Optical Imaging of Rydberg Atoms

by

Anton Mazurenko

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of

Bachelor of Science in Physics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

Author ....
Department of Physics
May 11, 2012

Certified by ....
Vladan Vuletić
Lester Wolfe Professor of Physics
Thesis Supervisor

Accepted by ....
Nergis Mavalvala
Professor of Physics, Senior Thesis Coordinator

Optical Imaging of Rydberg Atoms

by

Anton Mazurenko

Submitted to the Department of Physics
on May 11, 2012, in partial fulfillment of the
requirements for the degree of
Bachelor of Science in Physics

Abstract

We present an experiment exploring electromagnetically induced transparency (EIT) in Rydberg atoms in order to observe optical nonlinearities at the single photon level. ^{87}Rb atoms are trapped and cooled using a magneto-optical trap (MOT) and a far off resonance dipole trap (FORT). Once the system is prepared, a ladder EIT scheme with Rydberg atoms is used to map the photon field onto the ensemble. The powerful dipole interaction between Rydberg atoms allows the system to exhibit many-body quantum mechanical effects. We also describe an imaging method to observe the Rydberg blockade. Last of all, we present a preliminary measurement of EIT in a Rydberg system. In this measurement, the transmission shows sensitivity to the applied photon flux, and exhibits temporal correlations in the photons exiting the EIT medium.

Thesis Supervisor: Vladan Vuletić
Title: Lester Wolfe Professor of Physics

Acknowledgments

In working on this project I have collaborated with a wonderful group of people. I am extremely grateful to Prof. Vladan Vuletić, who gave me the opportunity to join his lab, guided me throughout the work, recommended me for graduate school and gave valuable advice. Working under him has been a pleasure.

Further, I am thankful to the graduate studnets and post-docs I have worked with: Thibault Peyronel, Qiyu Liang, Sebastian Hofferberth and Ofer Firstenberg. All of them have taught me many valuable skills, answered my many questions and were extremely patient. I am especially indebted to Thibault for giving me so many interesting projects to work on and for his willingness to give advice on anything from physics to fine dining.

Last of all, I am grateful to my family and friends for supporting me throughout my studies.

Being part of the MIT community for the past years has been a joy, and I look forward to working with the CUA in the future!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Inducing photon-photon interactions and correlations through atomic media is an important step in studying quantum many-body systems, generating nonclassical states of light and processing quantum computation [17, 12]. One approach to quantum information processing (QIP) is creating quantum networks where localized nodes perform operations while photons carry information between them. For this approach to work, photons have to be effectively forced to interact with each other in a coherent way.

Since direct photon-photon interactions are too weak, the interaction must be mediated by atoms. One such approach is using Rydberg systems - atoms in a highly excited state [1, 12]. The dipole interaction between these systems is strong enough that the atoms can interact strongly over a long range ($\approx 5\mu$m).

The goal of the experiment is to observe optical nonlinearities at the single photon level by mapping the photon field onto strongly interacting Rydberg states. Creation of such a system could be a stepping stone toward single-photon controlled-not (CNOT) gates, an invaluable tool for a universal quantum computer [7]. One of the possible projects is to image single Rydberg excitations in a cold atom cloud. The strength of the interatomic interaction prevents more than one Rydberg state from existing in a particular volume (typically microns across). This thesis describes a method of optically imaging these collective excitations.

This experiment uses laser cooled atom and quantum optics techniques to prepare the cold atom cloud used to mediate photon interactions. Unlike conventional cold atom systems, working with Rydberg excitations presents novel challenges in terms of confinement and lifetime [9]. Chapter 2 will explore the basics of Rydberg atoms and nonlinear optics involved. Chapter 3 will present the theory of atomic cooling and trapping, and demonstrate how this was implemented in practice. Specifically, the laser system that performed measurements on the atoms required good precision and stability. The frequency control scheme is described in detail, it is designed to preserve laser frequency locking and to scan frequencies more quickly than commercial setups. Chapter 4 describes the imaging system designed to observe the Rydberg blockade mechanism. This imaging system has a resolution of 3μm, making it capable of resolving individual excitations.

Chapter 2

Nonlinear Quantum Optics and Rydberg Atoms

2.1 Electromagnetically Induced Transparency

Electromagnetically induced Transparency (EIT) is a phenomenon arising from laser induced coherence leading to interference between excitation paths [10, 28]. Typically, this interference is induced by driving the transition between two states in a three level system and thereby allowing, under certain conditions, light resonant on the other transition to be transmitted without absorption.

EIT schemes require that three levels are chosen. There are three possible configurations, but only one is relevant to this thesis, the so called 'ladder' configuration, shown in Figure 2-1 [28].

The Hamiltonian for this system can be written as:

$$\mathcal{H} = \mathcal{H}_0 + \mathcal{H}_1 \tag{2.1}$$

where

$$\begin{aligned} \mathcal{H}_0 &= \hbar\omega_g \left|g\right\rangle \left\langle g\right| + \hbar\omega_e \left|e\right\rangle \left\langle e\right| + \hbar\omega_n \left|n\right\rangle \left\langle n\right| \\ \mathcal{H}_1 &= -\tfrac{1}{2} \left[\mathcal{E}\mathcal{P}_{ge} e^{-i\nu t} \left|g\right\rangle \left\langle e\right| + \hbar\Omega_\mu e^{-i\phi_\mu} e^{-i\nu_\mu t} \left|e\right\rangle \left\langle n\right|\right] + c.c. \end{aligned} \tag{2.2}$$

here $\hbar\omega_i$ is the energy associated with energy level $|i\rangle$, $\mathcal{E}$ is the applied electric field

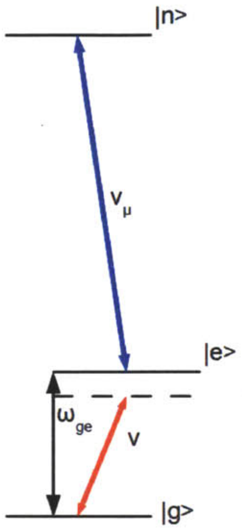


Figure 2-1: Ladder scheme for EIT. The top transition is driven, while the bottom one exhibits transmission near the resonance.

amplitude and $\mathcal{P}_{ge}$ is the matrix element of the dipole moment, ν and ν_μ are the frequencies of the probe and control fields, respectively, and $\Omega_\mu e^{-i\phi_\mu}$ is the complex Rabi frequency associated with the $|e\rangle \to |n\rangle$ transition. The rotating wave approximation (RWA) is assumed. Further, the field ν_μ is taken to be on resonance. Following the treatment of Scully and Zubairy [28], the equation of motion in the density matrix formalism is [28]:

$$\dot{\rho} = -\frac{i}{\hbar}\left[\mathcal{H},\rho\right] - \frac{1}{2}\{\Gamma,\rho\} \tag{2.3}$$

Where ρ is the density matrix and $\Gamma_{nm} = \gamma_n \delta_{nm}$ is the relaxation matrix. In general relaxation is more complicated, but this model is widely applicable. Let us define the decay rates associated with ρ_{ne}, ρ_{eg} and ρ_{ng} as γ_1,γ_2 and γ_3.

The dispersive and absorptive properties of the $|g\rangle \to |e\rangle$ transition are governed by the complex polarization P or the complex susceptibility χ, which are related to ρ_{ge} by [28]:

$$P = 2\mathcal{P}_{ge}\mathcal{E}\rho_{ge}e^{i\nu t} = \frac{\chi\mathcal{E}}{4\pi} \tag{2.4}$$

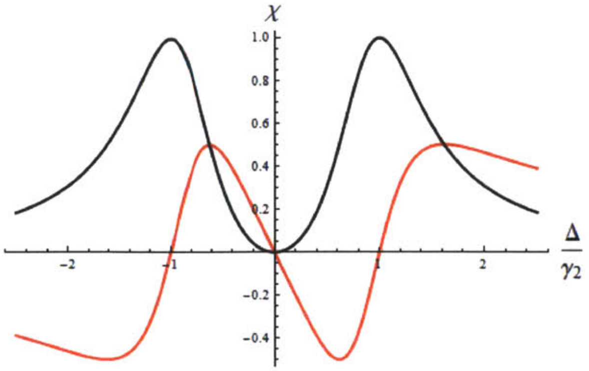


Figure 2-2: Real (red) and imaginary (black) components of χ are shown for reasonable parameters. χ is measured in units of $\frac{4\pi N_a|\mathcal{P}_{ge}|^2}{\hbar}$.

In order to find ρ_{ge}, it is possible to write down a system of differential equations using Eq. 2.3, under the initial condition that $\rho_{gg}(t=0)=0$:

$$\begin{aligned}
\dot{\rho}_{ge} &= -(i\omega_{ge}+\gamma_2)\rho_{ab} - \tfrac{i}{2}\tfrac{\mathcal{P}_{ge}\mathcal{E}}{\hbar}e^{-i\nu t}(\rho_{ee}-\rho_{gg}) + \tfrac{i}{2}\Omega_\mu e^{-i\phi_\mu}e^{-i\nu_\mu t}\rho_{ng} \\
\dot{\rho}_{en} &= -(i\omega_{ng}+\gamma_3)\rho_{ng} - \tfrac{i}{2}\tfrac{\mathcal{P}_{ge}\mathcal{E}}{\hbar}e^{-i\nu t}\rho_{ne} + \tfrac{i}{2}\Omega_\mu e^{-i\phi_\mu}e^{-i\nu_\mu t}\rho_{eg} \\
\dot{\rho}_{gn} &= -(i\omega_{ne}+\gamma_1)\rho_{ne} - \tfrac{i}{2}\Omega_\mu e^{-i\phi_\mu}e^{-i\nu_\mu t}(\rho_{ee}-\rho_{nn}) + \tfrac{i}{2}\tfrac{\mathcal{P}_{ge}\mathcal{E}}{\hbar}e^{-i\nu t}\rho_{ng}
\end{aligned} \tag{2.5}$$

Solving these differential equations for ρ_{ge} and using Eq. 2.4, the susceptibility is found to be [28]:

$$\begin{aligned}
\Re(\chi) &= \tfrac{4\pi N_a|\mathcal{P}_{ge}|^2}{\hbar Z}\left[\Delta\gamma_3(\gamma_2+\gamma_3)+\Delta(\Delta^2-\gamma_2\gamma_3-\Omega_\mu^2/4)\right] \\
\Im(\chi) &= \tfrac{4\pi N_a|\mathcal{P}_{ge}|^2}{\hbar Z}\left[\Delta^2(\gamma_2+\gamma_3)-\gamma_3(\Delta^2-\gamma_2\gamma_3-\Omega_\mu^2/4)\right]
\end{aligned} \tag{2.6}$$

Here the detuning $\Delta = \omega_{ge}-\nu$, N_a is atom density, and and $Z = (\Delta^2-\gamma_2\gamma_3-\Omega_\mu^2/4)^2+\Delta^2(\gamma_2+\gamma_3)^2$. For reasonable parameters, χ is shown in Figure 2-2. The real part of χ corresponds to refraction, and the imaginary part corresponds to absorption. From the figure it is clear that on resonance, the medium is transparent and therefore non-refractive, which means that near resonance, EIT holds.

Property	Expression	n^x	Rb(41d)	Rb(43s)
Level Spacing	$W_n - W_{n-1}$	n^{-3}	107.8 GHz	109.66 GHz
Orbit radius $\langle r \rangle$	$\frac{1}{2}(3n^2 - l(l+1))$	n^2	$2355.46a_0$	$2384.2a_0$
Lifetime (spontaneous)	$\tau = \tau'(n0 - \delta_{n,j,l})^\gamma$	n^3	78μs	99μs
Lifetime (blackbody)	$\tau_{bb} = \frac{3\hbar n^2}{4\alpha^3 k_B T}$	n^{-2}	82μs	90μs
Fine structure splitting	$10.8n^{-3}$ THz	n^{-3}	172MHz	none

Table 2.1: Important properties of particular Rydberg levels.

2.2 Rydberg Atom Properties

Many types of Rydberg atoms are used in various experiments. For this purpose, relatively low-lying Rydberg states are used, with principal quantum number between $n \approx 40$ and $n \approx 100$. The properties of two possible states are given in Table 2.1. Overall, the typical states used are about 125 nm in radius, and with a lifetime of 10's of μs. The important property is the dipole interaction, given by:

$$\Delta W = -\frac{C_6}{R^6} \propto \frac{n^{11}}{R^6} \tag{2.7}$$

where C_6 is the state dependent dipole strength, and R is the interatomic separation. For interactions between two atoms, both in the $46S$ state, $C_6 \approx -5.5\mathrm{GHz}\mu\mathrm{m}^6$. In an ensemble of cold atoms, the first excited state is $1/\sqrt{N}\left[\sum_i^N |g_1, g_2...e_i...g_N\rangle\right]$ [27], which corresponds to a single excitation in the cloud. The energy spectrum in not evenly spaced, however, due to dipole interactions. The second excited state (corresponding to two excitations), will be shifted by energy V_{int}. Therefore, if the cloud is illuminated by a laser of bandwidth $\Delta\nu < V_{int}/h$, then only one excitation will occur in a given volume. The effect of bringing two Rydberg atoms together is seen in Figure 2-3.

2.3 Theory of EIT in Rydberg Atoms

Optical response in a strongly interacting EIT medium has been theoretically explored using Monte-Carlo methods [1] and using analytic methods in certain regimes [12].

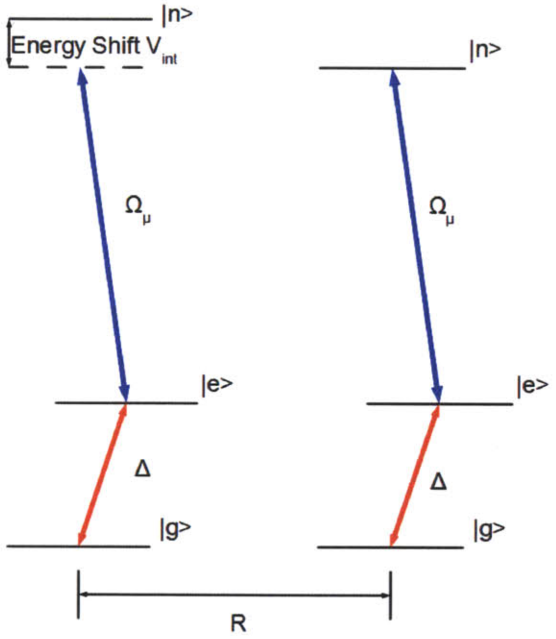


Figure 2-3: Rydberg-Rydberg interactions. The $|e\rangle \rightarrow |n\rangle$ transition is has Rabi frequency Ω and detuning Δ. Bringing two excitations close together shifts the upper levels.

This research has shown EIT in Rydberg atoms to be a viable approach to inducing coherent photon-photon interactions. The qualitative response is similar to the EIT curve derived in Figure 2-2.

The volume of the relevant blockade can be estimated simply by stipulating that the EIT condition will fail if the energy level is shifted by more than the width of the central EIT peak. The width is Ω_μ^2/Γ, where Ω_μ is the control field Rabi frequency, and Γ is the lifetime. The condition for blockade is then:

$$V(z_b) = \frac{\Omega_\mu^2}{\Gamma} = \frac{C_6}{z_b^6} \tag{2.8}$$

which yields the blockade radius:

$$z_b = \left(\frac{\Gamma C_6}{\Omega_\mu^2}\right)^{1/6} \tag{2.9}$$

For the relevant states and parameters, these lengths $z_b \approx 5\mu\text{m}$.

To understand how EIT is affected by dipole-dipole interactions in a Rydberg system, consider a two atoms while neglecting interactions. EIT will be observed because the system will evolve into a dark state [24]:

$$|D\rangle = \frac{\Omega_\mu^2 |gg\rangle - \Omega_p\Omega_\mu(|gn\rangle + |ng\rangle) + \Omega_p^2 |nn\rangle}{\Omega_p^2 + \Omega_\mu^2} \tag{2.10}$$

where Ω_p is the probe Rabi frequency. Accounting for dipole-dipole interactions will suppress the $|nn\rangle$, effectively forcing one of the atoms to act as a two level system [24, 23]. While one atom is in the Rydberg state, which allows EIT, the other will couple to the probe light and attenuate its transmission. This is another way of understanding the blockade effect [24]. The single Rydberg excitation is shared among all the atoms in a particular volume. Since only one atom in many contributes this excitation, EIT will only work for small numbers of photons, leading to a nonlinear transmission that decreases with the number of photons in the probe field [24].

2.3.1 Experimental Results in EIT of Rydberg Atoms

Measurements of EIT in Rydberg atoms have been performed by C.Adams et al [24]. The atoms in their system were trapped in a magneto-optical trap (MOT), which contained, on average, 11 atoms per blockade volume. Probe and control fields were incident on the cloud and the transmitted probe field was measured by a photodetector. This experiment showed clear optical nonlinearities, where transmission depended on the probe's Rabi frequency, although the probe field remained in the classical domain. The transmission observed in this experiment was well explained by a model simulating dipole-dipole interactions between three atoms [24].

The experiment described in this work takes C. Adams' experiment further. The use of a dipole trap increases the number of atoms sharing each Rydberg excitation by about two orders of magnitude. The dipole trap is made into an elongated shape and illuminated with a probe laser of waist less than the blockade diameter, as shown in Figure 2-4. This limits the propagation of the dark state polariton to one dimension. Dark state polaritons are excitations shared by the atoms and the field that propagate with slow group velocity through EIT media [11]. The directional confinement and increased density serve to limit the number of probe photons that can pass through the Rydberg medium under EIT. One of the major consequences is the fact that only single photons will exit the cloud, leading to correlations in time. By using two photodetectors, as shown in Figure 2-4, it is possible to measure these correlations. The beam is evenly split between the two detectors. We can define the correlation function as:

$$g_2(t) = \frac{\langle I_1(t) I_2(0) \rangle}{\langle I_1(0) \rangle \langle I_2(0) \rangle} \tag{2.11}$$

where I_1, I_2 are the intensities at detectors 1,2, respectively. If the photons were uncorrelated, $g_2 = 1$ for Poissonian light, while $g_2(t = 0) = 0$ for an ideal single-photon source.

The greater Figure 2-5 shows a preliminary measurement of EIT in the proposed scheme for various photon flux rates. As predicted, EIT becomes less visible as more photons are applied and the g_2 function rises from zero to one.

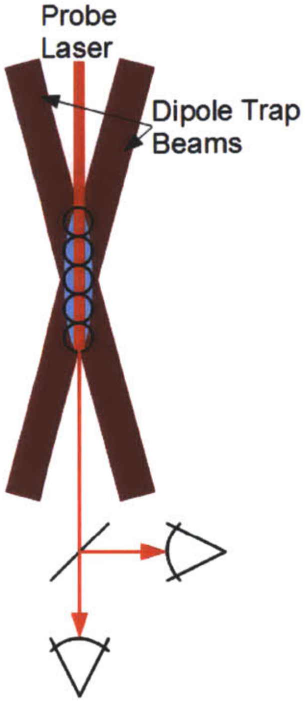


Figure 2-4: Basic experimental setup for the EIT measurement. The trapped cold atom cloud is represented by the blue outline, and the Rydberg excitations by the black circles. The trapping lasers (dark red), probe laser (light red), and detectors are shown. The double detector scheme is used to measure correlations of photons coming out of the cloud.

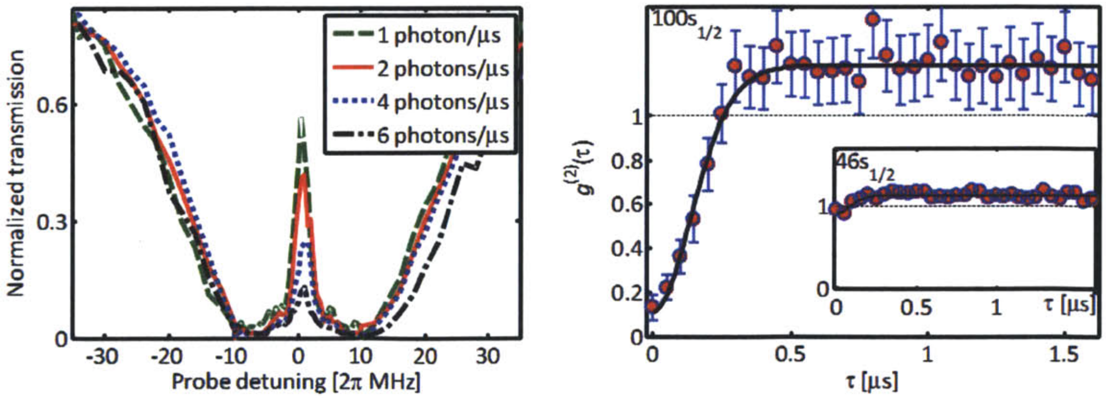


Figure 2-5: Experimental measurement of EIT in a Rydberg ladder scheme, for various incident photon flux rates (left) and the measured g_2 function (right) for states $100S$ and $46S$.

2.3.2 Direct Imaging Of Rydberg Atoms

One of the possible experiments to perform in order to test our understanding of EIT in Rydberg atoms is to directly image the collective Rydberg excitations. We now propose to directly observe the Rydberg atoms by imaging the cloud, and seeing where the EIT condition breaks down due to the Rydberg-Rydberg interactions. The method is similar to that proposed in [19], where impurities (Rydberg atoms) are imaged based on their interaction with the surrounding medium. This approach to imaging Rydberg atoms has also been successfully simulated numerically [15].

The cold atom cloud is prepared by trapping and cooling, and atoms are excited to specific Rydberg levels using a two-photon transition. The cloud is prepared in an elongated shape, allowing the Rydberg atoms to only sit in a one dimensional chain. In contrast to the theoretical discussion of section 2.1, four levels will be involved. A two photon transition is used to create Rydberg excitations in the $|n+1\rangle$ state, where $n = 46S$ or another convenient state. These Rydberg excitations will selectively allow or reject EIT in volumes within the cold atom cloud. To perform EIT, the drive light is applied, connecting a low lying excited state $|e\rangle$ to the Rydberg state $|n\rangle$. The probe laser is incident on the atoms from below and connects the ground state $|g\rangle$ to $|e\rangle$. The level diagram for the experiment is shown in Figure 2-6, and shows how the presence of the atoms in the $|n+1\rangle$ state will cause the EIT effect to fail within a blockade radius by shifting the $|n\rangle$ state. Outside this radius, the EIT effect will be observed. The transmitted probe light is then caught by an imaging system. A picture of the proposed experiment is shown in Figure 2-7. Design of the objective used for this imaging is the central task of this thesis.

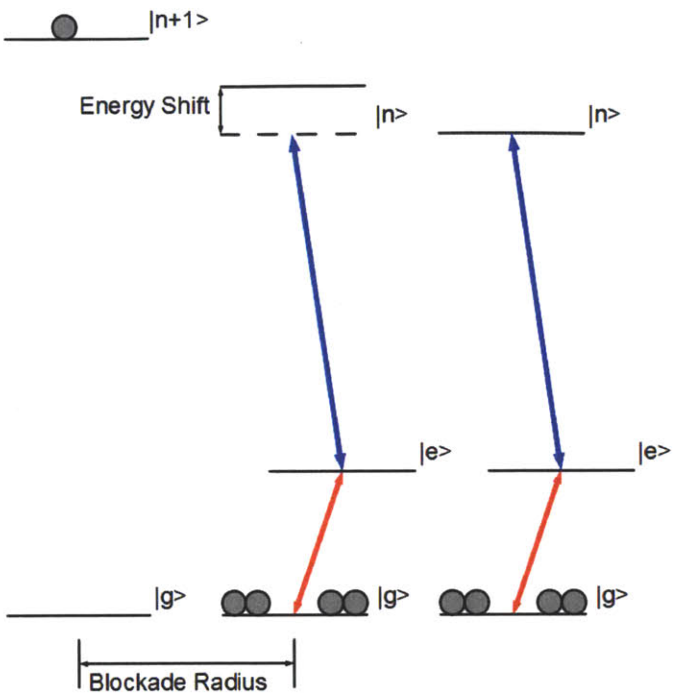


Figure 2-6: Rydberg blockade energy diagram

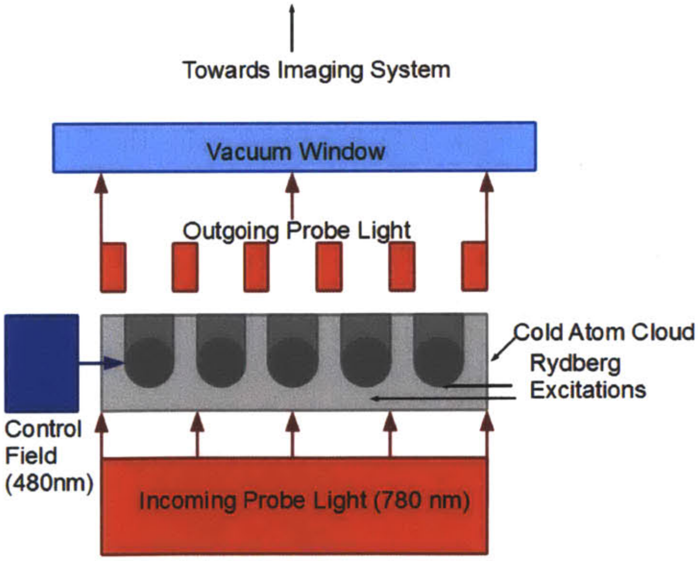


Figure 2-7: Imaging method conceptual diagram. The cigar shaped cloud of cold atoms is illuminated with the probe and control fields. The probe is selectively transmitted in regions where the EIT condition holds.

Chapter 3

Laser Cooling and Trapping Theory and Implementation

3.1 Magneto-Optical Trap

The magneto-optical trap (MOT) is a commonly used method for trapping and cooling atoms by spatially confining them [26, 22]. To illustrate the action of a MOT, consider an atom with ground state $|g\rangle = |J = 0, m = 0\rangle$ and excited states $|e, m_i\rangle = |J = 1, m_i\rangle$. Under a spatially varying magnetic $B(\mathbf{r})$ field, the atoms will have excited energy levels:

$$E_i = g\mu_B B(\mathbf{r}) m_i \tag{3.1}$$

where g is the g-factor, and m_i is the magnetic quantum number. If the field contains a strong gradient passing through zero, then it can be used to trap atoms by adding a light field. Conservation of momentum implies that if $\sigma^{\pm}$ light is absorbed by an atom, the atom will get a 'kick' of $\hbar|\mathbf{k}|$, where $\mathbf{k}$ is the wavevector. If the applied light is red detuned and σ^{+} and σ^{-} beams propagate in opposite directions, as shown in Figure 3-1, atoms will preferentially absorb light moving towards the origin, receiving a 'kick' in that direction. The same is true for the other side. This type of field can be created by a pair of coils running current in opposite directions. Figure 3-2 shows a 3D MOT in action. The MOT is limited to cooling atoms to the Doppler limit,

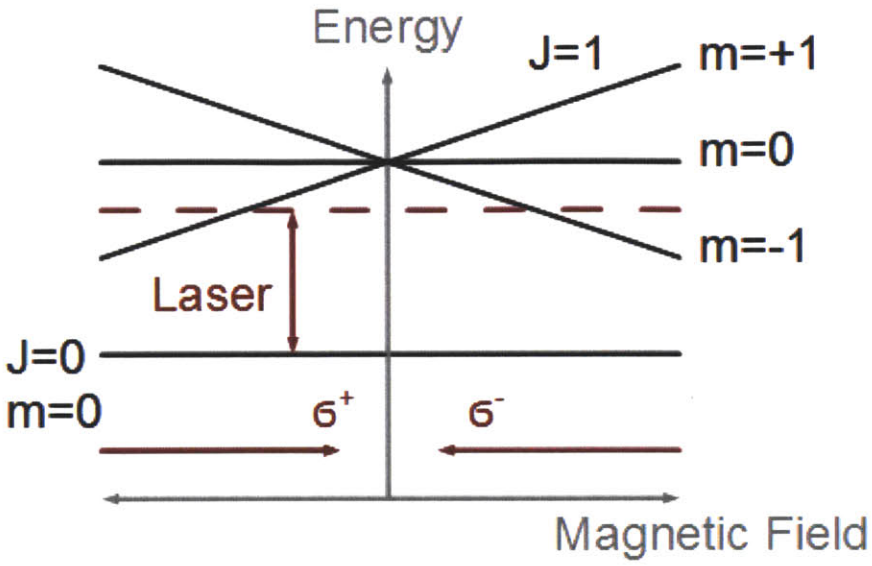


Figure 3-1: Toy energy level diagram illustrating the use of a MOT.

dictated by the natural linewidth Γ of the atoms [22]:

$$T_D = \frac{\hbar\Gamma}{2k_B} \approx 40\mu\text{K} \tag{3.2}$$

In practice, MOT's are more complicated than this model. The experimental MOT works between the $5S_{1/2}(F = 2)$ and $5P_{3/2}(F = 3)$ levels. This level scheme gives good performance, but suffers from optical pumping. Therefore, a repumper laser is used to couple the $5S_{1/2}(F = 1)$ and $5P_{3/2}(F = 3)$ states, so as to repump the atoms from the $(F = 1)$ hyperfine state into the states involved in the MOT transition.

The cooling scheme does not end with the stationary MOT, however. When the MOT is created, the magnetic field gradient at the zero point is about 10 Gauss/cm. When dispensers inject rubidium into the environment, this MOT captures about 10^6 atoms. The gradient is then ramped up to about 35 Gauss/cm, compressing the MOT and improving the phase space density and optical depth of the cloud.

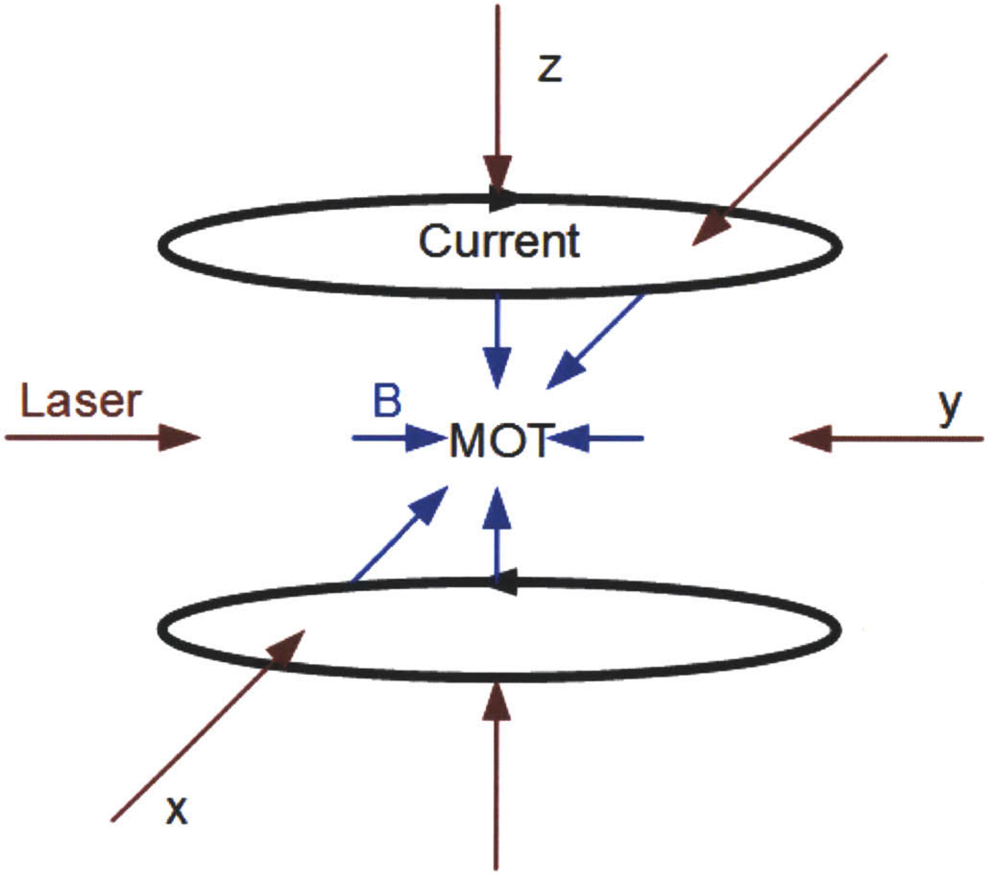


Figure 3-2: Diagram illustrating the field created by the anti-aligned coils and its interaction with laser light.

3.1.1 Optical Molasses

While the MOT is an excellent tool for the confinement of atoms, the magnetic field is not necessary for the cooling of atoms. In fact, the setup described in Figure 3-2 is considered without the field, atoms traveling in a particular direction will preferentially absorb photons traveling the opposite way, causing the atom to slow down. This effect is called optical molasses, and is the last part of the cooling stage. The advantage of optical molasses is that while it does not spatially confine the atoms, it does cool them below the Doppler limit, to close to the recoil limit [8]:

$$T_R = \frac{\hbar^2 \left|\mathbf{k}\right|^2}{2mk_B} \tag{3.3}$$

where m is the mass of the species. In the present experiment, the molasses cooling stage allows the temperature to be lowered to $T \approx 10\mu\text{K}$.

3.1.2 Magnetic Field Generation

The MOT requires a specific set of coils, in order to provide the required gradient and minimum. The coil system is divided into two parts, a large pair of coils providing the gradient, and three smaller pairs of coils designed for fine adjustments, to compensate the field and shift the MOT.

MOT Fields

The MOT fields are generated by two coils in an anti-Helmholtz configuration, where the separation is equal to the radius. Each coil is made up of ten layers of nine turns of 0.1 inch sided square copper wire, wrapped about a cylinder of radius 3.175 inches. The coils provide a gradient of .97 Gauss/cm/A, and have a resistance of 2 Ω.

Bias Fields

The bias fields are designed to control the MOT position and to subtract out the ambient field of the earth. The field in each direction is provided by a pair of coils. The

Orientation	Field (Gauss/A/100 Turns)	Resistance (Ohm)	Number of turns
X	2.1	4.28	100
Y	1.7	5.49	125
Z	4.3	5.4	100
Orientation	Geometry	Separation (in)	Side or Radius (in)
X	Square	9.8	6.00
Y	Square	10.8	6.00
Z	Circle	7.66	3.175

Table 3.1: Bias coil pair properties.

coils are designed to provide a uniform field (inhomogeneous effects can be neglected over the MOT size). The properties are summarized in table 3.1

The bias fields are controlled by a bi-polar controller using a feedback system to control the current in the coils [6]. The controller is controlled by an analog signal from a Pulse-Blaster card programmed by the computer.

3.2 Dipole Trap

Optical dipole traps are one of the most efficient ways to confine atoms [14]. In the present experiment, the atoms captured by the compressed MOT are loaded into a dipole trap.

Dipole traps exploit the atom's dipole interaction with far-detuned light, and are thus one of the weakest types of trapping potentials, exhibiting typical depths of less than 1 mK [14]. For this reason they are used in the later stages of atomic cooling. They have the advantages of low optical excitation and insensitivity to particular sublevels.

These traps rely on atomic polarizability in order to hold the atom. Assuming the atoms are under the applied electric field [14]:

$$\mathbf{E}(\mathbf{r},t) = \hat{\mathbf{e}}E(r)e^{-i\omega t} + c.c. \tag{3.4}$$

where $\hat{\mathbf{e}}$ is the polarization. The atoms gain a dipole moment of:

$$\mathbf{p}(\mathbf{r},t) = \hat{\mathbf{e}}\alpha E(r)e^{-i\omega t} + c.c. \tag{3.5}$$

where α is the atomic polarizability - a complex quantity that defines the amplitude and phase shift of the resulting oscillation.

Calculating the Polarizability

The simplest approach to calculating α is a semi-classical calculation utilizing the oscillator model. This model approximates the electron as a harmonic oscillator of resonant frequency ω_0, being driven with a force $E(t)e/c$. Thus, the position of the electron x satisfies [14]:

$$\frac{d^2x}{dt^2} + \frac{dx}{dt}\Gamma_\omega + \omega_0 x = E(t)e/c \tag{3.6}$$

The damping term Γ_ω comes from the fact that the electron emits dipole radiation. The Larmor formula, which gives the power emitted by an oscillating point source [16] is $\frac{2e^2a^2}{3c^3}$, where a is the instantaneous acceleration. Therefore, the damping term is:

$$\Gamma = \frac{2e^2\omega^2}{3mc^3} \tag{3.7}$$

For convenience, we introduce $\Gamma = \Gamma_\omega(\omega_0)$, leading to a polarization of:

$$\alpha = \frac{3}{2}\frac{c^3\Gamma/\omega_0^2}{\omega_0^2 - \omega^2 - i\left(\omega^3/\omega_0^2\right)\Gamma} \tag{3.8}$$

The classical model is useful enough to give the right answer, but not determine Γ, the natural decay rate. A more complete quantum treatment shows dependence on the dipole element between the two states $|e\rangle$ and $|g\rangle$, and yields [14, 13]:

$$\Gamma_{sc} = \frac{4\omega_0^3}{3\hbar c^3}\left|\langle e|\, ex\,|g\rangle\right|^2 \tag{3.9}$$

Dipole Trap Properties

There are two important parameters that govern a dipole trap - depth and scattering rate. The interaction energy, accounting for the fact that the moment is induced and assuming a spatially varying intensity, is given by [14]:

$$U_{int} = -\frac{1}{2}\langle exE(t)\rangle = \frac{2\pi}{c}\Re(\alpha)I = -\frac{3\pi c^2\Gamma}{2\omega_0^3}\left(\frac{1}{\omega_0-\omega}+\frac{1}{\omega_0+\omega}\right)I(\mathbf{r}) \tag{3.10}$$

where I is the applied field intensity. From this equation, it is clear that the atoms will seek out high fields if the detuning, δ, is negative. By similar logic, the absorbed power is given by:

$$P_{abs} = \left\langle e\frac{dx}{dt}E(t)\right\rangle = \frac{4\pi\omega}{c}\Im(\alpha)I \tag{3.11}$$

The scattering rate can be calculated from the power to be [14]:

$$\Gamma_{sc} = \frac{P_{abs}}{\hbar\omega} = \frac{3\pi c^2\Gamma}{2\hbar\omega_0^3}\frac{\omega^3}{\omega_0^3}\left(\frac{1}{\omega_0-\omega}+\frac{1}{\omega_0+\omega}\right)^2 I(\mathbf{r}) \tag{3.12}$$

In the widely used rotating wave approximation (RWA), the $1/\delta$ term is assumed to be much greater than the $1/(\omega_0+\omega)$ term. Clearly, the scattering rate will limit the lifetime of the atoms in the trap. It is useful to observe that the trap depth varies as $U_{int} \approx -1/\delta$ and $\Gamma_s \approx 1/\delta^2$. Therefore, practical dipole traps use a large detuning at high power to provide low temperature trapping and cooling at a low scattering rate. The extreme version of such a trap is termed the Far Off Resonant Trap (FORT), and is implemented in the experiment.

3.3 Practical Optical Dipole Traps and Optical Lattices

Using the formalism derived in the previous section, it is possible to make interesting trap geometries for various purposes. One of the possibilities is the optical lattice, a periodic potential that keeps every atom localized to less than an optical wavelength within a larger cloud [13]. Optical lattices can be understood using the tools of section 3.2: a standing light wave is created using a retroreflected, Gaussian, laser beam, creating local maxima in the field, spaced by $\frac{1}{2}\lambda$ in the direction of propagation. The atoms will then seek out these maxima. To increase spacing, the beams can intersect at a smaller angle.

The optical lattice and dipole trap setup used by the experiment is shown in figure 3-3. It is not always advantageous to form an optical lattice. In particular, the high trapping frequencies of the lattice limit the possibilities for modulating the trap. For this reason, in the current implementation, a waveplate is placed in the beam path before the beam is retroreflected, which prevents the lattice interference pattern from forming. In this scenario, the trap resembles a cigar shaped pit, shown in Figure 3-4.

Loading the atoms captured by the MOT and cooled by optical molasses is the last stage of the trapping phase. Typically, the FORT retains about 10^4 atoms. The actual experiment, however, does not take place in the FORT. Looking at Figure 3-4, it is clear that atoms in different positions of the FORT will suffer a different light shift, introducing an inhomogeneous broadening to any spectroscopy done on the atoms. Therefore, after the confinement is finished the dipole trap is turned off and the EIT portion of the experiment proceeds. Alternatively, the trap can be modulated to turn on and off, with the experiment taking place while the trap is off.

3.4 Trapping Rydberg Atoms

FORTs are an excellent tool for conventional cold atom physics, but they have a serious issue in the present experiment - Rydberg atoms are repelled from the trap. This effect is caused by the fact that electrons in Rydberg atoms are nearly free. A free electron in an oscillating electric field of $\mathbf{E}(t) = \mathcal{E}\cos\omega t$ will have a time averaged kinetic energy, known as the ponderomotive energy [9]:

$$E_{pond.} = \frac{e^2 \left|\mathcal{E}\right|^2}{4m_e\omega^2} \tag{3.13}$$

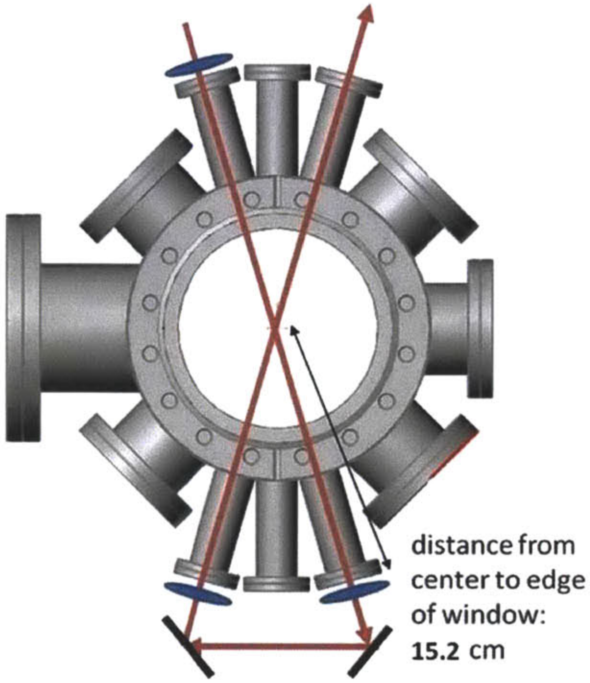


Figure 3-3: Dipole trap geometry of the experiment.

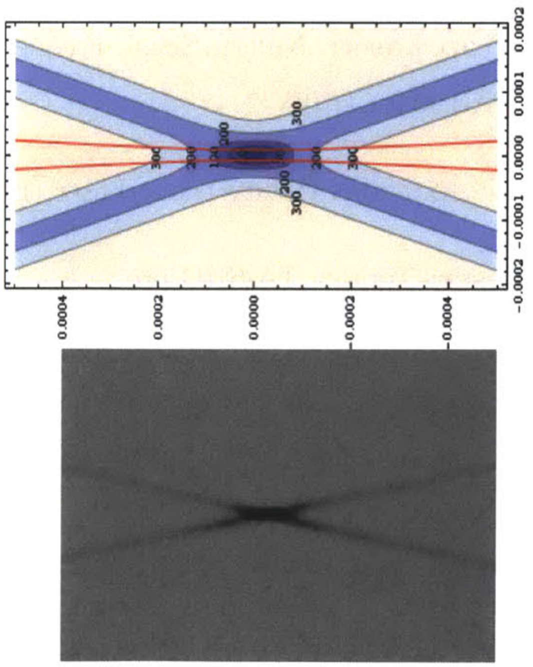


Figure 3-4: Theoretical calculation and experimental picture of the dipole trap. All lengths are in meters and surfaces of equal depth are marked. The probe laser outline is marked in red.

where m_e is the electron mass and $-e$ is the electron charge. Thus, Rydberg atoms in a dipole trap will be ejected because the high field region which traps the cold atoms will serve as a potential peak for the Rydberg excitations. For this reason, once the cold atom cloud is prepared, the dipole trap is shut off, and the experiment is performed without a trap.

3.5 Lasers

The manipulation and measurement of the atoms is done primarily through laser light. The lasers used for various purposes are:

- **MOT Laser:** Vitaly Extended Cavity Diode Laser (ECDL) with a tapered amplifier
- **780nm Repump Laser:** Eagleyard DFB laser, EYP-DFB-0780-00080-1500-SOT02-0000
- **1064nm Dipole Trap Laser:** Nufern Single Frequency Polarization Maintaining Amplifier:SFA-PM1064-10W-2
- **Probe Laser:** Vitaly Extended Cavity Diode Laser (ECDL)
- **480nm Control Laser:** Toptica TA-SHG pro
- **Reference Laser:** Vitaly Extended Cavity Diode Laser (ECDL)

The laser frequencies are graphically illustrated in Figure 3-5.

3.5.1 Laser Frequency Stabilization

As with many experiments in atomic physics, control of the laser with a resolution of about an atomic linewidth ($\approx$ 1 MHz) is desirable. For this particular experiment, the goal is to produce stable, tunable sources of 780 nm light and 480 nm light. Such frequency stability is desired in the probe and control lasers which serve to drive the

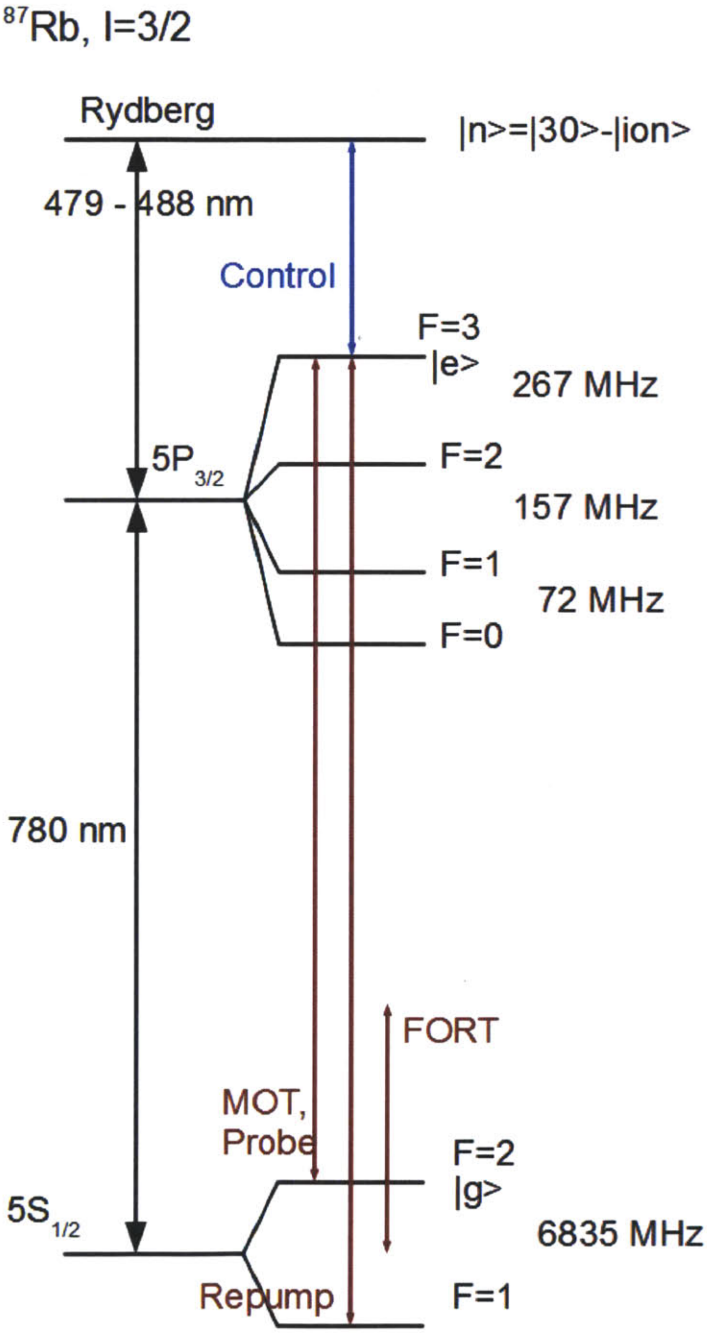


Figure 3-5: Full level diagram of ^{87}Rb, with the relevant lasers marked.

EIT scheme. Predictably, these lasers are not stable enough 'out of the box', and require that the frequency be stabilized. These lasers, however, are tunable, and with proper feedback their linewidth can be reduced.

The laser scheme is shown in Figure 3-6. There are two lasers that need to be stabilized and tuned, so each laser is passed through an electro-optic modulator (EOM), and the first sideband is locked to a cavity using the Pound Drever Hall method (PDH). This method is detailed in section 3.5.2, but simply serves to narrow the laser linewidth by comparing it it to the cavity resonant frequency. When the EOM modulation is adjusted, the first sideband light is compared to the cavity and the feedback shifts the laser to bring the first sideband light back to its original frequency, pushing the main carrier in the opposite direction. Tuning the EOM therefore independently adjusts the laser frequency, and is detailed in section 3.5.4.

Since the laser frequency is locked to the cavity length, thermal fluctuations will result in slow frequency drifts. There are two possible solutions to this problem. The first method is to use a very stable cavity, typically mounted in a vacuum chamber. The other method, that is used in this case, is to actively stabilize the cavity. Beating a reference laser frequency (generated with the Doppler-free DAVLL spectroscopy method) with the experiment laser gives a comparison between their frequencies. The cavity can then be actively stabilized to maintain the frequency relationship, and compensate for environmental distortions. This active feedback is < 1Hz, since the cavity drifts over a longer timescale.

3.5.2 Implementation of the Pound Drever Hall Method (PDH)

Cavity Physics

Since the system is heavily reliant on a cavity, it is important to understand its properties. The primary resonator is a confocal cavity of 5 cm length. The confocal cavity can be qualitatively understood in terms of a simple, planar, Fabry-Perot cavity, that is, two parallel mirrors with some absorption and transmission. The respective geometries are shown in Figure 3-7.

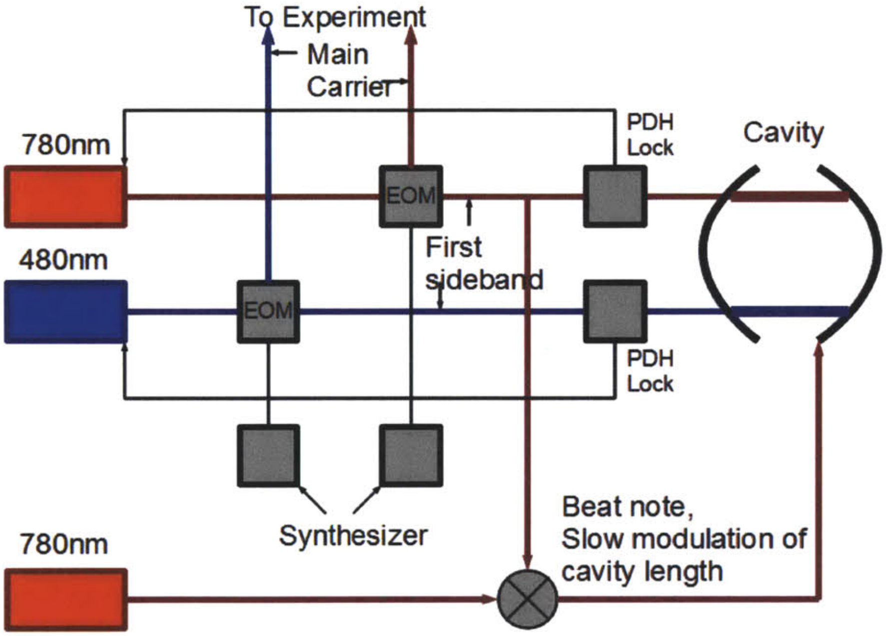


Figure 3-6: The current laser locking scheme for the control and probe fields. The first sidebands of the 780nm and 480nm lasers are locked to a cavity using the PDH method (detailed in section 3.5.2). Modulating these sidebands with an EOM while maintaining locking causes the carrier to be modulated the opposite way, allowing precise scanning of the laser frequency. The cavity is locked to a reference laser using a beat note, where the reference is generated through Doppler-free DAVLL spectroscopy (section 3.5.3).

The effects of a cavity are typically measured by observing transmitted and reflected light, where R, T are the reflectivity and transmission of the individual mirrors. Clearly, certain frequencies will add constructively or destructively. Since the absorption and reflection of the cavities are typically high, the light makes many round trips, which results in a frequency dependent transmission and reflection, due to interference[29].

The transmission of a Fabry-Perot cavity is shown in Figure 3-8, and is given by [29]:

$$T = \frac{1}{1 + \frac{4R}{(1-R)^2}\sin^2(\delta/2)} \tag{3.14}$$

where the reflectivity is R, the optical path $\delta = \frac{2\pi}{\lambda}2nl$, assuming refractive index n, spacing l and wavelength λ. An immediately useful parameter is called the free spectral range (FSR), the spacing between adjacent transmission peaks:

$$\nu_{FSR} = \frac{c}{2nl} \tag{3.15}$$

Another useful parameter in dealing with cavities is the finesse:

$$F = \pi\frac{\sqrt{R}}{1-R} \tag{3.16}$$

Figure 3-8, which shows transmission against freqeuency, illustrates several important points. Qualitatively, it is clear that higher finesse (better mirrors), results in narrower peaks. Further, within a single FSR, the cavity acts as a very good 'ruler' for the frequency, which is why cavities are so frequently used to stabilize frequencies.

Pound-Drever-Hall Method

The Pound Drever Hall Method (PDH) is a method for stabilizing the frequency of a laser. First, the laser is assumed to have some sort of built in tunability, usually implemented by a movable grating which is used to adjust the frequency within the resonator. However, environmental and thermal effects can cause this frequency to drift, requiring active feedback.

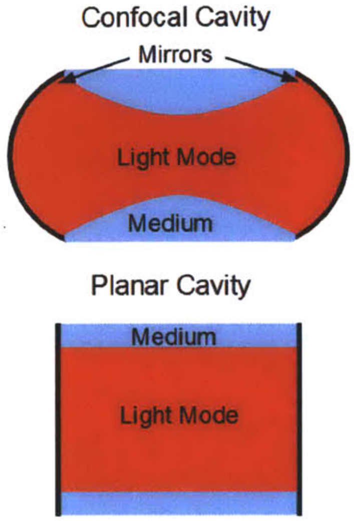


Figure 3-7: Two types of Fabry-Perot cavities, confocal and planar.

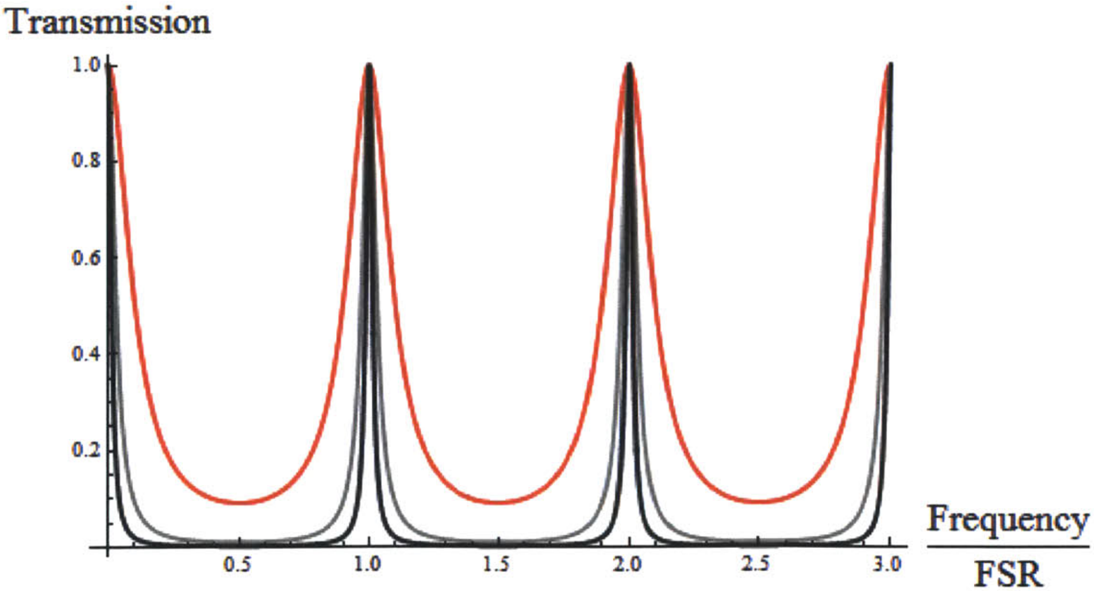


Figure 3-8: Transmission through a Fabry Perot Cavity as a function of frequency for finesse 10 (Red), 100 (Black), 500 (Gray)

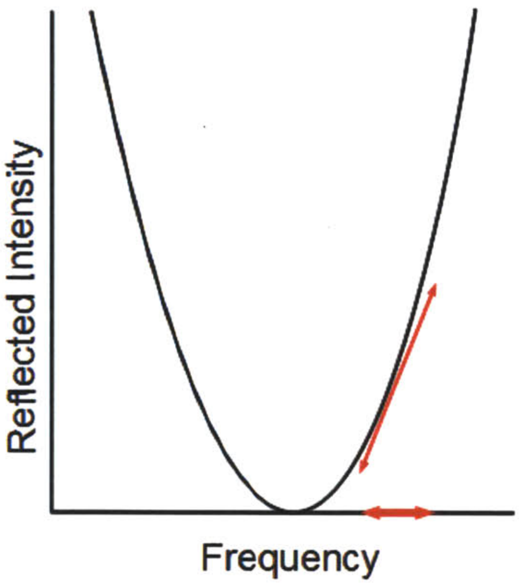


Figure 3-9: Reflection of a cavity near the minimum (resonance). By dithering the applied frequency, it is possible to determine on which side of the minimum the laser frequency is.

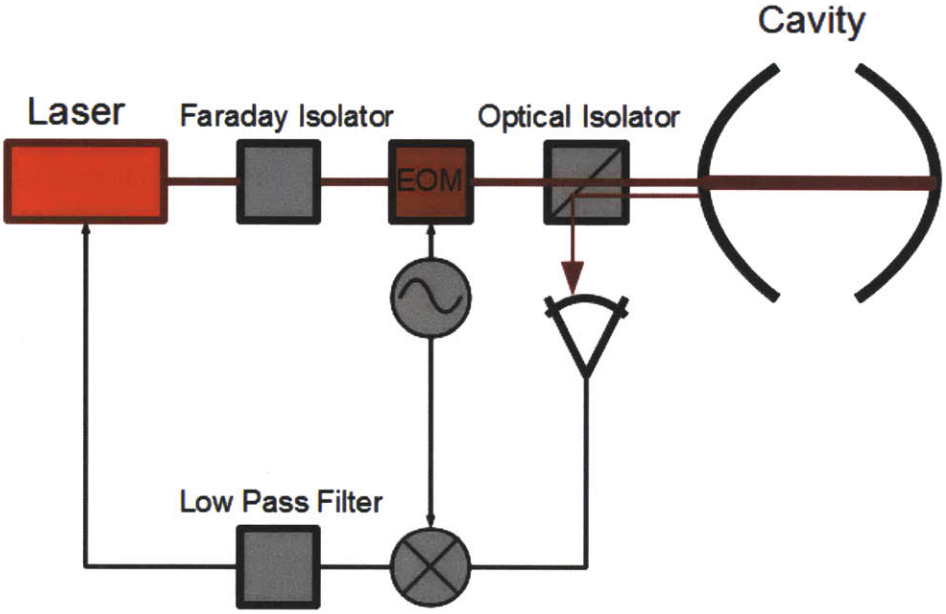


Figure 3-10: Illustration of the PDH method of laser locking

As suggested in section 3.5.2: Cavity Physics, the resonant cavity is a perfect candidate to be used to create an error signal for feedback [2, 29]. The error signal is simply a measure of how far from a reference frequency the laser has drifted. Figure 3-9, shows the reflection of the cavity near a minimum - an approximately parabolic shape, and the reference is defined to be the frequency corresponding to minimal reflection. However, just by measuring the reflection of the cavity, it is impossible to tell which side of the resonance the laser frequency is on. To solve this problem, it is sufficient to observe that while the reflectivity $R_{cav}(\nu)$ is even about the minimum, its derivative, $R'_{cav}(\nu)$ is odd, meaning that $R'_{cav}(\nu)$ would be a good error signal.

To find this derivative, the laser frequency is modulated sinusoidally [3]. When this is done, if the frequency $f > f_0$ where f_0 is the reference, the reflection varies sinusoidally with a phase shift of zero and if $f < f_0$, the phase shift is π. The amplitude of the response is proportional to the error. This is illustrated in Figure 3-9.

The actual implementation is relatively straightforward, and illustrated in Figure 3-10. The laser is first passed through a Faraday isolator, to prevent retroreflection. Then it is modulated by an Electro-Optic Modulator (EOM), which is controlled by an oscillator, and sent to the cavity. The reflected beam is split off into a photodetector. The photodetector signal is mixed with the oscillator signal, resulting in beating phenomena. The low frequency component is then isolated and used as an error signal[3].

The applied, dithered, field of maximal intensity E_0 can be expressed as [3]:

$$\begin{aligned} E_inc &= E_0 e^{i(\omega t+\beta \sin \Omega t)} \\ &\approx E_0[J_0(\beta)e^{i\omega t} + J_1(\beta)e^{i(\omega+\Omega)t} - J_1(\beta)e^{i(\omega-\Omega)t}] \end{aligned} \tag{3.17}$$

Where J_i is the i'th Bessel function, Ω is the modulation frequency, and β is the 'modulation depth', assumed $\beta < 1$. This equation clearly decomposes the applied field into a carrier frequency ω and two sidebands of frequency $\omega \pm \Omega$. For the remaining analysis, it is useful to define the powers contained in the carrier and side bands [3]:

$$\begin{aligned} P_0 &= |E_0|^2 \\ P_c &= J_0^2(\beta)P_0 \\ P_s &= J_1^2(\beta)P_0 \end{aligned} \tag{3.18}$$

For the assumed case of weak modulation (small β), $P_0 \approx P_c + P_s$. By summing the waves reflected by individual trips through the cavity, it is possible to determine the phase relationship between the incident and reflected wave [3].

$$F(\omega) = \frac{E_{ref}}{E_{inc}} = \frac{r\left(e^{i\omega/\nu_{FSR}} - 1\right)}{\left(1 - r^2 e^{i\omega/\nu_{FSR}}\right)} \tag{3.19}$$

Combining equations (3.17) and (3.19) yields the reflected wave [3]:

$$E_{ref} = E_0 \left[F(\omega)J_0(\beta)e^{i\omega t} + F(\omega+\Omega)J_1(\beta)e^{i(\omega+\Omega)t} - F(\omega-\Omega)J_1(\beta)e^{i(\omega-\Omega)t}\right] \tag{3.20}$$

Meaning that the reflected power is [3]:

$$
\begin{aligned}
P_{ref} \;=\;& P_c\left|F(\omega)\right|^2 + P_s(\left|F(\omega+\Omega)\right|^2 + \left|F(\omega-\Omega)\right|^2) + 2\sqrt{P_cP_s}(\Re[F(\omega)\overline{F}(\omega+\Omega) \\
& -\overline{F}(\omega)F(\omega-\Omega)]\cos(\Omega t) + \Im[F(\omega)\overline{F}(\omega+\Omega) - \overline{F}(\omega)F(\omega-\Omega)]\sin(\Omega t) \\
& +(2\Omega \text{ terms}))
\end{aligned}
\tag{3.21}
$$

The 2Ω terms will turn out to never be important and so they are not calculated explicitly.

Equation (3.21) gives a full expression for the reflected power, but is very difficult to interpret physically. There are two regimes which simplify the expression considerably, and allow a practical error signal to be calculated. The simple regime is one of very low frequency dithering. Under slow dithering:

$$F(\omega)\overline{F}(\omega+\Omega) - \overline{F}(\omega)F(\omega-\Omega) \approx \frac{d\left|F\right|^2}{d\omega}\Omega \tag{3.22}$$

Assuming that $\sqrt{P_cP_s} \approx P_0\beta/2$, the reflected power will vary as:

$$P_ref \approx (\text{constants}) + P_0\frac{d\left|F\right|^2}{d\omega}\Omega\beta\cos(\Omega t) + (2\Omega \text{ terms}) \tag{3.23}$$

The mixing and low pass filter will eliminate everything except the $\cos(\Omega t)$ terms, meaning that the error signal will be given by [3]:

$$\epsilon = 2\sqrt{P_cP_s}\frac{d\left|F\right|^2}{d\omega}\Omega \tag{3.24}$$

This error signal is shown in Figure 3-11. This curve clearly has the desired shape about the resonance, showing that the method works, at least in this regime. In practice, however, the dithering is done very fast, so the carrier is near the resonance and the sidebands are far enough to assume that they are fully reflected. Thus, the error signal is drawn mostly from the *Im* part of equation (3.21) [3].

$$\epsilon = -2\sqrt{P_cP_s}\Im[F(\omega)\overline{F}(\omega+\Omega) - \overline{F}(\omega)F(\omega-\Omega)] \tag{3.25}$$

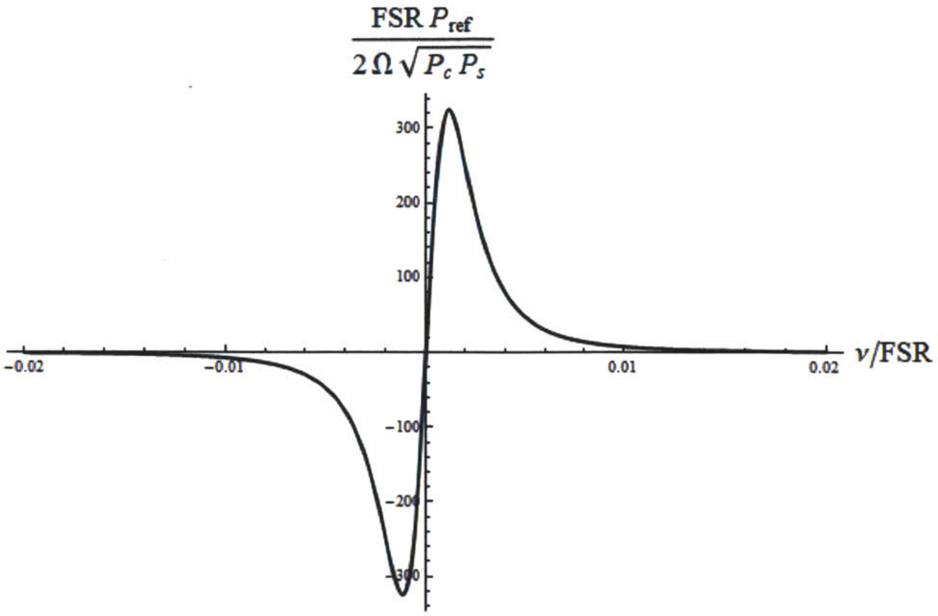


Figure 3-11: Error signal of a PDH setup at slow dithering. Finesse $\approx 10^3$.

The error is shown in Figure 3-12. Its shape also exhibits the desired property around the resonance, allowing effective stabilization of the laser.

3.5.3 Doppler Free Dichroic Atomic Vapor Laser Lock (DAVLL) and Reference Laser Lock

The PDH method locks the laser frequency to a stable value, but this value is itself undefined with respect to atomic lines. In order to reference the laser to a meaningful frequency, the cavity must be locked as well, which requires a narrow laser signal at a known frequency. One way to generate this is through Doppler-free Dichroic Atomic Vapor Laser Lock (DAVLL) [21].

DAVLL

The basic idea is that laser light is passed through a dichroic material (the absorption lines for σ^+ and σ^- light are different), and the measured absorption lineshapes are used to create an error signal by subtracting them from one another. This signal is further narrowed by eliminating the Doppler broadening of the system. Figure 3-13 shows the results for the Doppler-broadened case (a), and for the Doppler-free case,

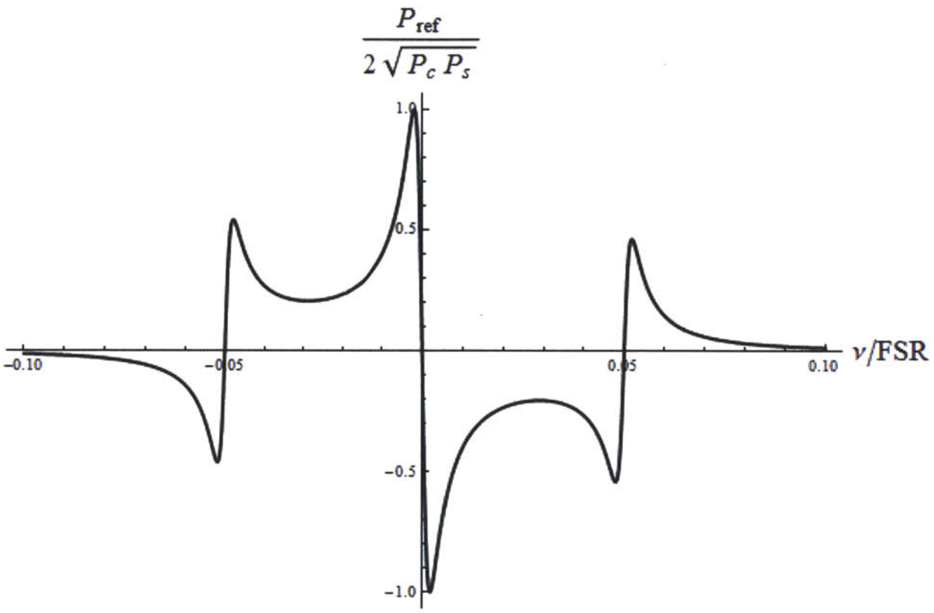


Figure 3-12: Error signal of a PDH setup at fast dithering. Finesse $\approx 10^3$.

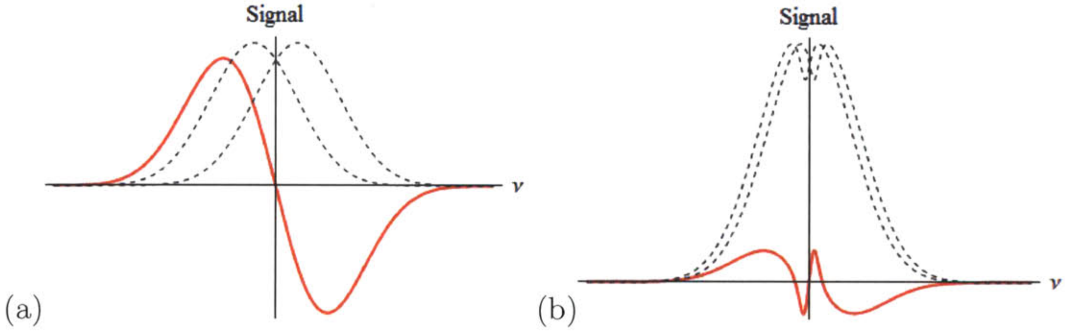


Figure 3-13: Construction of an error signal $S1 - S2$ (red) from the individual absorption lines $S1, S2$ (gray) of $\sigma^{\pm}$ light is shown for the Doppler-broadened case (a), and the Doppler-free case (b).

achieved by saturation absorption spectroscopy [18, 21].

The basic setup is shown in Figure 3-14, where linearly polarized light is divided into two beams, a weak 'probe' and strong 'pump'. The probe is passed through the gas under a magnetic field parallel to the beam axis. The field induces a dichroism via the Zeeman effect, causing the absorption line for the σ^+ and σ^- light to be shifted by equal and opposite amounts. The pump is sent through the gas in the opposite direction, as in standard saturation absorption spectroscopy [18, 21]. The probe is then decomposed into its σ^+ and σ^- polarizations and these beams are sent to detectors. Subtracting the two signals gives the result shown in Figure 3-13-(b).

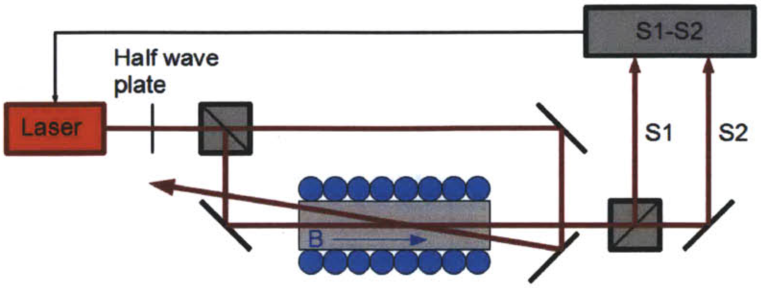


Figure 3-14: Experimental realization of the doppler free DAVLL setup. Standard saturated spectroscopy is done in a vapor, where a dichroism is induced by means of an applied magnetic field.

Cavity Lock to Reference

It is now necessary to use the narrow ($\approx$ 1MHz) reference laser generated with Doppler-free DAVLL in rubidium vapor to stabilize the cavity. This is done by beating the 780nm light from the laser sent to the experiment with the 780nm reference laser. A feedback system ensures that the beat frequency stays near a set value, thereby setting the relation between the experiment laser and reference laser, while keeping the cavity at a fixed length. To avoid high frequency noise, we cut the bandwidth of the feedback at 1 Hz.

3.5.4 FPGA Based Laser Frequency Sweep

As mentioned before, the laser light used in the locking scheme of the 780 and 480 nm lasers was modulated by an EOM, with the first sideband sent to the cavity and the main carrier sent to the experiment. Thus, the EOM frequency determines the relationship between the reference (cavity) frequency and the light sent to the experiment. Shifting the EOM modulation frequency causes the laser to re-lock, and for the carrier light to be shifted to a new value. However, frequency shifts that are too large cannot be compensated for by the PDH method, because the error signal is only valid in a narrow region of typically several MHz. Thus, while it is possible to shift the laser frequency by as much as GHz, this must be done in small steps.

The EOM is modulated by a synthesizer (PTS3200), which is capable of producing frequencies between 1 and 3200 MHz. It is able to shift frequency by about 1 MHz in 5 μs, and about 100 MHz in 20 μs [25]. By quickly scanning the frequency of the probe light, it is possible to generate EIT curves predicted by equation (2.6), plotted in Figure 2-2. The PTS3200's ability to scan the frequency over μs has a distinct practical advantage. As discussed previously, the system has a limited lifetime, and takes about 500-1000 ms to produce. If the frequency could only be changed once every few hundred milliseconds, then each preparation yields only a single data point on the transmission vs. frequency plot. Scanning fast, however, allows capture of the whole curve with a single sample.

PTS Control Scheme

The PTS3200 is a versatile frequency synthesizer capable of outputting frequencies between 1 and 3200 MHz [25]. While it comes in many variants and with many options, just two variants are currently in use by the experiment. While a fully functional PTS3200 is a standalone device, the 'minimalist' versions we use lack a 10 MHz clock and manual control. The lack of an internal clock is a cost saving feature, since a single external clock can run all of the synthesizers simultaneously. One of the synthesizers currently in use is a refurbished device that was used for an unknown, specialized, purpose in the past. It also lacks an internal clock and its control pin configuration is substantially different, but otherwise it functions the same way. The serial number corresponding to the basic, minimalist version is '3200RJN20'. The serial number corresponding to the specialized device is '3200RHN1X-13/X-54/SX-72/X-84/X-85'.

The PTS3200 has a straightforward remote-control scheme. An amphenol-57 connector in the back panel contains 50 pins that allow the user to set the frequency, amplitude, latching and phase rotation options [25]. The only features we needed were the amplitude and frequency control. The amplitude control is done by sending an analog voltage to a particular pin, resulting in the RMS output being 1/2 the signal level.

Decade	1 (BCD Pin)	2 (BCD Pin)	4 (BCD Pin)	8 (BCD Pin)
GHz	43	44		
100 MHz	15	16	40	41
10 MHz	17	18	19	20
1 MHz	1	2	26	27
100 KHz	3	4	28	29
10 KHz	5	6	30	31
1 KHz	7	8	32	33
100 Hz	9	10	34	35
10 Hz	11	12	36	37
1 Hz	13	14	38	39
Control	42			
Amplitude	21 (ground)	22 (2x level)		

Table 3.2: BCD Control Scheme for PTS 3200RJN20

Decade	1 (BCD Pin)	2 (BCD Pin)	4 (BCD Pin)	8 (BCD Pin)
GHz	13	14		
100 MHz	43	44	49	38
10 MHz	15	16	40	41
1 MHz	17	18	19	20
100 KHz	1	2	26	27
10 KHz	3	4	28	29
1 KHz	5	6	30	31
Control	42			
Amplitude	21 (ground)	22 (2x level)		

Table 3.3: BCD Control Scheme for PTS 3200RHN1X-13/X-54/SX-72/X-84/X-85

The rest of the control is done by TTL-level digital logic, where $0V = 1$ and $> 3.3V = 0$. All pins are initially pulled up, and must be pulled down by external control. To enable remote frequency programming, the control pin (42 on either model) must be enabled. Beyond that, control of the PTS is done by inputting the frequency in a BCD format - every decimal digit is individually controlled by 4 binary bits. The correspondence between digit and pin, as well as the other control pins are identified in Tables 3.2 and 3.3.

In order to integrate these devices into the experiment, a specialized controller is required, and was implemented using FPGA boards.

Logic Pin Output	Bus Pin Label	Logic Pin Output	Bus Pin Label
31	11	15	29
30	12	14	30
29	13	13	33
28	14	12	34
27	15	11	35
26	16	10	36
25	17	9	37
24	18	8	38
23	21	7	39
22	22	6	40
21	23	5	43
20	24	4	44
19	25	3	45
18	26	2	46
17	27	1	47
16	28	0	48

Table 3.4: Correspondence between logic pins and physical pin labels in the XEM3001v2 FPGA

FPGA Based Controller

Field Programmable Gate Arrays (FPGAs) are a versatile technology allowing the user to effectively create temporary integrated circuits by determining the arrangement of gates on the board, making it an ideal tool for creation of a controller that needs to operate in a variety of distinct regimes. The FPGA used was a minimalist board produced by Opal Kelly, the XEM3001v2 [20]. The board's 83 pins, set by the user's logic, are used to set the PTS's input to the proper value. The FPGA contains three 'buses', arrays of pins some of which can be set by the logic of the FPGA.

The frequency is encoded in a 32 bit number in BCD format, corresponding to 8 decimal digits of precision, the most significant one of which is the GHz digit. The FPGA's `ybus` encodes the frequency, there are 8 groups of 4 bits corresponding to 8 digits of precision. The most significant bits are `ybus[31:28]`, corresponding to GHz, every next 4 pins is less significant by 1 order of magnitude (`ybus[27:24]` represent 100's of Mhz etc...). Table 3.4 shows this correspondence.

Before the synthesizer can be controlled, the PTS must be initialized, a 'bit file', which defines the FPGA's logic, is loaded onto the FPGA via USB connector. This file is compiled from the language Verilog into a binary form that accounts for the specifications of the chosen hardware. From that point on, the FPGA and computer can communicate via the USB, and the user can set pre-defined parameters to desired values. Setting these parameters can alter the FPGA's state or trigger it to perform certain actions. To control the laser efficiently and preserve the laser lock, the FPGA should have two main control modes. In the first control mode (termed 'fast control'), it should be able to take small steps triggered by TTL pulses, on the timescale of μs, to quickly scan the frequency of the laser by many discrete steps. When the ramping is completed, it should be able to return the laser to its original frequency without losing the laser lock, and it should be able to take the small steps both up and down. In the second control mode (termed 'slow control'), the device should be able to go to an arbitrary frequency defined by the user on the computer, but also retain the ability to ramp down to a certain minimum value.

Fast Control Mode

The fast control mode was designed to perform a quick sweep of the laser frequency. The computer cannot instruct the FPGA on the timescale of μs, so the only way that this is possible is if the FPGA is primed with the parameters necessary to take steps and ramp down, and then communicated with by means of a PulseBlaster card which sends a train of pulses to increment the frequency over the duration of the experimental cycle, followed by a 'cool-down' pulse instructing the FPGA to ramp the frequency down at the end of the experimental lifetime. Two pins are set to receive TTL pulses representing 'step' and 'ramp'. Listing A.1 shows a pseudo-Verilog implementation of the fast control mode, where the steps are taken up, and the ramp brings the frequency back down. In this mode, the user controls the following parameters:

- **'maxclkdiv':** This is the number of cycles that the FPGA waits between subsequent steps in the ramp, 1 cycle $\approx$ 24 ns.

- **Maximum:** The frequency will never ramp or step to above this value.
- **Minimum:** The frequency will never ramp or step to below this value
- **Step:** The step size taken when a step signal is received. In practice, this number is given as a power and a coefficient, limiting the options for possible step sizes to $a10^b$, where a, b are integers, but simplifying the FPGA's calculations.
- **Ramp Step:** The step size taken when a ramp signal is received.

When these parameters are set by the user via LabView, the FPGA needs nothing more from the computer, and is controlled by the TTL pulses from the pulseblaster.

For convenience, there are actually two fast control modes, one where the steps go up and the ramp comes down, and one which is reversed. The reason that these two modes were split is that within a single experimental cycle there is no need to have capability to step up and down, and reprogramming the FPGA between cycles is trivial. Full implementations of the fast control modes can be found in Appendix A.

One constraint that is abstracted over in the pseudocode is how arithmetic on the BCD number structure was done. Since Verilog cannot handle this type of calculation in a built in fashion, the addition/subtraction in BCD was done manually. It can be seen in full in Appendix A.

```
//Declare Variables:

//User Defined Settings
maxclkdiv; //Value that clkdiv wraps around to - that is, how many cycles are waited between subsequent steps during a ramp to the minimum
step; //Step size added (up)
stepdown; //Step size subtracted (down)
max; //Maximum to saturate at
min; //Minimum to saturate at

//Internal System Variables
```

```
clkdiv; //Meant to slow down ramping down - a step down is taken every time clkdiv wraps around
count; //Current value of the frequency
satHi; //Boolean dictating if the system is saturated high
satLo; //Boolean dictating if the system is saturated low
stepUPnow; //Boolean dictating that the system should take a step up during this clock cycle
rampDOWN; //Boolean dictating that the sytem should start to ramp down during this clock cycle
rampingDOWN; //Boolean dictating that the system is currently ramping down
clk; //Clock signal generating a positive edge every 24 ns

outputpins<=count;

always @(posedge clk) begin

  //Saturation
  if (count >= max) begin
    satHi=1;
  end else begin
    satHi=0;
  end

  if (count <= min) begin
    satLo=1;
  end else begin
    satLo=0;
  end

  //Look for positive edges on the pins that trigger steps and ramps
  if ((TriggerUPpin == 1) && (TriggerUPpin(previousCycle) == 0)) begin
    stepUPnow <= 1;
  end else begin
    stepUPnow <= 0;
  end
```

```

if ((TriggerDOWNpin == 1) && (TriggerDOWNpin(previousCycle) == 0)) begin
  rampDOWN <= 1;
end else begin
  rampDOWN <= 0;
end

//STEP UP
if (stepUPnow == 1) begin
//StepUP
rampingDown <= 0;
rampingUp <= 0;
if (satHi == 0) begin
    count = count + stepUP;
  end else begin
  count <= max;
end
end

//RAMP Down Trigger
if (rampDOWN == 1) begin
  rampingDown <= 1;
end

//RAMP Down Steps
if (rampingDown == 1) begin
  if (count <= min) begin
    rampingDown <= 0;
  end else begin
    if (clkdiv == 0) begin
      clkdiv <= maxclkdiv;
      if (satLo == 0) begin
   //StepDown
        count <= count - stepdown;
        else begin
        clkdiv <= clkdiv - 1;
      end
    end
  end

  if (count > max) begin
    count <= max;
  end

  if (count < min) begin
    count <= min;
  end
end
```

Listing 3.1: Pseudo-Verilog implementation of fast control mode

Slow Control Mode

In certain situations, the frequency needs to remain constant over a single experimental cycle, but varied over subsequent cycles. Further, after some number of cycles, the frequency needs to be ramped back to minimum. In this control mode, it is unnecessary to use PulseBlaster pulses, and signals can be passed to the FPGA over USB. These signals temporarily set a user controlled parameter, which acts as a trigger to shift frequency or initiate ramp. The parameters for this mode are:

- **'maxclkdiv':** This is the number of cycles that the FPGA waits between subsequent steps in the ramp, 1 cycle $\approx$ 24 ns.

- **Minimum:** The frequency will never ramp to below this value.

- **GoToFreq:** When the user sets this to one, the FPGA jumps to the given frequency. Activating this trigger is done by a LabView VI.

- **Frequency:** When 'GoToFreq' is set to 1, the FPGA jumps to this frequency.

- **Ramp:** When the user sets this to one, the FPGA initiates a ramp. Activating this trigger is done by a LabView VI.

- **Ramp Step:** The step size taken when a ramp signal is received.

- **Number of Ramp Steps:** During the ramp, ramping down will not exceed taking this number of steps.

A pseudo-Verilog implementation is shown in Listing 3.2, and a full implementation is shown in Appendix A.

```
//Declare Variables:

//User Defined Settings
maxclkdiv; //Value that clkdiv wraps around to - that is, how many cycles are waited between subsequent steps during a ramp to the minimum
stepdown; //Step size subtracted (down)
freq; //Destination frequency to go to on command
min; //Minimum to saturate at
goToFreq; //Boolean dictating that the system should go to a frequency
ramp; //Boolean dictating that the system should ramp
MaxNumberSteps; //Counter of how many steps to take during ramp down

//Internal System Variables
clkdiv; //Meant to slow down ramping down - a step down is taken every time clkdiv wraps around
count; //Current value of the frequency
satHi; //Boolean dictating if the system is saturated high
satLo; //Boolean dictating if the system is saturated low
goToFreqNow; //Boolean dictating that the system should go to a frequency during this clock cycle
rampNow; //Boolean dictating that the sytem should start to ramp down during this clock cycle
rampingDown; //Boolean dictating that the system is currently ramping down
clk; //Clock signal generating a positive edge every 24 ns
```

```
numberSteps; //Counter of how many steps are remaining in the step-down process

always @(posedge clk) begin

  //Saturation
  if (count <= min) begin
    satLo=1;
  end else begin
    satLo=0;
  end

  //Posedges

  if ((goToFreq == 1) && (goToFreq(previousCycle) == 0)) begin
    goToFreqNow <= 1;
  end else begin
    goToFreqNow <= 0;
  end

  if ((ramp == 1) && (ramp(previousCycle) == 0)) begin
    rampNow <= 1;
  end else begin
    rampNow <= 0;
  end


  //Go to frequency
  if (goToFreqNow == 1) begin
    rampingDown <= 0;
    if (freq > min) begin
      count <= freq;
    end else begin
      count <= min;
    end
  end


  //RAMP Down Trigger
  if (rampNow == 1) begin
    rampingDown <= 1;
    numberSteps <= MaxNumberSteps;
  end

  //RAMP Down Steps
  if (rampingDown == 1) begin
    if (numberSteps == 0) begin
      rampingDown <= 0;
    end else begin
      if (clkdiv == 0) begin
        clkdiv <= maxclkdiv;
        if (satLo == 0) begin
        //StepDown
          count <= count - stepdn
        end else begin
          count <= min;
        end
        //Adjust counter of steps remaining
        numberSteps <= numberSteps - 1;
      end else begin
        clkdiv <= clkdiv - 1;
      end
    end
  end

  if (count < min) begin
    count <= min;
  end
end
```

Listing 3.2: Pseudo-Verilog implementation of slow control mode

Chapter 4

Imaging Rydberg Super-Atoms

4.1 Experimental Constraints

The purpose of imaging the Rydberg excitations has been made clear Chapter 1, but doing this is not easy. This chapter will illustrate the proposed imaging method and prove its viability using preliminary trials on test targets. Using theoretical predictions it is possible to propose multiple criteria required for successful imaging. In addition, there are other, practical, constraints on the design and implementation of the system. The key problems are:

- **Resolution:** The Rydberg blockade volumes are about 5μm in diameter. This means that a resolution of better than 2.5μm is necessary to convincingly characterize the system.

- **Magnification:** The lifetime of the Rydberg atoms is limited to 20kHz, which means that the camera needs to have a high quantum efficiency. Due to technological limitations, such cameras typically have large pixels (or are prohibitively expensive), implying that the magnification must be 10x or more.

- **Field of View:** To see multiple Rydberg Excitations in some sort of crystal, the field of view (or the area that can be seen sharply), must be several tens of microns.

- **Depth of Field:** Ideally, the depth of field should not need to be more than a single Rydberg volume, but due to limitations on the precision of the positioning components, the focus may be up to 50μm away from the cloud.

- **Spatial Constraints:** The primary spatial constraint is the vacuum chamber window. This window is located 60.6 mm above the cloud, and is 90.9 mm thick. Thus, the first lens must be above this window.

- **Robustness:** Ideally, the system would be convenient to adjust. Further, since the area around the vacuum chamber is crowded, the system must consume relatively little space.

The greatest problem with this system is the large separation from the object, and the resulting aberrations due to such a large cone of light passing through the vacuum chamber window. Constructing a system which neutralizes these concerns is the ultimate goal.

4.2 Basics of Imaging Optics

Before fully developing the system, it is necessary to understand the basics of conventional optics. The primary approximation assumed in this section is the 'Paraxial' approximation, which claims that all relevant angles are small enough to assume $\sin\theta \approx \theta$ [4].

4.2.1 Spherical and Thin Lenses

The most common type of lens is the spherical lens - a pair of spherical surfaces whose intersection defines a lens. The focal length f of such a device is given by the lens-maker's equation:

$$\frac{1}{f} = (n-1)\left[\frac{1}{r_1} - \frac{1}{r_2}\right] \tag{4.1}$$

Where r_i are the radii of the spherical surfaces, and are taken to be > 0 for convex and < 0 for concave surfaces. In many situations, it is useful to assume the 'thin lens' approximation, which claims that rays passing through a lens will only change their angle θ:

$$\theta_2 = \theta_1 - y/f \tag{4.2}$$

Where y is the distance from the center. Under this approximation, rays emanating from an object in one plane s_0, will be perfectly imaged by the lens onto another plane s_i. The distances to these 'conjugate' planes are given by:

$$\frac{1}{s_0} + \frac{1}{s_i} = \frac{1}{f} \tag{4.3}$$

The magnification of such a system [5] can be found to be simply:

$$M = -\frac{s_i}{s_0} \tag{4.4}$$

Thus far, the lens has been assumed to be 'large enough' to work, but its size has not had any effect. This in not the case in practice, therefore, one important parameter of the lens is the f-number, defined by:

$$n_f = \frac{f}{\phi} \tag{4.5}$$

Where ϕ is the lens diameter that rays pass through, called the aperture [5].

The numerical aperture, NA, is more practical for scientific applications. It is defined as

$$NA = \sin\theta_m \tag{4.6}$$

where θ_m is the marginal ray entering the imaging system.

While the characterization is basic, it serves as a very good 'zeroth' order approximation to the full performance of the system. The thin lens approximation will be violated, and the aperture will have significant effects, but the tools will remain valid. Further, this formalism can be scaled without difficulty to many lenses with the use of equation (4.3), by taking the object surface of each lens lens to be the image surface of the previous one.

4.2.2 Diffraction Limit

It may be convenient to refer to imaging in terms of rays, but since the rays are merely wavefronts, new effects will emerge at wavelength sized length scales due to diffraction. The primary effect is the diffraction limit of the resolution. Famously [4], a point source going through a circular hole will produce an Airy pattern. The same effect is seen in imaging systems. The size of the first peak produced by diffraction serves as a convenient measure of resolutions. It's size is given by the Rayleigh criterion:

$$2r_d = 1.22\lambda/NA \tag{4.7}$$

where r_d is the minimum distance between distinguishable objects. This limit is impossible to get around in conventional imaging systems. However, other effects may limit the system to an even lower resolution, and must be considered as well. The goal of the imaging system design is to get as close to the diffraction limit as possible.

4.2.3 Aberrations

There are many sorts of aberrations which force rays to deviate from the idealized thin lens behavior [5, 4]. In this section, however, we will consider only the ones relevant in this situation.

Spherical Aberration

The dominant effect is the fact that outside the thin lens approximation, spherical lenses do not focus rays all rays emanating from an object plane to an image plane. The intuitive reason for this to be the case is that the rays will propagate inside the lens, since the lens is not infinitely thin. A relevant example of spherical aberration is the effect of a glass slab in the imaging system. As since rays of different angles

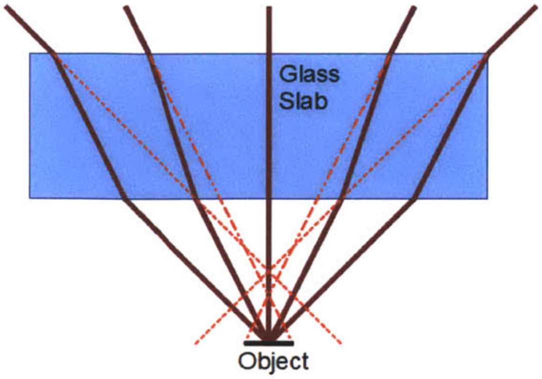


Figure 4-1: Spherical aberration by a glass slab.

will refract by different angles and propagate at a modified angle inside the slab, the angular distribution of rays will be different upon exiting the slab. This situation is shown in figure 4-1, and corresponds to spherical aberration in an infinite focal length lens.

Spherical aberration can be overcome by using non-spherical surfaces. One type of non-spherical lens is called the aspherical lens. This lens has a surface designed to converge incident parallel rays onto a single point on its optical axis, and the reverse. This lens greatly reduces such aberration, but suffers from very bad off-axis performance, narrowing the accessible field of view.

Coma and Astigmatism

This pair of effects deals with the off axis performance of an imaging system. Off axis points can suffer from being magnified by different amounts and focuses to different planes. These effects will be clearly observed on the ray tracing spot diagrams and can be seen as flaring or elongation of bundles of rays. These effects primarily serve to limit field of view, but are relatively unimportant to the problem at hand because the required field of view is small.

Device	Focal Length (mm)	Maximum Device Thickness (mm)	Distance to Next Device (mm)
Object	-	-	60.58
Vacuum Window	∞	9.09	10.37
Aspheric Lens	100.0	36.00	1.85
Plano-Convex Lens	500.0	9.00	451.80
Plano-Concave Lens	-30.0	3.00	89.81
CCD Camera	-	-	-

Table 4.1: Summary of the imaging system's optical path

4.3 Imaging System Design

To design the imaging system, Zemax, a program for the numerical simulation of optical systems, was used. The most effective setup uses a "telephoto" arrangement. This setup has two lenses to collect, collimate and converge the light, and another lens to diverge it in order to increase magnification. To offset the spherical aberrations inherent to large lenses, the collection lens was a large aspheric lens (AL100100-C, from Thor Labs), with a focal length of 100 mm, and it served to collimate the light. The next lens was a plano-concave lens with a focal length of 500mm (A100-100LPX, from Ross Optical), to recombine the light. The imaging system could have stopped there, but the magnification of the combined system would have been small (≈ 5). To avoid using a very long focal length lens for magnification, a powerful, diverging, plano-concave lens (LC1060, from Thor Labs) was inserted to make the rays converge less quickly on the CCD. Even better, moving this lens allows fine-tuning the magnification of the system without strongly affecting the resolution (possible magnifications range from 10x to 40x). The imaging system is shown in Figure 4-2, and a summary can be found in Table 4.1.

There are four important degrees of freedom: separation between the window and the aspheric lens, the separation between the plano-convex lens and the plano-concave lens, and the separation between the plano-concave lens and the CCD. The last degree of freedom is an iris - a device which allows the user to limit the NA. In the calculations, the iris was located between the large diameter lenses, so its value gave a convenient measure of the size of the light cone. The system is very tolerant of CCD position errors, and the concave lens position affects primarily the magnification, not the resolution. The most sensitive parameter is the separation of the window and the aspheric lens. This parameter is sensitive because aspheric lenses are designed to work at infinite conjugation ratios and aberrations occur if they do not. The separation between the aspheric lens and the plano-convex lens is irrelevant, since the rays between the two are parallel to a very good approximation.

The proposed design can be simulated and diagnosed. The most telling figure is the spot diagram, Figure 4-3. This diagram shows the separate effects of geometric and diffraction aberrations. The figure shows how a point source in the object space will be seen in image space. The deviations in the spot diagram were the key parameter in optimizing the system. Further, the spot diagram gives a valuable idea of how to balance the geometric aberrations with the Airy disks - as NA grows, the Airy disks shrink but the geometric aberrations grow. Calculations showed that opening the iris as wide as possible gave optimal results, shown in Figure 4-3.

The separation of the two effects of aberration robs the user of a good understanding of what a test picture will look like in practice. Further, there are two more issues to contend with - coarse graining due to pixelation and the fact that laser imaging is coherent. These effects can be simulated as well. Figure 4-4, shows a test figure shaped like the letter 'F', with a side of 5μm in object space. The figure is clearly readable, despite obvious distortion.

4.4 Experimental Imaging Diagnostics

The imaging system was implemented using standard optomechanical cage components and some custom made components, to accommodate the large lenses. The full system is shown in Figure 4-6. The USAF test target was held by an arm secured to a 3D translation stage, allowing movement of the object. For convenience, the system was folded form vertical to horizontal several inches before the diverging lens, but aligned such that this folding did not affect performance. An iris was placed before

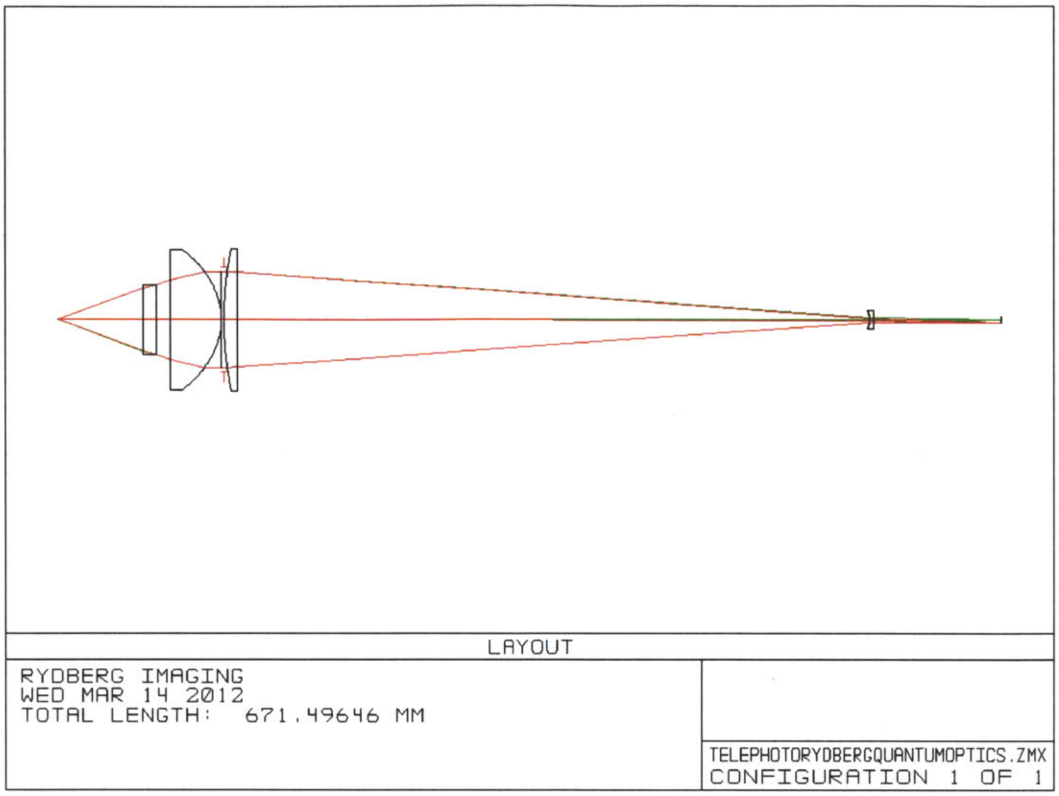
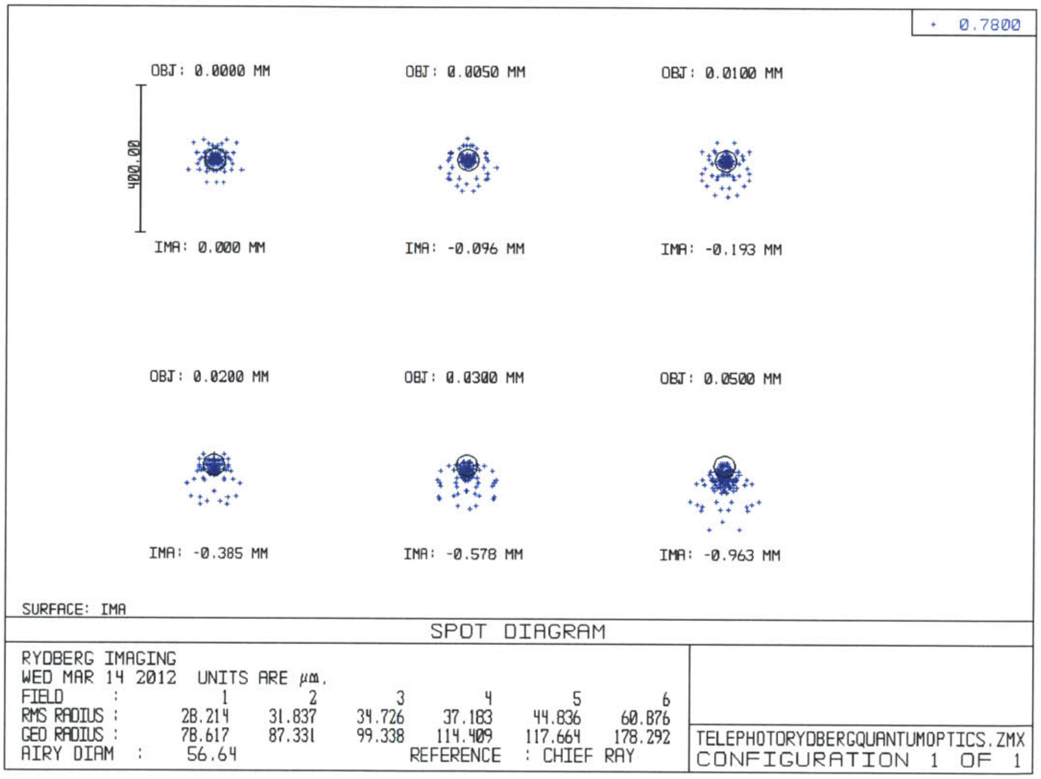


Figure 4-2: Imaging system layout

the diverging lens, in order to control the NA. Two sets of trials were done, with coherent and incoherent light. The primary camera used for alignment testing was the Unibrain 501b Fire-i Camera.

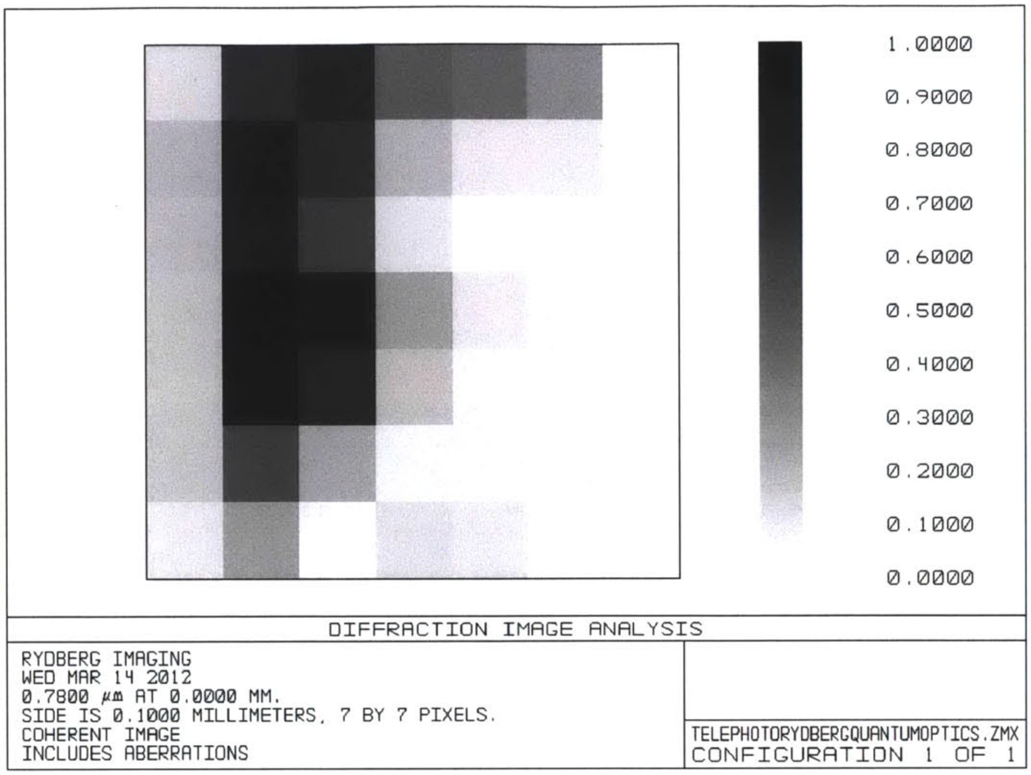
In order to measure the resolution, a 1951 USAF test target was used. The target pattern is shown in Figure 4-7. The target consists of seven groups, each of six test elements. Each element contains a picture of three horizontal and three vertical black lines. The elements get progressively smaller, as shown in Table 4.2, to a maximum spatial frequency of 228.0 cycles/mm, which corresponds to a spatial period of 4.38μm. From the Nyquist theorem, resolving this frequency requires a resolution of 2.19 μm. Resolving this group was the purpose of the diagnostics.

There was one serious difference between the trial system and the real experiment. The vacuum window of the chamber is 10 cm in diameter. In the trial setup, a smaller window was used (5cm), but set such that it captured the same NA. Simulations in Zemax showed that the effects on the system were negligible. The reason for this



```
                              SPOT DIAGRAM
RYDBERG IMAGING
WED MAR 14 2012  UNITS ARE µm.
FIELD       :       1        2        3        4        5        6
RMS RADIUS  :  28.214   31.837   34.726   37.183   44.836   60.876
GEO RADIUS  :  78.617   87.331   99.338  114.409  117.664  178.292   TELEPHOTORYDBERGQUANTUMOPTICS.ZMX
AIRY DIAM   :   56.64                 REFERENCE  : CHIEF RAY         CONFIGURATION 1 OF 1
```

Figure 4-3: Theoretical spot diagram. The black circles are Airy disks, and blue markers are geometrically traced rays.



```
DIFFRACTION IMAGE ANALYSIS
RYDBERG IMAGING
WED MAR 14 2012
0.7800 µm AT 0.0000 MM.
SIDE IS 0.1000 MILLIMETERS, 7 BY 7 PIXELS.
COHERENT IMAGE
INCLUDES ABERRATIONS
TELEPHOTORYDBERGQUANTUMOPTICS.ZMX
CONFIGURATION 1 OF 1
```

Figure 4-4: Diffraction effects on a test image

is simple: the aberrations induced by the window come from the fact that light rays are 'shifted up' by a distance depending on their angle of incidence, due to refraction, but their initial angle to the optical axis does not change. Whether this shift takes place close to or far from the object is irrelevant, as shown by Figure 4-5.

4.4.1 Incoherent Imaging Trials

The first set of trials was done by imaging the target illuminated by a commercial red LED. The results are shown in 4-10. Sadly, the smallest group was difficult to resolve with the naked eye. In order to resolve the feature, computer analysis is needed. To determine the spatial variation in the $\hat{x}$ direction, the feature was cropped from the image, and all the pixels in the $\hat{y}$ direction were averaged. This 'compression' averaged out noise, and produced a clearer image. A sample result (for element 3) is shown in figure 4-8. This process was performed on all elements for their vertical and horizontal component. Each plot gives four measures of the contrast, dubbed as the

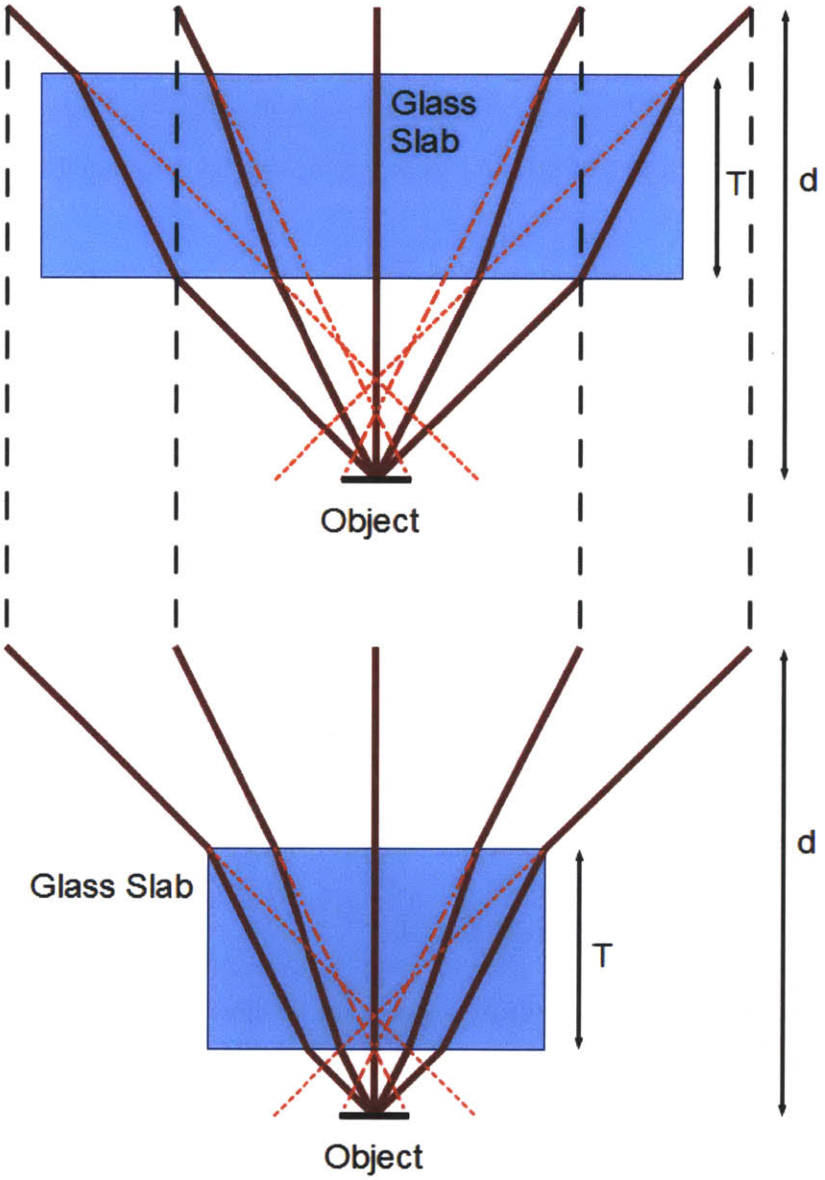


Figure 4-5: Aberration caused by a glass window of thickness T. Rays are traced back to their apparent object to reveal the aberration. This figure also illustrates that the distance between the object and window has no effect on the ray profile a distance d from the object.

Group Number (spatial frequency, cycles/mm)										
Element	-2	-1	0	1	2	3	4	5	6	7
1	0.250	0.500	1.00	2.00	4.00	8.00	16.00	32.0	64.0	128.0
2	0.280	0.561	1.12	2.24	4.49	8.98	17.95	36.0	71.8	144.0
3	0.315	0.630	1.26	2.52	5.04	10.10	20.16	40.3	80.6	161.0
4	0.353	0.707	1.41	2.83	5.66	11.30	22.62	45.3	90.5	181.0
5	0.397	0.793	1.59	3.17	6.35	12.70	25.39	50.8	102.0	203.0
6	0.445	0.891	1.78	3.56	7.13	14.30	28.50	57.0	114.0	228.0

Table 4.2: 1951 USAF test target spatial frequencies

left/right contrasts and two central contrasts, shown in 4-8. Due to symmetry, the central contrast were averaged together. Figure 4-9 shows the contrasts as a function of half the target period (i.e. the resolution). There is a negative trend clearly visible in the plot, with the maxima and minima converging close to the point where the real features are indistinguishable from noise.

The features are weakly observable, but not as predicted by Zemax. However, the blurring is due to an as yet unaccounted effect - vibration in the target. The exposure times were about 0.25 s, so any vibrations that took place on a shorter time scale degraded the image. These vibrations became clearly visible when a brighter light source was used - a red laser, since the image gained resolution, but visibly vibrated.

4.4.2 Coherent Imaging Trials

Coherent imaging is a more accurate simulation of how the system will be used. A 780 nm diode laser was attenuated and used to illuminate the target. Initially, the Unibrain 501b Fire-i camera was used for trials. Typical exposure times were about 50ms. An exposure of the smallest element is shown in Figure 4-11. Clear artifacts of coherent imaging can be seen from diffraction and interference effects. Most importantly, the resolution clearly allows the features to be distinguished. Knowing the size of the pixels of the camera and the size of the object also allows a basic determination of the magnification $M = -19 \pm 1$, where the negative merely shows that the image is inverted. The blurred features and coarse graining of the pixels limits the precision of the measurement.

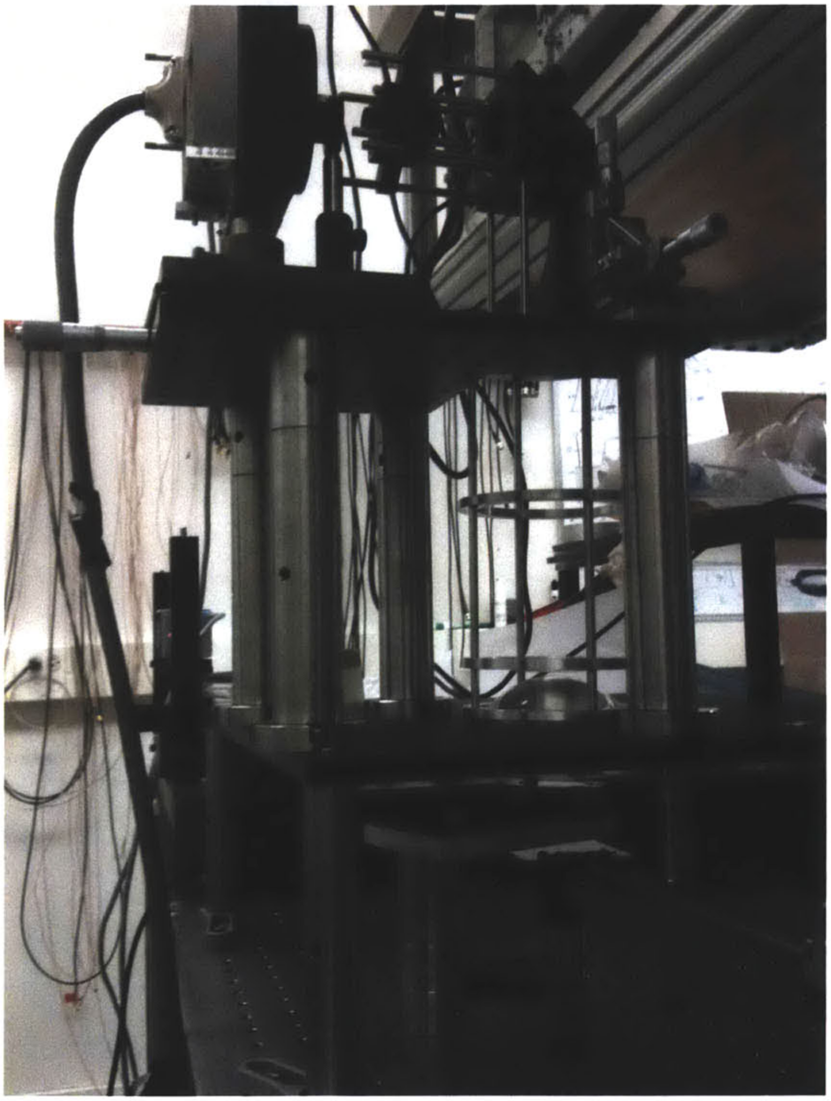


Figure 4-6: Actual Setup

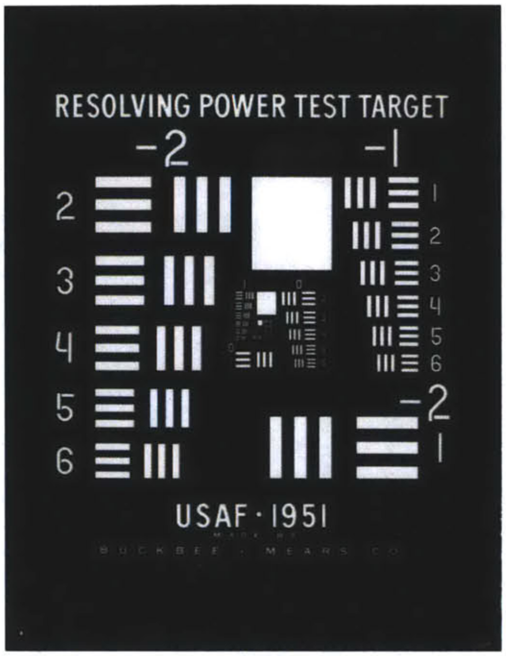


Figure 4-7: 1951 USAF Test Target

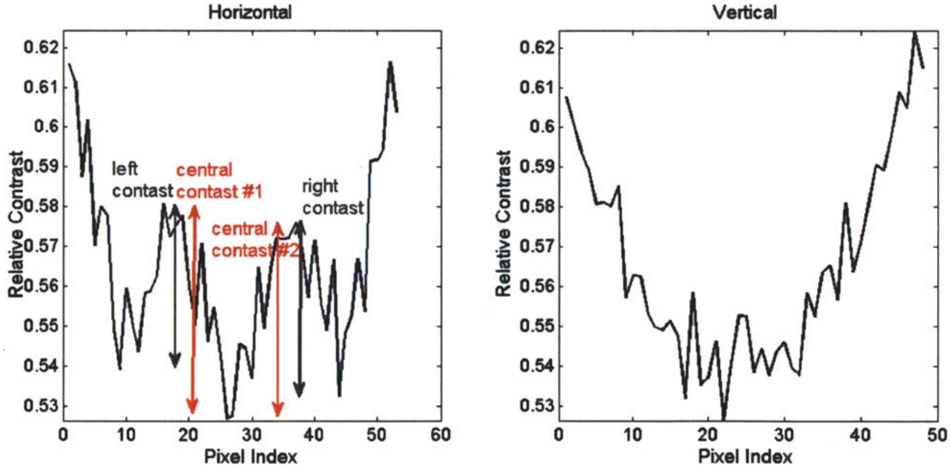


Figure 4-8: Results of the described averaging process to resolve the horizontally varying half of element 7-3 of the USAF target (hence why one plot shows several dips, while the other, just one). Relative contrast is a measure of how bright the pixel is, 1 is saturation and 0 is perfect darkness.

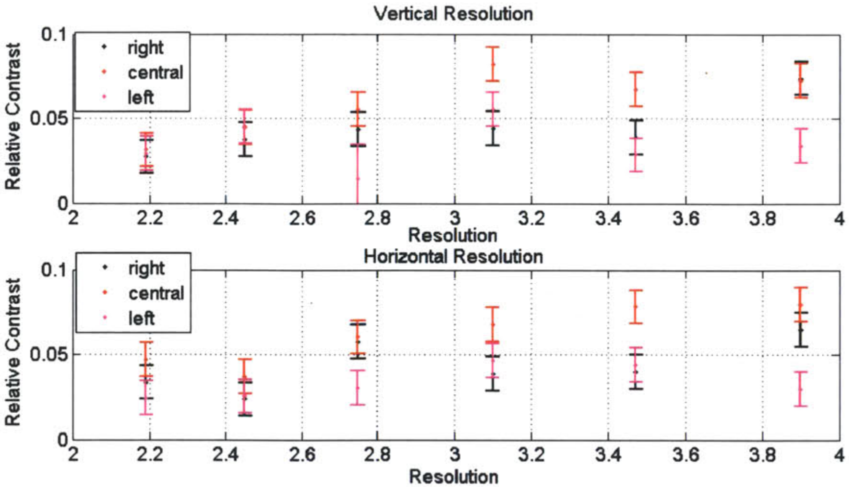


Figure 4-9: The contrast is shown as a function of resolution for the various features.

In reality, the camera to be used will be Apogee 260e, a high end astronomy camera. The quantum efficiency is much larger, but so are the pixels. Using this camera yields the image in Figure 4-12. Though the image quality is degraded, the smallest feature is still distinguishable.

One of the experimental constraints is the retroreflection of the MOT laser. This laser leaves the same window that serves the imaging system. In order to send it back to the MOT, a thin rod with a mirror is inserted above the two large lenses. The mirror sends the beam towards an optical system where the beam is re-collimated, and sent back into the MOT. This element cuts down the NA of the system, but the effects are practically invisible.

4.4.3 Trial Results

With the system fully simulated using the test target, it is necessary to compare the results with the demands. There are multiple constraints on the design and implementation of the system. The key problems are

- **Resolution:** The resolution, accounting for all blurring mechanisms is about 2.5μm - small enough to detect the Rydberg volumes.

- **Magnification:** As seen from Figure 4-12, the magnification is high enough to

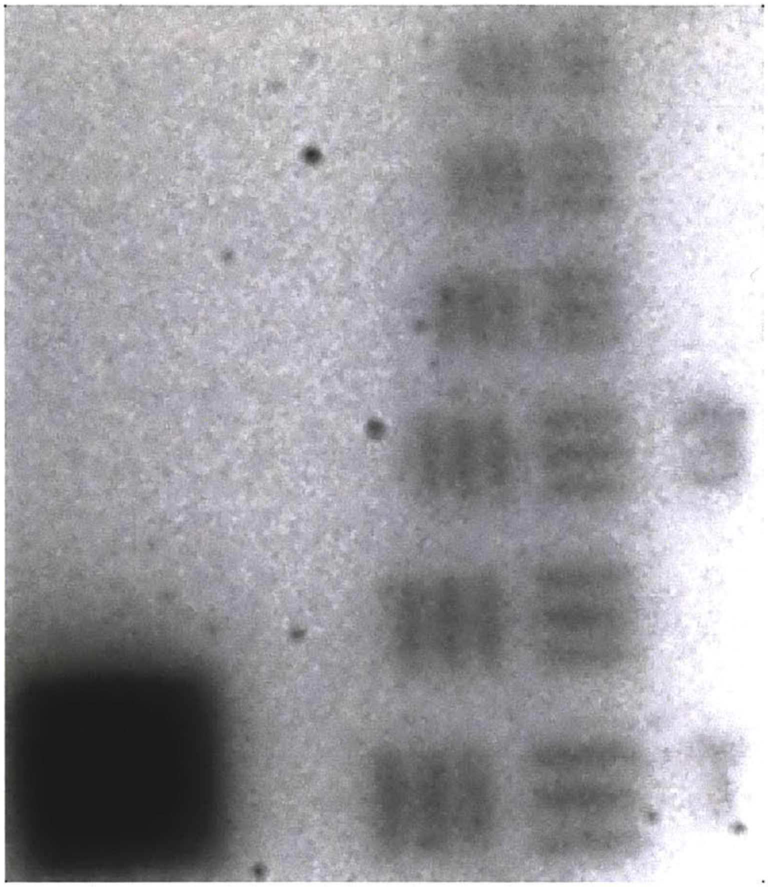


Figure 4-10: Incoherent image of the 7 series of the USAF target, using the Unibrain 501b Fire-i camera

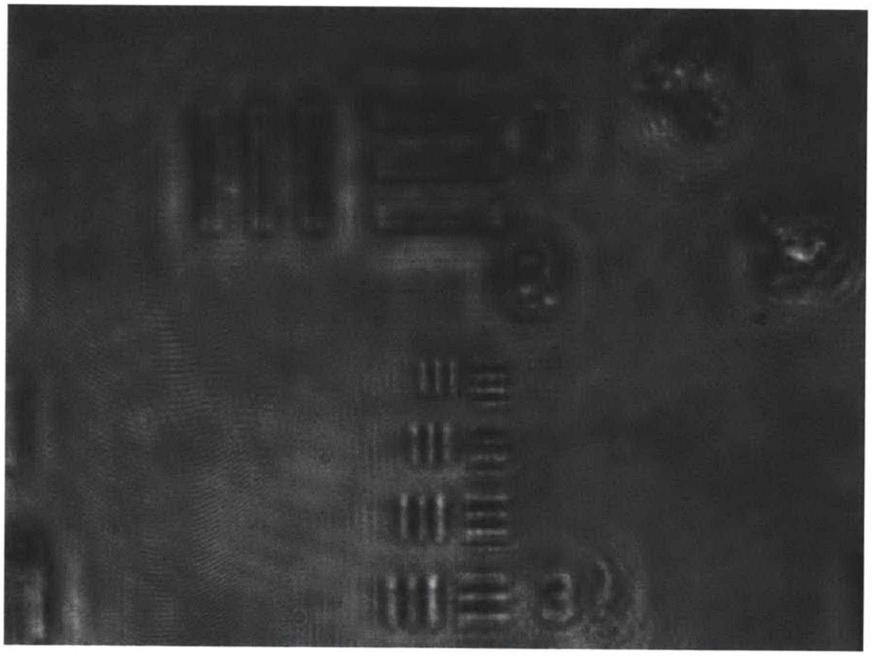


Figure 4-11: Coherent image of the 7 series of the USAF target, using the Unibrain 501b Fire-i camera

be able to use the Apogee camera's large pixels, but small enough such that the signal is not dispersed over a great area.

- **Field of View:** As predicted, the resolution decays slowly when the object is off the axis. However, the resolution was sufficient to resolve the lines within at least a 40μm radius.

- **Depth of Field:** Since the system could be aligned by moving the objective with a standard micrometer stage, the depth of field can be estimated to be about 80μm.

- **Spatial Constraints:** The system was explicitly designed to fit into the required space.

- **Robustness:** The system is relatively robust, fault tolerant, and can be conveniently aligned.

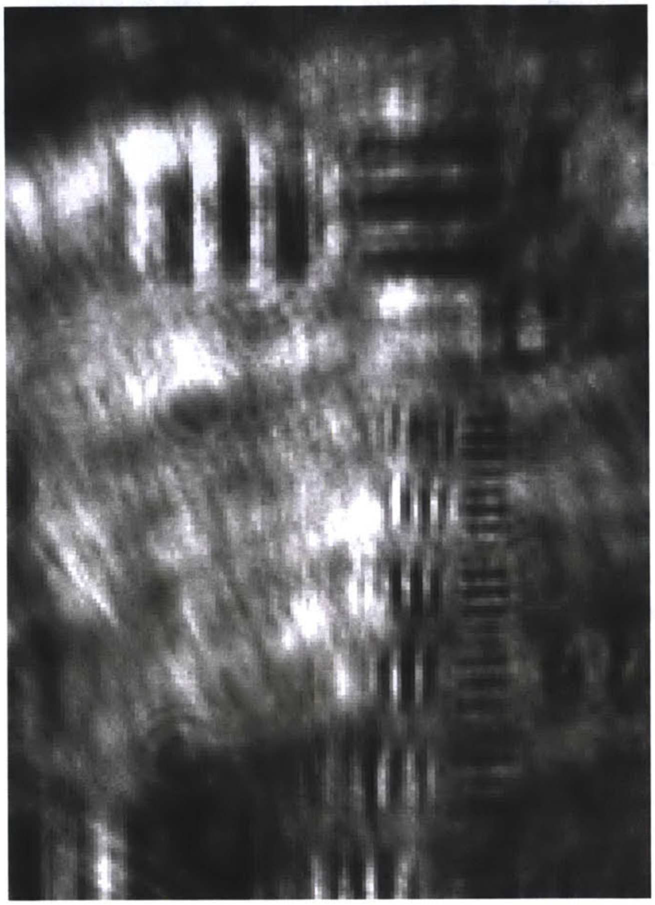


Figure 4-12: Coherent image of the 7 series of the USAF target, using the Apogee 260e camera

Appendix A

FPGA Programming

A.1 Control Modes

A.2 TTL-Controlled Step Up, Ramp Down Mode

```
`default_nettype none
`timescale 1ns / 1ps

module FastControl(
    input  wire [7:0]  hi_in,
    output wire [1:0]  hi_out,
    inout  wire [15:0] hi_inout,

      input clk,
      input button,
      output wire [7:0] led,
    output wire [31:0] ybus,
    input [13:2] zbus
      );

reg [31:0] clkdiv; //Meant to slow down ramping down - a step down is taken every time clkdiv wraps around
reg [31:0] maxclkdiv; //Value that clkdiv wraps around to.
reg [31:0] count;
reg [7:0] decade;
reg [31:0] step; //Step size added (up)
reg [31:0] stepdown; //Step size subtracted (down)
reg [7:0] index; //Iterator dummy variable in for loop to implement multi integer steps
reg [31:0] max; //Maximum to saturate at
reg [31:0] min; //Minimum to saturate at
reg satHi; //Boolean dictating if the system is saturated high
reg satLo; //Boolean dictating if the system is saturated low

reg trigger2; // Triggering booleans
reg trigger3;

reg prevzbus2;
reg prevzbus3;

reg goUp;
reg ramp;
reg goToMin;

reg rampingDown; //Boolean dictating if the system is in auto ramp-down mode
reg rampingUp; //Boolean dictating if the system is in auto ramp-up mode

reg [15:0] numberSteps; //Counter of how many steps are remaining in the step-down process

reg zbus2temp;
reg zbus3temp;



// Opal Kelly Module Interface Connections
wire         ti_clk;
wire [30:0] ok1;
wire [16:0] ok2;
```

```

// Endpoint connections:
wire [15:0] ep00wire;//Starting #LSB (AKA MIN)
wire [15:0] ep01wire;//Starting #MSB
wire [15:0] ep02wire;//Step1 LSB's
wire [15:0] ep03wire;//Step2 MSB's
wire [15:0] ep04wire;//Ending #LSB (AKA MAX)
wire [15:0] ep05wire;//Ending #MSB

wire [15:0] ep06wire;//stepdown LSB
wire [15:0] ep07wire;//stepdown MSB

wire [15:0] ep08wire;//Number of steps during ramp
wire [15:0] ep09wire;//Maxclkdiv LSB
wire [15:0] ep10wire;//Maxclkdiv MSB

wire [15:0] ep20wire;
wire [15:0] ep21wire;





assign led = ~count[31:24];
assign ybus[31:0] = ~count;

assign ep20wire = ~ybus[15:0];
assign ep21wire = ~ybus[31:16];


// Instantiate the okHost and connect endpoints.
wire [17*2-1:0] ok2x;
okHost okHI(
   .hi_in(hi_in), .hi_out(hi_out), .hi_inout(hi_inout), .ti_clk(ti_clk),
   .ok1(ok1), .ok2(ok2));

```

```
okWireOR # (.N(2)) wireOR (.ok2(ok2), .ok2s(ok2x));


//Opal Kelly Wires from PC
okWireIn     ep00 (.ok1(ok1),                          .ep_addr(8'h00), .ep_dataout(ep00wire));
okWireIn     ep01 (.ok1(ok1),                          .ep_addr(8'h01), .ep_dataout(ep01wire));
okWireIn     ep02 (.ok1(ok1),                          .ep_addr(8'h02), .ep_dataout(ep02wire));
okWireIn     ep03 (.ok1(ok1),                          .ep_addr(8'h03), .ep_dataout(ep03wire));
okWireIn     ep04 (.ok1(ok1),                          .ep_addr(8'h04), .ep_dataout(ep04wire));
okWireIn     ep05 (.ok1(ok1),                          .ep_addr(8'h05), .ep_dataout(ep05wire));
okWireIn     ep06 (.ok1(ok1),                          .ep_addr(8'h06), .ep_dataout(ep06wire));
okWireIn     ep07 (.ok1(ok1),                          .ep_addr(8'h07), .ep_dataout(ep07wire));
okWireIn     ep08 (.ok1(ok1),                          .ep_addr(8'h08), .ep_dataout(ep08wire));
okWireIn     ep09 (.ok1(ok1),                          .ep_addr(8'h09), .ep_dataout(ep09wire));
okWireIn     ep10 (.ok1(ok1),                          .ep_addr(8'h10), .ep_dataout(ep10wire));

okWireOut    ep20 (.ok1(ok1), .ok2(ok2x[ 0*17 +: 17 ]), .ep_addr(8'h20), .ep_datain(ep20wire));
okWireOut    ep21 (.ok1(ok1), .ok2(ok2x[ 1*17 +: 17 ]), .ep_addr(8'h21), .ep_datain(ep21wire));



always @(posedge clk) begin

```

```
zbus2temp <= zbus[2];
zbus3temp <= zbus[3];

//Wire In's
step [31:16] <= ep03wire;
step [15:0] <= ep02wire;
max [31:16] <= ep05wire;
max [15:0] <= ep04wire;
min [31:16] <= ep01wire;
min [15:0] <= ep00wire;
stepdown [31:16] <= ep07wire;
stepdown [15:0] <= ep06wire;
maxclkdiv [15:0] <=ep09wire;
maxclkdiv [31:16] <=ep10wire;


//Saturation
if (count >= max) begin
  satHi=1;
end else begin
  satHi=0;
end

if (count <= min) begin
  satLo=1;
end else begin
  satLo=0;
end

//Posedges

if ((zbus2temp == 1) && (prevzbus2 == 0)) begin
  trigger2 <= 1'b1;
end else begin
  trigger2 <= 0;
end
```

```

if ((zbus3temp == 1) && (prevzbus3 == 0)) begin
  trigger3 <= 1;
end else begin
  trigger3 <= 0;
end


//STEP UP trigger + step
if (trigger2 == 1) begin
//StepUP
rampingDown <= 0;
rampingUp <= 0;
if (satHi == 0) begin
  if (step[3:0] == 0 || count[3:0] + step[3:0] > 9) begin
    if (count[3:0] + step[3:0] > 9) begin
      count[3:0] <= count[3:0] + step[3:0] - 10;
    end
    if (step[7:4] == 0 || count[7:4] + step[7:4] > 9) begin
      if (count[7:4] + step[7:4] > 9) begin
        count[7:4] <= count[7:4] + step[7:4] - 10;
        end
      if (step[11:8] == 0 || count[11:8] + step[11:8] > 9) begin
        if (count[11:8] + step[11:8] > 9) begin
          count[11:8] <= count[11:8] + step[11:8] - 10;
          end
          if (step[15:12] == 0 || count[15:12] + step[15:12] > 9) begin
            if (count[15:12] + step[15:12] > 9) begin
            count[15:12] <= count[15:12] + step[15:12] - 10;
            end
          if (step[19:16] == 0 || count[19:16] + step[19:16] > 9) begin
            if (count[19:16] + step[19:16] > 9) begin
              count[19:16] <= count[19:16] + step[19:16] - 10;
              end
        if (step[23:20] == 0 || count[23:20] + step[23:20] > 9) begin
          if (count[23:20] + step[23:20] > 9) begin
            count[23:20] <= count[23:20] + step[23:20] - 10;
          end
          if (step[27:24] == 0 || count[27:24] + step[27:24] > 9) begin
            if (count[27:24] + step[27:24] > 9) begin
              count[27:24] <= count[27:24] + step[27:24] - 10;
            end
            if (count[31:28] + step[31:28] > 9) begin
                count[31:28] <= count[31:28] + step[31:28] - 10;
              end else begin
              count[31:28] <= count[31:28] + step[31:28];
            end
          end else begin
            count[27:24] <= count[27:24] + step[27:24];
          end
        end else begin
          count[23:20] <= count[23:20] + step[23:20];
        end
      end else begin
        count[19:16] <= count[19:16] + step[19:16];
      end
    end else begin
        count[15:12] <= count[15:12] + step[15:12];
      end
    end else begin
      count[11:8] <= count[11:8] + step[11:8];
    end
  end else begin
    count[7:4] <= count[7:4] + step[11:8];
  end
end else begin
  count[3:0] <= count[3:0] + step[3:0];
end
```

```
    end else begin
    count <= max;
  end
  end


//  //RAMP Down Trigger
  if (trigger3 == 1) begin
    rampingDown <= 1;
    numberSteps[15:0] <= ep08wire;
  end

  //RAMP Down Steps
  if (rampingDown == 1) begin
    if (count <= min) begin
      rampingDown <= 0;
    end else begin
      if (clkdiv == 0) begin
        clkdiv <= maxclkdiv;
        if (satLo == 0) begin
     //StepDown
          if (stepdown[3:0] == 0 || count[3:0] < stepdown[3:0]) begin
            if (count[3:0] < stepdown[3:0]) begin
              count[3:0] <= count[3:0] - stepdown[3:0] - 6;
            end
            if (stepdown[7:4] == 0 || count[7:4] < stepdown[7:4]) begin
              if (count[7:4] < stepdown[7:4]) begin
                count[7:4] <= count[7:4] - stepdown[7:4] - 6;
              end
              if (stepdown[11:8] == 0 || count[11:8] < stepdown[11:8]) begin
                if (count[11:8] < stepdown[11:8]) begin
                  count[11:8] <= count[11:8] - stepdown[11:8] - 6;
                end
                if (stepdown[15:12] == 0 || count[15:12] < stepdown[15:12]) begin
```

```
                  if (count[15:12] < stepdown[15:12]) begin
                    count[15:12] <= count[15:12] - stepdown[15:12] - 6;
                  end
                  if (stepdown[19:16] == 0 || count[19:16] < stepdown[19:16]) begin
                    if (count[19:16] < stepdown[19:16]) begin
                      count[19:16] <= count[19:16] - stepdown[19:16] - 6;
                    end
                    if (stepdown[23:20] == 0 || count[23:20] < stepdown[23:20]) begin
                      if (count[23:20] < stepdown[23:20]) begin
                        count[23:20] <= count[23:20]   stepdown[23:20] - 6;
                      end
                      if (stepdown[27:24] == 0 || count[27:24] < stepdown[27:24]) begin
                        if (count[27:24] < stepdown[27:24]) begin
                          count[27:24] <= count[27:24] - stepdown[27:24] - 6;
                        end
                        if (count[31:28] < stepdown[31:28]) begin
                          count[31:28] <= count[31:28] - stepdown[31:28] - 6;
                        end else begin
                          count[31:28] <= count[31:28] - stepdown[31:28];
                        end
                      end else begin
                        count[27:24] <= count[27:24] - stepdown[27:24];
                      end
                    end else begin
                      count[23:20] <= count[23:20] - stepdown[23:20];
                    end
                  end else begin
                    count[19:16] <= count[19:16] - stepdown[19:16];
                  end
                end else begin
                  count[15:12] <= count[15:12] - stepdown[15:12];
                end
              end else begin
                count[11:8] <= count[11:8] - stepdown[11:8];
              end
            end else begin
              count[7:4] <= count[7:4] - stepdown[11:8];
            end
          end else begin
            count[3:0] <= count[3:0] - stepdown[3:0];
          end
        end else begin
          count <= min;
        end
      end else begin
        clkdiv <= clkdiv - 1;
      end
    end
  end

  if (count > max) begin
    count <= max;
  end

  if (count < min) begin
    count <= min;
  end

  prevzbus2=zbus2temp;
  prevzbus3=zbus3temp;
end

endmodule
```

Listing A.1: Verilog implementation of fast control mode, stepping up

A.3 TTL-Controlled Step Down, Ramp Up Mode

```
‘default_nettype none
‘timescale 1ns / 1ps

module FastControl(
    input  wire [7:0]  hi_in,
    output wire [1:0]  hi_out,
    inout  wire [15:0] hi_inout,

     input clk,
     input button,
     output wire [7:0] led,
    output wire [31:0] ybus,
    input [13:2] zbus
     );

reg [31:0] clkdiv; //Meant to slow down ramping down - a step down is taken every time clkdiv wraps around
reg [31:0] maxclkdiv; //Value that clkdiv wraps around to.
reg [31:0] count;
reg [7:0] decade;
reg [31:0] step; //Step size subtracted
reg [31:0] stepdown; //Step size added ramped
reg [7:0] index; //Iterator dummy variable in for loop to implement multi integer steps
reg [31:0] max; //Maximum to saturate at
reg [31:0] min; //Minimum to saturate at
reg satHi; //Boolean dictating if the system is saturated high
reg satLo; //Boolean dictating if the system is saturated low

reg trigger2; // Triggering booleans
reg trigger3;

reg prevzbus2;
reg prevzbus3;

reg goUp;
reg ramp;
reg goToMin;

reg rampingDown; //Boolean dictating if the system is in auto ramp-down mode
reg rampingUp; //Boolean dictating if the system is in auto ramp-up mode

reg [15:0] numberSteps; //Counter of how many steps are remaining in the step-down process

reg zbus2temp;
reg zbus3temp;



// Opal Kelly Module Interface Connections
wire        ti_clk;
wire [30:0] ok1;
wire [16:0] ok2;

// Endpoint connections:
wire [15:0]  ep00wire;//Starting #LSB (AKA MIN)
wire [15:0]  ep01wire;//Starting #MSB
wire [15:0]  ep02wire;//Step1 LSB's
wire [15:0]  ep03wire;//Step2 MSB's
wire [15:0]  ep04wire;//Ending #LSB (AKA MAX)
wire [15:0]  ep05wire;//Ending #MSB

wire [15:0]  ep06wire;//stepdown LSB
wire [15:0]  ep07wire;//stepdown MSB

wire [15:0]  ep08wire;//Number of steps during ramp
wire [15:0]  ep09wire;//Maxclkdiv LSB
wire [15:0]  ep10wire;//Maxclkdiv MSB

wire [15:0]  ep20wire;
wire [15:0]  ep21wire;




//wire [31:0]  ep20wire;//Out?

//assign led = ~max[31:24];

assign led = ~count[31:24];
//assign led[7:2] = ~count[31:26];
//assign led[1] = ~ramp;
//assign led[0] = ~goUp;
assign ybus[31:0] = ~count;

assign ep20wire = ~ybus[15:0];
assign ep21wire = ~ybus[31:16];


// Instantiate the okHost and connect endpoints.
wire [17*2-1:0]  ok2x;
okHost okHI(
  .hi_in(hi_in), .hi_out(hi_out), .hi_inout(hi_inout), .ti_clk(ti_clk),
  .ok1(ok1), .ok2(ok2));

okWireOR # (.N(2)) wireOR (.ok2(ok2), .ok2s(ok2x));


//Opal Kelly Wires from PC
```

```
okWireIn     ep00 (.ok1(ok1),                              .ep_addr(8'h00),
    .ep_dataout(ep00wire));
okWireIn     ep01 (.ok1(ok1),                              .ep_addr(8'h01),
    .ep_dataout(ep01wire));
okWireIn     ep02 (.ok1(ok1),                              .ep_addr(8'h02),
    .ep_dataout(ep02wire));
okWireIn     ep03 (.ok1(ok1),                              .ep_addr(8'h03),
    .ep_dataout(ep03wire));
okWireIn     ep04 (.ok1(ok1),                              .ep_addr(8'h04),
    .ep_dataout(ep04wire));
okWireIn     ep05 (.ok1(ok1),                              .ep_addr(8'h05),
    .ep_dataout(ep05wire));
okWireIn     ep06 (.ok1(ok1),                              .ep_addr(8'h06),
    .ep_dataout(ep06wire));
okWireIn     ep07 (.ok1(ok1),                              .ep_addr(8'h07),
    .ep_dataout(ep07wire));
okWireIn     ep08 (.ok1(ok1),                              .ep_addr(8'h08),
    .ep_dataout(ep08wire));
okWireIn     ep09 (.ok1(ok1),                              .ep_addr(8'h09),
    .ep_dataout(ep09wire));
okWireIn     ep10 (.ok1(ok1),                              .ep_addr(8'h10),
    .ep_dataout(ep10wire));

okWireOut    ep20 (.ok1(ok1), .ok2(ok2x[ 0*17 +: 17 ]), .ep_addr(8'h20),
     .ep_datain(ep20wire));
okWireOut    ep21 (.ok1(ok1), .ok2(ok2x[ 1*17 +: 17 ]), .ep_addr(8'h21),
     .ep_datain(ep21wire));



always @(posedge clk) begin
  zbus2temp <= zbus[2];
  zbus3temp <= zbus[3];

  //Wire In's
  step [31:16] <= ep03wire;
step [15:0] <= ep02wire;
max [31:16] <= ep05wire;
max [15:0] <= ep04wire;
min [31:16] <= ep01wire;
min [15:0] <= ep00wire;
stepdown [31:16] <= ep07wire;
stepdown [15:0] <= ep06wire;
maxclkdiv [15:0] <=ep09wire;
maxclkdiv [31:16] <=ep10wire;

//if (count[31:0] == 0) begin
//   count[31:0] <= max;
//end


//Saturation
if (count >= max) begin
  satHi=1;
end else begin
  satHi=0;
end

if (count <= min) begin
  satLo=1;
end else begin
  satLo=0;
end

//Posedges

if ((zbus2temp == 1) && (prevzbus2 == 0)) begin
  trigger2 <= 1'b1;
end else begin
  trigger2 <= 0;
end

```

```
  if ((zbus3temp == 1) && (prevzbus3 == 0)) begin
    trigger3 <= 1;
  end else begin
    trigger3 <= 0;
  end


  //STEP Down trigger + step
  if (trigger2 == 1) begin

    rampingDown <= 0;
    rampingUp <= 0;
if (satLo == 0) begin
     //StepDown
          if (stepdown[3:0] == 0 || count[3:0] < stepdown[3:0]) begin
            if (count[3:0] < stepdown[3:0]) begin
              count[3:0] <= count[3:0] - stepdown[3:0] - 6;
            end
            if (stepdown[7:4] == 0 || count[7:4] < stepdown[7:4]) begin
              if (count[7:4] < stepdown[7:4]) begin
                count[7:4] <= count[7:4] - stepdown[7:4] - 6;
              end
              if (stepdown[11:8] == 0 || count[11:8] < stepdown[11:8]) begin
                if (count[11:8] < stepdown[11:8]) begin
                  count[11:8] <= count[11:8] - stepdown[11:8] - 6;
                end
                if (stepdown[15:12] == 0 || count[15:12] < stepdown[15:12]) begin
                  if (count[15:12] < stepdown[15:12]) begin
                    count[15:12] <= count[15:12] - stepdown[15:12] - 6;
                  end
                  if (stepdown[19:16] == 0 || count[19:16] < stepdown[19:16]) begin
                    if (count[19:16] < stepdown[19:16]) begin
```

```
        count[19:16] <= count[19:16] - stepdown[19:16] - 6;
      end
      if (stepdown[23:20] == 0 || count[23:20] < stepdown[23:20]) begin
        if (count[23:20] < stepdown[23:20]) begin
          count[23:20] <= count[23:20] - stepdown[23:20] - 6;
        end
        if (stepdown[27:24] == 0 || count[27:24] < stepdown[27:24]) begin
          if (count[27:24] < stepdown[27:24]) begin
            count[27:24] <= count[27:24] - stepdown[27:24] - 6;
          end
          if (count[31:28] < stepdown[31:28]) begin
            count[31:28] <= count[31:28] - stepdown[31:28] - 6;
          end else begin
            count[31:28] <= count[31:28] - stepdown[31:28];
          end
        end else begin
          count[27:24] <= count[27:24] - stepdown[27:24];
        end
      end else begin
        count[23:20] <= count[23:20] - stepdown[23:20];
      end
    end else begin
      count[19:16] <= count[19:16] - stepdown[19:16];
    end
  end else begin
    count[15:12] <= count[15:12] - stepdown[15:12];
  end
end else begin
  count[11:8] <= count[11:8] - stepdown[11:8];
```

```
              end
            end else begin
              count[7:4] <= count[7:4] - stepdown[11:8];
            end
          end else begin
            count[3:0] <= count[3:0] - stepdown[3:0];
          end
        end else begin
          count <= min;
        end
  end


//  //RAMP Up Trigger
  if (trigger3 == 1) begin
    rampingUp <= 1;
    numberSteps[15:0] <= ep08wire;
  end

  //RAMP UP Steps
  if (rampingUp == 1) begin
    if (count >= max) begin
      rampingUp <= 0;
    end else begin
      if (clkdiv == 0) begin
        clkdiv <= maxclkdiv;
     //StepUP
      if (satHi == 0) begin
      if (step[3:0] == 0 || count[3:0] + step[3:0] > 9) begin
        if (count[3:0] + step[3:0] > 9) begin
            count[3:0] <= count[3:0] + step[3:0] - 10;
        end
        if (step[7:4] == 0 || count[7:4] + step[7:4] > 9) begin
          if (count[7:4] + step[7:4] > 9) begin
            count[7:4] <= count[7:4] + step[7:4] - 10;
          end
          if (step[11:8] == 0 || count[11:8] + step[11:8] > 9) begin
            if (count[11:8] + step[11:8] > 9) begin
              count[11:8] <= count[11:8] + step[11:8] - 10;
            end
            if (step[15:12] == 0 || count[15:12] + step[15:12] > 9) begin
              if (count[15:12] + step[15:12] > 9) begin
                count[15:12] <= count[15:12] + step[15:12] - 10;
              end
              if (step[19:16] == 0 || count[19:16] + step[19:16] > 9) begin
                if (count[19:16] + step[19:16] > 9) begin
                  count[19:16] <= count[19:16] + step[19:16] - 10;
                end
                if (step[23:20] == 0 || count[23:20] + step[23:20] > 9) begin
                  if (count[23:20] + step[23:20] > 9) begin
                    count[23:20] <= count[23:20] + step[23:20] - 10;
                  end
                  if (step[27:24] == 0 || count[27:24] + step[27:24] > 9) begin
                    if (count[27:24] + step[27:24] > 9) begin
                      count[27:24] <= count[27:24] + step[27:24] - 10;
                    end
                    if (count[31:28] + step[31:28] > 9) begin
                        count[31:28] <= count[31:28] + step[31:28] - 10;
                    end else begin
                      count[31:28] <= count[31:28] + step[31:28];
                    end
                  end else begin
                    count[27:24] <= count[27:24] + step[27:24];
                  end
                end else begin
                  count[23:20] <= count[23:20] + step[23:20];
                end
              end else begin
```

```
                count[19:16] <= count[19:16] + step[19:16];
              end
            end else begin
              count[15:12] <= count[15:12] + step[15:12];
            end
          end else begin
            count[11:8] <= count[11:8] + step[11:8];
          end
        end else begin
          count[7:4] <= count[7:4] + step[11:8];
        end
      end else begin
        count[3:0] <= count[3:0] + step[3:0];
      end
    end else begin
      count <= max;
    end
      end else begin
        clkdiv <= clkdiv - 1;
      end
    end
  end

  if (count > max) begin
    count <= max;
  end

  if (count < min) begin
    count <= min;
  end

  prevzbus2=zbus2temp;
  prevzbus3=zbus3temp;
end

endmodule
```

Listing A.2: Verilog implementation of fast control mode, stepping down

A.4 Fully Computer Controlled Mode

```
`default_nettype none
`timescale 1ns / 1ps

module SlowControl(
    input  wire [7:0]  hi_in,
    output wire [1:0]  hi_out,
    inout  wire [15:0] hi_inout,

     input clk,
     input button,
     output wire [7:0] led,
    output wire [31:0] ybus,
    input [13:2] zbus
     );

reg [31:0] clkdiv; //Meant to slow down ramping down - a step down is taken every time clkdiv wraps around
reg [31:0] maxclkdiv; //Value that clkdiv wraps around to.
reg [31:0] count;
reg [7:0] decade;
reg [31:0] step; //Step size added (up)
reg [31:0] stepdown; //Step size subtracted (down)
reg [7:0] index; //Iterator dummy variable in for loop to implement multi integer steps
reg [31:0] freq; //Maximum to saturate at
reg [31:0] min; //Minimum to saturate at
reg satHi; //Boolean dictating if the system is saturated high
reg satLo; //Boolean dictating if the system is saturated low

reg trigger2; // Triggering booleans
reg trigger3;

reg prevgoToFreq;
reg prevramp;
reg prevgoToMin;
reg prevzbus2;
reg prevzbus3;
reg prevzbus4;
reg prevzbus5;

reg goToFreq;
reg ramp;
reg goToMin;

reg rampingDown; //Boolean dictating if the system is in auto ramp-down mode
reg rampingUp; //Boolean dictating if the system is in auto ramp-up mode

reg [15:0] numberSteps; //Counter of how many steps are remaining in the step-down process

// Opal Kelly Module Interface Connections
wire          ti_clk;
wire [30:0] ok1;
wire [16:0] ok2;

// Endpoint connections:
wire [15:0]  ep00wire;//Starting #LSB (AKA MIN)
wire [15:0]  ep01wire;//Starting #MSB
wire [15:0]  ep02wire;//Step1 LSB's
wire [15:0]  ep03wire;//Step2 MSB's
wire [15:0]  ep04wire;//Frequency #LSB
wire [15:0]  ep05wire;//Frequency #MSB

wire [15:0]  ep06wire;//stepdown LSB
wire [15:0]  ep07wire;//stepdown MSB

wire [15:0]  ep08wire;//Number of steps during ramp
wire [15:0]  ep09wire;//Maxclkdiv LSB
wire [15:0]  ep10wire;//Maxclkdiv MSB

wire ep11wire;//Trigger Go to Frequency
wire ep12wire;//Trigger Ramp

wire [15:0]  ep20wire;
wire [15:0]  ep21wire;


assign led = ~count[31:24];
assign ybus[31:0] = ~count;

assign ep20wire = ~ybus[15:0];
assign ep21wire = ~ybus[31:16];


// Instantiate the okHost and connect endpoints.
wire [17*2-1:0]  ok2x;
okHost okHI(
  .hi_in(hi_in), .hi_out(hi_out), .hi_inout(hi_inout), .ti_clk(ti_clk),
  .ok1(ok1), .ok2(ok2));

okWireOR # (.N(2)) wireOR (.ok2(ok2), .ok2s(ok2x));


//Opal Kelly Wires from PC
okWireIn     ep00 (.ok1(ok1),                              .ep_addr(8'h00),
    .ep_dataout(ep00wire));
okWireIn     ep01 (.ok1(ok1),                              .ep_addr(8'h01),
    .ep_dataout(ep01wire));
okWireIn     ep02 (.ok1(ok1),                              .ep_addr(8'h02),
    .ep_dataout(ep02wire));
```

```
okWireIn     ep03 (.ok1(ok1),                          .ep_addr(8'h03),
    .ep_dataout(ep03wire));
okWireIn     ep04 (.ok1(ok1),                          .ep_addr(8'h04),
    .ep_dataout(ep04wire));
okWireIn     ep05 (.ok1(ok1),                          .ep_addr(8'h05),
    .ep_dataout(ep05wire));
okWireIn     ep06 (.ok1(ok1),                          .ep_addr(8'h06),
    .ep_dataout(ep06wire));
okWireIn     ep07 (.ok1(ok1),                          .ep_addr(8'h07),
    .ep_dataout(ep07wire));
okWireIn     ep08 (.ok1(ok1),                          .ep_addr(8'h08),
    .ep_dataout(ep08wire));
okWireIn     ep09 (.ok1(ok1),                          .ep_addr(8'h09),
    .ep_dataout(ep09wire));
okWireIn     ep10 (.ok1(ok1),                          .ep_addr(8'h10),
    .ep_dataout(ep10wire));
okWireIn     ep11 (.ok1(ok1),                          .ep_addr(8'h11),
    .ep_dataout(ep11wire));
okWireIn     ep12 (.ok1(ok1),                          .ep_addr(8'h12),
    .ep_dataout(ep12wire));
okWireIn     ep13 (.ok1(ok1),                          .ep_addr(8'h13),
    .ep_dataout(ep13wire));

okWireOut    ep20 (.ok1(ok1), .ok2(ok2x[ 0*17 +: 17 ]), .ep_addr(8'h20),
     .ep_datain(ep20wire));
okWireOut    ep21 (.ok1(ok1), .ok2(ok2x[ 1*17 +: 17 ]), .ep_addr(8'h21),
     .ep_datain(ep21wire));


always @(posedge clk) begin
  //Wire In's
  step [31:16] <= ep03wire;
  step [15:0] <= ep02wire;
  freq [31:16] <= ep05wire;
  freq [15:0] <= ep04wire;
  min [31:16] <= ep01wire;
min [15:0] <= ep00wire;
stepdown [31:16] <= ep07wire;
stepdown [15:0] <= ep06wire;
maxclkdiv [15:0] <=ep09wire;
maxclkdiv [31:16] <=ep10wire;


goToFreq <= ep11wire;
ramp <= ep12wire;

//Saturation
if (count <= min) begin
  satLo=1;
end else begin
  satLo=0;
end

//Posedges

if ((goToFreq == 1) && (prevgoToFreq == 0)) begin
  trigger2 <= 1'b1;
end else begin
  trigger2 <= 0;
end

if ((ramp == 1) && (prevramp == 0)) begin
  trigger3 <= 1;
end else begin
  trigger3 <= 0;
end


//STEP UP trigger + step
if (trigger2 == 1) begin
  rampingDown <= 0;
  if (freq > min) begin
```

```
      count <= freq;
    end else begin
      count <= min;
    end
  end


//  //RAMP Down Trigger
  if (trigger3 == 1) begin
    rampingDown <= 1;
    numberSteps[15:0] <= ep08wire;
  end

  //RAMP Down Steps
  if (rampingDown == 1) begin
    if (numberSteps == 0) begin
      rampingDown <= 0;
    end else begin
      if (clkdiv == 0) begin
        clkdiv <= maxclkdiv;
        if (satLo == 0) begin
     //StepDown
          if (stepdown[3:0] == 0 || count[3:0] < stepdown[3:0]) begin
            if (count[3:0] < stepdown[3:0]) begin
              count[3:0] <= count[3:0] - stepdown[3:0] - 6;
            end
            if (stepdown[7:4] == 0 || count[7:4] < stepdown[7:4]) begin
              if (count[7:4] < stepdown[7:4]) begin
                count[7:4] <= count[7:4] - stepdown[7:4] - 6;
              end
              if (stepdown[11:8] == 0 || count[11:8] < stepdown[11:8]) begin
                if (count[11:8] < stepdown[11:8]) begin
                  count[11:8] <= count[11:8] - stepdown[11:8] - 6;
                end
                if (stepdown[15:12] == 0 || count[15:12] < stepdown[15:12]) begin
                  if (count[15:12] < stepdown[15:12]) begin
                    count[15:12] <= count[15:12] - stepdown[15:12] - 6;
                  end
                  if (stepdown[19:16] == 0 || count[19:16] < stepdown[19:16]) begin
                    if (count[19:16] < stepdown[19:16]) begin
                      count[19:16] <= count[19:16] - stepdown[19:16] - 6;
                    end
                    if (stepdown[23:20] == 0 || count[23:20] < stepdown[23:20]) begin
                      if (count[23:20] < stepdown[23:20]) begin
                        count[23:20] <= count[23:20] - stepdown[23:20] - 6;
                      end
                      if (stepdown[27:24] == 0 || count[27:24] < stepdown[27:24]) begin
                        if (count[27:24] < stepdown[27:24]) begin
                          count[27:24] <= count[27:24] - stepdown[27:24] - 6;
                        end
                        if (count[31:28] < stepdown[31:28]) begin
                          count[31:28] <= count[31:28] - stepdown[31:28] - 6;
                        end else begin
                          count[31:28] <= count[31:28] - stepdown[31:28];
                        end
                      end else begin
                        count[27:24] <= count[27:24] - stepdown[27:24];
                      end
                    end else begin
                      count[23:20] <= count[23:20] - stepdown[23:20];
                    end
                  end else begin
                    count[19:16] <= count[19:16] - stepdown[19:16];
                  end
                end else begin
                  count[15:12] <= count[15:12] - stepdown[15:12];
                end
              end else begin
                count[11:8] <= count[11:8] - stepdown[11:8];
              end
            end else begin
              count[7:4] <= count[7:4] - stepdown[11:8];
            end
          end else begin
            count[3:0] <= count[3:0] - stepdown[3:0];
          end
        end else begin
          count <= min;
        end
        //Adjust counter of steps remaining
        numberSteps <= numberSteps - 1;
      end else begin
        clkdiv <= clkdiv - 1;
      end
    end
  end

  if (count < min) begin
    count <= min;
  end

  prevgoToFreq=goToFreq;
  prevramp=ramp;
end

endmodule
```

Listing A.3: Verilog implementation of slow control mode

Bibliography

[1] C. Ates, S. Sevincli, and T. Pohl. Electromagnetically induced transparency in strongly interacting rydberg gases. *Physical Review A*, 83(041802(R)), 2011.

[2] J. Bechhofer. Feedback for physicists: A tutorial essay on control. *Reviews of Modern Physics*, 77:783–836, 2005.

[3] E.D. Black. An introduction to pound-drever-hall laser frequency stabilization. *American Journal of Physics*, 2000.

[4] G. Brooker. *Modern Classical Optics*, chapter 7. Oxford University Press, 2003.

[5] R. Bucker. Fluorescence imaging of ultracold atoms. Master's thesis, University of Heidelberg, 2007.

[6] Y. Chu. Loading rubidium atoms into a hollow core fiber. B Sc. Thesis, MIT.

[7] Chuang and Nielsen. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

[8] J. Dalibard and C. Cohen-Tannoudji. Laser cooling below the doppler limit by polarization gradients: simple theoretical models. *JOSA B*, 6(11), 1989.

[9] S.K. Dutta, J.R. Guest, D. Feldbaum, A. Walz-Flannigan, and G. Raithel. Ponderomotive optical lattice for rydberg atoms. *PRL*, 85:55515554, 2000.

[10] M. Fleischhauer, A. Imamoglu, and J. Marangos. Electromagnetically induced transparency: Optics in coherent media. *REVIEWS OF MODERN PHYSICS*, 77(10.1103), 2005.

[11] M. Fleischhauer and M.D. Lukin.

[12] A.V. Gorshkov, J. Otterbach, M. Fleischhauer, T. Pohl, and M.D. Lukin. Photon-photon interactions via rydberg blockade alexey. *arXiv*, (1103.3700v1), 2011.

[13] M. Greiner. *Ultracold quantum gases in three-dimensional optical lattice potentials.* PhD thesis, Ludwig-Maximilians-Universitat Munchen, 2003.

[14] R. Grimm, M. Weidemuller, and Y. Ovchinnikov. Optical dipole traps for neutral atoms. *arXiv:physics/9902072v1*, 1999.

[15] G. Gunter, M. Robert-de Saint-Vincent, H. Schempp, C.S. Hofmann, S. Whitlock, and M. Weidemuller. Interaction enhanced imaging of individual rydberg atoms in dense gases. *PRL*, 108(013002), 2012.

[16] J. Jackson. *Classical Electrodynamics.*

[17] H.J. Kimble. Scalable quantum networks with atoms and photons. Caltech. tutorial given at QELS in Baltimore, MD.

[18] V. Letokhov. Saturation spectroscopy. In *Topics in Applied Physics*, pages 1–35. Springer-Verlag, 1976.

[19] B. Olmos, W. Li, S. Hofferberth, and I. Lesanovsky. Amplifying single impurities immersed in a gas of ultra cold atoms. *arXiv*, (1106.4444v2), 2011.

[20] Opal Kelly Incorporated, Hillsboro, OR. *XEM3001v2 User's Manual*, 2007.

[21] T. Petelski, M. Fattori, G. Lamporesi, J. Stuhler, and G.M. Tino. Doppler-free spectroscopy using magnetically induced dichroism of atomic vapor: a new scheme for laser frequency locking. *J. Opt. Soc. Am. B*, 6(12), 1989.

[22] Pethick and Smith. *Bose-Einstein Condensation in Dilute Gases.*

[23] J.D. Pritchard. *Cooperative Optical Non-linearity in a blockaded Rydberg Ensemble.* PhD thesis, Durham University, 2011.

[24] J.D. Pritchard, D. Maxwell, A. Gauguet, K.J. Weatherill, M.P.A. Jones, and C.S. Adams. Cooperative atom-light interaction in a blockaded rydberg ensemble. *PRL*, 105(193603), 2010.

[25] Programmed Test Sources, inc, Littleton, MA, USA. *PTS 3200 Operation Manual*, 1995.

[26] E. Raab, M. Prentiss, A. Cable, and S. Chu. Trapping of neutral sodium atoms with radiation pressure. *PRL*, 1987.

[27] A. Reinhard, T. Cubel Liebisch, B. Knuffman, and G. Raithel. Level shifts of rubidium rydberg states due to binary interactions. *Physical Review A*, 75(032712), 2007.

[28] Scully and Zubairy. *Quantum Optics*, chapter 7.3, pages 225–229. Cambridge University Press, 1997.

[29] A. Yariv. *Optical Electronics.* Saunders College Publishing, 1991.