich durch Magnetfeldfluktuationen verursacht. Der Verlauf deckt sich deshalb wie erwartet mit den Drifts des 729 nm-Übergangs.

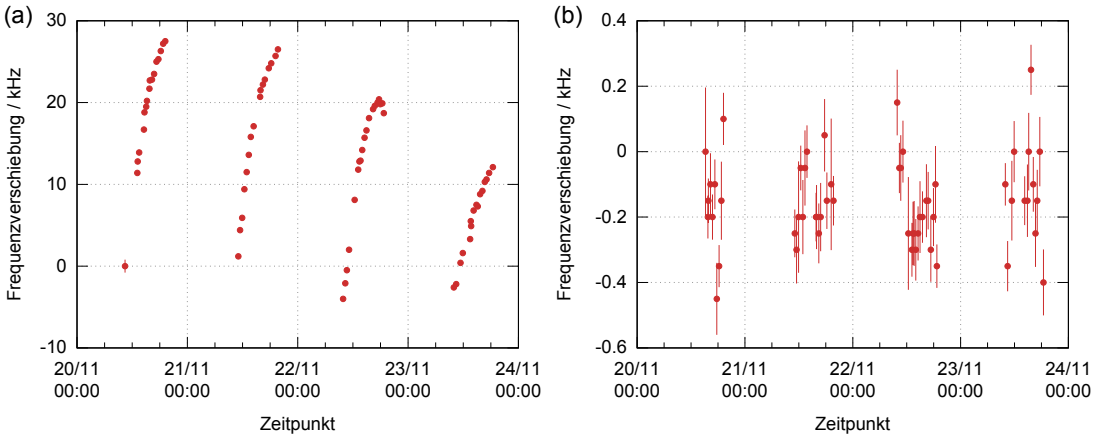


Abbildung 4.10.: Frequenzdrifts (a) des CoCarrier-Übergangs und (b) in der axialen Fallenfrequenz

Durch Messung der Frequenz verschiedener Seitenbänder kann außerdem der Drift in der axialen Fallenfrequenz betrachtet werden (Abbildung 4.10b). Fluktuationen konnten hier nicht festgestellt werden, die Fallenfrequenz ist im sub-kHz-Bereich stabil.

4.5.2. $\frac{\pi}{2}$-Kalibration

Im Praxisbetrieb hat sich die automatische Ramsey-Spektroskopie als sehr zuverlässig und schnell herausgestellt. Daher wird nun das Prinzip, Nulldurchgänge durch Differenzbildung zu erzeugen, als Basis für weitere automatische Messungen genutzt. Zur Kalibration der $\frac{\pi}{2}$-Zeit eines Übergangs kann das Prinzip durch eine Verkettung mehrerer Laserpulse erreicht werden (Abbildung 4.11). Im ersten Sequenzteil wird ein einzelner Laserpuls über einen Zeitraum τ, der in der Messung variiert wird, eingestrahlt. Im zweiten Teil der Sequenz wird dieser Laserpuls dann dreimal hintereinander ausgeführt. Entspricht τ genau der $\frac{\pi}{2}$-Zeit, so landet der Blochvektor des Zustandes in beiden Sequenzteilen auf dem Äquator, d.h. die Popupation der Zustände ist identisch. Varriiert man die Pulszeit, so nimmt man Rabi-Oszillationen auf, wobei die Frequenz der Oszillationen im zweiten Teil dreimal so hoch ist wie im ersten Teil. Nach der Differenzbildung entsteht bei der richtigen $\frac{\pi}{2}$-Zeit ein Nulldurchgang (Abbildung 4.12).

Automatische Messung 5:
Bei der automatischen Auswertung der Messdaten wird analog zur Ramsey-Spektroskopie vorgegangen. Man misst zunächst einen Punkt bei der zuletzt bekannten $\frac{\pi}{2}$-Zeit und erhöht dann die Pulszeit, falls die Differenz der beiden Sequenzteile größer als Null ist, ansonsten verringert man sie, bis der Nulldurchgang gefunden ist.

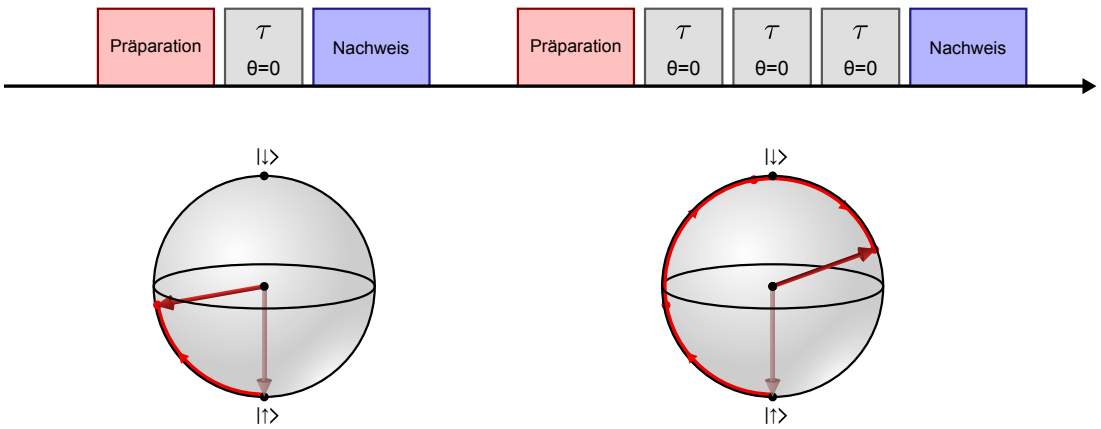
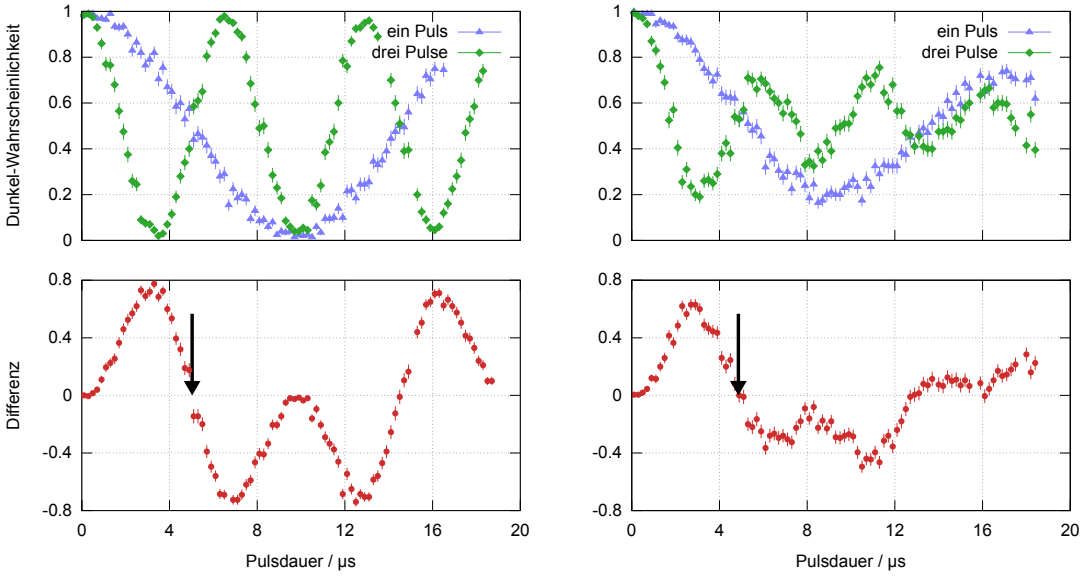
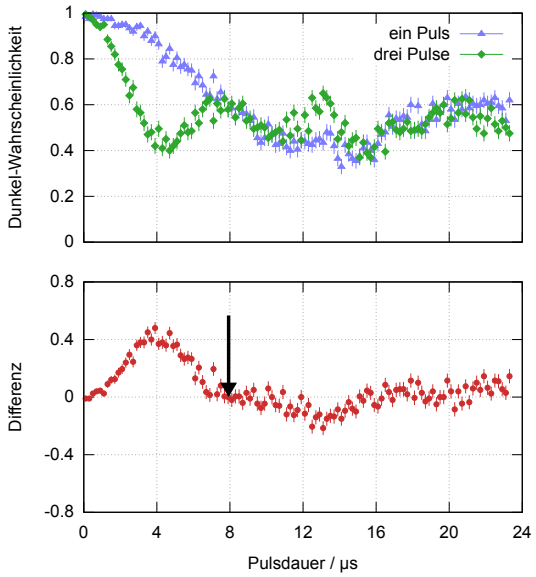


Abbildung 4.11.: Ablauf der $\frac{\pi}{2}$-Kalibrationssequenz sowie die Veranschaulichung beider Sequenzteile auf der Blochkugel



(a) Cocarrier-Übergang

(b) erstes rotes Seitenband



(c) zweites rotes Seitenband

Abbildung 4.12.: Ergebnisse der $\frac{\pi}{2}$-Kalibrationssequenz für verschiedene Übergänge. Die schwarzen Pfeile zeigen das Ergebnis der automatischen Auswertung.

Wie bei der Ramsey-Spektroskopie auch beobachtet man bei den roten Seitenbändern einen massiven Kontrastverlust. Beim ersten roten Seitenband reicht der Kontrast zur Bestimmung des Nulldurchgangs noch aus, beim zweiten roten Seitenband kann die Messung nicht mehr zuverlässig ausgeführt werden. Daher muss die $\frac{\pi}{2}$-Zeit des zweiten roten Seitenbandes weiterhin manuell abgeschätzt werden.

Bei Bedarf kann die Genauigkeit der Sequenz erhöht werden, indem in beide Sequenzteile eine zusätzliche gerade Anzahl an Laserpulsen mit der Pulszeit τ hinzugefügt werden. Damit werden die Oszillationen schneller und somit die Steigung im Nulldurchgang erhöht, im Gegenzug wird aber auch der Bereich um den Nulldurchgang mit negativer Steigung schmaler. Auch wird dadurch die gesamte Pulsfläche in den Sequenzteilen höher, was die Genauigkeit in den roten Seitenbändern weiter verschlechtert. Beim Cocarrier-Übergang kann die Anzahl der Laserpulse aber ohne Bedenken erhöht werden.

4.5.3. π-Kalibration

Auch die π-Kalibrationssequenz besteht aus zwei Sequenzteilen (Abbildung 4.13). Beim ersten Teil wird lediglich ein $\frac{\pi}{2}$-Puls ausgeführt. Beim zweiten Sequenzteil kommen zum $\frac{\pi}{2}$-Puls zwei weitere Pulse hinzu, deren Pulszeit variiert wird. Entspricht die Pulszeit der beiden Pulse genau der π-Zeit, so befindet sich der Bloch-Vektor bei beiden Sequenzteilen am selben Ort. Abbildung 4.14 zeigt Messergebnisse dieser Kalibrationssequenz. Bildet man die Differenz der Populationen beider Sequenzteile, so erhält man einen Nulldurchgang bei der korrekten π-Zeit. Der Nulldurchgang ist unabhängig von der eingestellten $\frac{\pi}{2}$-Zeit, lediglich die Genauigkeit des Verfahrens wird durch eine nicht perfekte Einstellung vermindert. Das kann man sich leicht an der Bloch-Kugel klarmachen: Auf dem Äquator ändert sich bei Variation der Pulszeit die Population der Zustände am meisten, je weiter man sich vom Äquator entfernt, desto weniger wirkt sich die Pulszeit auf die Population aus.

Automatische Messung 6:
Als ein guter Schätzwert kann die doppelte $\frac{\pi}{2}$-Zeit als Startwert verwendet werden. Da diese Sequenz einen $\frac{\pi}{2}$-Puls verwendet, bietet es sich an, sie unmittelbar nach der $\frac{\pi}{2}$-Kalibration auszuführen. Die Suche des Nulldurchgangs wird analog zur Ramsey-Spektroskopiesequenz und zur $\frac{\pi}{2}$-Kalibrationssequenz durchgeführt.

Ähnlich wie bei der $\frac{\pi}{2}$-Kalibration kann auch hier durch Hinzufügen weiterer Laserpulse mit der Pulszeit τ in beiden Sequenzteilen die Steigung um den Nulldurchgang erhöht werden. Auch hier muss sich dann für eine erfolgreiche Kalibrierung der Startwert näher am wahren Wert befinden.

Die regelmäßige Messung der $\frac{\pi}{2}$-Zeit bzw. der π-Zeit ermöglicht es, den Drift in der Rabi-Frequenz der Übergänge über lange Zeit zu dokumentieren. Abbildung 4.15 zeigt

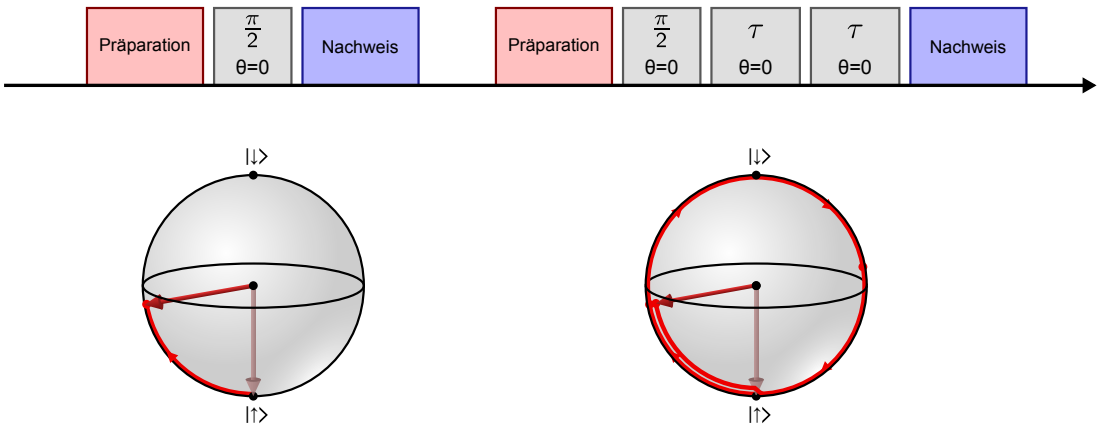


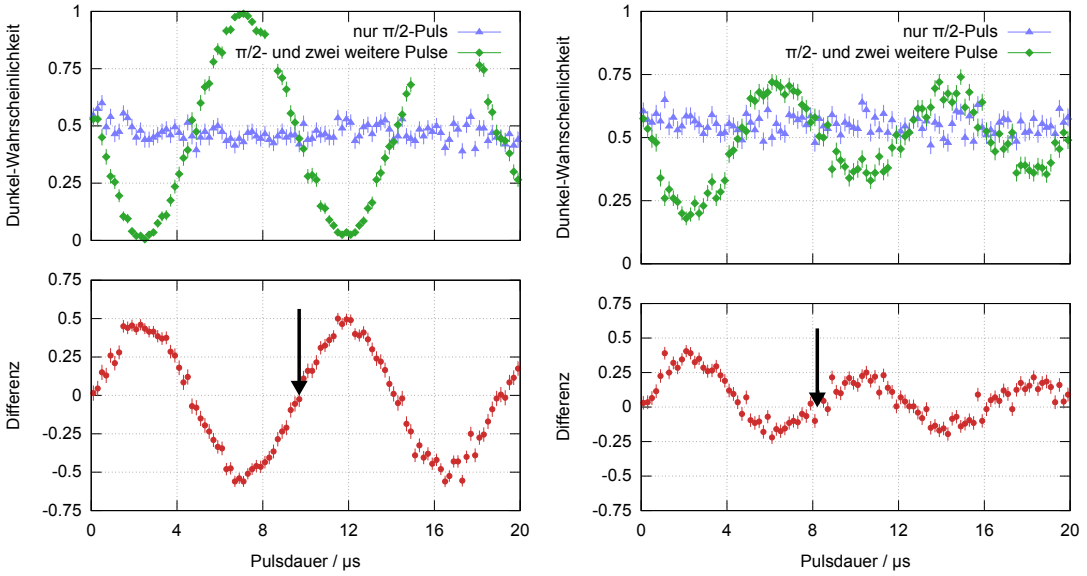
Abbildung 4.13.: Ablauf der π-Kalibrationssequenz sowie die Veranschaulichung beider Sequenzteile auf der Blochkugel, wobei die Pulszeit des $\frac{\pi}{2}$-Pulses nicht perfekt eingestellt ist.

die Rabi-Frequenz des Cocarrier-Übergangs für mehrere Messtage. Man kann deutliche Drifts von teilweise über 10% pro Tag erkennen. Da für erweiterte Sequenzen, wie z.B. Ramsey-Sequenzen oder die später im Rahmen der Transportexperimente verwendete Pseudoenergie-Sequenz genau eingestellte Pulszeiten notwendig sind, sollte die Kalibration der Pulszeiten etwa alle 30 Minuten durchgeführt werden.

Da die Intensität des Raman-Lasers aktiv stabilisiert wird, kann der Effekt nicht von Intensitätsdrifts des Lasers verursacht werden. Mögliche Ursachen könnten Positionsdrifts der Laserstrahlen sein, oder thermische Drifts der Wellenplatten, welche die Intensitätsaufspaltung des Ramanlasers in die drei verschiedenen Strahlen bestimmen. Eventuell ändert sich auch die Beugungseffizienz der verwendeten akustooptischen Modulatoren über die Zeit. Da die Polarisation der Strahlen gefiltert wird, können auch Polarisationsdrifts eine Ursache sein.

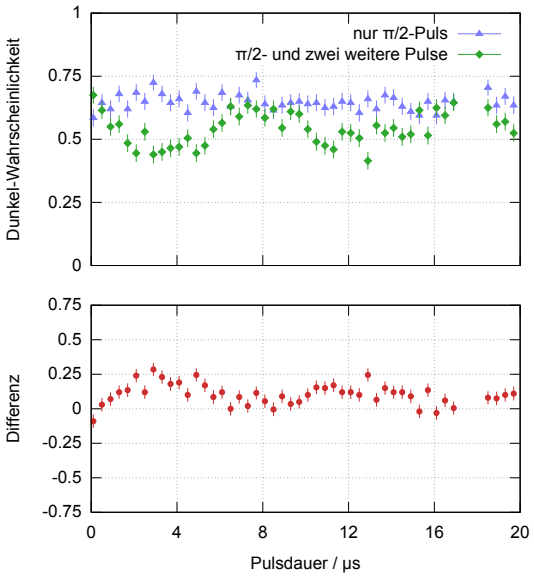
4.6. Detektionsverfahren

Während jeder experimentellen Sequenz werden Kamerabilder aufgenommen, die nach der Messung ausgewertet werden müssen. Bei der Auswertung geht es darum, dunkle Ionen, bei denen das Shelving erfolgreich war, und helle Ionen, bei denen das Shelving keine Wirkung hatte, zu unterscheiden. Das Aufnehmen der Kamerabilder nimmt in den meisten experimentellen Sequenzen den größten Zeitraum in Anspruch ($\approx 30\%$ der Sequenz), daher sollte die Belichtungszeit der Bilder möglichst so kurz gewählt sein, dass Ionen gerade noch vom Hintergrund unterschieden werden können. Außerdem wird durch kurze Belichtungszeiten die Wahrscheinlichkeit eines Zerfalls des metastabilen $3D_{5/2}$-Zustandes minimiert.



(a) Cocarrier-Übergang

(b) erstes rotes Seitenband



(c) zweites rotes Seitenband

Abbildung 4.14.: Ergebnisse der π-Kalibrationssequenz für verschiedene Übergänge. Die schwarzen Pfeile zeigen das Ergebnis der automatischen Auswertung. Beim zweiten roten Seitenband lässt sich die π-Zeit nicht automatisch bestimmen.

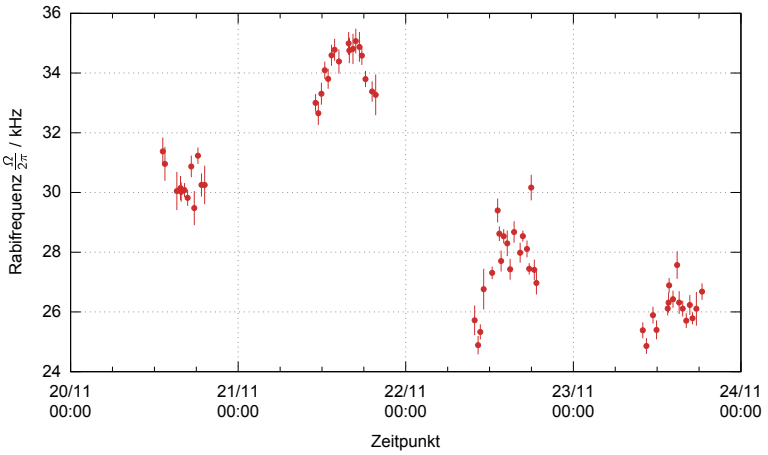
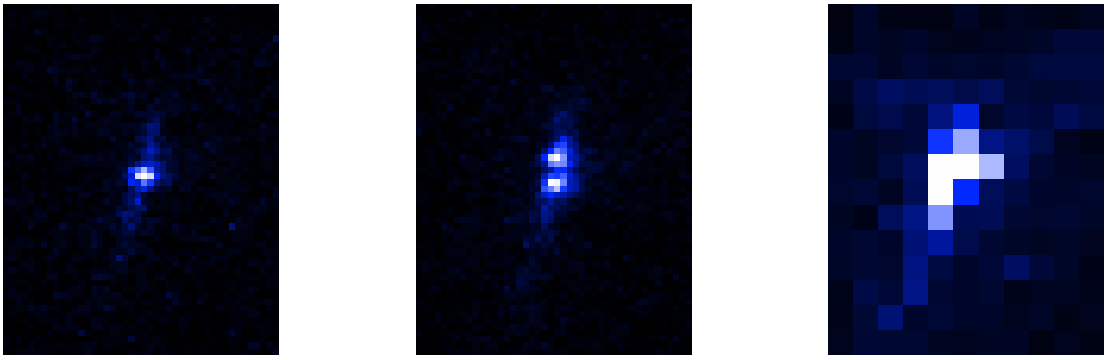


Abbildung 4.15.: Verlauf der Rabifrequenz des Cocarrier-Übergangs über mehrere Tage

Ist keine örtliche Information zur Unterscheidung mehrerer Ionen nötig, so kann anstatt der EMCCD-Kamera auch ein Photomultiplier zur Ionendetektion verwendet werden. Im Gegensatz zu EMCCD-Kameras weisen Photomultiplier geringeres elektronisches Rauschen auf und ermöglichen somit kürzere Detektionszeiten, in [MSW+08] konnte so ein einzelnes Ion in einer Detektionszeit von 145 μs mit einer Wahrscheinlichkeit von 99,99% korrekt detektiert werden. Um die selbe Erfolgswahrscheinlichkeit mit einer EMCCD-Kamera zu erreichen, war eine Detektionszeit von 400 μs notwendig [BSWL10]. Das gelang nur, da mithilfe geeigneter Verfahren die durch die Kamera verfügbare örtliche Information optimal ausgenutzt wurde. Diese Verfahren wurden für unser Experiment im Wesentlichen übernommen und werden daher in den folgenden Abschnitten vorgestellt.

Abbildung 4.16a zeigt ein Kamerabild eines gefangenen Ions, aufgenommen von der EMCCD-Kamera. Zur optimalen Detektion sollte die Abbildungsoptik so ausgerichtet sein, dass Ionen möglichst punktförmig abgebildet werden. Das ist in unserem Experiment nicht der Fall, und lässt sich nur durch einen neuen experimentellen Aufbau erreichen, der bereits in Arbeit ist. Die Abbildungsfehler wirken sich vor allem auf die Detektion von zwei (oder mehreren) Ionen aus (Abbildung 4.16b): Die Ionen überlappen, sodass es unmöglich wird, ihnen vollständig getrennte Bereiche zuzuordnen. So entstehen vor allem Detektionsfehler in den *ungeraden Zuständen*, d.h. den Zuständen, bei denen nur eines beider Ionen Fluoreszenzlicht emittiert. Es ist aber gelungen, ein Auswerteverfahren zu entwickeln, welches robust gegen durch Überlappung der Ionen entstehende Fehler ist.

Darüber hinaus treten weitere Störquellen in unserem Experiment auf. Erstens ist der Aufbau der Detektionsoptik nicht vollständig lichtdicht, sodass Umgebungslicht den Kamerasensor erreichen kann. Ein zweites, generelles Problem von mikrostrukurierten Fallen ist an den Fallenelektroden der Mikrofalle gestreutes Laserlicht, welches die Ka-



(a) (b) (c)

Abbildung 4.16.: Kamerabilder von (a) einem einzelnen Ion (b) zwei in einem Fallensegment gefangenen Ionen (c) einem einzelnen Ion mit eingeschaltetem 4x4-Binning

mera erreicht. Durch die Summe aller in unserem Experiment vorliegenden Probleme beträgt die Detektionszeit in der Regel einige Millisekunden.

Abbildung 4.16c visualisiert eine Funktion der EMCCD-Kamera, das *Binning*. Dabei werden benachbarte Kamerapixel noch vor der Ausleseprozedur zu größeren *Superpixeln* gruppiert, was das Signal zu Rauschen-Verhältnis verbessert [Bur10] und somit eine Reduzierung der Detektionszeit um etwa die Hälfte ermöglicht. Für Kristalle aus mehreren Ionen ist diese Technik in unserem Experiment nicht gewinnbringend anwendbar, da sie so nicht mehr räumlich unterschieden werden können.

Der nächste Abschnitt beschreibt zunächst das einfachste mögliche Ausleseverfahren, den *Schwellwert-Algorithmus*. Danach wird das neu entwickelte *Support-Vector-Machine-Verfahren*, mit dem, wie schon angedeutet, die Detektionsfehler in den ungeraden Zuständen bekämpft werden können. Im letzten Abschnitt wird das *Maximum-Likelihood-Verfahren* vorgestellt, welches die durch die EMCCD-Kamera verfügbare räumliche Information am besten ausnutzt.

4.6.1. Der Schwellwert-Algorithmus

Bei der einfachsten Variante des Schwellwert-Algorithmus müssen einmal per Hand rechteckige Regionen, die *Regions of Interest*, um die von Ionen beleuchteten Kamerapixel platziert werden. Nur diese Pixel werden bei der späteren Auswertung berücksichtigt. Bei der Platzierung muss darauf geachtet werden, möglichst wenige nichtbeleuchtete Pixel einzubeziehen, um Rauschen dieser Pixel herauszufiltern. Da abgebildete Ionen aber nicht rechteckig sind, gehen dabei aber auch zwangsläufig beleuchtete Pixel und damit verfügbare Informationen verloren.

Bei der Auswertung von Kamerabildern in experimentellen Sequenzen wird dann der gesamte Fluoreszenzwert der gesetzten Regionen mit einer vorher festgelegten Schwelle σ verglichen. Ist der Wert höher, wird ein Ion als hell gewertet, ist er niedriger, wird es als dunkel gewertet. Die Schwelle kann automatisch mithilfe einer Histogramm-Sequenz bestimmt werden.

Automatische Messung 7:
Zur automatischen Bestimmung der Schwelle σ werden abwechselnd Kamerabilder mit fluoreszierenden Ionen und mit nicht fluoreszierenden Ionen aufgenommen. Die Fluoreszenz kann gezielt gesteuert werden, indem der Laser bei 866 nm an- bzw. ausgeschaltet wird. Diese Sequenz kann die Detektion eines dunklen Ions fast perfekt simulieren, es fehlt lediglich an der Falle gestreutes Laserlicht durch den 866 nm-Laser. Dieses Licht wird aber zum größten Teil von einem Filter vor der Kamera herausgefiltert, sodass der so verursachte Fehler minimiert wird. Außerdem wird die endliche Lebensdauer des $D_{5/2}$-Zustandes nicht berücksichtigt, was bei kurzen Detektionszeiten von wenigen Millisekunden aber ebenfalls vernachlässigbar ist.

Die gemessenen Fluoreszenzwerte der Regions of Interest werden dann in ein Histogram eingefügt (Abbildung 4.17). Die Schwelle kann unter der Voraussetzung, dass beide Verteilungen näherungsweise einer Poissonverteilung entsprechen [1], berechnet werden durch [Roo00]

$$\sigma = \sqrt{\overline{n}\,\overline{s}} \tag{4.10}$$

wobei $\overline{n}$ bzw. $\overline{s}$ die Mittelwerte der Verteilungen des dunklen bzw. hellen Ions sind.

Die Genaugkeit des Verfahrens lässt sich verbessern, indem beispielsweise nicht rechteckige Regions of Interest verwendet werden, oder einzelne Pixel je nach Beleuchtungsstärke Gewichtungsfaktoren erhalten [Wie11]. Bei der Detektion mehrerer Ionen muss dabei allerdings klar sein, welchem Ion ein Pixel zugeordnet werden muss. Da dies in unserem Experiment nicht einwandfrei möglich ist, wird nun ein Verfahren präsentiert, welches einen Überlapp mehrerer Ionen berücksichtigen kann.

4.6.2. Das Support-Vector-Machine-Verfahren

Der Einfachheit halber wird das Support-Vector-Machine-Verfahren zunächst am Beispiel von Zwei-Ionen-Kristallen erläutert. Später wird dann diskutiert, inwiefern es sich auf eine höhere Ionenzahl verallgemeinern lässt. Es werden zunächst genauso wie beim Schwellwert-Verfahren per Hand Regions of Interest um die jeweiligen Ionen gesetzt. Wäre die Fluoreszenz der Ionen auf dem Kamerabild klar voneinander getrennt, so ließe

[1] Aufgrund des komplexen Ausleseverfahrens einer EMCCD-Kamera trifft diese Näherung für einzelne Pixel nicht zu, in der Praxis ist die angegebene Formel aber aufgrund der Summierung über viele Pixel ausreichend.

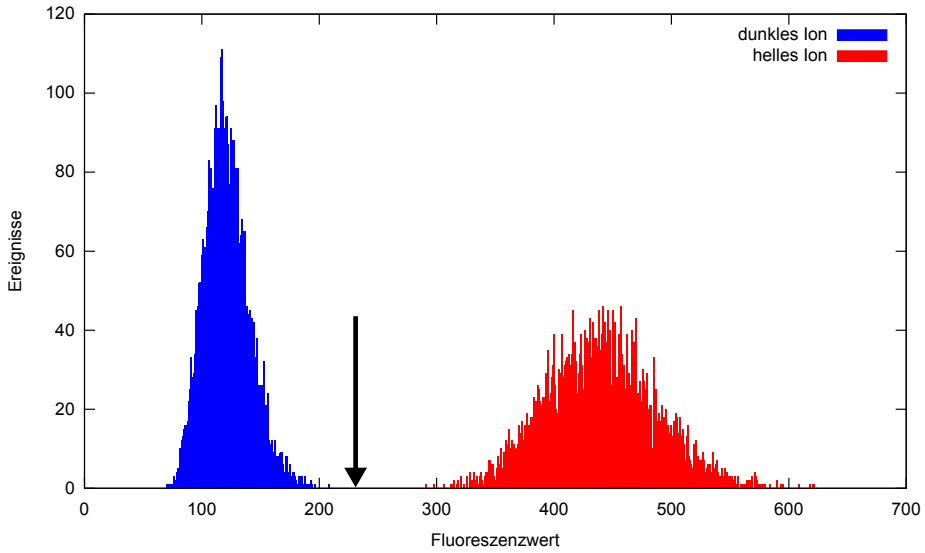


Abbildung 4.17.: Ein Beispielhistogramm mit einer Detektionszeit von 2 ms bei 4x4-Binning. Der schwarze Pfeil zeigt die ausgerechnete Schwelle σ.

sich die Unterscheidung zwischen hellen und dunklen Zuständen separat für beide Ionen durchführen. Genau das wird nun aber vermieden - stattdessen werden beide Ionen ein ganzes, zusammenhängendes System betrachtet, mit vier möglichen Zuständen, die es zu unterscheiden gilt:

$$M = \{(\text{dunkel}, \text{dunkel}), (\text{dunkel}, \text{hell}), (\text{hell}, \text{dunkel}), (\text{hell}, \text{hell})\}$$

Es ist also eine Abbildung f gesucht, die den Fluoreszenzwerten der beiden Regions of Interest den richtigen Zustand zuordnet:

$$f\colon \{f_1, f_2\} \to M$$

mit $f_{1/2}$ den Fluoreszenzwerten der jeweiligen Ionen. Zunächst wird die Summe und die Differenz der Fluoreszenzwerte gebildet, also

$$\Sigma = f_1 + f_2 \qquad \Delta = f_1 - f_2$$

Wie beim Schwellwert-Verfahren auch muss zunächst ein Datensatz aufgenommen werden, anhand dem entschieden werden kann, welchem Zustand eine Kombination (Δ, Σ) zugeordnet werden muss. In einem Graphen, in dem die Summe Σ gegen die Differenz Δ aufgetragen ist, können den vier möglichen Zuständen verschiedene, voneinander abgetrennte Bereiche zugeordnet werden (Abbildung 4.18).

In unserem Experiment sind Summe und Differenz der ungeraden Zustände korreliert, daher werden diese Bereiche nicht exakt kreisförmig, sondern elliptisch sein. Trotzdem

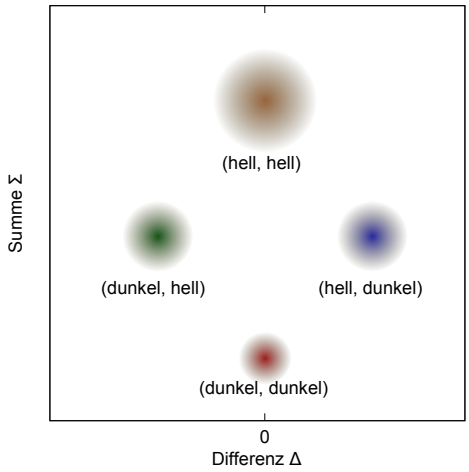


Abbildung 4.18.: Skizze der Bereiche von den vier möglichen Zuständen in einem Σ-Δ-Diagramm.

können im zweidimensionalen Raum dann immernoch Trennlinien gezogen werden, die alle Zustände eindeutig voneinander abgrenzen.

Zur Abschätzung der Bereiche muss analog zum Schwellwert-Verfahren eine Messequenz durchlaufen werden, die jeden der möglichen Zustände präpariert und dann die gemessenen Paare (Δ, Σ) als Punkte in einen Graphen einträgt. Da die Laser in unserem Experiment nicht auf nur ein einzelnes Ion fokussiert werden können, können nicht alle Zustände durch An- bzw. Abschalten des 866 nm-Lasers erzeugt werden, wie das beim einzelnen Ion der Fall ist. Eine Möglichkeit zur Präparation aller Zustände ist, wie in Abbildung 4.19 gezeigt, einen $\frac{\pi}{2}$-Puls mit dem 729 nm-Laser auszuführen. Dadurch wird jedes Ion in einem Superpositionszustand $|\text{hell}\rangle + |\text{dunkel}\rangle$ präpariert. Bei der anschließendenen Detektion werden die Ionen im Idealfall mit einer Wahrscheinlichkeit von 25% in einen der vier möglichen Zustände projeziert. Aber auch eine nicht gleichmäßige Verteilung der Zustände beeinträchtigt die Genauigkeit des Auswerteverfahrens nicht, sodass eine genaue Kalibrierung der Pulsfläche nicht nötig ist.

Die Verwendung des Lasers bei 729 nm hat den Nachteil, dass, bevor die Sequenz durchgeführt werden kann, die Frequenz des Lasers kalibriert werden muss. Für die Kalibra-

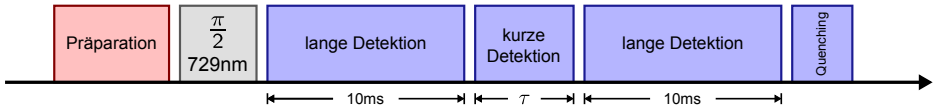


Abbildung 4.19.: Experimentelle Sequenz zur Messung eines Datensatzes für das Support-Vector-Machine-Verfahren

tion ist aber schon eine funktionierende Detektion nötig - daher kann auf die Benutzung des gewöhnlichen Schwellwert-Verfahrens nicht verzichtet werden.

Da die Zustände der beiden Ionen nach der Projektion rein zufällig sind, können die bei der Detektion gemessenen Punkte (Δ, Σ) zunächst nicht einem der möglichen Zustände zugeordnet werden. Um das zu ermöglichen, wird, wie von Burrell et al. [BSWL10] beschrieben, vor dem Detektionsabschnitt mit der eigentlich gewünschten Detektionszeit τ ein zusätzlicher Detektionsabschnitt mit einer hohen Belichtungszeit von 10ms eingefügt. Dieser wird mithilfe des Schwellwert-Verfahrens ausgewertet. So kann der wahre Zustand eindeutig festgestellt werden, denn bei einer so langen Detektionszeit hat selbst das simple Schwellwert-Verfahren eine Erfolgsquote von annähernd 100%. Um sicherzugehen, dass der Zustand während des kürzeren Detektionsabschnitts auch wirklich dem der langen Detektion entspricht, wird als weitere Sicherheit danach nochmals eine weitere lange Detektion eingefügt. Nur wenn beide langen Detektionen das gleiche Ergebnis liefern, wird das Resultat der Sequenz weiterverwendet. So können Fälle, in denen beispielsweise der $D_{5/2}$-Zustand spontan zerfällt, ausgeschlossen werden.

Wurde ein Datensatz mit obiger Sequenz aufgenommen, so erhält man vier verschiedene Punktmengen, wobei jede Punktmenge einem Zustand zugeordnet ist. Zur Auswertung von Kamerabildern in späteren Sequenzen müssen die Punktmengen noch durch Trennlinien getrennt werden. Bei der Berechnung von Trennlinien können *Support-Vector-Machines* [HDO+98] verwendet werden. Support-Vector-Machines platzieren Hyperebenen im n-Dimensionalen Raum so, dass der Abstand zu den Randpunkten zweier Mengen maximiert wird. Die Randpunkte heißen auch *Support Vectors*. Durch Transformation in höherdimensionale Räume und anschließende Rücktransformation können auch gekrümmte Trennebenen erzeugt werden. Es hat sich aber herausgestellt, dass für das hier vorliegende Problem auch ohne Verwendung einer Support-Vector-Machine gute Trennlinien gefunden werden können. Dazu wird die *Hauptkomponentenanalyse* angewendet - ein Verfahren, das analog zur Hauptachsentransformation in der klassischen Mechanik funktioniert. Man berechnet zunächst die Kovarianzmatrix der vier verschiedenen Punktmengen

$$\mathrm{Cov}\,(\Sigma, \Delta) = \frac{1}{N} \sum_{i=0}^{N} \begin{pmatrix} \left(\Delta_i - \overline{\Delta}\right)^2 & \left(\Delta_i - \overline{\Delta}\right)\left(\Sigma_i - \overline{\Sigma}\right) \\ \left(\Delta_i - \overline{\Delta}\right)\left(\Sigma_i - \overline{\Sigma}\right) & \left(\Sigma_i - \overline{\Sigma}\right)^2 \end{pmatrix} \tag{4.11}$$

mit N der Gesamtzahl an Punkten einer der vier verschiedenen Punktmengen, $\overline{\Delta}$ dem Erwartungswert der Differenzen und $\overline{\Sigma}$ dem Erwartungswert der Summen. Der Normierungsfaktor $\frac{1}{N}$ macht die Kovarianzmatrix unabhängig von der Anzahl aufgenommener Punkte.

Diese Matrix wird nun diagonalisiert, um die Eigenwerte σ und ρ zu berechnen. Die Diagonalisierung entspricht einem Basiswechsel in eine Basis, in der die Korrelation

zwischen der Differenz und der Summe der Punkte verschwindet. Die Eigenwerte entsprechen den Standardabweichungen der Punktmengen in dieser Basis. Die Eigenvektoren, multipliziert mit den dazugehörigen Standardabweichungen $\sqrt{\sigma}$ bzw $\sqrt{\rho}$ und eingezeichnet auf die Schwerpunkte der Punktmengen, spannen Ellipsen auf, welche die Orientierung und Ausdehnung der Punktmengen sehr gut beschreiben.

Für neue gemessene Punkte (Δ, Σ) lässt sich jetzt eine Vorschrift konstruieren, um die Zugehörigkeit zu einer der Punktmengen zu entscheiden. Die Fragestellung dazu lautet: Welche der vier Ellipsen muss am wenigsten hochskaliert werden, bis sie den Punkt (Δ, Σ) schneidet? Die Skalierungsfaktoren der Ellipsen s_j, wobei $j = 0, 1, 2, 3$ der Index einer der vier möglichen Zustände ist, können durch die Gleichung

$$s_j = \left(\frac{\Delta_j}{\sigma_j}\right)^2 + \left(\frac{\Sigma_j}{\rho_j}\right)^2 \tag{4.12}$$

berechnet werden, wobei (Δ_j, Σ_j) die Koordinaten des Punktes im Bezugssystem der jeweiligen Ellipse mit den Halbachsen (σ_j, ρ_j) angibt. Man erhält die Koordinaten durch die Transformation

$$\begin{pmatrix}\Delta_i \\ \Sigma_i\end{pmatrix} = R_i \cdot \left[\begin{pmatrix}\Delta \\ \Sigma\end{pmatrix} - \begin{pmatrix}\overline{\Delta}_i \\ \overline{\Sigma}_i\end{pmatrix}\right] \tag{4.13}$$

wobei R_i eine Drehmatrix ist, die durch die Eigenvektoren der jeweiligen Punktmenge bestimmt wird. Die Parameter Δ_i und Σ_i sind die Erwartungswerte der jeweiligen Zustände. Schließlich wird der Punkt (Δ, Σ) der Menge mit dem kleinsten Skalierungsfaktor zugeordnet. Der Vorteil dieses Verfahrens ist, dass keine explizite Trennlinie berechnet werden muss.

Abbildung 4.20 zeigt Beispielmessungen der Sequenz zur Bestimmung der Ellipsen. Um die Effektivität des Verfahrens zu veranschaulichen, wurden per Hand die sich aus dem oben dargestellten Verfahren ergebenden Trennlinien eingezeichnet.

Die Effektivität des Verfahrens wurde in Beispielmessungen mit verschiedenen Detektionszeiten mit dem Schwellwert-Verfahren verglichen. Abbildung 4.21 zeigt die Ergebnisse bei einer Detektionszeit von 2ms, aufgeschlüsselt nach Fehldetektionen in die falschen Zustände. Die Detektionsfehler der ungeraden Zustände konnten um einen Faktor von etwa 2-5 reduziert werden. Allerdings geht dies auf die Kosten der Fehler bei den (dunkel, dunkel)-Zuständen, diese haben um etwa einen Faktor 10 zugenommen. Abbildung 4.22 zeigt die Ergebnisse für verschiedene Detektionszeiten. Um die Effizienz beider Verfahren auf einen Blick vergleichen zu können, wurde noch die Mittelung aus den Fehlerwahrscheinlichkeiten der einzelnen Zustände als Kenngröße berechnet. Gerade für Detektionszeiten größer als 1,5 ms, die für unser Experiment von Bedeutung sind, kann das neue Support-Vector-Machine-Verfahren die Detektionsfehler um einen Faktor 2-5 reduzieren.

Betrachtet man sich die Datensätze in Abbildung 4.20, so sollte man erwarten, dass die (dunkel, dunkel)-Zustände sehr gut von den anderen Zuständen unterschieden werden können sollten. Tendenziell befinden sich die Trennlinien allerdings zu nahe an diesen

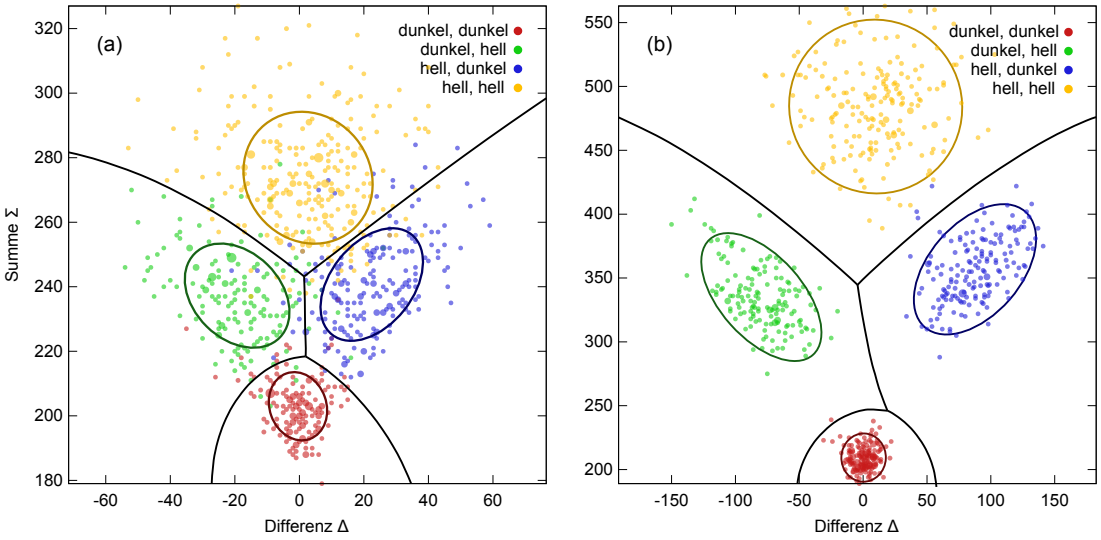


Abbildung 4.20.: Datensätze für das Support-Vector-Machine-Verfahren. Für jede Punktmenge wurde durch Diagonalisierung der Korrelationsmatrix eine Ellipse ausgerechnet und Trennlinien eingezeichnet. Bild (a) wurde bei einer Detektionszeit von 1ms, Bild (b) mit 4ms Detektionszeit aufgenommen.

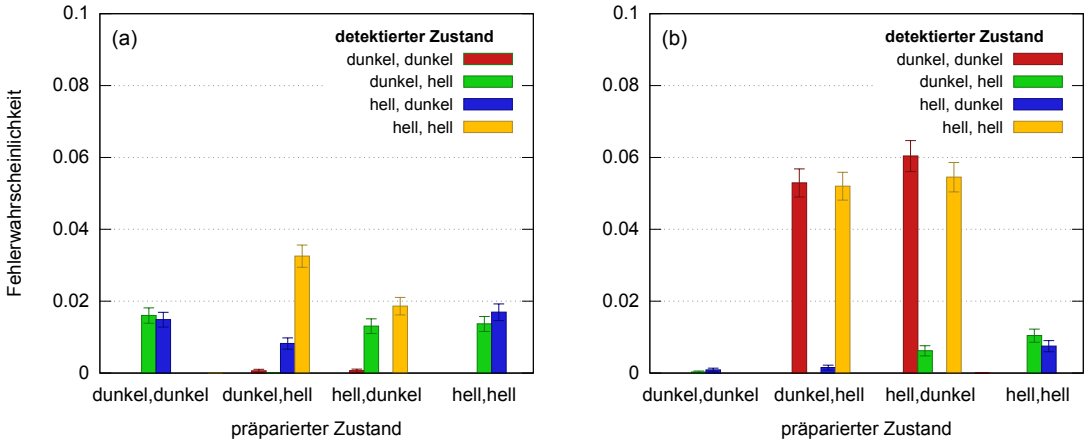


Abbildung 4.21.: Wahrscheinlichkeiten für Fehldetektionen in die verschiedenen Zustände bei einer Detektionszeit von 2 ms mit (a) dem Support-Vector-Machine-Verfahren und (b) dem Schwellwert-Verfahren.

Zuständen - die Streuung von Werten vergrößert die Ellipsen der anderen Zustände zu stark. Dem könnte durch Einführung zusätzlicher Korrekturfaktoren entgegengewirkt werden. Die Effizienz des Verfahrens kann außerdem wie beim Schwellwert-Verfahren

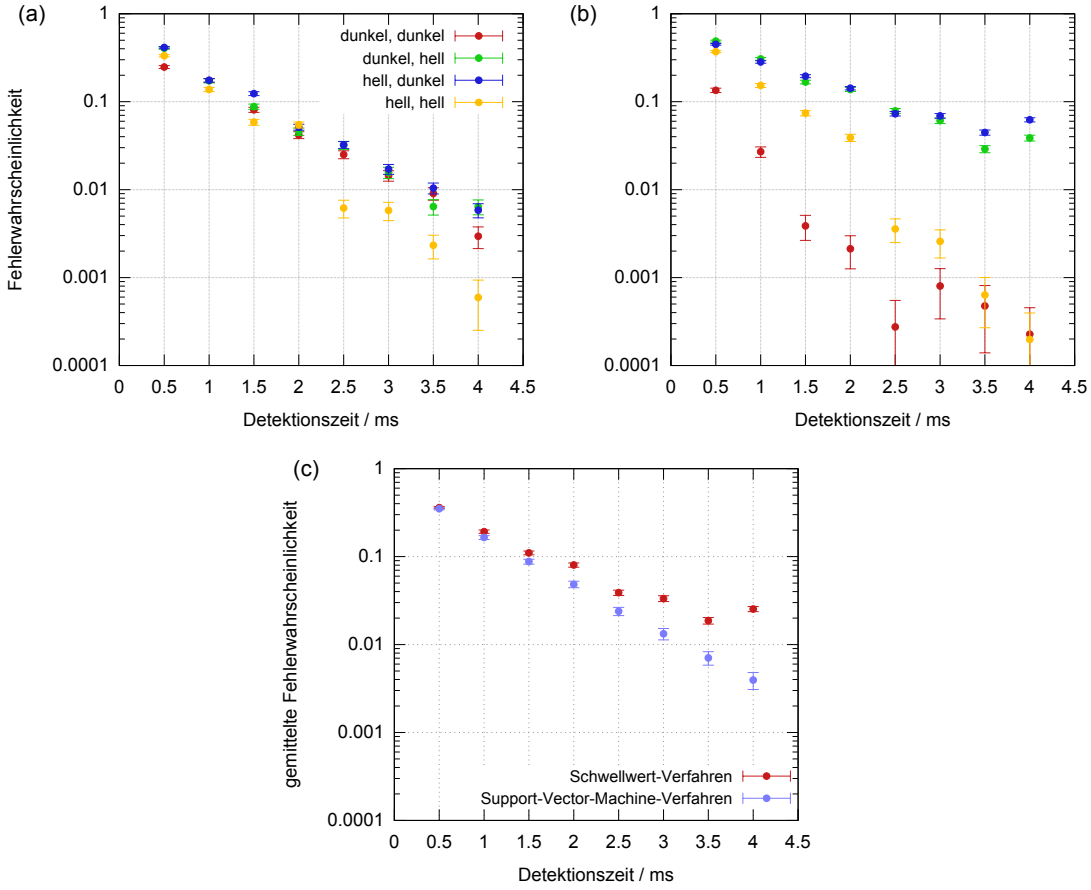


Abbildung 4.22.: Fehlerwahrscheinlichkeiten aufgeteilt auf die verschiedenen präparierten Zustände (a) des Support-Vector-Machine-Verfahrens, (b) des Schwellwert-Verfahrens und (c) die Mittelung der Fehlerwahrscheinlichkeiten für beide Verfahren in Abhängigkeit der Detektionszeit.

noch weiter gesteigert werden, indem die Form der Regions of Interest besser an die jeweiligen Ionen angepasst wird.

Aufgrund der mathematischen Einfachheit ist das Verfahren auch auf größere Ionenzahlen anwendbar. Für N Ionen müssen Trennlinien im N-dimensionalen Raum gefunden werden. Die Korrelationsmatrix ist dann eine $N \times N$-Matrix, die diagonalisiert werden muss. Die Ellipsengleichung ist ebenfalls einfach auf höhere Dimensionen verallgemeinerbar. Der Rechenaufwand zur Auswertung und auch die Messdauer, um genügend Daten zur Erstellung der Ellipsen aufzunehmen, steigt exponentiell mit der Ionenzahl an. Da aber das Prinzip des Mikrofallenexperiments darin besteht, immer nur wenige Ionen gleichzeitig abbilden zu müssen, ist die Skalierung des Verfahrens mit der Ionenzahl kein limitierender Faktor.

4.6.3. Das Maximum-Likelihood-Verfahren

Beim Maximum-Likelihood-Verfahren nach dem Vorbild von [BSWL10] wird die Form von abgebildeten Ionen berücksichtigt, indem Pixel verschieden stark gewichtet werden. Ähnlich wie beim Schwellwert-Algorithmus wird hier ebenfalls zunächst ein Histogramm mit experimentellen Daten erzeugt. Anstatt aber nur ein einziges Histogramm für eine Gruppe von Pixeln zu erstellen, wird nun für *jeden* Pixel ein Histogramm abgespeichert. Man spricht dann auch von einem zweidimensionalen Histogramm. Je nachdem, wie stark ein Pixel beleuchtet ist, überlappen sich die Verteilungen für dunkle und für helle Ionen mehr oder weniger stark (Abbildung 4.23).

Bei der Auswertung von Kamerabildern wird folgende Likelihood-Funktion berechnet

$$L_{\text{hell}} = \prod_{i=1}^{N} B_i(n_i) \qquad L_{\text{dunkel}} = \prod_{i=1}^{N} D_i(n_i) \tag{4.14}$$

mit $B_i(n_i)$ der Wahrscheinlichkeit, den Fluoreszenzwert des i-ten Pixels n_i für ein helles Ion zu beobachten und $D_i(n_i)$ der Wahrscheinlichkeit, den Fluoreszenzwert des i-ten Pixels n_i für ein dunkles Ion zu beobachten. Die Werte $B_i(n_i)$ und $D_i(n_i)$ werden aus den Daten des zweidimensionalen Histogramms bestimmt. Die Likelihood-Funktion gibt die Wahrscheinlichkeit an, genau das Bild, welches gerade ausgewertet wird, im Falle eines hellen bzw. dunklen Ions zu beobachten. Ist $L_{\text{hell}} > L_{\text{dunkel}}$, so ist es wahrscheinlicher, dass das beobachtete Ion hell ist, und umgekehrt.

Pixel, die keine Information erhalten, liefern sowohl für L_{hell} als auch für L_{dunkel} den gleichen Beitrag und beeinflussen damit nicht das Ergebnis. Nur bei beleuchteten Pixeln

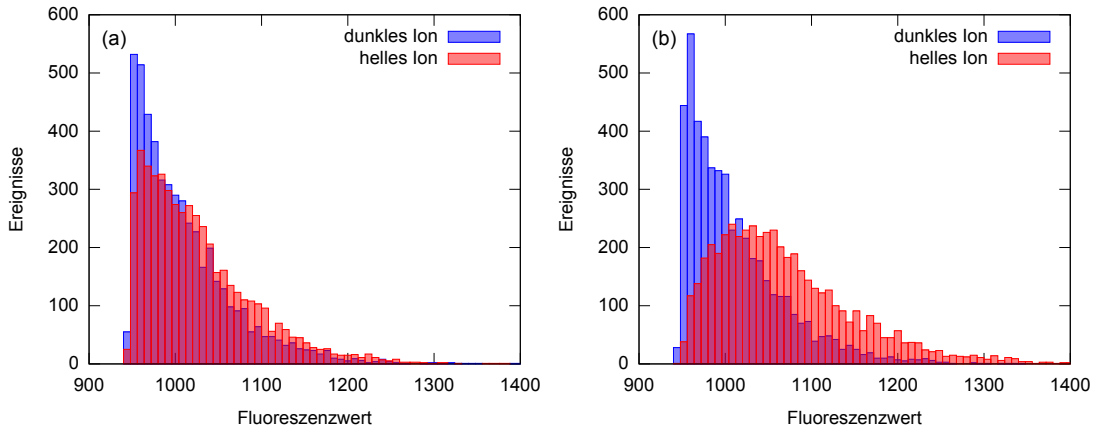


Abbildung 4.23.: Histogramme eines einzelnen Pixels (a) für einen schwach beleuchteten Pixel (b) für einen stark beleuchteten Pixel

unterscheiden sich die Beiträge - somit ergibt sich eine Gewichtung der Pixel vollautomatisch. Dadurch ist es prinzipiell sehr einfach, das Verfahren auf eine große Anzahl an Ionen anzuwenden, da keine Regions of Interest mehr per Hand gewählt werden müssen. Allerdings ist der Speicherbedarf höher als beim Schwellwert-Verfahren: Anstatt eines einzelnen Wertes pro Ion müssen Histogramme für jeden Pixel zwischengespeichert werden.

Eine technische Schwierigkeit dieses Algorithmus ist die hohe Empfindlichkeit gegenüber statistischen Fluktuationen in den Histogramm-Daten. Im Gegensatz zum Schwellwert-Verfahren und dem Support-Vector-Machine-Verfahren, wo statistische Verteilungen durch wenige Kenngrößen charakterisiert werden, werden in diesem Verfahren einzelne Werte in den Histogrammen herangezogen. Daher sind entweder eine extrem hohe Anzahl an Kamerabildern zur Erstellung eines zweidimensionalen Histogramms erforderlich (mehr als 10000 Bilder, um die Genauigkeit des Schwellwert-Verfahrens zu erreichen), oder die Daten müssen softwareseitig nachbearbeitet werden. Hierfür wurden zwei Ansätze implementiert:

1. *Binning*: Beim Binning werden mehrere Werte in den Histogrammen der einzelnen Pixel zusammengefasst. Dadurch werden die relativen statistischen Fluktuationen einzelner Werte verringert, da die Datenmenge pro Wert erhöht wird. Allerdings geht dabei auch Information verloren, sodass die Detektionseffizienz bei zu hoch eingestelltem Binning wieder abnimmt. Je mehr Kamerabilder für ein zweidimensionales Histogramm aufgenommen werden, desto weniger Binning ist erforderlich. In der Praxis hat es sich bewährt, 6 oder 8 benachbarte Werte zusammenzufassen.

2. *Filterung*: Durch einen Glättungsfilter kann das Rauschen in den Daten des zweidimensionalen Histogramms verringert werden. Es wird ein Binomialfilter verwendet, der an die Breite der Verteilungen angepasst wird: Je breiter die zu glättende Verteilung ist, desto breiter kann der Binomialfilter gewählt werden, ohne dass die Verteilungen verformt werden.

Die Kombination beider Verfahren lieferte in Testmessungen die besten Ergebnisse (Abbildung 4.24). Das zweidimensionale Histogramm bestand aus $2 \cdot 5\,000$ Kamerabildern, wobei die Bilder bei einer Detektionszeit von 1ms und bei eingeschaltetem 4×4-Kamerabinning aufgenommen wurden. Bei optimalen Einstellungen des Maximum-Likelihood-Verfahrens konnte der Detektionsfehler gegenüber dem Schwellwert-Verfahren etwa halbiert werden. Sind die Parameter nicht optimal eingestellt, ist eine Assymetrie in den Ergebnissen erkennbar. Das liegt daran, dass die Fluoreszenzwerte der hellen Zustände stärker streuen, und somit die relativen Fehler in den Histogrammen höher sind. Das kann durch Binning ausgeglichen werden. Ist dieses zu hoch eingestellt, führt es aber wiederum zu einem Informationsverlust in den dunklen Zuständen, was die Detektionswahrscheinlichkeit verschlechtert. Je mehr Kamerabilder für ein zweidimensionales Histogramm verwendet werden, desto symmetrischer werden die Ergebnisse.

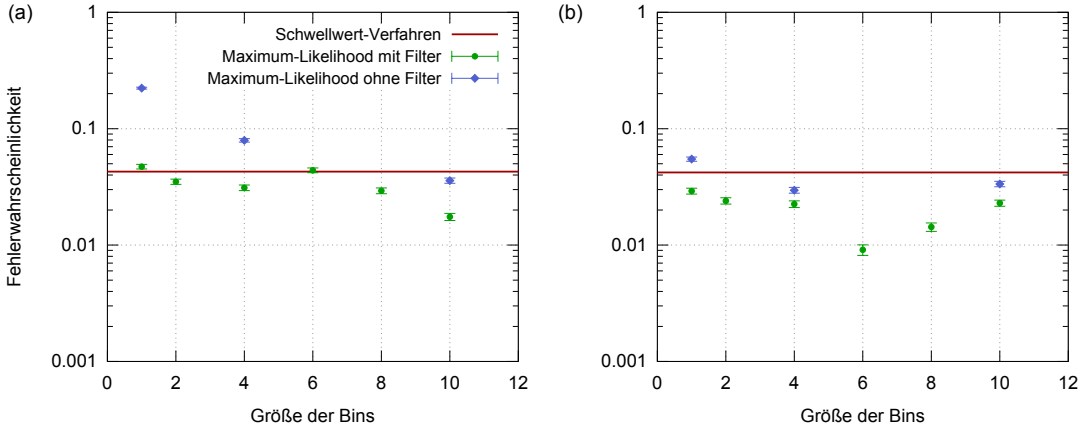


Abbildung 4.24.: Fehlerwahrscheinlichkeiten des Maximum-Likelihood-Verfahrens verglichen mit dem Schwellwert-Verfahren (a) bei präpariertem hellen Zustand und (b) bei präpariertem dunklen Zustand.

Das Maximum-Likelihood-Verfahren hat, was die Messung mehrerer Ionen angeht, die selben Probleme wie das Schwellwert-Verfahren: Pixel müssen eindeutig zu einem bestimmten Ion zugeordnet werden, was in unserem Experiment nicht ohne weiteres möglich ist. Es kann jedoch nach dem selben Prinzip, das schon beim Support-Vector-Machine-Verfahren zum Einsatz kam, verbessert werden: Beide Ionen müssen als ein gemeinsames System betrachtet werden. Die Likelihood-Funktion muss für zwei Ionen dann vier verschiedene Fälle unterscheiden: $L_{\text{dunkel,dunkel}}$, $L_{\text{dunkel,hell}}$, $L_{\text{hell,dunkel}}$ und $L_{\text{hell,hell}}$. Anstatt zwei verschiedene Histogramme pro Pixel müssen dann vier aufgenommen werden, wobei die Zustandsunterscheidung durch eine Sequenz wie im vorherigen Kapitel ermöglicht werden kann. Das Problem, dass für dieses Verfahren viel Statistik benötigt wird, ist dann allerdings noch größer: Durch die kompliziertere Detektionssequenz dauert es länger, Kamerabilder aufzunehmen, und es werden schon für zwei Ionen die doppelte Anzahl an Kamerabildern benötigt.

Aus diesem Grund wurde das Verfahren bisher für zwei Ionen noch nicht implementiert. Der Genauigkeitsgewinn des Verfahrens kann in der Praxis einfacher durch Erhöhung der Detektionszeit erreicht werden. Möglicherweise kann in Zukunft durch einen geeigneten Kurvenfit an die Daten, wie er von Burrell vorgeschlagen wurde [Bur10], die benötigte Datenmenge weiter reduziert werden, sodass dieses Verfahren für zwei Ionen in der Praxis eingesetzt werden kann. Auch dann ist aber noch nicht klar, ob es tatsächlich eine signifikante Verbesserung gegenüber dem Support-Vector-Machine-Verfahren bietet.

5

Realisierung von schnellen Ionentransporten

Im letzten Kapitel wurde Schritt für Schritt erklärt, welche Vorkehrungen zur Durchführung eines Transportexperimentes getroffen werden müssen. Daher wird hier nun immer davon ausgegangen, dass gefangene Ionen in experimentellen Sequenzen im Bewegungsgrundzustand initialisiert werden, und außerdem das Shelving und die Detektion des Zustandes verlässlich funktionieren. Alle hier präsentierten Messungen wurden der Einfachheit halber an einem einzelnen Ion durchgeführt. Modellexperimente zeigten aber, dass Transporte mit zwei gefangenen Ionen die selben qualitativen Eigenschaften besitzen wie mit einzelnen Ionen.

Bevor die eigentlichen Messungen vorgestellt werden, beginnt Abschnitt 5.1 mit der theoretischen Behandlung von *kohärenten Zuständen*. Diese Zustände werden, wie in Abschnitt 5.2 gezeigt wird, von Transportoperationen im Bewegungsfreiheitsgrad erzeugt und sind damit essentiell zum Verständnis des Themas. Da Ionen nicht nur möglichst schnell transportiert werden sollen, sondern auch im Bewegungsgrundzustand belassen werden sollen, müssen die Trajektorien der Ionen während eines Transports bestimmte Bedingungen erfüllen. In Kapitel 5.3 werden verschiedene Trajektorien vorgestellt, mit denen sich kalte Transporte verwirklichen lassen. Nicht alle Trajektorien sind für unser Experiment geeignet, daher konzentrieren sich die Messungen auf zwei verschiedene Arten, mit denen kalte Transporte gelungen sind:

1. Beim *paarweise neutralen Transport* werden gefangene Ionen von Fallensegment 5 nach Fallensegment 4 und zurück transportiert, mit identischer Form des Hin- und Rückweges. Durch eine Wartezeit zwischen den Wegen lässt sich die Schwingungsphase eines Ions so kontrollieren, dass der Rückweg die Anregung des Hinweges rückgängig macht.

2. Beim *selbstneutralen Transport* wird der Rücktransport im adiabatischen Regime durchgeführt, d.h. der Bewegungszustand von Ionen wird nicht mehr geändert. Dann muss sich ein gefangenes Ion bereits nach Abschluss des Hinweges wieder im Grundzustand befinden. Die Anregung, die durch den schnellen Transport entsteht, wird hier durch einen kontrollierten schnellen Spannungspuls rückgängig gemacht.

Damit Ionen die korrekten Trajektorien durchlaufen, müssen sich zeitlich verändernde Spannungen an die Gleichspannungselektroden angelegt werden. Diese Spannungsrampen müssen verschiedene Voraussetzungen erfüllen, diesem Thema widmet sich Abschnitt 5.4.

Um die korrekten Paramter für kalte Transporte zu finden, ist die Messung des Bewegungszustandes von essentieller Bedeutung. Wie in Kapitel 2.2 besprochen wurde, müssen dafür Rabi-Oszillationen von verschiedenen Seitenbändern aufgenommen werden. Eine solche Messung ist aber zeitraubend: Sie dauert mehrere Minuten. Um schneller optimale Parameter für kalte Transporte finden zu können, wurde daher ein neues Messverfahren entwickelt, die *Pseudo-Energie-Messung*. Der Messung des Bewegungszustand für einen bestimmten Satz an Parametern konnte so um einen Faktor von etwa 100 beschleunigt werden. Auf die Pseudo-Energie wird in Abschnitt 5.5 eingegangen.

Die eigentlich Ergebnisse der Messungen werden im darauffolgenden Abschnitt 5.6 vorgestellt. Damit schnelle Transporte in einem Quantencomputer eingesetzt werden können, muss außerdem sichergestellt sein, dass die im Zustand der Ionen kodierte Quanteninformation nach dem Transport erhalten bleibt. Das wird in Kapitel 5.7 genauer untersucht.

Zum Abschluss wird noch in Abschnitt 5.8 gezeigt, wie der Zustand von Ionen durch die Kombination von optischen und elektrischen Wechselwirkungen gezielt manipuliert werden kann. Die Erfahrungen, die durch die Transportexperimente gesammelt werden, können so zur Erzeugung *kohärent verschobener Fockzustände* genutzt werden.

5.1. Theorie der kohärenten Zustände

Da die Ausdehnung des von den Fallenelektroden erzeugten elektrostatischen Potentials um Größenordnungen höher ist als die Ausdehnung des Wellenpaketes eines Ions, kann intuitiv vermutet werden, dass sich die Bewegung des Potentials im Wesentlichen als klassische Kraft beschreiben lässt. Um diese Annahme quantenmechanisch zu untermauern, werden kohärente Zustände eingeführt. Das sind quasiklassische Zustände, deren Orts- und Impulserwartungswerte sich wie die eines klassischen harmonischen Oszillators verhalten und deshalb eine große Rolle zur Beschreibung des Zustandes während und nach Transporten spielen.

Wir definieren kohärente Zustände $|\alpha\rangle$ als das Ergebnis des *Verschiebeoperators* $\hat{D}(\alpha)$, wenn er auf den Vakuumzustand des harmonischen Oszillators angewandt wird:

$$|\alpha\rangle = \hat{D}(\alpha)\,|0\rangle \qquad \text{mit} \qquad \hat{D}(\alpha) = \exp\left(\alpha\hat{a}^\dagger - \alpha^*\hat{a}\right) \tag{5.1}$$

Dabei ist α eine beliebige komplexe Zahl. Entwickelt man kohärente Zustände nach Fock-Basiszuständen, so erhält man

$$|\alpha\rangle = \mathrm{e}^{-\frac{1}{2}|\alpha|^2} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle \tag{5.2}$$

Die Wahrscheinlichkeit, nach einer Messung die Phononenzahl n zu messen, ergibt sich aus der Poisson-Verteilung

$$p_n = |\langle n|\alpha\rangle|^2 = \frac{\langle n\rangle^n}{n!} \mathrm{e}^{-\langle n\rangle} \qquad \text{mit} \qquad \langle n\rangle = |\alpha|^2 \tag{5.3}$$

Diese Beziehung kann zur Messung der mittleren Phononenzahl $\langle n\rangle$ für kohärente Zustände in unserem Experiment verwendet werden. Dazu muss nach der Messung von Rabi-Oszillationen lediglich p_n in Gleichung 2.20 eingesetzt werden.

Real- bzw. Imaginärteil von α geben Auskunft über den Orts- bzw. Impulserwartungswert des kohärenten Zustandes:

$$\langle\hat{x}\rangle = \langle\alpha|\hat{x}|\alpha\rangle = \sqrt{\frac{2\hbar}{m\omega}}\,\mathrm{Re}(\alpha) \qquad\qquad \langle\hat{p}\rangle = \langle\alpha|\hat{p}|\alpha\rangle = \sqrt{2m\hbar\omega}\,\mathrm{Im}(\alpha) \tag{5.4}$$

Berechnet man die Ortswellenfunktion eines kohärenten Zustandes, so erhält man eine verschobene Grundzustands-Wellenfunktion des harmonischen Oszillators

$$\psi^{(\alpha)}(x) = \left(\frac{m\omega}{\pi\hbar}\right)^{1/4} \exp\left(-\frac{m\omega}{2\hbar}\left(x - \langle\hat{x}\rangle\right)^2\right) \exp\left(\frac{\mathrm{i}}{\hbar}\langle\hat{p}\rangle\, x\right) \tag{5.5}$$

Die Zeitentwicklung im Schrödingerbild ergibt sich aus der Anwendung des Zeitentwicklungsoperators

$$\hat{\mathcal{U}}(t) = \exp\left(-\frac{\mathrm{i}}{\hbar}\hat{\mathcal{H}}t\right) \tag{5.6}$$

auf den zeitunabhängigen Zustand $|\alpha\rangle$. Das Ergebnis lautet

$$|\alpha(t)\rangle = \exp\left(-\frac{\mathrm{i}}{2}\omega t\right) |\exp\left(-\mathrm{i}\omega t\right)\alpha\rangle \tag{5.7}$$

Der kohärente Zustand bleibt also ein kohärenter Zustand, wobei sich die Orts- und Impulserwartungswerte wie beim klassischen harmonischen Oszillator verhalten. Das Wellenpaket behält zu allen Zeiten seine Form bei.

5.2. Bewegungsgleichung eines transportierten Ions

In diesem Abschnitt wird nun der Ansatz skizziert, mit dem sich nach [LJ11] herleiten lässt, dass durch Transportoperationen kohärente Zustände erzeugt werden. Da Transporte entlang der Fallenachse stattfinden, genügt es, die axiale Bewegung eines Ions zu

betrachten, das sich innerhalb eines sich „zeitlich verändernden Gleichspannungsfeldes“ befindet. Damit ist gemeint, dass die Spannungen auf den Gleichspannungselektroden der segmentierten Falle so variiert werden, dass das Ion seine Position ändert. Es soll hier davon ausgegangen werden, dass das Ion zu jedem Zeitpunkt durch einen harmonischen Oszillator mit der konstanten Frequenz ω_z beschrieben werden kann. Unter diesen Voraussetzungen ist der Hamiltonoperator gegeben durch

$$\hat{\mathcal{H}}_{\text{Transport}} = \frac{\hat{p}^2}{2m} + \frac{1}{2}m\omega_z^2 \left(\hat{x} - s(t)\right)^2 \tag{5.8}$$

mit $s(t)$ einer Funktion, die jedem Zeitpunkt ein momentanes Potentialminimum zuordnet. Nehme nun an, dass sich das Potentialminimum an einem Ort s_0 befindet. Da das Ion um das Minimum herum oszilliert, erhält man den korrekten Zustand $|\chi\rangle$ im um s_0 verschobenen Bezugssystem, wenn man den Verschiebeoperator $\hat{D}$ auf den dazugehören Zustand $|\psi\rangle$ im unverschobenen Potential anwendet:

$$|\chi\rangle = \hat{D}\left(-\sqrt{\frac{m\omega_z}{2\hbar}}s_0\right)|\psi\rangle \tag{5.9}$$

Ersetzt man nun $s_0 \to s(t)$, so kann man diese Beziehung als Ansatz in die Schrödinger-Gleichung einsetzen. Man erhält

$$\mathrm{i}\hbar\frac{\partial}{\partial t}|\chi(t)\rangle = \left(\hat{\mathcal{H}}_0 + \dot{s}(t)\hat{p}\right)|\chi(t)\rangle \tag{5.10}$$

Dabei bezeichnet $\hat{\mathcal{H}}_0$ den Hamiltonoperator mit einem völlig unverschobenen Potential. Im Interaktionsbild lässt sich diese Gleichung leicht lösen [LJ11]. Startet man zum Zeitpunkt $t = 0$ im Grundzustand $|0\rangle$, so ist die Lösung $|\psi(t)\rangle$ ein kohärenter Zustand $|\alpha(t)\rangle$, wobei der Parameter $\alpha(t)$ gegeben ist durch

$$\alpha(t) = \sqrt{\frac{m\omega_z}{2\hbar}}\left(s(t) - \mathrm{e}^{-\mathrm{i}\omega_z t}\int_0^t \dot{s}(t_1)\mathrm{e}^{\mathrm{i}\omega_z t_1}\,\mathrm{d}t_1\right) \tag{5.11}$$

Daraus folgt, dass Transporte, sofern die vorausgesetzten Näherungen gültig sind, rein klassisch behandelt werden können. Abweichungen vom klassischen Verhalten können durch zwei Effekte verursacht werden:

1. *Anharmonisches Potential*: Werden Ionen zu sehr hohen Quantenzahlen n des Bewegungszustandes angeregt, so ist die Näherung eines harmonischen Potentials nicht länger gültig. Die Bewegung im anharmonischen Potential führt zur Dispersion des Wellenpaketes.

2. *Variierende Fallenfrequenz*: Ändert sich die axiale Fallenfrequenz schnell gegenüber der Schwingungsdauer der Ionen, so kann sich das Wellenpaket dieser Änderung nicht adiabatisch anpassen, was dazu führt, dass höhere Bewegungszustände bevölkert werden. Es entsteht dann ein gequetschter Zustand. Dieser Effekt kann für kleine Quantenzahlen n ebenfalls vernachlässigt werden [SPSSK06].

Die Formel 5.11 kann nun benutzt werden, um Trajektorien zu finden, die den Bewegungszustand möglichst wenig anregen. Das wird im nächsten Abschnitt behandelt.

5.3. Transport-Trajektorien

Betrachte zunächst einen Modelltransport zwischen zwei benachbarten Segmenten an den Positionen 0 und s_0, mit zwei verschiedenen Trajektorien

$$s_1(t) = s_0 \frac{t}{\tau} \tag{5.12}$$

$$s_2(t) = \sin^2\left(\frac{\pi t}{2\tau}\right) \tag{5.13}$$

mit $t \in [0, \tau]$ und τ der Gesamtdauer des Transports. Abbildung 5.1 zeigt die Anregung für die beiden Trajektorien nach Gleichung 5.11, wobei die Transportdauer τ variiert wird. Man erkennt periodisch im Abstand der Schwingungsdauer T des gefangenen Ions auftretende Minima. In der doppelt logarithmischen Darstellung äußert sich die $\frac{1}{t}$-förmige Beziehung zwischen den Maxima in der Anregung und der Transportzeit durch einen linearen Abfall. Das ist Ausdruck für den Übergang in das adiabatische Regime, in dem sich das elektrostatische Potential langsam genug bewegt, damit sich der Bewegungzustand zu allen Zeiten in den selben Energieniveaus befindet. Bei der $\sin^2$-förmigen Trajektorie $s_2(t)$ werden Ionen insgesamt weniger angeregt. Da bei dieser Trajektorie im Gegensatz zur linearen Trajektorie $s_1(t)$ die Beschleunigung nicht instantan stattfindet, entspricht dies der intuitiven Erwartung.

Um kalte Transporte experimentell zu realisieren, muss der Verlauf der Spannungen an den Fallensegmenten prinzipiell so gewählt werden, dass eine Trajektorie reproduziert wird, die einem Minimum in der Anregung entspricht. Da Spannungen im Experiment aber lediglich in Zeitschritten von mindestens 400 ns aktualisiert werden können, ist eine Variation der Transportdauer nicht in genügend kleinen Schritten möglich. Es

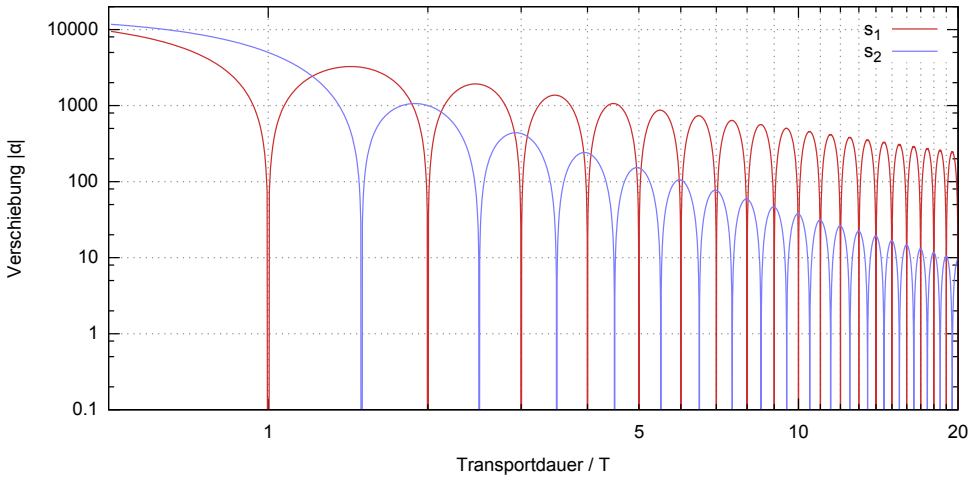


Abbildung 5.1.: Verschiebung $|\alpha|$ nach Abschluss des Transports für eine lineare (rot) und eine $\sin^2$-förmige Trajektorie (blau) in Abhängigkeit der Transportdauer. Die Transportdauer ist in Einheiten der Schwingungsdauer eines Ions angegeben.

können zwar Wartezeiten von 10 ns zwischen den Abschnitten eingeschoben werden, diese müssen jedoch in den meisten Fällen ungleichmäßig auf die Abschnitte verteilt werden, was experimentell keine befriedigenden Resultate ergab. Da alle Laser und die EMCCD-Kamera im Experiment auf ein Segment ausgerichtet sind, können außerdem nur Fälle betrachtet werden, bei denen ein Hin- und Rücktransport involviert ist.

Die Fallen-Spannungsversorgung ermöglicht es, zwei verschiedene Parameter sehr präzise zu variieren:

1. Zeitverzögerungen (über 400 ns hinaus) können in Schritten von 10 ns geändert werden
2. Spannungen können in Schritten von etwa 0,3 mV geändert werden

Mit diesen Möglichkeiten konnten zwei verschiedene Trajektorienverläufe erfolgreich reproduziert werden:

1. Beim paarweise neutralen Transport sind Hin- und Rücktransport gleich schnell, mit einer Wartezeit τ_w zwischen den Transporten. Abbildung 5.2 zeigt die Anregung nach dem Transport, sie weist periodische Minima in Abhängigkeit der Wartezeit auf.
2. Beim selbstneutralen Transport wird ein Ion nach dem Hintransport schlagartig mithilfe eines Spannungspulses ausgelenkt, wobei die Stärke der Auslenkung und die Zeitverzögerung variiert werden kann. So kann die Anregung des Hintransports schon vor dem Rücktransport rückgängig gemacht werden. Um das experimentell zu beweisen, wird der Rücktransport im adiabatischen Regime durchgeführt. Auch

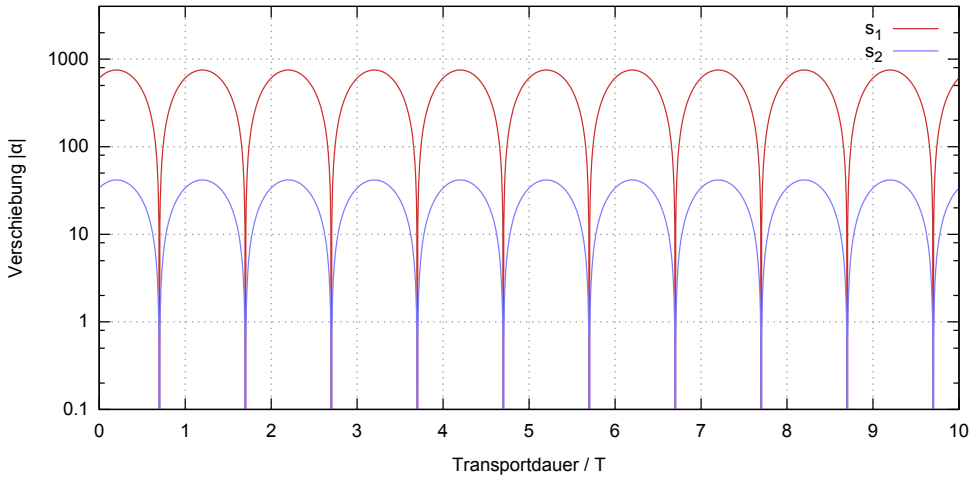


Abbildung 5.2.: Verschiebung $|\alpha|$ nach Abschluss des paarweisen Transports für eine lineare (rot) und eine $\sin^2$-förmige Trajektorie (blau) in Abhängigkeit der Wartezeit. Die Wartezeit ist in Einheiten der Schwingungsdauer eines Ions angegeben.

hier erhält man periodische Minima in der Anregung in Abhängigkeit der Zeitverzögerung, analog zu Abbildung 5.2. Der Spannungspuls kann entweder mit positiver oder negativer Spannung ausgeführt werden - die Minima sind dann um 180° phasenverschoben.

5.4. Erzeugung von Transportspannungen

Um die oben beschriebenen Trajektorien im Experiment zu verwirklichen, müssen passende Spannungsverläufe an die Fallenelektroden angelegt werden. Bei der Berechnung optimaler Spannungen wurde auf numerische Simulationen zurückgegriffen. Zunächst wurden unter Berücksichtigung der Fallengeometrie in unserem Experiment die von den Fallenelektroden generierten Potentiale simuliert [SPM+10]. Umgekehrt konnten dann bei vorgegebenen Potentialen die Spannungen berechnet werden, die an den Fallensegmenten angelegt werden müssen. Dabei wurden mehrere Randbedingungen berücksichtigt:

- Es sollen nur an den beiden benachbarten Fallensegmenten, zwischen denen die Transporte stattfinden, Spannungen angelegt werden.
- Die angelegten Spannungen können minimal $-10V$ und maximal $+10V$ betragen.
- Das Potential soll harmonisch sein, wobei die axiale Fallenfrequenz entlang des gesamten Transportes konstant bleiben soll.

Mit diesen Randbedingungen wurde eine Tabelle generiert, die jeder möglichen Ionenposition die passenden Fallenspannungen zuordnet. So kann unter Vorgabe einer Trajektorie ein Spannungsverlauf generiert werden. In allen folgenden Messungen wurde eine $\sin^2$-förmige Trajektorie verwendet. Abbildung 5.3 zeigt die auf diese Weise generierten Spannungsverläufe gegen die Zeit am Beispiel eines paarweise neutralen Transports.

Diese Spannungsverläufe können aber nicht direkt von den Digital-Analog-Wandlern der Spannungsversorgung erzeugt werden. Die Zeitauflösung der Digital-Analog-Wandler ermöglicht maximal eine Spannungsänderung alle 400 ns. Durch die Ausgangsstufe der Digital-Analog-Wandler werden die im Idealfall rechteckigen Spannungsverläufe wieder verzerrt. Die abschließende Filterung der Spannungen führt zu Assymetrien zwischen dem Hin- und Rücktransport. Außerdem schwingt die Spannung noch einige Mikrosekunden nach dem vollendeten Transport weiter.

Wie im vorherigen Abschnitt besprochen, müssen zur Realisierung eines kalten Transports die generierten Spannungsrampen gezielt manipuliert werden. Die in Abbildung 5.4 eingezeichneten Parameter wurden in unserem Experimenten zur Findung von Anregungsminima variiert. Da die Minima nur einige Nanometer breit sind, müssen die Parameter in sehr kleinen Schritten variiert und die Anregung bestimmt werden. Die effiziente Messung der Anregung wird im nächsten Kapitel behandelt.

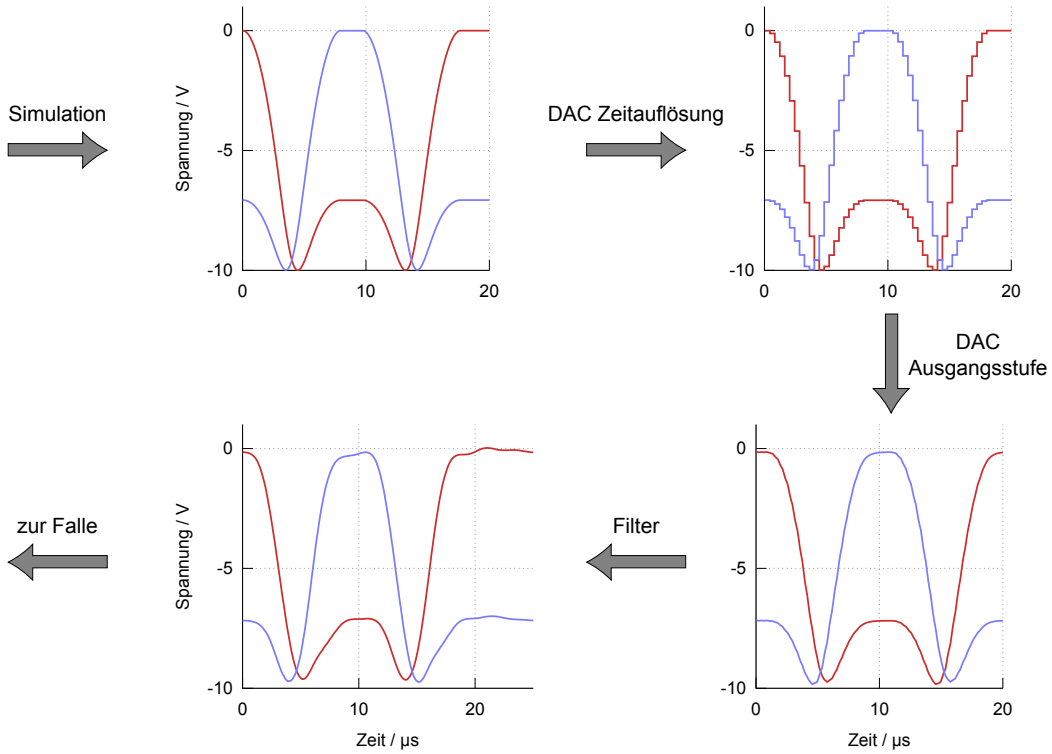


Abbildung 5.3.: Spannungsverlauf des Transports nach verschiedenen Verarbeitungsschritten. Die Spannungsverläufe beinhalten einen Transport zum Nachbarsegment (roter Spannungsverlauf), eine Wartezeit von 2μs und einen Rücktransport zum ursprünglichen Segment (blauer Spannungsverlauf).

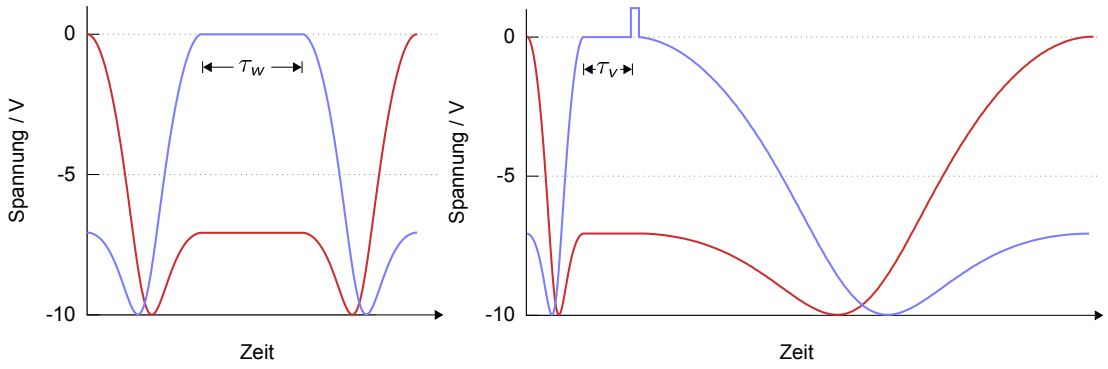


Abbildung 5.4.: Modifikation der Transportspannungen (a) beim paarweise neutralen Transport und (b) beim selbstneutralen Transport. Der adiabatische Teil des selbstneutralen Transports ist nicht maßstabsgetreu dargestellt.

5.5. Die Pseudo-Energie

Zur genauen Bestimmung des Bewegungszustandes eines Transportes müssen Rabi-Oszillationen mehrerer Seitenbänder gemessen und ausgewertet werden. Zur Auswertung kommt ein Fit-Algorithmus zum Einsatz, der den Kontrastverlust durch fehlerhafte Detektion, fehlerhaftes optisches Pumpen und Shelving sowie einen exponentiellen Kontrastabfall durch Dekohärenzeffekte berücksichtigt. Der kombinierte Fit mehrerer Seitenbänder erlaubt eine hochpräzise Bestimmung des Bewegungszustandes.

Abbildung 5.5 zeigt die Rabi-Oszillationen nach zwei unterschiedlichen Transporten: In (a) verursachte der Transport kaum Anregung des Bewegungszustandes, während in (b) die Parameter des Transportes so gewählt wurden, dass eine erhebliche Anregung stattfand. Im ersten Fall wurden der Trägerübergang, das erste rote und blaue Seitenband zur Auswertung verwendet, während im zweiten Fall das blaue Seitenband mit dem zweiten roten Seitenband ersetzt wurde. Beim Fit wird angenommen, dass die Phononenverteilung einer Faltung zwischen thermischem und kohärentem Zustand entspricht:

$$p_n = \sum_{m=n}^{\infty} \frac{\overline{n}_{\mathrm{th}}^m}{(\overline{n}_{\mathrm{th}}+1)^{m+1}} \mathrm{e}^{-\overline{n}_\alpha} \frac{\overline{n}_\alpha^{(n-m)}}{(n-m)!} \tag{5.14}$$

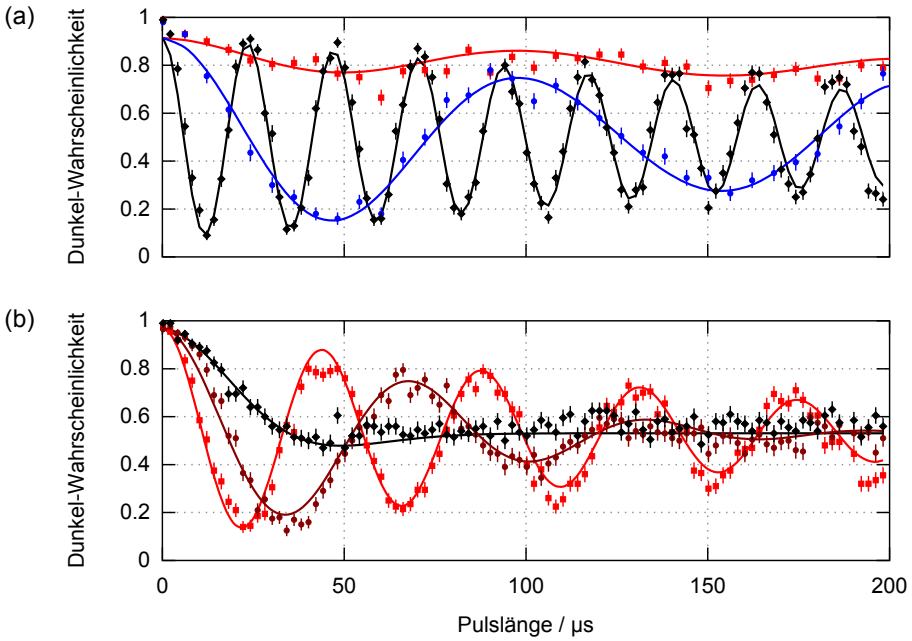


Abbildung 5.5.: Rabi-Oszillationen (a) für Träger, rotes Seitenband und blaues Seitenband im Falle von $\overline{n} = 0{,}10 \pm 0{,}01$ (b) für Träger, erstes rotes und zweites rotes Seitenband im Falle von $\overline{n} = 20{,}00 \pm 0{,}13$. Die durchgezogenen Linien zeigen einen kombinierten Fit an die jeweiligen Seitenbänder.

Dadurch wird noch berücksichtigt, dass der Bewegungszustand unmittelbar nach dem Seitenbandkühlen nicht mit einer Wahrscheinlichkeit von 100% dem Grundzustand entspricht.

Um eine grobe Übersicht über die Wirkung der Transportparameter zu erhalten, ist eine zeitaufwendige Hochpräzisionsmessung allerdings nicht geeignet. Hierfür ist man an einer schnellen Messmethode interessiert, bei der die Genauigkeit von untergeordneter Bedeutung ist. Für diesen Zweck wurde die Pseudo-Energiemessung entwickelt. Es werden zunächst einige Annahmen gemacht:

1. Der Bewegungszustand ist ein kohärenter Zustand.
2. Detektion, optisches Pumpen und Shelving sind so eingestellt, dass in den Rabi-Oszillationen ein Kontrast von etwa 95% erreicht wird.
3. Der Lamb-Dicke-Parameter beträgt $\eta \approx 0{,}24$.
4. Die Messung wird an einem einzelnen Ion durchgeführt. Für mehrere Ionen sind zusätzliche Korrekturen nötig.

Alle Annahmen wurden auf die Gegebenheiten in unserem Experiment optimiert und werden daher naturgemäß erfüllt. Die Pseudo-Energiemessung beruht auf dem Prinzip, die Rabi-Oszillationen der verschiedenen Seitenbänder durch einzelne Punkte zu approximieren. Die Pseudo-Energie ist definiert als

$$E_p = 2\left[P_{\uparrow,+1}(3\tau_{\pi,\mathrm{car}}) - P_{\uparrow,-1}(3\tau_{\pi,\mathrm{car}}) - P_{\uparrow,-2}(2\tau_{\pi,\mathrm{car}})\right] + \tfrac{7}{2} \tag{5.15}$$

mit $P_{\uparrow,\Delta n}(\tau)$ der Wahrscheinlichkeit, bei einem Seitenband mit der Pulszeit τ den Spin-Up-Zustand auszulesen. Der Index Δn bezieht sich auf die Phononenänderung des Seitenbandübergangs: $\Delta n = +1$ steht für das blaue, $\Delta n = -1$ für das rote und $\Delta n = -2$ für das zweite rote Seitenband. Mit $\tau_{\pi,\mathrm{car}}$ ist die π-Zeit des Trägerübergangs im Bewegungsgrundzustand gemeint. Um einen Wert der Pseudo-Energie im Experiment zu bestimmen, werden also lediglich drei Messpunkte benötigt.

Aus dem Wert der Pseudo-Energie lässt sich die mittlere Phononenzahl des Bewegungszustandes berechnen. Abbildung 5.6 zeigt den Zusammenhang der beiden Größen. Für Werte kleiner als 1,7 steigt die mittlere Phononenzahl monoton mit der Pseudo-Energie an. Für höhere Werte ist die Umrechnung nicht mehr eindeutig, es existieren mehrere Zweige. In Messungen, bei denen ein Transportparameter in kleinen Schritten variiert wird, kann aus dem Verlauf der Pseudo-Energie aber noch geschlossen werden, in welchem Zweig des Zusammenhangs sich die Messwerte befinden. Für Anregungen bis $\overline{n} \approx 10$ ist die Pseudo-Energie das ideale Werkzeug, um Minima oder Periodizitäten in der Anregung zu erkennen. Für den absoluten Wert der mittleren Phononenzahl ist die Pseudo-Energie lediglich eine Schätzung, da der genaue Zusammenhang von der Shelving-Effizienz und von der gewählten π-Zeit des Trägerübergangs abhängt.

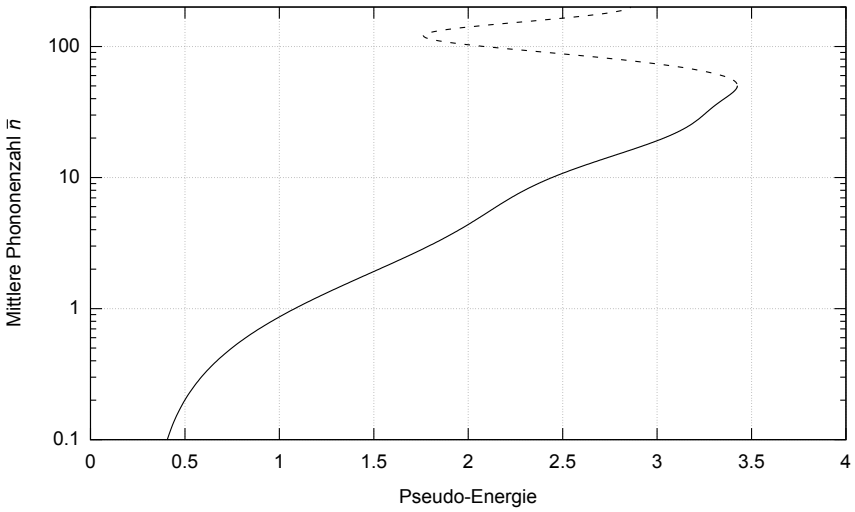


Abbildung 5.6.: Mittlere Phononenzahl als Funktion der Pseudo-Energie

5.6. Experimentelle Ergebnisse

Abbildung 5.7 zeigt die mittlere Anregung $\overline{n}$ des Bewegungszustandes beim paarweise neutralen Transport. Der Transport wurde mit einer Transportdauer von 11,2 μs in jede Richtung durchgeführt, wobei die Wartezeit zwischen den Transporten variiert wurde. Der qualitative Verlauf der Anregung stimmt mit der erwarteten theoretischen Kurve überein, die genauen Positionen der Minima oder die Zahlenwerte der maximalen Anregung konnten jedoch nicht mit theoretischen Rechnungen reproduziert werden, da die genaue Wirkung der verwendeten Elektronik und der Filter bisher nicht vollständig verstanden ist.

Um die Korrektheit der Pseudo-Energiemessung zu verifizieren, wurden zusätzlich Messwerte mithilfe des Verfahrens der Rabi-Oszillationen aufgenommen. Die kaum sichtbaren Fehlerbalken sind ein Ausdruck der hohen Präzision dieses Messverfahrens. Deutliche Abweichungen zwischen den Verfahren treten vor allem bei hohen Anregungen auf. In diesem Bereich können die Werte der Pseudo-Energie-Messung nicht mehr eindeutig den Zweigen der theoretischen Beziehung zugeordnet werden, somit ist in diesem Bereiche eine Abweichung zwischen den Verfahren akzeptabel. Die niedrigste gemessene Phononenzahl innerhalb des Minimums lautet

$$\overline{n}_{\min} = 0{,}10 \pm 0{,}01 \tag{5.16}$$

Diese Zahl ist geringer als die durch die Pseudo-Energie ermittelten Werte. In diesem Bereich reagiert die Pseudo-Energie sehr empfindlich auf Fluktuationen in der Shelving-Effizienz und in der π-Zeit des Trägerübergangs, was hier als Grund der Abweichung vermutet wird.

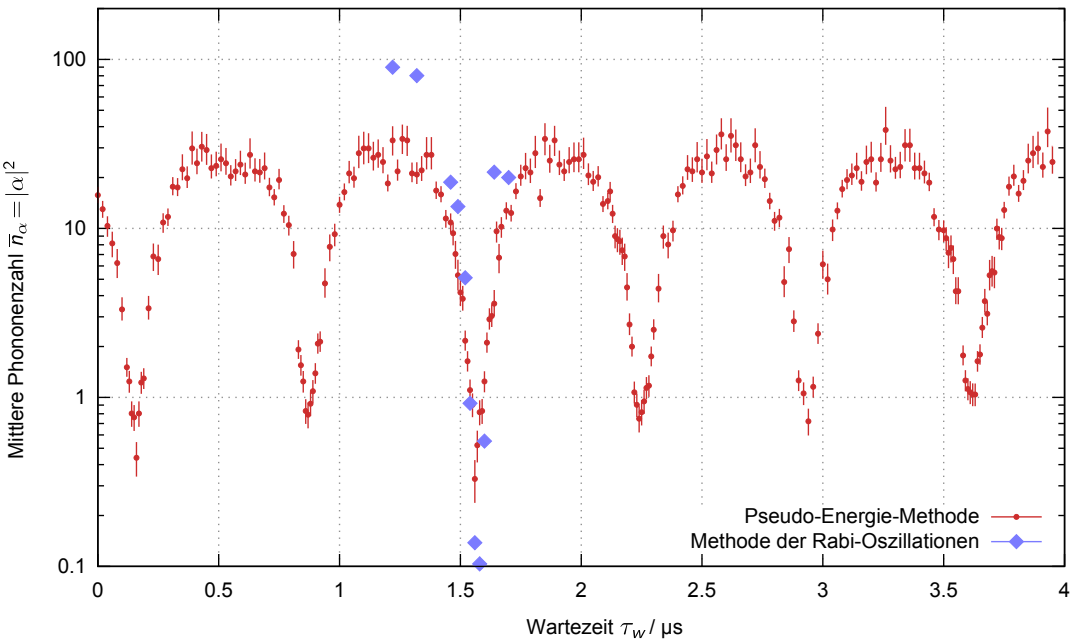


Abbildung 5.7.: Mittlere Anregung $\overline{n}$ des Bewegungszustandes beim paarweise neutralen Transport in Abhängigkeit der Wartezeit.

Beim selbstneutralen Transport müssen zwei Parameter angepasst werden: Die Verzögerungszeit τ_v des Spannungspulses und dessen Stärke. Letztere hat, bis auf das Vorzeichen, keinen Einfluss auf die Periodizität der Anregung bezüglich der Verzögerungszeit. Daher kann zunächst durch Variation der Verzögerungszeit ein Anregungsminimum gesucht werden, und bei dieser Zeit die Pulsstärke optimiert werden. Abbildung 5.8 zeigt die Ergebnisse der Pseudo-Energie-Messung bei Variation der Verzögerungszeit mit positivem und mit negativem Spannungspuls. Die Pulsstärke wurden bei dieser Messung bereits optimal kalibriert. Es zeigt sich wie erwartet ein Phasenversatz in der Periodizität um 180°. Die Transportzeiten betrugen 4 μs auf dem Hinweg und 72 μs auf dem Rückweg.

Aufgrund der niedrigen Transportzeit waren die Anregungen in den Maxima so hoch, dass sich die Messwerte über drei verschiedene Zweige der Beziehung zwischen Pseudo-Energie und mittlerer Phononenzahl erstreckten. Da ein monotoner Anstieg der Anregung mit steigendem Abstand zum Anregungsminimum erwartet wird, konnten die Pseudo-Energie-Werte den passenden Zweigen zugeordnet werden. Abbildung 5.9a zeigt die Messwerte des Pseudo-Energie-Verfahrens vor und nach der Zuordnung.

Zur Bestätigung der Korrektheit der Pseudo-Energie-Werte nach der Umrechnung wurden zusätzlich einige Messwerte mithilfe der Methode der Rabi-Oszillationen aufgenommen (Abbildung 5.9b). Hier ist anzumerken, dass sich das Anregungsminimum zwischen

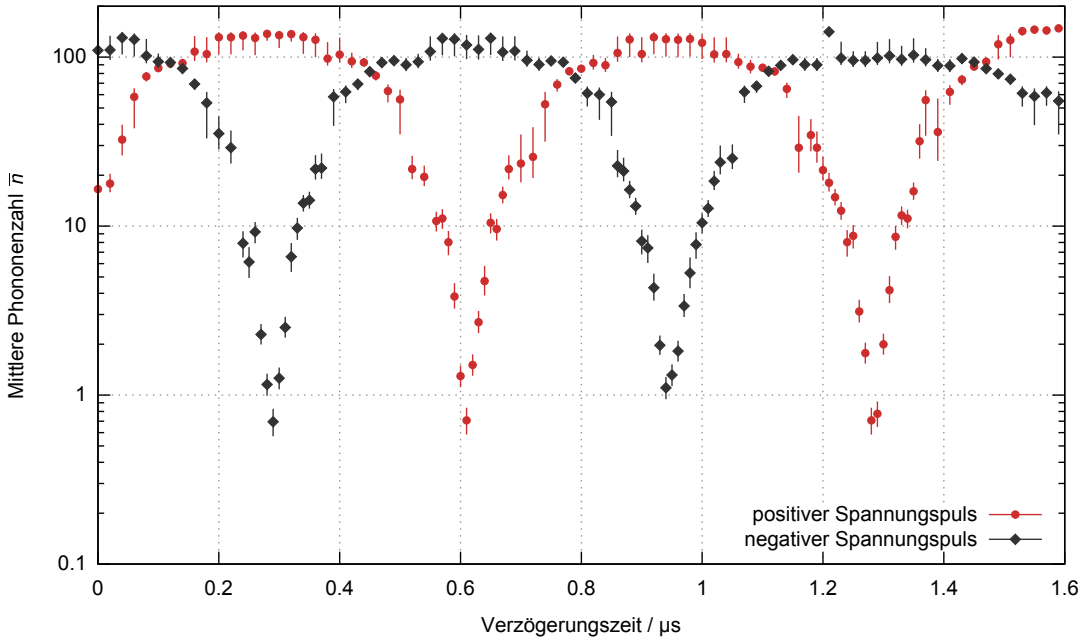


Abbildung 5.8.: Mittlere Anregung $\overline{n}$ des Bewegungszustandes beim selbstneutralen Transport in Abhängigkeit der Verzögerungszeit für negative und positive Spannungspulse.

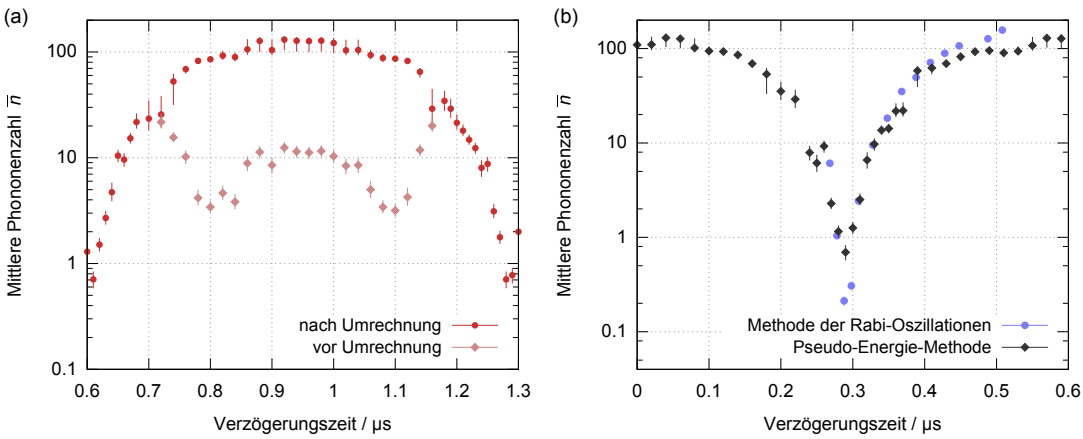


Abbildung 5.9.: (a) Messwerte mit der Pseudo-Energie-Methode vor- und nach der Umrechnung (b) Vergleich der Pseudo-Energie-Werte nach der Umrechnung mit der Methode der Rabi-Oszillationen

den beiden Messungen um etwa 110 ns verschoben hatte und somit die Messwerte der Methode der Rabi-Oszillationen nachträglich auf das Minimum der Pseudo-Energie-Werte gelegt wurden.

Der Grund für diese Verschiebung konnte bisher nicht geklärt werden. In Kapitel 4.5 wurde gezeigt, dass die Fallenfrequenz über viele Tage im Promillebereich stabil ist, daher können Drifts in der Fallenfrequenz ausgeschlossen werden. In der Vergangenheit ist es aufgrund von Softwarefehlern aber immer wieder vorgekommen, dass ungewollt Spannungen an vom Ion weit entfernten Segmenten angelegt wurden, was zu einer minimalen Positionsänderung und auch zu einer Änderung der axialen Fallenfrequenz führen kann und somit eine Erklärung für eine plötzliche Änderung der Transportparameter wäre.

Nach der Verschiebung der Messwerte ist der Verlauf der Werte beider Verfahren aber konsistent, was die Qualität der Pseudo-Energiemessung bestätigt. Die niedrigste gemessene Anregung nach dem Transport beträgt

$$\overline{n}_{\mathrm{min}} = 0{,}21 \pm 0{,}02 \tag{5.17}$$

was zu diesem Zeitpunkt einer Erhöhung um lediglich 0,01 gegenüber eines nichttransportierten Ions entsprach.

Der schnellste kalte Transport, der je in unserem Experiment gelungen ist, wurde mit einer Transportdauer von 3,6μs (zuzüglich weiterer 620 ns für den Spannungspuls) durchgeführt. Eine weitere Verringerung der Transportdauer führt zu einer schlagartigen Änderung des qualitativen Verhaltens der Anregung - die sonst beobachteten Minima verschwinden vollständig. Derart niedrige Transportdauern erreichen die Größenordnung der Grenzfrequenz des Filters der Spannungsversorgung (300 kHz), sodass die an der Falle angelegten Spannungen zu stark verzerrt werden.

5.7. Erhaltung der Quanteninformation

Nachdem nun schnelle Transporte erfolgreich durchgeführt wurden, muss noch geklärt werden, ob vorher präparierte quantenmechanische Zustände beim Transport erhalten bleiben. Experimentell äußert sich das dadurch, dass bei wiederholter Durchführung einer Transportsequenz exakt die selben Vorgänge ablaufen. Magnetfeldfluktuationen führen beispielsweise dazu, dass sich der Energieunterschied der beiden Qubit-Zustände ständig ändert und somit kohärente Operationen zwischen diesen Niveaus beim wiederholten Durchlauf einer Sequenz unterschiedliche Ergebnisse liefern - anstatt einer reinen quantenmechanischen Superposition der beiden Zustände erhält man dann ein statistisches Gemisch, welches durch eine Dichtematrix beschrieben werden muss. Auf der Bloch-Kugel werden statistische Gemische durch Bloch-Vektoren mit reduzierter Länge dargestellt.

Um zu untersuchen, ob durch Transportoperationen statistische Gemische entstehen, wurden Spin-Echo-Sequenzen durchgeführt, die ihren Namen aufgrund der Analogie zur gleichnamigen Sequenz in Kernspinresonanz-Experimenten tragen. Abbildung 5.10 zeigt das Prinzip einer Spin-Echo-Sequenz. Nach einem $\frac{\pi}{2}$-Puls wird eine Wartezeit τ abgewartet, in welcher der Bloch-Vektor bei verstimmtem Laser entlang des Äquators wandert. Die Verstimmung wird hier beispielsweise durch ein sich änderndes Magnetfeld verursacht. Nach einem π-Puls wird nochmals die selbe Zeit τ abgewartet. Hat sich die Verstimmung nicht geändert, so wandert der Bloch-Vektor mit gleicher Winkelgeschwindigkeit weiter, bis insgesamt ein Halbkreis durchlaufen ist. Aufgrund des π-Pulses ist die Endposition unabhängig von der Verstimmung, sofern diese über den Zeitraum einer Spin-Echo-Sequenz konstant ist. Mit dieser Technik werden gegenüber der Wartezeit τ langsame Fluktuationen herausgefiltert. Schnelle Fluktuationen verhalten sich in beiden Zweigen der Spin-Echo-Sequenz unterschiedlich und führen daher bei jedem Durchlauf der Sequenz zu einer anderen Endposition - dann entsteht ein statistisches Gemisch. Beim abschließenden $\frac{\pi}{2}$-Puls kann dann nicht mehr eine vollständige Population in einem der beiden Zustände erreicht werden. Im zweiten Zweig der Spin-Echo-Sequenz wird nun noch ein Ionentransport eingefügt. Da sich dieser nur innerhalb eines Zweiges befindet, führt eine mögliche durch den Transport hinzukommende Verstimmung dazu, dass sich die Position des Blochvektors nach der zweiten Wartezeit ändert. Außerdem werden durch den Transport entstehende Fluktuationen nicht durch die Spin-Echo-Sequenz herausgefiltert, sodass dann ein statistisches Gemisch entsteht. Um eine konstante Verschiebung von Fluktuationen unterscheiden zu können, wird die Phase des letzten $\frac{\pi}{2}$-Puls zwischen 0 und 2π variiert. Man erhält dann ein Sinus-Signal, dessen Kontrast Aufschluss über die Reinheit des Zustandes liefert. Eine konstante Drehung des Bloch-Vektors führt hingegen zu einer Phasenverschiebung des Sinus-Signals.

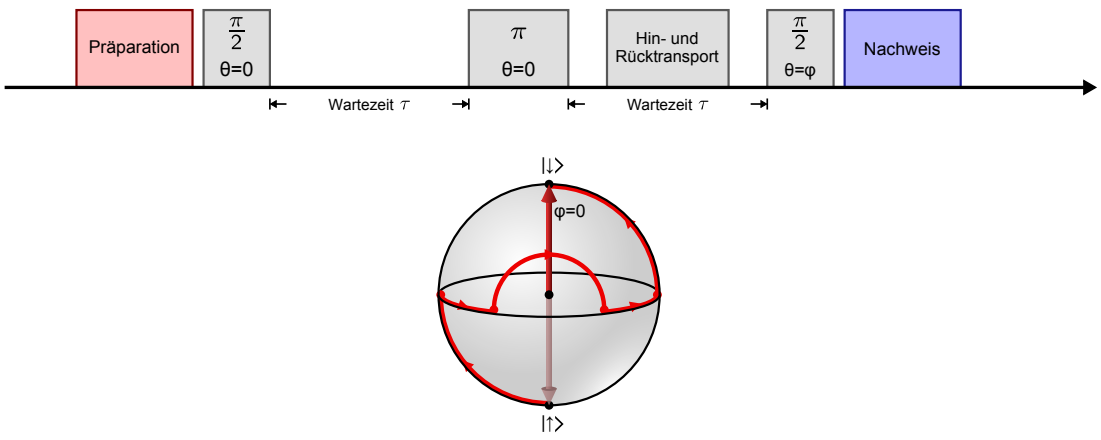


Abbildung 5.10.: Ablauf der Spinecho-Sequenz sowie die Veranschaulichung auf der Blochkugel

Abbildung 5.11a zeigt Ergebnisse von Spin-Echo-Sequenzen mit und ohne Transport, wobei mit den Laserpulsen der Trägerübergang zwischen beiden Spin-Zuständen des Ions getrieben wurde. Man erkennt, dass in beiden Fällen sowohl Kontrast als auch Phasenverschiebung des Signals nahezu identisch sind. Eine Phasenverschiebung würde entstehen, wenn das von außen angelegte Magnetfeld entlang des Transportweges inhomogen wäre. Solche Inhomogenitäten wurde aber vorher unter Benutzung genau dieser Messung bereits kompensiert [WPZ+11]. Da kein Kontrastverlust stattfindet, zeigt sich, dass die im Spinzustand kodierte Quanteninformation erhalten bleibt.

Quanteninformation kann aber nicht nur im Spinzustand, sondern auch im Bewegungszustand eines Ions gespeichert werden. Um zu überprüfen, ob diese Information ebenfalls durch Transporte erhalten bleiben, wurden die Laserpulse der Spin-Echo-Sequenz in einer weiteren Messung nicht auf dem Trägerübergang, sondern auf dem blauen Seitenband durchgeführt. Der erste $\frac{\pi}{2}$-Puls der Sequenz versetzt das Ion in den Zustand $|\uparrow, n=0\rangle + |\downarrow, n=1\rangle$, bei dem der Spin mit dem Bewegungszustand verschränkt ist. Die Phase, die während der Spin-Echo-Sequenz aufgesammelt wird, setzt sich nun zusammen aus der aufgesammelten Phase des Spin-Zustandes und der Phase des Bewegungszustandes. Die Energieniveaus des letzteren hängen von der axialen Fallenfrequenz ab, sodass sich Änderungen in dieser auf die Phase des Gesamtzustandes auswirken. Abbildung 5.11b zeigt die Ergebnisse dieser Messung mit und ohne durchgeführten Transport. Der Kontrast beider Kurven ist niedriger als im vorherigen Fall, das wird auf Rauschen der an der Falle angelegten Spannungen zurückgeführt. Allerdings ergeben sich im Kontrast beider Kurven keine signifikanten Unterschiede, sodass gezeigt ist, dass Transportoperationen Quanteninformation sowohl im Spin- als auch im Bewegungszustand erhält.

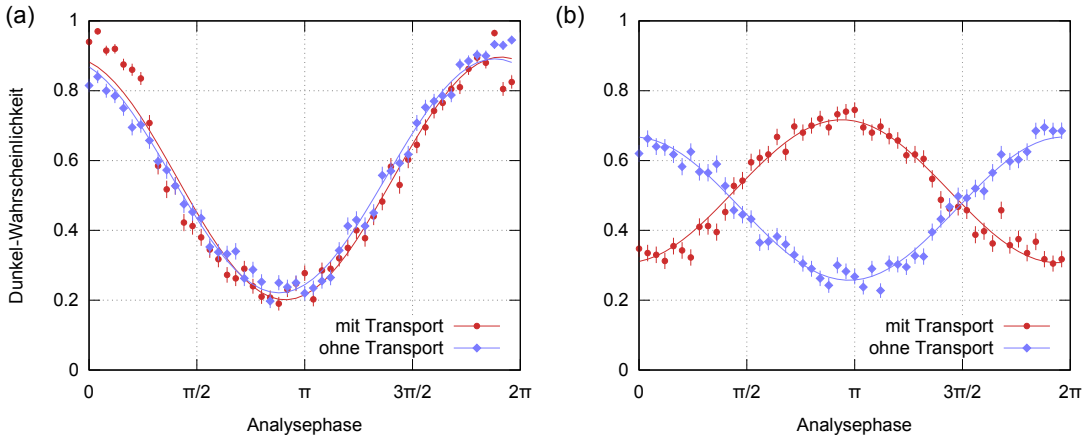


Abbildung 5.11.: Ergebnisse der Spin-Echo-Sequenzen mit und ohne Transport für (a) den Trägerübergang und (b) das blaue Seitenband. Die durchgezogenen Linien beschreiben Fits an die Messpunkte.

Die Messung mit durchgeführtem Transport beinhaltet eine Phasenverschiebung gegenüber der Messung ohne Transport, was zeigt, dass sich die axiale Fallenfrequenz auf reproduzierbare Weise während des Transportes ändert. Die Spin-Echo-Sequenz auf dem blauen Seitenband eröffnet nun aber die Möglichkeit, die Fallenfrequenz durch Korrekturen der Fallenspannungen während des Transports auf ein konstantes Niveau zu eichen. Das Schema zur Durchführung dieser Messung lautet

- Transportiere in der Spin-Echo-Sequenz nicht die volle Distanz zum Nachbarsegment, sondern zunächst nur einen Teil davon, sodass das Potentialminimum am Ort z_w zum stehen kommt.

- Warte nach dem Hintransport einen Zeitraum τ_w ab, bevor das Ion zurücktransportiert wird.

- Ist die Fallenfrequenz ω_{z_w} am transportieren Ort z_w von der Frequenz am Fangsegment ω_0 verschieden, so wird während der Wartezeit eine Phase $\varphi = \tau_w \left(\omega_o - \omega_{z_w}\right)$ aufgesammelt. Korrigiere nun die zum Ort z_w gehörigen Fallenspannungen so, dass die Phasenverschiebung verschwindet.

- Um die Genauigkeit der Kompensation zu erhöhen, kann in weiteren Messungen die Wartezeit τ_w erhöht werden, in unseren Messungen betrug sie maximal 100 μs. Je höher die Wartezeit, desto empfindlicher reagiert die Phasenverschiebung auf Abweichungen in der Fallenfrequenz.

- Nun sind die Spannungen an einem einzelnen Ort zwischen beiden Segmenten kompensiert. Die Kompensation muss nun schrittweise für höhere Transportdistanzen durchgeführt werden, bis die volle Distanz zum Nachbarsegment erreicht ist.

Abbildung 5.12 zeigt die Ergebnisse der Spin-Echo-Sequenzen für den Fall ohne und mit Korrektur der Fallenspannungen. Für unkorrigierte Spannungen befindet sich die größte Abweichung genau in der Mitte beider Segmente - sie beträgt etwa 7%. Durch geeignete Wahl der Korrekturfaktoren konnte die Änderung der Fallenfrequenz während des Transports um zwei Größenordnungen reduziert werden.

Nach der Einführung der Korrekturfaktoren wurden die Experimente zur Realisierung von kalten Transporten wiederholt. Es stellte sich heraus, dass sich für die Anregung der Ionen nach einem Transport keine signifikante Änderung ergab. Die mittleren Phononenzahlen in den Minima der paarweise neutralen Transport bzw. der selbstneutralen Transporte sind bereits in der Größenordnung der Phononenzahlen unmittelbar nach dem Seitenbandkühlen und bieten daher kaum Spielraum für Verbesserungen. Auch der qualitative Verlauf der Anregung änderte sich nicht.

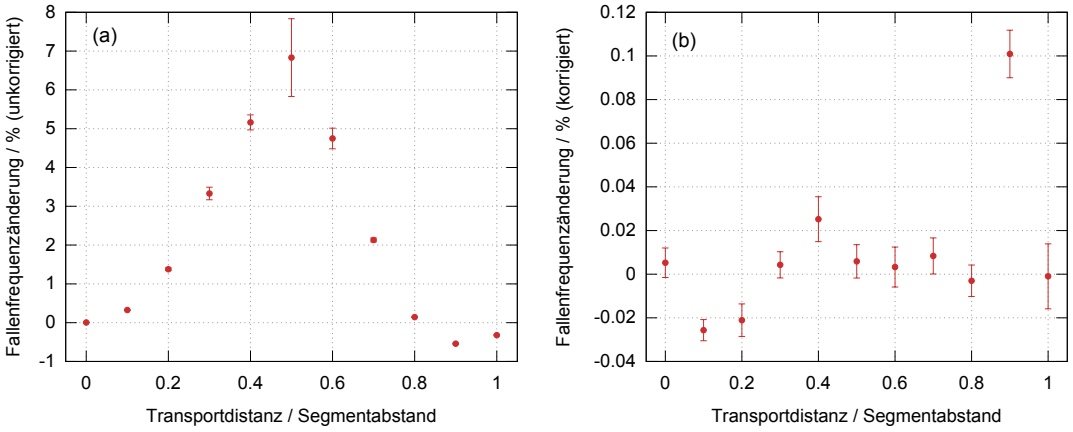


Abbildung 5.12.: Relative Abweichung der Fallenfrequenz in % als Funktion der Transportdistanz (a) ohne Korrekturfaktoren und (b) mit Korrekturfaktoren

5.8. Präparation kohärent verschobener Fockzustände

Die positiven Erfahrungen mit kurzen Spannungspulsen bei der Realisierung selbstneutraler Transporte können als Abschluss dieses Kapitels noch dazu genutzt werden, um eine besondere Klasse von quantenmechanischen Zuständen herzustellen: die kohärent verschobenen Fockzustände $|\alpha, n\rangle$. Das sind nichtklassische Zustände, die sich aus der Anwendung des Verschiebeoperators $\hat{D}(\alpha)$ auf Fockzustände $|n\rangle$, insbesondere mit $n > 0$, ergeben.

Bei der Präparation wird eine Kombination aus optischen Manipulationen des Bewegungszustandes durch Laser und elektrischen Manipulationen durch Spannungspulse an Nachbarsegmenten eingesetzt - eine Demonstration der exzellenten Kontrolle über den Zustand einzelner Ionen in unserem Experiment. Abbildung 5.13 zeigt das verwendete Schema zur Präparation des Zustandes. Ein Ion wird zunächst im Grundzustand initialisiert und dann durch abwechselnde Laserpulse auf das blaue Seitenband und den Trägerübergang in den nächsthöheren Fockzustand präpariert, wobei im Folgenden Zustände bis maximal $n = 2$ untersucht werden. Da nach dem Seitenbandkühlen außer dem Grundzustand auch höhere Zustände mit kleinen Wahrscheinlichkeiten besetzt sind, entstehen Fehler, die sich entlang der Pulssequenz fortpflanzen. Die Rabifrequenzen für die verschiedenen Bewegungszustände sind unterschiedlich, was einerseits eine Korrektur der π-Zeiten der Laserpulse für höhere Bewegungszustände notwendig macht, andererseits aber auch die Fidelität der einzelnen Laserpulse aufgrund der Besetzung mehrerer Zustände beeinträchtigt. Durch optisches Pumpen wird aber sichergestellt, dass sich das Ion nach den Trägerübergangen auf jeden Fall wieder im Spin-Up-Zustand befindet. Das erleichtert die spätere Auswertung des Bewegungszustandes.

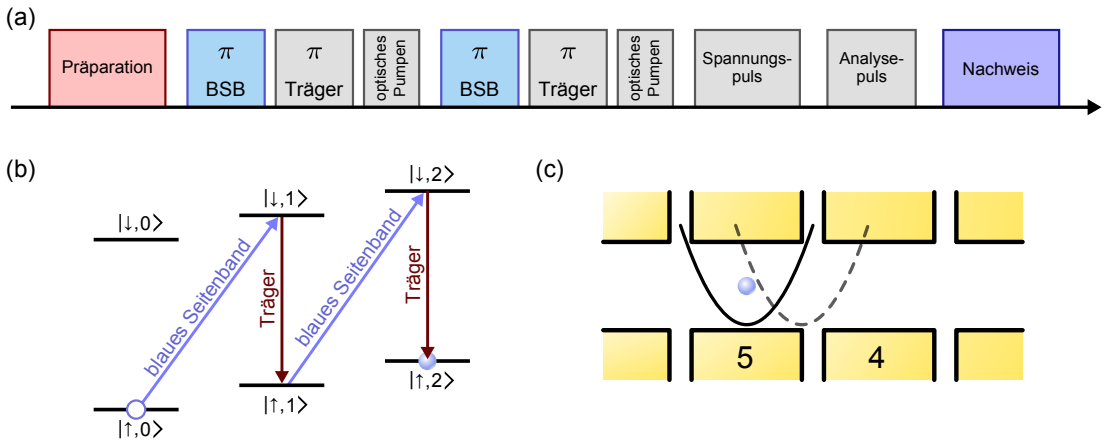


Abbildung 5.13.: (a) Die volle experimentelle Pulssequenz zur Herstellung eines kohärent verschobenen Zustandes (b) Wirkung der Laserpulse am Energieniveauschema eines Ions (c) Wirkung des Spannungspulses auf das elektrische Potential: Die Verschiebung des Potentials führt zur Auslenkung des Ions aus der Ruhelage

Nachdem ein Fockzustand präpariert wurde, wird dieser mithilfe eines Spannungspulses im Phasenraum verschoben, sodass ein kohärent verschobener Fockzustand entsteht. Die Stärke des Spannungspulses bestimmt dabei die Amplitude der Auslenkung $|\alpha|$ des Ions. Es wurden Messreihen aufgenommen für Spannungspulse in Schritten von 200 mV bis zu einer maximalen Spannung von 2 V, was einer maximalen Auslenkung von $|\alpha| \approx 2{,}8$ entspricht.

Zur Auswertung des Bewegungszustandes wird die Methode der Rabi-Oszillationen verwendet, wie sie schon im vorhergehenden Abschnitt zum Einsatz kam. Dazu wird nach der Spannungspuls ein Analysepuls durchgeführt, wobei die Sequenz für Analysepulse auf verschiedenen Seitenbändern wiederholt werden muss. Bei der Rekonstruktion des Bewegungszustandes durch einen Kurvenfit wurde keine spezifische Annahme für den vorliegenden Zustand getroffen, sondern alle Phononenzahlen unabhängig voneinander bestimmt. Abbildung 5.14 zeigt die durch Fits bestimmte Phononen-Verteilungen für die präparierten Fock-Zustände ohne zusätzlichen Spannungspuls und die Fidelitäten F_n der experimentellen Zustände, d.h. den Übereinstimmungsgrad des experimentellen mit dem theoretisch erwünschten Zustand.

Die theoretisch erwartete Phononenverteilung für einen präparierten Zustand $|\alpha, n\rangle$ lautet [BL01, OKKB90]

$$p_k^{(n)}(\alpha) = |\langle k|\alpha, n\rangle|^2 = e^{-|\alpha|^2} |\alpha|^{2(k-n)}\, n!\, k! \times \left| \sum_{l=0}^{n} \frac{(-1)^l |\alpha|^{2(n-l)}}{l!(n-l)!(k-l)!} \right|^2 \tag{5.18}$$

Zum Vergleich der experimentellen Werte mit der theoretischen Erwartung müssen die Fehler in der Präparation der Fock-Zustände berücksichtigt werden. Das geschieht durch

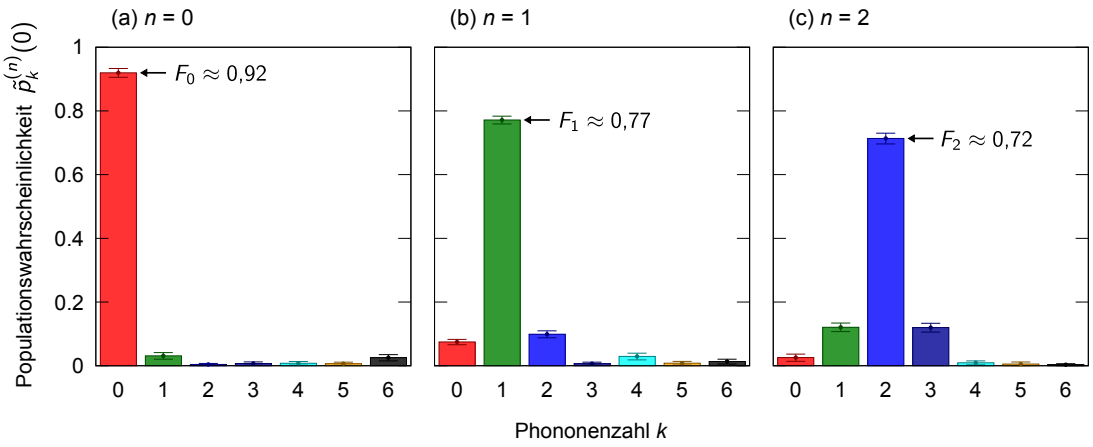


Abbildung 5.14.: Experimentelle Phononen-Verteilungen $\tilde{p}_k^{(n)}$ ohne Spannungspuls für die gewünschten Zustände (a) $n = 0$, (b) $n = 1$ und (c) $n = 2$

Gewichtung des theoretischen Zusammenhangs 5.18 mit den experimentell bestimmten Populationen $\tilde{p}_k^{(n)}(0)$ in den jeweiligen Fock-Zuständen:

$$\tilde{p}_k^{(n)}(\alpha) = \sum_{m=0}^{n_{\max}} \tilde{p}_k^{(n)}(0) \cdot p_k^{(n)}(\alpha) \tag{5.19}$$

Den anhand der experimentellen Daten bestimmten Phononenverteilungen für Messungen mit Spannungspuls können dann durch einen Kurvenfit an Gleichung 5.19 die zugehörigen Verschiebungen $|\alpha|$ zugeordnet werden. Abbildung 5.15 zeigt, dass die Zuordnungen zwischen Pulsspannung und Verschiebung $|\alpha|$ zwischen den verschiedenen Messreihen miteinander konsistent sind. Außerdem wurde die Korrektheit des Auswerteverfahren zusätzlich mit Simulationsergebnissen untermauert. Dafür wurden Spannungsverläufe für alle verwendeten Spannungspuls-Amplituden mit einem Oszilloskop nach der Filterung der Spannungen aufgenommen Mithilfe der simulierten elektrostatischen Potentiale unserer Mikrofalle konnte die klassische Bewegungsgleichung eines Ions gelöst und so die Anregungen $|\alpha|$ bestimmt werden.

Abbildung 5.16 zeigt die experimentellen Ergebnisse für verschiedene Verschiebungen $|\alpha|$ sowie den Vergleich zu den theoretischen Vorhersagen $\tilde{p}_k^{(n)}(\alpha)$. Man erkennt eine exzellente Übereinstimmung zwischen den Messdaten und den Vorhersagen. Eine besondere Eigenschaft der kohärent verschobenen Fock-Zustände ist das Auftreten von Minima bei der Besetzung von Phononenzahlen in Abhängigkeit von $|\alpha|$. Für $n = 1$ tritt ein Minimum auf, für $n = 2$ konnten zwei Minima beobachtet werden. Die Minima können als Interferenzeffekte im Phasenraum verstanden werden [OKKB90].

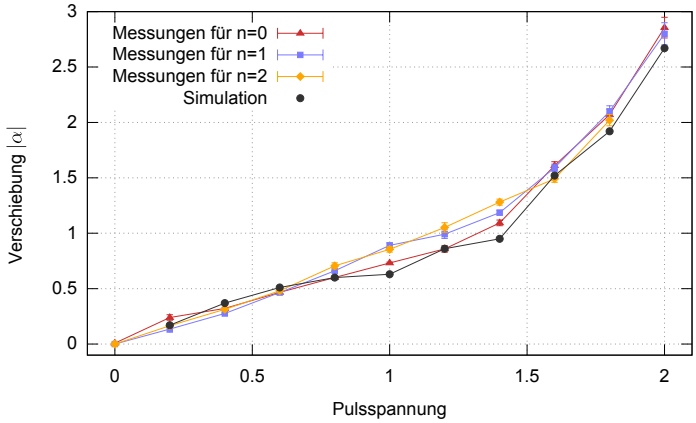


Abbildung 5.15.: Aus den Messungen extrahierte Beziehung zwischen der Verschiebung $|\alpha|$ und der Pulsspannung für $n = 0, 1, 2$ verglichen mit den aus der Simulation erhaltenen Daten. Die Punkte wurden zur besseren Übersicht mit Linien verbunden.

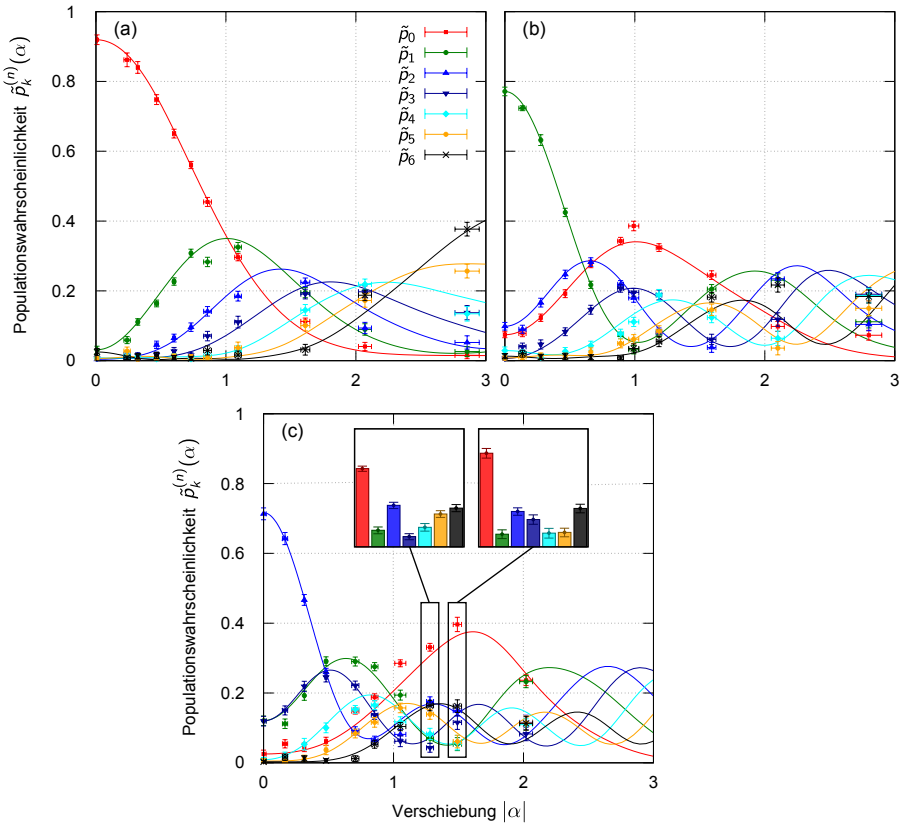


Abbildung 5.16.: Experimentelle Phononen-Verteilungen $\tilde{p}_k^{(n)}$ in den Fällen (a) n=0, (b) n=1 und (c) n=2 für verschiedene Auslenkungen $|\alpha|$, die aus dem Fit der Messdaten an den theoretisch erwarteten Zusammenhängen extrahiert wurden. Diese sind als durchgezogene Linien ebenfalls dargestellt.

6 Ausblick

Mit schnellen Transporten grundzustandsgekühlter Ionen ist ein wichtiger Schritt in Richtung skalierbarer Quantencomputer gelungen. Es wurde experimentell gezeigt, dass die nötigen schnellen Transportoperationen zwischen zwei benachbarten Fallensegmenten möglich sind. Bei der Erweiterung auf eine höhere Anzahl an Fallensegmenten sind keine größeren Schwierigkeiten zu erwarten. Da Ionen sehr empfindlich auf elektrische Felder reagieren, ist allerdings davon auszugehen, dass sich schon winzige Fehler bei der Fertigung von Mikrofallen auf das Verhalten transportierter Ionen auswirken. Aus diesem Grund müssen Transportparameter sehr wahrscheinlich für jedes Segment verschieden eingestellt werden. Automatisierte Messungen können den Zeitaufwand hierfür aber überschaubar halten.

Zunächst müssen aber weitere Bausteine für einen skalierbaren Quantencomputer realisiert werden. Dazu zählen unter anderem die noch verbleibenden Transportoperationen, und zwar das schnelle Trennen und Zusammenführen zweier Ionen. Diese konnten in bisherigen Messungen in unserer Mikrofalle nur mit einer erheblichen Anregung des Bewegungszustandes der Ionen realisiert werden. Beim Trennen zweier Ionen muss die Potentialtiefe des Fangsegments reduziert werden, während gleichzeitig Spannung auf den Nachbarsegmenten angelegt wird. Während des Übergangs kann die axiale Frequenz nicht konstant gehalten werden, was momentan als Hauptursache des Aufheizens vermutet wird. Bowler et al. [BGL+12] konnten schnelles Auftrennen und Zusammenführen bereits erfolgreich durchführen, bei einer Anregung von $\overline{n} \approx 1{,}5$. Im dortigen Experiment wurde eine auf Trennoperationen optimierte Mikrofalle verwendet, bei der zwischen den eigentlichen Fallenelektroden noch eine „Keilelektrode“ mit geringerer Breite vorhanden ist. Diese ermöglicht es, bessere auf das Trennen optimierte Potentialverläufe zu erzeugen. Sollten alle weiteren Versuche scheitern, schnelle Trennoperationen in unserer Mikrofalle zu verwirklichen, muss hierfür in Zukunft eine neue, ähnlich konzipierte Falle entwickelt werden.

Ein weiterer Baustein für einen Quantencomputer sind Quantengatter mit zwei Ionen. In momentan laufenden Anstrengungen konnte die Fidelität des geometrischen Phasengatters [Ott12] von in letztem Jahr 64% bereits auf über 90% gesteigert werden. Das genügt, um erste Modellexperimente für kompliziertere Quantenalgorithmen durchzuführen, in denen Gatter- und Transportoperationen kombiniert werden sollen. In diese

Kategorie fallen beispielsweise die Quantenteleportation und der Verschränkungsaustausch (Entanglement-Swapping) mit räumlich getrennten Ionen.

Bei der Realisierung komplexerer Algorithmen wird den in dieser Arbeit entwickelten automatischen Kalibrationssequenzen eine hohe Bedeutung zukommen. In Zukunft werden zu den Sequenzen zur Einstellung von Laserparametern noch neue Sequenzen hinzukommen, mit denen Parameter des geometrischen Phasengatters und Transportparameter automatisch bestimmt werden. Für letztere wurde das ideale Werkzeug bereits geschaffen: Die Pseudo-Energie. Diese ist ein ideales Beispiel dafür, wie durch kreative Messmethoden die Bestimmung von Parametern um Größenordnungen beschleunigt werden kann.

A

Software-Dokumentation

A.1. Das Master-Control-Programm

Das Master-Control-Programm (MCP) ist ein auf C++ basiertes allgemeines Experimentsteuerungsprogramm, das zu großen Teilen von Kilian Singer entwickelt wurde. Es wird zur Steuerung jedes Experiments der AG Schmidt-Kaler eingesetzt. Die Funktionalität des Programms lässt sich in drei Kategorien einteilen:

- **Bereitstellung eines grafischen User-Interfaces (GUI)**
  Das Master-Control-Programm benutzt die quelloffene Qt-Bibliothek, um ein flexibles grafisches Interface zu erzeugen. Der Benutzer des Programms kann sehr schnell und einfach verschiedene Elemente, wie z.B. Tabs, Eingabefelder und Schaltflächen nach seinem Belieben erzeugen.
- **Hardware-Interaktion**
  Im Master-Control-Programm sind Klassen zur Ansteuerung verschiedener Hardwarekomponenten enthalten, die in den Experimenten der AG Schmidt-Kaler eingesetzt werden, z.B. EMCCD-Kameras oder NI-PCI-Karten.
- **Laden einer .Dll-Datei**
  Das Master-Control-Programm ist in der Lage, beliebige .Dll-Dateien zu laden, um seine Funktionalität zu erweitern und an ein spezifisches Experiment anzupassen. Es erkennt automatisch Variablen, Objekte und Funktionen, die in einer .Dll-Datei festgelegt werden. Diese lassen sich vom Benutzer aus einer Liste auswählen und, je nach Typ der Variable bzw. Typ der Rückgabe- und Eingabevariablen von Funktionen, automatisch als passendes grafisches Element in das User-Interface einbinden. So lassen sich grafische Anwendungen in C++ erstellen, ohne GUI-Programmierung beherrschen zu müssen. Dll-Dateien, die zu den jeweiligen Experimenten gehören, werden traditionsgemäß *Script.Dll* genannt.

Abbildung A.1 veranschaulicht die Prozedur zur Erstellung von grafischen Elementen. Im Quelltext wird eine neue Klasse `pump_s` erstellt, in der verschiedene Variablen mit dem `prop()`-Makro, welches vom Master-Control-Programm zur Verfügung gestellt wird, deklariert werden. Der Ausdruck `EXPORT` in Zeile 1 ist nötig, damit die Klasse für das Master-Control-Programm sichtbar wird. Nach der Klassendeklaration

wird in Zeile 13 ein Objekt der Klasse definiert. Die Variablen dieses Objekts können im User-Interface dann durch Rechtsklick → New Control hinzugefügt werden. Für Bool-Variablen erscheinen Kontrollkästchenen, Integer- und Double-Variablen können in Eingabefeldern geändert werden.

```
1  class EXPORT pump_s {
2  public:
3    prop(bool,pump729);
4    prop(bool,pumpToPlus);
5    prop(bool,useFull397);
6    prop(int,pump729plusCycles);
7    prop(int,pump729minusCycles);
8    prop(double,pumpToPlusMotorAngle
         );
9    prop(double,
         pumpToMinusMotorAngle);
10   prop(bool,adjustMotor);
11 };
12
13 EXPORT pump_s Gpump;
```

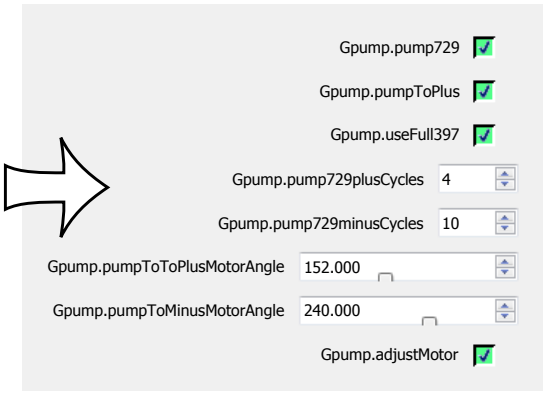


Abbildung A.1.: Erstellung von GUI-Elementen durch Script.Dll-Code. Die im Quelltext (links) deklarierten Variablen können per Rechtsklick in das User-Interface des Master-Control-Programms (rechts) platziert werden.

A.2. Dokumentation der Kamera-Software

Am 22. Juni 2012 wurde eine neue Kamera-Software in das Master-Control-Programm integriert, welche gegenüber der alten Software einen deutlich erhöhten Funktionsumfang bietet. Die neue Software wurde mit dem Ziel entwickelt, möglichst viel Funktionalität, die zuvor in den Script.Dll-Dateien einzelner Experimente verstreut war, zu vereinen und außerdem selbst leicht erweiterbar zu sein, ohne Quelltext des Master-Control-Programms bearbeiten zu müssen. Folgende Klassen bilden die Basis des neuen Systems:

- **Camera** (`AndorSDK.h`)
  Diese Klasse ist die direkte Schnittstelle zur Hardware. Funktionsaufrufe werden direkt an den Andor-Treiber der Kamera weitergeleitet. Im Normalfall sollte es jedoch nicht nötig sein, Methoden aus dieser Klasse aufzurufen, da die `CameraHelper` -Klasse für einfache Benutzung ausgelegte Methoden bietet. Möchte man dennoch direkt auf Methoden der `Camera`-Klasse zugreifen, muss man beachten, dass diese bei Fehlern `CameraExceptions` erzeugen, welche unbedingt abgefangen werden müssen.

- **Data2DPlotArray** (`Data2DPlotArray.h`)
  Diese Klasse repräsentiert das grafische Element, welches das Kamerabild anzeigt. Außerdem ist sie dafür zuständig, die verschiedenen Regions of Interest (ROIs) zu verwalten. Objekte dieser Klasse können durch Rechtsklick direkt im User-Interface des Master-Control-Programms platziert werden.

- **Data1DPlotArray** (`Data1DPlotArray.h`)
  Mithilfe dieser Klasse werden X/Y-Projektions-Graphen und ROI-Fluoreszenz-Graphen angezeigt. X/Y-Projektions-Graphen zeigen die Projektion des Kamerabildes auf die X- bzw. Y-Achse, während ROI-Fluoreszenz-Graphen die Summe des Fluoreszenzwertes aller Pixel innerhalb einer Region of Interest anzeigen.

- **CCameraHelper** (`CameraHelper.h`)
  Das Herzstück der neuen Kamera-Software. Diese Klasse stellt benutzerfreundliche und intelligente Methoden zur Steuerung der Kamera zur Verfügung. Darunter zählen z.B. der Video Modus, extern getriggerte sequenzielle Auslese von Bildern und automatische Ionendetektion. Außerdem beinhaltet diese Klasse alle für den Anwender nötigen GUI-Elemente.

- **CameraImage, CameraImageList** (`AndorSDK.h`)
  Von der Kamera im Video-Modus ausgelesene Bilder werden in `CameraImage`-Objekten gespeichert, im Sequenzmodus ausgelesene Bilder in `CameraImageList`-Objekten. In `CameraImageList`-Objekten sind mehrere `CameraImages` zusammengefasst. Beide Klassen beinhalten verschiedene Methoden zur Manipulation und Auswertung von Bildern.

Abbildung A.2 zeigt eine Zusammenstellung aller relevanten GUI-Elemente. Um den vollen Zugriff auf eine Kamera zu erhalten, muss man lediglich wenige Zeilen in die Script.Dll einfügen:

```
1 #include "AndorSDK.h"
2 #include "Data2DPlotArray.h"
3 #include "CameraHelper.h"
4
5 EXPORT Camera iXon;
6 EXPORT Data2DPlotArray camImg(iXon.getXPixels(), iXon.getYPixels());
7 EXPORT CCameraHelper camHelper(&iXon, &camImg);
```

In den folgenden Abschnitten wird auf die einzelnen Funktionalitäten der Kamera-Software näher eingegangen.

A. Hardware-Einstellungen

Zu den Hardware-Einstellungen zählen die Kontrolle des Shutters, des Kühlers und verschiedene Einstellungen zur Beeinflussung der Bildaufnahme. Die Software enthält eine automatische Shutter-Kontrolle, die den Shutter öffnet, sobald der Video-Modus oder

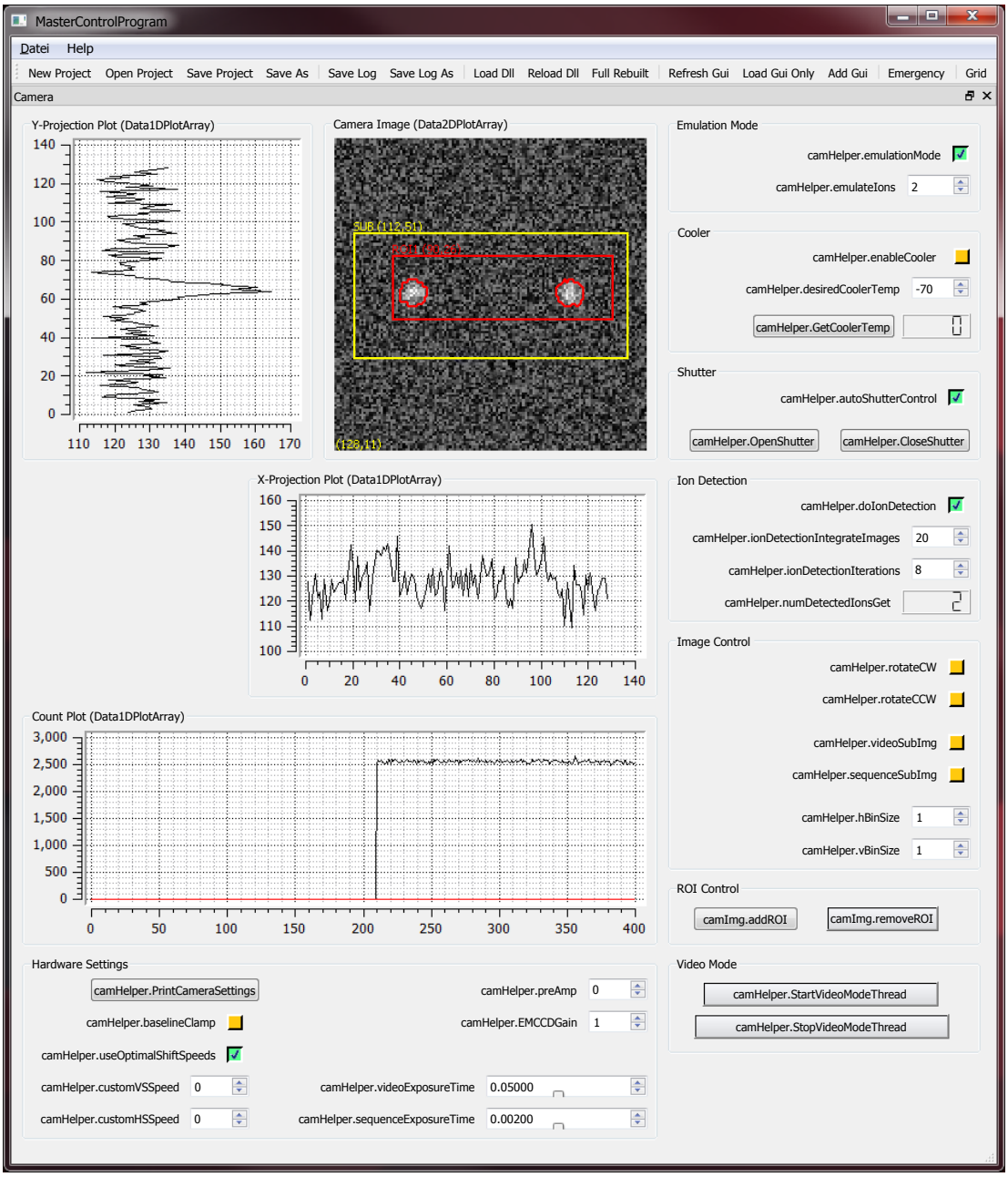


Abbildung A.2.: Zusammenstellung aller GUI-Elemente der Kamera-Software

Sequenzen gestartet werden und ihn wieder schließt, wenn das Programm geschlossen wird oder die Kamera 30 Sekunden lang nicht verwendet wird. Diese Funktion lässt sich über das Kontrollkästchen `autoShutterControl` aktivieren. Darüber hinaus exis-

tieren die Schaltflächen `OpenShutter` und `CloseShutter` zur manuellen Kontrolle des Shutters.

Der Kühler wird über die Option `enableCooler` aktiviert. Mit ihm wird der EMCCD-Sensor auf die Temperatur gekühlt, die in `desiredCoolerTemp` eingetragen ist. Mit der Schaltfläche `GetCoolerTemp` lässt sich die momentane Temperatur überprüfen.

Die Geschwindigkeit, mit der Ladung auf dem CCD-Sensor verschoben wird, kann über die Optionen `customVSSpeed` für die vertikale und `customHSSpeed` für die horizontale Richtung eingestellt werden. Die eingegebene Zahl wählt zwischen mehreren verfügbaren Einstellungen. Welcher Einstellung welcher Geschwindigkeit entspricht, lässt sich mit der Schaltfläche `PrintCameraSettings` ausgeben. Möchte man die Geschwindigkeiten nicht manuell festlegen, kann man sie sich durch Aktivierung des Kontrollkästchens `useOptimalShiftSpeeds` automatisch bestimmen lassen.

Weitere Hardware-Optionen sind

`baselineClamp` Auf die ausgelesenen Werte wird immer ein gewisser Basiswert dazuaddiert. Dem Andor-Manual zufolge sorgt diese Funktion dafür, dass dieser Basiswert konsistenter zwischen den Bildern einer Sequenz ist. Die genaue Wirkung dieser Option ist unklar.

`preAmp` Einstellung des integrierten Vorverstärkers. Wie bei den Shift-Geschwindigkeiten wählt man hier mit der eingegebenen Zahl zwischen verschiedenen Vorverstärker-Einstellungen. Die zugeordneten Verstärkungsfaktoren lassen sich mit der `PrintCameraSettings`-Funktion anzeigen. Messungen haben ergeben, dass die Vorverstärker-Einstellung keinen Einfluss auf die Qualität der Hell-/Dunkel-Unterscheidung von Ionen hat.

`EMCCD-Gain` Eine ganze Zahl zwischen 1 und 255, die den Verstärkungsfaktor des EMCCD-Chips einstellt

`videoExposureTime` Belichtungszeit pro Bild, die im Videomodus verwendet wird

`sequenceExposureTime` Belichtungszeit pro Bild im Sequenzmodus

`hBinSize` Fasst mehrere Pixel in horizontaler Richtung zu einem einzigen zusammen. Dies geschieht hardwareseitig und kann das Signal-/Rauschen-Verhältnis verbessern

`vBinSize` Fasst mehrere Pixel in vertikaler Richtung zu einem einzigen zusammen. Dies geschieht hardwareseitig und kann das Signal-/Rauschen-Verhältnis verbessern

`rotateCW` Rotiert das Bild im Uhrzeigersinn. Die Rotation wird softwareseitig auf dem Kontrollrechner durchgeführt

`rotateCCW` Rotiert das Bild gegen den Uhrzeigersinn. Die Rotation wird softwareseitig auf dem Kontrollrechner durchgeführt

B. Regions of Interest

Auf dem Kamerabild können beliebig viele Regions of Interest platziert werden. Das sind rechteckige Bereiche, die auf verschiedene Weise benutzt werden können. Beim Klick auf die `camImg.addROI`- bzw. `camImg.removeROI`-Schaltflächen werden neue Regionen hinzugefügt bzw. entfernt. Durch Linksklick auf die Rahmen im Kamerabild lassen sich die Dimensionen der Regionen verändern, die gesamte Region kann mithilfe der mittleren Maustaste verschoben werden. Auf die Koordinaten kann man im Quelltext folgendermaßen zugreifen:

```
for(unsigned int i = 0; i < camImg.getNumROIs(); i++) {
  CRoi roi = camImg.getROI(i);
  cout << "ROI Koordinaten: (" << roi.xMin << "," << roi.yMin << "), ("
      << roi.xMax << "," << roi.yMax << ")" << endl;
}
```

Die Region mit dem Index 0 ist das *SubImage*. Durch Verwendung der Optionen `videoSubImage` bzw. `sequenceSubImage` wird die Kamera so eingestellt, nur den durch das SubImage markierten Bildbereich auszulesen. Dadurch wird die aufgenommene Datenmenge reduziert und Bilder können schneller hintereinander aufgenommen werden.

Die `CameraImage`-Klasse stellt die Methode `CameraImage::SumInROI(const CRoi & roi)` zur Verfügung, die die Fluoreszenzwerte aller in einer Region of Interest eingeschlossenen Pixel aufsummiert. Die Regionen werden außerdem für die automatische Ionendetektion genutzt.

C. Video-Modus

Beim Video-Modus werden fortlaufend Bilder aufgenommen und im `Data2DPlotArray`-Element auf dem GUI direkt angezeigt. Der Video-Modus lässt sich über die Schaltflächen `Start`- bzw. `StopVideoModeThread` an- bzw. ausschalten. Es lassen sich zusätzlich zur eigentlichen Bildanzeige noch X/Y-Projektions-Graphen und ROI-Fluoreszenz-Graphen einbinden. Diese müssen in die Script.Dll eingefügt werden durch

```
1 #include "Data1DPlotArray.h"
2
3 EXPORT Data1DPlotArray camCntPlot1;
4 EXPORT Data1DPlotArray camCntPlot2;
5 EXPORT Data1DPlotArray camXProjPlot;
6 EXPORT Data1DPlotArray camYProjPlot;
7
8 class Init1DPlots {
9 public:
10   Init1DPlots() {
11     camHelper.SetPlotROI(1, 400, camCntPlot1);
12     camHelper.SetPlotROI(2, 400, camCntPlot2);
13
14     camHelper.SetXProjectionPlot(camXProjPlot);
15     camHelper.SetYProjectionPlot(camYProjPlot);
16   };
17 };
18
19 Init1DPlots init1dPlots; // Init1DPlots Konstruktor starten
```

Es wurden zunächst in den Zeilen 3-6 Objekte für die einzelnen Graphen definiert. Dann wurde eine Klasse `Init1DPlots` definiert, deren einziger Sinn darin besteht, Code beim Programmstart auszuführen. Das geschieht durch den Aufruf des Konstruktors bei der Initialisierung des `init1dPlots`-Objekts in Zeile 19. Im Konstruktor wird das `CameraHelper`-Objekt mit den einzelnen Graphen verbunden. Das erste Argument des `SetPlotROI`-Befehls ist die ROI-Nummer, das zweite Argument die Anzahl an Werten, die maximal angezeigt werden soll. Es können prinzipiell beliebig viele ROI-Fluoreszenz-Graphen eingefügt werden. Möchte man aufgenommene Bilder auswerten oder manipulieren, kann man dies in der Script.Dll tun, ohne Änderungen am Quelltext des Master-Control-Programms durchführen zu müssen. Dafür stellt die `CameraHelper` -Klasse die `WaitForNewImageAndPauseVideoMode()` und `ResumeVideoMode()`-Methoden zur Verfügung. So lässt sich der Video-Modus für kurze Zeit pausieren, um auf dem momentanen Kamera-Bild Operationen durchführen zu können. Der folgende Codeabschnitt speichert das nächste aufgenommene Bild in einer Textdatei ab:

```
1   camHelper.WaitForNewImageAndPauseVideoMode();
2
3   CameraImage * img = camHelper.GetLastImage();
4   img->SaveAsTxt("Bild.txt");
5
6   camHelper.ResumeVideoMode();
```

D. Sequenzen

Im Sequenzmodus ist die Bildaufname nicht kontinuierlich, sondern wird von einem externen Trigger gesteuert. Die Kamera speichert dann eine vorher definierte Anzahl von Bildern ab, die dann alle zusammen zum Rechner transferiert werden können. Dazu

gibt es in der `CameraHelper`-Klasse die `StartTriggeredKineticSeries(int imageCount)`- und `ReadoutSequenceData()`-Methoden, welche nach folgendem Beispiel benutzt werden können:

```
1  camHelper.StartTriggeredKineticSeries(imgCount);
2
3  // Triggersignale senden ....
4  // ..
5
6  // Wenn alle Triggersignale gesendet wurden, können Daten ausgelesen
       werden
7  CameraImageList * list = camHelper.ReadoutSequenceData();
8
9  // Auf einzelne Bilder der Sequenz zugreifen
10 CameraImage img;
11 list->ExtractImage(imageNum, img);
```

E. Emulationsmodus

Damit Softwarearbeiten durchgeführt werden können, wenn keine Kamera am Rechner angeschlossen oder wenn keine gefangenen Ionen vorhanden sind, wurde ein Emulationsmodus implementiert. Dieser lässt sich mit dem `emulationMode`-Kontrollkästchen auf dem GUI aktivieren. Mit dem Eingabefeld `emulateIons` kann die gewünschte Ionenzahl eingestellt werden. Der Emulationsmodus ist sowohl für den Video- als auch für den Sequenzmodus implementiert worden.

F. Ionendetektion

Die neue Software verfügt über einen Algorithmus, der im Video-Modus automatisch Ionen detektieren kann. Dafür muss lediglich die `doIonDetection`-Option aktiviert werden. Gefundene Ionen werden dann im Kamerabild rot umrandet, die Anzahl an gefundenen Ionen wird im Feld `numDetectedIonsGet` ausgegeben. Der Algorithmus arbeitet intern mit einer Überlagerung der letzten aufgenommenen Kamerbilder. Für jedes neue Kamerabild werden die einzelnen Pixel des überlagerten Bildes nach der Vorschrift

$$p_{\text{neue Überlagerung}} = p_{\text{Überlagerung}} \cdot \frac{\mathrm{n}-1}{\mathrm{n}} + p_{\text{neues Kamerabild}} \cdot \frac{1}{\mathrm{n}}$$

aktualisiert, mit p dem Fluoreszenzwert eines Pixels. Die Variable n ist ein Maß dafür, wieviele vergangene Bilder in die Überlagerung eingehen sollen. Diese Variable entspricht der GUI-Variable `ionDetectionIntegrateImages`. Je höher der Wert der Variable eingestellt ist, desto stabiler wird der Algorithmus gegen kurze Fluktuationen im Kamerabild. Gleichzeitig wird aber auch mehr Zeit benötigt, um neu gefangene Ionen zu erkennen. Nachdem das überlagerte Kamerabild berechnet wurde, wird für jede platzierte Region of Interest folgender Algorithmus durchlaufen:

- Berechne den Mittelwert $\overline{p}$ und die Standardabweichung σ_p der Werte aller Pixel des überlagerten Bildes.
- Markiere alle Pixel, deren Wert über $\overline{p} + 3\sigma_p$ liegt
- Wiederhole die letzten beiden Schritte mehrmals, wobei markierte Pixel ignoriert werden sollen. Die Anzahl an Wiederholungen kann über die GUI-Variable `ionDetectionIterations` eingestellt werden.

Sofern in einer Region of Interest genügend nichtbeleuchtete Pixel enthalten sind, kann durch dieses Verfahren effektiv Rauschen und Ionensignal getrennt werden. Der zuletzt berechnete Mittelwert $\overline{p}$ und die zuletzt berechnete Standardabweichung σ_p beziehen sich dann ausschließlich auf nichtbeleuchtete Pixel. Alle Pixel, deren Wert oberhalb $\overline{p}+5\sigma_p$ liegt, werden dann Ionen zugeordnet. Die Zuordnung auf einzelne Ionen geschieht nach folgendem Verfahren:

- Suche den Pixel mit dem höchsten Fluoreszenzwert
- Finde alle benachbarten Pixel, deren Wert über $\overline{p} + 5\sigma_p$ ist
- Alle gefundenen Pixel werden markiert und einem Ion zugeordnet. Wiederhole dann die letzten beiden Schritte, wobei markierte Pixel ignoriert werden, bis keine Pixel mehr verfügbar sind.

Um in der Praxis bestmögliche Ergebnisse zu erzielen, sollte man eine große Region of Interest platzieren, die viele dunkle Pixel, aber kein Laser-Streulicht beinhaltet. Sind Ionen sehr nah beieinander oder unterschiedlich hell, so kann es notwendig sein, mehrere Regionen zu setzen. Dabei sollten sich die einzelnen Regionen nicht überlappen, sonst können Ionen doppelt gezählt werden. Ist die Option `videoSubImg` aktiviert, müssen sich die Regionen komplett innerhalb des SubImages befinden.

Die Daten der detektierten Ionen werden in `CIon`-Objekten abgespeichert. Über die `CIon` -Objekte erhält man Zugriff auf die einzelnen Pixel, die den Ionen zugeordnet wurden. Man kann sich z.B. auf einfache Art und Weise die Positionen aller detektierten Ionen des nächsten Kamerabildes ausgeben lassen:

```
1  double x,y;
2  camHelper.WaitForNewImageAndPauseVideoMode();
3
4  for(unsigned int i = 0; i < camHelper.GetNumIons(); i++) {
5    camHelper.GetIon(i)->CalculatePosition(x,y);
6    cout << "Ion " << i+1 << " Position: (" << x << "," << y << ")" <<
         endl;
7  }
8
9  camHelper.ResumeVideoMode();
```

A.3. Struktur der Mikrofallen-Script.Dll

Die Weiterentwicklung des Script.Dll-Codes des Mikrofallen-Experiments war eine der Hauptaufgaben dieser Diplomarbeit. Bei der Programmierung waren besonders folgende Gesichtspunkte wichtig:

- **Flexibilität**
  Das Mikrofallenexperiment hat als Ziel, die physikalen Grundlagen zum Aufbau eines Quantencomputers zu erforschen. Dazu ist es nötig, alle Aspekte von auftretenden Phänomenen genau zu untersuchen. Die Software muss deshalb einerseits in der Lage sein, sehr grundlegende Messungen, bei denen alle Parameter vom Benutzer kontrolliert werden können, durchzuführen, andererseits soll sie bei komplexen Messungen aber auch möglichst viele Aufgaben automatisch erledigen.

- **Wartbarkeit**
  Das Softwaresystem wird ständig von mehreren Personen erweitert, um den aktuellen Anforderungen eines Experiments gerecht zu werden. Dafür ist es besonders wichtig, dass der Quelltext gut dokumentiert, übersichtlich und leicht verständlich ist.

- **Einfachheit**
  Es soll auch unerfahrenen Programmierern möglich sein, innerhalb weniger Minuten neue experimentelle Sequenzen zu programmieren und auszuführen.

Besonders das Schreiben von gut dokumentiertem Quelltext nimmt in der Praxis eine Menge Zeit in Anspruch und konnte deshalb oft nur unzureichend realisiert werden. Gerade, wenn während eines Messtages schnelle Änderungen notwendig sind, fehlt oft die Zeit, um hochwertigen Quelltext zu schreiben.

Die Funktionalität der `Script.Dll` lässt sich in drei Kategorien einteilen:

- **Erstellung von experimentellen Pulssequenzen**
  Es werden die für die Hardware nötigen Daten erzeugt und transferiert.

- **Auswertung von Messergebnissen**
  Darunter fällt die Auswertung von Kamerabildern, das Speichern und das Anzeigen der Daten auf sinnvolle Art und Weise.

- **Automatisierung**
  Dieser Punkt wird besonders in Zukunft für komplexere Messungen wichtig werden. Damit sich der Experimentator auf das Wesentliche konzentrieren kann, sollten möglichst viele Parameter des Experiments automatisch bestimmt und regelmäßig zur Kompensation von Drifts korrigiert werden.

Zur Erstellung von Pulssequenzen wurden die `CMainSequence`-Klasse sowie die `CSequenceItem`-Klasse entwickelt. Die `CMainSequence`-Klasse fasst Sequenzen einzelner Hardware-Teile (Ni-Karten, VFG, Fallen-Spannungsversorgung, Kamera) des Experiments zusammen und kümmert sich um den synchronen Ablauf. Der eigentliche Inhalt der Sequenzen wird durch Funktionsaufrufe geregelt, die in zuvor erstellten `CSequenceItems` festgelegt sind. Die Auswertung wird durch verschiedene *ResultProcessor*-Klassen realisiert, die darauf spezialisiert sind, Rohdaten einer Messung auf eine bestimmte Weise aufzubereiten und mithilfe von Gnuplot Graphen auszugeben. Zur Verwaltung von automatischen Messungen dient die `CAutoMeasurementScheduler`-Klasse. Damit die Durchführung von automatischen Messungen nicht mit vom Benutzer gestarteten Messungen kollidiert, wurde die `CMainSequenceScheduler`-Klasse entwickelt, welche alle existierenden `CMainSequence`-Objekte verwaltet. So können Messungen in eine Warteschleife eingereiht, laufende Messungen unterbrochen und zu einem späteren Zeitpunkt wieder fortgesetzt werden. Von den beiden Scheduler-Klassen existiert im Programm jeweils nur genau ein Objekt, von allen anderen Klassen können mehrere Objekte existieren. Abbildung A.3 zeigt die wichtigsten Klassen der Script.Dll und ihre Beziehungen zueinander. In den folgenden Kapiteln wird genauer auf die einzelnen Teilaspekte der Software eingegangen.

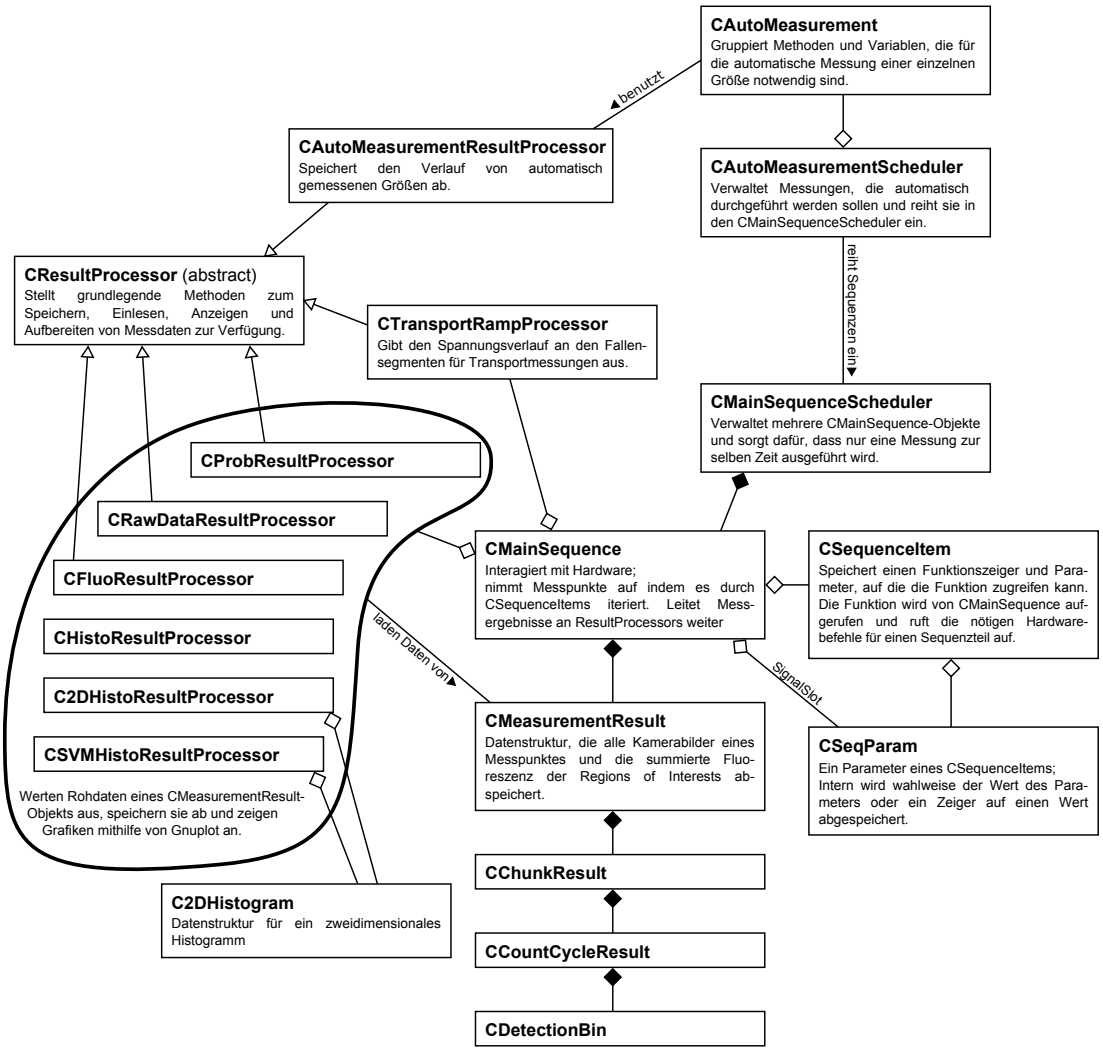
A. Durchführung einer Messung

Programmierung einer neuen Sequenz

Messungen sind im Script.Dll-Quelltext in der Datei `ExperimentsNew.cpp` definiert. Um eine neue Messung zu erstellen, muss lediglich eine neue Funktion eingefügt werden. Das Schema, nach dem eine solche Funktion aufgebaut ist, wird nun anhand einer simplen Spektroskopiesequenz erklärt: In dieser Sequenz soll die Frequenz eines akustooptischen Modulators variiert werden, der die Laserfrequenz des Lasers bei 729 nm ändert. Für jede eingestellte Frequenz wird dann folgende Prozedur durchgeführt:

1. Ein Ion wird durch Einstrahlung des Lasers bei 397 nm dopplergekühlt.
2. Durch optisches Pumpen wird das Ion im Spin-Up-Zustand initialisiert.
3. Es wird ein Laserpuls mit dem 729 nm-Laser eingestrahlt.
4. Dann wird wieder der Laser bei 397 nm eingeschaltet, während gleichzeitig ein Kamerabild aufgenommen wird. War die Anregung des 729 nm-Lasers erfolgreich, so befindet sich das Ion im $3D_{5/2}$-Zustand und emittiert kein Fluoreszenzlicht mehr, es ist dunkel. War die Anregung nicht erfolgreich, ist es hell.

Diese Sequenz wird dann mehrmals wiederholt, um die Anregungswahrscheinlichkeit des 729 nm-Pulses herauszubekommen. Der Quelltext zur Definition des gesamten Experiments lautet:



Legende:

A ——▷ B Generalisierung:
A erbt von B

A ——◇ B Aggregation:
A ist Bestandteil von B

A ——◆ B Komposition:
A ist untrennbarer Bestandteil von B; Die Existenz von A wird vollständig von B gesteuert

Abbildung A.3.: Klassendiagramm der Script.Dll.

```
1 EXPORT void Do729Spec(double startFreq, double endFreq, double stepFreq)
2 {
3   // Eine neue Sequenz vom MainSeqScheduler anfordern
4   CMainSequence * MainSeq = GMainSeqScheduler.CreateSequence("729
        Spectroscopy");
5
6   // Einstellungen setzen
7   MainSeq->SetPlotData(true); // Hell/Dunkel-Wahrscheinlichkeiten
        anzeigen
8   MainSeq->SetScanningValues(startFreq, endFreq, stepFreq);
9
10  // SequenceItems generieren
11  AddLaserInit(*MainSeq);
12  AddDopplerCool(*MainSeq, Gtiming.dopplerCoolTime);
13  AddSpinInitialization(*MainSeq);
14  Add729Pulse(*MainSeq,Gspec729.pulseTime,Gspec729.power,MainSeq->
        GetSignalSlot(),0.0,true);
15  AddDetection(*MainSeq, Gfluomeasurement.satCount, Gscan.DoStrongCool,
        DETECTION_WITH_866 + DETECTION_ADD_QUENCH);
16
17  // Sequenz in Warteschlange einreihen
18  GMainSeqScheduler.QueueExecution(MainSeq);
19 }
```

- Das `EXPORT`-Keyword in Zeile 1 sorgt dafür, dass die Funktion direkt auf dem grafischen Interface des Master-Control-Programms platziert werden kann. Die übergebenen Parameter legen fest, in welchen Grenzen und mit welcher Schrittweite die Frequenz des akustooptischen Modulators variiert werden soll.
- In Zeile 4 wird vom global definierten `CMainSequenceScheduler`-Objekt eine neue Sequenz mit Namen „729 Spectroscopy“ angefordert. Die Methode `CreateSequence` erstellt eine neue `CMainSequence` und gibt einen Zeiger auf diese zurück. Die Erstellung einer `CMainSequence` auf andere Art ist nicht möglich. So wird gewährleistet, dass alle Hardwarezugriffe vom globalen `CMainSequenceScheduler`-Objekt synchronisiert werden können.
- In Zeile 7 wird eine Einstellung gesetzt, die bewirkt, dass während der Messung die Hell/Dunkel-Wahrscheinlichkeiten in einem Graphen ausgegeben werden. Es gibt zahlreiche weitere Darstellungsmöglichkeiten der Ergebnisse, diese werden in Abschnitt B näher erläutert.
- In Zeile 8 werden die Grenzen und die Schrittweite der Frequenz des akustooptischen Modulators an das `CMainSequence`-Objekt übergeben.
- In den Zeilen 11-15 werden dann verschiedene Add-Befehle aufgerufen, um die Pulssequenz im Baukastenprinzip zusammenzusetzen. Alle Add-Befehle befinden sich in der Datei `sequenceToolkit.h`. Von jedem Add-Befehl existieren zwei Überladungen: Die erste Überladung wird in dieser Funktion bei der Erstellung einer Sequenz aufgerufen - sie erstellt ein neues `CSequenceItem`-Objekt und über-

gibt es an ein `CMainSequence`-Objekt, das als Argument mitgeliefert wird. Das `CSequenceItem`-Objekt besitzt einen Funktionszeiger, der auf die zweite Überladung des Add-Befehls zeigt. Dieser Funktionszeiger wird später durch das `CMainSequence`-Objekt aufgerufen, um die eigentlichen Hardwarebefehle zu erstellen. Besonders wichtig für die Sequenz ist hier der Aufruf des 729 nm-Pulses:

```
Add729Pulse(CMainSequence & parentSeq, const CSeqParam & pulseTime
    , const CSeqParam & power, const CSeqParam & freq, const
    CSeqParam & phase, const CSeqParam & setconfig);
```

Für die Frequenz wurde hier `MainSeq->GetSignalSlot()` als Argument verwendet. Das führt dazu, dass die Laserfrequenz während der Messung nach den Parametern des `SetScanningValues`-Aufrufs variiert wird. Der Typ der einzelnen Parameter, `CSeqParam`, wurde entwickelt, um nach Wahl Variablen oder Zeiger übergeben zu können. Letztere können während der Laufzeit des Experiments nachträglich geändert werden, während bei der Übergabe von Variablen Kopien erstellt werden, auf die der Benutzer nicht mehr zugreifen kann. Die Übergabe von Zeigern ist z.B. sinnvoll für Laserfrequenzen, die zwischendurch von automatischen Messungen nachgemessen werden können. Die `CSeqParam`-Klasse definiert zur einfachen Benutzen verschiedene Konstruktoren und Typcast-Operatoren der Typen `double`, `int`, `bool` und `string`, sodass diese Typen implizit in `CSeqParam`-Objekte umgewandelt werden können und umgekehrt.

- Nachdem das Zusammensetzen der Sequenz abgeschlossen ist, wird in Zeile 18 dem globalen `CMainSequenceScheduler`-Objekt mitgeteilt, dass es die Sequenz in seine Warteschlange einreihen soll. Sie wird dann ausgeführt, sobald alle momentan laufenden Sequenzen abgeschlossen sind. Als optionalen weiteren Parameter kann man hier zusätzlich einen Funktionszeiger auf eine `ExecuteFunction` angeben.

Verwaltung mehrerer Sequenzen

Die `CMainSequenceScheduler`-Klasse ist in der Lage, drei `CMainSequence`-Objekte gleichzeitig zu verwalten (Die Anzahl kann bei Bedarf durch Änderung der Konstante `SCHEDULER_MAX_NUM_SEQUENCES` erhöht werden). Beim Anfordern einer neuen `CMainSequence` durch die `CreateSequence`-Methode werden zunächt bereits vollendete Messungen gelöscht, um Platz freizugeben. Ist bereits eine Messung im Gange, so werden die weiteren verfügbaren Plätze aufgefüllt. Sind alle Plätze belegt, gibt die `CreateSequence`-Methode einen `NULL`-Pointer zurück. Die Sequenzen können über das GUI verwaltet werden (Abbildung A.4). Es wird einerseits der Status aller Messungen angezeigt, andererseits lassen sie sich auch direkt pausieren oder beenden. Automatische Messungen werden immer zuerst in den dritten Sequenzplatz gelegt, um Kollisionen zwischen automatischen und manuellen Messungen zu vermeiden. Mit den `SaveResults` -Schaltflächen lassen sich die Messergebnisse des jeweiligen Scans auf dem Gruppenlaufwerk abspeichern. Die `RemoveSeq`-Schaltflächen löschen Messungen aus der Liste

und schließen alle offenen Gnuplot-Fenster, die mit der jeweiligen Messung verbunden waren.

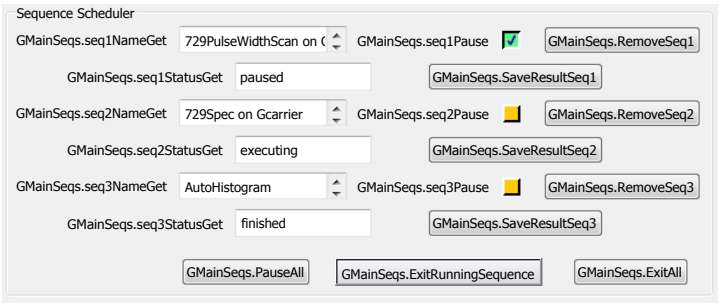


Abbildung A.4.: GUI-Elemente zur Steuerung des `CMainSequenceSchedulers`. Hier wurde ein Pulslängenscan mit dem 729 nm-Laser pausiert, um eine Spektroskopiesequenz einzuschieben.

ExecuteFunctions

Der Sinn von `ExecuteFunctions` wird deutlich, wenn man die gesamte Struktur einer Messung am Mikrofallenexperiment betrachtet (Abbildung A.5). Eine Messung besteht aus mehreren *Messpunkten*, wobei für jeden Messpunkt der Parameter, der variiert wird, einen anderen Wert hat. Dieser Parameter wird *SignalSlot* genannt. Für das Setzen des SignalSlots und Wiederholen der Messungen ist die `ExecuteFunction` verantwortlich. Wird keine eigene `ExecuteFunction` angegeben, kommt die `DefaultExecuteFunc` zum Einsatz. Diese stellt drei verschiedene Arten von Messungen zur Verfügung, die für den Großteil der Experimente bereits ausreichend sind:

- **Der Standardscan**
  Hier werden vom Benutzer ein Startwert, Endwert und eine Schrittweite vorgegeben. Beim ersten Messpunkt wird der SignalSlot auf den Startwert gesetzt, für alle weiteren Punkte wird die Schrittweite dazuaddiert, bis der Endwert erreicht ist. Die Parameter werden durch den oben verwendeten Aufruf `CMainSequence::SetScanningValues` eingestellt.

- **Der CenterScan**
  Hier müssen ein Startwert, eine Schrittweite und eine Scan-Breite vorgegeben werden. Beim ersten Messpunkt wird der SignalSlot wie beim Standardscan auf den Startwert gesetzt, dann werden aber abwechselnd mit steigendem Abstand auf der linken und der rechten Seite des Startwertes Messpunkte aufgenommen, bis die Scan-Breite erreicht ist. Diese Messung ist besonders sinnvoll, um Größen zu bestimmen, die bereits ungefähr bekannt sind. So kann die Umgebung um den vermuteten Wert schnell untersucht werden. Der CenterScan lässt sich durch den Aufruf `CMainSequence::SetCenterScan` aktivieren.

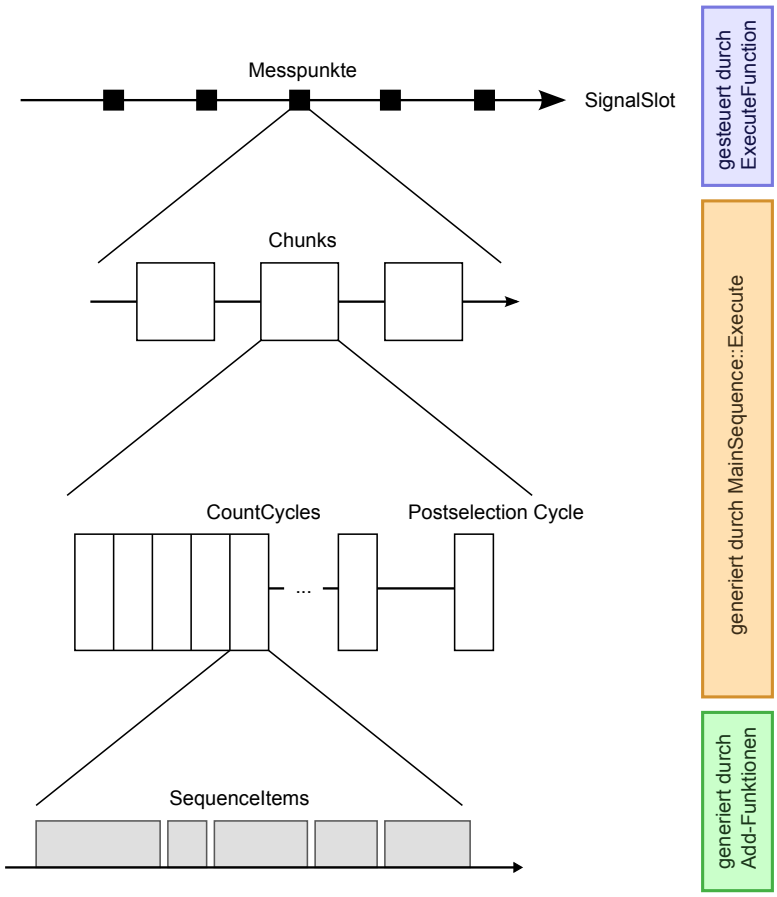


Abbildung A.5.: Vollständige Struktur einer Messsequenz

- **Der PseudoScan**
  Hier wird ein fester Wert für den SignalSlot vorgegeben. Die Messpunkt wird dann ständig mit genau diesem Wert wiederholt, bis der Benutzer die Messung abbricht. Der PseudoScan wird hauptsächlich während Justagearbeiten verwendet, um z.B. Rabifrequenzen zu maximieren. Er lässt sich durch den Aufruf `CMainSequence::SetPseudoScan` aktivieren. Außerdem kann durch Setzen der globalen Variable `Gfluoemasurement.DoPseudoScan` ein Standard- oder CenterScan in einen PseudoScan umgewandelt werden.

Abbildung A.6 zeigt zur Illustration einen Auszug aus der `DefaultExecuteFunc`. Jede `ExecuteFunction` wird vom globalen `CMainSequenceScheduler`-Objekt in einem eigenen Thread gestartet. Der übergebene Parameter ist ein Zeiger auf die gerade ausgeführte `CMainSequence`. Die Variablen `seq->mStopSequence` und `seq->mPauseSequence` werden vom `CMainSequenceScheduler`-Objekt gesetzt, um zu signalisieren, ob eine Messung gestoppt oder pausiert werden muss. Ist eine von beiden Variablen gesetzt, wird das `seq`

```
1 DWORD WINAPI CMainSequenceScheduler::DefaultExecuteFunc(LPVOID param)
2 {
3   CMainSequence *seq =(CMainSequence *)param;
4
5   double signalSlotVal = seq->mScanFrom;
6
7   while(!seq->mStopSequence) {
8
9     if(seq->mPauseSequence) {
10      SetEvent(seq->mSequenceFinishedEvent);
11      WaitForSingleObject(seq->mResumeEvent, INFINITE);
12      if(seq->mStopSequence) break;
13    }
14
15    if(seq->Execute(signalSlotVal)) {
16      seq->PlotResult();
17
18      signalSlotVal += seq->mScanStep;
19
20      if(signalSlotVal > (seq->mScanTo + DBL_EPSILON) && seq->mScanStep
            >0.0) break; // break out of loop here
21      if(signalSlotVal < (seq->mScanTo - DBL_EPSILON) && seq->mScanStep
            <0.0) break; // break out of loop here
22
23    } else if(!seq->mStopSequence) {
24      cout << "Measurement invalid, setting to pause" << endl;
25      GMainSeqScheduler.PauseAllSequences();
26    }
27
28  }
29
30  SetEvent(seq->mSequenceFinishedEvent);
31  return 0;
32 }
```

Abbildung A.6.: Auszug aus der `DefaultExecuteFunction`. Die Center- und PseudoScan-Funktionalität wurde zur besseren Übersicht weggelassen.

`->mSequenceFinishedEvent` gesetzt und die Messung gestoppt bzw. pausiert. Das `seq->mSequenceFinishedEvent` informiert das `CMainSequenceScheduler`-Objekt darüber, dass er mit der nächsten Messung fortsetzen kann. In Zeile 15 wird dann ein Messpunkt aufgenommen, darum kümmert sich die Methode `CMainSequence::Execute`. Bei erfolgreicher Ausführung werden die Ergebnisse angezeigt und der Wert des SignalSlots geändert. Bei einem Fehler wird die Messung pausiert.

Struktur eines Messpunktes

Messpunkte werden von `CMainSequence::Execute` in mehreren *Chunks* abgearbeitet. Chunks sind wiederum aufgeteilt in mehrere *CountCycles* und ein optionales *Postselection Cycle*. Die CountCycles werden im Experiment direkt hintereinander ausgeführt - sie enthalten die eigentliche Pulssequenz, die durch die verschiedenen Add-Befehle definiert wurde. Zur Erzeugung der Sequenzdaten (für alle CountCycles auf einmal) für Ni-Karten, VFG und Fallen-Spannungsversorgung werden die Funktionszeiger aufgerufen, die in den vorher erstellten `CSequenceItem`-Objekten hinterlegt sind. Die Hardwaredaten werden dann, je nach Anzahl der Chunks, mehrmals transferiert und ausgeführt. Beim Postselection-Cycle am Ende eines Chunks wird überprüft, ob von allen Ionen noch Fluoreszenzlicht erzeugt wird. Falls nicht, wird die Messung verworfen und wiederholt. Das Aufteilen in Chunks kann aus mehreren Gründen sinnvoll sein:

1. Bei großen Sequenzen oder einer hohen Anzahl an CountCycles kann die Sequenz zu groß für den Speicher der einzelnen Geräte werden und muss in kleinere Teile aufgeteilt werden.

2. Es sind mehrere Postselection-Cycles für einen Messpunkt erwünscht.

3. Linetriggering: Ist diese Option eingeschaltet, wird der Ablauf des Experiments mit der Frequenz der Netzspannung synchronisiert, um Magnetfeldfluktuationen zu minimieren. Es werden dann Wartezeiten berechnet, sodass jeder CountCycle bei gleicher Spannungsamplitude stattfindet. Die Synchronisation mit der Netzspannung findet zu Beginn eines jeden Chunks statt.

Wie ein Chunk von der Hardware abgearbeitet wird, ist in Abbildung A.7 visualisiert. Zunächst werden die benötigten Daten an die verschiedenen Geräte transferiert. VFG, Fallen-Spannungsversorung und Kamera gehen daraufhin in einen Wartemodus. Die Ni-Karten sind für den gesamten zeitlichen Ablauf der Sequenz verantwortlich, sie senden Triggersignale an die anderen Geräte, damit diese ihre Aufgaben erledigen, wenn sie benötigt werden. Der genaue zeitliche Ablauf der Trigger-Signale hängt vom spezifischen Experiment ab.

B. Datenauswertung

Nachdem eine Messung durchgeführt wurde, müssen alle aufgenommenen Kamerabilder ausgewertet werden. Diese Aufgabe wird von der `CMeasurementResult`-Klasse und den verschiedenen ResultProcessor-Klassen erledigt.

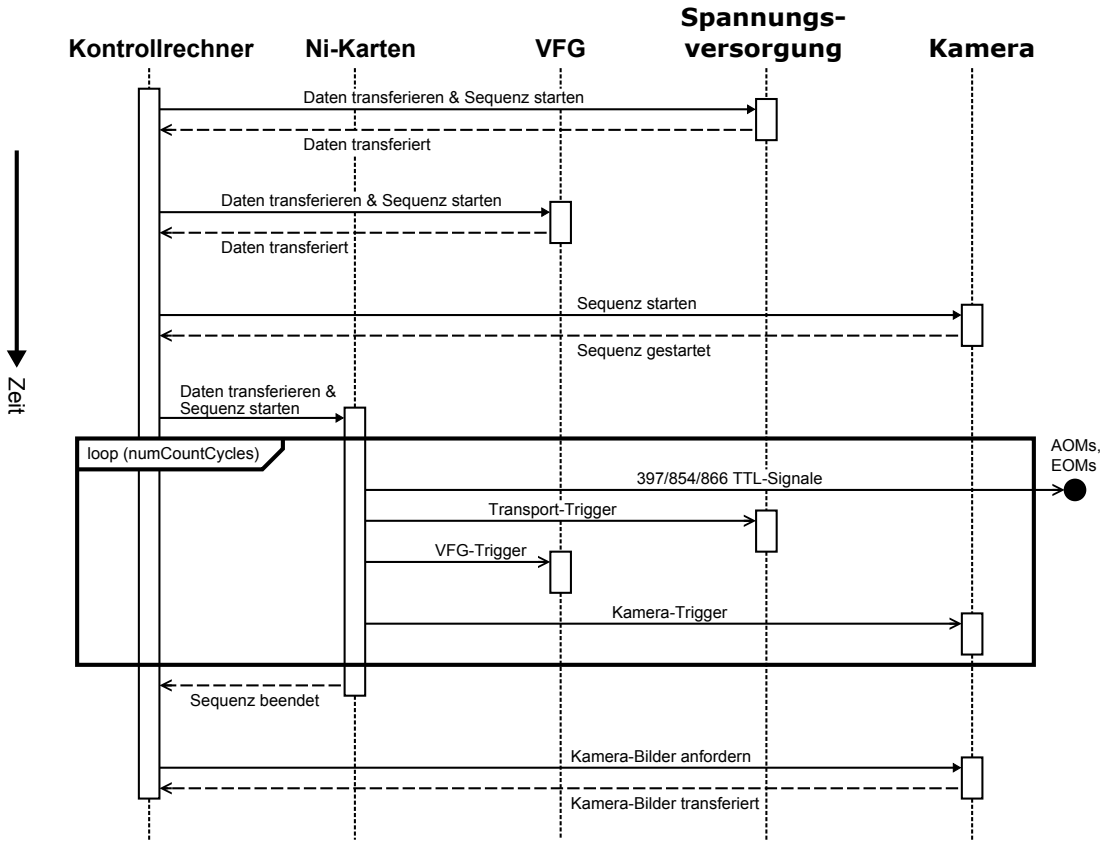


Abbildung A.7.: Sequenzdiagramm des hardwareseitigen Ablaufs eines Chunks. Die Zeitachse läuft von oben nach unten. Gestrichelte Linien symbolisieren passive Zeit, in der die jeweilige Hardwarekomponente abwartet - weiße Balken stehen für aktive Zeit.

Die CMeasurementResult-Klasse

Die `CMeasurementResult`-Klasse stellt eine Datenstruktur zur Verfügung, welche alle Kamerabilder eines Messpunktes nach Chunks und CountCycles geordnet abspeichert. Darüberhinaus werden Kamerabilder mit einem Attribut versehen, welches die Bilder von echten CountCycles und Postselection-Cycles unterscheidet. Pro CountCycle können mehrere Kamerabilder aufgenommen werden - den Kamerabildern eines CountCycles werden dann verschiedene Spuren (Traces) zugeordnet, die beim Auswerten der Ergebnisse getrennt voneinander behandelt werden. Während der Erstellung der Ni-Sequenzen für einen Chunk in der `CMainSequence`-Klasse wird in einem zunächst leeren `CMeasurementResult`-Objekt ein Muster angelegt. Das Muster ist vom Typ `CChunkResult`, es besteht intern aus mehreren `CCountCycleResult`-Objekten, die selbst wiederum aus mehreren `CDetectionBin`-Objekten bestehen. In diesen Objekten

werden die Daten der einzelnen Kamerabilder dann gespeichert. Bei der Erstellung der Ni-Sequenz wird in jedem Detektionsabschnitt die Methode

```
CMeasurementResult::AddDetectionBinToPattern(unsigned int countCycle,
    TcountBin label, double duration);
```

aufgerufen. Diese erweitert das Muster um ein `CDetectionBin`. Außerdem erstellt sie automatisch, falls notwendig, neue `CCountCycleResult`-Objekte und zählt die Anzahl aufzunehmener Spuren. Ist das Muster fertig erstellt, muss es so oft kopiert werden, bis eine Datenstruktur aufgebaut ist, die Platz für alle Chunks bietet. Das geschieht nach der Erstellung der Ni-Sequenz mit dem Befehl

```
CMeasurementResult::ApplyPattern();
```

Abbildung A.8 veranschaulicht den Zustand des `CMeasurementResult`-Objekts vor und nach dem Aufruf der `ApplyPattern`-Methode.

Nach einem vollendeten Messpunkt beginnt die Datenauswertung in `CMainSequence::Evaluate()`. Dort wird zunächst die Methode `PreprocessMeasurementDataEMCCD()` aufgerufen, die Kamerabilder in die einzelnen `CDetectionBins` einsortiert. Außerdem wird bereits die summierte Fluoreszenz jeder Region of Interest mit abgespeichert. Zur weiteren Auswertung wird dann das gefüllte `CMeasurementResult`-Objekt an verschiedene ResultProcessor-Objekte weitergeleitet.

Die ResultProcessor-Klassen

Die ResultProcessor-Klassen erfüllen mehrere Aufgaben:

1. Sie können die Rohdaten eines `CMeasurementResult`-Objekts auf bestimmte Art und Weise auswerten.
2. Die ausgewerteten Messergebnisse können in Textdateien abgespeichert werden. Zuvor generierte Textdateien können nachträglich wieder geladen werden.
3. Die Ergebnisse können mithilfe des Programms „Gnuplot“ visualisiert werden.

Die Basisfunktionalität zum Schreiben und Lesen von Textdateien wird von der abstrakten Templateklasse `CResultProcessor<S, T>` zur Verfügung gestellt. Dabei ist `S` der Typ des SignalSlots, d.h. des Paramters, der bei der Messung variiert wurde, und `T` der Typ der ausgewerteten Daten. Die Daten werden intern als `map<S, vector<T>>` abgespeichert. So werden jedem SignalSlot-Wert beliebig viele Werte des Typs `T` zugeordnet. Beim Schreiben in die Textdatei wird für jeden Wert des SignalSlots eine neue Zeile angelegt, in der zuerst der Wert des SignalSlots selbst gespeichert wird und dann von Tabulatoren getrennt die Elemente des Datenvektors aufgelistet werden. Die Datenstruktur `map` hat hier den Vorteil, dass die Zeilen automatisch für die korrekte Ausgabe durch Gnuplot nach dem SignalSlot vorsortiert werden.

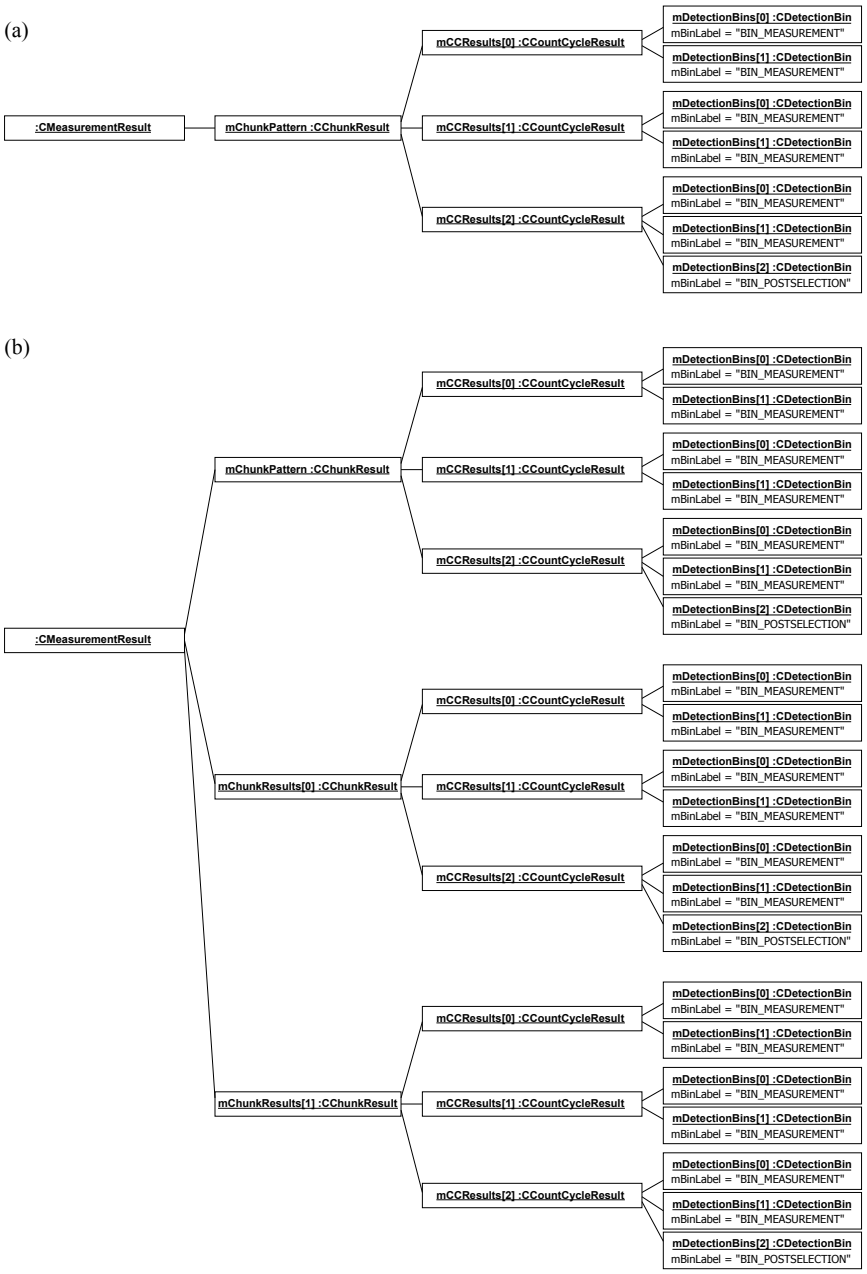


Abbildung A.8.: Objektdiagramme (a) vor und (b) nach dem `ApplyPattern`-Aufruf in einem `CMeasurementResult`-Objekt, das zu einem Experiment mit zwei Chunks, drei CountCycles und zwei Detektionen pro CountCycle gehört. Jedes Kästchen entspricht einem Objekt, die Notation innerhalb der Kästchen ist `Objektname:Klasse`. Bei `DetectionBins` ist das `mBinLabel`-Attribut zur Unterscheidung von normalen zu Postselection-Detektionen angegeben.

Von `CResultProcessor<S, T>` sind mehrere spezialisierte Klassen abgeleitet, die zusätzliche Funktionalität zur Auswertung eines `CMeasurementResult`-Objekts enthalten. Der Datenvektor der abstrakten `CResultProcessor`-Klasse wird jeder spezialisierten Klasse in einem anderen Format befüllt und gespeichert.

- **CProbResultProcessor**
  Diese Klasse stellt Messergebnisse als hell/dunkel-Wahrscheinlichkeiten dar. Für n Ionen existieren 2^n hell/dunkel-Kombinationen, deren Wahrscheinlichkeiten und Fehler nacheinander in den Datenvektor geschrieben werden. Falls die Messung aus mehreren Spuren besteht, werden die verschiedenen hell/dunkel-Kombinationen für jede Spur nacheinander eingetragen. Die Ergebnisse können visualisiert werden, indem die verschiedenen Wahrscheinlichkeiten gegen den SignalSlot aufgetragen werden.

- **CFluoResultProcessor**
  Hier werden die summierten Fluoreszenzwerte der einzelnen Regions of Interest über alle CountCycles gemittelt und nacheinander in den Datenvektor geschrieben. Sofern mehrere Spuren existieren, werden die Fluoreszenzwerte der einzelnen Spuren ebenfalls nacheinander eingetragen. Die grafische Ausgabe zeigt die Fluoreszenzwerte aufgetragen gegen den SignalSlot.

- **CRawDataResultProcessor**
  Mit dieser Klasse werden die Fluoreszenzwerte der einzelnen Regions of Interest für alle CountCycles einzeln abgespeichert. Das ist für Fälle gedacht, in denen Messergebnisse nachträglich ausgewertet werden müssen. Diese Klasse besitzt keine Möglichkeit zur grafischen Visualisierung.

- **CTransportRampProcessor**
  In dieser Klasse wird kein `CMeasurementResult`-Objekt ausgewertet, sondern eine von der `CMainSequence` erzeugte FPGA-Sequenz, welche die Fallenspannungen steuert. Die Klasse ist in der Lage, die fertig generierten Daten, die an den FPGA der Spannungsversorgung transferiert werden, auszuwerten und anzuzeigen. Das Ergebnis sind die Fallenspannung der einzelnen Segmente, aufgetragen gegen die Zeit innerhalb eines CountCycles.

- **CAutoMeasurementResultProcessor**
  Diese Klasse ist ebenfalls nicht zur Auswertung eines `CMeasurementResult`-Objekt gedacht. Sie wird von automatischen Messungen genutzt, um den zeitlichen Verlauf eines oder mehrerer Parameter zu speichern. So lässt sich beispielsweise der Drift von Laserfrequenzen in Abhängigkeit der Zeit anzeigen.

- **CHistoResultProcessor**
  Mithilfe dieser Klasse können die summierten Fluoreszenzwerte der Regions of Interest von allen CountCycles in ein Histogramm eingetragen und angezeigt werden. Für jede Spur und Ion wird ein separates Histogramm angelegt.

- **C2DHistoResultProcessor**
  Diese Klasse dient zur Erstellung und Verwendung eines 2D-Histogramms für die hell/dunkel-Unterscheidung von Ionen mithilfe des Maximum-Likelihood-Verfahrens.
- **CSVMHistoResultProcessor**
  In dieser Klasse können Daten zur hell/dunkel-Unterscheidung von zwei Ionen mithilfe des Support-Vector-Machine-Verfahrens ausgewertet und angezeigt werden.

Die drei Histogramm-Klassen sind nicht von `CResultProcessor<S, T>` abgeleitet, da sie eine andere zugrundeliegende Datenstruktur besitzen. Sie stellen trotzdem die selbe Funktionalität zur Verfügung.

C. Automatische Messungen

Zur automatischen Durchführung von Messungen müssen zwei wesentliche Komponenten bereitgestellt werden: Einerseits experimentelle Sequenzen, die in Echtzeit Messungen auswerten und daraus Parameter extrahieren können, andererseits müssen diese Messungen automatisch in regelmäßigen Zeitabständen durchgeführt werden.

Programmierung einer automatischen Messung

Automatische experimentelle Sequenzen werden in `AutoExperiments.h` abgelegt. Die Programmierung unterscheidet sich nur geringfügig von normalen Experimenten, hier am Beispiel der automatischen 729-Spektroskopiesequenz:

```
1  void DoAuto729Spec( CAutoMeasurement & autoMeasurement )
2  {
3    CMainSequence * MainSeq = GMainSeqScheduler.CreateSequence(
         autoMeasurement);
4    MainSeq->SetPlotData(true); // plot probability data
5    MainSeq->SetNumCountCycles(Gscan.countCyclesSpec);
6    MainSeq->SetPseudoScan(false); // automeasurements should not use
         pseudoscan
7
8    AddLaserInit(*MainSeq);
9    AddDopplerCool(*MainSeq, Gtiming.dopplerCoolTime);
10   if(autoMeasurement.GetPeak()->GetName() == "GcarrierM")
         AddSpinInitialization(*MainSeq, Gpump.pump729, false);
11   else AddSpinInitialization(*MainSeq);
12   Add729Pulse(*MainSeq,Gspec729.pulseTime,Gspec729.power,MainSeq->
         GetSignalSlot(),0.0,true);
13   AddDetection(*MainSeq, Gfluomeasurement.satCount, Gscan.DoStrongCool,
         DETECTION_WITH_866 + DETECTION_ADD_QUENCH);
14
```

```
15   GMainSeqScheduler.QueueExecution(MainSeq, Auto729SpecExecuteFunc);
16 }
```

Die Funktion erhält als Übergabeparameter ein `CAutoMeasurement`-Objekt. Für jede Größe, die automatisch gemessen werden soll, muss ein solches globales Objekt definiert werden. Alle globalen Objekte sind in `globals.cpp` definiert, beispielsweise die 729-Spektroskopiemessungen:

```
EXPORT CAutoMeasurement GAuto729SpecCarrier(GAutoScheduler, "
    Auto729Spec", &DoAuto729Spec, Gcarrier);
EXPORT CAutoMeasurement GAuto729SpecCarrier2(GAutoScheduler, "
    Auto729Spec", &DoAuto729Spec, Gcarrier2);
EXPORT CAutoMeasurement GAuto729SpecCarrierM(GAutoScheduler, "
    Auto729Spec", &DoAuto729Spec, GcarrierM);
```

Der erste Parameter ist eine Referenz auf das globale `CAutoMeasurementScheduler`-Objekt, der zweite Parameter der Name der Messung, der dritte Parameter ein Funktionszeiger auf die oben definierte Funktion und der vierte Parameter ist eine Referenz auf ein globales `peak_s`-Objekt, welches Eigenschaften eines bestimmten Übergangs gruppiert. Hier sind z.B. drei automatische Messungen definiert, die jeweils Frequenzen von Übergängen zwischen verschiedenen Zeeman-Unterzuständen messen sollen.

Im Gegensatz zu normalen Sequenzen wird beim Erstellen einer neuen `CMainSequence` eine andere Überladung der `CMainSequenceScheduler::CreateSequence`-Methode genutzt, die als Parameter das übergebene `CAutoMeasurement`-Objekt erhält. So wird der `CMainSequenceScheduler` dazu veranlasst, die Sequenz in seinen letzten Platz abzuspeichern, sodass keine Kollisionen mit manuell gestarteten Sequenzen entstehen. Dann werden wie üblich Parameter der `CMainSequence` gesetzt und die Sequenz mit verschiedenen Add-Methoden zusammengebaut. Es ist hier nicht nötig, die Methode `CMainSequence::SetScanningValues` aufzurufen, da in Zeile 15 eine eigene `ExecuteFunction` angegeben ist, die die Aufnahme von Messpunkten steuert. Dort steckt dann auch die eigentliche Funktionalität der automatischen Messung: Nach jedem Messpunkt wird in den Daten nach einem Peak gesucht, und falls gefunden, die Frequenz des Peaks mit einem gewichteten Mittelwert bestimmt.

Regelmäßige Ausführung von Messungen

Die regelmäßige Ausführung von automatischen Messungen wird vom globalen `CAutoMeasurementScheduler`-Objekt und den verschiedenen `CAutoMeasurement`-Objekten kontrolliert. Jedes `CAutoMeasurement`-Objekt verfügt über zwei GUI-Variablen `enable` und `measureFreqMinutes`, mit denen sich einstellen lässt, ob und wie oft ein automatisches Experiment gestartet werden soll. Zur manuellen Ausführung existiert außerdem die `DoNow()`-Methode. Im Konstruktor registriert sich jedes `CAutoMeasurement`-Objekt automatisch im `CAutoMeasurementScheduler`. Dieser entscheidet dann, wann

und in welcher Reihenfolge die nächsten automatischen Messungen ausgeführt werden sollen. Die nächste geplante Messung wird im GUI mitgeteilt (Abbildung A.9). Mit `DoNext` lässt sich die nächste geplante Messung direkt starten, während mit `DelayMeasurements` alle Messungen um 10 Minuten verzögert werden.

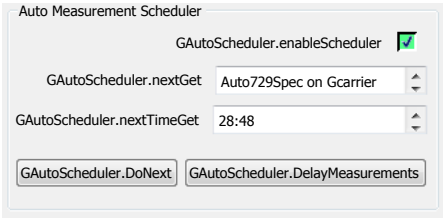


Abbildung A.9.: GUI-Elemente zur Kontrolle automatischer Messungen

D. Ideen zur Weiterentwicklung

In diesem Abschnitt sollen kurz Ideen zur Weiterentwicklung der Software gegeben werden, die bisher aus Zeitgründen nicht umgesetzt werden konnten. Die Ideen konzentrieren sich darauf, die Software für die Zukunft flexibler und wartbarer zu gestalten.

- **CSequenceItems als Baumstruktur**
  In einigen Fällen ist es sinnvoll, in den Add-Funktionen zur Erstellung von Pulssequenzen andere Add-Funktionen aufzurufen. So greift die allgemeine Funktion `AddSpinInitialization` z.B. auf die beiden Funktionen `AddOpticalPump` und `AddOpticalPump729` zurück. Das wird an dieser Stelle gelöst, indem die erste Überladung der Funktion, welche das `CSequenceItem`-Objekt erstellt, die zweite Überladung, in der eigentlich die Hardware-Sequenzen erstellt werden, mit dem CallFlag `CALL_CREATE` aufruft. Das kann zu unvorhergesehenem Verhalten führen, wenn man in der zweiten Überladung der `AddSpinInitialization`-Funktion selbst Hardware-Befehle geben möchte. Eine solche Situation gibt es in der `AddDetection` -Funktion: Dort ruft die zweite Überladung der Funktion zur Ausführung von PostSelection-Cycles die zweite Überladung der `AddDopplerCool`-Funktion auf. Da hier kein zweites `CSequenceItem` erstellt wird, nutzt die `AddDopplerCool`-Funktion die Parameterliste des ursprünglichen `AddDetection`-SequenceItems, was in der Vergangenheit schon oft zu unvorhergesehenen Fehlern geführt hat. Da der gegenseitige Aufruf von Add-Befehlen in Zukunft mit steigendem Komplexitätsgrad des Experiments mit hoher Wahrscheinlichkeit häufiger auftreten wird, müssen die genannten Zwischenlösungen unbedingt durch ein solides System ersetzt werden. Ein möglicher Ansatz wäre zum Beispiel, `CSequenceItems` als Baumstruktur aufzubauen. Dann könnte im Fall des `AddDetection`-Aufrufs ein `CSequenceItem` erstellt werden, welches ein durch `AddDopplerCool` erzeugtes `CSequenceItem` als Kind-Objekt beinhaltet.

- **Add-Funktionen ersetzen durch Vererbung von CSequenceItem**
  Anstatt ein `CSequenceItem`-Objekt zunächst in einer Add-Funktion zu erstellen, könnte man stattdessen auch Klassen von `CSequenceItem` ableiten, die im Konstruktor schon direkt alle benötigten Parameter entgegennehmen. Das würde zu übersichtlicherem Quelltext führen, da man nicht mehr zwischen den verschiedenen Überladungen der Add-Funktionen unterscheiden müsste. Auch könnte man in abgeleiteten Klassen leichter spezifische Funktionalität unterbringen, die für bestimmte `CSequenceItems` benötigt werden könnte.

- **ExecuteFunctions ersetzen durch Function Objects**
  Ein Problem am Konzept der `ExecuteFunctions` ist, dass keine für Spezialfälle benötigten Parameter übergeben werden können. Das wird momentan so gelöst, dass Variablen, die für spezielle `ExecuteFunctions` benötigt werden, als public-Variablen in die `CMainSequence`-Klasse eingetragen werden. Dadurch wird die `CMainSequence`-Klasse mit der Zeit allerdings immer unübersichtlicher. Ein Ausweg wäre die Benutzung von *Function Objects*: Das sind Objekte, für die ein eigener `()`-Operator definiert ist, und die so auch als Funktionszeiger übergeben werden können. In der `ExecuteFunction` könnte dann auf Membervariablen des Objekts zugegriffen werden.

  Ein möglicher anderer Ansatz wäre, die Funktionalität der `ExecuteFunctions` in die `CMainSequence`-Klasse zu verlagern. Man könnte dann eigene `ExecuteFunctions` implementieren, indem man eine neue von `CMainSequence` abgeleitete Klasse erstellt. Der Quelltext zum Erstellen einer neuen Messung würde dann so aussehen:

```
CCenterScan * MainSeq = new CCenterScan(start, step, width);

// Einstellungen setzen und Add-Funktionen aufrufen ...
// ...

GMainSeqScheduler.QueueExecution(MainSeq);
```

- **Alle ResultProcessoren sollten von CResultProcessor erben**
  Momentan erben nicht alle ResultProcessor-Klassen von der abstrakten `CResultProcessor`-Klasse, was zur Folge hat, dass identischer Quelltext an mehreren Stellen verwendet wird. Zur einfacheren Wartbarkeit sollte ein Weg gefunden werden, möglichst viel Quelltext zusammenzufassen.

- **Auslagerung der Klassenstruktur in das Master-Control-Programm**
  Die Script.Dll der Mikrofalle beinhaltet viel Funktionalität zur Sequenzsteuerung, die auch für andere Experimente der AG Schmidt-Kaler nützlich wäre. Daher wäre es sinnvoll, die wichtigsten Klassen in das Master-Control-Programm zu verschieben. Das momentane System nutzt allerdings an sehr vielen Stellen, z.B. für Hardwarezugriffe, globale Variablen, die im Master-Control-Programm nicht verfügbar

sind. Deshalb müssen zunächst alle Zugriffe zu globalen Variablen entfernt werden. Eine Alternative wäre zum Beispiel, dem `CMainSequenceScheduler`-Objekt im Konstruktor Referenzen zu Hardware-Objekten mitzugeben.

B

Heizmessungen an einem akustooptischen Modulator

Bei verschiedenen Messungen am Mikrofallenexperiment kam der Verdacht auf, dass Aufheizeffekte von den im Experiment genutzen akustooptischen Modulatoren eine systematische Fehlerquelle darstellen. So konnten beispielsweise Änderungen der Rabifrequenzen der Ramanübergänge festgestellt werden, wenn vor jedem CountCycle Aufwärmpulse hinzugefügt werden. Bei Aufwärmpulsen werden wie bei normalen Pulsen Radiofrequenz vom VFG auf die akustooptischen Modulatoren gesendet, allerdings bei abgeschaltetem Laser.

Um das Aufheizverhalten genauer zu untersuchen, wurde ein Experiment gemäß Abbildung B.1 aufgebaut. Es wurde ein Akustooptischer Modulator geöffnet und ein Thermoelement auf den Kristall mit Thermokleber aufgeklebt, sodass über ein Multimeter die Temperatur an der Kristalloberfläche ausgelesen werden konnte. Das Multimeter war in der Lage, jede halbe Sekunde automatisch einen Messpunkt aufzunehmen. Da eine halbe Sekunde deutlich länger ist als ein CountCycle ($\approx$10 ms) und nicht die Temperatur im inneren des Kristalls gemessen werden konnte, ließen sich so nur Aufheizeffekte auf langen Zeitskalen untersuchen, sodass nur indirekt Rückschlüsse auf die Problematik der Aufwärmpulse eingegangen werden kann. Die Untersuchung von Aufheizeffekten auf langen Zeitskalen ist dennoch von großer Bedeutung.

Um typische experimentelle Pulssequenzen simulieren zu können, wurde eine Radiofrequenz von 82 MHz durch einen Verstärker auf den präparierten akustooptischen Modulator übertragen. Mithilfe eines Reflektometers konnte die Amplitude des Signals vor den Messungen so eingestellt werden, dass eine bestimmte Leistung, in den meisten Experimenten 2 Watt, transmittiert wurde. Dabei fiel auf, dass, als Folge des aufgeklebten Thermoelements ein Teil der Leistung wieder reflektiert wurde (bei 2 Watt wurden 1,1 Watt reflektiert), was möglicherweise ein leicht unterschiedliches Verhalten gegenüber anderen Akustooptischen Modulatoren bedeuten könnte. Dennoch wurden die folgenden Messungen mit diesen Parametern durchgeführt. Zwischen Signalgenerator und Verstärker wurde noch ein Switch angeschlossen, der mit einem Rechteck-Signalgenerator verbunden war. Der Switch blockierte die 82 MHz-Radiofrequenz, wenn sich die Rechteckspannung nicht auf dem hohen Niveau befand - sodass die Radiofrequenz wie bei Pulssequenzen in regelmäßigen Abständen an- und ausgeschaltet werden konnte.

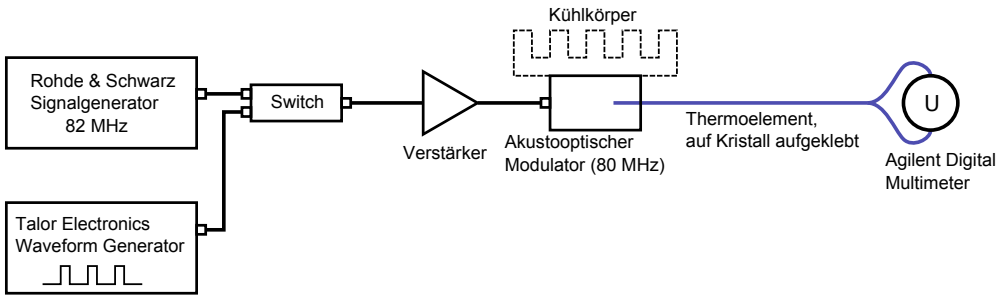


Abbildung B.1.: Skizze des Versuchsaufbaus

Alle Messungen wurden nach folgendem Schema durchgeführt: Vor jeder Messung wurde zunächst gewartet, bis der Akustooptische Modulator auf Labortemperatur abgekühlt war. Für eine bestimmte eingestellte Leistung und Rechteckpuls-Dauer wurden dann 30 Minuten lang Spannungen am Thermoelement aufgezeichnet. Danach wurde die Radiofrequenz abgeschaltet und nochmals 30 Minuten lang Spannungen am Thermoelement gemessen. So wurden Aufwärm- und Abkühlkurven erhalten. Die Temperatur $T(t)$ gehorcht der Differentialgleichung

$$\dot{T}(t) = R\,(T_{\mathrm{eq}}(P) - T(t))$$

mit $T_{\mathrm{eq}}(P)$ einer von der Leistung abhängigen Gleichgewichtstemperatur und R einer Aufheizrate bzw. Abkühlrate. Beim Abkühlen ist die Leistung $P = 0$ und somit entspricht $T_{\mathrm{eq}}(0)$ der Labortemperatur. Beim Aufheizen erhält man somit

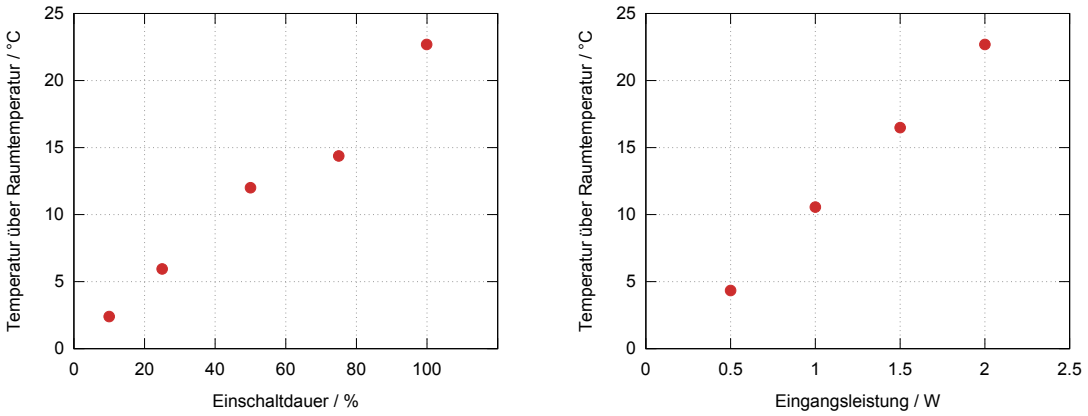
$$T(t)_{\mathrm{heizen}} = T_{\mathrm{eq}}(P) - (T_{\mathrm{eq}}(P) - T_{\mathrm{eq}}(0))\exp(-R \cdot t)$$

und für das Abkühlen

$$T(t)_{\mathrm{kühlen}} = T_{\mathrm{eq}}(0) + (T_{\mathrm{eq}}(P) - T_{\mathrm{eq}}(0))\exp(-R \cdot t)$$

Durch Fits an die Messkurven kann so die Gleichgewichtstemperatur $T_{\mathrm{eq}}(P)$ und die Rate R bestimmt werden. Da die Rate R für alle Messungen ungefähr gleich ist, wird sie im folgenden nicht weiter betrachtet. Alle Messungen dienen lediglich dazu, eine grobe Idee vom Aufheizverhalten des Akustooptischen Modulators zu bekommen - deshalb wurde auf die Berechnung von Fehlern verzichtet.

Zunächst wurde der Signalgenerator für die Rechteckspannung so eingestellt, dass die Radiofrequenz dauerhaft in den Akustooptischen Modulator eingestraht wurde und die Leistung der Radiofrequenz variiert. Das Ergebnis ist eine lineare Abhängigkeit zwischen Gleichgewichtstemperatur und Leistung (Abbildung B.2a). Die Leistung wurde dabei in einem Bereich zwischen 0 Watt und 2 Watt variiert, was in etwa den verwendeten Leistungen im realen Experiment entspricht. In einem zweiten Schritt wurde die Leistung fest auf 2W gesetzt und stattdessen die Einschaltdauer der Radiofrequenz mithilfe des zweiten Signalgenerators geändert. Die Radiofrequenzpulse wurden dabei



(a) (b)

Abbildung B.2.: Gemessene Gleichgewichtstemperaturen (a) gegen unterschiedliche Einschaltdauern der Radiofrequenz bei maximaler Leistung und (b) gegen unterschiedliche Leistungen bei maximaler Einschaltdauer

immer im Abstand von 2 ms gestartet. Hier konnte ebenfalls ein linearer Zusammenhang zwischen Gleichgewichtstemperatur und Einschaltdauer beobachtet werden (Abbildung B.2b). Die Abstände und die Dauern der Pulse waren zu kurz, um mit diesem Experiment Fluktuationen in der Temperatur feststellen zu können. Über einen langen Zeitraum gesehen entspricht das Pulsen der Radiofrequenz einer effektiven Reduzierung der Leistung.

Im nächsten Schritt sollte dann untersucht werden, inwiefern von außen Einfluss auf das Aufheiz- und Abkühlverfahren des Akustooptischen Modulators genommen werden konnte. Es wurde ein großer Kühlkörper mit Wärmeleitpaste auf die metallische Oberfläche des Gehäuses aufgelegt. In weiteren Messungen wurde dann ein Peltierelement zwischen Kühlkörper und Gehäuse gelegt, wobei alle Oberflächen mit Wärmeleitpaste verbunden wurden. Der angelegte Strom durch das Peltierelement bestimmt, ob das Gehäuse geheizt oder gekühlt wird. Die besten Ergebnisse wurden erzielt bei 0,6 A angelegtem Strom (kühlen) bzw. -1,2 A angelegtem Strom (heizen). Abbildung B.3 fasst die Ergebnisse zusammen. Die Anbringung des Kühlkörpers bringt eine deutliche Verbesserung: Nicht nur die Gleichgewichtstemperatur ist deutlich niedriger, es verschwinden auch Fluktuationen auf einer Sekundenzeitskala. Das Peltierelement verschlechtert das Verhalten geringfügig gegenüber der Situation, bei der nur der Kühlkörper angebracht ist. Besonders wenn das Gehäuse geheizt wird, sind die kurzzeitigen Fluktuationen noch besonders ausgeprägt.

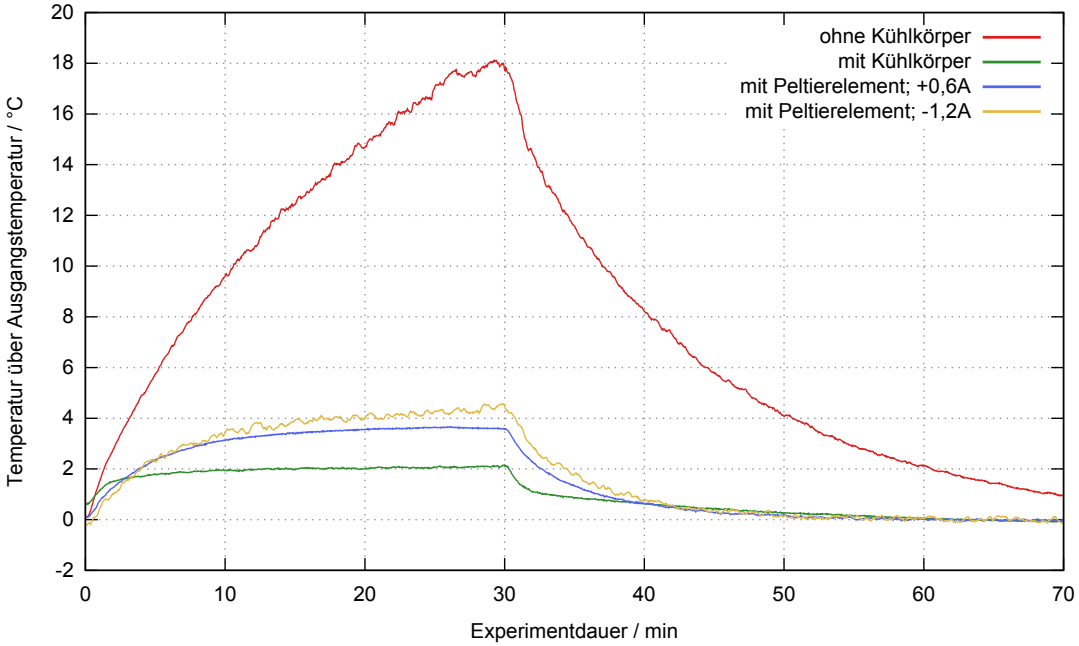


Abbildung B.3.: Aufheiz- und Abkühlkurven für verschiedene Konfigurationen

Literaturverzeichnis

[BBC+95] Barenco, A. ; Bennett, C.H. ; Cleve, R. ; DiVincenzo, D.P. ; Margolus, N. ; Shor, P. ; Sleator, T. ; Smolin, J.A. ; Weinfurter, H.: Elementary gates for quantum computation. In: *Physical Review A* 52 (1995), Nov, Nr. 5, S. 3457–3467

[BGL+12] Bowler, R. ; Gaebler, J. ; Lin, Y. ; Tan, T. R. ; Hanneke, D. ; Jost, J. D. ; Home, J. P. ; Leibfried, D. ; Wineland, D. J.: Coherent Diabatic Ion Transport and Separation in a Multizone Trap Array. In: *Phys. Rev. Lett.* 109 (2012), Aug

[BL01] Boiteux, M. ; Levelut, A.: Semicoherent states. In: *Journal of Physics A: Mathematical, Nuclear and General* 6 (2001), Nr. 5, S. 589

[Bla01] Black, E.D.: An introduction to Pound–Drever–Hall laser frequency stabilization. In: *American Journal of Physics* 69 (2001), S. 79

[BMB+98] Berkeland, DJ ; Miller, JD ; Bergquist, JC ; Itano, WM ; Wineland, DJ: Minimization of ion micromotion in a Paul trap. In: *Journal of applied physics* 83 (1998), Nr. 10, S. 5025–5033

[BOV+09] Blakestad, R. B. ; Ospelkaus, C. ; VanDevender, A. P. ; Amini, J. M. ; Britton, J. ; Leibfried, D. ; Wineland, D.: High fidelity transport of trapped-ion qubits through an X-junction trap array. In: *Physical Review Letters* 102 (2009), S. 153002

[BOV+11] Blakestad, R. B. ; Ospelkaus, C. ; VanDevender, J. H. ; Wesenberg, J. M. J. B. M. J. Biercuk ; Leibfried, D. ; Wineland, D.: Near-ground-state transport of trapped-ion qubits through a multidimensional array. In: *Physical Review A* 84 (2011), S. 032314

[BSWL10] Burrell, AH ; Szwer, DJ ; Webster, SC ; Lucas, DM: Scalable simultaneous multiqubit readout with 99.99% single-shot fidelity. In: *Physical Review A* 81 (2010), Nr. 4

[Bur10] Burrell, Alice: High Fidelity Readout of Trapped Ion Qubits. In: *Doktorarbeit, Exeter College Oxford* (2010)

[CG69] Cahill, KE ; Glauber, RJ: Ordered expansions in boson amplitude operators. In: *Physical Review* 177 (1969), Nr. 5, S. 1857

[CZ95] Cirac, J.I. ; Zoller, P.: Quantum computations with cold trapped ions. In: *Physical Review Letters* 74 (1995), Nr. 20, S. 4091–4094

[Div00] Divincenzo, D.P.: The Physical Implementation of Quantum Computation. In: *Fortschritte der Physik* 48 (2000), S. 771–783

[DJ92] Deutsch, D. ; Jozsa, R.: Rapid solution of problems by quantum computation. In: *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* 439 (1992), Nr. 1907, S. 553–558

[Dub05] Dubash, Manek: Moore's Law is dead, says Gordon Moore. In: *Techworld, abgerufen am 10.12.2012* (2005). http://news.techworld.com/operating-systems/3477/moores-law-is-dead-says-gordon-moore/

[Fey86] Feynman, R.P.: Quantum mechanical computers. In: *Foundations of physics* 16 (1986), Nr. 6, S. 507–531

[Fox11] Fox, Mark: *Quantum Optics, An Introduction*. 2011

[GRB+01] Gulde, S. ; Rotter, D. ; Barton, P. ; Schmidt-Kaler, F. ; Blatt, R. ; Hogervorst, W.: Simple and efficient photo-ionization loading of ions for precision ion-trapping experiments. In: *Applied Physics B: Lasers and Optics* 73 (2001), Nr. 8, S. 861–863

[Gro96] Grover, L.K.: A fast quantum mechanical algorithm for database search. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing* ACM, 1996, S. 212–219

[HDO+98] Hearst, M.A. ; Dumais, ST ; Osman, E. ; Platt, J. ; Scholkopf, B.: Support vector machines. In: *Intelligent Systems and their Applications, IEEE* 13 (1998), Nr. 4, S. 18–28

[HDS+08] Huber, G. ; Deuschle, T. ; Schnitzler, W. ; Reichle, R. ; Singer, K. ; Schmidt-Kaler, F.: Transport of ions in a segmented linear Paul trap in printed-circuit-board technology. In: *New Journal of Physics* 10 (2008), Nr. 1

[HHJ+09] Home, J. P. ; Hanneke, D. ; Jost, J. D. ; Amini, J. M. ; Leibfried, Dietrich ; Wineland, D. J.: Complete Methods Set for Scalable Ion Trap Quantum Information Processing. In: *Science* 325 (2009), S. 1227

[Hub10] Huber, Gerhard: Quantum Thermodynamics with Trapped Ions. In: *Doktorarbeit, Universität Ulm* (2010)

[KMW02] Kielpinski, D. ; Monroe, C. ; Wineland, D.J.: Architecture for a large-scale ion-trap quantum computer. In: *Nature* (2002)

[LBD+05] Lee, PJ ; Brickman, KA ; Deslauriers, L. ; Haljan, PC ; Duan, LM ; Monroe, C.: Phase control of trapped ion quantum gates. In: *Journal of Optics B: Quantum and Semiclassical Optics* 7 (2005), Nr. 10

[LBMW03] Leibfried, D. ; Blatt, R. ; Monroe, C. ; Wineland, D.: Quantum dynamics of single trapped ions. In: *Reviews of Modern Physics* 75 (2003), Nr. 1, S. 281

[LGRS04] Letchumanan, V. ; Gill, P. ; Riis, E. ; Sinclair, AG: Optical Ramsey spectroscopy of a single trapped $^{88}Sr^{+}$ ion. In: *Physical Review A* 70 (2004), Nr. 3

[LJ11] Lau, H.K. ; James, D.F.V.: Decoherence and dephasing errors caused by the dc Stark effect in rapid ion transport. In: *Physical Review A* 83 (2011), Nr. 6

[Mac12] Macha, Tobias: Frequenzstabilisierung eines Titan-Saphir-Lasers und Verbesserung der Kohärenz von Qubits mit Ca^{+}-Ionen. In: *Diplomarbeit, Universität Mainz* (2012)

[Mai06] Maiwald, Robert: Abbildungs- und Steuerungssystem für ein Experiment mit einzelnen Ionen. In: *Diplomarbeit, Universität Ulm* (2006)

[Moo65] Moore, G.E.: Cramming more components onto integrated circuits. In: *Electronics* 38 (1965), Nr. 8

[Moo75] Moore, G.E.: Progress in digital integrated electronics. In: *Electron Devices Meeting, 1975 International* Bd. 21 IEEE, 1975, S. 11–13

[MSW+08] Myerson, AH ; Szwer, DJ ; Webster, SC ; Allcock, DTC ; Curtis, MJ ; Imreh, G. ; Sherman, JA ; Stacey, DN ; Steane, AM ; Lucas, DM: High-fidelity readout of trapped-ion qubits. In: *Physical review letters* 100 (2008), Nr. 20, S. 200502

[OKKB90] Oliveira, F. A. M. ; Kim, M. S. ; Knight, P. L. ; Buek, V.: Properties of displaced number states. In: *Phys. Rev. A* 41 (1990), S. 2645–2652

[Ott12] Ott, Konstantin: Realisierung eines Zwei-Ionen-Quantengatters. In: *Diplomarbeit, Universität Mainz* (2012)

[Pos10] Poschinger, Ulrich: Quantum Optics Experiments in a Microstructured Ion Trap. In: *Doktorarbeit, Universität Ulm* (2010)

[RBKD+02] Rowe, M.A. ; Ben-Kish, A. ; DeMarco, B. ; Leibfried, D. ; Meyer, V. ; Beall, J. ; Britton, J. ; Hughes, J. ; Itano, W.M. ; Jelenkovic, B. ; Langer, C. ; Rosenband, T. ; Wineland, D.J.: Transport of quantum states and seperation of ions in a dual rf ion trap. In: *Quantum Information and Computation* 2 (2002), S. 257

[Roo00] Roos, Christian: Controlling the quantum state of trapped ions. In: *Doktorarbeit, Universität Innsbruck* (2000)

[Sch09] Schulz, Stephan: Scalable Microchip Ion Traps for Quantum Computation. In: *Doktorarbeit, Universität Ulm* (2009)

[Sho97] Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. In: *SIAM Journal on Computing* 26 (1997), Nr. 5, S. 1484–1509

[SPM+10] Singer, K. ; Poschinger, U. ; Murphy, M. ; Ivanov, P. ; Ziesel, F. ; Calarco, T. ; Schmidt-Kaler, F.: Colloquium: Trapped ions as quantum bits: Essential numerical tools. In: *Reviews of Modern Physics* 82 (2010), Nr. 3, S. 2609

[SPSSK06] Schulz, S. ; Poschinger, U. ; Singer, K. ; Schmidt-Kaler, F.: Optimization of segmented linear Paul traps and transport of stored particles. In: *Fortschritte der Physik* 54 (2006), Nr. 8-10, S. 648–665

[Wie11] Wiens, Alex: Detection of Qubit Registers in a Micro Trap. In: *Diplomarbeit, Universität Mainz* (2011)

[WMI+98] Wineland, D.J. ; Monroe, C. ; Itano, WM ; Leibfried, D. ; King, BE ; Meekhof, DM: Experimental issues in coherent quantum-state manipulation of trapped atomic ions. In: *Journal of Research of the National Institute of Standards and Technology* 103 (1998), Nr. 3, S. 259–328

[WPZ+11] Walther, A. ; Poschinger, U. ; Ziesel, F. ; Hettrich, M. ; Wiens, A. ; Welzel, J. ; Schmidt-Kaler, F.: Single ion as a shot-noise-limited magnetic-field-gradient probe. In: *Physical Review A* 83 (2011), Nr. 6, S. 062329

[WZR+12] Walther, A. ; Ziesel, F. ; Ruster, T. ; Dawkins, S.T. ; Ott, K. ; Hettrich, M. ; Singer, K. ; Schmidt-Kaler, F. ; Poschinger, U.: Controlling fast transport of cold trapped ions. In: *Physical Review Letters* 109 (2012), Nr. 8, S. 80501

[ZRW+12] Ziesel, F. ; Ruster, T. ; Walther, A. ; Kaufmann, H. ; Singer, K. ; Schmidt-Kaler, F ; Poschinger, U.: Experimental creation and analysis of displaced number states. In: *arxiv:1211.5490* (2012)

Eidesstattliche Erklärung

Ich versichere, dass ich meine Diplomarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe. Alle Stellen, die wörtlich oder sinngemäß aus anderen Veröffentlichungen entnommen wurden, sind im Text als solche gekennzeichnet.

21. Dezember 2012

Thomas Ruster