Laser Synchronized Optical Nuclear Magnetic Resonance via Larmor Beat Detection - Imaging Electronic Wavefunctions in Gallium Arsenide Device Structures

Thesis by

Michael A. Miller

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

2001

(submitted May 24, 2001)

Acknowledgements

I would like to thank a great many people. Firstly, I thank my parents and family members for shaping the person that I am, giving me many of the requisite skills that I have needed in life, and being so supportive of my efforts here at Caltech. I also greatly appreciate the instruction, mentoring and support provided by my undergraduate chemistry professors both at the University of California at Irvine and at Saddleback College. Special thanks go out to Craig Martens, A.J. Shaka, Terence Miraglia, and Hal Freidmann. I am deeply grateful to Dan Weitekamp for allowing me to come into his lab and to be part of an awesome research team, pursuing the most intriguing research I've ever seen. I would like to thank the entire Weitekamp group for what has mostly been the time of my life. The environment provided by Margat Werner, Steve Buratto, Dan Jones, Pedro Pizarro, Jack Hwang, Paul Carson, Len Mueller, Jim Kempf, and Bruce Lambert was unparalleled. Additionally, I owe a lot to Steve Buratto for getting the optical NMR project going, Jack Hwang for getting us to 2 K in a new dewar in record time and passing the torch, and John Marohn for listening to Dan's idea about a reference field and developing the LBD detection technique. A special thank you to Jim Kempf for his assistance in lab, his leadership in jumping into subprojects like the computer data acquisition board and code, and for being such a great lab partner, including the times when I had to retreat to developing theory in order to gain some perspective on the data and the opportunities it presented.

The largest and most heart filled thank you goes to my wife Arezoo. She has put up with more than her fair share of absentee husband syndrome from the day I started at Caltech until now. It has not been easy for her, and, still, she has been extremely supportive of my efforts. She has cooked breakfast, lunch, and dinner and brought them to me in lab during candidacy, props, and thesis to assure that I wont have to waste time going to get my meals. She has let me go when she has most wanted me by her side, and for that I am eternally grateful. It is hard to make a marriage that began just before graduate school to last all of the way through it, and it is because of Arezoo that we are still together, beginning this new chapter in our lives.

Finally, I need to thank someone who has felt the loss of my presence at home as much as Arezoo, but had no recourse to change it. Mariam, our daughter, has been an incredible joy to my life, and I would like to thank her for putting up with a daddy who always has "homework" to do when she wants to go to the park, pool, playground, etcetera. She is such an incredible little girl, and she and Arezoo have been my guiding inspiration in life.

Abstract

We have accomplished Optical Nuclear Magnetic Resonance (ONMR) experiments in an $Al_{0.36}Ga_{0.64}As$/GaAs heterojunction sample at ~2K with rf-optical pulse synchronization. The hyperfine coupling of the electron spin to the nuclear spins enable this spectroscopy in several ways, which are discussed herein. Moreover, the interactions experienced by nuclear spins in III-V semiconductors, in general, and the phenomena encountered when they are in the vicinity of a shallow donor or pseudo-donor, specifically, are developed. Furthermore, the most accurate calculation of spin diffusion in a spin-three-halves system to date is developed and presented using a methodology can be readily applied to any spin-larger-than-one-half system to a yield a set of coupled differential equations for a set of orthogonal polarizations. The behavior of these equations under a number of physical situations is also investigated.

We have captured the first ever radially resolved Knight shift images from the nuclei near a point defect in GaAs using laser synchronized ONMR. A deconvolution of these images into their constituent physical interactions has been approximately carried out using the theoretical advances developed and presented in this thesis, yielding the shape and size of the electronic orbital in which the electron is trapped, the occupancy of that electronic orbital, and the quadrupolar interactions in the vicinity of the defect, including the charge state of the defect.

Computational approaches include both full, real-time analyses of every one of the hundreds of thousands of nuclei surrounding a defect in GaAs, modeling the time domain

evolution for each individual nucleus including its Knight shift, quadrupolar interactions (both secular and nonsecular), individual optical polarization conditions, optical detection weighting, and rigorously exact rf effects, and analyses of a variety of continuous medium approximations. The only computations that fit the experimental spectra are those that calculate spin diffusion along a radial line of spins, and use this approximation to the radial profile of nuclear polarization in a continuous medium approximation. The successful interface of this spin diffusion calculation and the single nucleus calculations, leveraging their individual strengths, is clearly a desirable route to further increase computational accuracy.

Table of Contents

List of Figures

List of Tables

I. Optical Nuclear Magnetic Resonance - Background

The effects resulting from the coupling of the nuclear and electronic spin systems by the contact hyperfine interaction in semiconductors and, specifically, the ramifications of optically exciting the electrons as well as monitoring their optical decay process have been studied for over thirty years[1-28]. In this chapter those aspects of the contact hyperfine coupling that pertain to optically detected NMR, optical polarization of the nuclear spin system, and Knight shifts experienced by the nuclear spins will be reviewed. Additionally, we will consider the sites at which the effects of these couplings on the nuclear spins are most likely being recorded in the optical nuclear magnetic resonance (ONMR) spectra recorded in the Weitekamp laboratory, and, indeed, in many laboratories that are studying optically relevant defects in epitaxial GaAs microstructures.

Optical nuclear polarization (ONP) in direct-gap III-V semiconductors is discussed in section A. The band structure and selection rules for interband transitions in direct-gap III-V semiconductors are briefly reviewed along with the cooling potential of the electron spin system resulting from illuminating a sample with circularly polarized light. Additionally, the contact hyperfine interaction is introduced and the magnitude of the potential cooling of the nuclear spin system by the electron spin system to which it is coupled is calculated.

The Hanle effect is introduced in section B. Moreover, its capacity for detecting nuclear spin polarization as a depolarization of the luminescence emitted by electrons during their return to the valence band is explained along with the benefits of detection of

optical photons over radio frequency photons. The Hanle effect provides the foundation of all the techniques of optical detection (OD) of NMR.

Specific detection techniques such as Time sequenced optical nuclear magnetic resonance (TSONMR) and continuous wave (CW) NMR are reviewed in section C, and compared to each other. Indeed, the work present in this thesis is based on a modified form of TSONMR. The modification is in the technique used during the detection period of the TSONMR experiment, which was previously not able to detect precessing magnetization in real time.

In section D, Larmor beat detection (LBD) is presented in detail. Both real time and point wise versions of the detection scheme are considered. The potential for unwanted nonlinearity intruding into the detected signal is developed, and detailed calculations are presented graphically for both the 99% and 90% linear signal thresholds.

The Knight shift is discussed in section E. It is a shift of the energy levels of the nuclear spins by the presence of spin-polarized electrons at the nuclear sites, and is third incarnation of the hyperfine interaction encountered in this chapter. In chronological order during an experiment it is, thus, possible to use ONP to create a large nuclear spin polarization, subject the nuclei to the Knight Shift during evolution, and detect the evolution of the nuclear spins with LBD. This is precisely the sort of experiment that we have accomplished, though with greater sophistication, and which will be presented in chapter 5 of this thesis. By so detecting the Knight shift, one has the sensitivity to the localized electronic spin density, to which the Knight shift is proportional, that the electronic wavefunction may be extracted from a careful analysis of the data.

The first experiments to combine ONP and OD were conducted by Ekimov and Safarov[2,3]. The dominant luminescence line from their sample was attributed to a donor to acceptor transition; therefore, the NMR they detected was only from nuclei in the immediate vicinity of shallow donor. This is yet another benefit of OD. That is, selective detection of a certain luminescence band provides site specificity not available when using inductive detection of the NMR. The properties of shallow donors and further motivation for their study are discussed in section F. Indeed, as will become evident in chapter 5, the model that best explains the experiments presented there is none other than the shallow donor hypothesis. Moreover, the theoretical work in chapter 3 of this thesis is centered on precisely this hypothesis and variations of it.

A. Optical Nuclear Polarization

When semiconductors, and certain other materials, absorb circularly polarized light, dramatically enhanced nuclear polarization is observed. This effect is referred to as optical nuclear polarization (ONP), and Lampel first observed it in experiments on silicon[1]. The mechanism for this effect can be understood as follows.

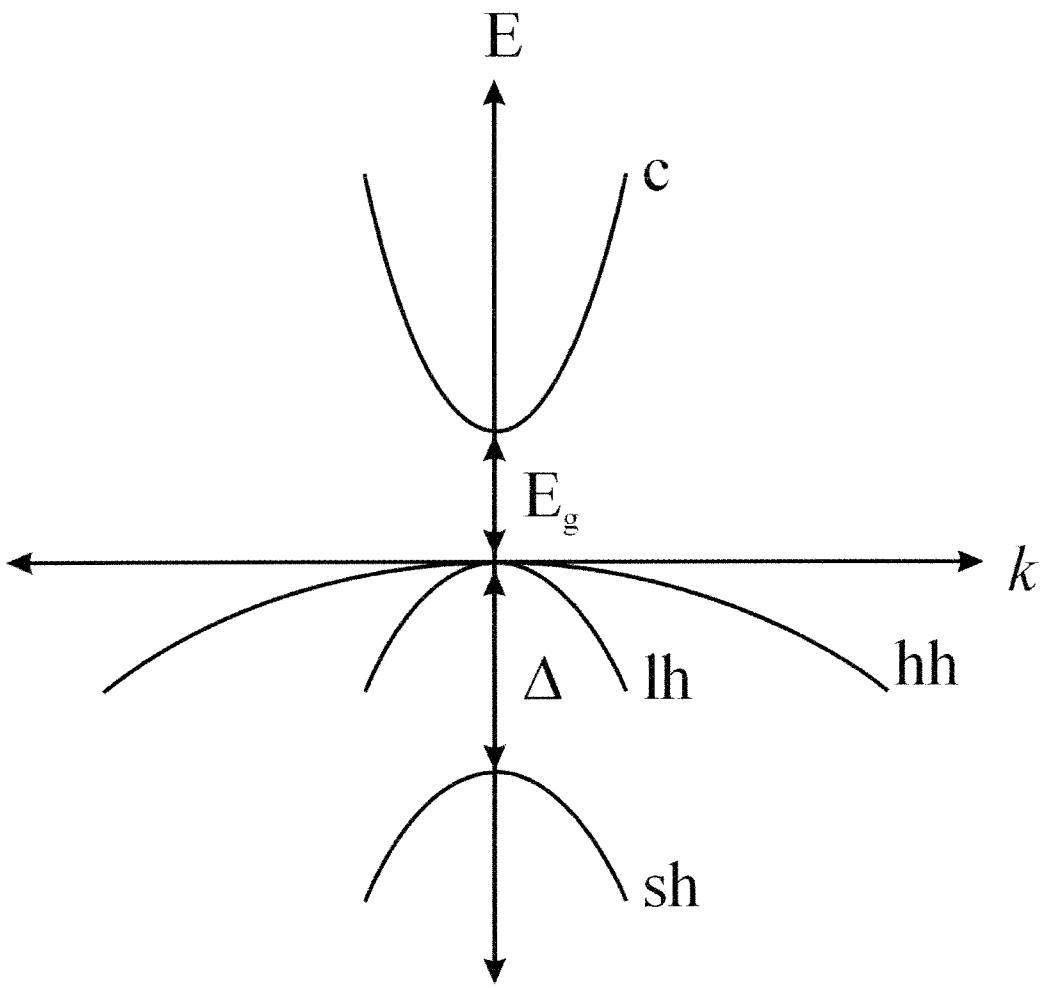


Figure 1.1 The band structure of bulk GaAs, a direct gap semiconductor. E_g is the bandgap energy, (c) denotes the conduction band with $\left(J = \frac{1}{2},\ m_j = \pm\frac{1}{2}\right)$, (hh) the heavy hole band with $\left(J = \frac{3}{2},\ m_j = \pm\frac{3}{2}\right)$, (lh) the light hole band with $\left(J = \frac{3}{2},\ m_j = \pm\frac{1}{2}\right)$, and (sh) the slit off band with $\left(J = \frac{1}{2},\ m_j = \pm\frac{1}{2}\right)$ located Δ below the valence band edge.

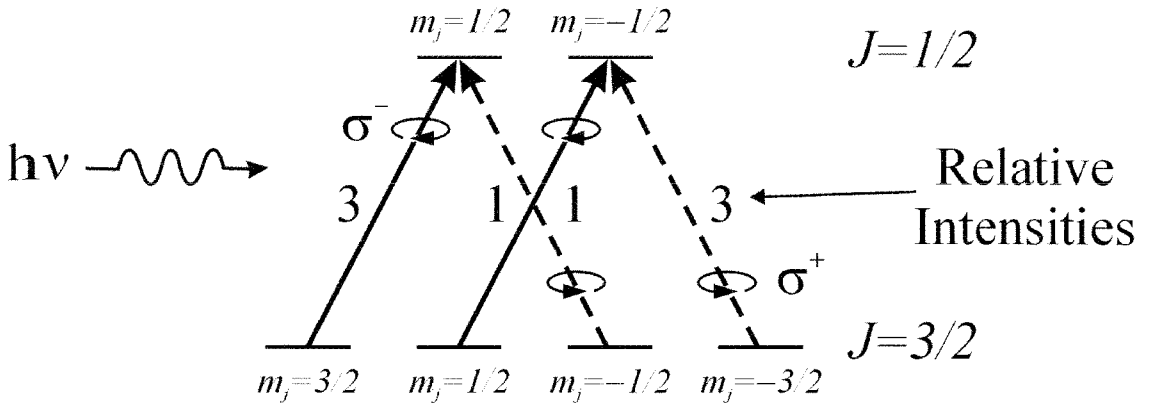


Figure 1.2 The transitions between the conduction and valence bands that are induced by exposure to circularly polarized light of energy equal to E_g. The arrows connecting the different bands depict the optically allowed transitions, their relative transition probability, and the optical polarization required for the transition. There is an additional transition available in the presence of linearly polarized light from the light hole states vertically to the conduction band state with the same value of m_j. This transition has a relative intensity of 2.

Figure 1.1 shows the energy level spacing of the electron bands in a direct gap semiconductor like GaAs while figure 1.2 shows the circularly polarized transitions available to an electron at $k = 0$. Selection rules dictate that the transition probability from either of the two spin-three-halves valence band states to the conduction band is three times higher than that of either of the spin-one-half valence band states. It is apparent that if unpolarized GaAs is exposed to circularly polarized light at the band gap frequency, that three times more electrons will enter the conduction band with one sign of electron spin than will enter with the other. That is, 50% of the conduction band electrons will be spin polarized. This corresponds to an extremely cold electron spin temperature. For example, in the ~0.25 Tesla magnetic fields used in our labs, this is the

population difference that arises spontaneously in the conduction band electrons of GaAs at a temperature of about 70 millikelvin.

Additionally, these conduction band electrons have a strong contact hyperfine coupling to the nuclear spins, expressed in Hz, using rationalized MKSA units as[29]

$$\mathcal{H}_{\mathrm{HF},\alpha} = A_\alpha \mathbf{I}\cdot\mathbf{S} \tag{1.1}$$

where

$$A_\alpha = \tfrac{2}{3}\mu_0 g_0 \mu_B \gamma_{\alpha,\mathrm{Hz}} d_\alpha \tag{1.2}$$

A_α is referred to as the hyperfine coupling of the nuclear species α. Here, α will index the specific nuclei (^{71}Ga, ^{69}Ga, or ^{75}As), and μ_0 is the vacuum magnetic permeability. The factor $g_0\mu_B$, the free electron g-factor multiplied by the Bohr magneton, is equivalent to $\gamma_{e,\mathrm{Hz}}h$, the gyromagnetic ratio of the electron multiplied by Planck's constant. Note that g_0 and not g_0^*, the effective electron *g*-factor, is used in the above equation because the effect in question is due to the Fermi contact interaction with primarily "*s*-like" states (that is, ones with spatial overlap with the nucleus). The difference between g_0 and g_0^* is attributed to the electron's orbital angular momentum, and the latter appears in the electron Larmor frequency. As usual, γ_α is the gyromagnetic ratio of the α nucleus. The quantity d_α is equal to $|\psi(\mathbf{r}_\alpha)|_\alpha^2$ which is the probability density of an electron at the nucleus located at $\mathbf{r}_\alpha$ given that the one-electron orbital ψ is occupied.

If the magnitude of this interaction is taken and the gyromagnetic ratio of the nuclear spin is divided out, one arrives at a quantity that has units of magnetic field and looks like an extra Zeeman Hamiltonian on the nucleus. This "electron-hyperfine-field" also fluctuates rapidly due to the short lifetime (picosecond time scale) of the optically excited electron in the conduction band. This allows for simultaneous nuclear and electronic spin flips, allowing the electrons a pathway to thermal equilibration with their surroundings. However, the nuclear spin lattice relaxation time is very long in these materials, occurring on a time scale of minutes to days. The nuclear spin system, therefore, is able to reach considerable thermal equilibration with the electron spin system.

Let us analyze the equilibration of the electron and nuclear spins in a large enough magnetic field that nuclear spin-spin interactions can be neglected. The hyperfine coupling conserves the total spin of the system. Therefore, if the nuclear spin is to flip between states having angular momentum μ and $\mu-1$ with respect to the magnetic field, the relaxation rates W, nuclear spin populations N, and electron spin populations n are related by

$$W_{\mu,\mu-1} N_{\mu} n_{-} = W_{\mu-1,\mu} N_{\mu-1} n_{+} . \tag{1.3}$$

However, the rates W are related to one another by the Boltzmann factor for the combined nuclear and electronic transition. Given the relative size of the nuclear gyromagnetic ratios compared to the electron gyromagnetic ratio, this factor can be taken with respect to the energy of the electronic spin flip only.

$$W_{\mu-1,\mu} = e^{-g\mu_B B/T} W_{\mu,\mu-1} \tag{1.4}$$

The temperature in equation (1.4) is the temperature of the reservoir that provides the energy for the simultaneous spin flip. This is taken as the lattice temperature, which is equivalent to saying that it is the electron spin lattice relaxation that is driving the reaction. As such, the Boltzmann factor will be approximately equal to 1 in all of the following work. This leads us to conclude that ratio of the populations of the each of the adjacent nuclear spin levels will achieve the same ratio as those of the optically polarized electrons.

$$\frac{N_\mu}{N_{\mu-1}} = \frac{n_+}{n_-} \tag{1.5}$$

This is a truly remarkable result since it says that, in principle, the optical polarization that was imparted to the electron spin system can be transferred to the nuclear spin system in its entirety. This corresponds to our bath of conduction band electrons at about 70 millikelvin refrigerating the nuclear spin system to a temperature of approximately 100 microkelvin. It is worth noting that equation (1.3), and any equation derived from it, leads to an optically pumped nuclear spin system that can be described by a single spin temperature.

B. Optical Detection - The Hanle Effect

The electronic transitions from the conduction band to the valence band can be used in another context, as probes of the state of nuclear polarization. The process of

optical detection (OD) uses optical photons to detect NMR evolution occurring at radio frequencies. The energy of the optical photon is much higher than the energy of the radio frequency photon, and this improves signal to noise in two ways. First, sensitive photo-detectors may be used to detect the photon instead of inductively coupled coils. Secondly, the thermal background of radio frequencies guarantees a much higher noise level than in the optically detected case. That is, the blackbody radiation from the sample and experimental equipment at room temperature has only a negligible portion of its high frequency tail in the optical region while radio frequencies are located in the central bulge of this blackbody curve.

Optical detection is performed in this work, as in most of ONMR, by monitoring the circular polarization of the luminescence arising from decay of optically excited electrons back to the valence band. Returning to figure 1.1, one can see that, due to conservation of angular momentum, each of the transitions carries with it the restriction that the exciting light must provide the difference in angular momentum between the initial and final state. In other words, the angular momentum of the exciting light must be absorbed by the electron in its transition to its final state.

These same selection rules now govern the transitions back down to the valence band. If only these selection rules are taken into account, one arrives at the conclusion that the polarization of the light ρ is the normalized projection of the average electron spin $\mathbf{S}$ onto the direction of observation $\boldsymbol{n}_o$.

$$\rho = -\mathbf{S} \cdot \boldsymbol{n}_o \tag{1.6}$$

This relationship is valid when dealing with reasonably p-type samples, so that hole dynamics and spin dependent recombination can safely be ignored. Additionally, when the electron decays back to the valence band, it will emit luminescence and this emission will be spatially isotropic. This is equivalent to saying that each electron radiates as if it were a rotating magnetic dipole with its axis of rotation aligned along the direction of its spin. That is, the light field will carry away one unit of angular momentum so photons will be left circularly polarized when observed from the direction where the electron spin is oriented, right circularly polarized when observed from the opposite direction, linearly polarized when observed from anywhere in the plane orthogonal to the electron spin, and elliptically polarized when observed from a direction between any of these.

Moreover, the presence of a magnetic field oriented transverse to the electron spin will cause a reduction in **S** due to precession of the electron spin about the transverse field. If the precession is rapid over the average lifetime of an electron, the electron will have a negligible average spin orientation. This is known as the Hanle effect, and figure 1.3 depicts the Hanle effect graphically where the screw axis is time and the magnetization is rotating in the *yz*-plane. The average spin of the electrons is initially oriented along the *z*-axis, and the presence of a transverse field along the *x*-axis causes precession in the *yz*-plane.

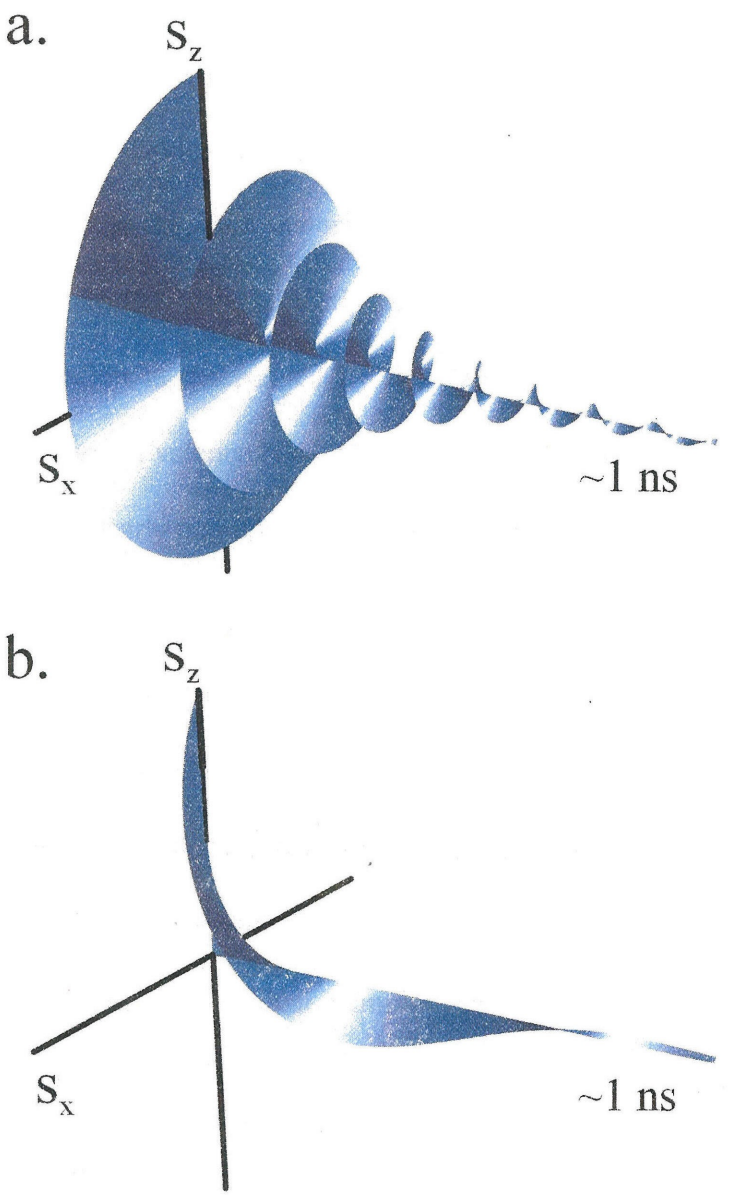


Figure 1.3 A graphical illustration of the Hanle effect. a) When the precession period is short compared to the electron spin relaxation time, the electron spin is rapidly changing orientation, yielding a small average polarization. b) When the precession period is long compared to the electron spin relaxation, the orientation of the electron spin is roughly constant during its lifetime, yielding a large average polarization. The observed luminescence polarization is a measure of the average lifetime of the electron spin polarization. The more precession periods there are before decay, the smaller the resulting luminescence polarization will be.

When the precession frequency is low enough that most electrons decay before they can precess an appreciable amount, the average electron spin is not significantly reduced. However, when the vast majority of electrons precess many times before decaying, the average electron spin is nearly zero. Equivalently, each electron emits a photon whose polarization is determined by the direction of the electron spin at the moment of decay back to the valence band, yet, when the electron spin is nearly randomly oriented in the *yz*-plane, nearly unpolarized luminescence is produced.

One may quantify these concepts by analyzing the optical pumping cycle of the electrons in terms of the electron Bloch equation. Presently, this approach will be used to derive the equation for the polarization of the conduction electrons and explain the Hanle effect. If **S** is defined to be the electron spin magnetization, and if P is the electron polarization as defined below, then the following rate equations describe the dynamics of the electronic system during the optical pumping cycle. In these equations τ is the electron-hole recombination time, T_1 is the electron spin relaxation time, $\mathbf{S}_0/\tau$ is the rate at which electrons are being promoted to the conduction band through optical pumping with σ^+ polarized light written in terms of $\mathbf{S}_0$, the electron spin magnetization in the case where $T_1 = \infty$ and $\mathbf{B}_T = 0$, and, finally, $\mathbf{B}_T$ is a magnetic field that is transverse to $\mathbf{S}_0$. It should be noted that $\mathbf{S}_0$ is the maximum possible value of **S**, and that, for optical pumping with the quantization axis of the light along the z-axis, $\mathbf{S}_0$ is pointed along the z-axis as well.

$$P(B)=\frac{\mathbf{S}_z(B)}{\mathbf{S}_0} \tag{1.7}$$

$$\frac{d\mathbf{S}}{dt}=\omega\times\mathbf{S}+\frac{\mathbf{S}_0}{\tau}-\frac{\mathbf{S}}{\tau}-\frac{\mathbf{S}}{T_1} \tag{1.8}$$

The steady state solution $\left(\frac{d\mathbf{S}}{dt}=0\right)$ for $\omega = 0$ is as follows

$$0=\frac{\mathbf{S}_0}{\tau}-\left(\frac{1}{\tau}+\frac{1}{T_1}\right)\mathbf{S}_z(0) \tag{1.9}$$

$$\mathbf{S}_z(0)=\frac{\mathbf{S}_0}{\tau\left(\frac{1}{\tau}+\frac{1}{T_1}\right)}=\frac{\mathbf{S}_0}{1+\frac{\tau}{T_1}} \tag{1.10}$$

Thus,

$$P(0)=\frac{\mathbf{S}_z(0)}{\mathbf{S}_0}=\frac{T_1}{T_1+\tau} \tag{1.11}$$

This is the well-known master equation for the polarization of the electrons in the conduction band at zero magnetic field, again neglecting hole dynamics and spin dependent recombination. Moreover, the steady state solution $\left(\frac{d\mathbf{S}}{dt}=0\right)$ for $\omega=\frac{\mu_B g^*\mathbf{B}_T}{\hbar}$, $\omega=\|\omega\|=\frac{\mu_B g^* B_T}{\hbar}$, where $\frac{1}{T_s}=\frac{1}{\tau}+\frac{1}{T_1}$, is

$$\frac{d\mathbf{S}}{dt}=\frac{\mu_B g^*}{\hbar}\mathbf{B}_T\times\mathbf{S}+\frac{\mathbf{S}_0}{\tau}-\left(\frac{1}{\tau}+\frac{1}{T_1}\right)\mathbf{S} \tag{1.12}$$

$$0 = \omega \times \mathbf{S} - \frac{\mathbf{S} - \mathbf{S}_z(0)}{T_s} \tag{1.13}$$

$$S_z(\omega) = \frac{S_z(0)}{1 + T_s^2 \omega^2} \tag{1.14}$$

The depolarization of the luminescence is, therefore, a Lorentzian that is centered at zero and whose width at half height is $\omega_{1/2} = 1/T_s$ or, in terms of magnetic field, $B_H = B_{1/2} = \hbar/(\mu g^* T_s)$. This allows us to write

$$S_z(B_T) = \frac{S_z(0)}{1 + (B_T/B_H)^2} \tag{1.15}$$

Therefore, as the transverse magnetic field grows larger, the electrons lose their polarization. If ω is large compared to T_S, the electron spin precesses many times before decay, producing nearly depolarized luminescence.

Moreover, the Hanle width will change when an extra magnetic field aligned along the direction of the electron spin orientation is present. This is referred to as the incomplete Hanle effect, and can be understood by reconsidering equations (1.12) through (1.15). The electron will now precess around the effective magnetic field, which is the vector sum of the transverse and longitudinal magnetic fields. Thus, equation (1.15) may be rewritten as

$$\mathbf{S}_z(\mathbf{B}_T) = \frac{\mathbf{S}_z(0)}{1 + B_T^2/(B_H^2 + B_z^2)} \tag{1.16}$$

That is, the transverse field must be large compared to both the longitudinal field and the Hanle width if it is to result in an effective field that is both perpendicular to the electron spins and large enough to depolarize these spins. Note that the longitudinal field includes both the Zeeman magnetic field and any nuclear hyperfine fields that have been left along the Zeeman field.

The general solution for the average electron spin orientation as a function of an arbitrary magnetic field B has been derived by Fleisher and Merkulov[15] as

$$\mathbf{S} = \frac{B_{\mathrm{H}}^2 \mathbf{S}_0 + (\mathbf{B}\cdot\mathbf{S}_0)\mathbf{B} + (\mathbf{B}\times\mathbf{S}_0)B_{\mathrm{H}}}{B_{\mathrm{H}}^2 + B^2} \tag{1.17}$$

Equation (1.17) reduces to our previous expressions when evaluated under the specific conditions for which they were derived, yet it is far more general.

Now, returning to the concept developed in an earlier section of the electron acting as an extra magnetic field on the nucleus through their mutual hyperfine interaction, we may instead view this interaction from the standpoint of the electron as the nucleus acting as an extra magnetic field on the electron. Therefore, if the optically pumped nuclear spins are reoriented to an alignment perpendicular to the electron spins, they will precess at a frequency proportional to the size of this nuclear hyperfine field, producing a quantifiable depolarization of the luminescence. Thus, the size of the nuclear magnetization in the region of the electronic localization may be measured using the depolarization of optical photons.

C. Time Sequenced ONMR

The early work in optical detection of optically polarized NMR (ONMR) was all done using continuous wave (CW) NMR. Optical detection of this technique, where one slowly sweeps a weak radio frequency magnetic field (rf) though the resonance frequency of the nuclei, was performed with optical excitation on continuously though out the experiment to provide continuous measurement of the NMR. However, the presence of optically polarized electrons during the experiment lead to shifting and broadening the resulting NMR due to the presence of the hyperfine coupling between the electrons and nuclei during evolution. Indeed, the best NMR linewidth in all of these works is approximately 50 kHz[3].

A great improvement over these experiments is one in which the different pieces of the experiment are segregated in time, so that each may act on its own without interference or complication induced by the

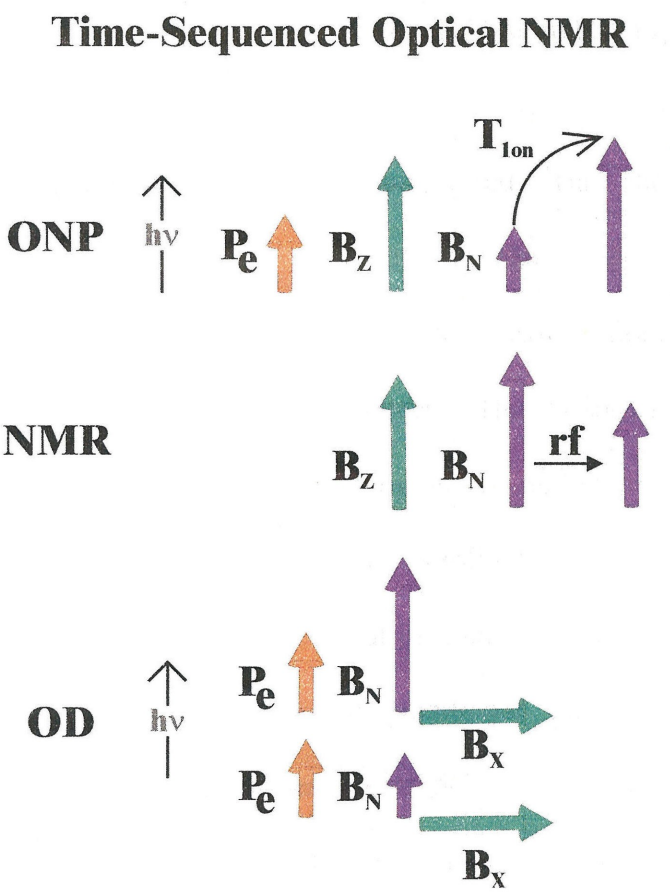


Figure 1.4 A schematic illustration of the steps in the TSONMR experiment.

others. The natural periods into which the experiment should be broken are ONP, NMR evolution, and OD. As shown in figure 1.4, the sample first undergoes optical nuclear polarization, and the exciting light is then turned off. The NMR experiment is now performed over some evolution time t1, and the portion of the magnetization that one desires to measure is then stored along the z-axis of the magnetic field. The light is then turned back on, so that optical detection can be carried out.

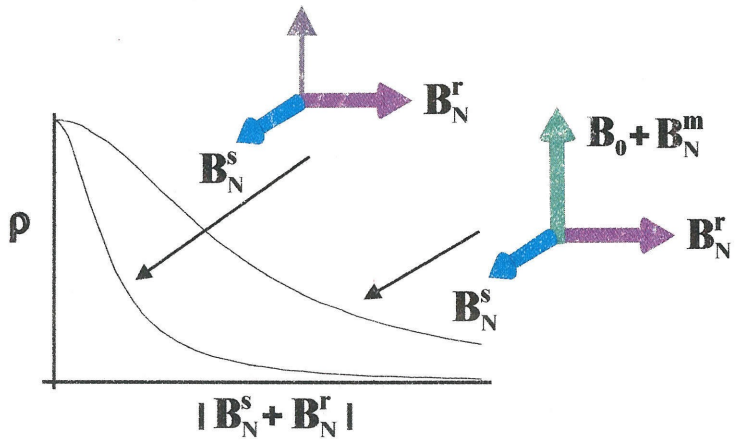


Figure 1.5 A graphical illustration of the increase in Hanle width that accompanies the addition of a longitudinal field, allowing the change in Hanle width to be used to measure this longitudinal field.

Optical detection of the field stored along the z-axis is accomplished via the change in Hanle width that this extra field induces. This is shown in figure 1.5, and can be seen as follows. Imagine that in figure 1.3 there is an additional magnetic field along the z-axis that is very much larger than the transverse fields. Then, the electrons will precess about the total field, which is currently very close to the z-axis. Thus, the z-axis orientation of the electron spins is predominately conserved. Indeed, it will now take a transverse field that is very much larger than the magnitude of this new longitudinal field to depolarize the electron spins. Thus, if an external transverse

magnetic field is swept when there is a nuclear field stored along the z-axis, the resultant Hanle width will be a measure of the size of that stored nuclear field.

Buratto and Weitekamp[21,24] were the first to present these ideas and their experimental verification. They observed ONMR in GaAs with a linewidth of only about 3 kHz. This allowed NMR spectra related to optically relevant defects in GaAs to be detected with enough sensitivity that differences were detected between samples with nominally the same preparation[23].

D. Larmor Beat Detection

There is a problem with detection via the Hanle effect using the previously described methods. One may only measure the magnitude of the precessing hyperfine field. That is, the Hanle effect is sensitive to the sum total of the perpendicular magnetic fields, irrespective of the direction of this sum in the plane perpendicular to the Zeeman field. Thus, one has previously been restricted to either CW detection or point wise detection of the free-induction-decay (FID) of the nuclear hyperfine fields as accomplished with TSONMR.

There is, however, a better experimental technique that uses the Hanle effect to record an FID in real time. If one places a reference magnetic field along a direction in the *xy*-plane, the plane perpendicular to the Zeeman magnetic field, then the nuclear hyperfine field will subsequently beat against that field, creating a sinusoidal modulation of the transverse component of the magnetic field. That is, if the two magnetic fields are

initially aligned along each other, the two fields will then add to each other to produce a transverse magnetic field whose magnitude is the sum of their individual magnitudes. The nuclear field will precess, and at one half of the period of its precession it will be exactly opposed to the reference field, producing a transverse magnetic field whose magnitude is the difference of the magnitudes of the individual magnetic fields. Then, after one full period of precession the two fields will be exactly colinear again. Furthermore, the reference field may be imagined to rotate in the *xy*-plane at some frequency ω_R. Then the time period required for the two precessing fields to recover their colinear orientation will be the period of the difference between their frequencies of oscillation, and one half of this period will be the time required for the two fields to oppose one another.

This may be seen algebraically by considering the vector sum of the two precessing magnetic fields

$$B_T = B_N e^{i\omega_N t} + B_R e^{i\omega_R t}.$$

This can be rearranged to produce

$$B_T = e^{i\omega_N t}\left(B_N + B_R e^{i(\omega_R - \omega_N)t}\right), \tag{1.18}$$

and this itself naturally yields

$$\|B_T\| = \sqrt{B_N^2 + B_R^2 + 2B_N B_R \cos\left((\omega_R - \omega_N)t\right)} \tag{1.19}$$

which is the same result one would obtain by applying the law of cosines at each and every point along the precession of the two fields. This result may be inserted into equation (1.16) to give

$$\mathbf{S}_z(\mathbf{B}_\mathrm{T}) = \frac{\mathbf{S}_z(0)}{1+\left(B_N^2 + B_R^2 + 2B_N B_R \cos\left(\left(\omega_R - \omega_N\right)t\right)\right)/\left(B_\mathrm{H}^2 + B_z^2\right)}. \tag{1.20}$$

Moreover, if one chooses the substitutions

$$B_{½} = \sqrt{B_\mathrm{H}^2 + B_z^2}, \tag{1.21}$$

$$b_n = B_N / B_{½}, \tag{1.22}$$

and

$$b_r = B_R / B_{½}, \tag{1.23}$$

equation (1.20) reduces to

$$\mathbf{S}_z(\mathbf{B}_\mathrm{T}) = \frac{\mathbf{S}_z(0)}{1+b_n^2+b_r^2+2b_n b_r \cos\left(\left(\omega_R - \omega_N\right)t\right)} \tag{1.24}$$

which can be rewritten as

$$\mathbf{S}_z(\mathbf{B}_\mathrm{T}) = \frac{\mathbf{S}_z(0)/\left(1+b_n^2+b_r^2\right)}{1+\left(2b_n b_r/\left(1+b_n^2+b_r^2\right)\right)\cos\left(\left(\omega_R - \omega_N\right)t\right)}. \tag{1.25}$$

Figure 1.6 illustrates the relationship between equation (1.25), the two precessing nuclear fields, and the Hanle curve. The upper portion, labeled as part "a," is a schematic of two nuclear hyperfine fields, $\mathbf{B}_\mathrm{N}$, precessing about the static magnetic field of the NMR experiment. The purple and red vectors represent isotopes with distinct Larmor

frequencies. The vector sum of transverse hyperfine fields is time dependent at the beat between the two Larmor frequencies. The purple nuclear field is used as a reference for detection of NMR of the red isotope. Prior to our group's introduction of such reference fields, all ODNMR was performed as a low-resolution frequency domain experiment.

The lower portion of figure 1.6, labeled as part "b," depicts the projection of the vector sum the two precessing nuclear fields in part "a" onto the Hanle curve. The Hanle curve represents response of luminescence polarization, ρ, to a transverse field. When electron spins precess about such a field, changing the z component of their spin angular momentum, ρ is altered (see figure 1.3 above). This response in ρ to transverse fields is time dependent when the vector sum of the two beating hyperfine fields from above is projected onto the Hanle curve.

Now the different frequency components contained within the modulation of ρ by these two precessing nuclear fields will be extracted. Equation (1.25) can be decomposed using the series summation

$$\frac{1}{1+ax} = 1 - ax + a^2x^2 - a^3x^3 + a^4x^4 + \ldots \tag{1.26}$$

to obtain an approximation to equation (1.25) . That is,

$$\mathbf{S}_z(\mathbf{B}_T) = c\left(1 - a\cos\left((\omega_R - \omega_N)t\right) + \ldots + a^n \cos^n\left((\omega_R - \omega_N)t\right) + \ldots\right) \tag{1.27}$$

with

$$a = 2b_n b_r \Big/ \left(1 + b_n^2 + b_r^2\right) \tag{1.28}$$

and

$$c = \left(\mathbf{S}_z(0) \Big/ \left(1 + b_n^2 + b_r^2\right)\right) \tag{1.29}$$

It is clear that equation (1.27) has both linear and nonlinear components. However, finding the magnitude of the linear component is not as easy as simply taking the coefficient of the cosine term in equation (1.27). Powers of cosine can, of course, be decomposed into a sum of linear cosine terms with arguments that are integer multiples of the original argument (integers k with $k = 0,1,2,3...$). Carrying this process out to $n = 200$, one obtains the coefficients of the zeroeth, first, second, and third, harmonics with reasonably high precision as shown in figure 1.7 and figure 1.8. These coefficients are even and odd polynomials of order n or $n-1$ in a, depending on the parity of the function. The value $n = 200$ was chosen so that even out at $b_n = b_r = 10$ the curves would be accurate to within approximately 1%. There is an alternate analytical method of obtaining approximations to the curves in figures 1.5 and 1.6, using the series approximation in equation (1.27) along with some additional assumptions. This is skillfully illustrated in the thesis of John A. Marohn[27], and the curves thus obtained (zero and first harmonics) agree well with those that have been calculated here.

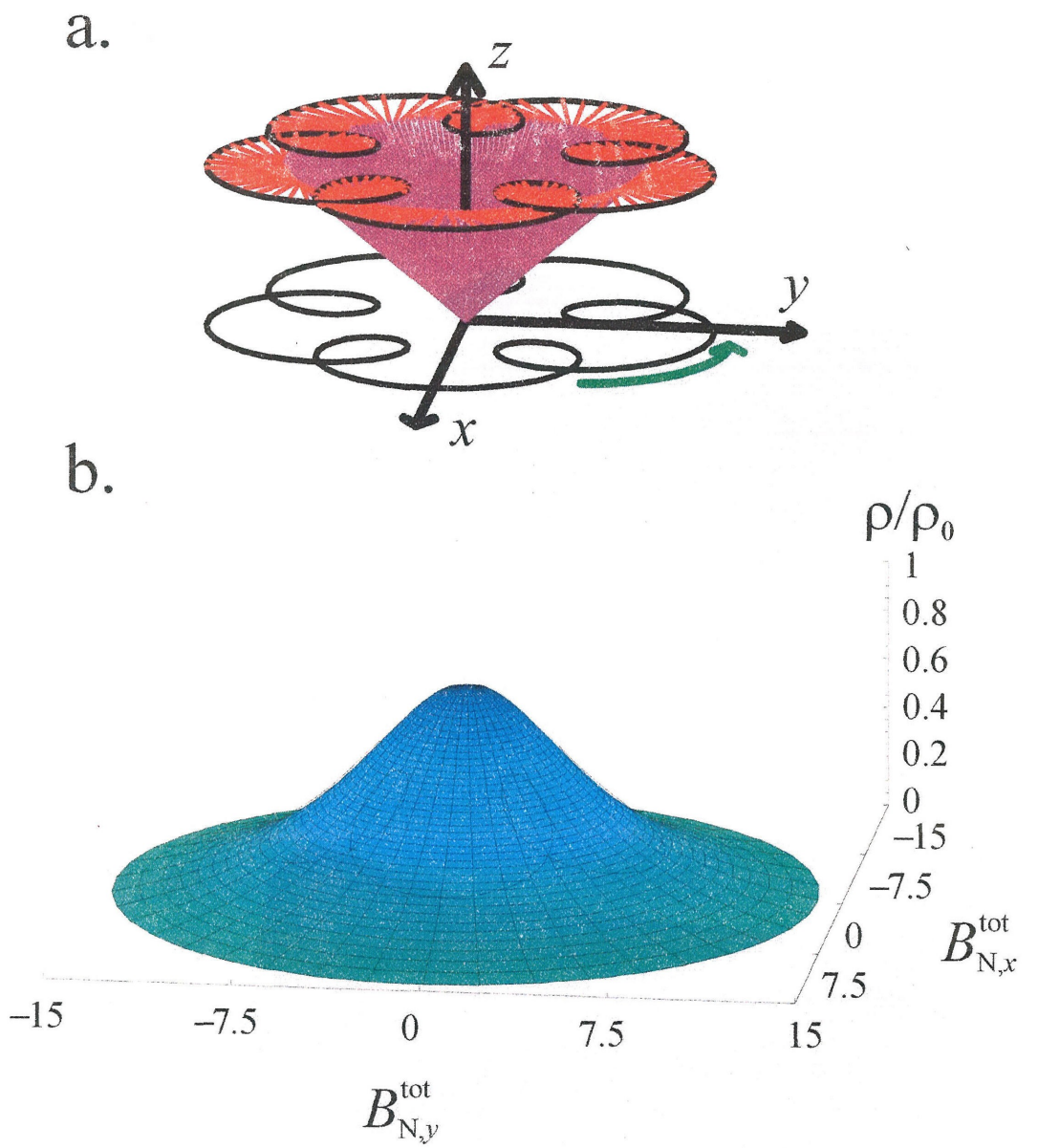


Figure 1.6 Schematic illustration of the Larmor beats and their effect on the polarization of the luminescence. a) The transverse components of two precessing magnetic fields alternately add to and subtract from one another, producing a total transverse magnetic field that is sinusoidally modulated at the difference frequency between the two precessing fields. b) This modulated sum will then produce a modulation of the circular polarization of the luminescence as indicated by the projection of the modulated transverse field on the Hanle curve.

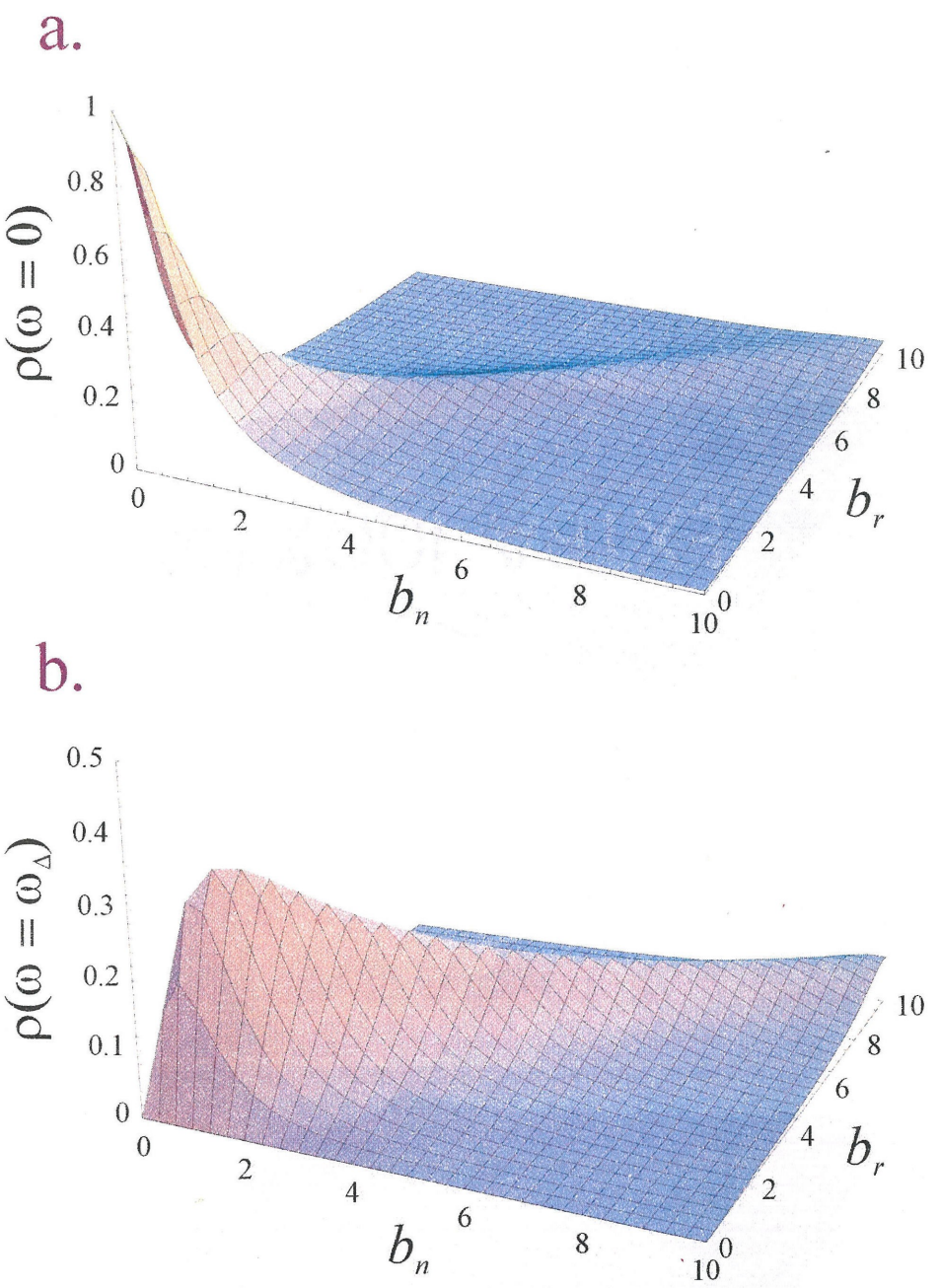


Figure 1.7 Upon decomposing all terms in equation (1.27) to appropriate sums of linear cosine terms and keeping all terms to n=200, we arrive at numerical approximations to a) the DC Fourier component of the depolarization of the luminescence by two rotating magnetic fields, and b) the first harmonic of the depolarization of the luminescence (i.e., the component at the difference between the frequencies of the two rotating magnetic fields).

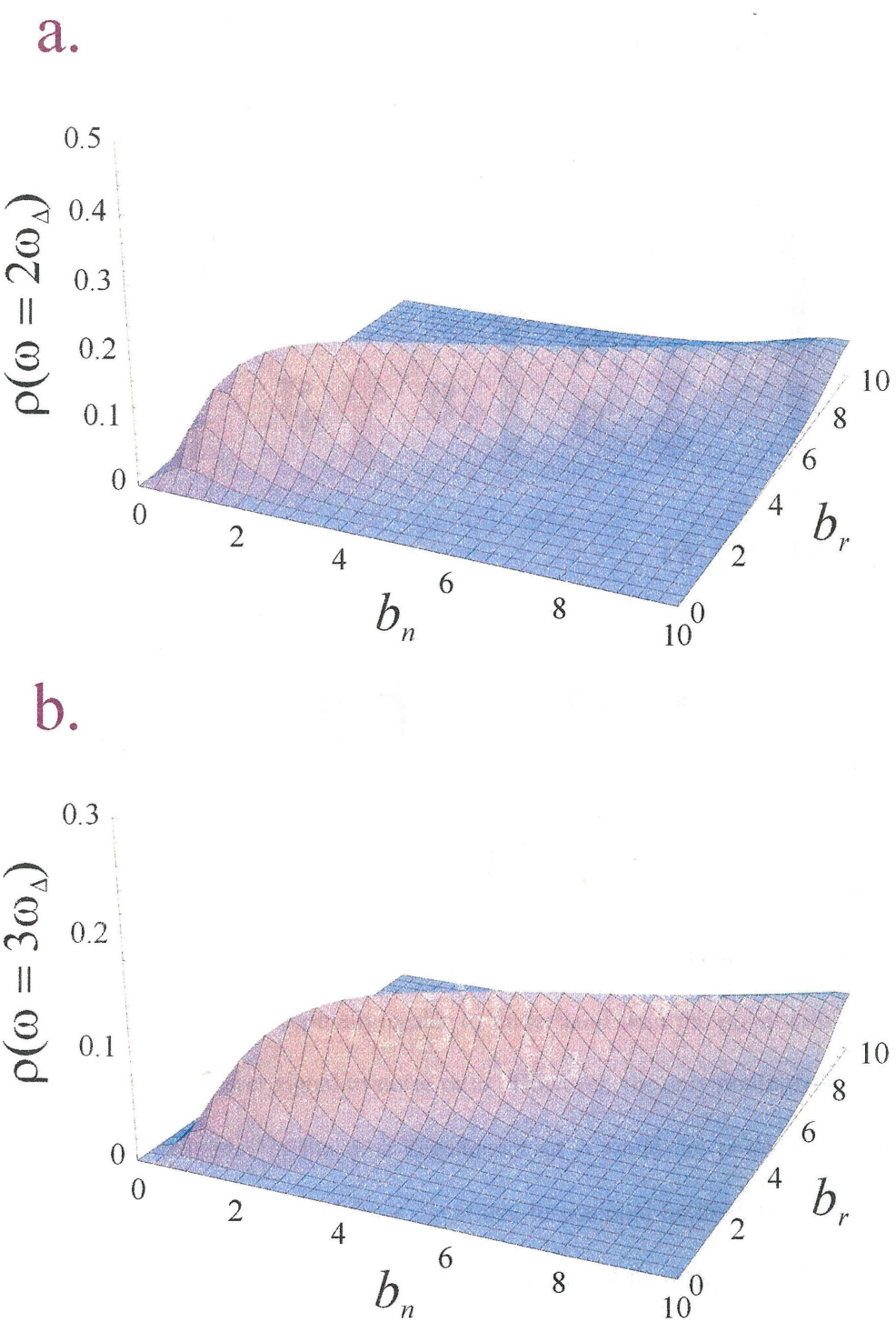


Figure 1.8 Numerical approximations to a) the second harmonic of the depolarization of the luminescence, and b) the third harmonic of the depolarization of the luminescence. Conditions and underlying equations are the same as for those in figure 1.7.

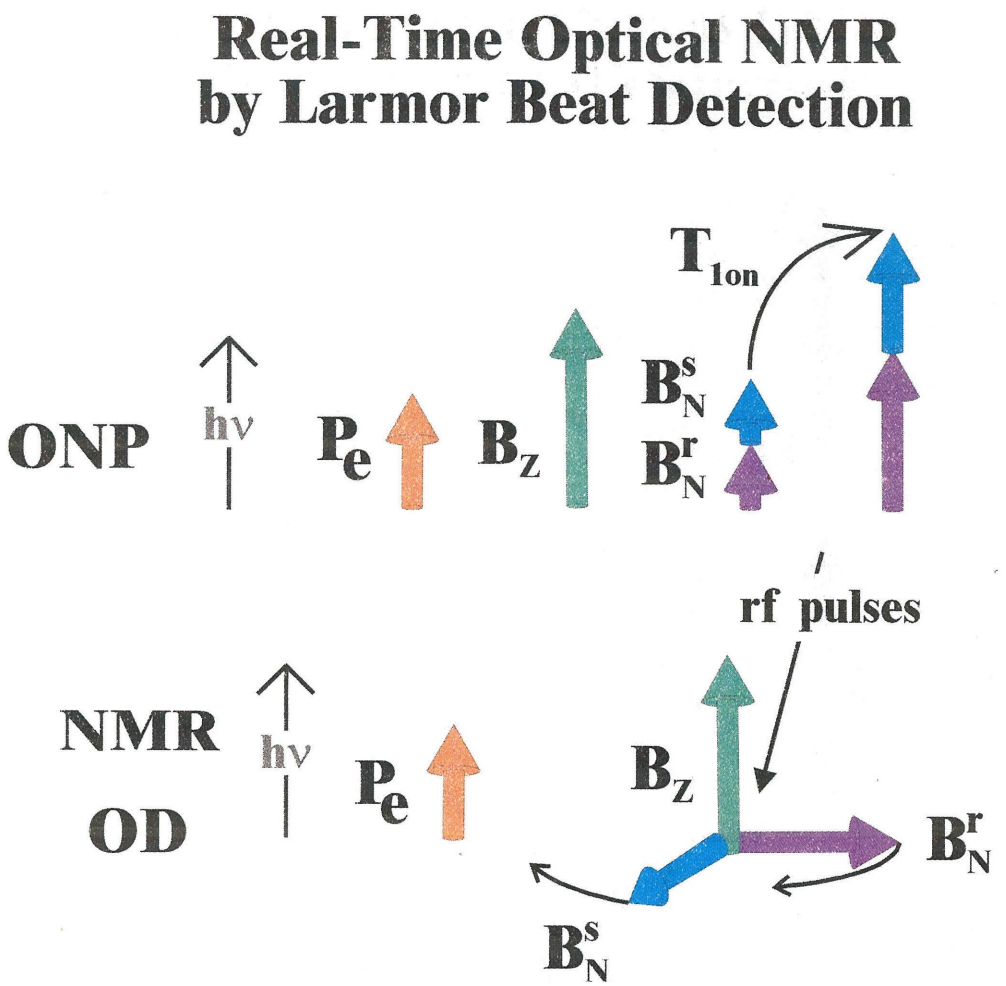


Figure 1.9 The real-time LBD detected ONMR experiment is schematically illustrated here. This experiment is conceptually easier to understand than the point-wise detected experiment, but suffers lower signal-to-noise ratio. However, the only nonlinearity present is that of higher harmonics in the time dependence of ρ while the point-wise detected scheme must worry about clipping of the signal due to potential nonlinearities in the first harmonic detection surface (figure 1.7b) as BN decays from its maximum value to zero. The same phenomenon only leads to a more Gaussian lineshape in the real-time detection case.

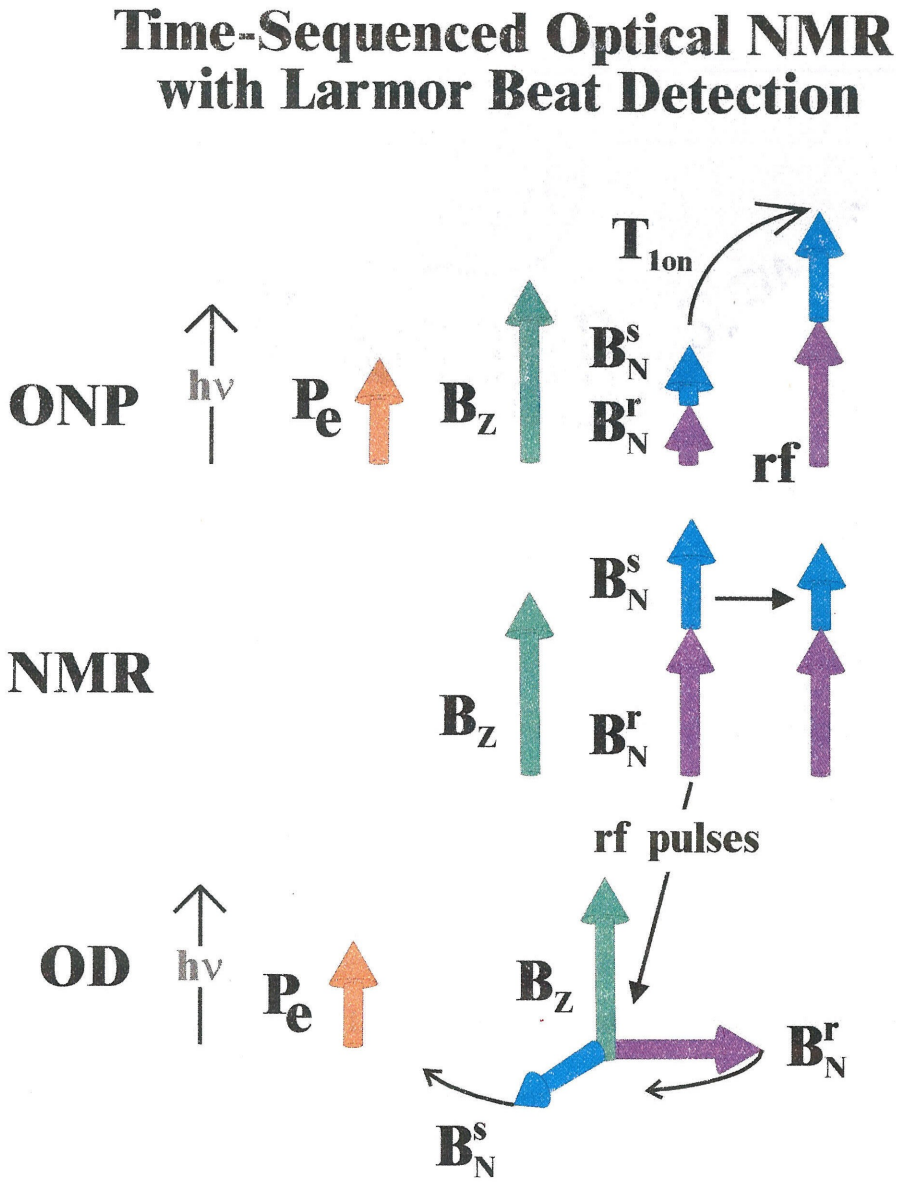


Figure 1.10 A schematic illustration of the point-wise detected LBD experiments. The point-wise experiment is done with spin-locking on during detection to increase the signal to noise ratio, but possesses an additional source of nonlinearity in that the magnitude of the higher amplitude points can be clipped due to saturation of the linear response of the first harmonic. See figure 1.11 and the text of this section for the relevant details.

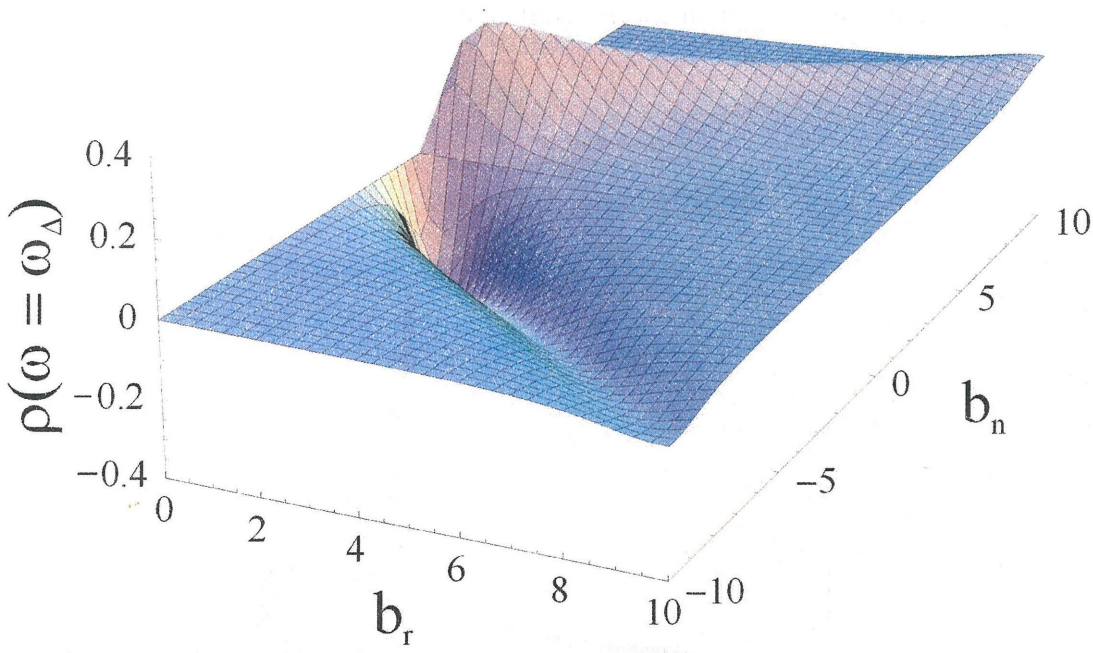


Figure 1.11 The first Fourier component of the depolarization (as in figure 1.7b) plotted for $-10 \le b_n \le 10$. Point-wise detection amounts to sinusoidally oscillating, at some constant value of b_r, from $b_n = b_n(t_2 = 0)$ to $b_n = -b_n(t_2 = 0)$.

Two different kinds of LBD detection conditions will now be considered, real time detection and point-wise detection. For real time detection, the only source of nonlinearity that comes from using LBD is from harmonics that are inadequately filtered out of the NMR signal. It is easy to reduce this source to levels that are well below the signal-to-noise level of the NMR experiment through the use of standard high and low pass filters. However, in the case of point-wise detection, one incurs an additional source of signal nonlinearity. Point-wise detection amounts to sinusoidally oscillating along a single value of b_r for different values of b_n (from $b_n(t_2 = 0)$ to $-b_n(t_2 = 0)$). This can be

seen in figure 1.11, which is the same function as in figure 1.8b, but plotted over a different range.

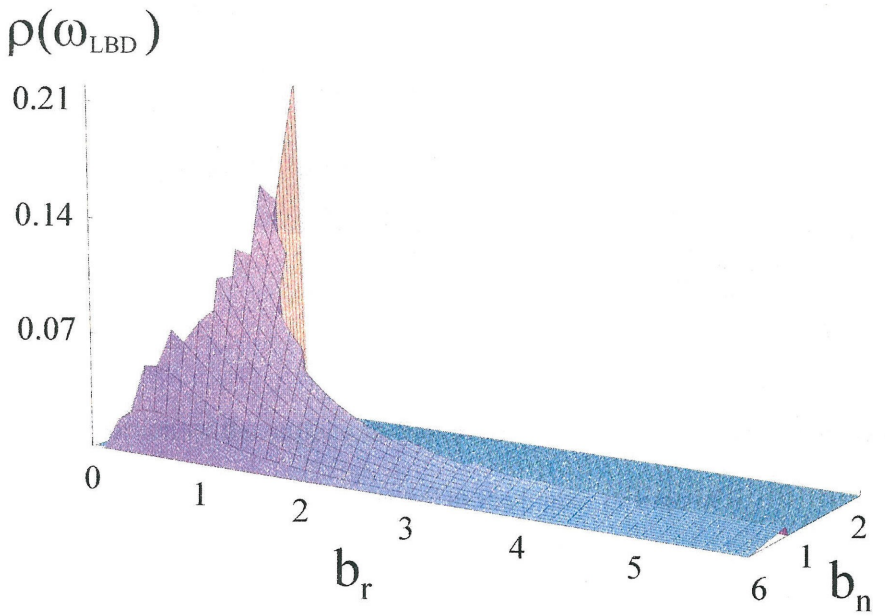


Figure 1.12 The portion of the surface in figure 1.11 where the fraction of the signal due to the term linear in a is 99% or greater compared to the higher powers of a.

Figures 1.12 and 1.13 show that portion of the surface in figure 1.11 that leads to linear point-wise detection of the ONMR signal by LBD to 99% and to 90%, respectively. This is computed by dividing that portion of the signal that comes from the term linear in a by the total signal. That is, if the signal stays within the area in figure 1.12, it is guaranteed to be clipped by less than 1%, and, if it stays within the area in figure 1.13, it is guaranteed to be clipped by less than 10%.

Another way of illustrating this same point is demonstrated in figure 1.14 where an approximation to the fraction of the signal at the difference frequency $(\omega_R - \omega_N)$ that

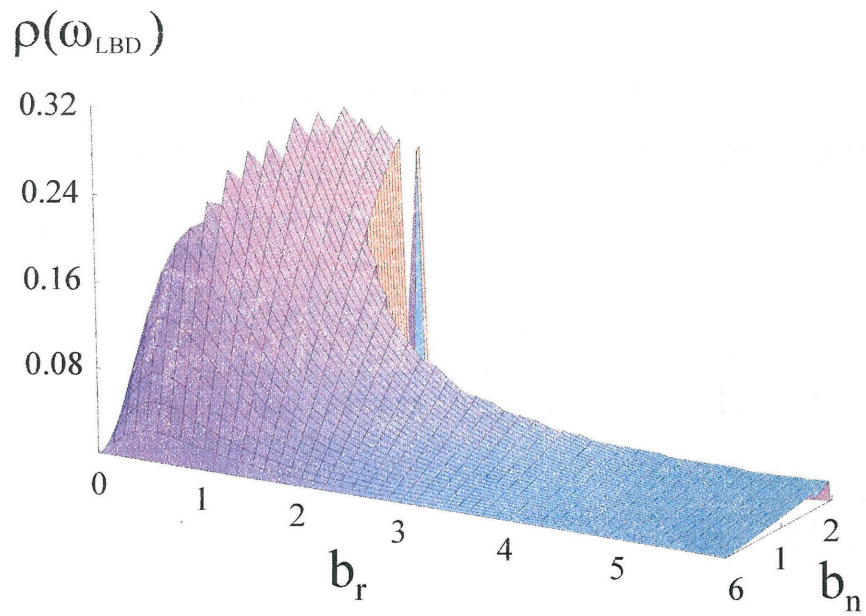


Figure 1.13 The portion of the surface in figure 1.11 where the fraction of the signal due to the term linear in a is 90% or greater compared to the higher powers of a.

is linear at a given magnitude of the transverse fields, and three separate cuts through the surface in figure 1.11, are shown. The linear fraction was computed by finding the ratio between the linear portion of the first harmonic (that portion represented by a^1) and the sum of all of the other of the terms in the first harmonic (a^n with $1 < n \leq 200$). Note that the signal has the highest linearity where it has the smallest magnitude. This runs counter to some claims that have been made about the position of the most linear signal.

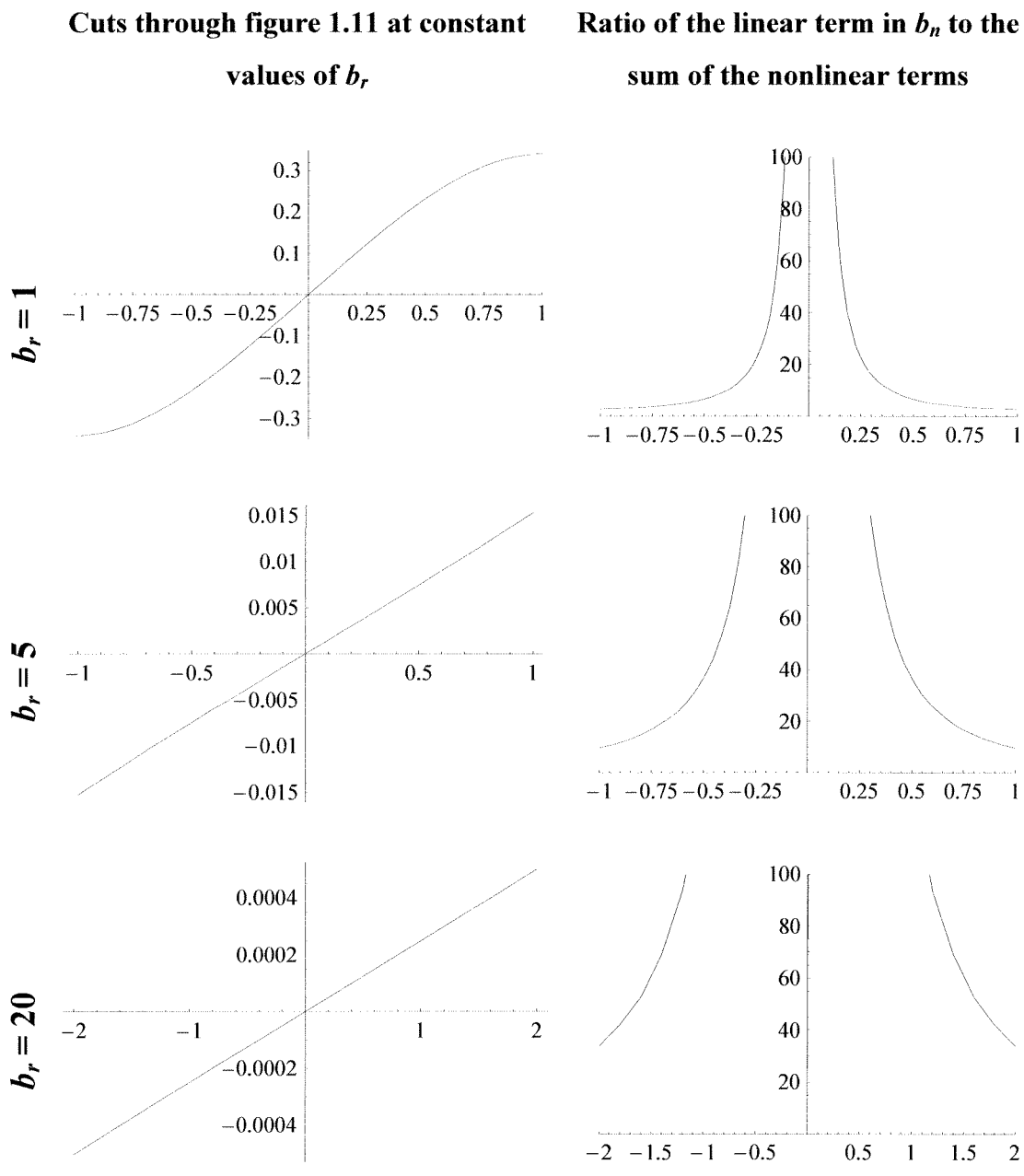


Figure 1.14. The signal linearity versus signal size tradeoff is illustrated graphically by showing three different cuts through figure 1.11 at different values of b_r, and the percentage of a signal which is detected in a point-wise fashion that is linear at given values of b_n and b_r.

Moreover, while the experiments are still detected in a point-wise fashion only with both nuclei spin locked during Larmor Beat Detection (to achieve the highest signal to noise) in general practice, these experiments still have much higher resolution than any previously accomplished because they are carried out at much higher magnetic fields than could be accommodated in the original TSONMR experimental setup (CWNMR does even warrant a comparison for the reasons mentioned earlier). Therefore, the LBD technique of optical detection has now opened the full menu of high-resolution NMR techniques to a level of availability that was heretofore unheard of.

E. The Knight Shift

In samples with short electron spin-lattice relaxation times, such as metals and semiconductors, the time-averaged spin magnetic moment of the electrons is felt by the spin magnetic moment of a given nucleus as if it were an extra magnetic field present at the nucleus. This extra "electron field" shifts the Zeeman resonance frequency of the nuclear spin involved, and, thus, the distribution of electron density in a given sample leads to a corresponding distribution of nuclear resonances in the NMR spectrum of that sample. These shifts are usually called Knight shifts when due to conduction electrons and paramagnetic shifts when due to highly localized electrons. Therefore, it is, in principle, possible to use the NMR spectrum of a sample as a fingerprint to identify the unpaired electron distribution that produced it. Electrons in states of uniform delocalization, like Bloch states, provide only one value of the Knight shift and $|\psi(\mathbf{r})|^2$ of

the electron state at the nucleus can be calculated from this shift. Such experiments have been done by low resolution NMR on a variety of ground state samples for more than twenty years.

The isotropic hyperfine Hamiltonian was introduced in equations (1.1) and (1.2) in Hz, using rationalized MKSA units as[29]

$$\mathcal{H}_{\mathrm{HF},\alpha} = A_\alpha \mathbf{I}\cdot\mathbf{S} \tag{1.30}$$

where

$$A_\alpha = \tfrac{2}{3}\mu_o g_o \mu_B \gamma_{\alpha,\mathrm{Hz}} d_\alpha \tag{1.31}$$

The constants in equation (1.31) are discussed immediately following equation (1.2); however, one quantity in this equation requires special attention before the Knight shift is quantitatively evaluated. This is $d_\alpha = \left|\Psi(\mathbf{r}_\alpha)\right|^2_\alpha$ which is the probability density of an electron at the nucleus located at $\mathbf{r}_\alpha$ given that the one-electron orbital ψ is occupied.

A theoretical estimation of the d_α requires a model wavefunction. The approach taken here is the simplest possible. It has been used previously to estimate Knight shifts around hydrogenic donor centers in GaAs[10]. The conduction band is assumed to be constructed from the 4s and 4p orbitals of both gallium and arsenic. The relative probability of these two one-electron excitations in the excited state GaAs molecular wavefunction is determined by analogy to the case of InSb where the conduction band ESR line is sufficiently narrow to measure the Fermi contact paramagnetic (Overhauser) shift of the electron Larmor frequency by known (equilibrium) polarization of the In and

Sb nuclei separately[30,31]. This is the converse effect to the Knight shift resulting from the same hyperfine interaction. Since at equilibrium the nuclei are uniformly polarized, the degree of localization of the electron is irrelevant and the observed Overhauser shifts can be directly interpreted in terms of the underlying atomic orbitals. The experimental shifts were found to agree with those calculated from atomic orbitals if the distribution of the s-electron between the In and Sb was in the ratio 0.43:0.53, respectively. We assume with Paget et al.[10] that the same ratio holds for Ga and As in GaAs, which is reasonable because the ionicities in the two molecules are similar (0.310 and 0.321 for InSb and GaAs, respectively, on the Philips scale)[30,31]

The energy of a nuclear spin in the presence of both an external field and an electron to which it has isotropic hyperfine contact is given by the Hamiltonian (in Hz).

$$\mathcal{H}_\alpha = -h^{-1}\mu_\alpha \cdot \mathbf{H}_o + h^{-1}\frac{8\pi}{3}\frac{\mu_o}{4\pi}\mu_\alpha \cdot \mu_s \left|\psi_k(\mathbf{r}_\alpha)\right|^2 \tag{1.32}$$

$$\mathcal{H}_\alpha = -\gamma_{\alpha,\mathrm{Hz}}\mathbf{I}\cdot\mathbf{H} + \tfrac{2}{3}\mu_o g_o \mu_B \gamma_{\alpha,\mathrm{Hz}}\mathbf{I}\cdot\mathbf{S}\, d_\alpha \tag{1.33}$$

The nuclear spin energy is the sum of the energy from its interaction with the external magnetic field and its interaction with the electron spin magnetic moment. If the direction of **H** is defined to be the *z*-axis, **I** and **S** are parallel to this axis since this work is at high magnetic field, providing an electron quantization axis that is parallel to **H**, so effectively (the high field limit)

$$\mathcal{H}_\alpha = -\gamma_{\alpha,\mathrm{Hz}} I_z H_o + \tfrac{2}{3}\mu_o g_o \mu_B \gamma_{\alpha,\mathrm{Hz}} I_z S_z d_\alpha \tag{1.34}$$

Since the electron spin states are short lived compared to the inverse of the interaction frequency, the electron spin operator S_z can be replaced by its average $\langle S_z \rangle$.

$$\mathcal{H}_\alpha = -\gamma_{\alpha,\mathrm{Hz}} I_z H_\mathrm{o} + \tfrac{2}{3}\mu_\mathrm{o} \mathrm{g}_\mathrm{o} \mu_\mathrm{B} \gamma_{\alpha,\mathrm{Hz}} I_z \langle S_z \rangle d_\alpha \tag{1.35}$$

Where $\langle S_z \rangle = \frac{1}{2} P$ and P is the electron spin polarization ($-1 \le P \le 1$).

$$\mathcal{H}_\alpha = -\gamma_{\alpha,\mathrm{Hz}} I_z H_\mathrm{o} + \tfrac{2}{3}\mu_\mathrm{o} \mathrm{g}_\mathrm{o} \mu_\mathrm{B} \gamma_{\alpha,\mathrm{Hz}} I_z P \tfrac{1}{2} d_\alpha \tag{1.36}$$

Hence, the hyperfine interaction acts like an extra field on the nuclear spin that adds to H_o, and, therefore, gives a shift in the resonance frequency. This shift is the Knight shift.

$$\Delta\nu_\alpha = \tfrac{1}{3}\mu_\mathrm{o} \mathrm{g}_\mathrm{o} \mu_\mathrm{B} \gamma_{\alpha,\mathrm{Hz}} P d_\alpha \tag{1.37}$$

Note the all-encompassing importance of the hyperfine contact interaction thus far. It was the thermal rod connecting the electron spin reservoir to the nuclear spin reservoir, which caused the optical nuclear polarization discussed in section II of this chapter. Furthermore, it was the conduit that allowed the electron spins to react to the transverse orientation of the nuclear spin angular momentum, yielding the Hanle effect upon which all of the optical detection techniques in sections B, C and D of this chapter are based. Now it serves as a resonance shift, and inhomogeneities in the hyperfine interaction will necessarily be a source of line broadening.

F. Optically Relevant Defects - The original quantum dots

Optically relevant defects [2-4,10-14,18,21,23,24,26-28] (ORDs) are sites within semiconductor crystals where excited state electrons become trapped long enough to share their polarization with the local nuclei, but not so long that a great many such electrons can still pass by a given nucleus over the time required for relaxation of the populations of the nuclear energy levels. Defects such as these determine the luminescence properties of many of the conduction band electrons since they become trapped at these sites and, subsequently, luminescence from these sites. Moreover, such defects occur even in crystals that are grown as purely as current techniques allow.

Hydrogenic shallow donor states (e.g., Si substituted at a Ga site in the lattice) are a leading candidate for the identity of the low E-field sites relevant to the optical NMR signal. In addition to their role as intentional or unintentional dopants, such states invite interest due to their similarity to semiconductor quantum dots and wells. Characteristics relevant to optical NMR and shared by quantum dots and wells and shallow donors include electron localization to $\lesssim 10$ nm, efficient luminescence, and the possibility of absorbing photon angular momentum as spin angular momentum. Thus, such systems are nearly ideal for ONMR experiments while characterization of the electronic properties of each of these systems down to the local atomic level is of fundamental importance in the development of device-grade semiconductor materials. A characteristic shallow donor is schematically illustrated in figure 1.15, which is a false color cutaway diagram of such a site.

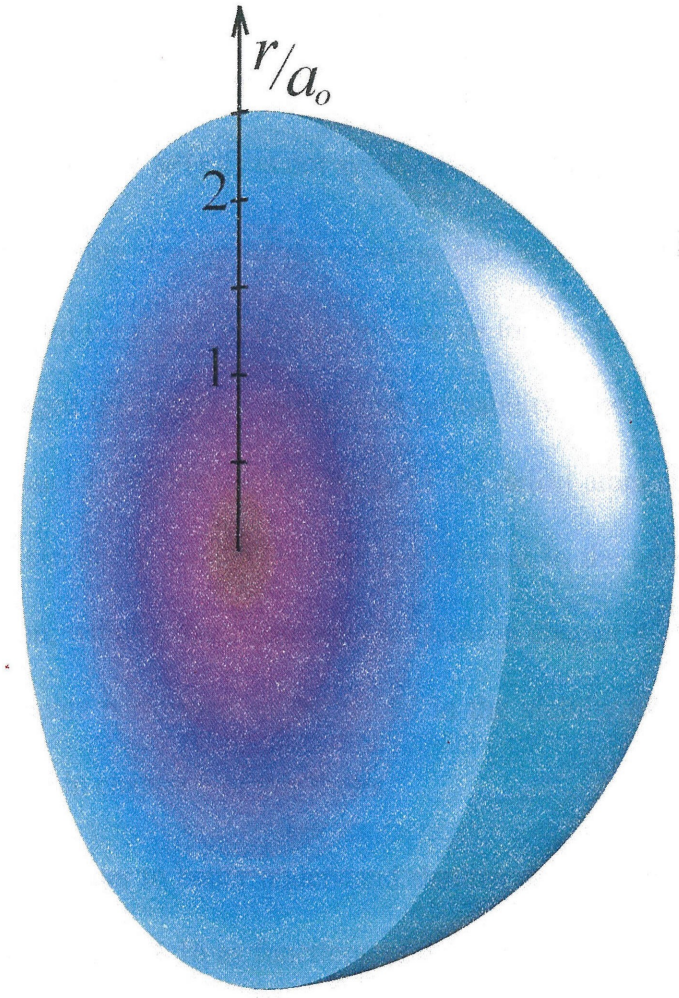


Figure 1.15 A 3D cutaway false-color representation of the electron probability density $|\Psi(r)|^2$ of a hydrogenic donor, where $\Psi(r) = A \exp(-r/a_0)$, a_0 (~ 10 nm) is the Bohr radius and A is a normalization constant.

Indeed, ONP, an NMR measurement of the Knight Shift, and optical detection by LBD, respectively, may be combined to yield one integrated experiment that will facilitate just such a characterization. The distribution of the Knight shifts thus measured

is the distribution of electron-nucleus hyperfine interactions, which is also the distribution of the $|\psi(\mathbf{r})|^2$ of the electrons taken at the positions of the nuclei. If $|\psi(\mathbf{r})|^2$ is radially symmetric, this experiment is an NMR imaging of the square of the wavefunction of the electrons, and if $|\psi(\mathbf{r})|^2$ is not symmetric, the image is still a radially averaged profile of the probability density of the electron that allows competing theoretical models to be tested until the match to this profile can be found. In either case, it is safe to say that one has imaged the quantum mechanical probability density of the electrons that are coupled to the nuclei, elucidating the charge of the defect atom and other relevant characteristics of the electronic state. The one impediment to the success of this experiment is spin diffusion, which I have addressed in detail in chapter 4 of this thesis. Results from such experiments performed on the nuclei in the vicinity of a hydrogenic shallow donor are presented in chapter 5 of this thesis.

In order to better understand shallow donors, some properties around a donor are plotted in figure 1.16 as functions of distance from the center ($r = 0$) of the donor. They are as follows, the magnitude of the Knight shift, which is proportional to $|\Psi(r)|^2$ and to the scalar product **I•S** of the nuclear and electronic spin angular momentum operators, in the NMR spectrum of a given nucleus, the relative effectiveness of optical nuclear polarization (ONP), which is dependent on fluctuations in the Hamiltonian governing the interaction that gives rise to the Knight shift, the radial density of nuclear spins that contribute to the observed NMR signal, and the relative contribution of nuclei to the optically detected (OD) NMR signal. This last weight consists of factors accounting for

the radial density of nuclear spins and for the electron density, to which the OD signal is proportional, at each nuclear site.

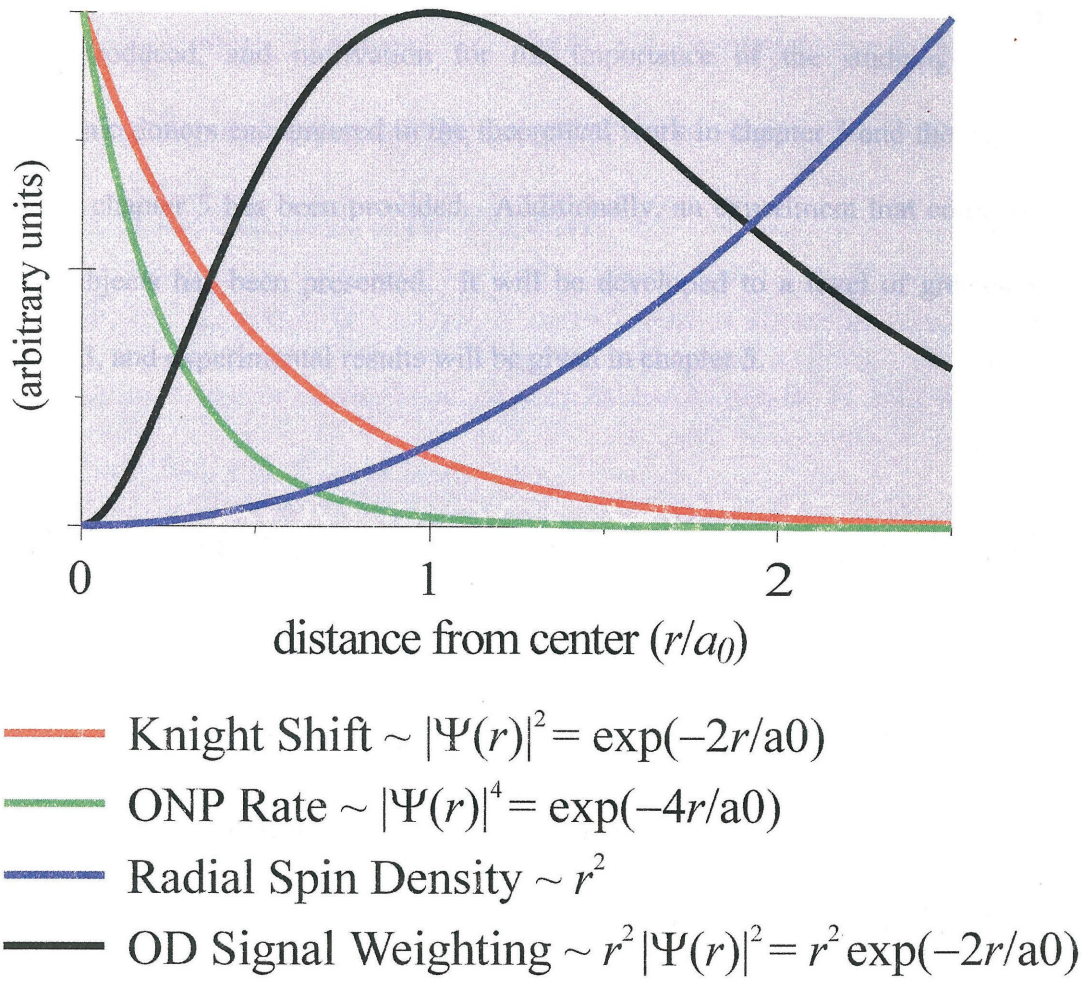


Figure 1.16 Properties of interest around a hydrogenic donor as a function of radial distance r, including a pictorial representation of the square $|\Psi(r)|^2$ of the electronic wavefunction at a donor site and other properties related to $|\Psi(r)|^2$, where $\Psi^2(r) = \mathrm{A}\exp(-r/\mathrm{a_o})$, a_0 is the Bohr radius and A is a normalization constant.

G. Conclusions

The fundamental phenomena used to conduct ONMR experiments have been explained from ONP through various OD techniques. Additionally, the Knight shift has been introduced, and motivation for the importance of the studying the shallow hydrogenic donors encountered in the theoretical work in chapter 3 and the experimental work in chapter 5 has been provided. Additionally, an experiment that combines all of these subjects has been presented. It will be developed to a level of greater detail in chapter 3, and experimental results will be given in chapter 5.

H. References

1) Lampel, G. *Physical Review Letters* **1968**, *20*, 491.

2) Ekimov, A. I.; Safarov, V. I. *JETP Letters* **1972**, *15*, 179.

3) Ekimov, A. I.; Safarov, V. I. *JETP Letters* **1972**, *15*, 319.

4) Berkovits, V. L.; Ekimov, A. I.; Safarov, V. I. *Sov. Phys.-JETP* **1974**, *38*, 169.

5) Dyakonov, M. I.; Perel, V. I. *Soviet Physics JETP* **1971**, *33*, 1053.

6) Dyakonov, M. I.; Perel, V. I. *Soviet Physics JETP* **1973**, *36*, 995.

7) Dyakonov, M. I.; Perel, V. I. *Soviet Physics-JETP* **1974**, *38*, 177.

8) Dyakonov, M. I.; Perel, V. I.; Berkovits, V. L.; Safarov, V. I. *Soviet Physics - JETP* **1974**, *40*, 950.

9) Dyakonov, M. I.; Perel, V. I. *Theory of Optical Spin Orientation of Electrons and Nuclei in Semiconductors*; Meier, F. and Zakharchenya, B. P., Ed.; North-Holland: Amsterdam, 1984, pp 11-71.

10) Paget, D.; Lampel, G.; Sapoval, B.; Safarov, V. I. *Physical Review B* **1977**, *15*, 5780.

11) Paget, D. *Physical Review B* **1981**, *24*, 3776.

12) Paget, D. *Physical Review B* **1982**, *25*, 4444.

13) Paget, D.; Berkovits, B. L. *Optical Investigation of Hyperfine Coupling Between Electronic and Nuclear Spins*; Meier, F. and Zakharchenya, B. P., Ed.; North-Holland: Amsterdam, 1984; Vol. 8, pp 173-258.

14) Meier, F.; Zakharchenya, B. P. *Optical Orientation*; 1 ed.; Maradudin, V. M. A. a. A. A., Ed.; Elsevier: Amsterdam, 1984; Vol. 8.

15) Fleisher, V. G.; Merkulov, I. A. *Optical Orientation of the Coupled Electron-NuclearSpin System of a Semiconductor*; Meier, F. and Zakharchenya, B. P., Ed.; Elsevier: Amsterdam, 1984; Vol. 8, pp 173-258.

16) Pikus, G. E.; Titkov, A. N. *Spin Relaxation under Optical Orientation in Semiconductors*; Meier, F. and Zakharchenya, B. P., Ed.; North-Holland: Amsterdam, 1984, pp 73-131.

17) Kalevich, V. K.; Korenev, V. L.; Fleisher, V. G. *Izvest. Akad. Nauk. SSSR. Ser. Fiz.* **1988**, *52*, 434.

18) Kalevich, V. K.; Korenev, V. L.; Fedorova, O. M. *JETP Letters* **1990**, *52*, 349.

19) Krapf, M.; Denninger, G.; Pascher, H.; Weimann, G.; Schlapp, W. *Superlattices and Microstructures* **1990**, *8*, 91-96.

20) Krapf, M.; Denninger, G.; Pascher, H.; Weimann, G.; Schlapp, W. *Solid State Communications* **1991**, *78*, 459.

21) Buratto, S. K.; Shykind, D. N.; Weitekamp, D. P. *Physical Review B* **1991**, *44*, 9035.

22) Buratto, S. K.; Shykind, D. N.; Weitekamp, D. P. *Journal of Vacuum Science and Technology B* **1992**, *10*, 1740.

23) Buratto, S. K. *Time-Sequenced Optical Nuclear Magnetic Resonance of Gallium Arsenide*; California Institute of Technology: Pasadena, CA, 1992.

24) Buratto, S. K.; Hwang, J. Y.; Kurur, N. D.; Shykind, D. N.; Weitekamp, D. P. *Bulletin of Magnetic Resonance* **1993**, *15*, 190.

25) Tycko, R.; Barrett, S. E.; Dabbagh, G.; Pfeiffer, L. N.; West, K. W. *Science* **1995**, *268*, 1460-1463.

26) Marohn, J. A.; Carson, P. J.; Hwang, J. Y.; Miller, M. A.; Shykind, D. N.; Weitekamp, D. P. *Physical Review Letters* **1995**, *75*, 1364-1367.

27) Marohn, J. A. *I. Multiple-Pulse Radio-Frequency Gradient Nuclear Magnetic Resonance Imaging of Solids. II. Optical Nuclear Magnetic Resonance Analysis of Epitaxial Gallium Arsenide Structures.*; California Institute of Technology: Pasadena, CA, 1996, pp 143.

28) Kempf, J. G. *Probing Quantum Confinement at the Atomic Scale with Optically Detected Nuclear Magnetic Resonance*; California Institute of Technology: Pasadena, 2001, pp 172.

29) Abragam *The Principles of Nuclear Magnetism*; Oxford University Press: Oxford, 1961.

30) Guéron, M.; Ryter, C. *Physical Review Letters* **1959**, *3*, 338-340.

31) Guéron, M. *Physical Review* **1964**, *135*, A200-A205.

II. The Experimental Apparatus and Sample

A reasonably detailed description of the experimental equipment utilized to carry out the experiments presented in chapter 5 will now be given. This same apparatus, in some form or another, has been used by several other members of the Weitekamp group for the experiments reported in their theses[1-4]. The apparatus that our group has built, like most prototypes, is a one-of-a-kind, constantly changing, integrated collection of high performance commercially purchased components, custom handmade components, and borrowed/salvaged components. Each major subsystem will be described, and its function within the larger experiment will be elucidated.

The structure of the sample, its characterization, and the originally intended purpose of studying it are all covered in section A. The handmade electrical components, rf generation, and rf delivery subsystems are discussed in section B. The optical subsystem can be divided into two braches, the excitation and the detection paths, both of which are explained in section C. The liquid Helium optical cryostat is presented in section D while the role of the laboratory computers in controlling the experiments and acquiring data is communicated in section E.

A. The Sample – Structure and Luminescence

The sample studied in all experiments presented in this thesis is a modulation-doped p-channel (Be-doped) $Al_xGa_{1-x}As/GaAs$ heterojunction ($x = 0.36$), schematically depicted in figure 2.1 below. Alex Ksendzov, John Liu, and Frank Grunthaner at the NASA Jet Propulsion Laboratory grew the sample via molecular beam epitaxy on a 001-oriented GaAs substrate. The purpose of the p-compensation of the top AlGaAs layer is to provide the band bending illustrated in figure 2.1. Moreover, the purpose of the thin undoped AlGaAs layer between the doped AlGaAs layer and the epitaxial GaAs is to provide a clean interface between this AlGaAs layer and the epitaxial GaAs layer by providing an interface internal to the AlGaAs material to catch migrating dopants from the p-compensated layer. That is, without this thin layer the dopants from the p-compensated layer would migrate to the AlGaAs-GaAs interface where they would interfere with the interfacial 2D trapped hole states that are shown in figure 2.1 at the

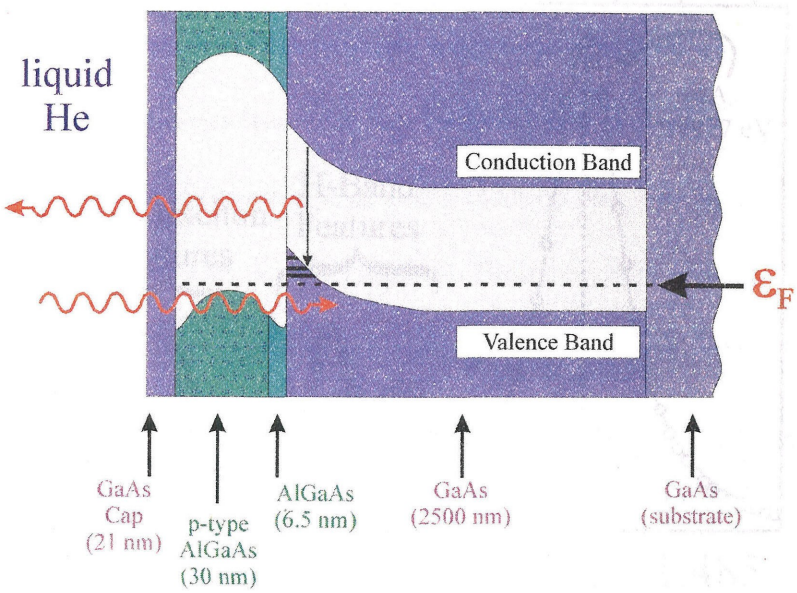


Figure 2.1 The sample used in all of the experiments reported in this thesis is an $Al_{0.36}Ga_{0.64}As/GaAs$ Heterojunction structure.

GaAs/AlGaAs interface as the horizontal lines to which the conduction band electrons are relaxing.

The luminescence to these states has previously been assigned to the H-band features of the luminescence spectrum shown in figure 2.2 below[5,6]. If these luminescence lines are indeed due to electron recombination with surface-trapped holes, then the electrons must have appreciable probability density located in the interfacial region for such recombination to occur. It stands to reason, therefore, that the nuclei that are hyperfine coupled to these electrons should experience the large interfacial electric field that is characteristic of this region. However, experiments that our group has performed

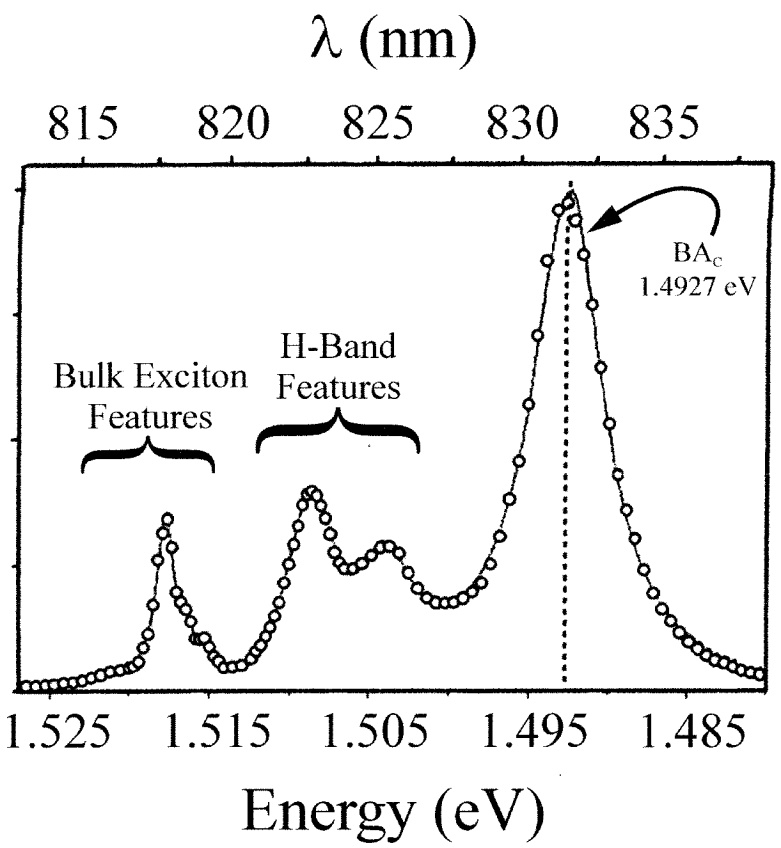


Figure 2.2 The photoluminescence spectrum of the sample that was used in the experiments reported in this thesis and that is schematically illustrated in figure 2.1.

conclusively prove that the sites that contribute to the ONMR signal in this region of the luminescence spectrum experience very low electric fields if any at all[3,7].

There are other competing theories for the identity of the H-band luminescence involving various excitons, free carriers and/or the trapped hole states previously mentioned, and there is evidence to support each claim[8-11]. ONMR is both in a position to shed some light on this subject, and unable to address the full detail of the controversy. On one hand, the results of the Knight shift imaging in chapter 5 and other experiments pursued in our laboratory are consistent with either a donor bound electron or an excitonic pseudo-donor (an isoelectronic defect that is substantially more attractive to the hole part of the exciton than to the electron part, leading to properties that mimic a donor bound electron as per the Hopfield-Thomas-Lynch model). However, we are not yet able to say that the ONMR is originating from the same sites that give rise to the bulk of the light observed in the region of the H-band. Indeed, we have seen ONMR signals from the 825-835nm region as well as from the 815-825nm region that are roughly equivalent in magnitude. Therefore, the proven dependence of H-band luminescence properties on factors that modulate the magnitude of the interfacial electric fields, as of yet, may not have any bearing upon the search for the identity of the sites giving rise to ONMR. One reason for this may be the requirement in the ONMR experiment that the nuclei can be polarized by the presence of the spin-polarized electrons in a time equal to or shorter than the ONP period of the ONMR experiments. Calculation of T1on in chapter 3 of this thesis shows that, even with the strong localization of a donor bound electron, the correlation time for the fluctuations in the hyperfine coupling needs to be $\gtrsim 10^{-8}$s in order

for T1on to be shorter than the required timescale. This is approximately equal to the residence time of an electron at a donor to which it is bound. It is unlikely that states with much less localization or much shorter electron lifetimes would posses the requisite ability to polarize the nuclear spins to which they are coupled by the hyperfine interaction.

B. The Dewar and Probe

The ONMR experiments on semiconductor samples must be performed with the sample at a temperature $\gtrsim$10K in order to suppress phonon assisted, nonradiative relaxation of the conduction band electrons back to the valence band. Thus, the optical cryostat is a crucial part of the experimental apparatus. It is a blown glass dewar manufactured by Pope Scientific that has three optical windows mounted on the tail of the dewar for optical access to the sample from a variety of directions. Unfortunately, since only one direction is required for the ONMR experiments, the other two windows serve only to increase the heating of the liquid He at the core of the tail by subjecting it to the blackbody radiation from the room temperature outer windows. A drawing of the dewar is shown in figure 2.3 while figure 2.4 shows a schematic cross-section of the same dewar, and figure 2.5 is a schematic illustration of the tail of the dewar.

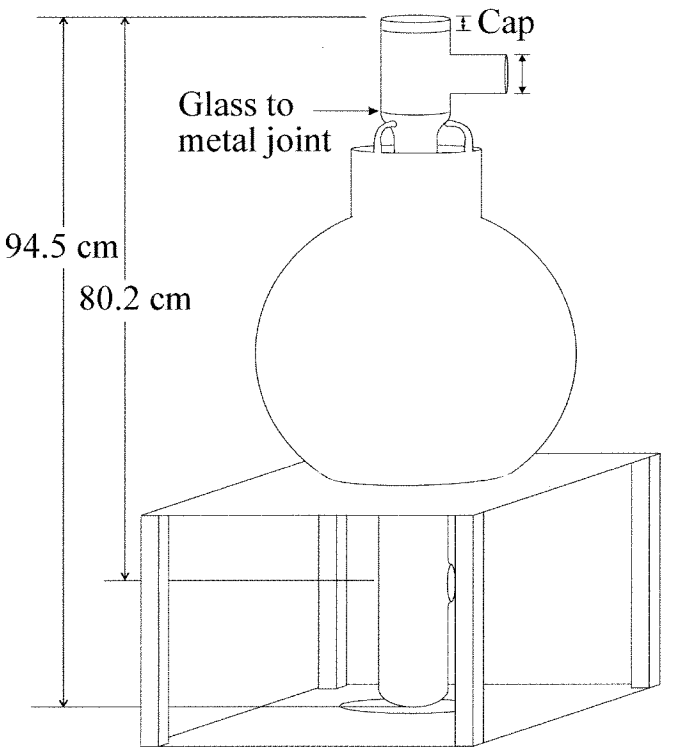


Figure 2.3 A drawing of the liquid helium cryostat used in the experiments reported in this thesis. It is a blown glass dewar that is fused to a copper pipe that has two branches. One branch is capped, and the other branch leads to the high capacity vacuum pump that is used to evaporatively cool the liquid helium to the λ point, the temperature where it becomes a superfluid (~2K).

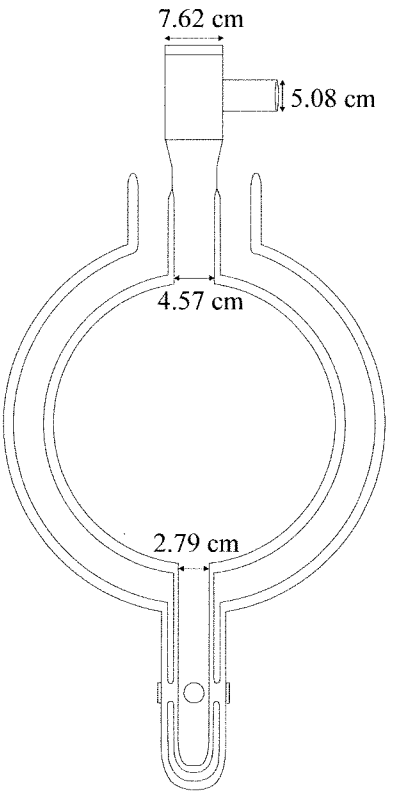


Figure 2.4. A cutaway view of the same cryostat. The inner chamber is for liquid helium, and it is surrounded with a vacuum space. The outer chamber completely surrounds these two chambers and is, itself, further surrounded by another vacuum space that is in

The glass walls of the dewar fold over themselves four times to yield two vacuum jackets and two separate spaces for cryogenic fluid. That is, there is one continuous vacuum space between all of the layers of glass, which are actually one continuous piece of glass. The purpose of the outer fluid space is to act as a thermal blanket for the central

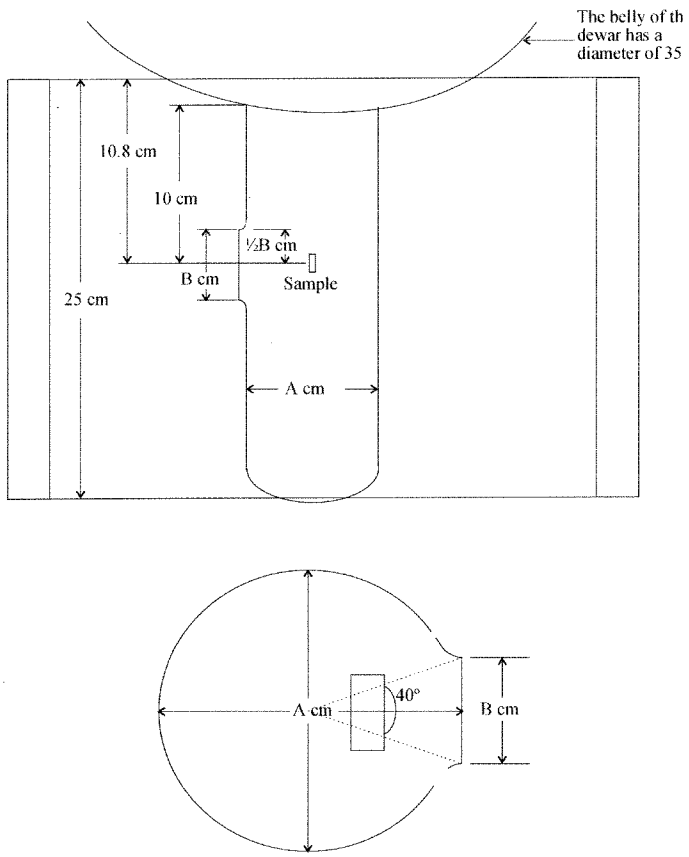


Figure 2.5 A cross-section of the dewar tail at the level of the optical window illustrates lengths A (diameter) and B which should be chosen such that $\text{Tan}(20^\circ) \leq B/A$ (i.e., such that from the center of the tail of the optical window occupies an angle of ~40°). The cryostat used for our experiments has $B \simeq 1\text{cm}$.

fluid space. The outer fluid space is filled with liquid nitrogen several times per day, and the "belly" of the dewar is only filled with liquid helium at the beginning of an experimental run. The sample sits at the height of the center of the optical windows, immersed in a bath of liquid helium.

The boiling point of the liquid helium at 1 atmosphere of pressure is 4.2K, fulfilling the need for sample temperatures below ~10K. However, as one might expect, the boiling helium is filled with bubbles of gaseous helium making their way to the surface of the liquid, and these bubbles will refract the incident laser light and the sample luminescence that are passing through the fluid.

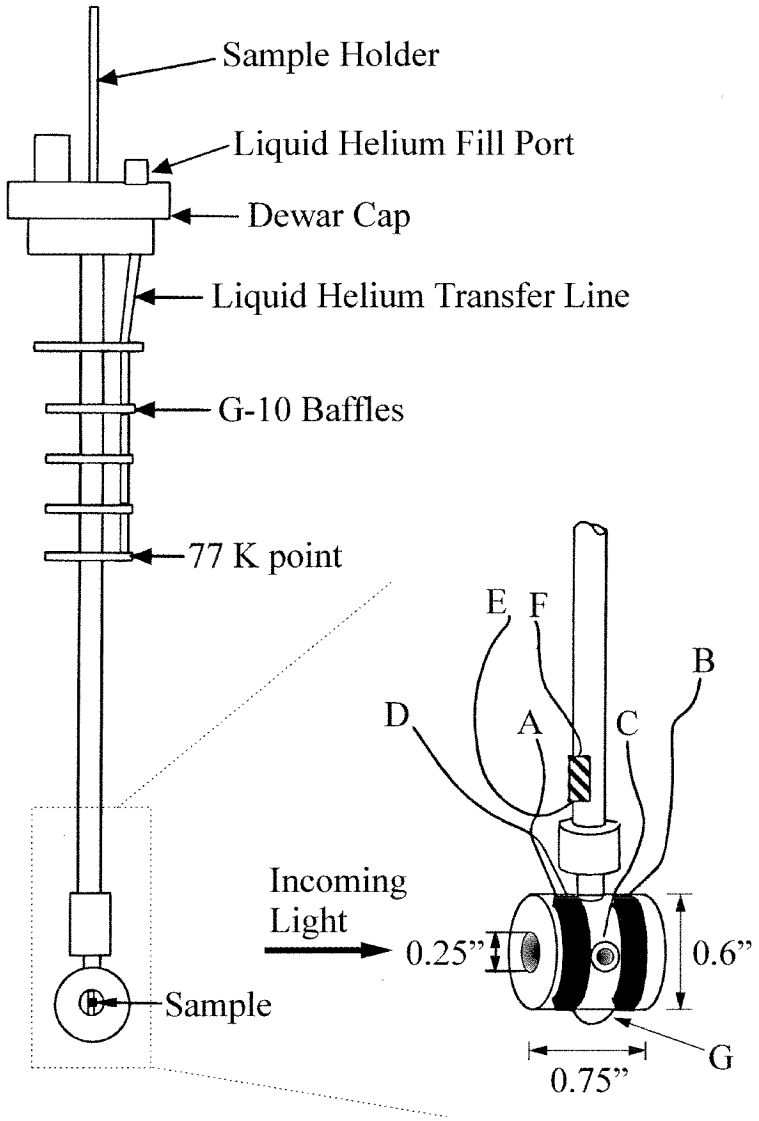


Figure 2.6 The home built NMR probe for the cryostat in figures 2.3-5. It has one set of superconducting magnets in the Helmholtz arrangement that provide the Zeeman field, and a second pair of Helmholtz coils made of 36 gauge copper wire wound in a 3x3 array that provide the rf irradiation of the sample. The leads are for (A,B) superconducting magnets, (C,D) rf coils, (E,F) carbon resistor leads, and (G) connection between the superconducting magnets.

The solution to this problem is to seek a different phase of liquid helium. If one applies a vacuum to the space holding the liquid helium, it will cool, and it will undergo a phase transition to superfluid liquid helium near 2K. This is a zero viscosity fluid that obeys Bose-Einstein statistics. The super fluid does not boil, and thus the optical polarization of both the laser light and the sample luminescence is preserved. The price paid for this improved performance is the loss of about 1/3 of the liquid helium in evaporative cooling of the remaining

helium to a temperature below that of the phase transition (called the λ point). When operating with superfluid helium, the hold time of the dewar is approximately 7.5 hours (that is, the helium will evaporate about 7.5 hours after the λ point is reached).

The NMR probe is illustrated in figure 2.6, including a detailed drawing of the probe head. The sample holder is a long, thin rod of G-10 fiberglass that threads through the center of the probe from top to bottom, emerging in the center of the Zeeman and rf coils in the probe head. This end of the sample rod is machined into a thin paddle, and the sample is mounted on the paddle with its growth direction facing the incoming laser beam (001 crystal orientation). Furthermore, it is mounted on one corner with a small amount of vacuum grease, which becomes sufficiently strong at the low temperatures it will experience in the cryostat (~77 K – 2 K). The sample used to

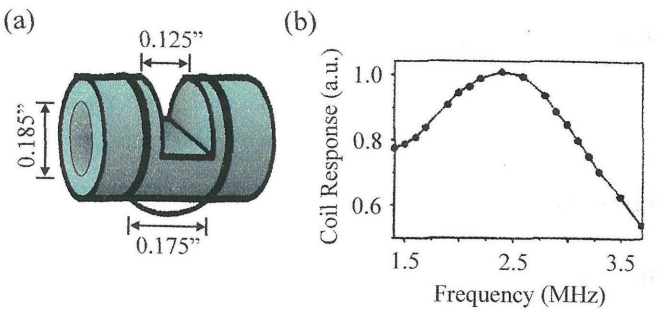


Figure 2.7 (a) A drawing of the rf insert to the NMR probe. The individual rf coils that make up the Helmholtz set are each a square array of 9 windings of 36 gauge copper wire (i.e., a 3x3 array) that lay in a trench 0.021" wide by 0.02" deep. Note that the 0.185" hole that is bored through the side of the plug runs the length of the plug, and that the notch cut into the front of the plug allows the sample rod mate into the plug placing the sample in the center of the Helmholz coil arrangement. (b) The response of this same coil to input from an rf amplifier, illustrating the breadth of the response over the region where ONMR is practiced in our lab.

be held with a thin coating of varnish across the entire surface adjacent to the paddle; however, this induced strain in the sample due to the difference in thermal contraction between the sample rod/paddle and the sample itself. This, in turn, induced a quadrupole splitting in the sample that provided some interesting information, but was ultimately in the way of the experiments being pursued. Additionally, besides mounting with vacuum grease in only one corner, the sample must be cooled slowly to avoid residual quadrupolar line broadening (greater than ~2 hours slow decent from cap to coils is sufficient).

With the exception of the dewar cap, liquid helium fill port and line, and the magnetic coils and their wiring, the probe is entirely made of G-10 fiberglass held together with "Shygoo," an epoxy that consists of equal parts resin (CIBA 6010), hardener (CIBA 825), and finely chopped quartz powder whose thermal expansion properties are matched to G-10 fiberglass[1]. G-10 fiberglass was selected because, while it experiences thermal expansion and contraction in the cooling process, the thermal changes cancel in such a way that its dimensions at 4K are approximately the same as at room temperature.

The superconducting electromagnets built into the probe head provide the Zeeman magnetic field, and, thereby, set the resonance frequency of the nuclear spins. These coils provide large variable magnetic fields with minimal heat dissipation, and are constructed of 54-filament Niobium Titanium alloy wire wound into groves in the probe head (G-10 fiberglass). The inductance of these coils at 2K is 1.67 mH, and they provide a field of ~20.5 mT/A at the center of the magnets (i.e., the sample position).

The rf coil is wound around an insert, which can be seen in figure 2.6 between the Zeeman coils (wire C leads to it), and is shown in greater detail in figure 2.7 along with the response of the coil to input from an rf amplifier. A ^{71}Ga rabi frequency of ~90 kHz at 3.2 MHz resonance frequency has been obtained with this coil. The LR characteristics of this coil at room temperature are L = 0.6 μH and R = 0.55 Ω while at cryogenic temperatures (2 K) the resistance is 0.36 Ω and the inductance is still 0.6 μH.

C. The Excitation and Detection Apparatus

The optical excitation path is depicted in figure 2.8, and consists of a laser that primarily emits linearly polarized light, a linear polarizer used to further enhance the linear polarization of the laser light, a focusing lens used to produce the smallest possible laser beam diameter at the sample, and a variable quarter-wave-plate used to produce either handedness of circularly polarized light depending upon its orientation with respect to the direction of the linear polarization of the initial beam (right circularly polarized has been used here). The optimal path for the excitation optics would be one that is normal to the sample surface: however, the detection optics are already in this location, and 15° from this location is about as close as the two paths can come to each other due to space constraints.

The excitation laser (Melles Griot, model 06DLD707) and its accompanying driver (Melles Griot, model 06DLD103) were chosen for the following reasons. Its wavelength was shorter than ~810 nm so as to markedly reduce the accidental detection the laser light on the detection arm of the optics, but close enough to the detected wavelength to assure that photoexcited electrons would not have to fall through numerous states before arriving at the sites from which they luminesce. It has a single-longitudinal, single-transverse mode structure. Multimode lasers are difficult to focus since each mode has its own *k*-vector. That is, each mode has its own direction and wavelength. Moreover, the modes are time dependent. Instead of all modes being present at a given fraction of the intensity at all times, one mode at a time usually dominates the others, and switching between modes is does not occur at constant light intensity. Thus, there are fluctuations in the intensity of the light, potentially leading to physics that would become convoluted with the results that were being sought. This laser also had the highest maximum power output, which gives one the flexibility to explore

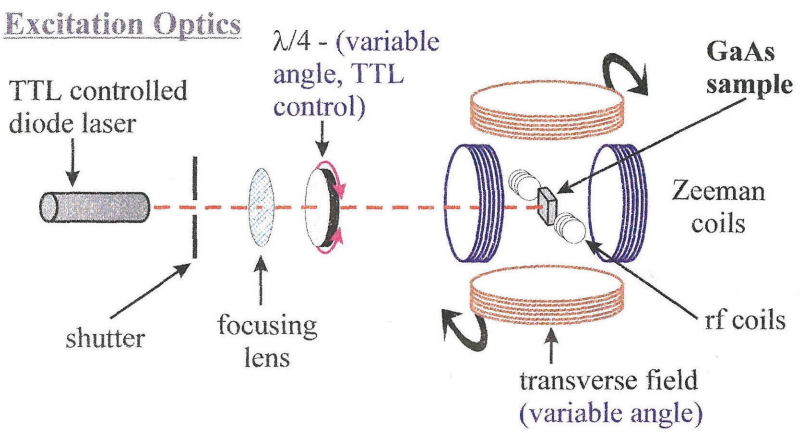


Figure 2.8 The excitation optics are aligned along a line that is about 15° from the axis of the magnetic field, which is colinear with the normal to the sample surface.

the effect of the intensity of the illumination upon the properties of the ONMR. Finally, it had fast rise times and fall times during digital modulation ($\ll 1\mu s$), enabling the laser to switched on and off precisely in selective windows of an NMR pulse sequence.

The detection optics are normal to the sample surface, which is colinear to the direction of the Zeeman magnetic field supplied by the superconducting coils mentioned in the description of the NMR probe. The optics configuration used during detection of NMR spectra is displayed in figure 2.9. As the luminescence emerges from the sample it is collected by a lens (1 inch in diameter, f = 10 cm) that is held ~5 cm from the sample, collimating a portion of the luminescence. This light then passes through a quarter-wave-plate followed by a linear polarizer, and they are oriented such that they pass the opposite handedness of light used in the excitation path (i.e., left circularly polarized light). The collimated beam is then focused by another lens through an 830 ± 5 nm bandpass filter, and onto a spherical ball lens that couples the light into a

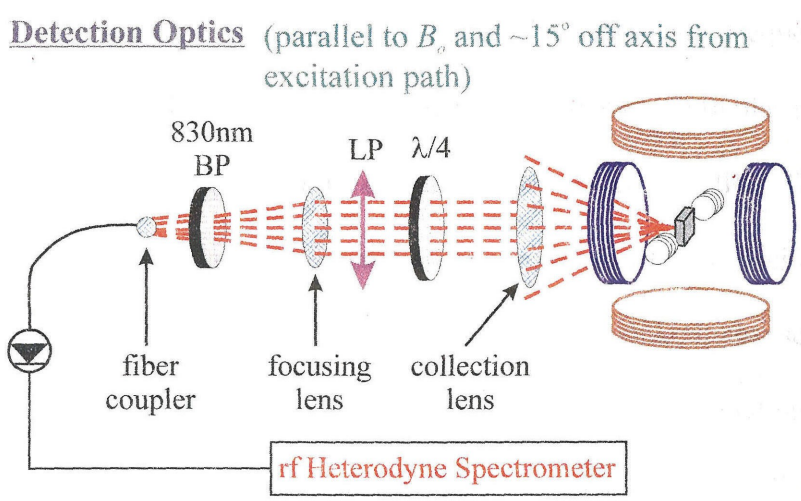


Figure 2.9 The detection optics are aligned along a line that is colinear the axis of the magnetic field as well as with the normal to the sample surface.

fiber coupled (200 um core, 0.24 NA, multimode) avalanche photodiode (EG&G Optoelectronics, model C30657-010QC-06), a sensitive detector of the total amount of light that reaches it.

The careful selection of the frequency and circular polarization of the light that will reach the photodiode in this apparatus serves several purposes. First, the frequency isolation allows the study of a particular luminescent feature and the nuclear sites in the vicinity of its origination. Secondly, the LBD signal is the modulation of the circular polarization of the luminescence at the difference frequency between two nuclear isotopes, and the detection of one of handedness of circular polarization separately from the other at a given time is required to detect ONMR. Thirdly, the selection of the orthogonal circular polarization when compared to the excitation path provides another level of filtration of the excitation light. Meanwhile, it also serves to set the null signal at a lower value. That is, the polarization used for excitation is likely present in larger amounts in the luminescence (this depends on spin dynamics and relative field orientations), providing a larger fractional modulation of the orthogonal polarization during to the NMR experiment. Finally, during the measurement of the photoluminescence spectrum, the normalized difference between the two orthogonal circular polarizations probes the electron spin dynamics at the sites that give rise to that feature in the luminescence.

The current from the avalanche photodiode is sent through a transformer that is also an 80 kHz high pass filter, and this is followed a 1.5 MHz low pass filter. This eliminates modulation of the photocurrent by 60 Hz electrical noise, higher harmonics of

the LBD frequency due to nonlinear real-time LBD conditions, higher harmonics of the LBD frequency in the photocurrent due to nonlinear detection such as saturation, and frequency components of the noise in the photocurrent that fall outside of this band. After the filters, the photocurrent enters the rf heterodyne spectrometer, which has been detailed elsewhere[1,3,4]. Finally, the amplified, mixed down signal produced by the spectrometer is digitized by a dual channel 16-bit A/D converter (Computer Boards, model CIO-DAS 1402).

D. RF and DC Magnetic Field Sources

The rf waveform is generated by an arbitrary waveform generator (Signatech model AWG502), passed through an rf switch, and amplified by a 500 W, 0.3 – 35 MHz, class-A, rf amplifier (ENI model 500A). The amplified signal is then sent through a high power directional coupler, and into the rf feed-through on the dewar cap where it is transmitted to the rf coil discussed in section B. Note the absence of tuning or matching circuits in the rf path. The rf coil must be able to simultaneously irradiate ^{75}As, ^{69}Ga and ^{71}Ga nuclei in order to fulfill its role in LBD experiments, and there is no way to tune and match to even two separate frequencies with the robustness required by our experiments while also achieving higher current (stronger B-field) in the coil at the specified frequencies. That is, most of the power may be reflected, but the current through the coil is high enough to fully meet the B-field producing requirements of the coil. This is possible only because in the low radio frequency band (~1.5 - 3.5 MHz), where ONMR

experiments are conducted in our lab, the coil is able to respond much more uniformly to rf input than if experiments were conducted at higher fields (where the complicated multiply tuned and matched circuitry would be required, and there would be virtually no benefit to the experiments). Finally, the class A rating on the rf amplifier is of utmost importance in enabling this unmatched circuit approach since this amplifier must be able to take nearly 100% reflected power coming back from the rf coil without harm to the amplifier. Class A amplifiers are rated to drive every circuit from an open to a short without incidence of failure on the part of the amplifier.

As previously mentioned, the Zeeman magnetic field sets the resonance frequency of the nuclear spins. Therefore, an accurate method of setting its magnitude is important to the success and reliability of the experimental apparatus. The magnitude of this field is controlled in two stages. In the first stage, a C program sends a GPIB command to a lock-in amplifier (Stanford Research Systems, model SR510) to set one of two TTL voltages. One of these voltages is the TTL voltage control input for an external power supply (HP, model 6264B) whose output is sent through the homebuilt current controller that is schematically illustrated in figure 2.10, providing ~10 ppm accuracy in setting the magnitude of the current. This controlled current is then sent to the Zeeman coils, which produce ~20.5 mT/A at the position of the sample. These coils are typically operated at ~250mT as calculated by the positions of the NMR resonance of the isotopes addressed in the discussion of the rf coils.

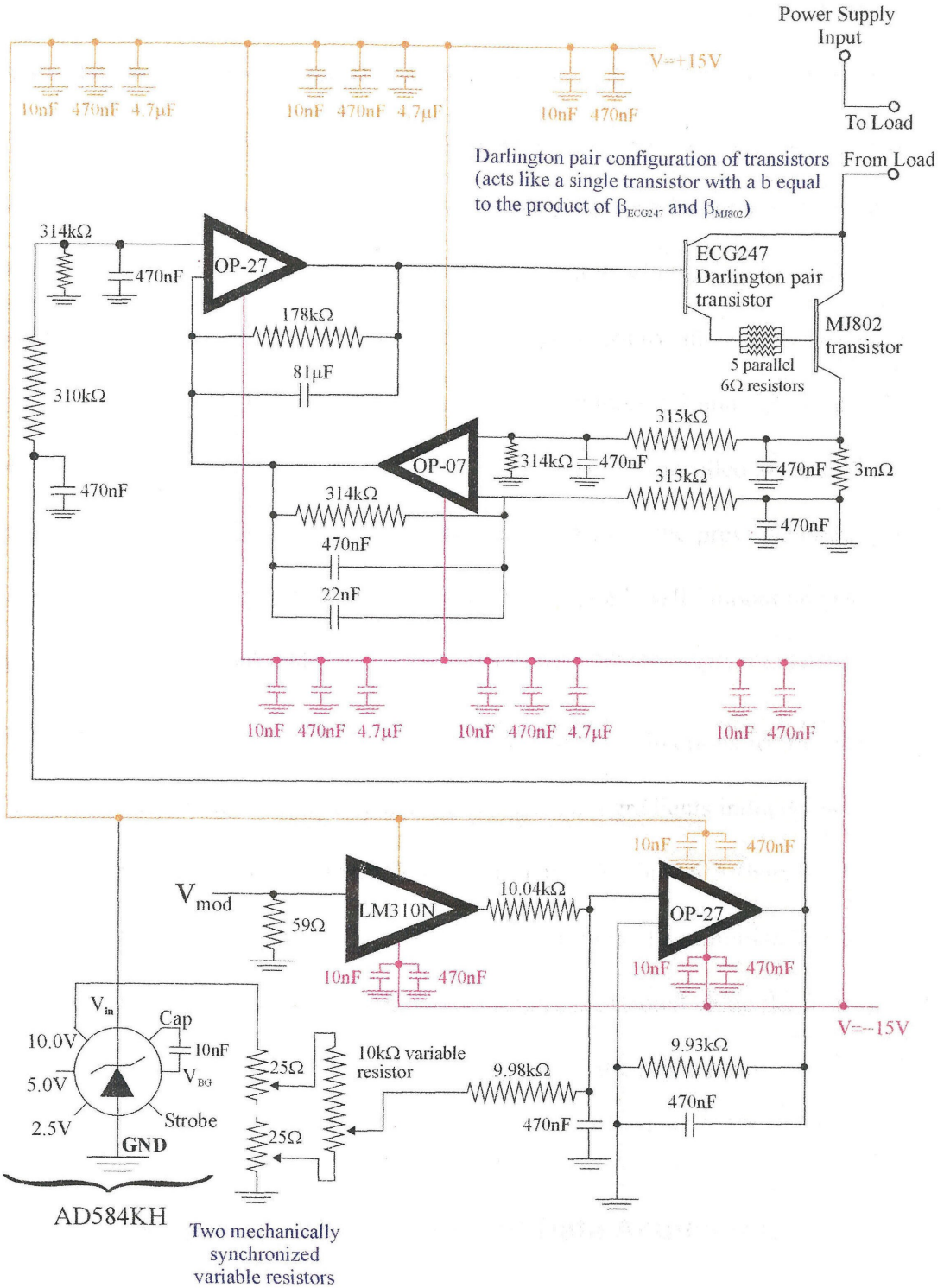


Figure 2.10 The current controller that is used to set the value of Zeeman field with an accuracy of ~1:10^5.

However, in figures 2.7 and 2.8, there is also an auxiliary set of coils that are orthogonal to both the Zeeman coils and the rf coils. This set of coils is external to the dewar, and each of the two sets is made of an array of 31×31 windings of square (1.7 mm on a side) copper wire supported on a copper yoke with 19 cm inner diameter, 31 cm outer diameter, and a width of 5.7 cm. Both yokes are held in variable angle, aluminum stand at a separation of 12 cm. The variable angle capacity allows the coils to be tilted $\pm 12^{\circ}$ in the direction of the rf coils (i.e., relative to figures 2.7 and 2.8, in and out of the page). The magnetic field produced by these coils is controlled by the second TTL channel from the SRS-510 lock-in amplifier mentioned in the previous paragraph. This TTL signal is the voltage control to a second power supply (HP, model 6675A) that feeds directly into the external coils.

The function of these coils is to provide alternate directions for the Zeeman field. This is necessary to more fully explore the electric field gradients induced by the presence of a homogeneous electric field, which are discussed in chapter 3 (part b of section VI), given that the secular portion of the NMR quadrupole Hamiltonian caused by these gradients vanishes when the Zeeman magnetic field is oriented along the 001-direction of the crystal (as it is for the superconducting magnets on the probe head).

E. The Computers - Command and Data Acquisition

The experiment and its timeline are orchestrated by a C program via GPIB control from a personal computer. Meanwhile, the data acquisition is controlled by another C

program on a second personal computer equipped with a dual channel 16-bit A/D converter (Computer Boards, model CIO-DAS 1402). The general sequence of events is as follows. The C program executing the experiment defines three parallel copies of the experimental timeline. Each parallel version is for a different device, and contains instructions relevant to that device and delay periods when it must wait for another device to complete its own commands. The first set of instructions is sent to an external digital word generator (Interface Technologies, model RS670) via the GPIB board (National Instruments, model AT-GPIB) that is plugged into an ISA expansion slot on the computer. The digital word generator is operated in 16-channel mode, where each channel is an independent TTL voltage control signal, and the internal system clock operates at 40 MHz, providing timing resolution of 25 ns. An additional set of instructions sent to the 12-bit arbitrary waveform generator (Signatech model AWG502) that is plugged into the second ISA slot of the same computer. It is responsible for digitally generating the rf waveforms and dispensing them at the exact moment they are needed. Pulses are sent only to the ampilifier and on to the rf coil, and the LBD reference frequency is sent to the spectrometer to be used to mix down the photocurrent to audio frequencies. Finally, one last set of instructions is maintained by the C program itself for operation of other devices (lock-in amplifier and laser diode driver) by GPIB command.

It is advantageous for as many of these sources as possible to run on the same clock, given that different clocks can have small drift that can add up to noticeable experimental interference. For example, if two 40 MHz clocks have 10 Hz drift between them, the difference after 5 seconds will be 50 cycles or ~1.25 μs, which is ~1.2μs too

long if one clock controls the rf pulses and the other controls the laser pulses that need to be in the windows between the rf pulses (see chapter 5). Synchronization of the two can be accomplished by configuring the AWG502 to accept a TTL pulse that tells it to begin running on a 40 MHz reference supplied by the RS670.

F. References

1) Hwang, J. Y. *NMR of Dilute Sites in GaAs*; California Institute of Technology: Pasadena, 1994, pp 149.

2) Carson, P. J. *I. Ultrasensitive Surface NMR Using Parahydrogen Spin Labeling, and II. High-Resolution Optical NMR of Semiconductor Heterostructures Using Larmor Beat Detection.*; California Institute of Technology: Pasadena, 1997.

3) Marohn, J. A. *I. Multiple-Pulse Radio-Frequency Gradient Nuclear Magnetic Resonance Imaging of Solids. II. Optical Nuclear Magnetic Resonance Analysis of Epitaxial Gallium Arsenide Structures.*; California Institute of Technology: Pasadena, CA, 1996, pp 143.

4) Kempf, J. G. *Probing Quantum Confinement at the Atomic Scale with Optically Detected Nuclear Magnetic Resonance*; California Institute of Technology: Pasadena, 2001, pp 172.

5) Ossau, W.; Bangert, E.; Weimann, G. *Solid State Communications* **1987**, *64*, 711.

6) Yuan, Y. R.; Pudensi, M. A. A.; Vawater, G. A.; Merz, J. L. *Journal of Applied Physics* **1985**, *58*, 397.

7) Marohn, J. A.; Carson, P. J.; Hwang, J. Y.; Miller, M. A.; Shykind, D. N.; Weitekamp, D. P. *Physical Review Letters* **1995**, *75*, 1364-1367.

8) Yuan, Y. R.; Merz, J. L.; Vawter, G. A. *J. Lumin.* **1988**, *40*, 755.

9) Shen, J. X.; Oka, Y.; Ossau, W.; Landwehr, G.; Friedland, K. J.; Hey, R.; Ploog, K.; Weimann, G. *Solid State Commun* **1998**, *106*, 495.

10) Shen, J. X.; Oka, Y.; Hu, C. Y.; Ossau, W.; Landwehr, G.; Friedland, K. J.; Hey, R.; Ploog, K.; Weimann, G. *Phys. Rev. B* **1999**, *59*, 8093.

11) Shen, J. X.; Pittini, R.; Oka, Y.; Kurtz, E. *Phys. Rev. B* **2000**, *61*, 2765.

III. Optically Relevant Defects - Theory and Calculations

Interpretation of many of the experimental phenomena we have observed has required extensive theoretical and numerical calculations of properties and phenomena in the GaAs lattice, especially around shallow defects. These computations are reviewed here on a component-by-component basis. First, a simple description of how one may locate all of the nuclei in the lattice, and perform a reasonably easy, efficient summation over all of these sites is presented. The discussion then covers the signal weighting as a function of spatially dependent polarization (ONP) before the experiment and during the detection as well as the optical detection weighting. Next, computations of electric field gradients due to both spatially varying electric fields, and homogeneous electric fields in GaAs are covered, including the spatial variation of the fluctuations in these electric field gradients as well as the implications of this new fluctuation hypothesis on the theory of the ONP rate. Finally, the calculation of the coherent evolution of the system without resorting to common approximations such as the nearly ubiquitous rotating wave approximation is described. The sum total of these components can be integrated to yield a site by site calculation of single nucleus spin dynamics that displays rigorously accurate rf effects upon each sites total Hamiltonian, including the Knight shift and nonsecular quadrupolar interactions. A sample calculation of this variety is included in appendix A of this thesis.

A. Summation over all sites in GaAs

The GaAs lattice consists of two interpenetrating face-centered-cubic (fcc) lattices. The gallium nuclei reside in one of these fcc lattices sites, and the arsenic nuclei reside in the other. Moreover, the sites of one lattice are simply offset from the other lattice by a translation by one quarter of a unit cell length (i.e., $5.65\text{Å}/4$) along each of the crystal axes (or, equivalently by $\sqrt{3}(5.65\text{Å})/4$ in the 111-direction). This is illustrated in figure 3.1 where the purple points comprise one fcc lattice and the turquoise points comprise a separate fcc lattice. It can be difficult to see that the turquoise points do indeed comprise an equivalent lattice. However, note that lattice indices of each turquoise point are produced by adding (111) to the lattice indices of a purple point. That is, the turquoise points form a replicate lattice based on the purple points but displaced by (111). Thus, the point (111) is a corner of a unit cell like the

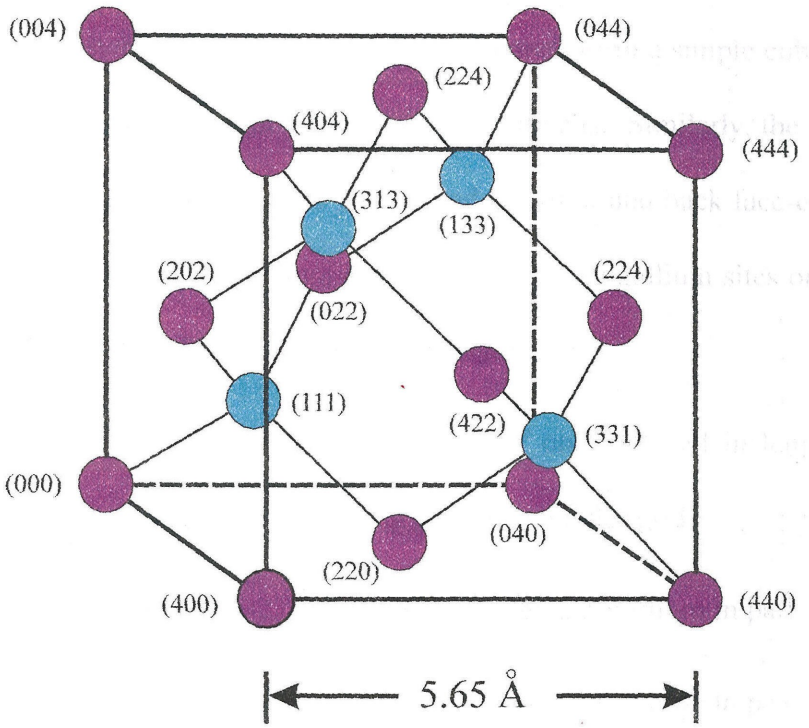


Figure 3.1 Representation of a unit cell in the GaAs lattice.

point (000), and the other three blue points are the face-centers of this new unit cell like the points (022), (220), and (202) in the first unit cell.

These concepts may extended one step further. Indeed, each fcc lattice can be viewed as a set of four interpenetrating simple cubic lattices. Looking again at figure 3.1, the corners of the unit cells are the most obvious of the simple cubic lattices. Furthermore, the top and bottom face-centers form a simple cubic lattice with the top and bottom face-centers of all of the other unit cells. Similarly, the right and left face-centers form another simple cubic lattice, and the front and back face-centers form the final one. Therefore, one way to sum over either all of the gallium sites or all of the arsenic sites is to follow the following rudimentary algorithm

1. Start at (000). Take steps of one unit cell in length along each of the crystal axes, mapping out a rectangular grid.
2. Start at (022). Repeat the stepping algorithm in part 1.
3. Start at (202). Repeat the stepping algorithm in part 1.
4. Start at (220). Repeat the stepping algorithm in part 1.

There is clearly much redundancy in this algorithm as the same thing is repeated four times from different starting points. This may easily be eliminated as follows.

It is readily verified that the coordinates of each of the purple points in figure 3.1 can be obtained by translation from (000) by either (022), (202), (220), or some combination of these. Rephrasing this, we can say that for any site in the lattice of purple

points its coordinates will sum to an integer multiple of four. This is a very useful result since it leads to the following alternate algorithm.

> Start at (000). Take steps of one half of a unit cell in length along each of the crystal axes, mapping out a denser rectangular grid. At each point ask if the sum of all lattice indices is an integer multiple of four, and do the calculation only if the answer is yes.

Besides being more efficient, this technique allows one to include the other fcc lattice by including a secondary conditional statement that asks if the sum of all lattice points minus three is an integer multiple of four.

The sum over all sites can be reduced to a sum over a subset of sites due to the symmetry of the crystal in cases where this symmetry is unbroken. While it is true that zinc-blend crystal structures such as GaAs lack inversion symmetry, each point in the lattice has S4 symmetry. Consequently, if we are summing over lattice sites in the presence of a perturbation with spherical symmetry (such as a point defect) that is centered at a lattice point, then summation over any quadrant of the lattice about that perturbation is equivalent to summation over any other quadrant. Indeed, if we are only concerned about the properties of elements of only one of the fcc lattices, then, so long as the center of symmetry is centered at one of these lattice points, we may restrict our calculation to a single octant and still be rigorously accurate.

In the case of summation over sites near a point defect, it is also efficient to limit our summation to those sites within a certain radius of the defect rather than using

rectangular boundaries. This is justified since contributions to the sum decrease with increasing radius (otherwise one would have to sum to the edges of the crystal). For summation over a quadrant using lattice indices i, j, and k, we let i take on values from $-n$ to n while j goes from zero to $\sqrt{n^2 - i^2}$ and k goes from zero to $\sqrt{n^2 - i^2 - j^2}$. This reduces the workload by approximately the ratio of the volume enclosed by the summation over spherical boundaries divided by the volume enclosed by the summation over rectangular boundaries. This can be estimated, in the limit of large n, by the ratio $\frac{\pi n^3/3}{2n^3} = \frac{\pi}{6} \approx 0.524$, and exact numerical evaluation of the ratio of the number of sites included with each type of boundaries gives about 0.516 for $n = 144$ (~200 Å).

Finally, to be exact when performing the lattice sum over one of these reduced sets of sites, the nuclei that are bisected by our chosen boundaries must be weighted by a factor of one half for each boundary plane that bisects them. This is cumbersome since these sites must be evaluated separately, or additional conditional statements must be applied during the calculation to determine the proper weighting of a given lattice site.

What error would be introduced if we ignored this weighting? We should expect this error to be very small since it goes as ratio of the surface area to the volume of the region included in the summation. For calculation of a quadrant out to $n = 144$, this error is only about 1% for either a uniformly weighted sum or an exponentially weighted sum (with a 1/e coefficient of 100 Å).

B. Polarization and Detection Weighting

The ONP process at the site of a point defect leads to the characteristic nuclear polarization and NMR signal distributions about the crystallographic point defect in both the position domain and the frequency domain. These distributions will now be considered. The two limiting cases of infinitely fast spin diffusion and negligible spin diffusion will now be addressed using a continuous medium approximation with the expectation that the observed signal will lie in some intermediate region between these polar opposites.

In the case of negligible spin diffusion, the ONMR signal from the nuclei in the vicinity of an ORD can be expressed as

$$S(t_{\mathrm{ONP}},t_1,t_2)=\sum_{\mathbf{r}}P(\mathbf{r},t_{\mathrm{ONP}})s(\mathbf{r},t_1)U(\mathbf{r},t_2) \tag{3.1}$$

where $P(\mathbf{r},t_{\mathrm{ONP}})$ is the polarization of the nucleus at a given position $\mathbf{r}$ at a time t_{ONP} (the end of the ONP period), $s(\mathbf{r},t_1)$ is the NMR signal that is dictated by the nuclear spin Hamiltonian at a given position and that the particular details of the NMR experiment, and $U(\mathbf{r},t_2)$ is the optical detection weighting of the nucleus at a given position when a certain detection condition is used for a duration t_2.

Spin diffusion effects could be negligible for one of two reasons. First, the rate of ONP may be much more rapid than the rate of spin diffusion on the time scale of the experiment for the nuclei detected by the experiment. That is, the experiment is too short to capture spin diffusion, but long enough to capture ONP. Second, the spin diffusion

process may be short circuited in the vast majority of the region around the point defect by the large differences in the Quadrupole and Knight Shift Hamiltonians of neighboring nuclei, preventing the spread of nuclear polarization to those sites where it could diffuse rapidly to other nuclei (the so-called frozen core effect, which is approximated for the case of an explicit spin diffusion calculation in section H of chapter 4). In any case, what would be observed is the sum of the ONP behavior of each of the individual nuclear sites over the time frame in question as expressed in equation (3.1).

The polarization weight P in equation (3.1) for the nuclei around a point defect will be calculated first. As discussed in chapter 1, spin polarized electrons that are in hyperfine contact with nuclear spins will tend to polarize the nuclei. This polarization will develop with a time constant T_1 that depends on the strength of this hyperfine contact. That is,

$$P(t) = 1 - \exp\left(-\frac{t}{T_1}\right). \tag{3.2}$$

Moreover, the time constant T_1 can be very well approximated as[1]

$$\frac{1}{T_1} = \Gamma A^2 \frac{2\tau_c}{1+\omega^2\tau_c^2}, \tag{3.3}$$

where Γ is the probability that the defect state is occupied by an electron, A is the strength of the hyperfine interaction of this electron with a nucleus, τ_c is the correlation time of the fluctuations of the hyperfine interaction, and ω is the Larmor frequency of the electron in the total magnetic field which it feels (the externally applied field plus the nuclear

hyperfine field). Of course, the hyperfine interaction is spatially dependent for those sites around an occupied defect, so equations (3.2) and (3.3) must be revised to bring them in line with equation (3.1). This yields

$$P(\mathbf{r},t) = 1 - \exp\left(-\frac{t}{T_1(\mathbf{r})}\right) \tag{3.4}$$

and

$$\frac{1}{T_1(\mathbf{r})} = \Gamma\left[A(\mathbf{r})\right]^2 \frac{2\tau_c}{1+\omega^2\tau_c^2} \tag{3.5}$$

Equation (3.5) can, in turn, be expressed as

$$\frac{1}{T_1(\mathbf{r})} = \Gamma\left[\tfrac{2}{3}\mu_o g_o \mu_B \gamma_{\alpha,\mathrm{Hz}} \left|\psi_\alpha(\mathbf{r})\right|^2\right]^2 \frac{2\tau_c}{1+\omega^2\tau_c^2} \tag{3.6}$$

where the subscript α refers to the nuclear species present at position $\mathbf{r}$ (α = ^{71}Ga, ^{69}Ga, or ^{75}As). Note that equation (3.6) demonstrates that the rate of ONP depends on the fourth power of the amplitude of the electronic wavefunction at the nuclear site. That is, at nuclear sites where the amplitude of the wavefunction has dropped by a factor of two, rate of ONP will drop by a factor of sixteen.

In the case of an occupied donor, equation (3.6) becomes

$$\frac{1}{T_1(\mathbf{r})} = \Gamma\left[\tfrac{2}{3}\mu_o g_o \mu_B \gamma_{\alpha,\mathrm{Hz}} \left(\tfrac{2}{a_o}\right) \exp\left(-\frac{2\|\mathbf{r}\|}{a_o}\right)\right]^2 \frac{2\tau_c}{1+\omega^2\tau_c^2} \tag{3.7}$$

or, collecting terms, this may be rewritten as

$$\frac{1}{T_1(\mathbf{r})} = \frac{1}{T_1(0)} \exp\left(-\frac{4\|\mathbf{r}\|}{a_o} \right) \tag{3.8}$$

with

$$\frac{1}{T_1(0)} = \Gamma \left[\tfrac{2}{3} \mu_o g_o \mu_B \gamma_{\alpha,\mathrm{Hz}} \left(\tfrac{2}{a_o} \right) \right]^2 \frac{2\tau_c}{1+\omega^2 \tau_c^2}. \tag{3.9}$$

Additionally, inserting equation (3.8) into equation (3.4) yields

$$P(\mathbf{r},t) = 1 - \exp\left(-\frac{t}{T_1(0)} \exp\left(-\frac{4\|\mathbf{r}\|}{a_o} \right) \right). \tag{3.10}$$

In the absence of spin diffusion, this equation provides an adequate description of the polarization weighting at a given nuclear site after a given duration of ONP. However, the detected nuclear polarization is generally the sum over all radial positions (the exception being the case of radially resolved Knight shifts, addressed later in this section). That is,

$$P(t) = \sum_{\mathbf{r}} 1 - \exp\left(-\frac{t}{T_1(0)} \exp\left(-\frac{4\|\mathbf{r}\|}{a_o} \right) \right). \tag{3.11}$$

Note that this polarization will be ever increasing until the edge of the crystal finally limits the amount of polarizable material that can be included in the sum. Yet, only those nuclei that can be optically detected will actually contribute to the optically measured polarization. The function $U(\mathbf{r},t_2)$ must then be addressed in order to proceed further. An extremely reasonable starting point for this function is just the $A(\mathbf{r})$, the

hyperfine coupling of the nucleus at position **r** to the electron bound to the shallow donor, since the optical detection at any given nucleus is proportional to the strength of the hyperfine interaction at that nuclear site. This leads to a detection factor with functional form that is proportional to $\psi^2(r)$ of the electron about the shallow donor. Using this optical weighting factor is essentially an assumption that the ONMR experiment is detected using real-time detection of the beat pattern between the two precessing nuclear species, and that spin diffusion does not interfere with the polarization during detection. If point-wise detection using spin-locked nuclear magnetization is being employed, then the effectiveness of the spin-lock in the face of strong quadrupole and Knight shift interactions near the donor site must be considered, calculating the nuclear spin-Hamiltonian at each site, propagating the density matrix through the detection period, and, finally, applying the $\psi^2(r)$ optical detection weighting to this calculation to obtain an accurate answer.

Simply applying the $\psi^2(r)$ weighting to equation (3.11), produces

$$P_{OD}(t) = \left(\frac{2}{a_o}\right)\sum_{\mathbf{r}} \exp\left(-\frac{2\|\mathbf{r}\|}{a_o}\right)\left(1-\exp\left(-\frac{t}{T_1(0)}\exp\left(-\frac{4\|\mathbf{r}\|}{a_o}\right)\right)\right). \qquad (3.12)$$

One may attempt to probe the accuracy of this approach by conducting a set of spin echo experiments where the ONP time before each spin echo experiment is incremented. The integrated peak intensity versus ONP time then will yield the sum of all the polarization weights for the individual atomic sites weighted by the optical detection profile. That is,

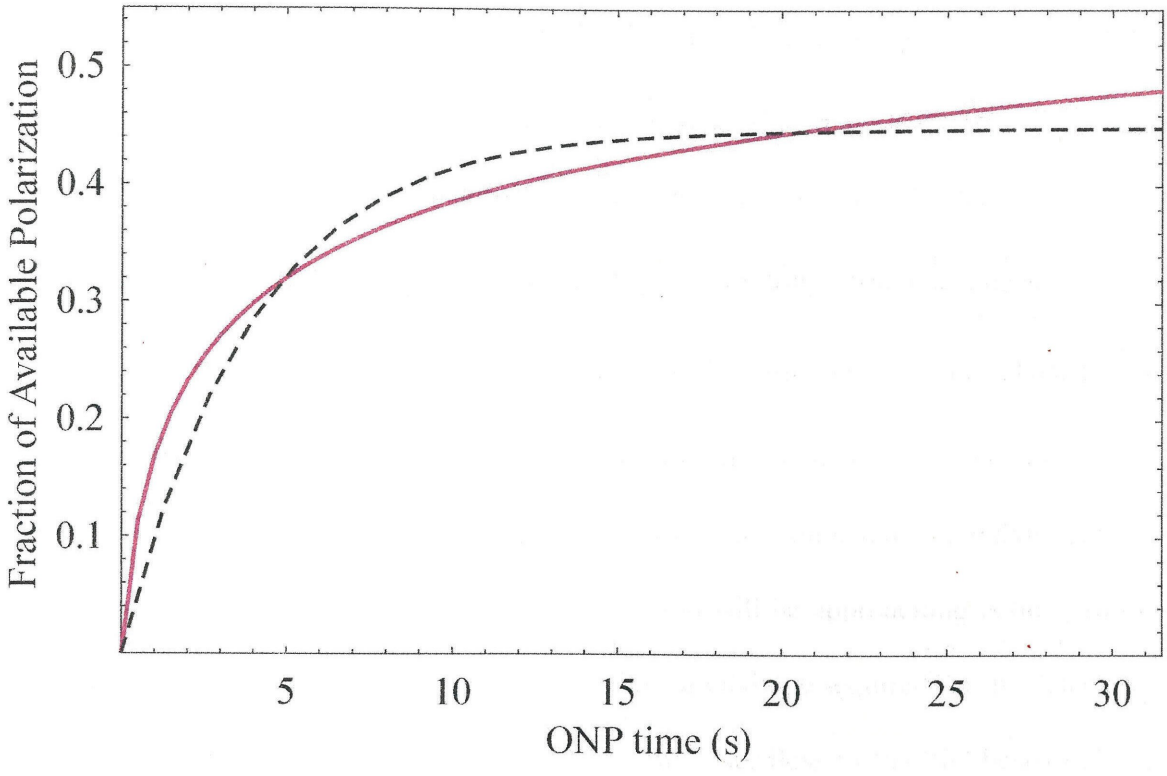


Figure 3.2 The weighted summation of the polarization from all nuclear sites within 300A of a shallow hydrogenic donor with a 100A Bohr radius and $T_1(0) = 80\text{ms}$ (solid purple line) and the best fit exponential to this particular data series (dashed black line). It will require ~10,000s for the purple line to rise to within one percent of reaching unity (its asymptotic value).

the accumulated nuclear polarization versus the ONP time T_{ONP} used to produce it is simply a weighted sum of the single exponentials for each nuclear site.

Evaluation of this sum has been accomplished both in Mathematica and by developing and using Fortran90 programs (these yield the same answers as the Mathematica calculations, but run approximately six times faster). The results of a representative calculation are shown in figure 3.2. The value of $T_1(0)$ in this

computation is 80 ms, and the sum is carried out to $\|\mathbf{r}\| = 3a_0$, assuring that ~99.4% of the donor-bound electron's probability density is included (i.e., at least 99.4% of the total detectable nuclear signal is included). Note the fast initial rise due to sites at $\|\mathbf{r}\| \lesssim a_0$ (sites with $T_1 \leq 4.4\text{s}$) and the exceedingly slow long-term rise due to sites at $\|\mathbf{r}\| \gtrsim a_0$ (sites with $T_1 \geq 4.4\text{s}$). Indeed, the units on the ordinate axis were chosen such that, when the calculation is done at sufficiently long times that all of the included nuclei are completely polarized and sufficiently far out from the donor that the radial sum has converged, the asymptotic value that the computation will be approaching is one. In the case of this particular calculation, ~10,000 seconds of ONP are required for the detectable polarization to be within 1% of its asymptotic limit. Needless to say, the best-fit single exponential is strongly dependent on how much of this long tail is included in the fitting process. However, even if it is possible to ignore spin diffusion for some earlier portions of this curve, certainly spin diffusion will be a factor long before 10,000s in any real GaAs crystal. Therefore, in a real sample, one should, at best, expect a transition from behavior somewhat like the beginning portion of this curve to a premature asymptotic limitation of the detectable polarization.

It should be pointed out, however, that these calculations are all relatively cumbersome, and they do not yield any analytical answers. Making a continuum approximation to the discreet atomic nature of the crystalline solid may advance the understanding of this physical system. Such a continuous dielectric medium approximation produces an analytical solution for the radial distribution of the NMR

signal about a shallow donor as well as for both the time and frequency domain of this same NMR signal. Moreover, the differences between the discrete calculations and the calculations using the continuous dielectric medium approximation indicate that experimental detection of the difference between the signals would require a signal to noise ratio at least of the order of magnitude of a thousand to one to for them to convincingly rise from the grass of spectral noise. This represents a considerably higher ratio of signal to noise than we have achieved to date, and, therefore, the continuous dielectric medium approximation should be sufficiently sound to allow accurate calculation of most physical properties of this system. Thus,

$$P(\mathbf{r},t) = 1 - \exp\left[-\frac{t}{T_1(0)} \exp\left(-\frac{4\|\mathbf{r}\|}{a_0} \right) \right] \tag{3.13}$$

gives way to

$$P(r,t) \approx \left(1 - \exp\left[-\frac{t}{T_1(0)} \exp\left(-\frac{4r}{a_0} \right) \right] \right) \tag{3.14}$$

where r the continuous variable has replaced $\mathbf{r}$ the discrete vector. Equation (3.14) is plotted in figure 3.3 for four different values of time, and the x-y axes are labeled in dimensionless units of r / a_0

In the time domain, the explicit expression for the optically detectable polarization

$$P_{OD}(t) = \left(\frac{2}{a_0} \right) \sum_{\mathbf{r}} \exp\left(-\frac{2\|\mathbf{r}\|}{a_0} \right) \left(1 - \exp\left(-\frac{t}{T_1(0)} \exp\left(-\frac{4\|\mathbf{r}\|}{a_0} \right) \right) \right) \tag{3.15}$$

gives way to

$$P_{OD}(t) \approx 4\pi\left(\frac{2}{a_o}\right)\int_{r=0}^{r=r_{max}} \exp\left(-\frac{2r}{a_o}\right)\frac{r^2}{a_o^3}\left(1-\exp\left[-\frac{t}{T_1(0)}\exp\left(-\frac{4r}{a_o}\right)\right]\right)dr \qquad (3.16)$$

with r_{max} effectively determined by the length of the ONP period as well as by the signal to noise ratio of the experiment. To be rigorously accurate one should set $r_{max} = \infty$, and this will be required for symbolic integration, which will be performed in Mathematica.

The symbolic integration yields

$$P_{OD}(t) = 1 - {}_qF_p(\{\tfrac{1}{2},\tfrac{1}{2},\tfrac{1}{2}\};\{\tfrac{3}{2},\tfrac{3}{2},\tfrac{3}{2}\};-\tfrac{t}{T_1(0)}) \qquad (3.17)$$

where ${}_qF_p$ is the generalized complex hypergeometric function. Evaluation of this function in Mathematica is accomplished by the following series solution

$${}_pF_q(\mathbf{a},\mathbf{b},z) = \sum_{k=0}^{\infty}\left(\frac{\prod_{i=1}^{m}\prod_{j=0}^{k-1}(j+a_i)}{\prod_{i=1}^{m}\prod_{j=0}^{k-1}(j+b_i)}\right)\frac{z^k}{k!} \qquad (3.18)$$

which is numerically stable for $z \lesssim 45$ (times less than 3.6s, given our sample $T_1(0)$ of 80ms); however, this class of functions is a very well studied, and other representations of the function that are stable over different intervals are likely available, enabling one to compute values along the entire curve. Figure 3.4 above shows this function plotted in Mathematica, where all of our earlier plotting was done, over the region that the series is

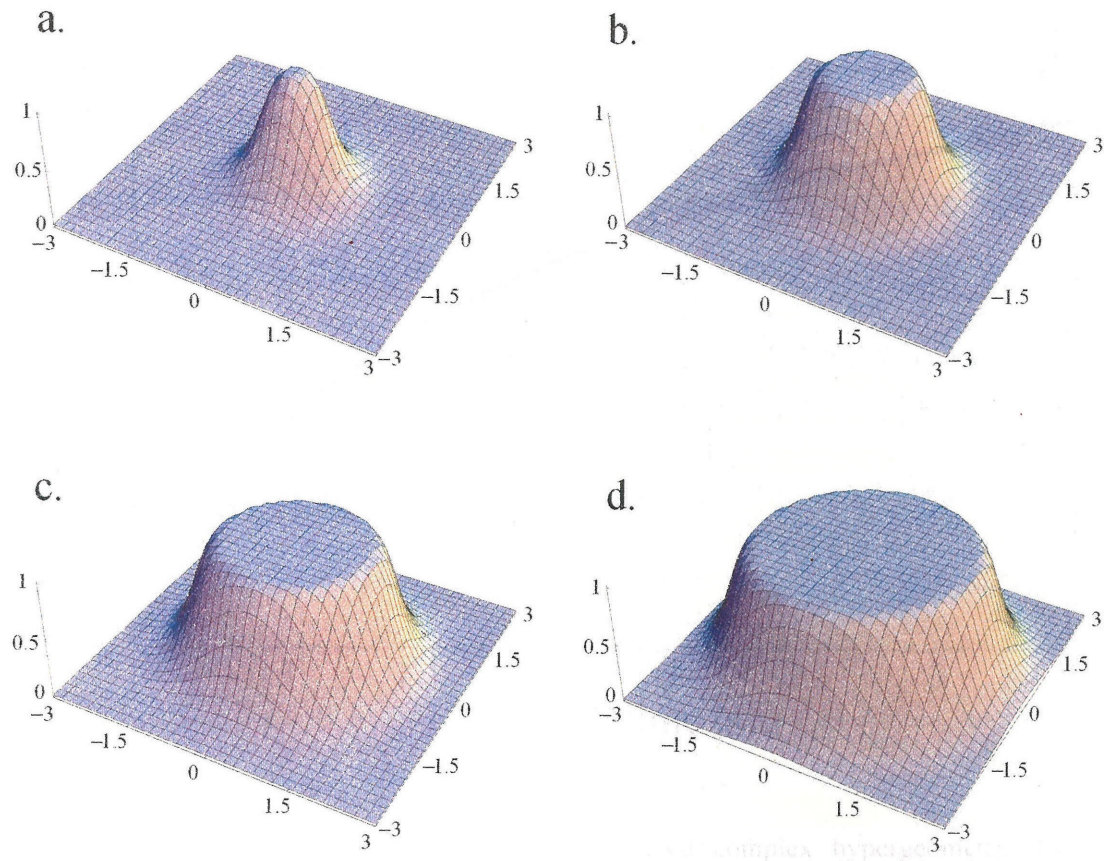


Figure 3.3 Time progression of the nuclear polarization as expressed in equation **(3.14)** at 10, 100, 1000, and 10,000 times $T_1(0)$ in a, b, c, and d, respectively. The x-y axes are plotted in dimensionless units of r/a_0. Note that each order of magnitude increase in the ONP time increases the polarized area by roughly $a_0/2$ in every direction.

useful, and on the same graph as the explicit sum over sites and the single exponential best-fit to the sum over sites.

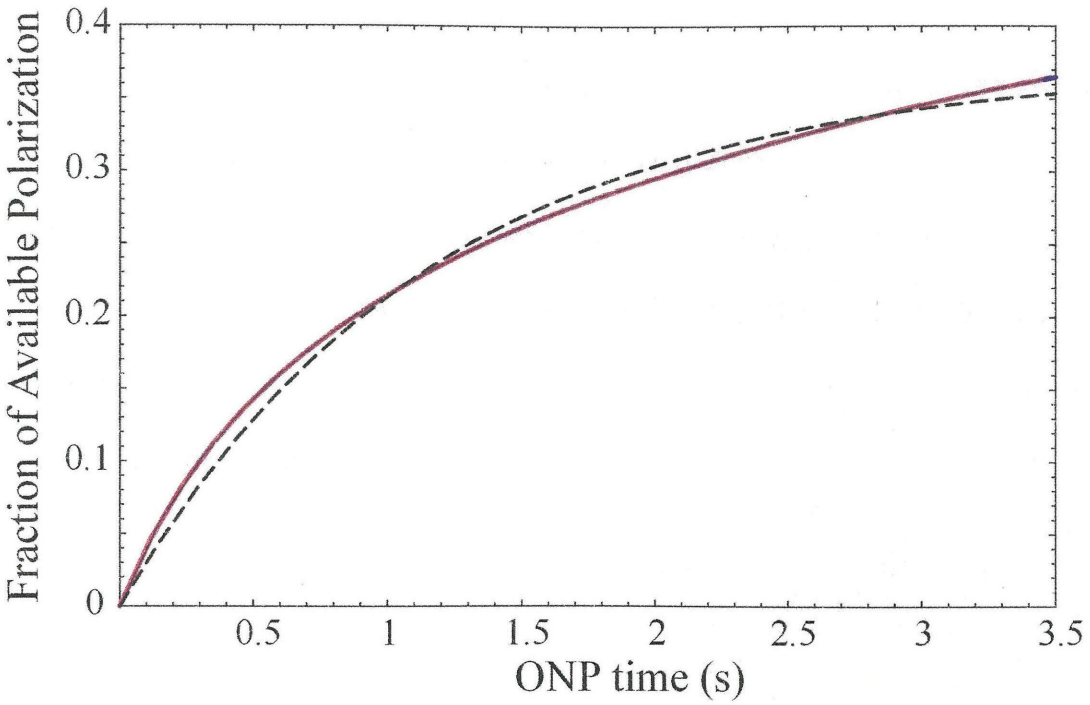


Figure 3.4 Overlay of the plot of the generalized complex hypergeometric function ${}_qF_p(\{\frac{1}{2},\frac{1}{2},\frac{1}{2}\};\{\frac{3}{2},\frac{3}{2},\frac{3}{2}\};-\frac{t}{T_1(0)})$ (solid blue line) on the summation over sites (solid purple line) and the single exponential best-fit to the summation over sites over this interval (dashed black line). The line from the hypergeometric, achieved using the continuum approximation, lies precisely on top of the line from the summation over all sites. Meanwhile, the best-fit single exponential can't duplicate the functional form of either of these, even over this abbreviated interval.

C. Shallow Donor-Knight Shift Calculations

Continuing upon the course set in the previous section, one of the types of NMR signal function that we have encountered in laboratory experiments will now be considered. That is the Knight shift, as discussed in section VI of chapter 1. This nuclear interaction is just the resonance shift provided by the Hyperfine coupling to spin polarized electrons. As such, when this resonance shift is the parameter being measured by the NMR experiment, equation (3.1) can be updated to

$$S(t_{ONP},t_1,t_2)=\left(\frac{2}{a_o}\right)\sum_{\mathbf{r}}\left(1-\exp\left(\frac{-t_{ONP}}{T_1(0)}e^{\frac{-4\|\mathbf{r}\|}{a_o}}\right)\right)\exp\left(-i2\pi t_1 f_0 e^{\frac{-2\|\mathbf{r}\|}{a_o}}\right)e^{\frac{-2\|\mathbf{r}\|}{a_o}} \qquad (3.19)$$

or in terms of continuum approximations

$$S(t_{ONP},t_1,t_2)=\left(\frac{8\pi}{a_o}\right)\int_{r=0}^{\infty}\frac{r^2}{a_o^3}\left(1-\exp\left(\frac{-t_{ONP}}{T_1(0)}e^{\frac{-4\|\mathbf{r}\|}{a_o}}\right)\right)\exp\left(-i2\pi t_1 f_0 e^{\frac{-2\|\mathbf{r}\|}{a_o}}\right)e^{\frac{-2\|\mathbf{r}\|}{a_o}}dr. \qquad (3.20)$$

Note that the loss of NMR signal in the detection period to repolarization of the nuclei has not been taken into account in these equations. This will be introduced toward the end of this section. The present calculation is relevant to real-time detection of ONMR while the latter discussion (the one including repolarization during detection) is relevant to spin-locked detection of ONMR signals.

It is instructive to contemplate the special case of equation (3.20) where the polarization weighting is assumed to be approximately spatially uniform. Progressively more general cases will then be considered. While it is true that we are paring down the

problem to lay a foundation for increasing levels of complexity, this is not just an academic exercise. ONP can produce spatially uniform polarization in either of the two following cases. First, if the ONP rate is exceedingly fast over the entire optically detected region on the experimental time scale, equation (3.4) will be equal to one for all sites considered. Similarly, if spin diffusion is very rapid compared to the ONP rate of the vast majority of optically detected nuclei, these nuclei will share polarization much more rapidly than it can accumulate at any given site, producing nearly uniform nuclear polarization over the optically detected area. Moreover, while there will always be a negligibly small number of nuclei with ONP rates that are faster than any possible spin diffusion, these sites are also the most likely to experience frozen core effects and extreme resonance shifts which further reduce their effect on the ONMR signal of the nuclei in the optically detected region. Additionally, if spin-lock detection were being used, the optical detection would be strongly weighted against sites with ultra fast ONP as these sites re-ONP during detection, losing all memory of their contribution to the NMR signal. The particulars of spin diffusion will be ignored here. The explicit case of simultaneous spin diffusion and ONP is considered in chapter 4.

Thus, with spatially uniform polarization equation (3.1) becomes

$$S\left(t_{\mathrm{ONP}},t_1,t_2\right)=\sum_{\vec{r}} s\left(\mathbf{r},t_1\right)U\left(\mathbf{r},t_2\right), \tag{3.21}$$

and equation (3.20) becomes

$$S\left(t_1\right)=\left(\frac{8\pi}{a_o}\right)\int_{r=0}^{\infty}\frac{r^2}{a_o^3}\exp\left(-i2\pi t_1 f_0 e^{\frac{-2\|\mathbf{r}\|}{a_o}}\right)e^{\frac{-2\|\mathbf{r}\|}{a_o}}\,dr\,. \tag{3.22}$$

Plotting just the detection weighting as a function of radius produces the familiar density profile of a 1s orbital. Indeed, one might expect that, since the peak of that curve is at $r = a_0$, the peak of the NMR spectrum will also be at the Knight shift of the nuclei at $r = a_0$. This couldn't be further from the truth. This can be illustrated very clearly by pursuing a u-substitution on equation (3.22), with

$$u = e^{\frac{-2\|\mathbf{r}\|}{a_0}}. \tag{3.23}$$

This yields

$$S(t_1) = \left(\frac{2\pi}{a_0}\right) \int_{u=0}^{1} \log^2(u) \exp(-i2\pi t_1 f_0 u)\, du. \tag{3.24}$$

If the Fourier transform of this function is now taken, its frequency domain spectrum will be obtained, yet this is a trivial operation since this equation is already an inverse Fourier transform of the function $\log^2(u)$ over the region from $0 \geq u \geq 1$ ($0 \leq f \leq f_0$ since $u = f/f_0$) back into the time domain. Therefore, $\log^2(u)$ is the functional form of the spectrum, yet this function has a sharp discontinuity at zero frequency. The value of the function is infinite at $f = 0$. This occurs because the electronic density, and thus the signal strength, is falling exponentially with increasing radial distance, but the Knight shift is falling with the identical exponential function. This means that, while the signal from a shell of nuclei at a given radius is declining extremely rapidly with increasing radial distance, the difference in frequency space between the Knight shifts of the two shells is also decreasing at the same rapid pace, allowing the signal to pile up to an

infinite spike for an infinite crystal with uniform polarization. The most important conclusion that can be drawn from this is that any attempt at analyzing an experimental radial resolution of the Knight shift where the polarization is anywhere near uniform *must* address the spatial profile of the polarization to obtain an accurate answer. That is, the peak of that spectrum will be determined by the exact profile of the polarization's decline rather than by the value of the Knight shift at any given radius, as has been assumed. A corollary to this point is that the broadening of the line by the Knight shift is a much more sensitive probe of the size of the hyperfine interaction than the shift of the peak of the resonance curve.

Returning to the case of non-uniform polarization, equation (3.20) will now be broken into the following two pieces

$$S(t_{ONP},t_1,t_2)=\left(\frac{8\pi}{a_o}\right)\int_{r=0}^{\infty}\frac{r^2}{a_o^3}\exp\left(-i2\pi t_1 f_0 e^{\frac{-2\|\mathbf{r}\|}{a_o}}\right)e^{\frac{-2\|\mathbf{r}\|}{a_o}}\,dr$$
$$-\left(\frac{8\pi}{a_o}\right)\int_{r=0}^{\infty}\frac{r^2}{a_o^3}\exp\left(\frac{-t_{ONP}}{T_1(0)}e^{\frac{-4\|\mathbf{r}\|}{a_o}}\right)\exp\left(-i2\pi t_1 f_0 e^{\frac{-2\|\mathbf{r}\|}{a_o}}\right)e^{\frac{-2\|\mathbf{r}\|}{a_o}}\,dr. \tag{3.25}$$

The first piece of this equation is just equation (3.22) while the second piece can be solved by the same u-substitution (the one in equation (3.23)), combining these results one obtains

$$S(u)\propto\log^2(u)\left(1-\exp\left(\frac{-t_{ONP}u^2}{T_1(0)}\right)\right). \tag{3.26}$$

Finally, the effect of repolarization of the nuclei during the detection period will be considered. When one detects for a time t_2, one integrates the signal obtained over this time frame. However, the detected nuclei are still being optically pumped during their detection, thereby losing their memory of the polarization state that they were in at the beginning of the detection period. Given that the fraction of nuclei that have repolarized after detection for a time t_2 is $1-\exp\left(-t_2u^2/T_1(0)\right)$, the fraction left in the polarization state present at the beginning of the detection period is $\exp\left(-t_2u^2/T_1(0)\right)$. Integrating this and combining it with equation (3.26) generates

$$S(u) \propto \log^2(u)\left(1-\exp\left(\frac{-t_{ONP}u^2}{T_1(0)}\right)\right)\frac{T_1(0)}{u^2}\left(1-\exp\left(\frac{-t_2u^2}{T_1(0)}\right)\right) \qquad (3.27)$$

This formula fits the experimental data for short ONP times ($t_{ONP} \lesssim 0.7\text{s}$) in both the t_1 and t_2 time domains, but is utterly incapable of producing the correct line shape for any longer ONP times[2]. It will see in chapter 5 that the addition of spin diffusion to these equations provides the capacity to produce the correct line shape. That is, the data is in an intermediate region where spin diffusion is dramatically changing the spatial profile of the nuclear polarization but still not producing totally uniform nuclear polarization. See chapter 5 for a detailed discussion of the fitting of experimental data of varying sorts experiments, and chapter 4 for a more thorough discussion of spin diffusion.

D. Electric-Field Induced Electric-Field-Gradients in GaAs

There are two sources of electric field gradient at each nuclear site in the vicinity of an electron trapped at a defect. The first of these is the gradient of the electric field of the net charge of the defect at that site. The net charge is the difference between the charge of the defect atom and the amount of electric charge that screens this charge (the amount of electronic density which is localized closer to the defect atom than the distance between the defect atom and the nuclear site). The second source was discovered by Bloembergen[3], and it results primarily from the changes in the hybridization of the chemical bonds between atoms in the crystal when a homogeneous electric field is applied. When the crystal lacks a center of inversion these deformations of the electronic structure of the crystal lead to nonzero electric field gradients at the nuclear sites. Thus, even the homogeneous component of the electric field around an occupied point defect will give rise to a quadrupole splitting of the nuclei in GaAs and most other semiconductors.

There is also a smaller homogeneous electric field effect due to the piezoelectricity of the crystal ($\sim 10^{-3}$ with respect to the hybridization effect in GaAs). This results from the positively charged atoms in the crystal moving in an opposite direction with respect to the negatively charged atoms in the crystal. This deforms the unit cells of the crystal, and, therefore, the crystal as a whole. Note that lack of an inversion center is not a requirement for piezoelectricity of the crystal.

1. The direct gradient of the electric field

First, let us consider the electric field gradient of a bare positive charge in a crystal. If the crystal is a semiconductor such as GaAs, this crystal defect is known as an unoccupied shallow donor. The electric potential in this case can be written as

$$V = -\frac{b}{r} \tag{3.28}$$

where b is a constant that depends on the electric properties of the material that is being studied. Computing all possible second partial derivatives of V will then generate the electrostatic field gradient tensor **V**. Starting with derivatives with respect to x, one obtains

$$\frac{\partial V}{\partial x} = -\frac{bx}{r^3}, \tag{3.29}$$

$$\frac{\partial^2 V}{\partial x^2} = -\frac{b\left(3x^2 - r^2\right)}{r^5} = -\frac{b}{r^3}\left(3\cos^2\gamma - 1\right) = -\frac{b}{r^3}\left(3\cos^2\phi\sin^2\theta - 1\right), \tag{3.30}$$

$$\frac{\partial^2 V}{\partial y \partial x} = -\frac{3bxy}{r^5} = -\frac{3b}{r^3}\cos\gamma\cos\beta = -\frac{3b}{2r^3}\left(\sin 2\phi\sin^2\theta\right), \tag{3.31}$$

and

$$\frac{\partial^2 V}{\partial z \partial x} = -\frac{3bxz}{r^5} = -\frac{3b}{r^3}\cos\gamma\cos\alpha = -\frac{3b}{2r^3}\left(\cos\phi\sin^2\theta\right) \tag{3.32}$$

where α, β, and γ are the angles between the direction of the electric field and the *z*, *y*, and *x* axes respectively for the axis system chosen, and θ and ϕ are the polar and azimuthal angles respectively for the axis system chosen.

Similar calculations lead us to write the entire electrostatic field gradient tensor as

$$\mathbf{V} = -\frac{b}{r^3}\begin{pmatrix} 3\cos^2\phi\sin^2\theta - 1 & \frac{3}{2}\sin 2\phi\sin^2\theta & \frac{3}{2}\cos\phi\sin^2\theta \\ \frac{3}{2}\sin 2\phi\sin^2\theta & 3\sin^2\phi\sin^2\theta - 1 & \frac{3}{2}\sin\phi\sin^2\theta \\ \frac{3}{2}\cos\phi\sin^2\theta & \frac{3}{2}\sin\phi\sin^2\theta & 3\cos^2\theta - 1 \end{pmatrix} \tag{3.33}$$

where the rows and columns are written in order of *x*, *y*, and *z* respectively. Note that, due to the radial symmetry of V, no choice of an axis system relative to the crystal has yet been forced upon us. The magnitude of the secular portion of the quadrupole interaction may be determined by computing the component of V along the vector describing the magnetic field, and multiplying this by the quantum of a single electric charge *e* and the nuclear quadrupole moment *Q*. The product of these two factors represents the energetic effect of an electric field gradient at the site of the nucleus. Moreover, that portion of the quadrupole interaction that commutes with the Zeeman Hamiltonian (i.e., that portion of *eQ***V** which is directed along the magnetic field), is approximately the only portion that affects the energy of the nuclei, given that the Zeeman interaction is much larger than the quadrupole interaction that arises in this way. This leads us to express the quadrupole splitting ω_q as

$$\omega_q = eQ\frac{\mathbf{B}\cdot\mathbf{V}\cdot\mathbf{B}}{\mathbf{B}\cdot\mathbf{B}}. \tag{3.34}$$

This must then be normalized by a spin dependent factor that accounts for the fact that Q has been defined in purely geometric terms. That is, it is defined by the extent to which the nucleus is ellipsoidally shaped, yet this does not account for the fact that this ellipsoid will precess at different angles and frequencies with respect to the magnetic field depending on the size of its intrinsic spin angular momentum. The appropriate spin dependent factor is $4I(2I-1)$ as given in Mehring[4]. This leads one to write

$$\mathcal{H}_{\mathrm{q}} = \frac{\omega_{\mathrm{q}}}{4I(2I-1)}\left(3I_z^2 - I(I+1)\right) \tag{3.35}$$

where the operator

$$\frac{3I_z^2 - I(I+1)}{4I(2I-1)} \tag{3.36}$$

is the normalized operator that expresses secular quadrupolar energy. If **B** is kept in a general form, described in terms of θ_{B} and ϕ_{B}, the following equation is produced

$$\omega_{\mathrm{q}} = eQ\left(\cos\phi_{\mathrm{B}}\sin\theta_{\mathrm{B}} \quad \sin\phi_{\mathrm{B}}\sin\theta_{\mathrm{B}} \quad \cos\theta_{\mathrm{B}}\right)\mathbf{V}\begin{pmatrix}\cos\phi_{\mathrm{B}}\sin\theta_{\mathrm{B}}\\ \sin\phi_{\mathrm{B}}\sin\theta_{\mathrm{B}}\\ \cos\theta_{\mathrm{B}}\end{pmatrix} \tag{3.37}$$

where ω_{q} is eQV_{zz} in the frame of the applied magnetic field. From this point on θ and ϕ measured from the direction of the electric field shall be refered to as θ_{E} and ϕ_{E}. Using this notation, and after some algebra, the following is obtained

$$\omega_{\mathrm{q}} = -\frac{eQb}{r^3}\left(3\left(\cos(\phi_{\mathrm{B}}-\phi_{\mathrm{E}})\sin\theta_{\mathrm{B}}\sin\theta_{\mathrm{E}} + \cos\theta_{\mathrm{B}}\cos\theta_{\mathrm{E}}\right)-1\right). \tag{3.38}$$

This simplifies further if one is willing define a new angle δ which is the angle between the magnetic field direction and the electric field direction. Using trigonometric identities one obtains

$$\omega_q = -\frac{eQb}{r^3}\left(3\cos^2\delta - 1\right). \tag{3.39}$$

This should be expected since it is what one would obtain if one introduced a magnetic field in the z-direction as defined by the axis system used to describe the electric field gradient tensor above. However, given the spherical symmetry of the system it is not surprising that any arbitrary direction of magnetic field will produce the same result relative to that magnetic field.

Numerical evaluation of the constant b in GaAs (in MKS units) yields

$$b = \frac{e}{4\pi\varepsilon_{o,GaAs}h} \cong 1.655\times10^{23} \tag{3.40}$$

and, therefore,

$$eQb \cong 2.970\times10^{-25} \tag{3.41}$$

where e is the charge of a single electron, and $\varepsilon_{o,GaAs}$ is the permeability of GaAs (~13.1 times the vacuum permeability). Moreover, Arsenic and both types of Gallium are spin-3/2 nuclei (I = 3/2). This yields the values in table 3.1 for the magnitude of ω_q as a function of distance (satellite to satellite frequency difference, not including angular factors). Note that the relative radius for a shallow donor with a 100 Angstrom Bohr

$z = r/a_o$	Amount of signal outside of this radius	$\|\omega_q\|$ (Hz)
0.1	99.9%	593.9
0.2	99.2%	74.24
0.3	97.7%	22.00
0.4	95.3%	9.28
0.5	92.0%	4.75
0.6	87.9%	2.75
0.8	78.3%	1.16
1.0	67.7%	0.59

Table 3.1 The quadrupole splitting arising from the direct field gradient of a bare charge.

radius and the relative amount of optically detected signal outside this radius (for the case of uniform nuclear polarization) are listed along side these values of ω_q.

This clearly demonstrates that, even with a 20 Hz linewidth, greater than 95% of the sites around an unoccupied shallow donor have a negligible quadrupole interaction due to the gradient of the electric field.

Now, let us consider the electric field gradient arising due to the spatial variation of the electronic charge distribution of an electron bound at a positively charged defect in the crystal. That is, a site which is positive in its unoccupied state, and neutral in its occupied state. This is referred to as an occupied shallow donor. As in the previous case, this system exhibits radial symmetry, so only the magnitude of the second derivative will be computed. The electric potential may be written as

$$V = -\frac{b\left(1 - \int_0^r \psi^2(r_1)\,dr_1\right)}{r} \tag{3.42}$$

where

$$\psi(r) = \frac{2}{\sqrt{a_o^3}} \exp(-z). \tag{3.43}$$

Computing V and taking its second derivative with respect to r yields

$$\omega_q = eQb \frac{2\exp(-2z)}{a_o^3 z^3}\left(1 + 2z + 2z^2 + 4z^4\right). \tag{3.44}$$

Evaluating this numerically, one obtains the values in table 3.2, assuming a 100Å Bohr radius.

This further illustrates that the gradient of the electric field around a singly occupied shallow donor so weak that roughly 98% of the detectable signal comes from sites that have a negligible quadrupole interaction, even with a 20 Hz linewidth. Indeed, even though the percent change in the quadrupole interaction of the occupied donor at a given radius with respect to the unoccupied donor is maximal at radii where a high fraction of the signal should originate, the quadrupole interaction itself has

$z = r/a_o$	Amount of signal outside of this radius	$\|\omega_q\|$ (Hz)
0.1	99.9%	593.5
0.2	99.2%	73.98
0.3	97.7%	21.88
0.4	95.3%	9.27
0.5	92.0%	4.81
0.6	87.9%	2.85
0.8	78.3%	1.29
1.0	67.7%	0.72

Table 3.2 The quadrupole splitting arising from the direct field gradient of an occupied shallow donor.

fallen off to negligible values at these radii.

Now, let us consider the electric field gradient arising due to the spatial variation of the electronic charge distribution of an electron bound at a neutral defect in the crystal. That is, a site that is neutral in its unoccupied state, and negatively charged when it is occupied by a free electron. Isoelectronic defects[5-10], for example, fall within this category. As in the previous cases, this system again exhibits radial symmetry, so only the magnitude of the second derivative will be computed. The electric potential may be written as

$$V = \frac{b\int_0^r \psi^2(r')\,dr'}{r} \tag{3.45}$$

where $\psi(r)$ depends on the defect potential. Assuming this is approximately a spherical box with a finite potential barrier, $\psi(r)$ can essentially be represented as

$$\psi(r) = \frac{4}{a_o^3\sqrt{\pi}}\exp\left(-z^2\right). \tag{3.46}$$

Computing V and taking its second derivative with respect to r gives rise to

$$\omega_q = eQb\frac{4}{a_o^3\sqrt{\pi}}\left(-2z^2\exp(-z^2) - \frac{2\exp(-z^2) - \sqrt{\pi}\ \mathrm{erf}(z)}{2z^3}\right) \tag{3.47}$$

where erf(x) is the standard error function evaluated at x. That is,

$$\mathrm{erf}(x) = \frac{1}{\sqrt{\pi}}\int_{-x}^{x}\exp\left(-y^2\right)dy. \tag{3.48}$$

Evaluating equation (3.47) numerically, one obtains the values in table 3.3 as a function of radius for the case where the Bohr radius is 100Å.

These values are clearly far too small to ever be of any experimental importance. Thus, the gradient of a spatially varying electric field around commonly occurring defects cannot give rise to a significant quadrupole interaction at the detectable sites around such defects.

$z = r/a_o$	Amount of signal outside of this radius	$\|\omega_q\|$ (Hz)
0.1	99.9%	0.431
0.2	99.4%	0.385
0.3	98.1%	0.313
0.5	92.0%	0.124
0.75	77.1%	-0.107
1.15	45.0%	-0.257
1.5	16.3%	-0.179
2.0	4.6%	-0.027

Table 3.3 The quadrupole splitting arising from the direct field gradient of a defect that is negatively charged when occupied by a free electron.

2. The electric field gradient induced by a homogeneous electric field

Bloembergen has showed that there is a linear stark effect in crystals that lack a center of inversion, which results in the appearance of a quadrupole splitting when a homogeneous electric field is applied[3]. That is, when a homogeneous electric field is applied to a crystal that lacks a center of inversion, it will induce electric field gradients at the nuclear sites. This is predominately due to mixing of the valence (bonding) electronic

orbitals of the atoms in the crystal. The electric field gradient tensor induced in the case of GaAs has been shown to be

$$\mathbf{V} = E\left(R_{14} + S_{44}d_{14}\right)\begin{pmatrix} 0 & \cos\theta_{\mathrm{E}} & \sin\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} \\ \cos\theta_{\mathrm{E}} & 0 & \cos\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} \\ \sin\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} & \cos\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} & 0 \end{pmatrix} \tag{3.49}$$

where the rows and columns are written in order of x, y, and z respectively. This equation is written with respect to the crystal axes since the interaction is due to the symmetry (or lack thereof) of the crystal. However, the crystal is often oriented differently with respect to the axes system of the NMR experiment. This greatly affects the NMR Hamiltonian that we write since the usual practice of writing the quadrupole Hamiltonian as

$$\mathcal{H}_{\mathrm{q}} = \frac{eQ}{4I(2I-1)}\mathbf{I}\cdot\mathbf{V}\cdot\mathbf{I} \tag{3.50}$$

is predicated on writing **I**, the vector whose x, y, and z components are the x, y, and z spin operators, in the same bases as **V**.

a. The case of a 001-oriented magnetic field

When the magnetic field (z-axis) is oriented along the 001-axis, and the x and y axes are oriented along the 100 and 010 axes respectively, the quadrupole Hamiltonian can be expressed as

$$\mathcal{H}_{q} = \frac{\omega_{q}}{4I(2I-1)}\begin{pmatrix} I_x & I_y & I_z \end{pmatrix}\begin{pmatrix} 0 & \cos\theta_{\mathrm{E}} & \sin\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} \\ \cos\theta_{\mathrm{E}} & 0 & \cos\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} \\ \sin\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} & \cos\phi_{\mathrm{E}}\sin\theta_{\mathrm{E}} & 0 \end{pmatrix}\begin{pmatrix} I_x \\ I_y \\ I_z \end{pmatrix} \tag{3.51}$$

where

$$\omega_q = eQE\left(R_{14} + S_{44}d_{14}\right). \tag{3.52}$$

In GaAs, the constants for the induced field evaluate to

$$R_{14} + S_{44}d_{14} = 2.863\times10^{12}. \tag{3.53}$$

Upon carrying out the matrix algebra in equation (3.51), one arrives at

$$\mathcal{H}_q = \frac{\omega_q}{4I(2I-1)}\left(\cos\theta_E\left[I_y, I_x\right]_+ + \sin\phi_E\sin\theta_E\left[I_z, I_x\right]_+ + \cos\phi_E\sin\theta_E\left[I_z, I_y\right]_+\right). \tag{3.54}$$

The secular portion of this is exactly zero since the electrostatic field gradient tensor is exactly off diagonal in the basis of the crystal axes with which the magnetic field has been aligned. To the extent that some small errors in alignment place the magnetic field off of the crystal axis, there will be a small secular component of the quadrupole Hamiltonian. For example, a 5° error will lead to a secular component which is of order one tenth the size of the off diagonal component (that is, approximately $\sin(5^\circ)$ in magnitude). It will be demonstrated below that this is worth considering in high-resolution spectroscopy, given the size of these nonsecular quadrupole interactions.

The magnitude of the induced gradient quadrupole Hamiltonian with this magnetic field orientation can be numerically evaluated for the case of a point charge in GaAs in exactly the same manner as has already been accomplished for the direct gradients. For the case of an unoccupied donor, a bare positive charge, one has

$$E = \frac{b}{r^2} \tag{3.55}$$

which gives

$$\omega_q = \frac{eQb}{r^2}\left(R_{14} + S_{44}d_{14}\right). \tag{3.56}$$

For the case of an occupied donor, the electric field becomes

$$E = \frac{b\exp(-2z)}{r^2}\left(1 + 2z + 2z^2\right) \tag{3.57}$$

which yields

$$\omega_q = \frac{eQb\left(R_{14} + S_{44}d_{14}\right)\exp(-2z)}{r^2}\left(1 + 2z + 2z^2\right). \tag{3.58}$$

Evaluating equations (3.56) and (3.58) numerically, the values of ω_q in table 3.4 are obtained as a function of radial distance from a donor with a 100Å Bohr Radius.

$z = r/a_o$	Amount of signal beyond z	unoccupied $\|\omega_q\|$ (Hz)	occupied $\|\omega_q\|$ (Hz)
0.04	~100%	5,694,000	5,694,000
0.1	99.9%	911,100	910,000
0.3	97.7%	101,200	98,890
0.5	92.0%	36,440	33,520
0.75	80.9%	16,200	13,100
1.0	67.7%	9,111	6,165
1.5	42.3%	4,049	1,714
2.0	23.8%	2,228	542

Table 3.4 The quadrupole splitting arising from the induced field gradient around an unoccupied shallow donor is shown next to those for an occupied shallow donor. The magnitude of this interaction is four orders of magnitude higher than the direct gradient of the electric field.

The most obvious difference between these values and those for the direct gradient is the far greater magnitude of the present effect. Indeed, one may safely ignore direct gradient effects when dealing with donors since such effects will necessarily be small with respect to induced gradient effects. That is, despite the fact that the induced gradient effects above are entirely nonsecular, they are so large with respect to our magnetic field (~3.2MHz) at some sites that they will still affect the energy of the nuclear spin at these sites more than the small secular effect from the direct electric field gradients. This is due somewhat to imperfections in the rotating frame approximation in the presence of nonsecular terms, but mostly to imperfections in the positioning of the

sample so that some portion of the above quadrupole interaction is secular. Another way to describe the relative size of the direct and induced gradient effects is to say that the ratio of the coefficients of the two effects and the scaling as one over radius squared for the induced gradient and one over radius cubed for the direct gradient dictate that the two effects are equal for some very small and physically insignificant distance (~10^{-12} M), and that the extra power of one over radius in the direct gradient reduces its size by a large factor compared to the induced gradient at any distance that might be of physical interest.

For the case of an occupied neutral defect such as an isoelectronic defect, it is seen that

$$E = \frac{b}{r^2}\left(\operatorname{erf}\left(\sqrt{2}z\right) - 2z\sqrt{\frac{2}{\pi}}\exp\left(-2z^2\right)\right) \tag{3.59}$$

if the electronic wavefunction is Gaussian, which produces

$$\omega_q = \frac{eQb\left(R_{14} + S_{44}d_{14}\right)}{r^2}\left(\operatorname{erf}\left(\sqrt{2}z\right) - 2z\sqrt{\frac{2}{\pi}}\exp\left(-2z^2\right)\right). \tag{3.60}$$

However, if the electron occupies an expontial wavefunction, the E-field will be

$$E = \frac{b\left(1-\exp\left(-2z\right)\left(1+2z+2z^2\right)\right)}{r^2}, \tag{3.61}$$

which will result in the following quadrupole interaction

$$\omega_q = \frac{eQb\left(R_{14} + S_{44}d_{14}\right)\left(1-\exp\left(-2z\right)\left(1+2z+2z^2\right)\right)}{r^2}. \tag{3.62}$$

Evaluating equations (3.60) and (3.62) numerically, the values of ω_q in table 3.5 are obtained as a function of radial distance from the defect (100Å Bohr Radius). Notice that the neutral defect has had negligible quadrupole interactions up to this point. First, it had

	exponential wavefunction		Gaussian wavefunction	
$z = r/a_o$	% of signal beyond z	$\|\omega_q\|$ (Hz)	% of signal beyond z	$\|\omega_q\|$ (Hz)
0.1	99.9%	979	99.8%	1,790
0.2	99.4%	1,690	98.4%	3,460
0.3	98.1%	2,190	94.8%	4,890
0.4	95.6%	2,530	88.7%	6,010
0.5	92.0%	2,740	80.1%	6,780
0.75	77.1%	2,900	52.2%	7,240
1.0	57.2%	2,760	26.1%	6,290
1.5	16.3%	2,180	2.9%	3,680
2.0	4.6%	1,620	0.11%	2,128
3.0	0.044%	888	0.000075%	947

Table 3.5 The quadrupole splitting arising from the induced field gradient around an occupied isoelectronic defect with either a Gaussian wavefunction or an exponential wavefunction. The magnitude of this interaction is orders of magnitude smaller than in the case of a shallow donor for sites close to the defect.

a tiny secular portion due to the gradient of the electric charge distribution, and now, it has a much larger nonsecular portion that is still insignificant with respect to our Zeeman field (or virtually anybody else's for that matter).

One may ask how it is possible that the Gaussian wavefunction can have consistently higher values of $\|\omega_q\|$ than the exponential wave function. The answer comes from what one defines as the Bohr radius. Here the Bohr radius is taken to be the point where the wavefunction has decayed to $1/e$ of its value at $z = 0$. Setting the $1/e$ point of the Gaussian wavefunction equal to that of the exponential wavefunction dictates that the Gaussian wavefunction has a much larger fraction of the electronic density is inside a given radius compared to an exponential wavefunction. This is demonstrated in figure 3.5 below, in which the two wavefunctions are plotted. In addition, figure 3.6 shows the values of $\|\omega_q\|$ on the previous page plotted, further illustrating that the quadrupole interactions in the two cases only reach equivalence at distances such that ~100% of the electronic density is located inside that radius, at which point the electric field takes on a simple $1/r^2$ functional dependence.

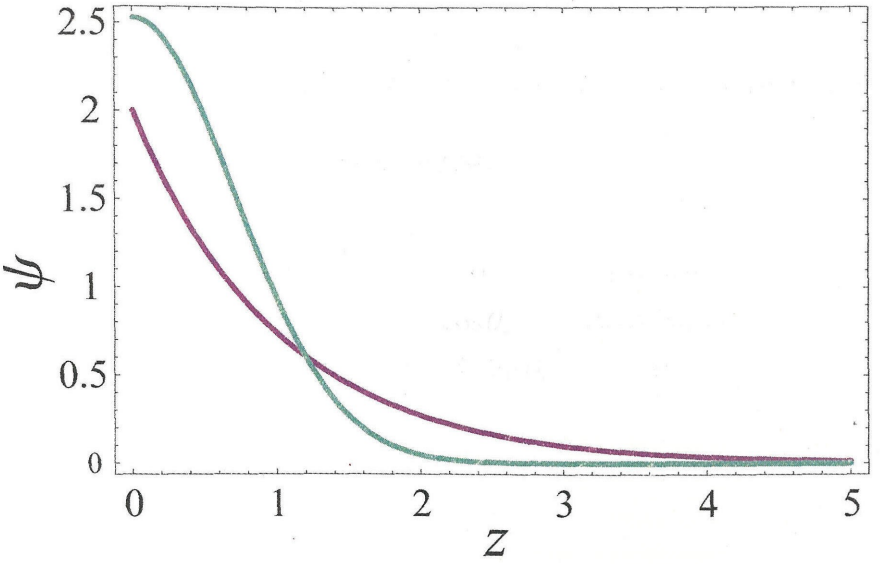


Figure 3.5 An electron in the Gaussian wavefunction (green line) is much more spatially localized than one in an exponential wavefunction (purple line) with the same Bohr radius (1/e point).

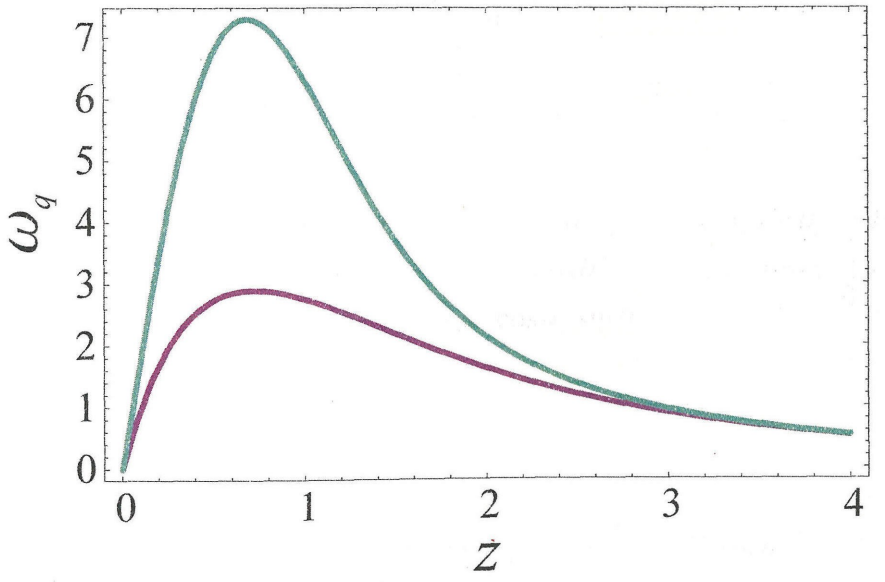


Figure 3.6 The greater spatial localization of the Gaussian wavefunction (green line) leads to a greater electric field induced quadrupole interaction at moderate values of z ($z \leq 3$) than the less spatially localized exponential wavefunction (purple line).

Finally, when the magnetic field (z-axis) is oriented along the 001-axis, and the x-axis and y-axis are oriented along the 110-axis and $\bar{1}10$-axis respectively, the electrostatic field gradient tensor can be expressed as

$$V = \omega_q \begin{pmatrix} -\cos\theta_E & 0 & \sin\delta_E \sin\theta_E \\ 0 & \cos\theta_E & \cos\delta_E \sin\theta_E \\ \sin\delta_E \sin\theta_E & \cos\delta_E \sin\theta_E & 0 \end{pmatrix} \tag{3.63}$$

where the angle δ_E is defined by

$$\cos\delta_E = \tfrac{1}{\sqrt{2}}\left(\sin\phi_E + \cos\phi_E\right). \tag{3.64}$$

This implies that

$$\delta_E = \phi_E + \frac{\pi}{4}. \tag{3.65}$$

The quadrupole Hamiltonian can be expressed as

$$\mathcal{H}_q = \frac{\omega_q}{4I(2I-1)} \begin{pmatrix} I_x & I_y & I_z \end{pmatrix} \begin{pmatrix} -\cos\theta_E & 0 & \sin\delta_E \sin\theta_E \\ 0 & \cos\theta_E & \cos\delta_E \sin\theta_E \\ \sin\delta_E \sin\theta_E & \cos\delta_E \sin\theta_E & 0 \end{pmatrix} \begin{pmatrix} I_x \\ I_y \\ I_z \end{pmatrix} \tag{3.66}$$

or more explicitly as

$$\mathcal{H}_q = \frac{\omega_q}{4I(2I-1)} \left(\cos\theta_E \left(I_y^2 - I_x^2 \right) + \sin\delta_E \sin\theta_E \left[I_z, I_x \right]_+ + \cos\delta_E \sin\theta_E \left[I_z, I_y \right]_+ \right). \tag{3.67}$$

The secular portion of this is still zero since the term in $I_y^2 - I_x^2$ is rigorously off diagonal in the eigenbasis of I_z. However, it is again true that to the extent that some small errors

in alignment place the magnetic field off of the crystal axis, there will be a small secular component of the quadrupole Hamiltonian. The magnitude of the quadrupole Hamiltonian with this magnetic field orientation is the same as that just previously considered with the x, y and z axis aligned along the crystal axes.

b. The case of a 110-oriented magnetic field.

When the magnetic field (z-axis) is oriented along the 110-axis, and the x-axis and y-axis are oriented along the $00\bar{1}$-axis and $\bar{1}10$-axis respectively, the electrostatic field gradient tensor can be expressed as

$$V = \omega_q \begin{pmatrix} 0 & \cos\delta_E \sin\theta_E & \sin\delta_E \sin\theta_E \\ \cos\delta_E \sin\theta_E & \cos\theta_E & 0 \\ \sin\delta_E \sin\theta_E & 0 & -\cos\theta_E \end{pmatrix}. \tag{3.68}$$

The quadrupole Hamiltonian can be expressed as

$$\mathcal{H}_q = \frac{\omega_q}{4I(2I-1)} \begin{pmatrix} I_x & I_y & I_z \end{pmatrix} \begin{pmatrix} 0 & \cos\delta_E \sin\theta_E & \sin\delta_E \sin\theta_E \\ \cos\delta_E \sin\theta_E & \cos\theta_E & 0 \\ \sin\delta_E \sin\theta_E & 0 & -\cos\theta_E \end{pmatrix} \begin{pmatrix} I_x \\ I_y \\ I_z \end{pmatrix} \tag{3.69}$$

or more explicitly as

$$\mathcal{H}_q = \frac{\omega_q}{4I(2I-1)} \left(\cos\theta_E \left(I_y^2 - I_z^2 \right) + \sin\phi_E \sin\theta_E \left[I_z, I_x \right]_+ + \cos\phi_E \sin\theta_E \left[I_x, I_y \right]_+ \right). \tag{3.70}$$

This can be rewritten as

$$\mathcal{H}_q = +\frac{\omega_q}{4I(2I-1)}\left(\tfrac{1}{2}\cos\theta_E\left(I_y^2 - I_x^2\right) + \sin\phi_E \sin\theta_E \left[I_z, I_x\right]_+ + \cos\phi_E \sin\theta_E \left[I_x, I_y\right]_+\right) - \frac{\omega_q}{8I(2I-1)}\cos\theta_E\left(3I_z^2 - I^2\right). \tag{3.71}$$

In this form, it is clear the magnitude of the secular portion of this Hamiltonian is

$$\mathcal{H}_q = -\tfrac{1}{2}\omega_q \cos\theta_E. \tag{3.72}$$

The maximum value of the secular portion of the quadrupole Hamiltonian at a given radius is just negative one half of the value calculated for the nonsecular induced gradient in a 001-oriented magnetic field. Moreover, in the case of a 001-oriented magnetic field, the magnitude of the nonsecular portion of the quadrupole Hamiltonian was independent of the orientation of the electric field despite the fact that the spin operators in that Hamiltonian where determined by this orientation. However, even though the total magnitude of the quadrupole Hamiltonian is still independent of the orientation of the electric field, now the quadrupole Hamiltonian has a secular portion which effects the NMR spectrum entirely differently than the nonsecular portion by its very definition. That is, those sites along the z-axis of the NMR experiment (those along the 011-axis of the crystal) will have this maximum value of the secular portion of the quadrupole Hamiltonian while those sites in the xy-plane of the NMR experiment will have no secular quadrupole Hamiltonian and a maximal nonsecular quadrupole Hamiltonian, yielding a strong effect on the ONMR signal originating around any of the types of defects considered thus far.

E. The optically pumped state, quadrupole relaxation and ONP

When optical pumping is conducted in the presence of these large quadrupole interactions the portion of the optically pumped state, which is putatively proportional to I_z, that does not commute with the eigenstate of the nucleus will precess about the direction of this eigenstate and eventually dephase away. Therefore, optical pumping will effectively accumulate nuclear polarization along the eigenstate of the nuclear spin essentially regardless of the actual state in which the optical pumping places the nuclear spins.

What is more, those portions of the quadrupole Hamiltonian that are off diagonal in the eigenbasis of the total Hamiltonian can lead to relaxation of the nuclear spin to a state of thermal equilibrium, which is nearly zero nuclear polarization for our experiments. In the case where the quadrupole Hamiltonian has no secular piece, the magnitude of sum of the squares of all such terms is approximately independent of the orientation of the nucleus with respect to the electric field. One may understand this by realizing that the magnitude of the nonsecular portion of the quadrupole Hamiltonian is normalized. That is, considering either equation (3.51) or equation (3.66) it is evident that

$$\mathcal{H}_q \cdot \mathcal{H}_q = \omega_q^2 . \tag{3.73}$$

Therefore, only the dependence of the magnitude of the electric field (radial dependence) is important in computations the relaxation caused by these quadrupole interactions.

The sum of the squares of the terms is the relevant quantity for a relaxation calculation since it has been assumed that each off diagonal term is equally efficient at relaxing the nuclear spins and since it is the square of the fluctuations in such terms that lead to relaxation. Moreover, unless there is an electron occupying the donor and its electronic wavefunction loses its spherical symmetry, the only fluctuations available to the system are in the magnitude of the radial electric field. This naturally leads to multiple hypotheses concerning the cause of such fluctuations.

One such hypothesis is that the electron is sometimes present at the defect and sometimes absent from the defect. This produces a fluctuation in the electric field that is the difference between the electric field of a bare charge in GaAs and the same bare charge occupied by an electron, yet this is obviously just the electric field due to the presence of the electron in the hydrogenic donor orbital. However, there is a problem with this situation. As the intensity of the light generating the electrons is turned up, the donor sites saturate to full occupancy. That is, the electron is virtually always present at the donor, and, therefore, the fluctuations should shrink in magnitude to a negligible level.

Another, more likely hypothesis is that the electron spends some amount of time in the excited states of the donor. There is reliable evidence that the 2s and 2p levels of the donor are bound states in the GaAs crystal. This excited state occupancy could be due to either electrons falling into these states from the conduction band before eventually falling down to the ground state of the donor, or electrons in the ground state of the donor atom that are excited to the higher states of the donor by collisions with other electrons

passing near the bound electron (other conduction electrons which have not yet been captured by a donor). In reality, both of these methods of excited state occupancy are possible. This mechanism leads to a source of fluctuations that does not vanish when the laser intensity is raised above the saturation limit of the shallow donor.

The magnitude of the fluctuations of the electric field is the weighted difference of the electric field when the electron is occupying the ground state and the electric field when the electron is occupying the excited state. The electric field around a donor whose ground state is occupied is

$$E_{\text{donor}_{1s}}(r) = E_{\text{bare charge}}(r) + E_{\text{electon}_{1s}}(r) \tag{3.74}$$

$$E_{\text{donor}_{1s}}(r) = \frac{b}{r^2} + \int_0^r \psi_{1s}^2(r_1)\,dr_1 \tag{3.75}$$

$$E_{\text{donor}_{1s}}(r) = \left(be^{-(2z)}\left(2z^2+2z+1\right)\right)\Big/\left(a_o z\right)^2 \tag{3.76}$$

where a_o is the Bohr radius of the donor ground state, and $z = r/a_o$. The electric field around a donor whose bound excited states are fully occupied is

$$E_{\text{donor}_{2s2p}}(r) = E_{\text{bare charge}}(r) + E_{\text{electon}_{2s}}(r) + E_{\text{electon}_{2p_x}}(r) + E_{\text{electon}_{2p_y}}(r) + E_{\text{electon}_{2p_z}}(r) \tag{3.77}$$

$$E_{\text{donor}_{2s2p}}(r) = \frac{b}{r^2} + \int_0^r \psi_{2s}^2(r_1)\,dr_1 + \int_0^r \psi_{2p_x}^2(r_1)\,dr_1 + \int_0^r \psi_{2p_y}^2(r_1)\,dr_1 + \int_0^r \psi_{2p_z}^2(r_1)\,dr_1 \tag{3.78}$$

$$E_{\text{donor}_{2s2p}}(r) = \left(be^{-z}\left(\tfrac{1}{12}z^4 + \tfrac{1}{12}z^3 + \tfrac{1}{2}z^2 + z + 1\right)\right)\Big/\left(a_o z\right)^2 \tag{3.79}$$

When computing the average electric field, the appropriate linear combination of these is

$$E_{\text{donor}_{\text{Ave}}}(r) = (1-f)E_{\text{donor}_{1s}}(r) + f\,E_{\text{donor}_{2s2p}}(r) \tag{3.80}$$

$$E_{\text{donor}_{\text{Ave}}}(z) = \left((1-f)be^{-2z}\left(2z^2+2z+1\right) + fbe^{-z}\left(\tfrac{1}{12}z^4+\tfrac{1}{12}z^3+\tfrac{1}{2}z^2+z+1\right)\right)\Big/(a_o z)^2 \tag{3.81}$$

where f is the probability of the electron being in one of the excited states. When computing the strength of the fluctuations between these two electric fields, the difference between the sum of them and the sum of their squares is taken. That is, we compute the variance of the electric field at each radial point from

$$\sigma^2 = \frac{\left(\Sigma x^2\right) - \left(\Sigma x\right)^2/N}{N}. \tag{3.82}$$

In the present situation this implies

$$\begin{aligned}\sigma^2_{E_{\text{donor}}}(z) = &\left(\left((1-f)E_{\text{donor}_{1s}}(z)\right)^2 + \left(fE_{\text{donor}_{2s2p}}(z)\right)^2\right) \\ &-\left((1-f)E_{\text{donor}_{1s}}(z) + f\,E_{\text{donor}_{2s2p}}(z)\right)^2\end{aligned} \tag{3.83}$$

where $N = (1-f)+f = 1$. Furthermore, upon insertion of the electric fields that we have calculated above into this equation, it becomes

$$\begin{aligned}\sigma^2_{E_{\text{donor}}}(z) = &\frac{\left(\left((1-f)be^{-2z}\left(2z^2+2z+1\right)\right)^2 + \left(fbe^{-z}\left(\tfrac{1}{12}z^4+\tfrac{1}{12}z^3+\tfrac{1}{2}z^2+z+1\right)\right)^2\right)}{(a_o z)^4} \\ &-\frac{\left((1-f)be^{-2z}\left(2z^2+2z+1\right) + fbe^{-z}\left(\tfrac{1}{12}z^4+\tfrac{1}{12}z^3+\tfrac{1}{2}z^2+z+1\right)\right)^2}{(a_o z)^4}.\end{aligned} \tag{3.84}$$

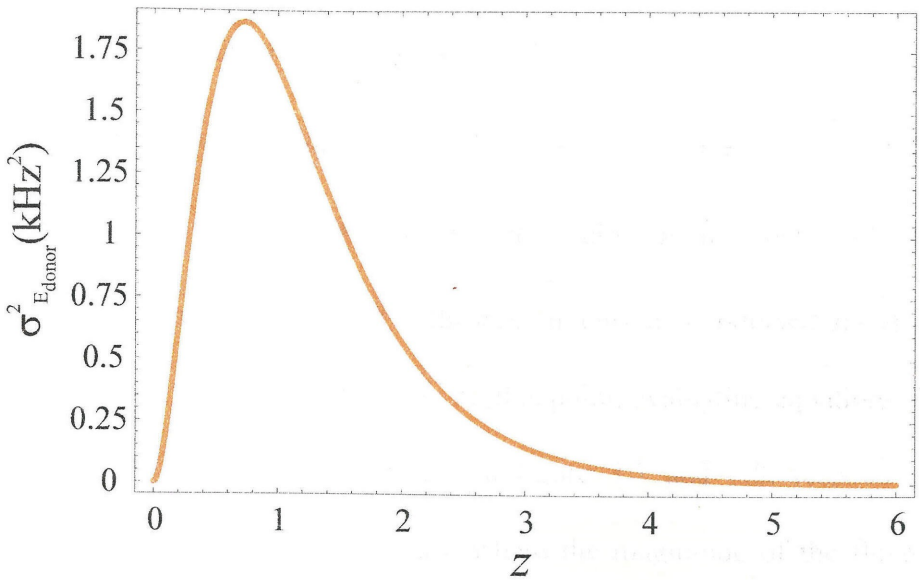


Figure 3.7 The magnitude of the fluctuations in the electric field as an electron that occupies the donor moves between the 1s, 2s and 2p orbitals of the donor.

Moreover, since the quadrupole interaction is linearly proportional to these electric fields, multiplication of $\sigma^2_{E_{donor}}(z)$ by the constants used in the previous section yields the variance of the magnitude of the quadrupole terms in the nuclear spin Hamiltonian as

$$\sigma^2_{\omega_q}(z) = eQ(R_{14} + S_{44}d_{14})\sigma^2_{E_{donor}}(z). \tag{3.85}$$

Finally, using (3.85) and the general form of a relaxation time as expressed in equation (3.3), one may write

$$\frac{1}{T_{1q}(z)} = \sigma^2_{\omega_q}(z)\frac{2\tau_c}{1+\omega^2\tau_c^2}. \tag{3.86}$$

If τ_c is taken to be the same order of magnitude as the residence time of a single electron at the donor ($\sim 10^{-8}$ s)[11], the very fastest possible $T_{1q}(z)$ relaxation times are

produced. Note that under any reasonable circumstances that the term $\omega^2\tau_c^2$ is approximately zero since even under this most optimistic assumption ($\tau_c \sim 10^{-8}$ s) it still yields only $\omega^2\tau_c^2 \simeq 0.04$ when calculated for ^{71}Ga. That is, in our experiments the highest ω is that of ^{71}Ga (the nucleus that most of the experiments are conducted upon), and ω for this nucleus is $2\pi \times 3.2\text{MHz} = 20\text{Mrad/s}$). At this point, evaluating equations (3.85) and (3.86) is straightforward, yet instructive. In figure 3.7 and 3.8, respectively, these equations are plotted for $f = 1/2$, the case where the magnitude of the fluctuations is maximized. Note that, while the values of T_{1q} thus obtained qualitatively match the time T_{1dark} in the light-off relaxation time experiments in chapter 5, there is essentially an additional source of decay of nuclear polarization. That is, spin diffusion carries spin polarization to ever-greater distances from the donor, where it is less optically detectable.

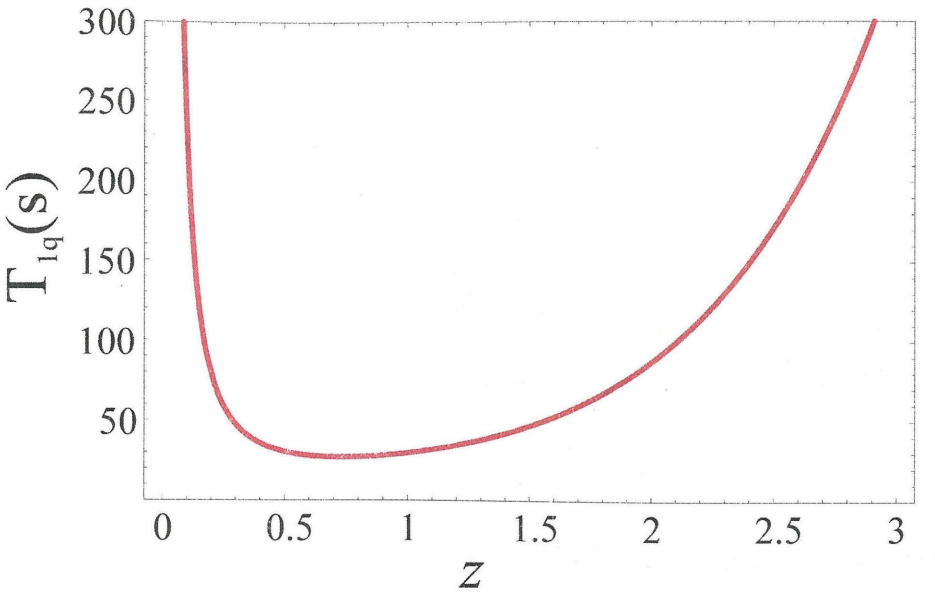


Figure 3.8 The relaxation time of the nuclei due to the fluctuations in electric field that are shown in figure 3.5. The minimum value of T_{1q} here is ~27s, and the optically detected average is obtained by integrating the OD weighting times the relaxation rate across $0 \le z \le \infty$, which yields ~76s.

Note that these very same oscillations in electric field are not without consequence to the ONP dynamics. The very same transitions into excited states of the hydrogenic donor will provide oscillations in the hyperfine coupling of the nuclei to these very same electrons. Applying the same procedure that was used with the electric field in equations (3.82) through (3.86) to the ψ^2 of the 1s, 2s and 2p states of the donor yields

$$\sigma^2_{\psi^2_{\text{donor}}}(z) = \left((1-f)\left(2e^{-z}\right)^2\right)^2 + \left(\frac{f}{2}\left(\frac{2-z}{2\sqrt{2}}e^{-z/2}\right)^2\right)^2 + \left(\frac{f}{2}\left(\frac{z}{2\sqrt{6}}e^{-z/2}\right)^2\right)^2 - \left((1-f)\left(2e^{-z}\right)^2 + \frac{f}{2}\left(\frac{2-z}{2\sqrt{2}}e^{-z/2}\right)^2 + \frac{f}{2}\left(\frac{z}{2\sqrt{6}}e^{-z/2}\right)^2\right)^2. \tag{3.87}$$

Equation (3.87) has been plotted in figure 3.9 versus radial distance z. It is interesting to note that this curve looks very much like the ψ^4 in figure 1.16 from chapter 1. Indeed, the ratio of these two is plotted in figure 3.10, proving the exact similarity of the functional form at low z. However, $z \geq 3$ the fluctuations represented in equation (3.87) start dropping much more slowly than the ψ^4 dependence that comes from fluctuation in the presence/absence of the electron in the 1s state.

Evaluating $T_{1_{on}}$ at $z = 0$ for a 100Å Bohr radius hydrogenic donor occupied by an electron with 25% spin polarization yields ~512ms for the 1s, 2s, and 2p fluctuations and ~113ms for the 1s electron presence/absence fluctuations (ψ^4), assuming that the time constant τ_c is approximately the same for both fluctuations with a value of $\sim 10^{-8}$.

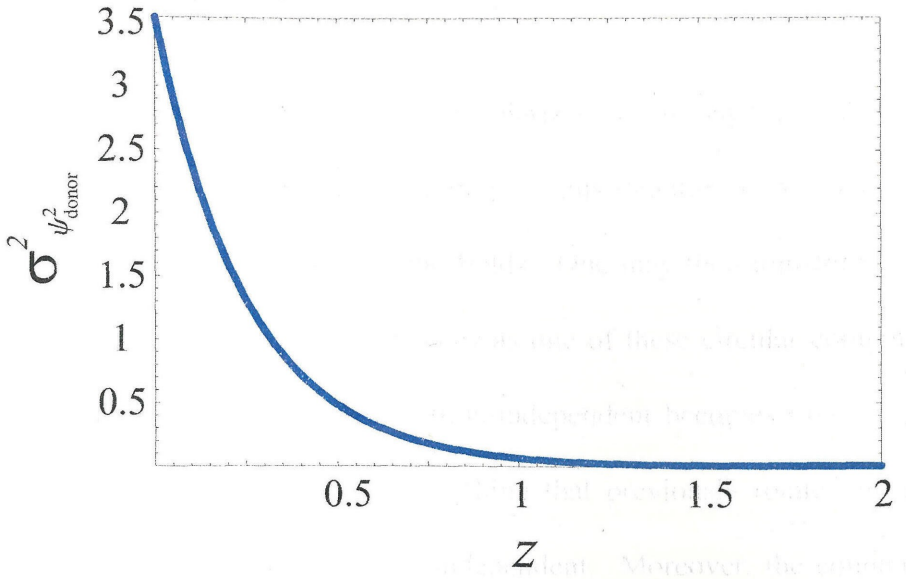


Figure 3.9 The spatial dependence of the fluctuations in the ψ^2 of the donor bound electron that lead to the nuclear spins achieving equilibrium with these very same spin polarized electrons.

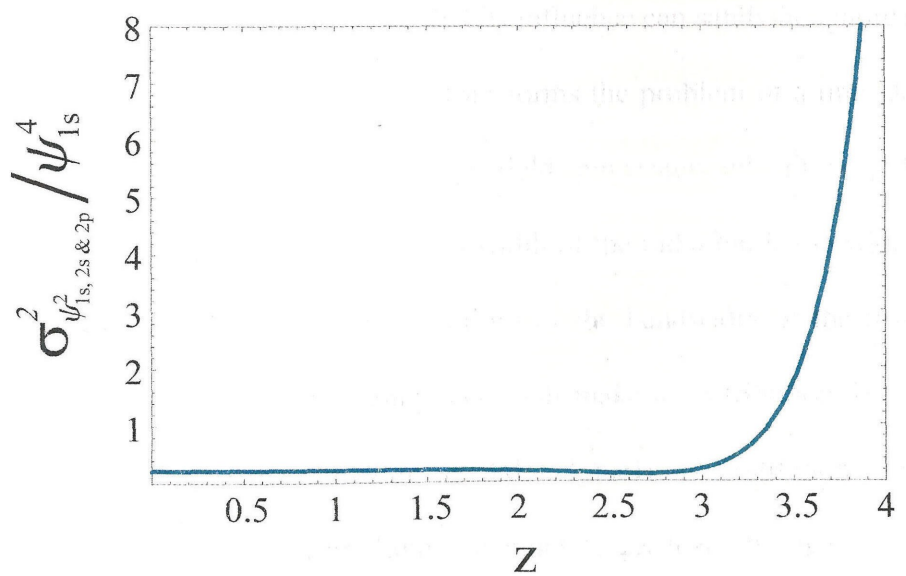


Figure 3.10 Dividing the fluctuations plotted in figure 3.9 by the fluctuations proportional to ψ^4 shows that the two have the same functional dependence, but are smaller than these ψ^4 fluctuations by a factor of ~3.5/16 until z~3.

F. The Rotating Wave Approximation

When considering the effect of linearly polarized electromagnetic radiation upon a given spin or set of spins, it is useful to decompose this radiation into a sum of counter rotating circularly polarized electromagnetic fields. One may then transform to a frame of reference that rotates at the same frequency as one of these circular components. In this frame everything that was previously time-independent becomes time-dependent at minus the frequency of rotation while everything that previously rotated in resonance with this circular component will be time-independent. Moreover, the counter rotating circular component appears to be rotating at exactly twice its original frequency when viewed from this new frame of reference; therefore, for frequencies that are high compared to the bandwidth of the radiation and resonances in question, the counter rotating component is so far off-resonance that its influence can safely be ignored. This is the famed rotating-wave approximation. It transforms the problem of a time-dependent, resonant (or nearly so) rf-field applied to a high-field spin system into a time-independent magnetic-field problem. However, if the bandwidth of the radiation is too wide (perhaps due to a relatively short pulse of such radiation) or the bandwidth of the resonance is extremely wide, the counter-rotating component will make a contribution to the physics of the system, making the approximation invalid. Finally the restriction to high-field spin systems, where the Zeeman Hamiltonian is much larger than all other Hamiltonians, is necessary since the transformation to the rotating frame is to be accomplished by rotation about the z-axis at the frequency of the applied rf-field, and this rotation must commute with the individual elements of the Hamiltonian (i.e., they must be truncated to

their secular portions). Thus, since ONMR in GaAs, as we practice it, combines relatively low Zeeman frequencies with potentially large, non-secular quadrupole Hamiltonians, and relatively short pulses (~10 cycles) of rf-radiation it is extremely unwise to make this assumption.

G. General Laboratory Frame Density Matrix Calculations

If one is not to make the rotating wave approximation, what path is available for such computations? The individual oscillations of the applied rf-field can be decomposed into short intervals of constant field just like a curve approximated by a series of slender rectangles in a Reimann sum. For these short intervals of time the rf-field is viewed as constant magnetic field in its instantaneous direction, changing in stepwise fashion to its next magnitude and/or direction for the next time interval. This allows us to solve for the quantum evolution of the system at each of these instantaneous values, and concatenate the individual evolution propagators to produce the evolution propagator for the entire period of time under investigation. This approach lacks aesthetic appeal and involves enormous numbers of complicated calculations, but it yields answers that cannot be obtained in any other manner.

Evolution propagators during rf-pulses may be computed by dividing the pulse into an integer number of cycles of the rf-wave and a remainder piece, computing one cycle of the rf with the correct laboratory-frame phase, exponentiating the resulting propagator by the integer number of cycles, computing the remainder piece, and, finally,

concatenating the remainder onto the end of the exponentiated propagator. Free evolution between the pulses is then calculated, and concatenated onto the previous pulse in the sequence. The next pulse is then computed such that it is phase coherent with the previous pulse, and the concatenation is con. Even if it rotating frame phase is the same, its laboratory frame phase will be different, so it must be recalculated. It is obviously to ones advantage to pursue any cyclic feature of experiments present due to the nature of the AWG board's generation of the rf-waves, such as the cyclic conditions on the CLSW-16 experiments that allow one cycle of the CLSW-16 sequence to be calculated and used repeatedly for the entire experiment.

Finally, because we have kept nonsecular quadrupole terms and carried out our computation in the laboratory frame, care must be taken that phenomena that would be filtered out of the signal in the actual experiment are not detected. Some good examples of this are frequency components at multiples of the Larmor frequency due to breaking the magnetic dipole selection rules and frequency components arising from the effect of nonsecular quadrupole terms upon the energy levels of nuclei close to the ORD. The computational intervention is relatively painless, but somewhat awkward and only approximately accurate. The detection operator that is used in the trace operation to detect the evolution of the density matrix can be edited as follows. It is placed into the eigenbasis of the free-evolution Hamiltonian, and evolution phase factors are computed for each of the matrix elements. Any term with a phase factor that falls outside of the detected frequency range is truncated to zero. The detection operator is then returned to

its original basis set, and is now insensitive to oscillation at any frequency that would not be detected in an actual experiment.

H. Conclusions

The individual components of a large-scale multi-site computation of single nucleus spin dynamics have been introduced and developed, and a representative computation is presented in appendix A of this thesis. Such isolated spin computations possess the unique capacity of addressing nonsecular phenomena such as the nonsecular terms in the quadrupolar Hamiltonian encountered in this chapter in an absolutely rigorous way. Unfortunately, it will be shown in chapters 4 and 5 that spin diffusion (multi-spin dynamics) is essential to describe the experiments presented in chapter 5 to the level of accuracy that motivated the rigorous laboratory-frame single-spin dynamics calculations to begin with. However, if the experiments are carried out in such a way as to defeat the interference of spin diffusion to begin with, the work presented in this chapter and the sample calculation in appendix A will be invaluable in simulating the results of those experiments.

I. References

1) Paget, D. *Physical Review B* **1982**, *25*, 4444.

2) Kempf, J. G. *Probing Quantum Confinement at the Atomic Scale with Optically Detected Nuclear Magnetic Resonance*; California Institute of Technology: Pasadena, 2001, pp 172.

3) Bloembergen, N. *Proc. Colloq. Ampere (Atomes Mol. Etudes Radio Elec.)* **1962**, *11*, 39-57.

4) Mehring, M. *High Resolution NMR in solids*; Second, Revised and Enlarged ed.; Springer-Verlag: Berlin, 1983.

5) Arfi, B.; Masselink, W. T.; Chang, Y. C. *Physical Review B* **1986**, *33*, 2401-2412.

6) Bergman, P.; Monomar, B.; Pistol, M. E. *Physical Review* **1989**, *40*, 12280-12289.

7) Di Ventra, M.; Mäder, K. A. *Physical Review B* **1997**, *55*, 13148-13154.

8) Watkins, S. P.; Thewalt, M. L. W. *Physical Review B* **1986**, *34*, 2598-2605.

9) Schranz, D. W.; Davison, S. G. *Int. J. Quantum Chem.* **1998**, *67*, 377-397.

10) Lozykowski, H. J. *Physical Review B* **1993**, *48*, 17758-17769.

11) Berkovits, V. L.; Ekimov, A. I.; Safarov, V. I. *Sov. Phys.-JETP* **1974**, *38*, 169.

IV. Spin Diffusion in a Spin-Three-Halves Spin System

We now address the theory of spin diffusion in detail. It will become clear as we do this that spin diffusion in GaAs proceeds at a previously unforeseen, extremely rapid rate. This is fully predicted by theory and is also now supported by experimental results that will be presented in chapter 5.

In section A, we provide a brief introduction to the dipolar Hamiltonian and discuss how it gives rise to spin diffusion. While reading this, it should be apparent that spin diffusion is simply "cross polarization" between nuclei of the same type. That is, cross polarization is radio frequency induced spin diffusion. This should help the reader who may be familiar with the concept of cross polarization in NMR, but is unaccustomed to considering dipolar couplings in solids.

We will consider those properties relating to dipolar couplings in solids that are calculable by summing over a static lattice of nuclei in section B. These properties are the necessary ingredients in calculations of spin dynamics, but they themselves are regarded as constants of the sample material. The homonuclear dipolar linewidth, the correlation time of homonuclear dipolar fluctuations, and T_{1I}, the time constant for the equilibration of polarization with the local neighborhood of spins, are introduced and calculated here.

Section C deals with the explicit calculation of spin diffusion from a microscopic perspective. This differs from most spin diffusion calculations in that the spin-three-

halves nature on the nuclei has been explicitly included. Novel equations for the polarization dynamics are developed both with and without simultaneous optical nuclear polarization.

The behavior of these same novel equations of motion for the nuclear polarization are explored and further developed in section D. Moreover, in this section, we will prove the common hypothesis that dipolar flip-flops act to drive the spin system to a state described by a single spin temperature, and do so rigorously, within the current theoretical constructs (a ladder of states exchanging population though a set of equations analogous to those used in reaction-kinetics).

These equations are further expanded in section E to include the case of ONP during spin diffusion about an ORD. In chapter 5, a simplified version of these results is compared to experiments that can now be understood in terms of rapid spin diffusion in a spin-three-halves spin system.

The modification of spin diffusion by the presence of a well-resolved quadrupole splitting is addressed in section F, and the behavior of the equations thus derived is explored in section G. This includes the nullification of the spin temperature hypothesis under select circumstances. Finally, an approximate inclusion of frozen core effects into the spin diffusion processes discussed in the earlier sections is developed in section H.

A. The Dipolar Hamiltonian

Two spins, I and S, separated by some distance, are each subject to the magnetic field associated with the magnetic moment of the other spin. This leads to an energetically preferential alignment of the two spins. Furthermore, if both spins are present in the same homogeneous external magnetic field, the Hamiltonian describing their coupling (in units of Hz) is given as follows[1], using MKS units.

$$\mathcal{H}_{\mathrm{d}} = \frac{\mu_{\mathrm{o}}\gamma_I\gamma_S h}{4\pi r_{IS}^3}\left(\mathbf{I}\cdot\mathbf{S} - 3\frac{(\mathbf{I}\cdot r_{IS})(\mathbf{S}\cdot r_{IS})}{r_{IS}^2}\right) \tag{4.1}$$

$$\mathcal{H}_{\mathrm{d}} = \frac{\mu_{\mathrm{o}}\gamma_I\gamma_S h}{4\pi r_{IS}^3}\left(A+B+C+D+E+F\right) \tag{4.2}$$

$$\begin{aligned}
A &= \mathbf{I}_z\mathbf{S}_z\left(1-3\cos^2\theta\right)\\
B &= \tfrac{1}{2}\left(\mathbf{I}_z\mathbf{S}_z - \mathbf{I}\cdot\mathbf{S}\right)\left(1-3\cos^2\theta\right)\\
C &= -\tfrac{3}{2}\left(\mathbf{I}_z\mathbf{S}_+ + \mathbf{I}_+\mathbf{S}_z\right)\sin\theta\cos\theta e^{-i\phi}\\
D &= -\tfrac{3}{2}\left(\mathbf{I}_z\mathbf{S}_- + \mathbf{I}_-\mathbf{S}_z\right)\sin\theta\cos\theta e^{i\phi}\\
E &= -\tfrac{3}{4}\mathbf{I}_+\mathbf{S}_+\sin^2\theta e^{-2i\phi}\\
F &= -\tfrac{3}{4}\mathbf{I}_-\mathbf{S}_-\sin^2\theta e^{2i\phi}
\end{aligned} \tag{4.3}$$

Where r_{IS} is the distance between the two spins, θ and ϕ are the polar angles that the vector joining the two spins makes with the magnetic field, and the $\mathbf{I}_\alpha$ and $\mathbf{S}_\alpha$ are the operators for the various components of spin angular momentum on spins I and S.

This particular factoring of terms is referred to as the dipolar alphabet. It is readily apparent that the A and B terms of the dipolar alphabet involve only diagonal operators and zero quantum operators. Therefore, the A and B terms commute with total

spin angular momentum operator $\mathbf{F}_z = \mathbf{I}_z + \mathbf{S}_z$, and, as such, they form the secular portion of the dipolar Hamiltonian. The C and D terms couple states that differ by one unit of spin-angular-momentum, and the E and F terms couple states that differ by two units of spin-angular-momentum. These terms are truncated by large applied magnetic fields since they represent small off-diagonal terms coupling states separated by large Zeeman energy differences. In other words, in the rotating frame these terms become time dependent at either plus or minus the Larmor frequency (terms C and D, respectively) or at plus or minus twice the Larmor frequency (terms E and F, respectively).

The B terms of the dipolar Hamiltonian conserve the total spin angular momentum. That is, they do not mix states with different amounts of total spin angular momentum. They do, however, mix states that have the same amount of total spin angular momentum (different many particle states that have the same numbers of spins up versus down). In terms of spin operators on two spins I and S these terms may be written as $\nu_d\left(I_+S_- + I_-S_+\right)$ where $\nu_d = \frac{\mu_o\gamma_I\gamma_S h}{4\pi r_{IS}^{\ 3}}\left(1-3\cos^2\theta_{IS}\right)$. These "flip-flop" terms correspond to a simultaneous but opposite spin flip on each of the nuclei when they are each in a state of definite angular momentum (that is, (+,–) → (–,+)). Indeed, if the spins start in the state (+,–), the flip-flop terms in the dipolar Hamiltonian will cause them to precess from this state into the state (–,+) in period $1/\nu_d$, and, then, cause them to precess back to the state (+,–) also in period $1/\nu_d$. Thus, the round trip will be characterized by oscillation at the frequency $\nu_d/2$.

This situation is formally analogous to the precession of a spin about a transverse field. The transition probability per unit time, W, in that case can, therefore, be used here. It is given at short times ($Wt << 1$) by the following perturbation expression [2].

$$W_{m,m-1} = \pi^2 \nu_d^{\,2} (I+m)(I-m+1) g(\nu) \tag{4.4}$$

The short time approximation used in the perturbation treatment is applicable in all of the following work since the time scale over which we will concern ourselves is very short compared to the inverse of the dipolar couplings. The function $g(\nu)$ is the resonance line shape of the individual nuclear spin transitions. It is well approximated by the homogeneous line-shape of the sample, which is nearly gaussian. Furthermore, we understand that, since equation (4.4) was drawn from the analogy of these transitions with the transitions due to a transverse magnetic field, it leaves out a factor of $(I+m')(I-m'+1)$ on the other nucleus (the one that is simultaneously flip-flopping with the nucleus we are calculating), leading us to write

$$W_{m,m-1} = \pi^2 \nu_d^{\,2} (I+m)(I-m+1)(I+m')(I-m'+1) g(\nu) \tag{4.5}$$

Finally, we can get a feel for what time and length scales we should expect to be pertinent to calculations in GaAs by evaluating W for each of the separate transitions in ^{71}Ga. In a 001-oriented magnetic field the largest coupling to a nearest neighbor is to the four spins in the plane perpendicular to the magnetic field (in the 110, $1\bar{1}0$, $\bar{1}10$, and $\bar{1}\bar{1}0$ directions). These sites are located at a distance of almost exactly 4Å. Using the calculated homogeneous linewidth from section B below, the

gyromagnetic ratio of ^{71}Ga (the largest of the available isotopes), and equation (4.5), we compute W for the different ^{71}Ga transitions as

$$\begin{aligned} W_{m,m-1} &\cong \pi^2 (175\text{Hz})^2 \frac{1}{(1239\text{Hz})\sqrt{2\pi}} (I+m)(I-m+1)(I+m')(I-m'+1) \\ &\cong 97.3(I+m)(I-m+1)(I+m')(I-m'+1)\ \text{s}^{-1} \end{aligned} \tag{4.6}$$

Where the spin dependent factor $(I+m)(I-m+1)(I+m')(I-m'+1)$ evaluates to sixteen if both transitions are between the $\frac{1}{2}$ and $-\frac{1}{2}$ levels, to nine if they are both between the $\pm\frac{3}{2}$ and $\pm\frac{1}{2}$ levels, and to twelve if the transitions are mixed (one of each type). We have left this factor explicitly unevaluated in equation (4.6) since it will become separated from the part we did evaluate in later equations.

B. Static Properties - Theory and Calculations

A number of properties that are either important in calculating spin diffusion or that provide insight into the general speed of polarization equilibration in GaAs crystals will be reviewed and calculated here.

1. The Dipolar Linewidth

The second moment of the resonance line due to the dipolar Hamiltonian is just the norm of the commutator of the secular portion of the dipolar Hamiltonian (the A and B terms) with the initial condition which we shall take to be $\mathbf{I}_x$. For two spins of the same nuclear species we compute this as.

$$M_2 = \left\langle \Delta\omega^2 \right\rangle = Tr\left([\mathcal{H}_A + \mathcal{H}_B, \mathbf{I}_x]^2 \right)$$

$$[\mathcal{H}_A + \mathcal{H}_B, I_x] = \left[\sum_{i \neq j} A_{ij} + B_{ij}, \sum_k I_{x_k} \right]$$

Evaluation of these equations is straightforward but cumbersome. Fortunately, the commutator portion has been solved[2], leaving only the summation over the lattice of nuclei. This is the well-known Van Vleck formula[2], (expressed here in MKS units)

$$\left\langle \Delta\omega^2 \right\rangle = \frac{3}{16\pi} \mu_o \gamma^4 h^2 I(I+1) \sum_{i<j} \frac{\left(1 - 3\cos^2\theta\right)}{r_{ij}^6}.$$

This homonuclear lattice sum has been performed for all three isotopes in GaAs in each of three orientations of the static magnetic field, and these results are presented table 4.1 below.

B_Z	$\left\langle \Delta\omega^2 \right\rangle_{^{71}Ga}$	$\left\langle \Delta\omega^2 \right\rangle_{^{69}Ga}$	$\left\langle \Delta\omega^2 \right\rangle_{^{75}As}$
[001]	285000	173000	74000
[110]	380000	242000	106000
[111]	459000	285000	116000

Table 4.1 The squares of the homonuclear dipolar linewidths of the three isotopes that are found in abundance in GaAs are displayed here for three different magnetic field orientations. Diffusion of spin order through the thermal reservoir of a given isotope is directly proportional to this quantity.

2. Calculation of τ_c and T_{II}

These two time scales are not used in the calculation of the spin diffusion below. They were originally computed as an intermediate result in a calculation of cross polarization efficiencies reported in a research proposition. However, they yield important insights into spin diffusion processes since they allow us a secondary and somewhat independent verification of the fast spin diffusion calculated below and observed in our experiments.

The time T_{II} is the time scale over which excess polarization is transferred from a single spin I in the sample to the other I spins in its local area, that is the time scale over which the I spin system will reach a local single spin temperature. It is simply the time scale T_{IS}, the time scale over which the I spin system comes to local equilibrium with a lone S spin, computed for the case where an I spin replaces the S spin. This time scale tells us how fast we should expect polarization to equilibrate over an area of several unit cells in radius. It is obviously related to spin diffusion, but its calculation includes concepts of the spectral density of dipolar fluctuations and correlations between the dipolar couplings of the lone spin to its neighbors and the dipolar couplings of those neighbors. As such, T_{II} is related not only to how likely a polarization transfer is, but how likely it is that a polarization transfer will happen and the polarization will never come back. That is, it has an additional element of irreversibility explicitly included in it that is absent in all of the further work in this chapter.

The time scale τ_c is simply the correlation time of the spectral density of the dipolar fluctuations. The relationship between T_{II}, τ_c, and $\langle \Delta\omega^2 \rangle$ is simply[3]

$$\frac{1}{T_{II}} = \frac{\pi}{2} \langle \Delta\omega^2 \rangle \tau_c .$$

Furthermore τ_c is readily calculable if we assume a Lorentzian functional form for the spectral density[1].

$$\frac{1}{\tau_c^2} = \frac{1}{2} N_2 = \frac{2}{3} I(I+1) \frac{\sum_{i<j} A_{ij}^2 \left(B_i - B_j \right)^2}{\sum_i B_i^2}$$

where

$$B_i = \frac{\mu_o \gamma_I \gamma_S h}{4\pi r_i^3} \left(1 - 3\cos^2 \theta_i \right)$$

$$A_{ij} = \frac{\mu_o \gamma_I^2 h}{8\pi r_{ij}^3} \left(1 - 3\cos^2 \theta_{ij} \right) .$$

Evaluation of these formulas over the lattice of spins in GaAs yields $\tau_c \approx 10.6$ ms for 71Ga and $T_{II} \approx 2.2$ μs for all three different nuclear species in GaAs. The identical values of T_{II} are not a coincidence. The factors of gyromagnetic ratio and isotopic abundance which make $\langle \Delta\omega^2 \rangle$ and τ_c different for each of the nuclear species exactly cancel in the computation of T_{II}.

We might at first think that there is some discrepancy between the value of $1/T_{II}$ and the value of W computed previously. However, we must keep in mind that the W we calculated is a single flip-flop transition rate for a nearest neighbor spin, and it is the sum of all these rates that is acting together (and in concert with the rates these neighboring spins have to yet other spins, and that those spins have to still others and so on) that leads to the fast polarization sharing present in GaAs. That is, if we imagine each site in the computation as the origin of an fcc crystal, then a full computation would include couplings not to just the 62 sites just within each of the eight unit cells that share that one site, but to hundreds of sites that also have a network of couplings to each other and to hundreds of even more distant sites.

The extremely short time scales that we calculate here are in general agreement with the fast transition rates for individual flip-flops calculated below. This means that spin diffusion is faster than almost any other process in GaAs, including ONP and any type of spin relaxation via quadrupolar effects.

C. Spin Diffusion - Theory

Spin diffusion in a system of spin-three-halves nuclei will be developed based on a straightforward extension of the theory for spin diffusion in a spin-one-halves system as presented in Abragam[2]. As such, this computation will be reviewed, and the equations relevant to a spin-three-halves system will then be developed for various situations.

1. The Spin-One-Half Polarization Diffusion Equation

We shall first concern ourselves with the case of a line of spin-one-half nuclei that have dipolar couplings to their nearest neighbors only. This is a bridge to a purely microscopic understanding of spin diffusion in a system of spin-three-halves nuclei since it begins with the sort of microscopic equations we will use shortly, and, then, takes the results of these into a continuous medium approximation.

If we define $p_-(x)$ as the probability that the spin at position x is in the minus one half state and $p_+(x)$ as the probability that the spin at position x is in the plus one half state, we see that the equations for the time dependence of these probabilities under the influence of the flip-flop terms are[2]

$$\frac{1}{W}\frac{\partial p_-(x)}{\partial t} = p_+(x)\big(p_-(x-a)+p_-(x+a)\big) - p_-(x)\big(p_+(x-a)+p_+(x+a)\big) \quad (4.7)$$

and

$$\frac{1}{W}\frac{\partial p_+(x)}{\partial t} = p_-(x)\big(p_+(x-a)+p_+(x+a)\big) - p_+(x)\big(p_-(x-a)+p_-(x+a)\big). \quad (4.8)$$

However, these probabilities are bound by the relationship $p_-(x)+p_+(x)=1$, so there is really only one free variable between these two quantities. One can express all of the dynamics by following only one of these variables. However, the polarization, P, the difference between them, is the most enlightening of the choices of variables since it represents both of the individual populations within one variable. Taking the difference between the above equations and simplifying, we arrive at

$$\frac{1}{W}\frac{\partial P(x)}{\partial t} = P(x-a) + P(x+a) - 2P(x). \tag{4.9}$$

Moreover, when we take the difference between the above equations, the terms from equation (4.7) that are quadratic in populations are precisely canceled by equal and opposite terms in equation (4.8) so that equation (4.9) is exact.

Therefore, to the extent that any spin is more polarized than its neighbors, its polarization will decrease while the neighbors polarization will increase, and to the extent that any spin is less polarized than its neighbors, its polarization will increase at the expense of the polarization of its neighbors.

If we view the crystal lattice points as fine grid points like those used in a numerical evaluation of a continuous polarization variable, we see that the above equation is an approximate form of a one-dimensional diffusion equation.

$$\frac{\partial P(x,t)}{\partial t} = Wa^2 \frac{\partial^2 P(x,t)}{\partial^2 x} \tag{4.10}$$

Extending the line of spins to a rectangular lattice of spins we see that we are left with one such equation in each dimension, so that the total rate of change of polarization with respect to time is the sum of these equations.

$$\frac{\partial P(\mathbf{r},t)}{\partial t} = Wa^2\left(\frac{\partial^2 P}{\partial^2 x} + \frac{\partial^2 P}{\partial^2 y} + \frac{\partial^2 P}{\partial^2 z}\right) = D\nabla^2 P(\mathbf{r},t) \tag{4.11}$$

Where the diffusion constant D is equal to Wa^2. This equation allows us to use a continuum model to approximate the many body effects of a discrete lattice of spins. This approximation is limited by the fact that we have considered only nearest neighbor

couplings. One may try to sum all of the couplings in a given direction, but this will underestimate the spin diffusion rate since it amounts to decreasing a to the nearest neighbor distance for all transitions. That is, diffusion is r^2 times faster (a^2 in equation (4.11)) for spins further away than we have allowed. This is due to the increase in efficiency in the polarization jumping directly to a given spin versus diffusing there over the intermediate spins. This can only be fixed by computing spin diffusion over all spins in the lattice, giving up our continuous medium advantage. This raises two more issues. Firstly, the diffusion in a real lattice will have anisotropy due solely to the grid of atoms if it is not a simple cubic lattice. Secondly, there will be spatial anisotropy in the spin diffusion due to the angular dependence of the dipolar coupling as expressed in equations (4.1) through (4.3), which in turn leads to anisotropy in W and, ultimately, to anisotropy in D.

Any realistic computation of spin diffusion will require an explicit summation over a large body of sites with respect to each and every site at each and every time step. This is an onerously high level of work given the extremely short time increments demanded by the high speed of the spin diffusion process, the massive number of sites with which we concern ourselves in the application of a such calculation to ONMR at shallow defect sites in GaAs, and large spatial extent of the area of nonzero couplings. Indeed, the problem will only increase in complexity as we add the very real concern of the spin-three-halves nature of the nuclei in GaAs to the present equations. However, the continued progression of Moore's Law indicates that this exact level of calculation will eventually become computationally tractable, allowing routine investigation of these

phenomena. The computational work included in this thesis was completed on 300-500MHz Pentium based computers with 64-96Mb of RAM. The top of the line Pentium and Athlon computers are currently at ~1.5GHz with ~256-512 of high speed RAM. If Moore's Law holds (that is, if continuing to throw exponentially more resources at doubling the speed of processor approximately every 18-24 months does not hit limiting forces), then in 5-10 years there should be computers with enough RAM available to any program that needs it to hold the entire computation in memory (~3Gb), eliminating disk caching of the data which is the single highest impediment to actual computational speed as processor speed continues to accelerate.

2. Microscopic Rate Equations and a Polarization Basis Set for a Spin-Three-Halves

Now, let us work through the case where the single quantum flip-flops of a spin-three-halves nucleus assist in the spatial equilibration of the polarization. We will only consider one pair of spins at a time since the effects of a second nucleus are additive as above.

$$\begin{aligned}\frac{1}{W}\frac{\partial p_{\frac{3}{2}}(x)}{\partial t} &= 3p_{\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a)) \\ &\quad -3p_{\frac{3}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\end{aligned} \tag{4.12}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial p_{\frac{1}{2}}(x)}{\partial t} &= 3p_{\frac{3}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &-3p_{\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &+4p_{-\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &-4p_{\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\end{aligned} \tag{4.13}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial p_{-\frac{1}{2}}(x)}{\partial t} &= 3p_{-\frac{3}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &-3p_{-\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &+4p_{\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &-4p_{-\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\end{aligned} \tag{4.14}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial p_{-\frac{3}{2}}(x)}{\partial t} &= 3p_{-\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &-3p_{-\frac{3}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\end{aligned} \tag{4.15}$$

Note that these equations can be transformed by introducing the following orthogonal basis for the spin-polarization. The Zeeman polarization (i.e., the magnetic dipole polarization of the nucleus) at a given position is defined as follows (I = 3/2 in all formulas here).

$$P_Z(x) = p_{\frac{3}{2}}(x)+\tfrac{1}{3}p_{\frac{1}{2}}(x)-\tfrac{1}{3}p_{-\frac{1}{2}}(x)-p_{-\frac{3}{2}}(x)) = \frac{\langle I_z\rangle}{I} \tag{4.16}$$

We can visualize Zeeman polarization as follows. When we conduct a standard NMR experiment, the Hamiltonian is proportional to $\mathbf{I}_z$, and, thus, the high temperature approximation to the density matrix yields

$$\rho = \frac{Exp(-\hbar\omega I_z/kT)}{Tr(Exp(-\hbar\omega I_z/kT))} \approx \frac{\mathbb{I}-\hbar\omega I_z/kT}{Tr(Exp(-\hbar\omega I_z/kT))} \propto \mathbb{I}-\hbar\omega I_z/kT\,. \tag{4.17}$$

We immediately see that this polarization is characterized by population differences, which precisely reflect the equidistant energy separations of the Zeeman Hamiltonian.

$$p_{\frac{3}{2}} - p_{\frac{1}{2}} \approx p_{\frac{1}{2}} - p_{-\frac{1}{2}} \approx p_{-\frac{1}{2}} - p_{-\frac{3}{2}} \propto \hbar\omega/kT \tag{4.18}$$

Indeed, it is clear that any polarization obtained from a high temperature approximation will precisely reflect the energy differences found in the dominant term of the Hamiltonian.

The magnetic quadrupole polarization of the nucleus at a given position $P_Q(x)$ and the magnetic octapole polarization of the nucleus at a given position $P_O(x)$ are defined as

$$P_Q(x) = p_{\frac{3}{2}}(x) - p_{\frac{1}{2}}(x) - p_{-\frac{1}{2}}(x) + p_{-\frac{3}{2}}(x) = \frac{\left\langle 3I_z^{\,2} - \mathbf{I}\cdot\mathbf{I} \right\rangle}{I(2I-1)} \tag{4.19}$$

$$P_O(x) = \tfrac{1}{3}p_{\frac{3}{2}}(x) - p_{\frac{1}{2}}(x) + p_{-\frac{1}{2}}(x) - \tfrac{1}{3}p_{-\frac{3}{2}}(x)) = \frac{\left\langle Tr(I_z^{\,2})I_z^{\,3} - Tr(I_z^{\,4})I_z \right\rangle}{9/2} \tag{4.20}$$

As a pedagogical point, the normalization (9/2) on the lower right side of equation (4.20) should be written as a function of *I*, but this function is complicated since the maximum matrix element that we a trying to normalize moves on the diagonal as a function of *I* rather than remaining stationary with respect to either the edge or the middle of the matrix. It is, therefore, expressed numerically here for the case of a spin-three-halves.

The quadrupolar polarization corresponds to the experimentally observed polarization in experiments, such as NQR, were the Hamiltonian is proportional to the

quadrupole operator ($3\mathbf{I}_z^2 - \mathbf{I}\cdot\mathbf{I}$). Likewise, the octapolar polarization is the polarization that would result if a nuclear octapole moment existed, and the interactions of this moment were the dominant term in the nuclear spin Hamiltonian. It useful to note, that these equations can be inverted to give the level populations as a function of the various kinds of nuclear spin polarization.

$$p_{\frac{3}{2}} = \tfrac{3}{20}\left(3P_Z + P_O\right) + \tfrac{1}{4}\left(P_Q + \mathbb{I}\right) \tag{4.21}$$

$$p_{\frac{1}{2}} = \tfrac{3}{20}\left(P_Z - 3P_O\right) - \tfrac{1}{4}\left(P_Q - \mathbb{I}\right) \tag{4.22}$$

$$p_{-\frac{1}{2}} = -\tfrac{3}{20}\left(P_Z - 3P_O\right) - \tfrac{1}{4}\left(P_Q - \mathbb{I}\right) \tag{4.23}$$

$$p_{-\frac{3}{2}} = -\tfrac{3}{20}\left(3P_Z + P_O\right) + \tfrac{1}{4}\left(P_Q + \mathbb{I}\right) \tag{4.24}$$

We may now re-express the equations for the time dependence of the populations of the various spin levels in terms of these orthogonal polarizations. This produces the following set of somewhat simplified equations.

$$\begin{aligned}\frac{1}{W}\frac{\partial P_Z(x)}{\partial t} &= +5\left(P_Z(x+a) - P_Z(x)\right) \\ &\quad + 2\left(P_Q(x+a)P_Z(x) - P_Q(x)P_Z(x+a)\right)\end{aligned} \tag{4.25}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial P_Q(x)}{\partial t} &= -15P_Q(x) + 6P_Q(x)P_Q(x+a) \\ &\quad - \tfrac{27}{5}\left(P_O(x) - 2P_Z(x)\right)\left(P_Z(x+a)\right)\end{aligned} \tag{4.26}$$

$$\frac{1}{W}\frac{\partial P_O(x)}{\partial t} = -30P_O(x) + 12P_O(x)P_Q(x+a) + 6P_Q(x)P_Z(x+a) \tag{4.27}$$

The reduction from a set of four coupled differential equations to a set of three coupled differential equations is possible since only three out of the four populations can be independently specified before the fourth one can be calculated from the others. Alternately we can say that we still have four differential equations, but the fourth is for the time evolution of the identity operator. This equation is trivially equal to zero since the identity operator is time independent.

We immediately see that the diffusion of the polarization for a spin-three-halves does not obey the same equation as for a spin-one-half. This is exactly as we would expect since there are now three separate types of polarization. In particular, it should be noted that there are qualitatively three types of terms in the above equations, those that involve a spatial difference in a polarization, those that involve bilinear couplings of the polarizations, and exponential decay terms. The terms that consist of a polarization difference taken between two spins are exactly analogous to spin-1/2 polarization diffusion. Note that the quadrupolar and octapolar polarizations have no such diffusive term, and that they have strong negative feedback terms on their local value (decay terms). The bilinear coupling terms keep the each of the polarizations connected to both the local value and the curvature of the other polarizations. Finally, another very interesting aspect of these equations is that the coefficient of Zeeman polarization diffusion is approximately 5 times greater than for a spin-one-half.

D. Behavior of the Spin-Three-Halves Diffusion Equations

It is worth integrating the above equations in an approximate sense under a few different initial conditions in order to build an understanding of their behavior. Of course, we must integrate these equations over all couplings in the solid to arrive at exactly what happens in the sample in the general case, but this earliest time behavior will give us some idea of what direction the system starts to move toward when it has been placed in a given initial condition.

1. Spin Diffusion in a Spin-Three-Halves System Initially in a State of Uniform Pure Zeeman Polarization

If there is equal Zeeman polarization on both of the spins and no quadrupolar or octapolar polarization on either spin, then after a short time increment Δt, equations (4.25) through (4.27) yield

$$P_Z(\Delta t) = P_Z(0) \tag{4.28}$$

$$P_Q(\Delta t) = \tfrac{54}{5} W \Delta t P_Z^2(0) \tag{4.29}$$

$$P_O(\Delta t) = 0 \tag{4.30}$$

Where we have used the equivalence of the spins in this specific case to eliminate the spatial variable. Note that, if we consider the sum over all couplings to other spins W should be replaced with $\sum_i W_i$.

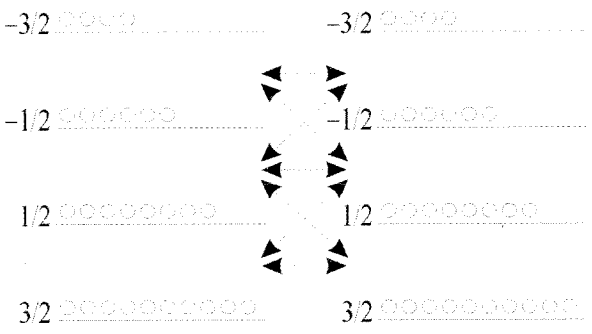


Figure 4.1 Level populations of two representative spins in states of pure Zeeman polarization.

It may appear bothersome that the quadrupolar polarization could "arise from nowhere" since the Zeeman polarization did not decrease. However, consider the two spins in figure 4.1, which are both in states of pure Zeeman polarization, and equations (4.12) through (4.15). We see that flip-flop rates for a flip from ³⁄₂ to ½ on one spin simultaneous with a flop on the other spin from ½ to ³⁄₂ is exactly canceled by the reverse reaction. This is true of all the "parallel" transitions down the ladder of states. However, if we consider the "diagonal transitions" such as ³⁄₂ to ½ simultaneous with $-$³⁄₂ to $-$½ we see that there is a net transition rate after the reverse reaction is considered. Indeed, using the relative populations in figure 4.1 and equations (4.12) through (4.15), we see that there will a net rate into the ³⁄₂ level that is proportional to $\left(24(32+18)-30(24+12)\right)/28^2 = 120/28^2$, a net rate into the $-$³⁄₂ level that is proportional to $\left(18(24+24)-12(30+32)\right)/28^2 = 120/28^2$, a net rate in to the ½ level that is proportional to $\left(30(24+12)-24(32+18)\right)/28^2 = -120/28^2$ plus $24(30+18)-32(24+12)=0$, and a net rate in to the $-$½ level that is proportional to $\left(12(30+32)-18(24+24)\right)/28^2 = -120/28^2$ plus $32(24+12)-24(30+18)=0$. The terms that are equal zero are the ones involving a flip-flop between the ½ and $-$½

levels, demonstrating again that the only net transfer of populations was from the $\frac{1}{2}$ and $-\frac{1}{2}$ levels to the $\frac{3}{2}$ and $-\frac{3}{2}$ levels. Immediately, we recognize the preferential depopulation of the $\frac{1}{2}$ and $-\frac{1}{2}$ levels and the simultaneous population of the $\frac{3}{2}$ and $-\frac{3}{2}$ levels as the addition of quadrupolar polarization, yet the Zeeman polarization did not change since there were equal movement up and down the ladder of states. Now that we have nonzero quadrupolar polarization, this will feedback into the equation for the octapolar polarization (the Zeeman feedback in equation (4.25) is zero due to the equivalence of the two spins). We will, therefore, have induced quadrupolar and octapolar polarizations without disturbing the Zeeman polarization in the slightest. However it was the very presence of the Zeeman polarization that induced the quadrupolar and octapolar polarizations. Finally, if we calculate $\frac{54}{5}P_Z^2(0)$ from equation (4.29) for the spins depicted in figure 4.1, we will see that the value is $480/28^2$, perfectly matching the value we arrive at by using equation (4.19) on the relative population shifts that we have just calculated above.

Before going any further, we must verify whether we are just generating the amounts of $\mathbf{I}_z^2$ and $\mathbf{I}_z^3$ necessary to describe the system with a single Zeeman spin temperature. We may suspect this because the term in equation (4.29) that we are concerned about is of order $(\hbar\omega/kT)^2$ since each term of $\mathbf{I}_z$ is of order $(\hbar\omega/kT)$ in the high temperature approximation. There are two ways to accomplish this calculation, by extending the high temperature approximation to higher powers of inverse temperature or by using exact formulas usually reserved for low temperature due the complexity of

solving them. We shall examine both here, and show that the high temperature approximation gives the wrong answer when it is taken out to quadratic powers of inverse temperature.

In using the high temperature approach, we compute the equilibrium density operator to higher powers of inverse temperature and insert this into equations (4.25) through (4.27) to see whether the rate of change of P_Q is still of order $(\hbar\omega/kT)^2$. To achieve this, we consider the expansion of the density operator again.

$$\rho = \frac{Exp(-\hbar\omega \mathbf{I}_z/kT)}{Tr(Exp(-\hbar\omega \mathbf{I}_z/kT))} = \frac{1-\hbar\omega \mathbf{I}_z/kT + \frac{1}{2}\mathbf{I}_z^2(\hbar\omega/kT)^2 - \frac{1}{6}\mathbf{I}_z^3(\hbar\omega/kT)^3 + \ldots}{Tr(Exp(-\hbar\omega \mathbf{I}_z/kT))} \quad (4.31)$$

We may obtain the various polarizations from this expansion by computing the expectation values given on the right side of equations (4.16), (4.19), and (4.20). These evaluate to

$$P_Z = \frac{\langle \mathbf{I}_z \rangle}{I} = \frac{Tr\left(\frac{\mathbf{I}_z}{I}\left(1-\hbar\omega \mathbf{I}_z/kT + \frac{1}{2}\mathbf{I}_z^2(\hbar\omega/kT)^2 - \frac{1}{6}\mathbf{I}_z^3(\hbar\omega/kT)^3 + \ldots\right)\right)}{Tr(Exp(-\hbar\omega \mathbf{I}_z/kT))} \quad (4.32)$$

$$P_Z = \frac{-5(\hbar\omega/kT) - \frac{41}{4}\frac{1}{3!}(\hbar\omega/kT)^3 - \frac{365}{16}\frac{1}{5!}(\hbar\omega/kT)^5 - \frac{3281}{64}\frac{1}{7!}(\hbar\omega/kT)^7 + \ldots}{\frac{3}{2}Tr(Exp(-\hbar\omega \mathbf{I}_z/kT))} \quad (4.33)$$

$$P_Q = \frac{\langle 3\mathbf{I}_z^{\,2} - \mathbf{I}\cdot\mathbf{I} \rangle}{I(2I-1)} = \frac{Tr\left(\frac{3\mathbf{I}_z^{\,2} - \mathbf{I}\cdot\mathbf{I}}{I(2I-1)}\left(1-\hbar\omega \mathbf{I}_z/kT + \frac{1}{2}\mathbf{I}_z^2(\hbar\omega/kT)^2 + \ldots\right)\right)}{Tr(Exp(-\hbar\omega \mathbf{I}_z/kT))} \quad (4.34)$$

$$P_Q = \frac{4\frac{1}{2}(\hbar\omega/kT)^2 + 10\frac{1}{4!}(\hbar\omega/kT)^4 + \frac{91}{4}\frac{1}{6!}(\hbar\omega/kT)^6 + \frac{205}{4}\frac{1}{8!}(\hbar\omega/kT)^8 + \ldots}{Tr(Exp(-\hbar\omega \mathbf{I}_z/kT))} \quad (4.35)$$

$$P_O = \tfrac{2}{9}\left\langle Tr(\mathbf{I}_z{}^2)\mathbf{I}_z{}^3 - Tr(\mathbf{I}_z{}^4)\mathbf{I}_z \right\rangle = \frac{Tr\left(\tfrac{2}{9}\left(Tr(\mathbf{I}_z{}^2)\mathbf{I}_z{}^3 - Tr(\mathbf{I}_z{}^4)\mathbf{I}_z\right)\left(1-\hbar\omega\mathbf{I}_z/kT+\ldots\right)\right)}{Tr(Exp(-\hbar\omega\mathbf{I}_z/kT))} \quad (4.36)$$

$$P_O = \frac{-2\tfrac{1}{3!}(\hbar\omega/kT)^3 - 5\tfrac{1}{5!}(\hbar\omega/kT)^5 - \tfrac{91}{8}\tfrac{1}{7!}(\hbar\omega/kT)^7 - \tfrac{205}{8}\tfrac{1}{9!}(\hbar\omega/kT)^9 + \ldots}{Tr(Exp(-\hbar\omega\mathbf{I}_z/kT))} \quad (4.37)$$

with

$$\begin{aligned} Tr(Exp(-\hbar\omega\mathbf{I}_z/kT)) = 1 &+ 5\tfrac{1}{2}(\hbar\omega/kT)^2 + \tfrac{41}{4}\tfrac{1}{4!}(\hbar\omega/kT)^4 \\ &+ \tfrac{365}{16}\tfrac{1}{6!}(\hbar\omega/kT)^6 + \tfrac{3281}{64}\tfrac{1}{8!}(\hbar\omega/kT)^8 + \ldots \end{aligned} \quad (4.38)$$

If we keep terms in inverse temperature squared, and insert these equations into equations (4.25) through (4.27), we should obtain cancellation of the time derivative of the quadrupolar polarization if that time derivative was previously due to not keeping such terms. Upon making these substitutions we arrive at

$$\frac{\partial P_Z}{\partial t} = 0 \quad (4.39)$$

$$\begin{aligned} \frac{1}{W}\frac{\partial P_Q(x)}{\partial t} = &-15\frac{4\tfrac{1}{2}(\hbar\omega/kT)^2}{1+5\tfrac{1}{2}(\hbar\omega/kT)^2} + 6\left(\frac{4\tfrac{1}{2}(\hbar\omega/kT)^2}{1+5\tfrac{1}{2}(\hbar\omega/kT)^2}\right)^2 \\ &+ 270(\hbar\omega/kT)^2 \end{aligned} \quad (4.40)$$

$$\frac{\partial P_O}{\partial t} = -6\frac{4\tfrac{1}{2}(\hbar\omega/kT)^2}{1+5\tfrac{1}{2}(\hbar\omega/kT)^2}\frac{5(\hbar\omega/kT)}{1+5\tfrac{1}{2}(\hbar\omega/kT)^2} \quad (4.41)$$

Note that there is no single temperature for which equations (4.40) and (4.41) yield a zero rate of change for the quadrupolar and octapolar polarizations. That is, setting equations (4.40) and (4.41) equal to zero produces equations that have complex roots. Thus, we see

that, taken at this level of accuracy, a high temperature approximation appears to indicate that dipolar flip-flops really do move the system away from a single spin temperature.

Let's try to calculate this again, but this time we will not make any high temperature approximation. We may carry out the traces in equations (4.32), (4.34), (4.36) exactly, producing

$$P_Z = \frac{Exp(3\hbar\omega/2kT) + \frac{1}{3}Exp(\hbar\omega/2kT) - \frac{1}{3}Exp(-\hbar\omega/2kT) - Exp(-3\hbar\omega/2kT)}{Exp(3\hbar\omega/2kT) + Exp(\hbar\omega/2kT) + Exp(-\hbar\omega/2kT) + Exp(-3\hbar\omega/2kT)} \quad (4.42)$$

$$P_Q = \frac{Exp(3\hbar\omega/2kT) - Exp(\hbar\omega/2kT) - Exp(-\hbar\omega/2kT) + Exp(-3\hbar\omega/2kT)}{Exp(3\hbar\omega/2kT) + Exp(\hbar\omega/2kT) + Exp(-\hbar\omega/2kT) + Exp(-3\hbar\omega/2kT)} \quad (4.43)$$

$$P_O = \frac{\frac{1}{3}Exp(3\hbar\omega/2kT) - Exp(\hbar\omega/2kT) + Exp(-\hbar\omega/2kT) - \frac{1}{3}Exp(-3\hbar\omega/2kT)}{Exp(3\hbar\omega/2kT) + Exp(\hbar\omega/2kT) + Exp(-\hbar\omega/2kT) + Exp(-3\hbar\omega/2kT)} \quad (4.44)$$

Inserting these equations into equations (4.25) through (4.27), and letting *Mathematica* do the pages of algebra to simplify the results, we obtain

$$\frac{\partial P_Z(x)}{\partial t} = \frac{\partial P_Q(x)}{\partial t} = \frac{\partial P_O(x)}{\partial t} = 0 \quad (4.45)$$

Thus, for a spatially uniform spin-three-halves system, we have proved the spin temperature hypothesis, which states that the dipolar couplings bring the spins towards a single spin temperature along $\mathbf{I}_z$.

Since the spatial dependence drops out in this particular case, we can numerically integrate the equations. That is, since all the nuclei are equivalent whatever one does to another is exactly reciprocated. This means that, whether we think of the spin

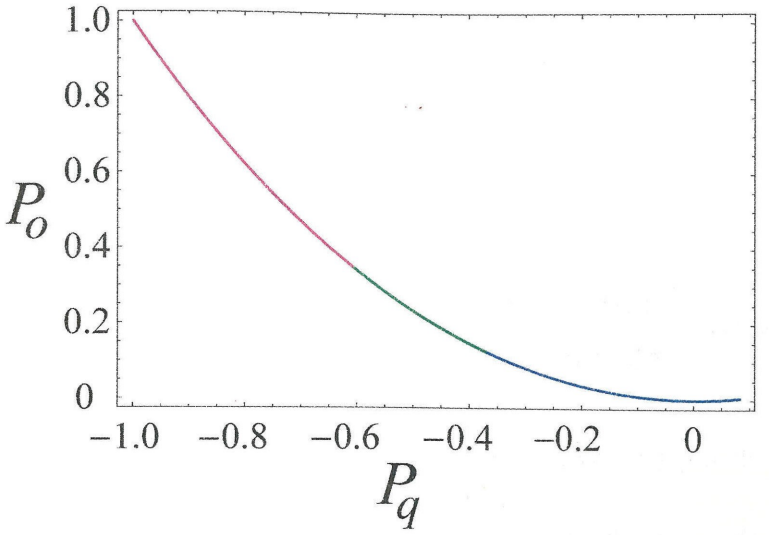


Figure 4.2 The recovery of a system from a non-Boltzmann state. Equations (4.25) through (4.27) have been numerically integrated for a spatially homogeneous system. The initial state is $P_Z = 1/3$, $P_Q = -1$, $P_O = 1$, which corresponds to $\rho = |-½\rangle\langle -½|$. The line is color coded to illustrate the time scales over which different polarizations have fallen to $1/e$ times their initial value. The purple line is $0 \le t \le 1/40W$, the green line is $1/40W \le t \le 1/20W$, and the blue line is $1/20W \le t \le 1/2W$.

reciprocating perfectly with a neighbor or we think of the spin cross coupled to its own polarization levels, it does not matter.

We choose to start the spins from a point where they have all been prepared in the $|-½\rangle$ state. The quadrupolar and octapolar polarizations then race towards equilibrium with the Zeeman polarization. However, since the flip-flops preserve the total Zeeman polarization and since each spin is equal to all others, the Zeeman polarization is absolutely constant here. Figure 4.2 illustrates just such a system. The time scale for recovery should be roughly $1/W_\Sigma$ where W_Σ is the sum of all of the individual rates driving this reaction, given by

$$W_{\Sigma} = \sum \frac{\pi^2 \nu_d^2}{\sqrt{2\pi}\left\langle \Delta\nu^2 \right\rangle^{\frac{1}{2}}} = \frac{\pi^2 \left\langle \Delta\nu^2 \right\rangle_{II}}{\sqrt{2\pi}\left\langle \Delta\nu^2 \right\rangle^{\frac{1}{2}}} \tag{4.46}$$

For ^{71}Ga this leads us to evaluate W_{Σ} as

$$W_{\Sigma} = \frac{\pi^2 (545.5 \text{ Hz})^2}{(1239 \text{ Hz})\sqrt{2\pi}} = 946 \text{ Hz} \tag{4.47}$$

We see that the entire system recovered from the extreme perturbation we have imposed after a time of $\approx 1/2W$ which evaluates to ~530 μs for ^{71}Ga. Moreover, the $1/e$ decay time of $\approx 1/40W$ for octapolar polarization evaluates to ~26 μs for 71Ga, and the $1/e$ decay time of $\approx 1/20W$ for quadrupolar polarization evaluates to ~53 μs for 71Ga. Keep in mind that the time constant T_{II} ($\approx$ 2.2 μs) that we previously calculated was for a single spin whose polarization differs from the thermal bath consisting of all of the other spins in the crystal. These time constants are for the entire bath to reach equilibrium within itself.

At first glance, it appears odd that the system can find the Boltzmann state defined by its Zeeman polarization without the introduction of thermal or statistical concepts into our mathematics. However, our very concept of a ladder of discrete levels that continuously exchange population through a reaction kinetics type of scheme, eventually leading to equal forward and reverse reactions, implicitly introduces thermal equilibrium into the equations.

2. Spin Diffusion in a Spin-Three-Halves System Initially in a State of Pure Zeeman Polarization with a Polarization Difference

Let us consider for a moment what would happen if we introduced a Zeeman polarization difference between the two nuclei considered above when there is no quadrupolar or octapolar polarization. Let the spin at position $x+a$ be in a state of pure Zeeman polarization, and let the spin at position x be completely depolarized. Again, at a short time Δt later, we may then write

$$P_Z(x,\Delta t) = 5W\Delta t P_Z(x+a,0) \tag{4.48}$$

$$P_Q(x,\Delta t) = 0 \tag{4.49}$$

$$P_O(\Delta t) = 0 \tag{4.50}$$

We may then extend this to

$$P_Z(x,2\Delta t) = P_Z(x,\Delta t) + 5W\Delta t\left(P_Z(x+a,\Delta t) - P_Z(x,\Delta t)\right) \tag{4.51}$$

$$P_Q(x,2\Delta t) = \tfrac{54}{5}W\Delta t P_Z(x,\Delta t)P_Z(x+a,\Delta t) \tag{4.52}$$

$$P_O(x,2\Delta t) = 0 \tag{4.53}$$

We see that the second term of equation (4.26) has now been activated, leading to quadrupolar polarization that will feed into octapolar polarization. Thus, we will again have an induction of quadrupolar and octapolar polarization by pure Zeeman polarization on the neighboring spins. In this case, we absolutely must integrate equations (4.25),

(4.26), and (4.27) simultaneously over each spin pair to allow for many body effects if we are to achieve an exactly correct description of spin diffusion.

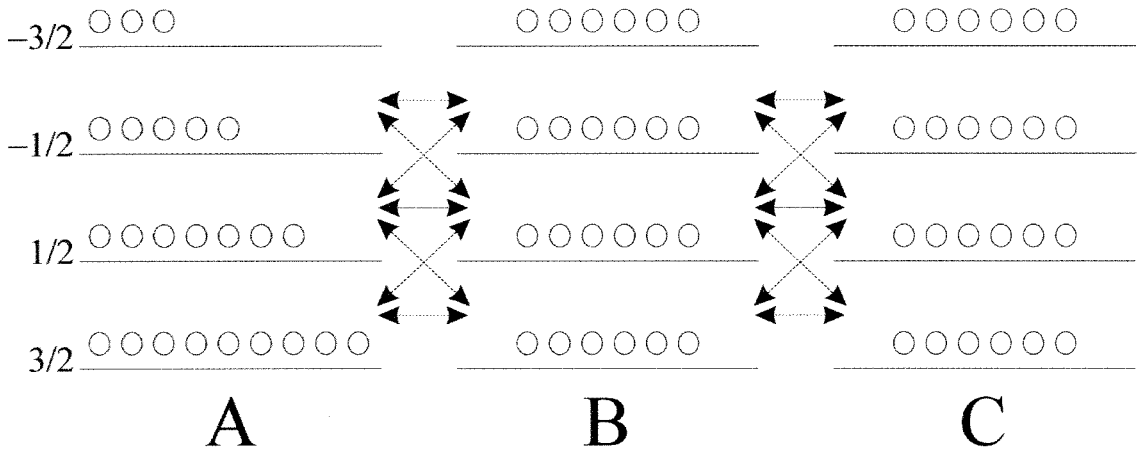


Figure 4.3 A schematic illustration of the energy levels and transition probabilities in a series of spin-three-halves nuclei. Nucleus A is in a state of pure Zeeman order while the other two nuclei are not polarized. The flip-flops represented by horizontal arrows have a 9:16:9 ratio of transition rates while the flip-flops represented by diagonal arrows all have a ratio of 12 relative to these. Note that the flip-flops between the $\frac{1}{2}$ to $\frac{3}{2}$ levels on one spin with the $-\frac{3}{2}$ to $-\frac{1}{2}$ levels on another spin are not drawn here but have a relative transition probability of 9.

We can arrive at the same conclusions about the cross coupling of the Zeeman, quadrupolar, and octapolar polarizations by simply considering figure 4.3 and using equations (4.25) through (4.27), as in the previous section, to gain a qualitative understanding of the behavior of the polarization as diffusion starts. Using the relative populations in figure 4.3, we see that for nucleus A there will a net rate into the $\frac{3}{2}$ level

that is proportional to $3(7-9)60=-360$, a net rate in to the $\frac{1}{2}$ level that is proportional to $-3(7+9)60-4(7+5)60=-120$, a net rate in to the $-\frac{1}{2}$ level that is proportional to $-3(5+3)60+4(7-5)60=120$, and a net rate into the $-\frac{3}{2}$ level that is proportional to $3(5-3)60=360$. Meanwhile, nucleus B will have a net rate into the $\frac{3}{2}$ level that is proportional to $18(70)-18(50)=360$, a net rate in to the $\frac{1}{2}$ level that is proportional to $-18(70)+18(50)+24(70)-24(50)=120$, a net rate in to the $-\frac{1}{2}$ level that is proportional to $18(70)-18(50)+24(50)-24(70)=-120$, and a net rate into the $-\frac{3}{2}$ level that is proportional to $18(50)-18(70)=-360$. We immediately recognize this as the transfer of Zeeman polarization from nucleus A to nucleus B. Moreover, now that there is Zeeman polarization on both nuclei, there will arise quadrupolar polarization on both nuclei in the next step precisely as we computed in equation (4.52), and in direct analogy to our work in the previous section that was based on using the populations in figure 4.1 of that section. Finally, normalizing all of these population shifts by dividing by 24^2 and subsequent application of equation (4.16) to calculate the relative quantity of Zeeman polarization being transferred exactly agrees with equation (4.48) when it is applied to the populations in figure 4.3.

3. The Spin-Three-Halves Continuous Medium Polarization Diffusion Equation

Equations (4.25) though (4.27) can be amended by adding a spin at $x-a$, equating the second difference with the second derivative as in equation (4.10), and by recognizing that

$$P_\alpha(x-a)+P_\alpha(x+a)=P_\alpha(x-a)+P_\alpha(x+a)-2P_\alpha(x)+2P_\alpha(x)$$
$$=\frac{\partial^2 P_\alpha(x)}{\partial x^2}+2P_\alpha(x) \quad , \tag{4.54}$$

The point of expressing these terms in this way is to make equation (4.55) through (4.57) depend only on the value of the polarizations at x and the derivatives of the polarizations at x, yielding

$$\frac{1}{W}\frac{\partial P_Z(x)}{\partial t}=5a^2\frac{\partial^2 P_Z(x)}{\partial x^2}+2a^2\left(\frac{\partial^2 P_Q(x)}{\partial x^2}P_Z(x)-\frac{\partial^2 P_Z(x)}{\partial x^2}P_Q(x)\right) \tag{4.55}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial P_Q(x)}{\partial t}&=-30P_Q(x)+6a^2\frac{\partial^2 P_Q(x)}{\partial x^2}P_Q(x)+12P_Q^2(x)\\&+\tfrac{27}{5}a^2\frac{\partial^2 P_Z(x)}{\partial x^2}\left(P_O(x)-2P_Z(x)\right)\\&+\tfrac{54}{5}P_Z(x)\left(P_O(x)-2P_Z(x)\right)\end{aligned} \tag{4.56}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial P_O(x)}{\partial t}&=-60P_O(x)+12a^2\frac{\partial^2 P_Q(x)}{\partial x^2}P_O(x)+6a^2\frac{\partial^2 P_Z(x)}{\partial x^2}P_Q(x)\\&+24P_Q(x)P_O(x)+12P_Q(x)P_Z(x)\end{aligned} \tag{4.57}$$

We may again generalize this result to a rectangular grid of such spins. Upon doing so we obtain

$$\frac{1}{W}\frac{\partial P_Z(r,t)}{\partial t}=5a^2\nabla^2P_Z(r,t)+2a^2\left(\left(\nabla^2P_Q(r,t)\right)P_Z(r,t)-\left(\nabla^2P_Z(r,t)\right)P_Q(r,t)\right) \tag{4.58}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial P_Q(r,t)}{\partial t}&=-90P_Q(r,t)+6P_Q(r,t)a^2\nabla^2P_Q(r,t)+36P_Q^2(r,t)\\&+\tfrac{27}{5}\left(P_O(r,t)-2P_Z(r,t)\right)a^2\nabla^2P_Z(r,t)\\&+\tfrac{162}{5}\left(P_O(r,t)-2P_Z(r,t)\right)P_Z(r,t)\end{aligned} \tag{4.59}$$

$$\frac{1}{W}\frac{\partial P_O(r,t)}{\partial t} = -180P_O(r,t) + 6a^2\left(2P_O(r,t)\nabla^2 P_Q(r,t) + P_Q(r,t)\nabla^2 P_Z(r,t)\right) + 72P_Q(r,t)P_O(r,t) + 36P_Q(r,t)P_Z(r,t) \tag{4.60}$$

These equations are necessarily more complicated than the corresponding spin-one-half equations, and, therefore, we do not expect the diffusion process to be characterized by a single constant in this case. However, we note that the factor $D = Wa^2$, the diffusion constant in the spin-one-half case, is present as a coefficient of each of the spatial second derivatives in this case as well.

E. Spin Diffusion During the ONP Process

We will now consider the effects of optical nuclear polarization, ONP, occurring simultaneously with spin diffusion. If optical polarization is taken to proceed through fluctuations in the hyperfine contact between the electron and nucleus, then the equilibrium polarization of each of the nuclear Zeeman transitions will approximately equal the polarization of the electron spin. This can be reworded by saying that the nuclear density matrix that results from optical pumping will be approximately the same as the thermal nuclear density matrix taken at the electron spin temperature. However, the electron spin temperature can easily be in the micro-Kelvin range given the optically polarized origin of the electrons, and the high temperature approximation will likely fail even for small nuclear spin energy spacing. What results is a sum of $\mathbb{I}$, $\mathbf{I}_z$, $\mathbf{I}_z^2$, and $\mathbf{I}_z^3$ terms that will converge upon $\rho = -\frac{1}{16}\mathbb{I} - \frac{1}{24}\mathbf{I}_z + \frac{1}{4}\mathbf{I}_z^2 + \frac{1}{6}\mathbf{I}_z^3 = \left|\frac{3}{2}\right\rangle\left\langle\frac{3}{2}\right|$ as the temperature

approaches absolute zero. In this state $p_{\frac{3}{2}}$ equals 1 while all of the other levels are unpopulated; consequently, P_Z and P_Q equal 1 while P_O equals $\frac{1}{3}$. It should not bother us that at extremely low electron spin temperatures we will directly optically pump quadrupolar and octapolar nuclear polarization since these are the natural basis functions at high temperature and, thus, our notions about them are quite out of place here.

Let us now consider the case where a spatially dependent optical nuclear polarization feeding the population differences and a competing relaxation process. These are included in equations (4.61) through (4.64).

$$\begin{aligned}\frac{\partial p_{\frac{3}{2}}(x)}{\partial t} = &\frac{1}{T_{1\text{on}}(x)}\left(p_{\frac{3}{2},t_{\text{ONP}}\to\infty} - p_{\frac{3}{2}}(x)\right) - \frac{1}{T_{1\text{off}}(x)}\left(p_{\frac{3}{2},\text{Lattice}} - p_{\frac{3}{2}}(x)\right)\\ &+W(3p_{\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &\quad -3p_{\frac{3}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a)))\end{aligned} \tag{4.61}$$

$$\begin{aligned}\frac{\partial p_{\frac{1}{2}}(x)}{\partial t} = &\frac{1}{T_{1\text{on}}(x)}\left(p_{\frac{1}{2},t_{\text{ONP}}\to\infty} - p_{\frac{1}{2}}(x)\right) - \frac{1}{T_{1\text{off}}(x)}\left(p_{\frac{1}{2},\text{Lattice}} - p_{\frac{1}{2}}(x)\right)\\ &+W(3p_{\frac{3}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &\quad -3p_{\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &\quad +4p_{-\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &\quad -4p_{\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a)))\end{aligned} \tag{4.62}$$

$$\begin{aligned}\frac{\partial p_{-\frac{1}{2}}(x)}{\partial t} = &\frac{1}{T_{1\text{on}}(x)}\left(p_{-\frac{1}{2},t_{\text{ONP}}\to\infty} - p_{-\frac{1}{2}}(x)\right) - \frac{1}{T_{1\text{off}}(x)}\left(p_{-\frac{1}{2},\text{Lattice}} - p_{-\frac{1}{2}}(x)\right)\\ &+W(3p_{-\frac{3}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a))\\ &\quad -3p_{-\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &\quad +4p_{\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a))\\ &\quad -4p_{-\frac{1}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a)))\end{aligned} \tag{4.63}$$

$$\frac{\partial p_{-\frac{3}{2}}(x)}{\partial t} = \frac{1}{T_{1on}(x)}\left(p_{-\frac{3}{2},t_{ONP}\to\infty} - p_{-\frac{3}{2}}(x)\right) - \frac{1}{T_{1off}(x)}\left(p_{-\frac{3}{2},\text{Lattice}} - p_{-\frac{3}{2}}(x)\right)$$
$$+W(3p_{-\frac{1}{2}}(x)(3p_{\frac{1}{2}}(x+a)+4p_{-\frac{1}{2}}(x+a)+3p_{-\frac{3}{2}}(x+a)) \qquad (4.64)$$
$$-3p_{-\frac{3}{2}}(x)(3p_{\frac{3}{2}}(x+a)+4p_{\frac{1}{2}}(x+a)+3p_{-\frac{1}{2}}(x+a)))$$

We may again express these equations in terms of the orthogonal polarizations. The equations are further simplified if we define the following asymptotically limiting values of the various types of polarization.

$$P_{Z,t_{ONP}\to\infty} = p_{\frac{3}{2},t_{ONP}\to\infty} + \tfrac{1}{3}p_{\frac{1}{2},t_{ONP}\to\infty} - \tfrac{1}{3}p_{-\frac{1}{2},t_{ONP}\to\infty} - p_{-\frac{3}{2},t_{ONP}\to\infty} \qquad (4.65)$$

$$P_{Q,t_{ONP}\to\infty} = p_{\frac{3}{2},t_{ONP}\to\infty} - p_{\frac{1}{2},t_{ONP}\to\infty} - p_{-\frac{1}{2},t_{ONP}\to\infty} + p_{-\frac{3}{2},t_{ONP}\to\infty} \qquad (4.66)$$

$$P_{O,t_{ONP}\to\infty} = \tfrac{1}{3}p_{\frac{3}{2},t_{ONP}\to\infty} - p_{\frac{1}{2},t_{ONP}\to\infty} + p_{-\frac{1}{2},t_{ONP}\to\infty} - \tfrac{1}{3}p_{-\frac{3}{2},t_{ONP}\to\infty} \qquad (4.67)$$

$$P_{Z,\text{Lattice}} = p_{\frac{3}{2},\text{Lattice}} + \tfrac{1}{3}p_{\frac{1}{2},\text{Lattice}} - \tfrac{1}{3}p_{-\frac{1}{2},\text{Lattice}} - p_{-\frac{3}{2},\text{Lattice}} \qquad (4.68)$$

$$P_{Q,\text{Lattice}} = p_{\frac{3}{2},\text{Lattice}} - p_{\frac{1}{2},\text{Lattice}} - p_{-\frac{1}{2},\text{Lattice}} + p_{-\frac{3}{2},\text{Lattice}} \qquad (4.69)$$

$$P_{Z,\text{Lattice}} = \tfrac{1}{3}p_{\frac{3}{2},\text{Lattice}} - p_{\frac{1}{2},\text{Lattice}} + p_{-\frac{1}{2},\text{Lattice}} - \tfrac{1}{3}p_{-\frac{3}{2},\text{Lattice}} \qquad (4.70)$$

The first three polarization values are those that the nuclei would reach if they were allowed to polarize until they reached thermal equilibrium with the electron spins without any couplings to the lattice, and the second three polarization values are those that the nuclei reach when they are only allowed to thermally equilibrate with the lattice. However, the polarization values due to thermal equilibration with the lattice are essentially the high temperature limit values, since the high temperature approximation holds to well below one degree Kelvin even for a one Gigahertz resonance frequency (our

resonance is at ~3.2 Megahertz). Inserting the asymptotic polarization limits above, we have

$$\frac{\partial P_Z(x)}{\partial t} = \frac{1}{T_{1on}(x)}\left(P_{Z,t_{ONP}\to\infty} - P_Z(x)\right) + \frac{1}{T_{1off}(x)}\left(P_{Z,\text{Lattice}} - P_Z(x)\right) + 5a^2\frac{\partial^2 P_Z(x)}{\partial x^2} + 2a^2\left(\frac{\partial^2 P_Q(x)}{\partial x^2}P_Z(x) - \frac{\partial^2 P_Z(x)}{\partial x^2}P_Q(x)\right) \tag{4.71}$$

$$\frac{\partial P_Q(x)}{\partial t} = \frac{1}{T_{1on}(x)}\left(P_{Q,t_{ONP}\to\infty} - P_Q(x)\right) + \frac{1}{T_{1off}(x)}\left(P_{Q,\text{Lattice}} - P_Q(x)\right) - 30P_Q(x) + 6a^2\frac{\partial^2 P_Q(x)}{\partial x^2}P_Q(x) + 12P_Q^2(x) + \tfrac{27}{5}a^2\left(\frac{\partial^2 P_Z(x)}{\partial x^2} + 2P_Z(x)\right)\left(P_Q(x) - 2P_Z(x)\right) \tag{4.72}$$

$$\frac{\partial P_O(x)}{\partial t} = \frac{1}{T_{1on}(x)}\left(P_{O,t_{ONP}\to\infty} - P_O(x)\right) + \frac{1}{T_{1off}(x)}\left(P_{O,\text{Lattice}} - P_O(x)\right) - 60P_O(x) + 12a^2\frac{\partial^2 P_Q(x)}{\partial x^2}P_O(x) + 6a^2\frac{\partial^2 P_Z(x)}{\partial x^2}P_Q(x) + 24P_Q(x)P_O(x) + 12P_Q(x)P_Z(x) \tag{4.73}$$

We see that the results of ONP and spin diffusion are not mixed or altered by the two processes occurring simultaneously. That is, the result of the sum of the processes is the sum of the results of the individual processes. Therefore, we may calculate the ONP exactly from one discreet time point to another by integrating the equations without spin diffusion. This polarization is then allowed to diffuse over a set number of sites before the time is moved forward. The ONP is then calculated for the next time step and the process repeats until we have stepped forward through the desired time interval. This a tremendously tedious process since the time scale for spin diffusion is so incredibly short

(~microseconds) compared to the time scale of the experiment (hundred of milliseconds to seconds), and since there are tens to hundreds of thousands of nuclei in the relevant volume about each defect.

However, because equations (4.71) though (4.73) are just equations (4.25) through (4.27) with ONP at each point added in, we suspect that moving to a continuum model will greatly improve the computational tractability of the problem. Upon making the appropriate changes, we arrive at

$$\begin{aligned}\frac{\partial P_Z(x)}{\partial t} &= \frac{1}{T_{1\text{on}}(x)}\left(P_{Z,t_{\text{ONP}}\to\infty} - P_Z(x)\right) + \frac{1}{T_{1\text{off}}(x)}\left(P_{Z,\text{Lattice}} - P_Z(x)\right) \\ &+5a^2\nabla^2 P_Z(r,t) + 2a^2\left(\left(\nabla^2 P_Q(r,t)\right)P_Z(r,t) - \left(\nabla^2 P_Z(r,t)\right)P_Q(r,t)\right)\end{aligned} \tag{4.74}$$

$$\begin{aligned}\frac{\partial P_Q(x)}{\partial t} &= \frac{1}{T_{1\text{on}}(x)}\left(P_{Q,t_{\text{ONP}}\to\infty} - P_Q(x)\right) + \frac{1}{T_{1\text{off}}(x)}\left(P_{Q,\text{Lattice}} - P_Q(x)\right) \\ &-90P_Q(r,t) + 6P_Q(r,t)a^2\nabla^2 P_Q(r,t) + 36P_Q^2(r,t) \\ &+\tfrac{27}{5}\left(P_O(r,t) - 2P_Z(r,t)\right)a^2\nabla^2 P_Z(r,t) \\ &+\tfrac{162}{5}\left(P_O(r,t) - 2P_Z(r,t)\right)P_Z(r,t)\end{aligned} \tag{4.75}$$

$$\begin{aligned}\frac{\partial P_O(x)}{\partial t} &= \frac{1}{T_{1\text{on}}(x)}\left(P_{O,t_{\text{ONP}}\to\infty} - P_O(x)\right) + \frac{1}{T_{1\text{off}}(x)}\left(P_{O,\text{Lattice}} - P_O(x)\right) \\ &-180P_O(r,t) + 6a^2\left(2P_O(r,t)\nabla^2 P_Q(r,t) + P_Q(r,t)\nabla^2 P_O(r,t)\right) \\ &+72P_Q(r,t)P_O(r,t) + 36P_Q(r,t)P_Z(r,t)\end{aligned} \tag{4.76}$$

This set of coupled differential equations describes the simultaneous polarization diffusion and ONP subject to the same constraints we encounter in the spin-1/2 case. Namely, the anisotropy of W was neglected in order to form ∇^2 from the one-

dimensional derivatives, and the anisotropy due to the structure of the crystal lattice not being simple cubic is also absent.

F. Spin Diffusion in the Presence of a Well Resolved Quadrupole Splitting.

The Hamiltonian of a spin that is subject to both Zeeman and secular quadrupole interactions and that has a Zeeman Hamiltonian that is large enough to truncate the quadrupole Hamiltonian is written as[1]

$$H = -\omega_L I_z + \omega_q \left(3I_z^2 - \mathbf{I}\cdot\mathbf{I}\right) \tag{4.77}$$

The effect of the quadrupole term above is to shift the ½ and $-$½ levels down in energy by $\omega_q/4$ and to shift the ³⁄₂ and $-$³⁄₂ levels up in energy by $\omega_q/4$. This results in satellite transitions at $\omega_L \pm \omega_q/2$ and an unshifted central transition. When the quadrupole Hamiltonian is strong enough to split the resonance into three resolved lines, flip-flops between the lines become a non-energy-conserving process. Equivalently, such flip-flops are truncated by the large quadrupole shift of one resonance from another. Therefore, the relevant spin diffusion equations are

$$\frac{1}{W}\frac{\partial p_{\frac{3}{2}}(x)}{\partial t} = 9p_{\frac{1}{2}}(x)p_{\frac{3}{2}}(x+a) - 9p_{\frac{3}{2}}(x)p_{\frac{1}{2}}(x+a) \tag{4.78}$$

$$\frac{1}{W}\frac{\partial p_{\frac{1}{2}}(x)}{\partial t} = 9p_{\frac{3}{2}}(x)p_{\frac{1}{2}}(x+a) - 9p_{\frac{1}{2}}(x)p_{\frac{3}{2}}(x+a) + 16p_{-\frac{1}{2}}(x)p_{\frac{1}{2}}(x+a) - 16p_{\frac{1}{2}}(x)p_{-\frac{1}{2}}(x+a) \tag{4.79}$$

$$\frac{1}{W}\frac{\partial p_{-\frac{1}{2}}(x)}{\partial t} = 9p_{-\frac{3}{2}}(x)p_{-\frac{1}{2}}(x+a) - 9p_{-\frac{1}{2}}(x)p_{-\frac{3}{2}}(x+a) + 16p_{\frac{1}{2}}(x)p_{-\frac{1}{2}}(x+a) - 16p_{-\frac{1}{2}}(x)p_{\frac{1}{2}}(x+a) \tag{4.80}$$

$$\frac{1}{W}\frac{\partial p_{-\frac{3}{2}}(x)}{\partial t} = 9p_{-\frac{1}{2}}(x)p_{-\frac{3}{2}}(x+a) - 9p_{-\frac{3}{2}}(x)p_{-\frac{1}{2}}(x+a) \tag{4.81}$$

Again, we may gain insight by converting these equations to our orthogonal polarization basis. This yields

$$\frac{1}{W}\frac{\partial P_Z(x)}{\partial t} = \tfrac{17}{10}\left(P_Z(x+a) - P_Z(x)\right) - \tfrac{3}{5}\left(P_O(x+a) - P_O(x)\right) - \tfrac{33}{10}\left(P_O(x)P_Q(x+a) - P_O(x+a)P_Q(x)\right) + \tfrac{26}{10}\left(P_Z(x)P_Q(x+a) - P_Z(x+a)P_Q(x)\right) \tag{4.82}$$

$$\frac{1}{W}\frac{\partial P_Q(x)}{\partial t} = \tfrac{9}{2}\left(P_Q(x+a) - P_Q(x)\right) + \tfrac{81}{10}\left(P_O(x)P_Z(x+a) - P_O(x+a)P_Z(x)\right) \tag{4.83}$$

$$\frac{1}{W}\frac{\partial P_O(x)}{\partial t} = \tfrac{54}{5}\left(P_O(x+a) - P_O(x)\right) - \tfrac{3}{5}\left(P_Z(x+a) - P_Z(x)\right) + \tfrac{27}{5}\left(P_O(x)P_Q(x+a) - P_O(x+a)P_Q(x)\right) + \tfrac{6}{5}\left(P_Z(x)P_Q(x+a) - P_Z(x+a)P_Q(x)\right) \tag{4.84}$$

These equations have several interesting features. Note that all three types of polarization now have diffusive behavior analogous to the spin-1/2 case, and the coefficients of these terms are in a ratio of 17:45:108 for Zeeman, quadrupolar and octapolar polarizations respectively. Additionally, there are now small negative diffusion

feedback terms between the Zeeman and octapolar polarizations. That is, diffusion of one type of polarization somewhat impedes diffusion of the other type of polarization. Moreover, while the quadrupolar polarization lacks such a diffusive term connecting it to the Zeeman and octapolar polarization, it is still connected to them by the bilinear coupling terms that attempt to equilibrate each of the polarizations with respect to the others. Finally, there are no decay terms like the ones we encountered earlier.

The continuous medium versions of these equations are

$$\begin{aligned}\frac{1}{W}\frac{\partial P_Z(r,t)}{\partial t} &= \tfrac{17}{10}a^2\nabla^2 P_Z(r,t) - \tfrac{3}{5}a^2\nabla^2 P_O(r,t) \\ &\quad - \tfrac{33}{10}a^2\left(P_O(r,t)\nabla^2 P_Q(r,t) - P_Q(r,t)\nabla^2 P_O(r,t)\right) \\ &\quad + \tfrac{26}{10}a^2\left(P_Z(r,t)\nabla^2 P_Q(r,t) - P_Q(r,t)\nabla^2 P_Z(r,t)\right)\end{aligned} \tag{4.85}$$

$$\frac{1}{W}\frac{\partial P_Q(r,t)}{\partial t} = \tfrac{9}{2}a^2\nabla^2 P_Q(r,t) + \tfrac{81}{10}a^2\left(P_O(r,t)\nabla^2 P_Z(r,t) - P_Z(r,t)\nabla^2 P_O(r,t)\right) \tag{4.86}$$

$$\begin{aligned}\frac{1}{W}\frac{\partial P_O(x)}{\partial t} &= \tfrac{54}{5}a^2\nabla^2 P_O(r,t) - \tfrac{3}{5}a^2\nabla^2 P_Z(r,t) \\ &\quad + \tfrac{27}{5}a^2\left(P_O(r,t)\nabla^2 P_Q(r,t) - P_Q(r,t)\nabla^2 P_O(r,t)\right) \\ &\quad + \tfrac{6}{5}a^2\left(P_Z(r,t)\nabla^2 P_Q(r,t) - P_Q(r,t)\nabla^2 P_Z(r,t)\right)\end{aligned} \tag{4.87}$$

G. Behavior of the Equations for Spin Diffusion in the Presence of a Well Resolved Quadrupole Splitting

We see that it is also worth integrating these new equations in an approximate sense under a few different initial conditions to build a similar understanding of their

behavior as we did in section D of this chapter. Again, we must integrate these equations over all couplings in the solid to arrive at exactly what happens in the sample in the general case. Nevertheless, this earliest time behavior gives us some idea of what direction the system starts to move toward when it has been placed in a given initial condition.

1. Spin Diffusion in the Presence of a Well Resolved Quadrupole Splitting for a System Initially in a State of Uniform Polarization

If there are equal amounts of each of the polarizations on both spins, then equations (4.82) through (4.84) yield

$$\frac{\partial P_Z}{\partial t} = \frac{\partial P_Q}{\partial t} = \frac{\partial P_O}{\partial t} = 0 \tag{4.88}$$

where we have used the equivalence of the spins in this specific case to eliminate the spatial variable. That is, the subset of dipolar flip-flops that are allowed here are incapable of equilibrating uniform polarization of any kind to any other state that is more preferred on the basis of enthalpy or entropy, standing in stark contrast to the effect of the sum of all flip-flops considered in previous sections. However, before we conclude that each polarization is its own thermal reservoir let us consider a gradient in one of the polarizations.

2. Spin Diffusion in the Presence of a Well Resolved Quadrupole Splitting for a System Initially in a State of Nonuniform Polarization

Let us consider for a moment what would happen if we introduced a Zeeman polarization difference between the two nuclei considered above when there is no quadrupolar or octapolar polarization. Let the spin at position $x+a$ be in a state of pure Zeeman polarization, and let the spin at position x be completely depolarized. Again, at a short time Δt later, we may then write

$$P_Z(x,\Delta t) = \tfrac{17}{10} W \Delta t P_Z(x+a,0) \tag{4.89}$$

$$P_Q(x,\Delta t) = 0 \tag{4.90}$$

$$P_O(\Delta t) = -\tfrac{3}{5} W \Delta t P_Z(x+a,0) \tag{4.91}$$

We may then extend this to

$$\begin{aligned} P_Z(x,2\Delta t) = P_Z(x,\Delta t) + \tfrac{17}{10} W \Delta t \left(P_Z(x+a,\Delta t) - P_Z(x,\Delta t)\right) \\ - \tfrac{3}{5} W \Delta t \left(P_O(x+a,\Delta t) - P_O(x,\Delta t)\right) \end{aligned} \tag{4.92}$$

$$P_Q(x,2\Delta t) = 9W\Delta t \left(P_O(x,\Delta t) P_Z(x+a,\Delta t) - P_O(x+a,\Delta t) P_Z(x,\Delta t)\right) \tag{4.93}$$

$$\begin{aligned} P_O(x,2\Delta t) = P_O(x,\Delta t) + \tfrac{54}{5} W \Delta t \left(P_O(x+a,\Delta t) - P_O(x,\Delta t)\right) \\ - \tfrac{3}{5} W \Delta t \left(P_Z(x+a,\Delta t) - P_Z(x,\Delta t)\right) \end{aligned} \tag{4.94}$$

We see that the second term of equation (4.83) has now been activated, leading to quadrupolar polarization. This is assured by the equivalent magnitude and reversed signs of the octapolar polarizations for the two spins. Thus, we will again have an induction of

quadrupolar and octapolar polarization by pure Zeeman polarization on the neighboring spins.

Now, let us examine what would happen if we introduced a quadrupolar polarization difference between the two nuclei considered above instead of a Zeeman polarization difference. Let the spin at position $x+a$ be in a state of pure quadrupolar polarization, and let the spin at position x be completely depolarized while the quadrupolar and octapolar polarizations are both still zero. Again, at a short time Δt later, we may then write

$$P_Z(x,\Delta t)=0 \tag{4.95}$$

$$P_Q(x,\Delta t)=\tfrac{9}{2}W\Delta t P_Q(x+a,0) \tag{4.96}$$

$$P_O(\Delta t)=0 \tag{4.97}$$

We may then extend this to

$$P_Z(x,2\Delta t)=0 \tag{4.98}$$

$$P_Q(x,2\Delta t)=\tfrac{9}{2}W\Delta t\left(P_Q(x+a,\Delta t)-P_Q(x,\Delta t)\right) \tag{4.99}$$

$$P_O(x,2\Delta t)=0 \tag{4.100}$$

We see that the quadrupolar polarization will not leak out to the Zeeman and octapolar polarizations precisely because they are initially zero. That is, if either the Zeeman or octapolar polarization is simply nonzero while the quadrupolar polarization is spatially inhomogeneous, both the Zeeman and octapolar polarizations will become coupled to the

quadrupolar polarization, thereby participating in the diffusion of spin order. Indeed, these equations show that it is practically impossible to maintain a system of pure quadrupolar polarization if gradients exist in that polarization since any small fluctuation in the Zeeman or octapolar polarizations about their zero value would quickly grow to an equilibration of all the three polarizations. Thus, in the absence of other polarizations quadrupolar order is an absolutely separate thermal reservoir, but it is in a position of unstable equilibrium like a pencil standing on its pointed end. It will not persist in this state, and will achieve thermal equilibrium on a rapid time scale.

H. Frozen Core Effects

There are two more ways in which the large electric fields and Knight shifts close to the center of a shallow hydrogenic donor can perturb with the process of probing these sites with an ONMR experiment. Firstly, the quadrupole interactions can be so strong as to interfere with the ONP of the nuclear spin. Secondly, the quadrupole and Knight shifts in the nuclear spin Hamiltonian can effectively truncate the homonuclear dipolar flip-flops. Both of these processes can be roughly approximated by application of the results of a single spin-one-half in the presence of an off-diagonal interaction. Following Cohen-Tannoudji et al.[4], one may write

$$\mathscr{H} = \begin{pmatrix} E_1 & w \\ w & E_2 \end{pmatrix}. \tag{4.101}$$

Moreover, if one calls the eigenstates of the unperturbed Hamiltonian $|\alpha\rangle$ and $|\beta\rangle$, then the new eigenstates are

$$|+\rangle = \cos\frac{\theta}{2}|\alpha\rangle + \sin\frac{\theta}{2}|\beta\rangle \tag{4.102}$$

and

$$|-\rangle = \cos\frac{\theta}{2}|\beta\rangle - \sin\frac{\theta}{2}|\alpha\rangle \tag{4.103}$$

where

$$\tan\theta = \frac{2w}{E_1 - E_2}. \tag{4.104}$$

With respect to ONP we take α and β as two adjacent states in the ladder of spin levels, and the w as the off-diagonal terms that tends to inhibit the transition between these two levels. Clearly the matrix element $\langle +|I_+|-\rangle$ captures this behavior. That is, when w is zero, the matrix element is one and ONP is unimpeded. However, when w is not negligible

$$\langle +|I_+|-\rangle = \left(\cos\frac{\theta}{2}\right)^2 - \left(\sin\frac{\theta}{2}\right)^2 = \cos\theta\,. \tag{4.105}$$

Moreover, it is clear from this that when w gets very much larger than $E_1 - E_2$ θ tends to $\pi/2$ and the matrix element $\langle +|I_+|-\rangle$ goes to zero, precisely as one would expect.

With respect to spin diffusion, α and β are taken to be middle states in a product operator basis set. That is, $|\alpha\rangle = |\gamma\delta\rangle$ and $|\beta\rangle = |\delta\gamma\rangle$, so that

$$|+\rangle = \cos\frac{\theta}{2}|\gamma\delta\rangle + \sin\frac{\theta}{2}|\delta\gamma\rangle \tag{4.106}$$

and

$$|-\rangle = \cos\frac{\theta}{2}|\delta\gamma\rangle - \sin\frac{\theta}{2}|\gamma\delta\rangle . \tag{4.107}$$

Inverting these equations gives

$$|\gamma\delta\rangle = \cos\frac{\theta}{2}|+\rangle - \sin\frac{\theta}{2}|-\rangle \tag{4.108}$$

and

$$|\delta\gamma\rangle = \cos\frac{\theta}{2}|-\rangle + \sin\frac{\theta}{2}|+\rangle \tag{4.109}$$

Using the orthogonality of the states $|+\rangle$ and $|-\rangle$, the matrix element of the dipolar flip-flops that give rise to spin diffusion can now be calculated as

$$\langle\gamma\delta|\delta\gamma\rangle = 2\cos\frac{\theta}{2}\sin\frac{\theta}{2} = \sin\theta .$$

Thus, when $E_1 - E_2$, in the form of Knight shift and quadrupole interaction differences, is very much larger than *w*, which is a flip-flop term in the homonuclear dipolar Hamiltonian, θ will go to zero, and so does the spin diffusion rate. Conversely, when the two spins have negligible Knight shift and/or quadrupole differences, θ goes to $\pi/2$ and the spin diffusion rate goes to the value that was calculated earlier in this section.

I. Conclusions

Spin diffusion has been addressed in more detail and more accurately here than in virtually any other work in the last 40 years. We have proved that the standard computational approach presented in Abragam[2].

It should be clear by now that, without theoretical foresight and the resultant experimental intervention, the results of the experimental determination of the electronic wavefunction at the optically relevant defect would be convoluted with spin diffusion to the point where they may very well be inseparable. The experimental intervention is relatively painless. Simply decouple the nuclei with any of a variety of simple dipolar decoupling multiple pulse sequences during the optical nuclear polarization. However, hindsight is usually 20/20, and this experiment has yet to be done.

However, we have shown that there is strong theoretical and experimental evidence for fast spin diffusion in GaAs. Moreover, the non-ideal satellite ratio observed in the presence of a uniform quadrupole Hamiltonian strongly suggests that, if the optical nuclear polarization process attempts to continuously support a single Zeeman spin temperature, the sample polarization is largely due to ONP at sites near the defect whose polarization has diffused outward.

However, it is interesting to note that, when we have had quadruple splitting of the transitions, we have observed spectra with about 40% more central transition than expected from Zeeman polarization, and, simultaneously, the satellite lines were asymmetric in a way that could not be fit by a single Zeeman spin temperature.

J. References

1) Mehring, M. *High Resolution NMR in solids*; Second, Revised and Enlarged ed.; Springer-Verlag: Berlin, 1983.

2) Abragam *The Principles of Nuclear Magnetism*; Oxford University Press: Oxford, 1961.

3) McArthur, D. A.; Hahn, E. L.; Walstedt, R. E. *Physical Review* **1969**, *188*, 609-638.

4) Cohen-Tannoudji, C.; Diu, B.; Laloe, F. *Quantum Mechanics*; John Wiley & Sons: New York, 1977.

V. Experiments on a Heterojunction - Optically Relevant Defects

The theoretical foundation presented in chapters 3 and 4 of this thesis has been developed to allow us to perform calculations simulating various experimental results. We now consider those experiments, their results, and the simulation models for those experiments. We shall also put forth suggestions for more accurate simulations.

The first LBD experiment completed, beyond the proof of the LBD effect, was to record a free-induction-decay, FID, of a single nucleus using LBD and to Fourier transform this FID. This is commonly referred to simply as an FT experiment, and consists of only pulses to the signal and reference nuclei, followed either by real-time LBD, or by an incremented period of free evolution and dual spin-locked LBD (mapping out the FID point by point). The position of the peak of the resonance curve yields the position of resonance relative to the rf frequency of the pulse. In the FT experiment the best estimation of what pulse duration will make the pulses into 90°-degree pulse is made, but, once the rf-frequency and/or the magnetic field is set so that the resonance matches the rf frequency, one is then able to set the 90°-degree pulse exactly by performing a nutation experiment. This is accomplished by measuring the transverse magnetization after a single pulse whose length is incremented. Again, this is a point-wise experiment, and it yields the frequency of the signal nucleus rotating about the rf field. This, consequently, provides the size of the rf field as well as the best pulse times. The FT experiment suffers from additional width in the spectral line due to inhomogeneity in first

rank and pseudo first rank tensor interactions of the nuclei, primarily inhomogeneity of the Zeeman field and heteronuclear dipolar couplings. This can be eliminated using the spin-echo (Hahn echo) experiment, illustrated in figure 5.1 below. This is essentially a point-wise FT experiment with a 180°-pulse half way through the free evolution period, resulting in a refocusing of all interactions that are inverted by the 180°-pulse at the detection time.

The Hahn echo is a sensitive probe of the magnitude of the nuclear magnetization. It has been utilized here to measure the rise of the polarization with the light on, and its dissipation with the light off. It also serves as a basis for obtaining a low resolution radially resolved Knight shift image of the signal nuclei when it or a modification to it, the stimulated echo, is synchronized with light in between the pulses that define one of the evolution periods. Moreover, a high resolution radially resolved Knight shift image of the signal nuclei is also obtained using the CLSW-16 pulse sequence[1-5]. A full dissection of the Knight Shift image into its constituent physical underpinnings is highly valuable. Putatively, one can learn the shape and size the electronic orbital in which the electron is trapped at the defect, the occupancy of that electronic orbital, and about quadrupolar interactions in the vicinity of the defect. We can easily obtain all or most of this information if spin diffusion is nonexistent or if it is so fast as to produce uniform polarization over the area of the electronic wavefunction even at the shortest ONP time; however, if it is in the intermediate region between these extremes, the spin diffusion will require its own set of calculations in order to deconvolute the from the Knight shift image.

Two different theoretical approaches where used in analysis of this work. They are somewhat polar opposites in their methodology. In the first technique, a full, real-time analysis of each and every one of the hundreds of thousands of nuclei surrounding a defect in GaAs has been accomplished and the results of computations with it will be presented here. This was achieved using a non-interacting nucleus model. That is, the time domain is computed for each and every individual nucleus including its Knight shift, quadrupolar interactions (both secular and nonsecular), individual optical polarization conditions, optical detection weighting, and rigorously exact rf effects, for a time of order of magnitude of a million cycles of the relevant rf frequency. This can be accomplished precisely because it is a summation of a large number of single nucleus calculations. If one were to require that the nuclei couple to one another, shifting resonances and redistributing polarization, then the computation would very rapidly become unworkably large. First, resonance shifts would involve increasing the number of atoms being considered during the quantum mechanical calculations, such as diagonalization, from 1 to at least 16 (keeping only nearest neighbor Ga and As atoms), requiring us to move from diagonalizing 4x4 matrices to diagonalizing 65536x65536 matrices. This matrix size is computationally intensive, yet it is still quite tractable by itself. However, our exact calculations require enormous numbers of such computations for each and every nucleus since we consider many time steps in the cycle of the rf wave, and do this computation for each and every laboratory-frame phase that the rf takes on. This clearly places such large matrix diagonalizations well beyond the level of computational complexity that is tractable for these calculations. The Mathematica code for this simulation and other less complicated experiments is located in appendix A of this thesis.

Moreover, the explicit computation of spin diffusion in three dimensions at every nuclear site during each portion of the experiment (some of them lasting for seconds) hits a similar catastrophically rapid increase in the computational complexity.

As such, using this approach, we are only able to compute spin diffusion effects when they are present or absent to such an extreme extent that they allow us to make certain blanket assumptions that apply to each individual computation. For example, if spin diffusion can be considered infinitely fast, then every nucleus polarizes at the same rate since spin diffusion redistributes polarization among the nuclei faster than they can accumulate it individually, leading the polarization as a function of time towards a single exponential functional form. In fact, the experiments presented in this chapter were exhaustively fit with every manor of single nucleus calculation, representing the asymptotic limits of the multi-nucleus computations, and no single nucleus calculation came reasonably close to fitting the spectra over the entire 5 second range of ONP times.

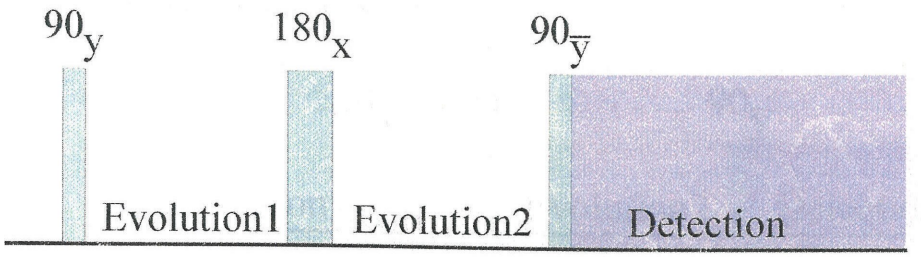
In the second technique, the GaAs crystal is reduced to a continuous dielectric medium. There are no matrices, no rf effects, and no discrete lattice. However, spin diffusion during ONP can be approximately calculated from a 1D radial integration of the relevant equations over time. This produces the best fits to the experimental data. That is, the very high speed of spin diffusion makes any analysis that ignores it incapable of reproducing a large fraction of the physics in the experiments. Moreover, estimation of the approximate behavior of the system in the presence of the physics that gives rise to frozen core effects is still possible with these computations while various approximations the anisotropy of both the lattice and the dipolar couplings can also be included here.

What one is left with is a collection of approximations that capture most of the relevant physics of the experiment rather than a collection of exact calculations that misses a large portion of the physics of the experiments. The best results that can be obtained in the near term are highly likely to be generated as follows. An approximate radial polarization distribution would be produced using this 1D integration, and that radial polarization distribution would be used as an input into an exact "sum over all individual spins" calculation like the one in appendix A of this thesis.

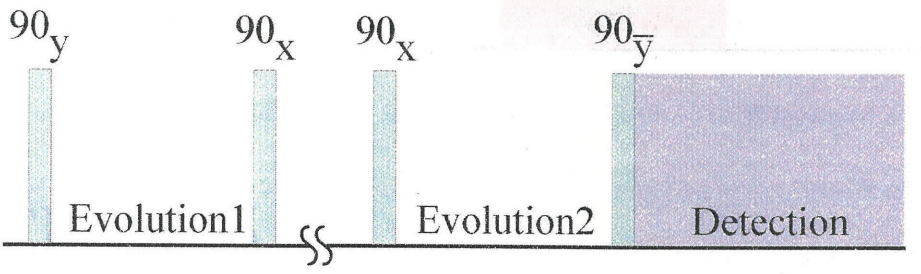
A. Experiments

The most important of the NMR pulse sequences used in this work are shown in figure 5.1 below. The common spin echo experiment, long employed to remove unwanted line broadening effects, functions essentially by inverting the magnetization after the free evolution period titled Evolution1 in the figure. This assures that the free evolution of the spin system in the period titled Evolution2 will occur in the opposite sense as that in the period titled Evolution1 for all first rank tensor interactions, when viewed in the rotating reference frame of the nuclear spins. This leads to a cancellation of evolution due to such first rank tensor interactions over the entire evolution period (Evolution1 and Evolution2). Essentially, this leaves the spin system with only evolution due to higher order tensor interactions at the end of the evolution period.

a. Spin Echo



b. Stimulated Echo



c. CLSW16

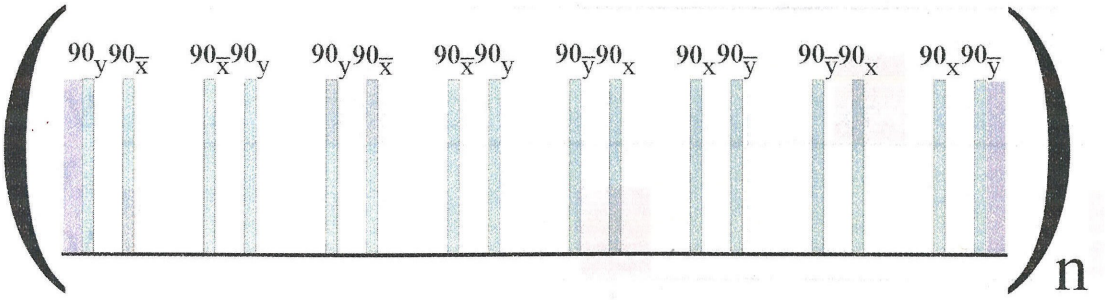
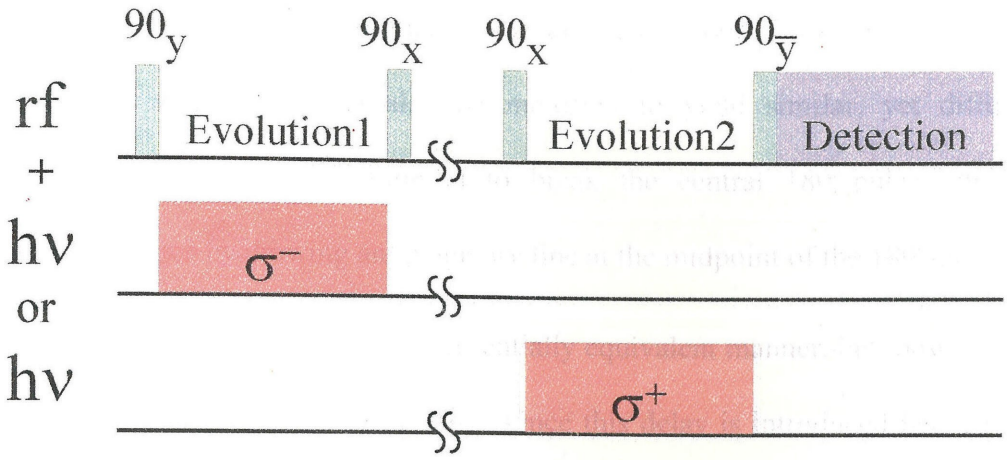


Figure 5.1 Pulse sequences of NMR experiments used in this work: a) the spin echo (Hahn echo) is used to remove first rank tensor interactions, b) the stimulated echo is, approximately, a spin echo cleaved in half with the two halves separated by a delay, c) the CLSW16 sequence is a higher order line narrowing sequence that removes virtually all NMR interactions. The pale-purple area represents the detection period in each experiment.

a. Stimulated Echo



b. CLSW16

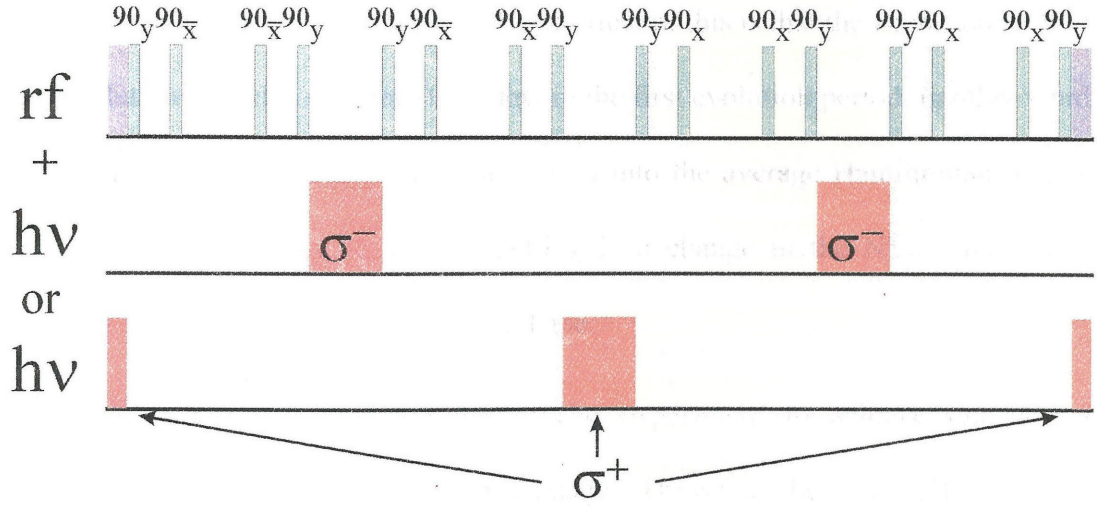


Figure 5.2 Synchronization of the laser light excitation to the NMR pulse sequence allows the nuclear spins to experience the frequency shifting influence of the hyperfine contact with spin polarized electrons for a select fraction of the cycle over which they are being averaged, assuring that this interaction will not be averaged away.

The spin echo experiment can be used in a variety of ways. For example, one may vary one of the parameters of the ONP period such as its duration or the intensity of the laser light, and perform the experiment once at each chosen value of the variable parameter. This experiment can also be modified to yield similar, yet different, experiments. One such modification is to break the central 180°-pulse into two consecutive 90°-pulses by drawing an imaginary line at the midpoint of the 180°-pulse.

The sequence still functions in an essentially equivalent manner, but, now, a delay can be inserted between the two 90°-pulses. Once this delay is introduced between the two halves of the former 180°-pulse, the experiment is called a stimulated echo. The delay allows a parameter, such as the Knight shift, that would have been refocused by the echo (due to its odd order symmetry under the influence of the "180°-pulse") to be introduced or altered during the delay. The net effect of this is that the Hamiltonian in the second evolution period is different from that in the first evolution period, in other words, a selective reintroduction a first rank interaction into the average Hamiltonian over the entire evolution period has occurred, resulting in a change in the NMR signal while maintaining a narrow underlying NMR lineshape.

Initially, we used this stimulated echo experiment to achieve our goal of measuring a radially resolved Knight shift image. However, there is a different, more accurate technique for achieving this goal. This is the CLSW16 pulse sequence, a high order multiple pulse line narrowing sequence capable of removing all NMR interactions regardless of their tensor rank in an average sense[1-5]. Pulse sequences such as CLSW-16 are referred to as time suspension sequences because, with all interactions removed from

the average evolution, time appears to stand still. The price paid for such spectacular performance is that one must observe the system stroboscopically. That is the spin system can only be monitored at the exact same point of the sequence on each cycle of the sequence for this averaged " time suspension" to be observed. An additional caveat is that there is a maximum size of interaction that can be averaged. This is of order of magnitude of the inverse of the time for one complete cycle of the pulse sequence, which is ~8.3 kHz in the present work. Interactions larger than this will be partially averaged at a level commensurate with the magnitude of the interaction.

Just as the Knight shift is averaged away by a stimulated echo experiment unless it was modulated during the course of the experiment, we must similarly modulate the laser light (the supply of polarized, excited-state electrons) so that it is present only in specified periods of the CLSW-16 experiment. The relevant modulation scheme for each of these experiments is illustrated in figure 5.2 above. Note that in each case there is a choice of two different schemes that produce the same average Hamiltonian. One may also change the sign of the circular polarization of the light in either of these schemes to produce an average Hamiltonian with the exact same magnitude of Knight shift yet introduced with the opposite sign.

B. Results and Discussion

A representative set of the experiments that I have conducted, contributed to, and/or provided theoretical models for will now be presented. These range from the characterization of the rise of the polarization from ONP to the imaging of the electronic

wavefunction of the electron trapped at the ORD. There are additional experiments that the could be presented and dissected using the 1D radial spin diffusion integration. First amongst these is a spin echo experiment from the days when a resolved quadrupole splitting was observed due to sample strain. It posses anomalous satellite to central transition intensity ratios, and application of theory in chapter 4 that shows that each type of polarization diffuses with a different rate could potentially explain this. Unfortunately, there is only so much that can be accomplished before it is time to move on and graduate, so this analysis waits for newer students in the Weitekamp lab, should they decide to take it on.

1. Initial Characterization of ONP Dynamics

Detailed experiments probing the ONP dynamics about the ORD as a function of ONP time and laser output power have been accomplished along with the a determination of the dependence of the signal intensity on the laser output power during the detection period. Without spin diffusion, the characterization of the amount of nuclear polarization generated as a function of ONP time yields insight into the correlation time for electron hyperfine field experienced by the nuclei (assuming a given donor Bohr radius) as well as some indications about the spatial distribution of the hyperfine interaction. With spin diffusion, the analytical fit to a simple functional form is replaced by a numerical calculation that takes hours to days, depending on the complexity of the simulation model and the speed of the computer carrying out the computations.

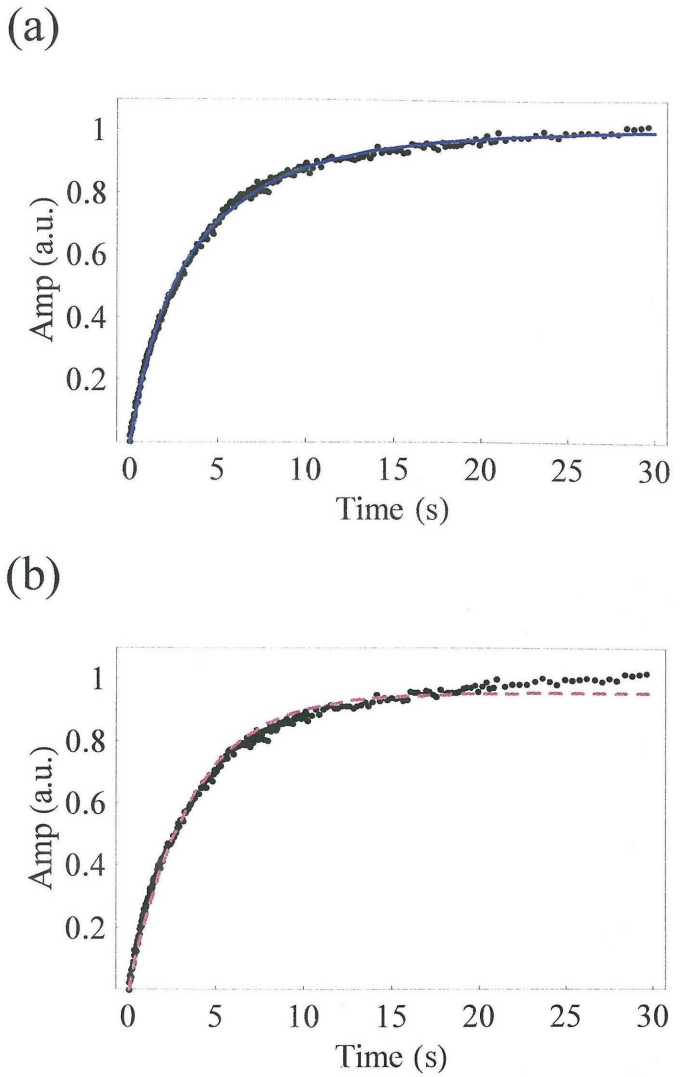


Figure 5.3 Two different fits to the time dependence of the optical nuclear polarization. (a) The solid blue curve is a continuous dielectric medium approximation to a shallow donor. (b) The dashed violet curve is a single exponential. The chi-squared for the blue curve is a factor of ~5.6 lower than the violet curve. The fit values for the blue curve are $T_{1on}(0) = 430$ ms and $T_{1off} = 11$ s while those for the violet curve are $T_{1on} = 3.5$ s and $T_{1off} = 6$ s.

Ignoring spin diffusion in the process of fitting data acquired with ONP times ranging from 0 to 30 seconds with laser output power at 32 mW produces the upper fit (blue line) in figure 5.3 while assuming that it is so fast as to produce one T_{1on} for all nuclei produces the lower fit (red line). Both fits accounted for a uniform background relaxation. Notice that the shallow donor with quasi-uniform polarization model does not fit the data as well as the shallow donor without spin diffusion model.

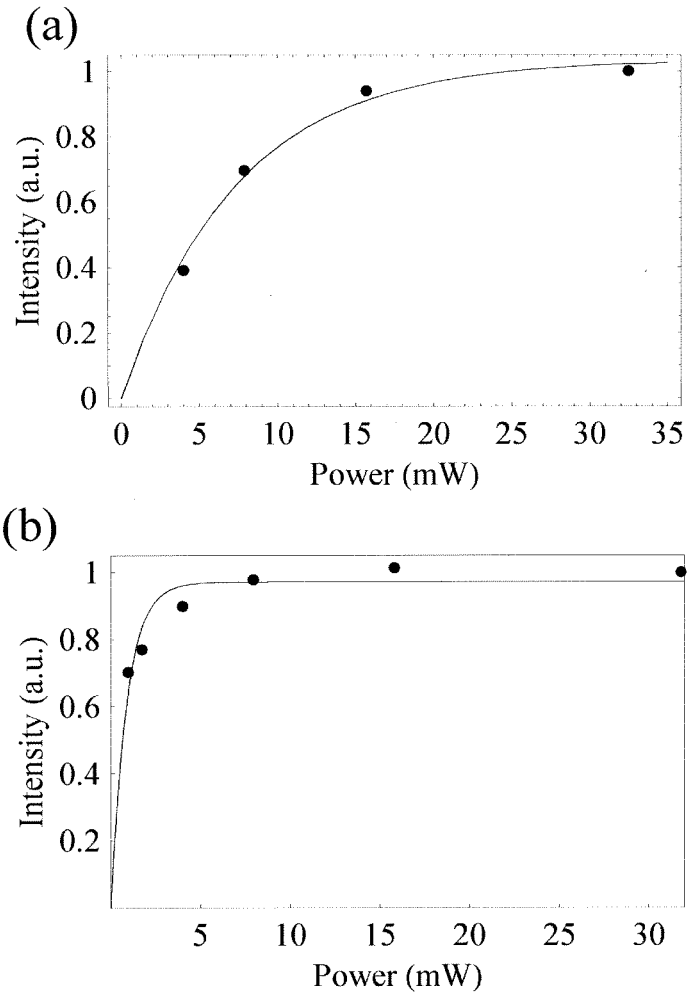


Figure 5.4 (a) The dependence of the signal intensity due to variation of the laser output for both the ONP and detection periods simultaneously. (b) The dependence of the signal intensity due to variation during only the detection period. Full occupancy of the sites that are being optically detected is achieved with ~10 mW of laser output power. The laser output power can be converted into power density by dividing by ~0.001 cm^2 (i.e., the diameter of the laser spot on the sample is ~350 μm).

Unfortunately, both models give progressively shorter T_{1on} times as they are applied to progressively truncated subsets of the data where only the earlier time data is retained. This is an indication that neither model captures all of the relevant physics in the data. For example, an analysis of the asymptotic limit of these shorter time fits to zero time yields a 16 ms value of $T_{1on}(r=0)$[5], implying a correlation time of the order of 10^{-7} s, about an order of magnitude longer than is expected for any variety of shallow defect (i.e., Bohr radius on the order of 10 nm). This is evidence of

importance of spin diffusion at all timescales relevant to ONP and detection. Further evidence of the importance of spin diffusion will be encountered during the analysis of the Knight shift imaging experiments below.

The dependence of the signal intensity as a function of varying the laser output power across the entire experiment as well as the dependence of varying it only during the detection period is presented in figure 5.4, showing that the signal intensity saturates well below the 32 mW maximum output power of the laser. Indeed, subsequent experiments prove that the signal intensity saturates at ~11 mW for both excitation and detection[5].

2. Polarization Decay in the Absence of Laser Illumination

With the intent of measuring the background relaxation of the nuclear polarization during the experiment, we have measured the dissipation of the nuclear polarization when the laser light is turned off. To this purpose, a set of experiments where ONP was carried out for a specified constant period of time that was followed by a variable delay where the laser light was turned off, and completed by optically detecting (via LBD) the remaining detectable nuclear polarization. The signal intensity, being a measure of the remaining nuclear polarization, for each of these experiments is plotted versus the delay time between the ONP and detection periods in figure 5.5 below.

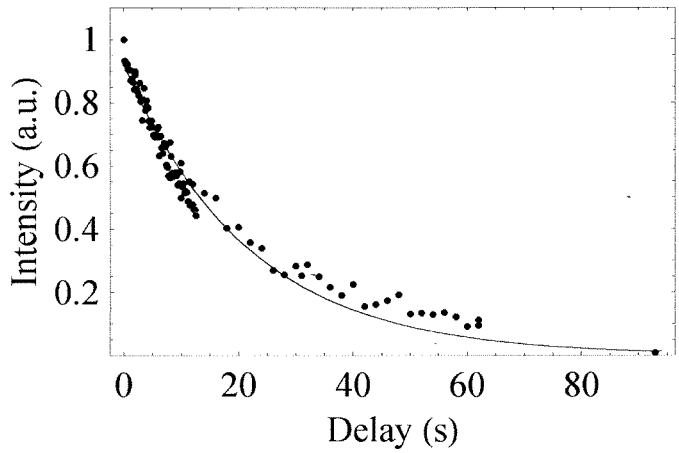


Figure 5.5 Signal intensity remaining after a light-off delay between ONP and LBD detection. The time constant is ~21 s.

The best-fit single exponential for this data is found to have a decay constant T_{1dark} that is ~21 s, but the single exponential is a less than optimal fit. It drops too slowly at early times and too rapidly at longer times. A better way to model the system would likely be to allow for both spin diffusion and a T_{1off} relaxation that is either uniform or spatially dependent (such as the fluctuations in transitions between 1s, 2s and 2p orbitals of the hydrogenic donor calculated in chapter 3).

3. Deconvolution of the Synchronized rf-Optical Simulated Echo – The First Radially Resolved Knight Shift Image

The results of the optically synchronized stimulated echo that is illustrated in figure 5.2, and described in the text of the earlier sections of this chapter are shown in figure 5.6, and will now be discussed. Additionally, a method of extracting a low-resolution radially resolved Knight shift image from this data is also presented.

The "light-on" stimulated echo signal may be viewed as the inverse Fourier transform of the weighted distribution of Knight shifts around a donor that has been line

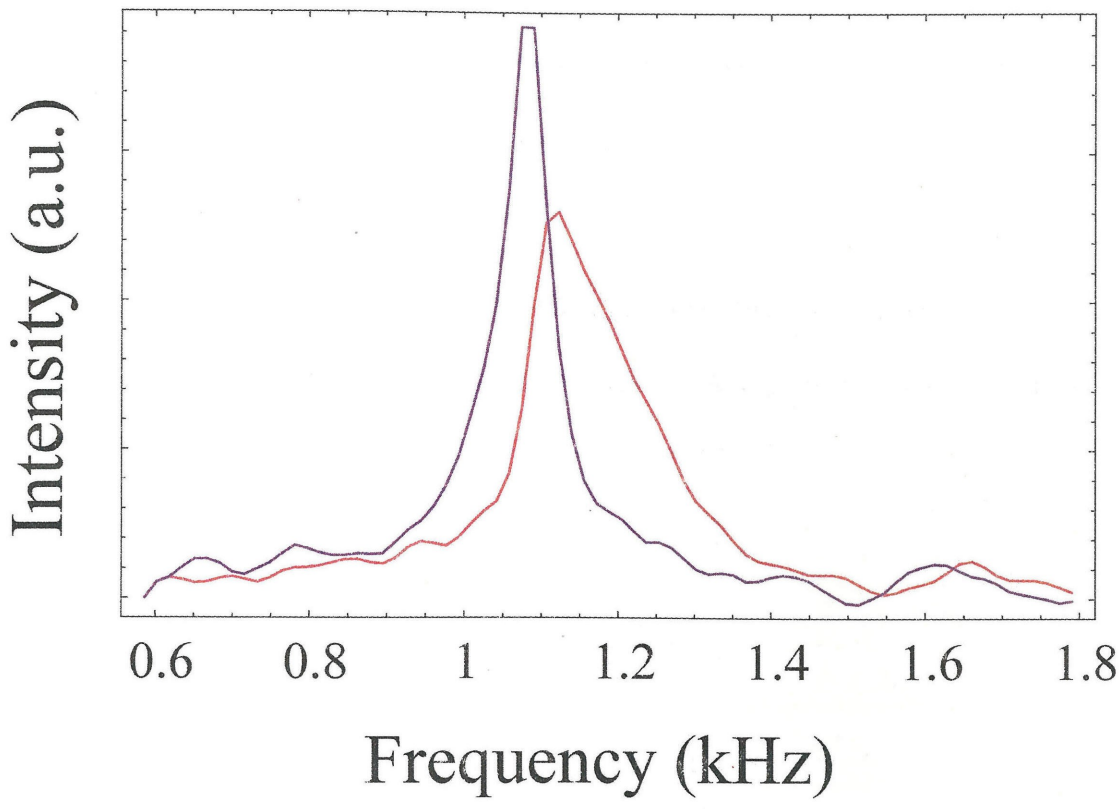


Figure 5.6 ^{71}Ga ONMR spectrum obtained using a stimulated echo without optical pulses in any of the windows (purple line) and with σ_+ laser light on in the second evolution period (red line). Both experiments had 5 s of ONP, and were detected using ^{75}As as the reference field. The resonance line is both shifted and asymmetrically broadened by the distribution of hyperfine interactions experienced by different nuclei.

broadened by the presence of homonuclear dipolar and scalar couplings. However, the homonuclear dipolar and scalar couplings are what broaden the "light off" stimulated echo signal, giving it its lineshape. Moreover, in the time domain, line broadening is simply the multiplication of the time domain of the unperturbed spectrum and the time domain of the lineshape of the interaction leading to the line broadening (the well known convolution theorem[6]). This may be inverted to say that extraction of the unbroadened

lineshape of an interaction can be accomplished by dividing its line broadened time domain by the time domain of the line broadening mechanism. This correlates to dividing the time domain of the light-on stimulated echo by the time domain of the light-off stimulated echo.

The one practical issue that arises when pursuing this is that there is no imaginary channel to the signals collected in the lab; therefore, zeroes in the time domain of the

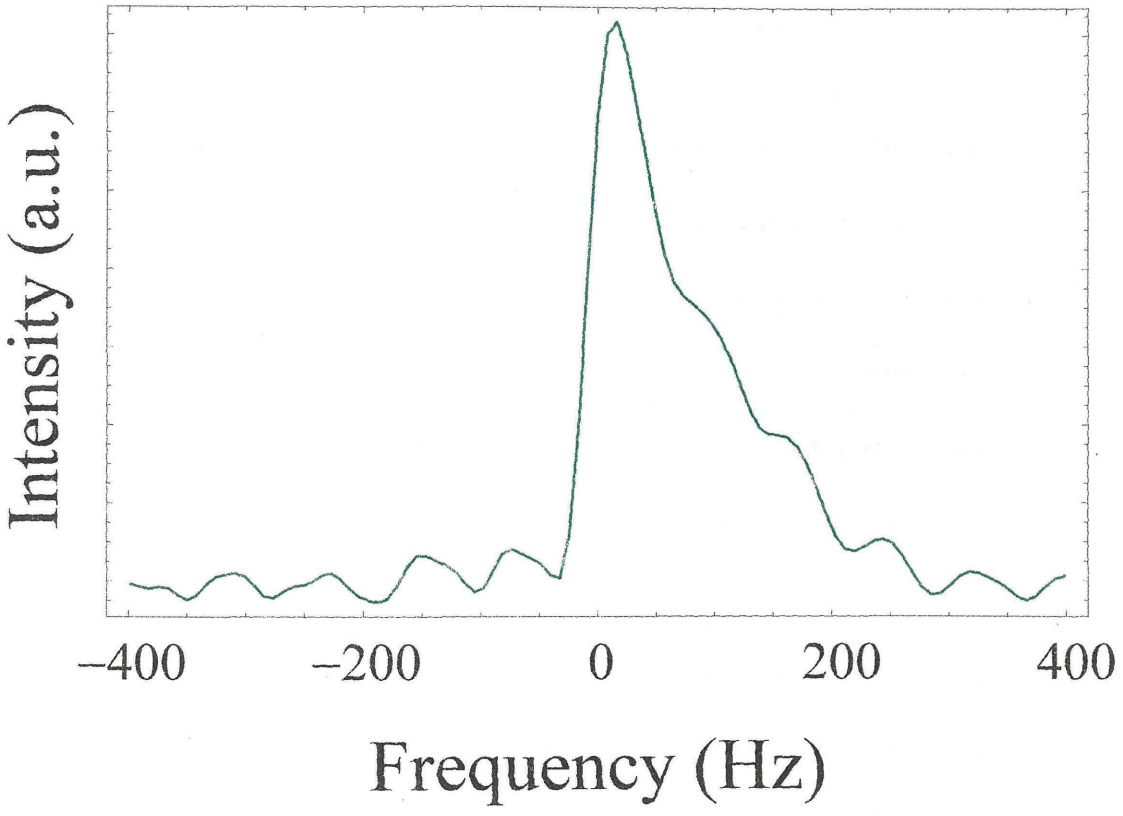


Figure 5.7 Deconvolution of the distribution of hyperfine interactions experienced by different nuclei from the light-on ONMR stimulated echo spectrum by utilizing the light off spectrum as a reference (dividing it out in the time domain). This distribution of hyperfine couplings is a low-resolution radially resolved Knight shift image.

light-off signal produce a quotient that diverges to infinity and is numerically unstable. The way to fix this is to separately Fourier transform both the light-on and light-off signals, leading to the symmetric spectra that are guaranteed by a real time domain. Then, one sets the branch of the spectrum that contains the reflection of the spectral feature of interest to zero. For example, if the spectral line really lies in the range of frequencies below zero, then the amplitude of the frequencies above zero are reset to zero. These spectra are then returned to the time domain via inverse Fourier transform where they can be divided, and, finally, the time domain this produces is Fourier transformed into the frequency domain.

This has been accomplished for both of the spectra in figure 5.7, and the result has been plotted in figure 5.7, showing that, indeed, one does extract a low resolution radially resolved image of the Knight shift distribution. This result can be compared to the 5 s ONP data in the CLSW-16 section below. The peak appears smaller and narrower because the signal to noise is much worse in the present case. That is, the noise in the light on and light off experiments is random, so division of the two signals produces a signal that is limited by the noise in these experiments and not by the noise in an experiment that really did eliminate the broadening interaction altogether. Moreover, the knight shift is spread out over the noise in the spectrum to a greater extent due to the increased duty cycle (one half here compared to one tenth for CLSW-16), burying the high frequency tail of the spectrum under the noise.

4. Optically Synchronized CLSW-16 – The High Resolution, Radially Resolved Knight Shift Image

The results of CLSW-16 experiments and light synchronized CLSW-16 experiments will now be presented and discussed along with some simulations of the physics that gives rise to their lineshapes and relative intensities. The experimental

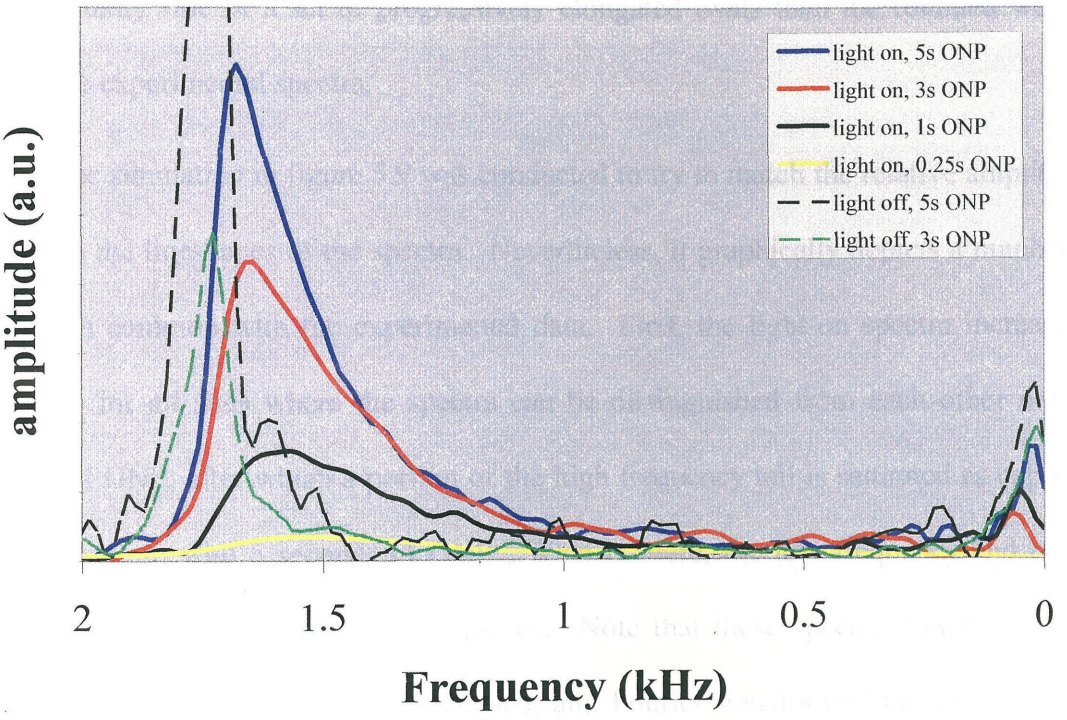


Figure 5.8 Experimental data from CLSW-16 Knight shift imaging experiments (light-on) with various ONP times and from CLSW-16 experiments without optical pulses (light-off). The excitation power was 32 mW (well over the saturation value), the magnetic field was longitudinal (001-direction), σ_- was detected at the LBD beat frequency, the sample temperature was ~2 K, and the CLSW-16 cycle time was 120 μs while the 71Ga pulse time ($\pi/2$) was 3 μs.

results are shown in figure 5.8, and the results of an exact "sum over all sites independently" calculation (developed in chapter 3 and presented in *Mathematica* code format in Appendix A following this chapter) are given in figure 5.9, illustrating the fundamental disconnect in this sort of approach between fitting the lineshape of a spectrum at a given ONP time and getting the relative amplitudes of the signals a different ONP times correct. That is, these lineshapes are far too rounded, looking more like the pointy side of a set of progressively elongated ovals than the rounded wedges seen in the experimental spectra.

The simulation in figure 5.9 was conducted to try to match the relative amplitudes more than the lineshapes of the spectra. Nevertheless, it graphically depicts a number of features in common with the experimental data. First, the light-on spectra increase in amplitude for all sites where the spectra can be distinguished from each other until 3 seconds of ONP, after which a portion of the high frequency tail is saturated as captured by the spectrum with 5 seconds of ONP. What is more, the low frequency end of the spectrum does not saturate at any frequency. Note that these spectra should really be inverse Fourier transformed, line broadened, and Fourier transformed to match the ~40 Hz of line broadening still exhibited by the light-off spectrum.

One may attempt to fit the lineshapes first; however, the relative intensities between the different ONP times become smaller and smaller (3 and 5 seconds lay almost directly on one another). Therefore, what is needed is a mechanism whereby the intensity at lower frequencies (higher radial distance from the donor) can be preferentially increased over intensity at higher frequencies in a way that matches the experiment. Spin diffusion answers this description and is expected to be too strong to ignore from

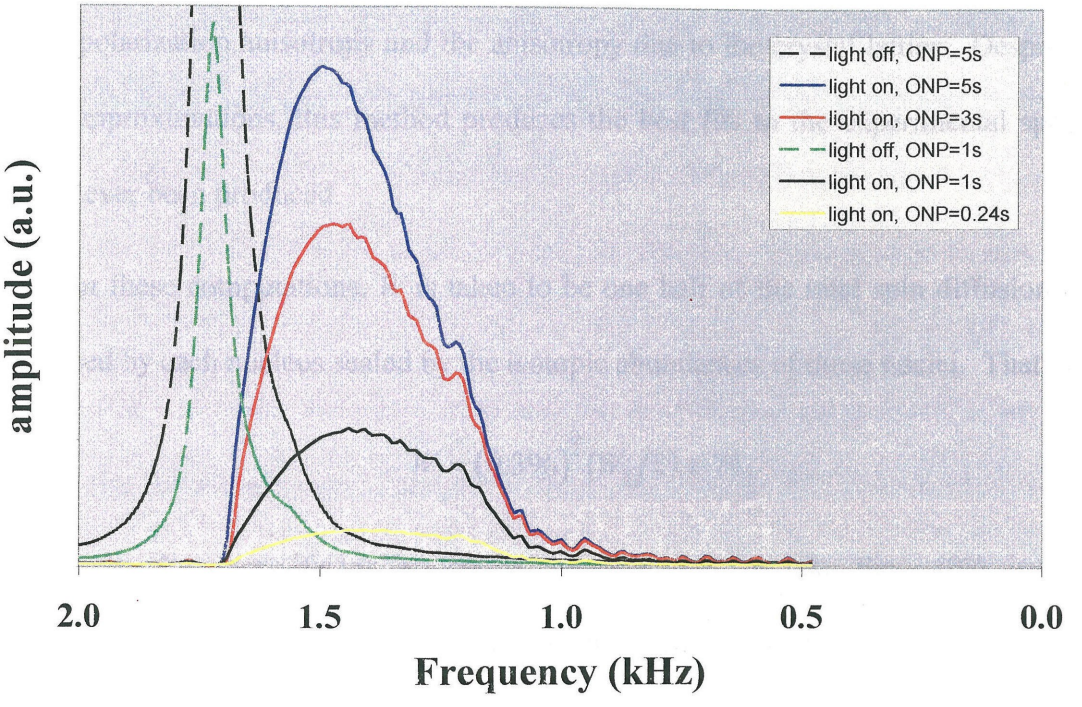


Figure 5.9 Results of the sum of single spin simulations approach to modeling that was presented in chapter 3 as applied to simulating the experimental data in figure 5.6. While these simulations can exactly replicate features of the light off data and the general trends in the light-on spectra, they are incapable of producing the exact line-shape of the light-on spectra for all ONP times with one set of parameters. For these calculations $T_{1on}(0) = 280$ ms, $\langle s \rangle = 0.06$

calculations presented in chapter 4. The full 3D spin diffusion problem would require a faster computer and more memory than is currently available (~3 GB of RAM, and a computer fast enough to rapidly deal with such large amounts of data), yet it is possible to do computations based solely on a radial line of spins that extends from the center of the donor outward. This is an approximation, lacking the irreversibility of a spin at the center sharing its polarization with a much larger volume of spins simultaneously through different routes. It also lacks a rigorous handling of the dipolar anisotropy that gives rise to radial polarization anisotropy and the anisotropy due to the crystal lattice. Despite all of these approximations, this method produces the best fits to the experimental spectra that have ever been produced.

For these computations, W is taken to be one half of the total spin diffusion rate experienced by each nucleus scaled by the isotopic abundances of those nuclei. That is,

$$W = (0.396)^2 (W_\Sigma/2) \cong 74 . \tag{5.1}$$

The isotopic abundance factor in this is the only way to take the ~40% isotopic abundance of ^{71}Ga into account, and maintain a reasonable level of computational complexity. Moreover, the use of the total rate of spin diffusion for a given site divided by two (one half for the forwards reaction and one half for the backwards reaction) is plausible given that, while there are transitions that do not advance or reduce the position of a given piece of nuclear polarization by one full inter-nuclear spacing, there are also transitions that advance or reduce its position by more than one full inter-nuclear spacing, and we are considering W to be a nearest neighbor coupling rate. Thus, spin diffusion will move polarization away from a given nucleus considerably faster than the at just the

rate of the nearest neighbor coupling, and estimation of the effective radius that the sum of the spin diffusion rates acts from on average ($\langle x^2 \Delta\omega^2 \rangle_{^{71}\mathrm{Ga}} / \langle \Delta\omega^2 \rangle_{^{71}\mathrm{Ga}}$ computed for the semicircle where x in is positive) shows that this value is close to one inter-nuclear distance, indicating that the choice of this value as an average is qualitatively correct.

This technique has been applied, at first, without any attempt to address frozen core effects. The best simulation results obtained are for $T_{1\mathrm{on}}(0) = 1$ s and $\langle s \rangle = 0.175$, which are reasonable values, and the results are shown in figure 5.10, overlaid on the experimental data. A crude approximation to the anisotropy of the lattice and the dipolar coupling is also included here. The anisotropy was handled by summing three separate computations, one with $W = 74$ to account for the diffusion of polarization transverse to the magnetic field to the nearest neighbors, one with $W = 74/4$ to account for diffusion to the nearest at 45° to the magnetic field and the next nearest neighbors that are in a direction that is colinear to the magnetic field, and one at $W = 0$ to account of the directions that require many zigzags to reach that are near the ~54.7° null of the dipolar coupling.

The weights of these individual calculations where initially worked out for the case where the third computation was along the z-axis, however this turns out to have half the rate of spin diffusion as the transverse nearest neighbors while the transverse next nearest neighbors ([100] and [010] directions) were found to have $W = 74/4$. This called for reassigning the some of the intensity from the first calculation to the second while

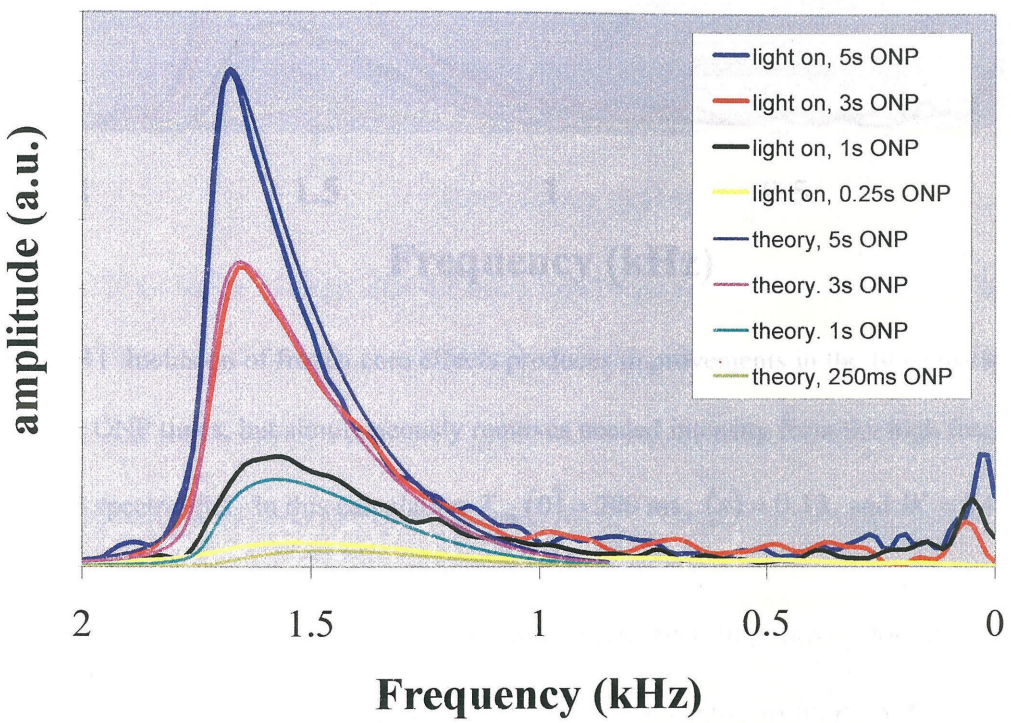


Figure 5.10 A reasonably good simulation of the data can be obtained by integrating the spin diffusion equations for a one dimensional radial line of spins during just the ONP period. This version approximates the 3D case where spin diffusion is uniform at all radii, and ONP is unperturbed by electric field induced quadrupole splitting (no frozen core). It also includes the sum over different directions approximation to the anisotropy of the spin diffusion that arises from the angular dependence of the dipolar coupling that is discussed in the text. For this computation $T_{1on}(0) = 1$ s, $\langle s \rangle = 0.175$, and $W = 74$.

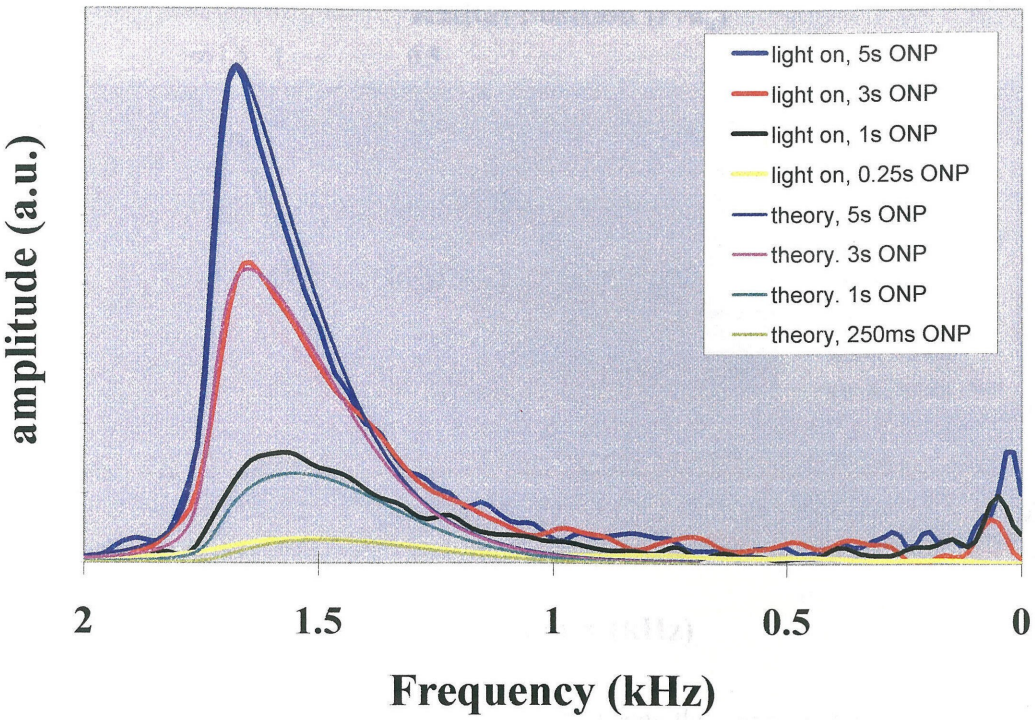


Figure 5.11 Inclusion of frozen core effects produces improvements in the fit of the spectra at shorter ONP times, but simultaneously removes needed intensity from the high frequency tail of the spectral fits. In this calculation $T_{1on}(0) = 200$ ms, $\langle s \rangle = 0.13$, and $W = 74$.

moving the third. In the end, the weights were kept since they have roots in the relative fractions of signal that are available from $0 \le \theta \le \pi$. The weights were obtained by breaking the calculation into the following pieces $0 \le \theta \le \pi/8$ and $7\pi/8 \le \theta \le \pi$, $\pi/8 \le \theta \le 3\pi/8$ and $5\pi/8 \le \theta \le 7\pi/8$, and, finally, $3\pi/8 \le \theta \le 5\pi/8$. The relative intensities of the three calculations obtained by integration are then $0.2:\sqrt{2}:1$, respectively.

The frozen core effects that were calculated in chapter 4 where added to these calculations, and the result are displayed in figure 5.11, note the improvement in

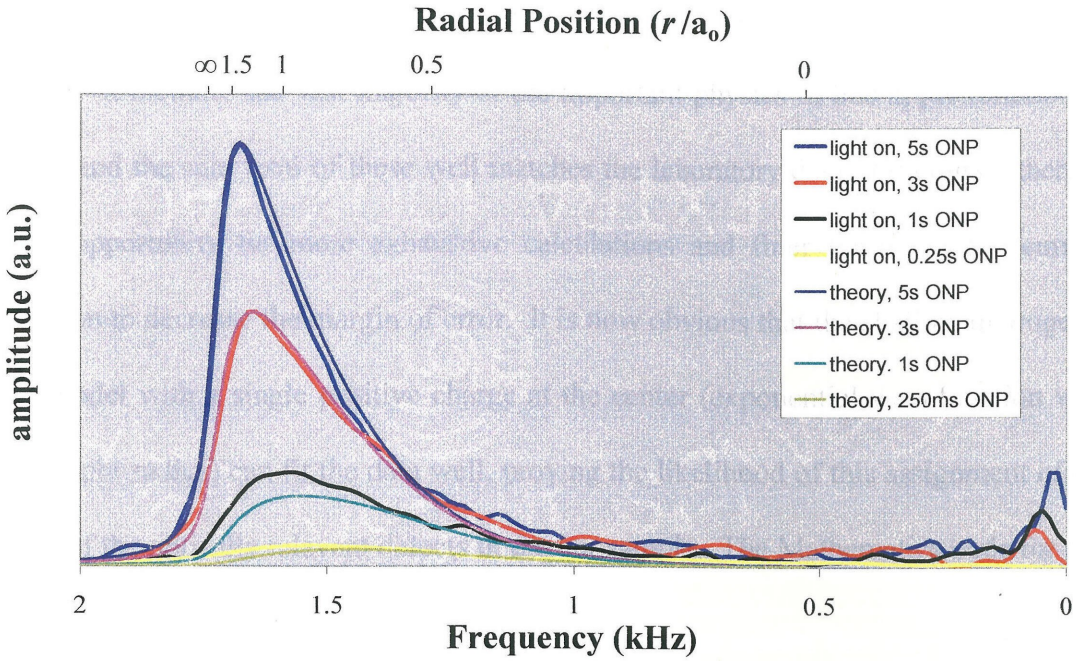


Figure 5.12 Using an average spin diffusion rate keeps the improvements in the fit of the spectra at shorter ONP times, and simultaneously restores intensity to the high frequency tail of the spectral fits. In this calculation $T_{1on}(0) = 200$ ms, $\langle s \rangle = 0.15$, and $W = 37$. This is the best fit to the experimental data obtained thus far. A second x-axis is provided here that displays the radial position of the nuclei giving rise to the signal at a given frequency.

lineshape of the 250 ms ONP time Knight shift image and the loss of high Knight shift intensity for the longer ONP time Knight shift images.

There is another way to pursue these calculations so as to produce a radial polarization profile that will fit the average radial polarization profile. This is to average the spin diffusion rates in different directions, performing one computation at this average spin diffusion rate. This leads to $W \simeq 74/2$, and the results of this computation are shown in figure 5.12.

Clearly this is the best fit to the radially resolved Knight shift images ever obtained. It includes the vast majority of the important physics in one approximation or another, and the sum total of these well matches the laboratory data. Of course, there is still an opportunity for more exhaustive calculations and finer tuning of the current calculation to decrease the margin of error. It is now obvious that the shallow hydrogenic donor model with a single positive charge at the center (exponential wavefunction with ~100Å Bohr radius) can fit the data well, proving the likelihood of this assignment of the identity of the optically relevant defects in the our sample. The Mathematica code for this calculation is given in appendix B of this thesis.

This figure also shows the radial position of the nuclei that gives rise to the signal at a given frequency. Comparing the Knight shift image at 3s ONP time from this figure with figure 4.18 of James Kempf's thesis[5], it is clear that these simulations show a much higher relative contribution to the spectrum from sites close to the center of the donor than the analysis using empirical weights yields. Indeed, all of the spin diffusion calculations presented here show a very similar radial profile of contributions to the detected signal. This relative disagreement is due to the inaccurate assumption that the spectra at 144ms to 720ms ONP time are roughly free of the effects of spin diffusion.

An analysis of the spin diffusion constant makes it apparent that, even if it is taken as solely arising from the nearest neighbor coupling, it is still significant at even times much earlier than the earliest ONP times. However, while spin diffusion will change the shape of the plot of polarization versus distance, it can still be fit to a continuous dielectric medium approximation without spin diffusion but with an artificially faster

ONP time, and it will only deviate from the family of shapes fit by any possible ONP rate at longer times. Moreover, these short ONP time fits without spin diffusion assign the signal as arising from artificially more distant sites than they are actually coming from since it is such functions that can still fit the altered shape of the spectrum produced by spin diffusion.

One consequence of this is that it may undermine the analysis of the electric field POWER NMR spectra presented in section D of chapter 4 of Jim's thesis by providing a wider spread to the quadrupole satellites surrounding the central transition. More accurate computations of the POWER NMR experimental results that can yield answers about which donor model(s) match the observed spectra should be accomplished using spin diffusion to obtain a polarization versus distance relationship from the Knight shift image with the same ONP time, and, then, using this as an input to a sum-over-all-sites-around-a-donor calculation.

C. Conclusions

We have captured the first ever radially resolved Knight shift images from those nuclei near a point defect in GaAs. Additionally, a deconvolution of these images into their constituent physical interactions has been approximately carried out, yielding the shape and size the electronic orbital in which the electron is trapped at the defect, the occupancy of that electronic orbital, and the quadrupolar interactions in the vicinity of the defect, including the charge state of the defect. We can most easily learn all of this information if spin diffusion is nonexistent or if it is extremely fast; however, the current

evidence is that it is in the intermediate region between these extremes, requiring a higher order approximation to spin diffusion in order to disentangle this information from the Knight shift image. Such a model has been developed, and representative calculations have been presented.

Additionally a full, real-time analysis of each and every one of the hundreds of thousands of nuclei surrounding a defect in GaAs has been accomplished. This was achieved using a non-interacting nucleus model that computes the time domain evolution for each and every individual nucleus including its Knight shift, quadrupolar interactions (both secular and nonsecular), individual optical polarization conditions, optical detection weighting, and rigorously exact rf effects. The successful interface of these two computations to leverage their individual strengths is clearly a desirable route to increase the accuracy of the computations to an even higher level.

There are two different techniques that have been employed to reach the goal of achieving a radially resolved Knight shift image. Initially, we used a stimulated echo experiment, and, later, we used the CLSW16 pulse sequence, a high order multiple pulse line narrowing sequence capable of removing all NMR interactions regardless of their tensor rank. In both cases, the Knight shift, a first rank tensor interaction, would be removed by these line-narrowing techniques if it were present at all times in the experiment. However, we overcame this by turning on the light (the supply of polarized, excited-state electrons) during certain periods of the experiment, and leaving it off during the rest of the experiment, thereby preventing this interaction from being averaged away.

The computations presented here provide insight in to the defect and its electronic

state as well as the shape, size, and occupancy of the electronic orbital. Moreover, a strong theoretical and computational framework has been developed, and this can serve as a basis for future work, solving these and other related problems. Furthermore, as the continued driving of Moore's Law increases the extent of computational power available for application to problems such as these, progressively more exact computations will naturally become tractable.

D. References

1) Cho, H. M.; Lee, C. J.; Shykind, D. N.; Weitekamp, D. P. *Physical Review Letters* **1985**, *55*, 1923.

2) Buratto, S. K.; Shykind, D. N.; Weitekamp, D. P. *Journal of Vacuum Science and Technology B* **1992**, *10*, 1740.

3) Buratto, S. K. *Time-Sequenced Optical Nuclear Magnetic Resonance of Gallium Arsenide*; California Institute of Technology: Pasadena, CA, 1992.

4) Marohn, J. A. *I. Multiple-Pulse Radio-Frequency Gradient Nuclear Magnetic Resonance Imaging of Solids. II. Optical Nuclear Magnetic Resonance Analysis of Epitaxial Gallium Arsenide Structures.*; California Institute of Technology: Pasadena, CA, 1996, pp 143.

5) Kempf, J. G. *Probing Quantum Confinement at the Atomic Scale with Optically Detected Nuclear Magnetic Resonance*; California Institute of Technology: Pasadena, 2001, pp 172.

6) Arfken, G. *Mathematical Methods for Physicists*; Academic Press: San Diego, 1985.

Appendix A – Mathematica Code for the Sum Over Single Spins Calculations

This Mathematica notebook computes some properties (such as quadrupole coupling and LBD signal) around a donor site in a GaAs semiconductor device.

The Rhodonor series of notebooks do a density matrix calculation at each atomic site. This notebook has been programmed to simulate a light-on/light-off CLSW-16 experiment in the laboratory frame (i.e., it computes the FID following a series of rf pulses without using the assumptions involved in rotating frame calculations), including pulse offsetting. Furthermore, this particular experiment has the RF pulses TPPI'ed into the rotating-frame(the way pulses are actually given in the experiment) and Spin-Lock detection.

Hq = I.V.I is calculated in this version (only for the specific case of B in the [001] direction). Also this version uses a Neutral ("shielded charge" model) Donor assumption to calculate the E-field around the donor when the light is on, and a charged donor ("bare charge in a dielectric medium" model) assumption to calculate the E-field around the donor when the light is off.

It starts by setting up various properties as a function of position in the lattice, and it uses the fact that only one quadrant need be calculated (due to the symmetry of the problem). Furthermore, it sets up the fcc lattice of galliums in GaAs in three separate steps. Firstly, it sets up a simple cubic lattice corresponding to the atoms in the corners of the fcc lattice. Secondly, it sets up another simple cubic lattice corresponding to the atoms in the face-centers of the tops and bottoms of the unit cells of the fcc lattice (note that this lattice and the lattice from step #1 are a pair of identical, interpenetrating simple cubic

lattices where the lattice from step #2 is just offset from the other lattice by 1/2 of a cell width in x and y). Next, it sets up yet another set of interpenetrating simple cubic lattices which correspond to the atoms in the front and back faces and the left and right faces of the unit cell respectively. The positions of the atoms in terms of array indices (where each increment in an array index is equal to a step which is one half of a unit cell in length) are as follows: the origin, which is (0,0,0), (0,1,1), (1,0,1), (1,1,0) and all points which differ from these by addition of an even number to each digit. For example, (2,2,2) = (0,0,0) + (2,2,2), (7,4,19) = (1,0,1) + (6,4,18) and (13,3,0) = (12,2,0) + (1,1,0) are all positions of atoms in the gallium lattice; however, there are no atoms at (1,1,1) = (0,0,0) + (1,1,1) and (17,4,8) = (1,0,1) + (16,4,7) = (1,0,0) + (16,4,8). The general rule is that points with indices which are all even numbers and points with indices which consist of two odd numbers and one even number have gallium atoms at them. Moreover, points with indices which which consist of all odd numbers or two even numbers and one odd number correspond to the octahedral holes of the gallium lattice. It should be noted that gallium arsenide has arsenic atoms at one half of the tetrahedral hole sites of the gallium fcc latice (at all of the positive tetrahedral hole sites or all of the negative tetrahedral hole sites depending on your definitions), and these four tetrahedrally arranged aresenic atoms are closer to this "octahedral" site than the six octahedrally arranged gallium atoms. Therefore, this octahedral hole site in the gallium fcc lattice is now a tetrahedral hole in the gallium arsenide lattice. Furthermore, the other type of tetrahedral hole in the gallium lattice is still vacant, and, thus, still tetrahedral in nature.

Step #0

■ Set Numerical Values to Constants

Note that RR = R(1,4) - S(4,4)*d(1,4) converted into units of 1/(meters)

```
ndigit = 16;
```

$$h = N\left[\frac{66260755}{10000000\ 10^{34}}, ndigit\right];$$

$$q = N\left[\frac{160219}{100000\ 10^{19}}, ndigit\right];$$

$$QQGa = N\left[\frac{112}{(1000\ 10^{24})\ (100\ 100)}, ndigit\right];$$

$$QQAs = N\left[\frac{3}{(10\ 10^{24})\ (100\ 100)}, ndigit\right];$$

$$eGaAs = N\left[\frac{131\ 88541878}{10\ 10000000\ 10^{12}}, ndigit\right];$$

$$RR = N\left[\left(\frac{285\ 10^{10}}{100} + \frac{2\ (65\ 10^{15})}{10^{8}\ 10}\right) 100, ndigit\right];$$

$$N2Pi = N[2\ \pi, ndigit];$$

$$NhalfSqrt2 = N\left[\frac{\sqrt{2}}{2}, ndigit\right];$$

```
gammaGa71 = N[12984000, ndigit];
gammaAs75 = N[7291900, ndigit];
electronDensityOnGa71 = N[58 10^31 / 10, ndigit];
electronDensityOnAs75 = N[98 10^31 / 10, ndigit];
hyperFineGa71 =
 N[2 (4 π) 9274 2 gammaGa71 electronDensityOnGa71 / (3 10^7 (1000 10^24) 2), ndigit
hyperFineAs75 =
 N[2 (4 π) 9274 2 gammaAs75 electronDensityOnAs75 / (3 10^7 (1000 10^24) 2), ndigit
LuminPolarization = N[1 / 8];
a0 = N[100 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π,
ksconstAs75 =
 N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
```

```
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit]
```

$$8.522777196766995 \times 10^{-13}$$

Step #1

Note - Pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase then Y is $-\pi/2$ and $\overline{Y}$ is $\pi/2$. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
rho = Table[N[0, ndigit], {i, 4}, {j, 4}];
One = Table[N[1, ndigit], {i, 4}, {j, 4}];
Von = Table[N[0, ndigit], {i, 3}, {j, 3}];
Voff = Table[N[0, ndigit], {i, 3}, {j, 3}];
propon = Table[N[0, ndigit], {i, 4}, {j, 4}];
propoff = Table[N[0, ndigit], {i, 4}, {j, 4}];
identity = IdentityMatrix[4];
```

$$\text{Iz} = \text{N}\left[\begin{pmatrix} \frac{3}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & -\frac{3}{2} \end{pmatrix}, \text{ndigit}\right];$$

$$\text{Ix} = \text{N}\left[\begin{pmatrix} 0 & \frac{\sqrt{3}}{2} & 0 & 0 \\ \frac{\sqrt{3}}{2} & 0 & 1 & 0 \\ 0 & 1 & 0 & \frac{\sqrt{3}}{2} \\ 0 & 0 & \frac{\sqrt{3}}{2} & 0 \end{pmatrix}, \text{ndigit}\right];$$

$$\text{Iy} = \text{N}\left[\begin{pmatrix} 0 & \frac{-\text{I}\sqrt{3}}{2} & 0 & 0 \\ \frac{\text{I}\sqrt{3}}{2} & 0 & -\text{I} & 0 \\ 0 & \text{I} & 0 & -\frac{\text{I}\sqrt{3}}{2} \\ 0 & 0 & \frac{\text{I}\sqrt{3}}{2} & 0 \end{pmatrix}, \text{ndigit}\right];$$

```
Iplus = Ix + I Iy;
Iminus = Ix - I Iy;
IxSqrd = MatrixPower[Ix, 2];
IySqrd = MatrixPower[Iy, 2];
IzSqrd = MatrixPower[Iz, 2];
ItotalSqrd = IzSqrd + IxSqrd + IySqrd;
IySqrdminusIxSqrd = IySqrd - IxSqrd;
IzIxplusIxIz = Iz . Ix + Ix . Iz;
IzIyplusIyIz = Iz . Iy + Iy . Iz;
rhonorm = Apply[Plus, Flatten[Ix Ix]];
```

Note that because V[[1,1]] = -V[[2,2]], V[[3,1]] = V[[1,3]] and
V[[3,2]] = V[[2,3]], the formula's below involve only V[[2,2]], V[[1,3]]
and V[[2,3]].

```
(0 + 11. / 60) 4661 / 60
```

```
14.2419
```

```
(0 + 11. / 60) 15215 / (60 24)
```

```
1.93709
```

```
(0 + 11. / 60) 117289 / (60 24)
```

```
14.9326
```

Light-On "nutation"-ized version with circularly polarized RF. Spin-lock detection has been removed from this version.

Note - Pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase then Y is $-\pi/2$ and $\overline{Y}$ is $\pi/2$. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
resonant90time = 28 / 10^7;
RFfreq = N[3200000, ndigit];
Larmor = N[3200000, ndigit];
StepsPerCycle = 25;
npts = 512;
SW = N[500 10^3, ndigit];
Idetect = Transpose[Iplus];
UniformOmegaQ = N[0 * 15000, ndigit];
Omega1 = N[1 / (4 resonant90time), ndigit];
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {4}, {4}];
Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq / SW], {n, 1, 4, 1}];
UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
    Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
a0 = N[100 / 10^10, ndigit];
```

```
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
Do[
 Do[
  Do[
   rr = N[√(i^2 + j^2 + k^2), ndigit]; z = N[C2 rr, ndigit];
   EEon = N[1 / (a0 z)^2, ndigit] Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   ks = 0 * ksconst Exp[-2 z];
   Hevolutnon = (Larmor - ks) Iz +
     0 * (Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
       Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3));
   norm = Exp[-2 z] / rhonorm;
   rhozero = Iz;
   time = InitTime;
   Ucycle = identity;
   Do[Htotal = Hevolutnon + Omega1 (Cos[N2Pi RFfreq delta (tt - 1/2)] Ix +
       Sin[N2Pi RFfreq delta (tt - 1/2)] Iy);
    {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
    Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
   , {tt, time + 1, time + StepsPerCycle, 1}];
   UpulseCycle = Ucycle;
   data1[[1]] = norm Apply[Plus, Flatten[rhozero Iplus]];
   Do[
    ncycles = Floor[RFfreq (t1 - 1) / SW];
    pulseRemainder =
     Round[StepsPerCycle ((RFfreq (t1 - 1) / SW) - ncycles)];
    Upulse = MatrixPower[UpulseCycle, ncycles];
    Ucycle = identity;
    Do[Htotal = Hevolutnon + Omega1 (Cos[N2Pi RFfreq delta (tt - 1/2)] Ix +
```

```
        Sin[N2Pi RFfreq delta (tt - 1/2)] Iy);
      {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
      Ucycle = Transpose[vecs] .
        DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
      , {tt, time + 1, time + pulseRemainder, 1}];
      Upulse = Ucycle . Upulse;
      rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
      data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
      Idetect = UtppiRFframe . Idetect . UtppiRFframeadj
    , {t1, 2, npts, 1}]
  , {k, 100, 100, 4}]
 , {j, 100, 100, 4}]
, {i, 100, 100, 4}];
```

Light-On "nutation"-ized version with linearly polarized RF. Spin-lock detection has been removed from this version.

```
resonant90time = 28/10^7;
RFfreq = N[3200000, ndigit];
Larmor = N[3200000, ndigit];
StepsPerCycle = 25;
npts = 512;
SW = N[500 10^3, ndigit];
Idetect = Transpose[Iplus];
UniformOmegaQ = N[0 * 15000, ndigit];
Omega1 = N[1/(4 resonant90time), ndigit];
delta = N[1/(StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {4}, {4}];
Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq/SW], {n, 1, 4, 1}];
UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
    Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR/(4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
a0 = N[100/10^10, ndigit];
```

```
C2 = N[565/(100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed/π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed/π, ndigit];
Do[
 Do[
  Do[
   rr = N[Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEon = N[1/(a0 z)^2, ndigit] Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k/rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i)/rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i)/rr;
   ks = 0 * ksconst Exp[-2 z];
   Hevolutnon = (Larmor - ks) Iz +
     0 * (Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
       Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd/3));
   norm = Exp[-2 z]/rhonorm;
   rhozero = Iz;
   time = InitTime;
   Ucycle = identity;
   Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
     {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
     Ucycle = Transpose[vecs] .
      DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
   , {tt, time + 1, time + StepsPerCycle, 1}];
   UpulseCycle = Ucycle;
   data1[[1]] = norm Apply[Plus, Flatten[rhozero Iplus]];
   Do[
    ncycles = Floor[RFfreq (t1 - 1) / SW];
    pulseRemainder =
     Round[StepsPerCycle ((RFfreq (t1 - 1) / SW) - ncycles)];
    Upulse = MatrixPower[UpulseCycle, ncycles];
    Ucycle = identity;
    Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
      {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
      Ucycle = Transpose[vecs] .
```

```
      DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
     , {tt, time + 1, time + pulseRemainder, 1}];
     Upulse = Ucycle . Upulse;
     rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
     data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
     Idetect = UtppiRFframe . Idetect . UtppiRFframeadj
    , {t1, 2, npts, 1}]
   , {k, 100, 100, 4}]
  , {j, 100, 100, 4}]
 , {i, 100, 100, 4}];
```

Light-On "1D-FT"-ized version. Spin-lock detection has been removed from this version.

```
t90pulse = 28125 / 10^10;
resonant90time = 28125 / 10^10;
RFfreq = N[3200000, ndigit];
Larmor = N[3200000, ndigit];
StepsPerCycle = 25;
nmax = 120;
npts = 512;
SW = N[8000 10^3, ndigit];
UniformOmegaQ = N[0 * 15000, ndigit];
Omega1 = N[1 / (4 resonant90time), ndigit];
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {i, 4}, {j, 4}];
Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq / SW], {n, 1, 4, 1}];
ncycles = Floor[RFfreq * t90pulse];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
     Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
a0 = N[100 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
```

```
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
Do[
 Do[
  Do[
   rr = N[√(i^2 + j^2 + k^2), ndigit]; z = N[C2 rr, ndigit];
   EEon = N[1 / (a0 z)^2, ndigit] (1 + 0 * Exp[-2 z] (2 z^2 + 2 z + 1));
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   ks = ksconst Exp[-2 z];
   Hevolutnon =
    (Larmor - ks) Iz + Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   Do[propon[[n, m]] = Exp[-I N2Pi (valson[[n]] - valson[[m]]) / SW],
    {n, 1, 4, 1}, {m, 1, 4, 1}];
   norm = Exp[-2 z] / rhonorm;
   Idetect = Conjugate[vecson] . Transpose[Iplus] . Transpose[vecson];
   UtppiRFframe = Conjugate[vecson] . UtppiRFframe . Transpose[vecson];
   UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
   rhozero = Iz;
   propa = propon;
   time = InitTime;
   Ucycle = identity;
   Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
    {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
    Ucycle = Transpose[vecs] .
      DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
   , {tt, time + 1, time + StepsPerCycle, 1}];
   Upulse = MatrixPower[Ucycle, ncycles];
   Ucycle = identity;
   Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
    {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
    Ucycle = Transpose[vecs] .
      DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
    , {tt, time + 1, time + pulseRemainder, 1}];
    Upulse = Ucycle . Upulse;
    rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
    rho = Conjugate[vecson] . rho . Transpose[vecson];
    Do[
      data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
      rho = rho propa;
      Idetect = UtppiRFframe . Idetect . UtppiRFframeadj
    , {t1, 1, npts, 1}]
  , {k, 0, 0, 4}]
 , {j, 20, 20, 4}]
, {i, 20, 20, 4}];
```

Light-On "2D-FT"-ized version (but no store-tip pulses). Spin-lock detection has been removed from this version.

```
t90pulse = 28/10^7;
resonant90time = 28/10^7;
RFfreq = N[3152000, ndigit];
Larmor = N[3152000, ndigit];
StepsPerCycle = 25;
nmax = 120;
npts = 256;
nt2pts = 10;
SW = N[200 10^3, ndigit];
TPPIfreq = N[50 10^3, ndigit];
UniformOmegaQ = N[15000, ndigit];
Omega1 = N[1/(4 resonant90time), ndigit];
Omega2 = (200/1400) Omega1;
delta = N[1/(StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {i, 4}, {j, 4}];
ncycles = Floor[RFfreq * t90pulse];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
     Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR/(4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
```

```
a0 = N[100/10^10, ndigit];
C2 = N[565/(100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed/π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed/π, ndigit];
Do[
 Do[
  Do[
   rr = N[Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEon = N[1/(a0 z)^2, ndigit] Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k/rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i)/rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i)/rr;
   ks = ksconst Exp[-2 z];
   Hevolutnon =
    (Larmor - ks) Iz + Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd/3);
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   Do[propon[[n, m]] = Exp[-I N2Pi (valson[[n]] - valson[[m]])/SW],
    {n, 1, 4, 1}, {m, 1, 4, 1}];
   norm = Exp[-2 z]/rhonorm;
   Idetect = Conjugate[vecson] . Transpose[Iplus] . Transpose[vecson];
   Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq/SW], {n, 1, 4, 1}];
   UtppiRFframe = Conjugate[vecson] . UtppiRFframe . Transpose[vecson];
   UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
   rhozero = Iz;
   propa = One;
   Do[
    time = InitTime;
    phi = (t1 - 1)/SW TPPIfreq N2Pi;
    Ucycle = identity;
    Do[
     Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + phi] Ix;
     {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
```

```
      Ucycle = Transpose[vecs] .
        DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
      , {tt, time + 1, time + StepsPerCycle, 1}];
     Upulse = MatrixPower[Ucycle, ncycles];
     Ucycle = identity;
     Do[
     Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + phi] Ix;
      {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
      Ucycle = Transpose[vecs] .
        DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
      , {tt, time + 1, time + pulseRemainder, 1}];
     Upulse = Ucycle . Upulse;
     rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
     rho = Conjugate[vecson] . rho . Transpose[vecson];
     rho = rho propa;
     data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
     Idetect = UtppiRFframe . Idetect . UtppiRFframeadj;
     propa = propa propon;
    , {t1, 1, npts, 1}]
   , {k, 20, 20, 4}]
  , {j, 20, 20, 4}]
 , {i, 20, 20, 4}];
```

Note that, because of the PI pulse at the end, all frequencies are negative compared to other experiments!

```
t90pulse = 28/10^7;
resonant90time = 28/10^7;
RFfreq = N[3152000, ndigit];
Larmor = N[3152000, ndigit];
StepsPerCycle = 25;
nmax = 120;
npts = 128;
nt2pts = 10;
SW = N[200 10^3, ndigit];
TPPIfreq = N[50 10^3, ndigit];
UniformOmegaQ = N[15000, ndigit];
Omega1 = N[1/(4 resonant90time), ndigit];
```

```
Omega2 = (200 / 1400) Omega1;
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {i, 4}, {j, 4}];
ncycles = Floor[RFfreq * t90pulse];
ncycles2 = Floor[RFfreq * 2 * t90pulse];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainder2 = Round[StepsPerCycle (RFfreq * 2 * t90pulse - ncycles2)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
    Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
a0 = N[100 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
Do[
 Do[
  Do[
   rr = N[√(i^2 + j^2 + k^2), ndigit]; z = N[C2 rr, ndigit];
   EEon = N[1 / (a0 z)^2, ndigit] Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   ks = ksconst Exp[-2 z];
   Hevolutnon =
    (Larmor - ks) Iz + Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   Do[propon[[n, m]] = Exp[-I N2Pi (valson[[n]] - valson[[m]]) / SW],
    {n, 1, 4, 1}, {m, 1, 4, 1}];
   norm = Exp[-2 z] / rhonorm;
   Idetect = Conjugate[vecson] . Transpose[Iplus] . Transpose[vecson];
```

```
Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq/SW], {n, 1, 4, 1}];
UtppiRFframe = Conjugate[vecson] . UtppiRFframe . Transpose[vecson];
UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
rhozero = Iz;
time = InitTime;
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rhozero = Upulse . rhozero . Conjugate[Transpose[Upulse]];
rhozero = Conjugate[vecson] . rhozero . Transpose[vecson];
propa = One;
Do[
 time = InitTime + pulseRemainder + (t1 - 1)/(delta SW);
 phi = (t1 - 1)/SW TPPIfreq N2Pi / 2;
 rho = rhozero propa;
 Ucycle = identity;
 Do[
 Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - phi] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
 , {tt, time + 1, time + StepsPerCycle, 1}];
 Upulse = MatrixPower[Ucycle, ncycles2];
 Ucycle = identity;
 Do[
 Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - phi] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
```

```
      DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
    , {tt, time + 1, time + pulseRemainder2, 1}];
    Upulse = Ucycle . Upulse;
    Upulse = Conjugate[vecson] . Upulse . Transpose[vecson];
    rho = Upulse . rho . Conjugate[Transpose[Upulse]];
    data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
    Idetect = UtppiRFframe . Idetect . UtppiRFframeadj;
    propa = propa propon;
   , {t1, 1, npts, 1}]
  , {k, 20, 20, 4}]
 , {j, 20, 20, 4}]
, {i, 20, 20, 4}];
```

Light-On "spin echo"-ized version. Spin-lock detection has been removed from this version and I_+ is detected instead of I_x.

```
t90pulse = 28/10^7;
resonant90time = 28/10^7;
RFfreq = N[3152000, ndigit];
Larmor = N[3152000, ndigit];
StepsPerCycle = 25;
nmax = 120;
npts = 128;
nt2pts = 10;
SW = N[200 10^3, ndigit];
TPPIfreq = N[-50 10^3, ndigit];
UniformOmegaQ = N[15000, ndigit];
Omega1 = N[1/(4 resonant90time), ndigit];
Omega2 = (200/1400) Omega1;
delta = N[1/(StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {i, 4}, {j, 4}];
ncycles = Floor[RFfreq * t90pulse];
ncycles2 = Floor[RFfreq * 2 * t90pulse];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainder2 = Round[StepsPerCycle (RFfreq * 2 * t90pulse - ncycles2)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
     Floor[((101752605 * 10^-6) / 100) * RFfreq])];
```

```
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
a0 = N[100 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
Do[
 Do[
  Do[
   rr = N[Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEon = N[1 / (a0 z)^2, ndigit] Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   ks = ksconst Exp[-2 z];
   Hevolutnon =
    (Larmor - ks) Iz + Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   Do[propon[[n, m]] = Exp[-I N2Pi (valson[[n]] - valson[[m]]) / (2 SW)],
    {n, 1, 4, 1}, {m, 1, 4, 1}];
   norm = Exp[-2 z] / rhonorm;
   Idetect = Conjugate[vecson] . Transpose[Iplus] . Transpose[vecson];
   Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq / SW], {n, 1, 4, 1}];
   UtppiRFframe = Conjugate[vecson] . UtppiRFframe . Transpose[vecson];
   UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
   rhozero = Iz;
   time = InitTime;
   Ucycle = identity;
   Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
    {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
    Ucycle = Transpose[vecs] .
      DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rhozero = Upulse . rhozero . Conjugate[Transpose[Upulse]];
rhozero = Conjugate[vecson] . rhozero . Transpose[vecson];
propa = One;
Do[
  time = InitTime + pulseRemainder + (t1 - 1)/(delta 2 SW);
  phi = (t1 - 1)/SW TPPIfreq (N2Pi / 2);
  rho = rhozero propa;
  Ucycle = identity;
  Do[
  Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + phi] Ix;
   {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
   Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
  , {tt, time + 1, time + StepsPerCycle, 1}];
  Upulse = MatrixPower[Ucycle, ncycles2];
  Ucycle = identity;
  Do[
  Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + phi] Ix;
   {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
   Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
  , {tt, time + 1, time + pulseRemainder2, 1}];
  Upulse = Ucycle . Upulse;
  Upulse = Conjugate[vecson] . Upulse . Transpose[vecson];
  rho = Upulse . rho . Conjugate[Transpose[Upulse]];
  rho = rho propa;
  data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
  Idetect = UtppiRFframe . Idetect . UtppiRFframeadj;
  propa = propa propon;
, {t1, 1, npts, 1}]
```

```
  , {k, 20, 20, 4}]
 , {j, 20, 20, 4}]
, {i, 20, 20, 4}];
```

"Spin echo"-ized version which retains light on/off in different windows. Spin-lock detection has been removed from this version and detection is of I_+ instead of I_x.

```
t90pulse = 28 / 10^7;
resonant90time = 28 / 10^7;
RFfreq = N[3200000, ndigit];
Larmor = N[3200000, ndigit];
StepsPerCycle = 25;
nmax = 120;
npts = 512;
nt2pts = 10;
SW = N[200 10^3, ndigit];
TPPIfreq = N[50 10^3, ndigit];
UniformOmegaQ = N[0 * 15000, ndigit];
Omega1 = N[1 / (4 resonant90time), ndigit];
Omega2 = (200 / 1400) Omega1;
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
UtppiRFframe = Table[N[0, ndigit], {i, 4}, {j, 4}];
ncycles = Floor[RFfreq * t90pulse];
ncycles2 = Floor[RFfreq * 2 * t90pulse];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainder2 = Round[StepsPerCycle (RFfreq * 2 * t90pulse - ncycles2)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
    Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
a0 = N[100 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
```

```
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
Do[
 Do[
  Do[
   rr = N[√(i^2 + j^2 + k^2), ndigit]; z = N[C2 rr, ndigit];
   EEoff = N[1 / (a0 z)^2, ndigit]; EEon = EEoff Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   Voff[[2, 2]] = C1 EEoff k / rr; Voff[[1, 3]] = NhalfSqrt2 C1 EEoff (-j + i) / rr;
   Voff[[2, 3]] = NhalfSqrt2 C1 EEoff (j + i) / rr;
   ks = ksconst Exp[-2 z];
   Hevolutnon =
    (Larmor - ks) Iz + Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   Hevolutnoff =
    Larmor Iz + Voff[[2, 2]] IySqrdminusIxSqrd + Voff[[1, 3]] IzIxplusIxIz +
     Voff[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   Do[propon[[n, m]] = Exp[- I N2Pi (valson[[n]] - valson[[m]]) / (2 SW)],
    {n, 1, 4, 1}, {m, 1, 4, 1}];
   {valsoff, vecsoff} = Eigensystem[Hevolutnoff]; valsoff = Re[valsoff];
   Do[propoff[[n, m]] = Exp[- I N2Pi (valsoff[[n]] - valsoff[[m]]) / (2 SW)],
    {n, 1, 4, 1}, {m, 1, 4, 1}];
   norm = Exp[-2 z] / rhonorm;
   Idetect = Conjugate[vecson] . Transpose[Iplus] . Transpose[vecson];
   Do[UtppiRFframe[[n, n]] = Exp[I N2Pi Iz[[n, n]] RFfreq / SW], {n, 1, 4, 1}];
   UtppiRFframe = Conjugate[vecson] . UtppiRFframe . Transpose[vecson];
   UtppiRFframeadj = Conjugate[Transpose[UtppiRFframe]];
   rhozero = Iz;
   time = InitTime;
   Ucycle = identity;
```

```
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rhozero = Upulse . rhozero . Conjugate[Transpose[Upulse]];
rhozero = Conjugate[vecson] . rhozero . Transpose[vecson];
propa = One;
propb = One;
Do[
 time = InitTime + pulseRemainder + (t1 - 1)/(delta 2 SW);
 phi = (t1 - 1)/SW TPPIfreq (N2Pi / 2);
 rho = rhozero propb;
 Ucycle = identity;
 Do[
 Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + phi] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
 , {tt, time + 1, time + StepsPerCycle, 1}];
 Upulse = MatrixPower[Ucycle, ncycles2];
 Ucycle = identity;
 Do[
 Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + phi] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
 , {tt, time + 1, time + pulseRemainder2, 1}];
 Upulse = Ucycle . Upulse;
 Upulse = Conjugate[vecson] . Upulse . Transpose[vecson];
 rho = Upulse . rho . Conjugate[Transpose[Upulse]];
 rho = rho propa;
```

```
    data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
    Idetect = UtppiRFframe . Idetect . UtppiRFframeadj;
    propa = propa propon;
    propb = propb propoff;
   , {t1, 1, npts, 1}]
  , {k, 20, 20, 4}]
 , {j, 20, 20, 4}]
, {i, 20, 20, 4}];
```

CLSW-16 version wth light on/off in different windows. Spin-lock detection has been removed from this version and detection is of I_+ instead of I_x. (no offsetting here)

Note that pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase then Y is $-\pi/2$ and $\overline{Y}$ is $\pi/2$. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
t90pulse = 28/10^7;
resonant90time = 28/10^7;
RFfreq = N[3200000, ndigit];
Larmor = N[3200000, ndigit];
StepsPerCycle = 25;
tONP = 5;
t1Off = 21;
t1OnZero = 0.2;
t2max = 2;
nmax = 120;
npts = 2048;
nt2pts = 10;
tcycle = N[120 10^-6, ndigit];
tWinS = (tcycle / 24) - t90pulse;
tWinL = (tcycle / 12) - t90pulse;
SW = 1 / tcycle;
UniformOmegaQ = N[0 * 16000, ndigit];
```

```
Omega1 = N[1 / (4 resonant90time), ndigit];
Omega2 = (200 / 1400) Omega1;
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
ncycles = Floor[RFfreq * t90pulse];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
      Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 2];
a0 = N[70 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
Do[
 Do[
  Do[
   rr = N[0.0000001 + Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEoff = N[1 / (a0 z)^2, ndigit]; EEon = EEoff Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   Voff[[2, 2]] = C1 EEoff k / rr; Voff[[1, 3]] = NhalfSqrt2 C1 EEoff (-j + i) / rr;
   Voff[[2, 3]] = NhalfSqrt2 C1 EEoff (j + i) / rr;
   ks = ksconst Exp[-2 z];
   Hevolutnon = (Larmor - ks) Iz +
     0 * (Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
       Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3));
   Hevolutnoff =
    Larmor Iz + 0 * (Voff[[2, 2]] IySqrdminusIxSqrd + Voff[[1, 3]] IzIxplusIxIz +
```

```
      Voff[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd / 3)) ;
{valson, vecson} = Eigensystem[Hevolutnon] ; valson = Re[valson] ;
{valsoff, vecsoff} = Eigensystem[Hevolutnoff] ; valsoff = Re[valsoff] ;
UevonL = Transpose[vecson] .
  DiagonalMatrix[Exp[-I N2Pi valson tWinL]] . Conjugate[vecson] ;
UevoffS = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinS]] . Conjugate[vecsoff] ;
UevoffL = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL]] . Conjugate[vecsoff] ;
UevoffLS = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL / 2]] . Conjugate[vecsoff] ;
t1on = Exp[4 z] t1OnZero;
Ts = (t1Off t1on) / (t1Off + t1on) ;
norm =
  1/rhonorm (8 Ts (1 - Exp[-tONP / Ts]) (1 - Exp[-t2max / t1on]) Exp[-2 z]) ;
Idetect = Transpose[Iplus] ;
rhozero = Iz;
time = InitTime;
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)]] Ix;
 {vals, vecs} = Eigensystem[Htotal] ; vals = Re[vals] ;
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}] ;
Upulse = MatrixPower[Ucycle, ncycles] ;
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)]] Ix;
 {vals, vecs} = Eigensystem[Htotal] ; vals = Re[vals] ;
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}] ;
Upulse = Ucycle . Upulse;
rhozero = Upulse . rhozero . Conjugate[Transpose[Upulse]] ;
Uclsw16 = UevoffLS;
time = time + pulseRemainder + Round[(tWinL / 2) / delta] ;
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal] ; vals = Re[vals] ;
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
```

```
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
```

```
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
```

```
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
```

```
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
```

```
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS . Upulse . Uclsw16;
```

```
time = time + pulseRemainder + Round[tWinS / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
 , {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
 , {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffLS . Upulse . Uclsw16;
Uclsw16adj = Conjugate[Transpose[Uclsw16]];
rho = rhozero;
Do[
  data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
  rho = Uclsw16 . rho . Uclsw16adj;
 , {t1, 1, npts, 1}]
, {k, 20, 20, 4}]
, {j, 20, 20, 4}]
, {i, 20, 20, 4}];
```

CLSW-16 version wth light on/off in different windows and pulse offsets to create an Average-Hamiltonian proportional to Iz. Spin-lock detection has been removed from this version and detection is of I_+ instead of I_x.

Note that pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase

then Y is -π/2 and $\overline{Y}$ is π/2. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
t90pulse = 29/10^7;
resonant90time = 2885/10^9;
RFfreq = N[3200000, ndigit];
Larmor = N[3200000, ndigit];
StepsPerCycle = 50;
tONP = 5;
t1OnZero = N[17 / 100, ndigit];
t2max = 2;
nmax = 20;
npts = 1024;
nt2pts = 10;
tcycle = N[120 10^-6, ndigit];
tOffset = N[150 10^-9, ndigit];
tWinS = (tcycle / 24) - t90pulse;
tWinL = (tcycle / 12) - t90pulse;
tWinS1 = tWinS - tOffset;
tWinS2 = tWinS + tOffset;
tWinL1 = tWinL - 2 * tOffset;
tWinL2 = tWinL + 2 * tOffset;
SW = 1 / tcycle;
UniformOmegaQ = N[0 * 15000];
Omega1 = N[1/(4 resonant90time), ndigit];
Omega2 = (200/1400) Omega1;
delta = N[1/(StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
ncycles = Floor[RFfreq * t90pulse];
ncyclesS = Floor[RFfreq * (t90pulse - 2 * tOffset)];
ncyclesL = Floor[RFfreq * (t90pulse + 2 * tOffset)];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainderS =
 Round[StepsPerCycle (RFfreq * (t90pulse - 2 * tOffset) - ncyclesS)];
pulseRemainderL =
 Round[StepsPerCycle (RFfreq * (t90pulse + 2 * tOffset) - ncyclesL)];
InitTime = Round[StepsPerCycle * ((((101752605 * 10^-6) / 100) * RFfreq) -
     Floor[((101752605 * 10^-6) / 100) * RFfreq])];
C1 = N[QQGa q^2 RR/(4 π eGaAs h), ndigit];
```

```
LuminPolarization = N[1 / 2];
a0 = N[70 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
time = InitTime;
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
phase = Upulse . Iz . Conjugate[Transpose[Upulse]];
coeffIx = Apply[Plus, Flatten[phase Ix]] / rhonorm;
coeffIy = Apply[Plus, Flatten[phase Iy]] / rhonorm;
IdetectInit = (Ix coeffIx - Iy coeffIy) / Sqrt[coeffIx^2 + coeffIy^2] +
  i (Ix coeffIy + Iy coeffIx) / Sqrt[coeffIx^2 + coeffIy^2];
Do[
 Do[
  Do[
   rr = N[0.00000001 + Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEoff = N[1 / (a0 z)^2, ndigit]; EEon = EEoff Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   Voff[[2, 2]] = C1 EEoff k / rr; Voff[[1, 3]] = NhalfSqrt2 C1 EEoff (-j + i) / rr;
   Voff[[2, 3]] = NhalfSqrt2 C1 EEoff (j + i) / rr;
```

```
ks = ksconst Exp[-2 z];
Hevolutnon = (Larmor - ks) Iz +
  0 * (Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd/3));
Hevolutnoff =
 Larmor Iz + 0 * (Voff[[2, 2]] IySqrdminusIxSqrd + Voff[[1, 3]] IzIxplusIxIz +
     Voff[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd/3));
{valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
{valsoff, vecsoff} = Eigensystem[Hevolutnoff]; valsoff = Re[valsoff];
UevonL = Transpose[vecson] .
  DiagonalMatrix[Exp[-I N2Pi valson tWinL]] . Conjugate[vecson];
UevoffL = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL]] . Conjugate[vecsoff];
UevoffL1 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL1]] . Conjugate[vecsoff];
UevoffL2 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL2]] . Conjugate[vecsoff];
UevoffLS = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL/2]] . Conjugate[vecsoff];
UevoffS1 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinS1]] . Conjugate[vecsoff];
UevoffS2 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinS2]] . Conjugate[vecsoff];
t1on = Exp[4 z] t1OnZero;
Ts = (t1Off t1on) / (t1Off + t1on);
norm =
 1/rhonorm (8 Ts (1 - Exp[-tONP / Ts]) (1 - Exp[-t2max / t1on]) Exp[-2 z]);
Idetect = Transpose[IdetectInit];
rhozero = Iz;
time = InitTime;
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
```

```
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rhozero = Upulse . rhozero . Conjugate[Transpose[Upulse]];
Uclsw16 = UevoffLS;
time = time + pulseRemainder + Round[(tWinL / 2) / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
```

```
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
```

```
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
```

```
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
```

```
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
```

```
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
```

```
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffLS . Upulse . Uclsw16;
Uclsw16adj = Conjugate[Transpose[Uclsw16]];
rho = rhozero;
Do[
 data1[[t1]] = norm Apply[Plus, Flatten[rho Idetect]];
 rho = Uclsw16 . rho . Uclsw16adj;
, {t1, 1, npts, 1}]
, {k, 0, 0, 4}]
, {j, 0, 0, 4}]
, {i, 0, 0, 4}];
```

"Spin-Lock Detected CLSW-16" with light on/off in different windows and pulse offsets to create an Average-Hamiltonian proportional to Iz. Signal is averaged in t2 to yield data1[[t1]].

Note that pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase then Y is $-\pi/2$ and $\overline{Y}$ is $\pi/2$. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
t90pulse = 29/10^7;
resonant90time = 289/10^8;
RFfreqAs75 = 1825000;
RFfreq = 3200000;
StepsPerCycle = 50;
tSpinLock = 1 / GCD[RFfreq, RFfreqAs75];
nStepsSpinLock = StepsPerCycle * tSpinLock * RFfreq;
tSpinLock = N[tSpinLock, ndigit];
RFfreqAs75 = N[RFfreqAs75, ndigit];
RFfreq = N[RFfreq, ndigit];
Larmor = N[3200000, ndigit];
tONP = 5;
t1OnZero = N[17 / 100];
t2max = 2;
nmax = 10;                          (* nmax should be an even number here *)
npts = 2 * 512;
UniformOmegaQ = N[0];
tcycle = N[120 10^-6, ndigit];
tOffset = N[15000 10^-11, ndigit];
tWinS = (tcycle / 24) - t90pulse;
tWinL = (tcycle / 12) - t90pulse;
tWinS1 = tWinS - tOffset;
tWinS2 = tWinS + tOffset;
tWinL1 = tWinL - 2 * tOffset;
tWinL2 = tWinL + 2 * tOffset;
SW = 1 / tcycle;
```

```
Omega1 = N[1 / (4 resonant90time), ndigit];
Omega2 = (160 / 1307) Omega1;
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
ncycles = Floor[RFfreq * t90pulse];
ncyclesS = Floor[RFfreq * (t90pulse - 2 * tOffset)];
ncyclesL = Floor[RFfreq * (t90pulse + 2 * tOffset)];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainderS =
 Round[StepsPerCycle (RFfreq * (t90pulse - 2 * tOffset) - ncyclesS)];
pulseRemainderL =
 Round[StepsPerCycle (RFfreq * (t90pulse + 2 * tOffset) - ncyclesL)];
InitTime =
 Round[StepsPerCycle * ((RFfreq * 34 * 10^-7) - Floor[RFfreq * 34 * 10^-7])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 2];
ONPInf = 5 (LuminPolarization / 2);
LatticeInf = N[10^-5];
tElectLifetime = 10^-13;
a0 = N[70 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
time = InitTime;
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
```

```
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
phase = Upulse . Iz . Conjugate[Transpose[Upulse]];
coeffIx = Apply[Plus, Flatten[phase Ix]] / rhonorm;
coeffIy = Apply[Plus, Flatten[phase Iy]] / rhonorm;
IdetectInit = (Ix coeffIx - Iy coeffIy) / Sqrt[coeffIx^2 + coeffIy^2];
Do[
 Do[
  Do[
   rr = N[Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEoff = N[1/(a0 z)^2, ndigit]; EEon = EEoff Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i) / rr;
   Voff[[2, 2]] = C1 EEoff k / rr; Voff[[1, 3]] = NhalfSqrt2 C1 EEoff (-j + i) / rr;
   Voff[[2, 3]] = NhalfSqrt2 C1 EEoff (j + i) / rr;
   ks = ksconst Exp[-2 z];
   HqOn = Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd / 3);
   Hevolutnon = (Larmor - ks) Iz + 0 * HqOn;
   HqOff = Voff[[2, 2]] IySqrdminusIxSqrd + Voff[[1, 3]] IzIxplusIxIz +
     Voff[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd / 3);
   Hevolutnoff = Larmor Iz + 0 * HqOff;
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   {valsoff, vecsoff} = Eigensystem[Hevolutnoff]; valsoff = Re[valsoff];
   UevonL = Transpose[vecson] .
     DiagonalMatrix[Exp[-I N2Pi valson tWinL]] . Conjugate[vecson];
   UevoffL = Transpose[vecsoff] .
     DiagonalMatrix[Exp[-I N2Pi valsoff tWinL]] . Conjugate[vecsoff];
   UevoffL1 = Transpose[vecsoff] .
     DiagonalMatrix[Exp[-I N2Pi valsoff tWinL1]] . Conjugate[vecsoff];
   UevoffL2 = Transpose[vecsoff] .
     DiagonalMatrix[Exp[-I N2Pi valsoff tWinL2]] . Conjugate[vecsoff];
   UevoffLS = Transpose[vecsoff] .
     DiagonalMatrix[Exp[-I N2Pi valsoff tWinL / 2]] . Conjugate[vecsoff];
   UevoffS1 = Transpose[vecsoff] .
     DiagonalMatrix[Exp[-I N2Pi valsoff tWinS1]] . Conjugate[vecsoff];
   UevoffS2 = Transpose[vecsoff] .
     DiagonalMatrix[Exp[-I N2Pi valsoff tWinS2]] . Conjugate[vecsoff];
```

```
NormHq = (Apply[Plus, Flatten[Abs[HqOn]]])^2;
t1on = Exp[4 z] t1OnZero;
t1Off = 1 / (NormHq * tElectLifetime);
Ts = (t1Off * t1on) / (t1Off + t1on);
Polarization =
 1/(t1Off + t1on) ((1 - Exp[-tONP / Ts]) * (ONPInf * t1Off + LatticeInf * t1on));
Detection = t1on (1 - Exp[-t2max * ((1 / t1on) + (1 / t1Off))]) Exp[-2 z];
norm = 8 Polarization Detection / rhonorm;
rhozero = Iz;
time = InitTime;
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
Uclsw16 = UevoffLS;
time = time + pulseRemainder + Round[(tWinL / 2) / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
```

```
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
```

```
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
```

```
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
```

```
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
```

```
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffLS . Upulse . Uclsw16;
Uclsw16adj = Conjugate[Transpose[Uclsw16]];
time = InitTime + pulseRemainder;
Ucycle = identity;
Do[Htotal = Hevolutnon +
   2 Omega2 (Sin[N2Pi RFfreq delta (time + (tt2 - 1 / 2) / 5) - Pi / 2] +
      Sin[N2Pi RFfreqAs75 delta (time + (tt2 - 1 / 2) / 5) - Pi / 2]) Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta / 5]] . Conjugate[vecs] . Ucycle
, {tt2, 1, StepsPerCycle, 1}];
{vals, vecs} = Eigensystem[Ucycle];
{uvals, uvecs} = Eigensystem[Ucycle];
Idetect = Conjugate[vecs] . IdetectInit . Transpose[vecs];
Do[
 Do[
  If[(Abs[(2 Pi / 5) - Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]]] <=
      2 Pi 19500 / (5 RFfreq))
    || (Abs[(2 Pi (2 RFfreqAs75 - RFfreq) / (5 RFfreq)) -
        Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]]] <=
      2 Pi 19500 / (5 RFfreq))
    , Idetect[[ii, jj]] = Idetect[[ii, jj]], Idetect[[ii, jj]] = N[0, ndigit]];
 , {jj, 1, 4, 1}];
, {ii, 1, 4, 1}];
Idetect = Transpose[vecs] . Idetect . Conjugate[vecs];
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega2 (Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] +
      Sin[N2Pi RFfreqAs75 delta (tt - 1/2) - Pi / 2]) Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + nStepsSpinLock, 1}];
{vals, vecs} = Eigensystem[Ucycle];
```

```
    Idetect = Conjugate[vecs] . Idetect . Transpose[vecs];
    Do[
     Do[
      If[(Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <= 2 Pi 20 * tSpinLock)
       || (Abs[(Pi / 4) - Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]]] <=
          2 Pi 20 * tSpinLock)
       , Idetect[[ii, jj]] = Idetect[[ii, jj]], Idetect[[ii, jj]] = N[0, ndigit]];
      , {jj, 1, 4, 1}];
     , {ii, 1, 4, 1}];
    Idetect = Transpose[vecs] . Idetect . Conjugate[vecs];
    Idetect = Transpose[Idetect];
    Do[
     data1[[t1]] = data1[[t1]] + normApply[Plus, Flatten[rho Idetect]];
     rho = Uclsw16 . rho . Uclsw16adj
    , {t1, 1, npts, 1}]
   , {k, 15, 15, 2}]
  , {j, 15, 15, 2}]
 , {i, 15, 15, 2}];
```

Sum the spectra over the individual sites (i,j,k). An abreviated number of sites is calculated here.

Note that pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase then Y is $-\pi/2$ and $\overline{Y}$ is $\pi/2$. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
(* nmax should be an even
   number here *)

t90pulse = 29/10^7;
resonant90time = 29/10^7;
RFfreqAs75 = 1800000;
RFfreq = 3200000;
StepsPerCycle = 25;
tSpinLock = 1 / GCD[RFfreq, RFfreqAs75];
```

```
nStepsSpinLock = StepsPerCycle * tSpinLock * RFfreq;
tSpinLock = N[tSpinLock, ndigit];
RFfreqAs75 = N[RFfreqAs75, ndigit];
RFfreq = N[RFfreq, ndigit];
Larmor = N[3200000, ndigit];
tONP = 5;
t1OnZero = N[28 / 100];
t2max = 2;
nmax = 10;                          (* nmax should be an even number here *)
npts = 512;
UniformOmegaQ = N[0];
tcycle = N[120 10^-6, ndigit];
tOffset = N[150 10^-9, ndigit];
tWinS = (tcycle / 24) - t90pulse;
tWinL = (tcycle / 12) - t90pulse;
tWinS1 = tWinS - tOffset;
tWinS2 = tWinS + tOffset;
tWinL1 = tWinL - 2 * tOffset;
tWinL2 = tWinL + 2 * tOffset;
SW = 1 / tcycle;
Omega1 = N[1 / (4 resonant90time), ndigit];
Omega2 = (160 / 1307) Omega1;
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
ncycles = Floor[RFfreq * t90pulse];
ncyclesS = Floor[RFfreq * (t90pulse - 2 * tOffset)];
ncyclesL = Floor[RFfreq * (t90pulse + 2 * tOffset)];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainderS =
 Round[StepsPerCycle (RFfreq * (t90pulse - 2 * tOffset) - ncyclesS)];
pulseRemainderL =
 Round[StepsPerCycle (RFfreq * (t90pulse + 2 * tOffset) - ncyclesL)];
InitTime =
 Round[StepsPerCycle * ((RFfreq * 34 * 10^-7) - Floor[RFfreq * 34 * 10^-7])];
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
ONPInf = 5 (LuminPolarization / 2);
LatticeInf = N[10^-5];
tElectLifetime = 500 * 10^-12;
a0 = N[100 / 10^10, ndigit];
```

```
C2 = N[565/(100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed/π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed/π, ndigit];
time = InitTime;
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
phase = Upulse . Iz . Conjugate[Transpose[Upulse]];
coeffIx = Apply[Plus, Flatten[phase Ix]] / rhonorm;
coeffIy = Apply[Plus, Flatten[phase Iy]] / rhonorm;
IdetectInit = (Ix coeffIx - Iy coeffIy) / Sqrt[coeffIx^2 + coeffIy^2];
Do[
 Do[
  Do[
   rr = N[Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEoff = N[1/(a0 z)^2, ndigit]; EEon = EEoff Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k/rr; Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i)/rr;
   Von[[2, 3]] = NhalfSqrt2 C1 EEon (j + i)/rr;
   Voff[[2, 2]] = C1 EEoff k/rr; Voff[[1, 3]] = NhalfSqrt2 C1 EEoff (-j + i)/rr;
   Voff[[2, 3]] = NhalfSqrt2 C1 EEoff (j + i)/rr;
   ks = ksconst Exp[-2 z];
   HqOn = Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
     Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd/3);
```

```
Hevolutnon = (Larmor - ks) Iz + HqOn;
HqOff = Voff[[2, 2]] IySqrdminusIxSqrd + Voff[[1, 3]] IzIxplusIxIz +
  Voff[[2, 3]] IzIyplusIyIz + UniformOmegaQ/4 (IzSqrd - ItotalSqrd / 3);
Hevolutnoff = Larmor Iz + HqOff;
{valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
{valsoff, vecsoff} = Eigensystem[Hevolutnoff]; valsoff = Re[valsoff];
UevonL = Transpose[vecson] .
  DiagonalMatrix[Exp[-I N2Pi valson tWinL]] . Conjugate[vecson];
UevoffL = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL]] . Conjugate[vecsoff];
UevoffL1 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL1]] . Conjugate[vecsoff];
UevoffL2 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL2]] . Conjugate[vecsoff];
UevoffLS = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinL / 2]] . Conjugate[vecsoff];
UevoffS1 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinS1]] . Conjugate[vecsoff];
UevoffS2 = Transpose[vecsoff] .
  DiagonalMatrix[Exp[-I N2Pi valsoff tWinS2]] . Conjugate[vecsoff];
NormHq = (Apply[Plus, Flatten[Abs[HqOn]]])^2;
t1on = Exp[4 z] t1OnZero;
t1Off = 1 / (NormHq * tElectLifetime);
Ts = (t1Off * t1on) / (t1Off + t1on);
Polarization =
  1/(t1Off + t1on) ((1 - Exp[-tONP / Ts]) * (ONPInf * t1Off + LatticeInf * t1on));
Detection = t1on (1 - Exp[-t2max * ((1 / t1on) + (1 / t1Off))]) Exp[-2 z];
norm = 8 Polarization Detection / rhonorm;
rhozero = Iz;
time = InitTime;
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
Uclsw16 = UevoffLS;
time = time + pulseRemainder + Round[(tWinL / 2) / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
```

```
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
```

```
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
```

```
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
```

```
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[
 Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffLS . Upulse . Uclsw16;
Uclsw16adj = Conjugate[Transpose[Uclsw16]];
time = InitTime + pulseRemainder;
Ucycle = identity;
Do[Htotal = Hevolutnon +
    2 Omega2 (Sin[N2Pi RFfreq delta (time + (tt2 - 1 / 2) / 5) - Pi / 2] +
       Sin[N2Pi RFfreqAs75 delta (time + (tt2 - 1 / 2) / 5) - Pi / 2]) Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta / 5]] . Conjugate[vecs] . Ucycle
, {tt2, 1, StepsPerCycle, 1}];
{vals, vecs} = Eigensystem[Ucycle];
{uvals, uvecs} = Eigensystem[Ucycle];
Idetect = Conjugate[vecs] . IdetectInit . Transpose[vecs];
```

```
Do[
 Do[
  If[(Abs[(2 Pi / 5) - Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <=
       2 Pi 195000 / (5 RFfreq))
    || (Abs[(2 Pi (2 RFfreqAs75 - RFfreq) / (5 RFfreq)) -
         Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <=
       2 Pi 195000 / (5 RFfreq))
    , Idetect[[ii, jj]] = Idetect[[ii, jj]], Idetect[[ii, jj]] = N[0, ndigit]];
 , {jj, 1, 4, 1}];
, {ii, 1, 4, 1}];
Idetect = Transpose[vecs] . Idetect . Conjugate[vecs];
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega2 (Sin[N2Pi RFfreq delta (tt - 1/2)] - Pi / 2] +
      Sin[N2Pi RFfreqAs75 delta (tt - 1/2)] - Pi / 2]) Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
   DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + nStepsSpinLock, 1}];
{vals, vecs} = Eigensystem[Ucycle];
Idetect = Conjugate[vecs] . Idetect . Transpose[vecs];
Do[
 Do[
  If[(Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <= 2 Pi 20 * tSpinLock)
    || (Abs[ (Pi / 4) - Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <=
       2 Pi 20 * tSpinLock)
    , Idetect[[ii, jj]] = Idetect[[ii, jj]], Idetect[[ii, jj]] = N[0, ndigit]];
 , {jj, 1, 4, 1}];
, {ii, 1, 4, 1}];
Idetect = Transpose[vecs] . Idetect . Conjugate[vecs];
Idetect = Transpose[Idetect];
Do[
 data1[[t1]] = data1[[t1]] + norm Apply[Plus, Flatten[rho Idetect]];
 rho = Uclsw16 . rho . Uclsw16adj
, {t1, 1, npts, 1}]
, {k, 1, Sqrt[nmax^2 - i^2 - j^2], 2}]
, {j, 1, Sqrt[nmax^2 - i^2], 2}]
, {i, -nmax, nmax, 2}];
```

Sum the spectra over the individual sites (i,j,k). This is the full calculation (every site).

Use an If-loop to assure that each site gets calculated. This is best done when large numbers of sites are to be computed since this does not have weighting factors for nuclei that are bisected by boundaries.

Note that pulse phases (X, Y, $\overline{X}$, $\overline{Y}$) are determined by the relative phase of the rf-since wave used to generate them. However, some backwardsness of *Mathematica* which we do not understand requires that if X is zero relative phase then Y is $-\pi/2$ and $\overline{Y}$ is $\pi/2$. This is the opposite of both the experiment (in C) and the fortran version of this simulation.

```
t90pulse = 29/10^7;
resonant90time = 29/10^7;
RFfreqAs75 = 1800000;
RFfreq = 3200000;
StepsPerCycle = 25;
tSpinLock = 1 / GCD[RFfreq, RFfreqAs75];
nStepsSpinLock = StepsPerCycle * tSpinLock * RFfreq;
tSpinLock = N[tSpinLock, ndigit];
RFfreqAs75 = N[RFfreqAs75, ndigit];
RFfreq = N[RFfreq, ndigit];
Larmor = N[3200000, ndigit];
tONP = 5;
t1OnZero = N[28 / 100];
t2max = 2;
nmax = 58;
npts = 512;
UniformOmegaQ = N[0];
tcycle = N[120 10^-6, ndigit];
tOffset = N[150 10^-9, ndigit];
tWinS = (tcycle / 24) - t90pulse;
tWinL = (tcycle / 12) - t90pulse;
tWinS1 = tWinS - tOffset;
tWinS2 = tWinS + tOffset;
tWinL1 = tWinL - 2 * tOffset;
```

```
tWinL2 = tWinL + 2 * tOffset;
SW = 1 / tcycle;
Omega1 = N[1 / (4 resonant90time), ndigit];
Omega2 = (160 / 1307) Omega1;
delta = N[1 / (StepsPerCycle RFfreq), ndigit];
data1 = Table[N[0, ndigit], {i, npts}];
ncycles = Floor[RFfreq * t90pulse];
ncyclesS = Floor[RFfreq * (t90pulse - 2 * tOffset)];
ncyclesL = Floor[RFfreq * (t90pulse + 2 * tOffset)];
pulseRemainder = Round[StepsPerCycle (RFfreq t90pulse - ncycles)];
pulseRemainderS =
 Round[StepsPerCycle (RFfreq * (t90pulse - 2 * tOffset) - ncyclesS)];
pulseRemainderL =
 Round[StepsPerCycle (RFfreq * (t90pulse + 2 * tOffset) - ncyclesL)];
InitTime = Round[StepsPerCycle * ((RFfreq * 34 * 10^-7) - Floor[RFfreq * 34 * 10^-
C1 = N[QQGa q^2 RR / (4 π eGaAs h), ndigit];
LuminPolarization = N[1 / 8];
ONPInf = 5 (LuminPolarization / 2);
LatticeInf = N[10^-5];
tElectLifetime = 500 * 10^-12;
a0 = N[100 / 10^10, ndigit];
C2 = N[565 / (100 10^10 2 a0), ndigit];
C2cubed = C2^3;
ksconst = N[2 LuminPolarization hyperFineGa71 C2cubed / π, ndigit];
ksconstAs75 = N[2 LuminPolarization hyperFineAs75 C2cubed / π, ndigit];
time = InitTime;
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Larmor Iz + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
  {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
```

```
  Ucycle = Transpose[vecs] .
    DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
phase = Upulse . Iz . Conjugate[Transpose[Upulse]];
coeffIx = Apply[Plus, Flatten[phase Ix]] / rhonorm;
coeffIy = Apply[Plus, Flatten[phase Iy]] / rhonorm;
IdetectInit = (Ix coeffIx - Iy coeffIy) / Sqrt[coeffIx^2 + coeffIy^2];
Do[
 Do[
  Do[
   If[(Abs[i] + Abs[j] + Abs[k] != 0) && (2 * Floor[(i + j + k) / 2] == (i + j + k)),
  {rr = N[Sqrt[i^2 + j^2 + k^2], ndigit]; z = N[C2 rr, ndigit];
   EEoff = N[1 / (a0 z)^2, ndigit]; EEon = EEoff Exp[-2 z] (2 z^2 + 2 z + 1);
   Von[[2, 2]] = C1 EEon k / rr;
     Von[[1, 3]] = NhalfSqrt2 C1 EEon (-j + i) / rr; Von[[2, 3]] = NhalfSqrt2 C1 EEon (j +
                                                                   rr
   Voff[[2, 2]] = C1 EEoff k / rr; Voff[[1, 3]] = NhalfSqrt2 C1 EEoff (-j + i) / rr;
     Voff[[2, 3]] = NhalfSqrt2 C1 EEoff (j + i) / rr;
   ks = ksconst Exp[-2 z];
   HqOn = Von[[2, 2]] IySqrdminusIxSqrd + Von[[1, 3]] IzIxplusIxIz +
       Von[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   Hevolutnon = (Larmor - ks) Iz + HqOn;
   HqOff = Voff[[2, 2]] IySqrdminusIxSqrd + Voff[[1, 3]] IzIxplusIxIz +
       Voff[[2, 3]] IzIyplusIyIz + UniformOmegaQ / 4 (IzSqrd - ItotalSqrd / 3);
   Hevolutnoff = Larmor Iz + HqOff;
   {valson, vecson} = Eigensystem[Hevolutnon]; valson = Re[valson];
   {valsoff, vecsoff} = Eigensystem[Hevolutnoff]; valsoff = Re[valsoff];
   UevonL = Transpose[vecson] .
       DiagonalMatrix[Exp[-I N2Pi valson tWinL]] . Conjugate[vecson];
   UevoffL = Transpose[vecsoff] .
       DiagonalMatrix[Exp[-I N2Pi valsoff tWinL]] . Conjugate[vecsoff];
   UevoffL1 = Transpose[vecsoff] .
       DiagonalMatrix[Exp[-I N2Pi valsoff tWinL1]] . Conjugate[vecsoff];
   UevoffL2 = Transpose[vecsoff] .
       DiagonalMatrix[Exp[-I N2Pi valsoff tWinL2]] . Conjugate[vecsoff];
   UevoffLS = Transpose[vecsoff] .
       DiagonalMatrix[Exp[-I N2Pi valsoff tWinL / 2]] . Conjugate[vecsoff];
   UevoffS1 = Transpose[vecsoff] .
       DiagonalMatrix[Exp[-I N2Pi valsoff tWinS1]] . Conjugate[vecsoff];
```

```
UevoffS2 = Transpose[vecsoff] .
    DiagonalMatrix[Exp[-I N2Pi valsoff tWinS2]] . Conjugate[vecsoff];
NormHq = (Apply[Plus, Flatten[Abs[HqOn]]])^2;
t1on = Exp[4 z] t1OnZero;
t1Off = 1 / (NormHq * tElectLifetime);
Ts = (t1Off * t1on) / (t1Off + t1on);
Polarization =
   1/(t1Off + t1on) ((1 - Exp[-tONP / Ts]) * (ONPInf * t1Off + LatticeInf * t1on));
Detection = t1on (1 - Exp[-t2max * ((1 / t1on) + (1 / t1Off))]) Exp[-2 z];
norm = 8 Polarization Detection / rhonorm;
rhozero = Iz;
time = InitTime;
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
rho = Upulse . rhozero . Conjugate[Transpose[Upulse]];
Uclsw16 = UevoffLS;
time = time + pulseRemainder + Round[(tWinL / 2) / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
```

```
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
```

```
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
```

```
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinL2 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesS];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderS, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS2 . Upulse . Uclsw16;
time = time + pulseRemainderS + Round[tWinS2 / delta];
```

```
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevonL . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinL / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainder + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
```

```
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffL1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinL1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncyclesL];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2)] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainderL, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffS1 . Upulse . Uclsw16;
time = time + pulseRemainderL + Round[tWinS1 / delta];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + StepsPerCycle, 1}];
Upulse = MatrixPower[Ucycle, ncycles];
Ucycle = identity;
Do[Htotal = Hevolutnoff + 2 Omega1 Sin[N2Pi RFfreq delta (tt - 1/2) + Pi / 2] Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + pulseRemainder, 1}];
Upulse = Ucycle . Upulse;
Uclsw16 = UevoffLS . Upulse . Uclsw16;
Uclsw16adj = Conjugate[Transpose[Uclsw16]];
time = InitTime + pulseRemainder;
```

```
Ucycle = identity;
Do[Htotal =
    Hevolutnon + 2 Omega2 (Sin[N2Pi RFfreq delta (time + (tt2 - 1 / 2) / 5) - Pi /
        Sin[N2Pi RFfreqAs75 delta (time + (tt2 - 1 / 2) / 5) - Pi / 2]) Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta / 5]] . Conjugate[vecs] . Ucycle
, {tt2, 1, StepsPerCycle, 1}];
{vals, vecs} = Eigensystem[Ucycle];
{uvals, uvecs} = Eigensystem[Ucycle];
Idetect = Conjugate[vecs] . IdetectInit . Transpose[vecs];
Do[
 Do[
  If[(Abs[(2 Pi / 5) - Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <=
        2 Pi 195000 / (5 RFfreq))
    || (Abs[(2 Pi (2 RFfreqAs75 - RFfreq) / (5 RFfreq)) -
          Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <= 2 Pi 195000 / (5 RFfreq
    , Idetect[[ii, jj]] = Idetect[[ii, jj]], Idetect[[ii, jj]] = N[0, ndigit]];
 , {jj, 1, 4, 1}];
, {ii, 1, 4, 1}];
Idetect = Transpose[vecs] . Idetect . Conjugate[vecs];
Ucycle = identity;
Do[Htotal = Hevolutnon + 2 Omega2 (Sin[N2Pi RFfreq delta (tt - 1/2) - Pi / 2] +
        Sin[N2Pi RFfreqAs75 delta (tt - 1/2) - Pi / 2]) Ix;
 {vals, vecs} = Eigensystem[Htotal]; vals = Re[vals];
 Ucycle = Transpose[vecs] .
     DiagonalMatrix[Exp[-I N2Pi vals delta]] . Conjugate[vecs] . Ucycle
, {tt, time + 1, time + nStepsSpinLock, 1}];
{vals, vecs} = Eigensystem[Ucycle];
Idetect = Conjugate[vecs] . Idetect . Transpose[vecs];
Do[
 Do[
  If[(Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]] <= 2 Pi 20 * tSpinLock)
    || (Abs[ (Pi / 4) -
          Abs[Arg[vals[[ii]] * Conjugate[vals[[jj]]]]] <= 2 Pi 20 * tSpinLock)
   , Idetect[[ii, jj]] = Idetect[[ii, jj]], Idetect[[ii, jj]] = N[0, ndigit]];
 , {jj, 1, 4, 1}];
, {ii, 1, 4, 1}];
Idetect = Transpose[vecs] . Idetect . Conjugate[vecs];
Idetect = Transpose[Idetect];
Do[
 data1[[t1]] = data1[[t1]] + norm Apply[Plus, Flatten[rho Idetect]];
 rho = Uclsw16 . rho . Uclsw16adj
, {t1, 1, npts, 1}]}]
```

```
  , {k, 0, √(nmax^2 - i^2 - j^2) , 1}]
 , {j, 0, √(nmax^2 - i^2) , 1}]
, {i, -nmax, nmax, 1}];
```

Appendix B – Mathematica Code for the Calculation of Spin Diffusion via a One Dimensional Radial Integration of the Spin-Three-Halves Spin Diffusion Equations

Calculate Signal for 250ms, 1s, 3s and 5s of ONP in the Direction of Horizontal and Diagonal Nearest Neighbors

```
ndigit = 16;
h = N[6626 / (1000 10^34), ndigit];
gammaGa71 = N[12984000, ndigit];
electronDensityOnGa71 = N[58 10^31 / 10, ndigit];
hyperFineGa71 =
 N[(2 (4 π) 9274 2 gammaGa71 electronDensityOnGa71) /
    (3 10^7 (1000 10^24) 2), ndigit];
q = N[160219 / (100000 10^19)];
QQGa = N[112 / ((1000 10^24) (100 100))];
eGaAs = N[131 88541878 / (10 10000000 10^12)];
RR = N[(285 10^10 / 100 + 2 (65 10^15) / (10^8 10)) 100];
C1 = N[QQGa q^2 RR / (4 π eGaAs h)];
tIzWindow = N[2 * (120 - 16 * (32 / 10)) / 24, ndigit];
DutyCycle = 2 * tIzWindow / 120;
a0 = N[100 / 10^10, ndigit];
LatticeSpacingGaAs = N[565 / (100 10^10 √2), ndigit];
scale = 1;
Δr = N[LatticeSpacingGaAs * scale, ndigit];
```

```
izRes1 = a0 / Δr;
izRes = a0 / Δr;
Wisotropic = N[(396/1000)^2 973/10 (1/scale)^2, ndigit];
Wisotropic = N[(396/1000)^2 1903/10 (1/scale)^2, ndigit];
Wisotropic = N[(396/1000)^2 473 (1/scale)^2, ndigit];
Wisotropic = N[(396/1000)^2 230 (1/scale)^2, ndigit];
W = Wisotropic;
zMaxTarget = 6;
zmax = Ceiling[zMaxTarget * izRes] / izRes;
izMin = 2;
t2 = 2;
tonp1 = N[250/1000, ndigit];
tonp2 = N[1, ndigit];
tonp3 = N[3, ndigit];
tonp4 = N[5, ndigit];
T1at0 = N[200/1000, ndigit];
tmin = N[40/1000000, ndigit];
tRes = 1/tmin;
tnpts1 = tonp1 * tRes + 1;
tnpts2 = tonp2 * tRes + 1;
tnpts3 = tonp3 * tRes + 1;
tnpts4 = tonp4 * tRes + 1;
znpts = Round[zmax * izRes + 1];
ElectronPolarization = N[30 / 100];
C2 = N[565/(100 10^10 2 a0), ndigit];
```

```
twoC2cubedoverPi = 2 C2^3 / π;
KSconst =
 ElectronPolarization hyperFineGa71 twoC2cubedoverPi;
alpha = ((1 + ElectronPolarization) / (1 - ElectronPolarization));
(* alpha = nplus/nminus *)
norm1 = 1 + alpha + alpha^2 + alpha^3;
ONPinfplus3half = alpha^3 / norm1;
ONPinfplus1half = alpha^2 / norm1;
ONPinfminus1half = alpha^1 / norm1;
ONPinfminus3half = 1 / norm1;
ONPinfPz = ONPinfplus3half + 1/3 ONPinfplus1half -
   1/3 ONPinfminus1half - ONPinfminus3half;
ONPinfPq = ONPinfplus3half - ONPinfplus1half -
   ONPinfminus1half + ONPinfminus3half;
ONPinfPo = 1/3 ONPinfplus3half - ONPinfplus1half +
   ONPinfminus1half - 1/3 ONPinfminus3half;
PolarizePzWorkingA =
 Table[N[0, ndigit], {iz, znpts}, {i, 2}];
PolarizePzWorkingB =
 Table[N[0, ndigit], {iz, znpts}, {i, 2}];
PolarizePqWorkingA =
 Table[N[0, ndigit], {iz, znpts}, {i, 2}];
PolarizePqWorkingB =
 Table[N[0, ndigit], {iz, znpts}, {i, 2}];
```

```
PolarizePoWorkingA =
 Table[N[0, ndigit], {iz, znpts}, {i, 2}];
PolarizePoWorkingB =
 Table[N[0, ndigit], {iz, znpts}, {i, 2}];
Do[
  z = N[(iz - 1) / izRes, ndigit];
  PolarizePzWorkingA[[iz, 1]] = z;
  PolarizePqWorkingA[[iz, 1]] = z;
  PolarizePoWorkingA[[iz, 1]] = z;
  PolarizePzWorkingB[[iz, 1]] = z;
  PolarizePqWorkingB[[iz, 1]] = z;
  PolarizePoWorkingB[[iz, 1]] = z;
, {iz, 1, znpts}];
Do[
 Ga71 = Table[N[0, ndigit], {iz, znpts}];
 t = N[(tt - 1) / tRes, ndigit];
 Do[
  z = N[(iz - 1) / izRes, ndigit];
  thetaONP = ArcTan[
    Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2) / (3200000)];
  ONPeff = Cos[thetaONP]^2;
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPz * ONPeff - PolarizePzWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPq * ONPeff - PolarizePqWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
```

```
 PolarizePzWorkingA[[iz, 2]] =
  PolarizePzWorkingA[[iz, 2]] +
   (ONPinfPo * ONPeff - PolarizePoWorkingA[[iz, 2]]) *
    (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff)/T1at0])
, {iz, izMin, znpts}];
PolarizePzWorkingB = PolarizePzWorkingA;
PolarizePqWorkingB = PolarizePqWorkingA;
PolarizePoWorkingB = PolarizePoWorkingA;
z = izMin / izRes;
zz = (izMin + 1) / izRes;
Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1))/(a0 z)^2;
Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1))/(a0 zz)^2;
theta = ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
    Sqrt[(KSconst Exp[-2 * zz])^2 + Qszz^2])];
eff = Sin[theta]^2;
PolarizePzWorkingB[[izMin, 2]] =
 PolarizePzWorkingA[[izMin, 2]] +
  W * eff/tRes (5 ( PolarizePzWorkingA[[izMin + 1, 2]] -
       PolarizePzWorkingA[[izMin, 2]]) +
     2 (PolarizePqWorkingA[[izMin + 1, 2]] *
        PolarizePzWorkingA[[izMin, 2]] -
       PolarizePqWorkingA[[izMin, 2]] *
        PolarizePzWorkingA[[izMin + 1, 2]]));
PolarizePqWorkingB[[izMin, 2]] =
 PolarizePqWorkingA[[izMin, 2]] +
  W * eff/tRes (-15 PolarizePqWorkingA[[izMin, 2]] +
```

```
      6 PolarizePqWorkingA[[izMin, 2]] *
       PolarizePqWorkingA[[izMin + 1, 2]] -
      27/5 (PolarizePoWorkingA[[izMin, 2]] -
         2 PolarizePzWorkingA[[izMin, 2]]) *
       PolarizePzWorkingA[[izMin + 1, 2]]);
PolarizePoWorkingB[[izMin, 2]] =
 PolarizePoWorkingA[[izMin, 2]] +
  W * eff/tRes (-30 PolarizePoWorkingA[[izMin, 2]] +
     12 PolarizePoWorkingA[[izMin, 2]] *
      PolarizePqWorkingA[[izMin + 1, 2]] +
     6 PolarizePqWorkingA[[izMin, 2]] *
      PolarizePzWorkingA[[izMin + 1, 2]]);
z = znpts / izRes;
zz = (znpts + 1) / izRes;
Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)/(a0 z)^2);
Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1)/(a0 zz)^2);
theta = ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
     Sqrt[(KSconst Exp[-2 * zz])^2 + Qszz^2])];
eff = Sin[theta]^2;
PolarizePzWorkingB[[znpts, 2]] =
 PolarizePzWorkingA[[znpts, 2]] +
  W * eff/tRes (5 (PolarizePzWorkingA[[znpts - 1, 2]] -
       PolarizePzWorkingA[[znpts, 2]]) +
    2 (PolarizePqWorkingA[[znpts - 1, 2]] *
       PolarizePzWorkingA[[znpts, 2]] -
       PolarizePqWorkingA[[znpts, 2]] *
```

```
          PolarizePzWorkingA[[znpts - 1, 2]]));
PolarizePqWorkingB[[znpts, 2]] =
 PolarizePqWorkingA[[znpts, 2]] +
  (W * eff)/tRes (-15 PolarizePqWorkingA[[znpts, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] -
    (27/5) (PolarizePoWorkingA[[znpts, 2]] -
       2 PolarizePzWorkingA[[znpts, 2]]) *
     PolarizePzWorkingA[[znpts - 1, 2]]);
PolarizePoWorkingB[[znpts, 2]] =
 PolarizePoWorkingA[[znpts, 2]] +
  (W * eff)/tRes (-30 PolarizePoWorkingA[[znpts, 2]] +
    12 PolarizePoWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePzWorkingA[[znpts - 1, 2]]);
Do[
 z = iz / izRes;
 zz = (iz + 1) / izRes;
 Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1))/(a0 z)^2;
 Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1))/(a0 zz)^2;
 theta =
  ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
      Sqrt[((KSconst Exp[-2 * zz])^2 + Qszz^2)])];
 eff = Sin[theta]^2;
 PolarizePzWorkingB[[iz, 2]] =
```

```
 PolarizePzWorkingA[[iz, 2]] +
  W * eff
  ------- (5 ( PolarizePzWorkingA[[iz + 1, 2]] +
   tRes
        PolarizePzWorkingA[[iz - 1, 2]] -
        2 PolarizePzWorkingA[[iz, 2]]) +
     2 (PolarizePqWorkingA[[iz + 1, 2]] *
         PolarizePzWorkingA[[iz, 2]] -
        PolarizePqWorkingA[[iz, 2]] *
         PolarizePzWorkingA[[iz + 1, 2]]) +
     2 (PolarizePqWorkingA[[iz - 1, 2]] *
         PolarizePzWorkingA[[iz, 2]] -
        PolarizePqWorkingA[[iz, 2]] *
         PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePqWorkingB[[iz, 2]] =
 PolarizePqWorkingA[[iz, 2]] +
  W * eff
  ------- (-30 PolarizePqWorkingA[[iz, 2]] +
   tRes
    6 PolarizePqWorkingA[[iz, 2]]
     (PolarizePqWorkingA[[iz + 1, 2]] +
       PolarizePqWorkingA[[iz - 1, 2]]) -
    27
    -- (PolarizePoWorkingA[[iz, 2]] -
    5
       2 PolarizePzWorkingA[[iz, 2]]) *
     ( PolarizePzWorkingA[[iz + 1, 2]] +
       PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePoWorkingB[[iz, 2]] =
 PolarizePoWorkingA[[iz, 2]] +
  W * eff
  ------- (-60 PolarizePoWorkingA[[iz, 2]] +
   tRes
    12 PolarizePoWorkingA[[iz, 2]]
     ( PolarizePqWorkingA[[iz + 1, 2]] +
       PolarizePqWorkingA[[iz - 1, 2]]) +
    6 PolarizePqWorkingA[[iz, 2]]
     ( PolarizePzWorkingA[[iz + 1, 2]] +
```

```
            PolarizePzWorkingA[[iz - 1, 2]]))
  , {iz, izMin + 1, znpts - 1}];
  PolarizePzWorkingA = PolarizePzWorkingB;
  PolarizePqWorkingA = PolarizePqWorkingB;
  PolarizePoWorkingA = PolarizePoWorkingB
 , {tt, 1, tnpts1}];
PolarizePz1 = PolarizePzWorkingA;
PolarizePq1 = PolarizePqWorkingA;
PolarizePo1 = PolarizePoWorkingA;
Do[
 Ga71 = Table[N[0, ndigit], {iz, znpts}];
 t = N[(tt - 1) / tRes, ndigit];
 Do[
  z = N[(iz - 1) / izRes, ndigit];
  thetaONP = ArcTan[
    Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2 / (3200000)];
  ONPeff = Cos[thetaONP]^2;
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPz * ONPeff - PolarizePzWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPq * ONPeff - PolarizePqWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPo * ONPeff - PolarizePoWorkingA[[iz, 2]]) *
```

$$\left(1 - \text{Exp}\left[\frac{-\frac{1}{\text{tRes}} * \text{Exp}[-4\,z] * \text{ONPeff}}{\text{T1at0}}\right]\right)$$

```
, {iz, izMin, znpts}];
PolarizePzWorkingB = PolarizePzWorkingA;
PolarizePqWorkingB = PolarizePqWorkingA;
PolarizePoWorkingB = PolarizePoWorkingA;
z = izMin / izRes;
zz = (izMin + 1) / izRes;
```

$$\text{Qsz} = \sqrt{12}\ \text{C1}\left(\frac{\text{Exp}[-2\,z]\ (2\,z^2 + 2\,z + 1)}{(\text{a0}\,z)^2}\right);$$

$$\text{Qszz} = \sqrt{12}\ \text{C1}\left(\frac{\text{Exp}[-2\,zz]\ (2\,zz^2 + 2\,zz + 1)}{(\text{a0}\,zz)^2}\right);$$

$$\text{theta} = \text{ArcTan}\left[(1239) \Big/ \left(\sqrt{(\text{KSconst}\,\text{Exp}[-2 * z])^2 + \text{Qsz}^2} - \sqrt{(\text{KSconst}\,\text{Exp}[-2 * zz])^2 + \text{Qszz}^2}\right)\right];$$

```
eff = Sin[theta]^2;
PolarizePzWorkingB[[izMin, 2]] =
 PolarizePzWorkingA[[izMin, 2]] +
  W * eff / tRes (5 ( PolarizePzWorkingA[[izMin + 1, 2]] -
       PolarizePzWorkingA[[izMin, 2]]) +
    2 (PolarizePqWorkingA[[izMin + 1, 2]] *
        PolarizePzWorkingA[[izMin, 2]] -
       PolarizePqWorkingA[[izMin, 2]] *
        PolarizePzWorkingA[[izMin + 1, 2]]));
PolarizePqWorkingB[[izMin, 2]] =
 PolarizePqWorkingA[[izMin, 2]] +
  W * eff / tRes (-15 PolarizePqWorkingA[[izMin, 2]] +
    6 PolarizePqWorkingA[[izMin, 2]] *
     PolarizePqWorkingA[[izMin + 1, 2]] -
```

```
        27
        --  (PolarizePoWorkingA[[izMin, 2]] -
        5
          2 PolarizePzWorkingA[[izMin, 2]]) *
         PolarizePzWorkingA[[izMin + 1, 2]]);
PolarizePoWorkingB[[izMin, 2]] =
 PolarizePoWorkingA[[izMin, 2]] +
   W * eff
   ------- (-30 PolarizePoWorkingA[[izMin, 2]] +
    tRes
     12 PolarizePoWorkingA[[izMin, 2]] *
      PolarizePqWorkingA[[izMin + 1, 2]] +
     6 PolarizePqWorkingA[[izMin, 2]] *
      PolarizePzWorkingA[[izMin + 1, 2]]);
z = znpts / izRes;
zz = (znpts + 1) / izRes;
Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2;
Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1)) / (a0 zz)^2;
theta = ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
      Sqrt[(KSconst Exp[-2 * zz])^2 + Qszz^2])];
eff = Sin[theta]^2;
PolarizePzWorkingB[[znpts, 2]] =
 PolarizePzWorkingA[[znpts, 2]] +
   W * eff
   ------- (5 (PolarizePzWorkingA[[znpts - 1, 2]] -
    tRes
        PolarizePzWorkingA[[znpts, 2]]) +
     2 (PolarizePqWorkingA[[znpts - 1, 2]] *
        PolarizePzWorkingA[[znpts, 2]] -
       PolarizePqWorkingA[[znpts, 2]] *
        PolarizePzWorkingA[[znpts - 1, 2]]));
PolarizePqWorkingB[[znpts, 2]] =
```

```
 PolarizePqWorkingA[[znpts, 2]] +
  W * eff / tRes (-15 PolarizePqWorkingA[[znpts, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] -
    27/5 (PolarizePoWorkingA[[znpts, 2]] -
      2 PolarizePzWorkingA[[znpts, 2]]) *
     PolarizePzWorkingA[[znpts - 1, 2]]);
PolarizePoWorkingB[[znpts, 2]] =
 PolarizePoWorkingA[[znpts, 2]] +
  W * eff / tRes (-30 PolarizePoWorkingA[[znpts, 2]] +
    12 PolarizePoWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePzWorkingA[[znpts - 1, 2]]);
Do[
 z = iz / izRes;
 zz = (iz + 1) / izRes;
 Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2;
 Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1)) / (a0 zz)^2;
 theta =
  ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
      Sqrt[((KSconst Exp[-2 * zz])^2 + Qszz^2)])];
 eff = Sin[theta]^2;
 PolarizePzWorkingB[[iz, 2]] =
  PolarizePzWorkingA[[iz, 2]] +
```

```
    W * eff
    ------- (5 ( PolarizePzWorkingA[[iz + 1, 2]] +
     tRes
          PolarizePzWorkingA[[iz - 1, 2]] -
          2 PolarizePzWorkingA[[iz, 2]]) +
       2 (PolarizePqWorkingA[[iz + 1, 2]] *
           PolarizePzWorkingA[[iz, 2]] -
          PolarizePqWorkingA[[iz, 2]] *
           PolarizePzWorkingA[[iz + 1, 2]]) +
       2 (PolarizePqWorkingA[[iz - 1, 2]] *
           PolarizePzWorkingA[[iz, 2]] -
          PolarizePqWorkingA[[iz, 2]] *
           PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePqWorkingB[[iz, 2]] =
 PolarizePqWorkingA[[iz, 2]] +
  W * eff
  ------- (-30 PolarizePqWorkingA[[iz, 2]] +
   tRes
    6 PolarizePqWorkingA[[iz, 2]]
     (PolarizePqWorkingA[[iz + 1, 2]] +
       PolarizePqWorkingA[[iz - 1, 2]]) -
     27
     -- (PolarizePoWorkingA[[iz, 2]] -
     5
       2 PolarizePzWorkingA[[iz, 2]]) *
     ( PolarizePzWorkingA[[iz + 1, 2]] +
       PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePoWorkingB[[iz, 2]] =
 PolarizePoWorkingA[[iz, 2]] +
  W * eff
  ------- (-60 PolarizePoWorkingA[[iz, 2]] +
   tRes
    12 PolarizePoWorkingA[[iz, 2]]
     ( PolarizePqWorkingA[[iz + 1, 2]] +
       PolarizePqWorkingA[[iz - 1, 2]]) +
    6 PolarizePqWorkingA[[iz, 2]]
     ( PolarizePzWorkingA[[iz + 1, 2]] +
       PolarizePzWorkingA[[iz - 1, 2]]))
```

```
 , {iz, izMin + 1, znpts - 1}];
 PolarizePzWorkingA = PolarizePzWorkingB;
 PolarizePqWorkingA = PolarizePqWorkingB;
 PolarizePoWorkingA = PolarizePoWorkingB
, {tt, tnpts1 + 1, tnpts2}];
PolarizePz2 = PolarizePzWorkingA;
PolarizePq2 = PolarizePqWorkingA;
PolarizePo2 = PolarizePoWorkingA;
Do[
 Ga71 = Table[N[0, ndigit], {iz, znpts}];
 t = N[(tt - 1) / tRes, ndigit];
 Do[
  z = N[(iz - 1) / izRes, ndigit];
  thetaONP = ArcTan[
    Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2 / (3200000)];
  ONPeff = Cos[thetaONP]^2;
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPz * ONPeff - PolarizePzWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPq * ONPeff - PolarizePqWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPo * ONPeff - PolarizePoWorkingA[[iz, 2]]) *
```

$$\left(1 - \text{Exp}\left[\frac{-\frac{1}{\text{tRes}} * \text{Exp}[-4\,z] * \text{ONPeff}}{\text{T1at0}}\right]\right)$$

```
, {iz, izMin, znpts}];
PolarizePzWorkingB = PolarizePzWorkingA;
PolarizePqWorkingB = PolarizePqWorkingA;
PolarizePoWorkingB = PolarizePoWorkingA;
z = izMin / izRes;
zz = (izMin + 1) / izRes;
```

$$\text{Qsz} = \sqrt{12}\ \text{C1}\left(\frac{\text{Exp}[-2\,z]\ (2\,z^2 + 2\,z + 1)}{(\text{a0}\,z)^2}\right);$$

$$\text{Qszz} = \sqrt{12}\ \text{C1}\left(\frac{\text{Exp}[-2\,zz]\ (2\,zz^2 + 2\,zz + 1)}{(\text{a0}\,zz)^2}\right);$$

$$\text{theta} = \text{ArcTan}\left[(1239)\Big/\left(\sqrt{(\text{KSconst}\,\text{Exp}[-2 * z])^2 + \text{Qsz}^2} - \sqrt{(\text{KSconst}\,\text{Exp}[-2 * zz])^2 + \text{Qszz}^2}\right)\right];$$

```
eff = Sin[theta]^2;
PolarizePzWorkingB[[izMin, 2]] =
 PolarizePzWorkingA[[izMin, 2]] +
  W * eff / tRes (5 ( PolarizePzWorkingA[[izMin + 1, 2]] -
       PolarizePzWorkingA[[izMin, 2]]) +
    2 (PolarizePqWorkingA[[izMin + 1, 2]] *
        PolarizePzWorkingA[[izMin, 2]] -
       PolarizePqWorkingA[[izMin, 2]] *
        PolarizePzWorkingA[[izMin + 1, 2]]));
PolarizePqWorkingB[[izMin, 2]] =
 PolarizePqWorkingA[[izMin, 2]] +
  W * eff / tRes (-15 PolarizePqWorkingA[[izMin, 2]] +
    6 PolarizePqWorkingA[[izMin, 2]] *
     PolarizePqWorkingA[[izMin + 1, 2]] -
```

```
        27
        --  (PolarizePoWorkingA[[izMin, 2]] -
        5
          2 PolarizePzWorkingA[[izMin, 2]]) *
         PolarizePzWorkingA[[izMin + 1, 2]]);
PolarizePoWorkingB[[izMin, 2]] =
 PolarizePoWorkingA[[izMin, 2]] +
   W * eff
   ------- (-30 PolarizePoWorkingA[[izMin, 2]] +
    tRes
     12 PolarizePoWorkingA[[izMin, 2]] *
      PolarizePqWorkingA[[izMin + 1, 2]] +
     6 PolarizePqWorkingA[[izMin, 2]] *
      PolarizePzWorkingA[[izMin + 1, 2]]);
z = znpts / izRes;
zz = (znpts + 1) / izRes;
Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2;
Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1)) / (a0 zz)^2;
theta = ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
      Sqrt[(KSconst Exp[-2 * zz])^2 + Qszz^2])];
eff = Sin[theta]^2;
PolarizePzWorkingB[[znpts, 2]] =
 PolarizePzWorkingA[[znpts, 2]] +
   W * eff
   ------- (5 (PolarizePzWorkingA[[znpts - 1, 2]] -
    tRes
        PolarizePzWorkingA[[znpts, 2]]) +
     2 (PolarizePqWorkingA[[znpts - 1, 2]] *
        PolarizePzWorkingA[[znpts, 2]] -
       PolarizePqWorkingA[[znpts, 2]] *
        PolarizePzWorkingA[[znpts - 1, 2]]));
PolarizePqWorkingB[[znpts, 2]] =
```

```
 PolarizePqWorkingA[[znpts, 2]] +
  W * eff / tRes (-15 PolarizePqWorkingA[[znpts, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] -
     27/5 (PolarizePoWorkingA[[znpts, 2]] -
       2 PolarizePzWorkingA[[znpts, 2]]) *
     PolarizePzWorkingA[[znpts - 1, 2]]);
PolarizePoWorkingB[[znpts, 2]] =
 PolarizePoWorkingA[[znpts, 2]] +
  W * eff / tRes (-30 PolarizePoWorkingA[[znpts, 2]] +
    12 PolarizePoWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePzWorkingA[[znpts - 1, 2]]);
Do[
 z = iz / izRes;
 zz = (iz + 1) / izRes;
 Qsz = √12 C1 (Exp[-2 z] (2 z^2 + 2 z + 1) / (a0 z)^2);
 Qszz = √12 C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1) / (a0 zz)^2);
 theta =
  ArcTan[(1239) / (√((KSconst Exp[-2 * z])^2 + Qsz^2) -
      √((KSconst Exp[-2 * zz])^2 + Qszz^2))];
 eff = Sin[theta]^2;
 PolarizePzWorkingB[[iz, 2]] =
  PolarizePzWorkingA[[iz, 2]] +
```

```
    W * eff
    ------- (5 ( PolarizePzWorkingA[[iz + 1, 2]] +
     tRes
          PolarizePzWorkingA[[iz - 1, 2]] -
          2 PolarizePzWorkingA[[iz, 2]]) +
       2 (PolarizePqWorkingA[[iz + 1, 2]] *
           PolarizePzWorkingA[[iz, 2]] -
          PolarizePqWorkingA[[iz, 2]] *
           PolarizePzWorkingA[[iz + 1, 2]]) +
       2 (PolarizePqWorkingA[[iz - 1, 2]] *
           PolarizePzWorkingA[[iz, 2]] -
          PolarizePqWorkingA[[iz, 2]] *
           PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePqWorkingB[[iz, 2]] =
 PolarizePqWorkingA[[iz, 2]] +
  W * eff
  ------- (-30 PolarizePqWorkingA[[iz, 2]] +
   tRes
    6 PolarizePqWorkingA[[iz, 2]]
     (PolarizePqWorkingA[[iz + 1, 2]] +
       PolarizePqWorkingA[[iz - 1, 2]]) -
     27
     -- (PolarizePoWorkingA[[iz, 2]] -
     5
       2 PolarizePzWorkingA[[iz, 2]]) *
     ( PolarizePzWorkingA[[iz + 1, 2]] +
       PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePoWorkingB[[iz, 2]] =
 PolarizePoWorkingA[[iz, 2]] +
  W * eff
  ------- (-60 PolarizePoWorkingA[[iz, 2]] +
   tRes
    12 PolarizePoWorkingA[[iz, 2]]
     ( PolarizePqWorkingA[[iz + 1, 2]] +
       PolarizePqWorkingA[[iz - 1, 2]]) +
    6 PolarizePqWorkingA[[iz, 2]]
     ( PolarizePzWorkingA[[iz + 1, 2]] +
       PolarizePzWorkingA[[iz - 1, 2]]))
```

```
 , {iz, izMin + 1, znpts - 1}];
 PolarizePzWorkingA = PolarizePzWorkingB;
 PolarizePqWorkingA = PolarizePqWorkingB;
 PolarizePoWorkingA = PolarizePoWorkingB
, {tt, tnpts2 + 1, tnpts3}];
PolarizePz3 = PolarizePzWorkingA;
PolarizePq3 = PolarizePqWorkingA;
PolarizePo3 = PolarizePoWorkingA;
Do[
 Ga71 = Table[N[0, ndigit], {iz, znpts}];
 t = N[(tt - 1) / tRes, ndigit];
 Do[
  z = N[(iz - 1) / izRes, ndigit];
  thetaONP = ArcTan[
    Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2 / (3200000)];
  ONPeff = Cos[thetaONP]^2;
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPz * ONPeff - PolarizePzWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPq * ONPeff - PolarizePqWorkingA[[iz, 2]]) *
     (1 - Exp[(-1/tRes * Exp[-4 z] * ONPeff) / T1at0]);
  PolarizePzWorkingA[[iz, 2]] =
   PolarizePzWorkingA[[iz, 2]] +
    (ONPinfPo * ONPeff - PolarizePoWorkingA[[iz, 2]]) *
```

$$\left(1 - \text{Exp}\left[\frac{-\frac{1}{\text{tRes}} * \text{Exp}[-4\,z] * \text{ONPeff}}{\text{T1at0}}\right]\right)$$

```
, {iz, izMin, znpts}];
PolarizePzWorkingB = PolarizePzWorkingA;
PolarizePqWorkingB = PolarizePqWorkingA;
PolarizePoWorkingB = PolarizePoWorkingA;
z = izMin / izRes;
zz = (izMin + 1) / izRes;
```

$$\text{Qsz} = \sqrt{12}\ \text{C1}\left(\frac{\text{Exp}[-2\,z]\ (2\,z^2 + 2\,z + 1)}{(\text{a0}\,z)^2}\right);$$

$$\text{Qszz} = \sqrt{12}\ \text{C1}\left(\frac{\text{Exp}[-2\,zz]\ (2\,zz^2 + 2\,zz + 1)}{(\text{a0}\,zz)^2}\right);$$

$$\text{theta} = \text{ArcTan}\left[(1239)\Big/\left(\sqrt{(\text{KSconst}\,\text{Exp}[-2*z])^2 + \text{Qsz}^2} - \sqrt{(\text{KSconst}\,\text{Exp}[-2*zz])^2 + \text{Qszz}^2}\right)\right];$$

```
eff = Sin[theta]^2;
PolarizePzWorkingB[[izMin, 2]] =
 PolarizePzWorkingA[[izMin, 2]] +
  W * eff / tRes (5 ( PolarizePzWorkingA[[izMin + 1, 2]] -
       PolarizePzWorkingA[[izMin, 2]]) +
     2 (PolarizePqWorkingA[[izMin + 1, 2]] *
        PolarizePzWorkingA[[izMin, 2]] -
       PolarizePqWorkingA[[izMin, 2]] *
        PolarizePzWorkingA[[izMin + 1, 2]]));
PolarizePqWorkingB[[izMin, 2]] =
 PolarizePqWorkingA[[izMin, 2]] +
  W * eff / tRes (-15 PolarizePqWorkingA[[izMin, 2]] +
     6 PolarizePqWorkingA[[izMin, 2]] *
      PolarizePqWorkingA[[izMin + 1, 2]] -
```

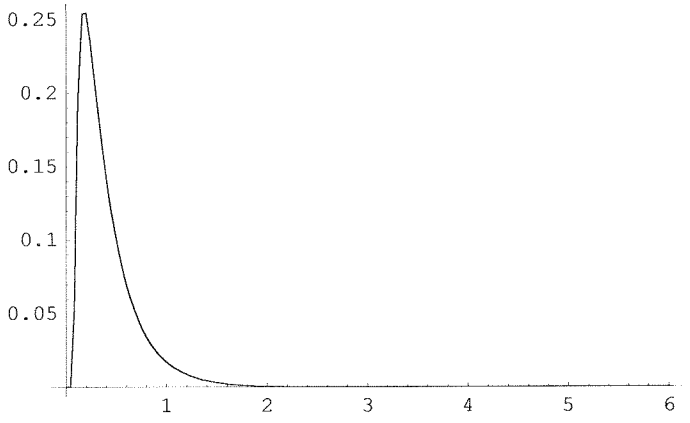
```
        27
        --  (PolarizePoWorkingA[[izMin, 2]] -
        5
          2 PolarizePzWorkingA[[izMin, 2]]) *
         PolarizePzWorkingA[[izMin + 1, 2]]);
PolarizePoWorkingB[[izMin, 2]] =
 PolarizePoWorkingA[[izMin, 2]] +
   W * eff
   ------- (-30 PolarizePoWorkingA[[izMin, 2]] +
    tRes
     12 PolarizePoWorkingA[[izMin, 2]] *
      PolarizePqWorkingA[[izMin + 1, 2]] +
     6 PolarizePqWorkingA[[izMin, 2]] *
      PolarizePzWorkingA[[izMin + 1, 2]]);
z = znpts / izRes;
zz = (znpts + 1) / izRes;
Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1)) / (a0 z)^2;
Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1)) / (a0 zz)^2;
theta = ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
      Sqrt[(KSconst Exp[-2 * zz])^2 + Qszz^2])];
eff = Sin[theta]^2;
PolarizePzWorkingB[[znpts, 2]] =
 PolarizePzWorkingA[[znpts, 2]] +
   W * eff
   ------- (5 (PolarizePzWorkingA[[znpts - 1, 2]] -
    tRes
        PolarizePzWorkingA[[znpts, 2]]) +
     2 (PolarizePqWorkingA[[znpts - 1, 2]] *
        PolarizePzWorkingA[[znpts, 2]] -
       PolarizePqWorkingA[[znpts, 2]] *
        PolarizePzWorkingA[[znpts - 1, 2]]));
PolarizePqWorkingB[[znpts, 2]] =
```

```
 PolarizePqWorkingA[[znpts, 2]] +
  W * eff / tRes (-15 PolarizePqWorkingA[[znpts, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] -
     27/5 (PolarizePoWorkingA[[znpts, 2]] -
       2 PolarizePzWorkingA[[znpts, 2]]) *
     PolarizePzWorkingA[[znpts - 1, 2]]);
PolarizePoWorkingB[[znpts, 2]] =
 PolarizePoWorkingA[[znpts, 2]] +
  W * eff / tRes (-30 PolarizePoWorkingA[[znpts, 2]] +
    12 PolarizePoWorkingA[[znpts, 2]] *
     PolarizePqWorkingA[[znpts - 1, 2]] +
    6 PolarizePqWorkingA[[znpts, 2]] *
     PolarizePzWorkingA[[znpts - 1, 2]]);
Do[
 z = iz / izRes;
 zz = (iz + 1) / izRes;
 Qsz = Sqrt[12] C1 (Exp[-2 z] (2 z^2 + 2 z + 1) / (a0 z)^2);
 Qszz = Sqrt[12] C1 (Exp[-2 zz] (2 zz^2 + 2 zz + 1) / (a0 zz)^2);
 theta =
  ArcTan[(1239) / (Sqrt[(KSconst Exp[-2 * z])^2 + Qsz^2] -
      Sqrt[((KSconst Exp[-2 * zz])^2 + Qszz^2)])];
 eff = Sin[theta]^2;
 PolarizePzWorkingB[[iz, 2]] =
  PolarizePzWorkingA[[iz, 2]] +
```

```
    W * eff
    ------- (5 ( PolarizePzWorkingA[[iz + 1, 2]] +
     tRes
          PolarizePzWorkingA[[iz - 1, 2]] -
          2 PolarizePzWorkingA[[iz, 2]]) +
       2 (PolarizePqWorkingA[[iz + 1, 2]] *
           PolarizePzWorkingA[[iz, 2]] -
          PolarizePqWorkingA[[iz, 2]] *
           PolarizePzWorkingA[[iz + 1, 2]]) +
       2 (PolarizePqWorkingA[[iz - 1, 2]] *
           PolarizePzWorkingA[[iz, 2]] -
          PolarizePqWorkingA[[iz, 2]] *
           PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePqWorkingB[[iz, 2]] =
 PolarizePqWorkingA[[iz, 2]] +
    W * eff
    ------- (-30 PolarizePqWorkingA[[iz, 2]] +
     tRes
      6 PolarizePqWorkingA[[iz, 2]]
       (PolarizePqWorkingA[[iz + 1, 2]] +
         PolarizePqWorkingA[[iz - 1, 2]]) -
      27
      -- (PolarizePoWorkingA[[iz, 2]] -
       5
         2 PolarizePzWorkingA[[iz, 2]]) *
       ( PolarizePzWorkingA[[iz + 1, 2]] +
         PolarizePzWorkingA[[iz - 1, 2]]));
PolarizePoWorkingB[[iz, 2]] =
 PolarizePoWorkingA[[iz, 2]] +
    W * eff
    ------- (-60 PolarizePoWorkingA[[iz, 2]] +
     tRes
      12 PolarizePoWorkingA[[iz, 2]]
       ( PolarizePqWorkingA[[iz + 1, 2]] +
         PolarizePqWorkingA[[iz - 1, 2]]) +
      6 PolarizePqWorkingA[[iz, 2]]
       ( PolarizePzWorkingA[[iz + 1, 2]] +
         PolarizePzWorkingA[[iz - 1, 2]]))
```

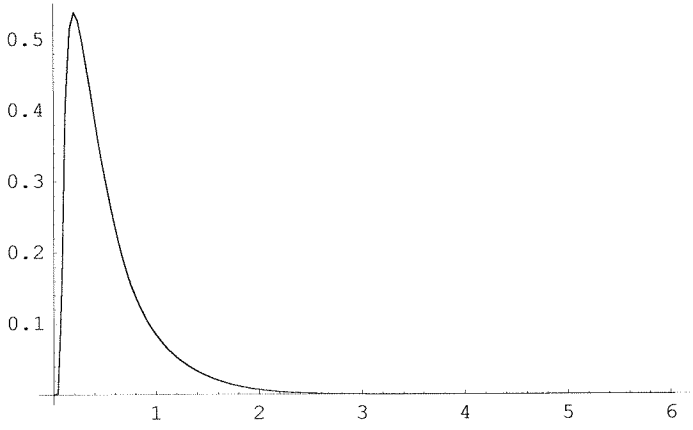
```
 , {iz, izMin + 1, znpts - 1}];
 PolarizePzWorkingA = PolarizePzWorkingB;
 PolarizePqWorkingA = PolarizePqWorkingB;
 PolarizePoWorkingA = PolarizePoWorkingB
, {tt, tnpts3 + 1, tnpts4}];
PolarizePz4 = PolarizePzWorkingA;
PolarizePq4 = PolarizePqWorkingA;
PolarizePo4 = PolarizePoWorkingA;

ListPlot[PolarizePz1, PlotJoined -> True,
 PlotRange -> All]
```



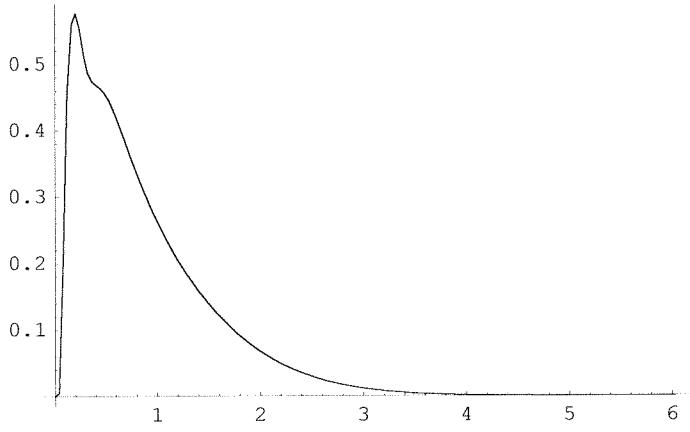
- Graphics -

```
ListPlot[PolarizePz2, PlotJoined -> True,
 PlotRange -> All]
```



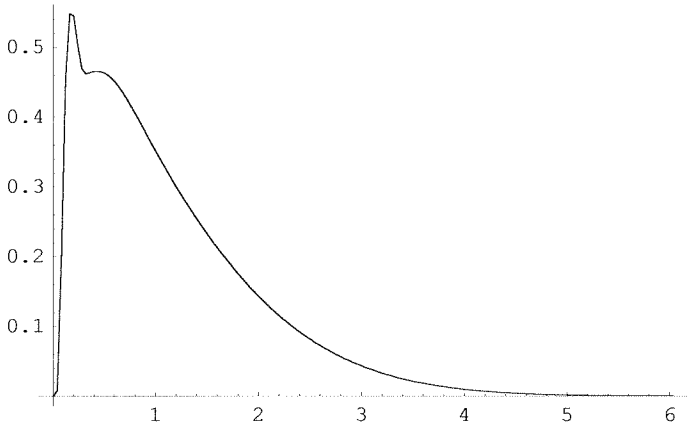
- Graphics -

```
ListPlot[PolarizePz3, PlotJoined -> True,
 PlotRange -> All]
```



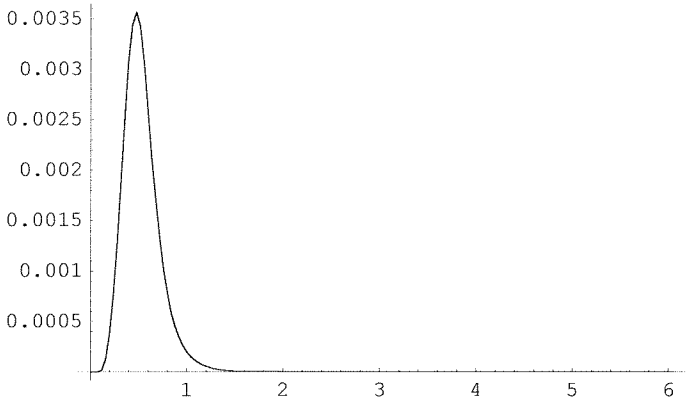
- Graphics -

```
ListPlot[PolarizePz4, PlotJoined -> True,
 PlotRange -> All]
```



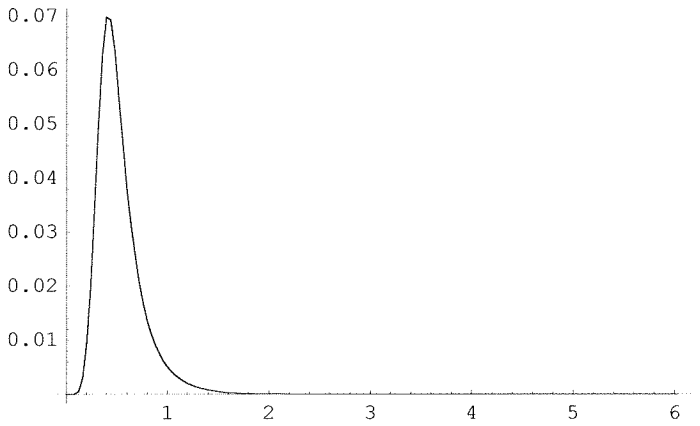
- Graphics -

```
ListPlot[PolarizePq1, PlotJoined -> True,
 PlotRange -> All]
```



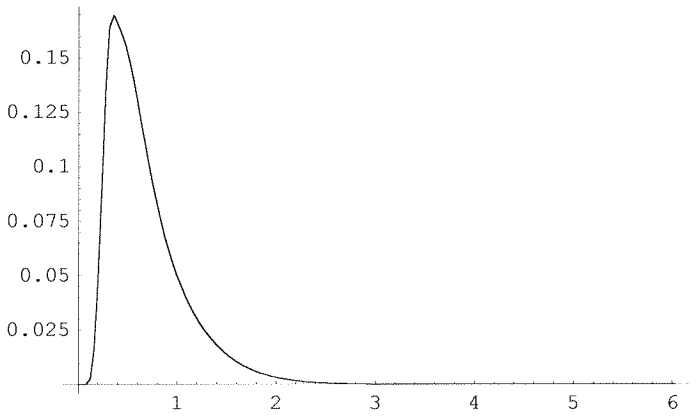
- Graphics -

```
ListPlot[PolarizePq2, PlotJoined -> True,
 PlotRange -> All]
```



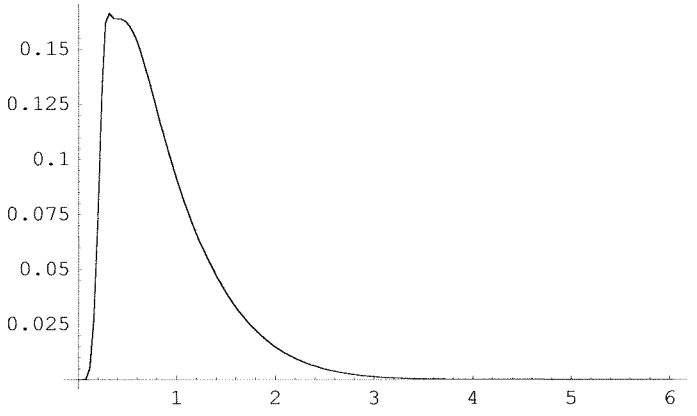

```
- Graphics -
```

```
ListPlot[PolarizePq3, PlotJoined -> True,
 PlotRange -> All]
```



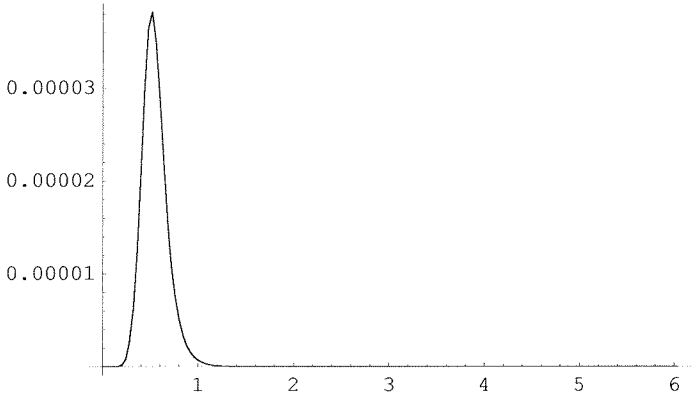

```
- Graphics -
```

```
ListPlot[PolarizePq4, PlotJoined -> True,
 PlotRange -> All]
```



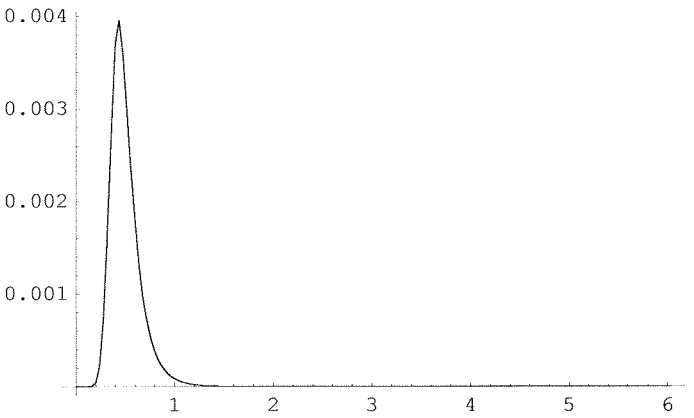

```
- Graphics -
```

```
ListPlot[PolarizePo1, PlotJoined -> True,
 PlotRange -> All]
```



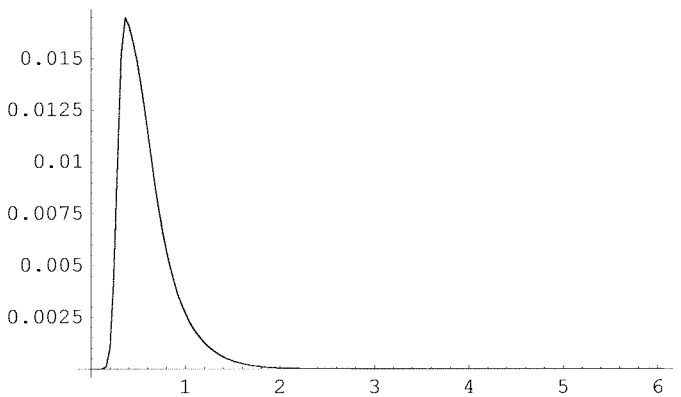

```
- Graphics -
```

```
ListPlot[PolarizePo2, PlotJoined -> True,
 PlotRange -> All]
```



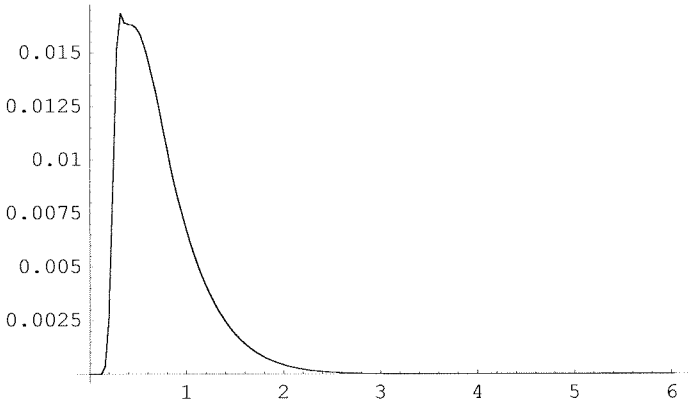

```
- Graphics -
```

```
ListPlot[PolarizePo3, PlotJoined -> True,
 PlotRange -> All]
```




```
- Graphics -
```

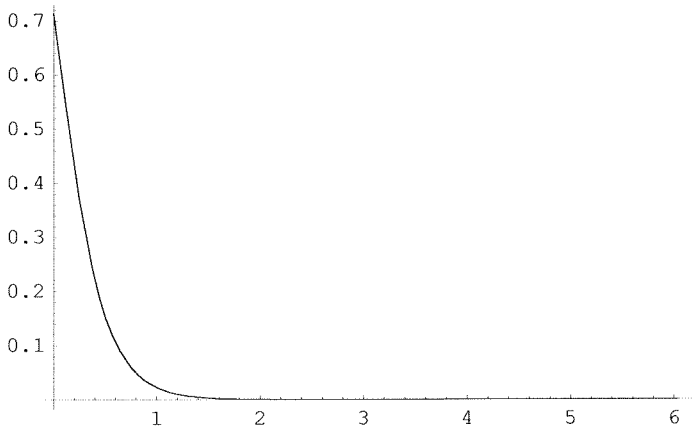
```
ListPlot[PolarizePo4, PlotJoined -> True,
 PlotRange -> All]
```



- Graphics -

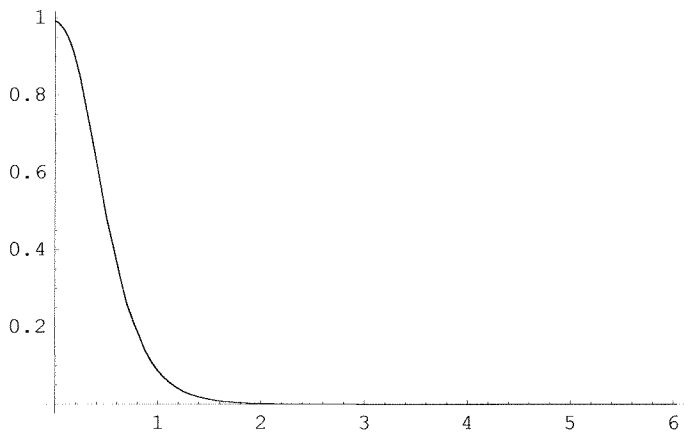
Set u=Exp[-2*r/a0] --> r=(-a0/2) Log[u] --> r^2 = (a0^2) (Log[u]^2)/4
and dr=(-a0/2)(1/u) du. Furthermore, when r=0, u=1, and when r=∞, u=0.

$$\text{Plot}\left[\left(1 - \text{Exp}\left[\frac{-\text{tonp1} * \text{Exp}[-4 * z]}{\text{T1at0}}\right]\right), \{z, 0, \text{zmax}\}, \text{PlotRange} \to \text{All}\right]$$



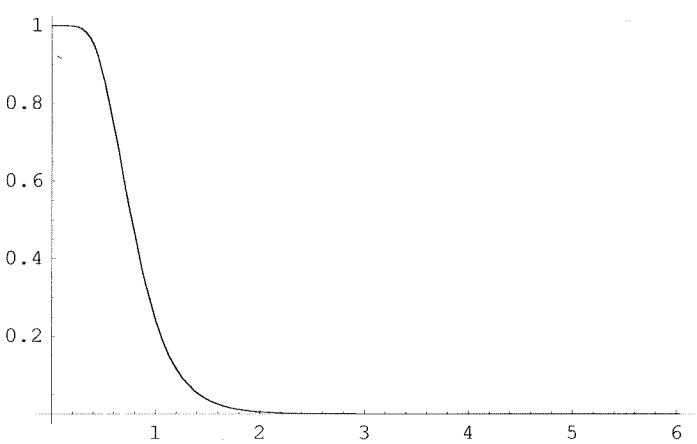
- Graphics -

$$\text{Plot}\left[\left(1 - \text{Exp}\left[\frac{-\text{tonp2} * \text{Exp}[-4 * z]}{\text{T1at0}}\right]\right), \{z, 0, \text{zmax}\}, \text{PlotRange} \rightarrow \text{All}\right]$$



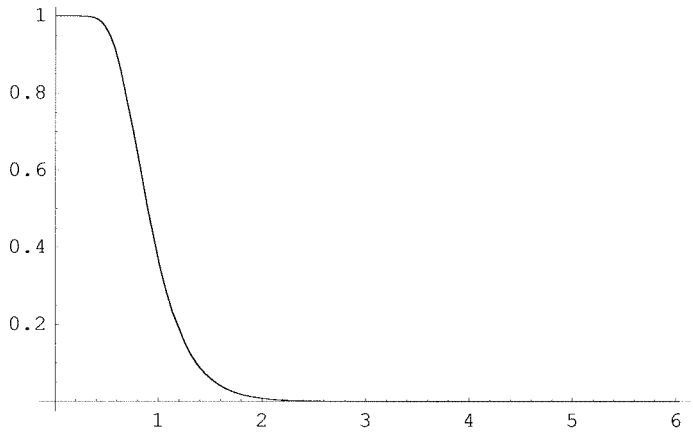
- Graphics -

$$\text{Plot}\left[\left(1 - \text{Exp}\left[\frac{-\text{tonp3} * \text{Exp}[-4 * z]}{\text{T1at0}}\right]\right), \{z, 0, \text{zmax}\}, \text{PlotRange} \rightarrow \text{All}\right]$$



- Graphics -

```
Plot[(1 - Exp[(-tonp4 * Exp[-4 * z])/T1at0]), {z, 0, zmax},
 PlotRange -> All]
```



- Graphics -

If we set u=Exp[-(r/a0)^2] --> r^2=(-a0^2)Log[u] --> r = Sqrt[(-a0^2)Log[u]]
and dr = -(a0/2)(1/u)/Sqrt[-Log[u]]. Furthermore, when r=0, u=1, and when r=∞, u=0.

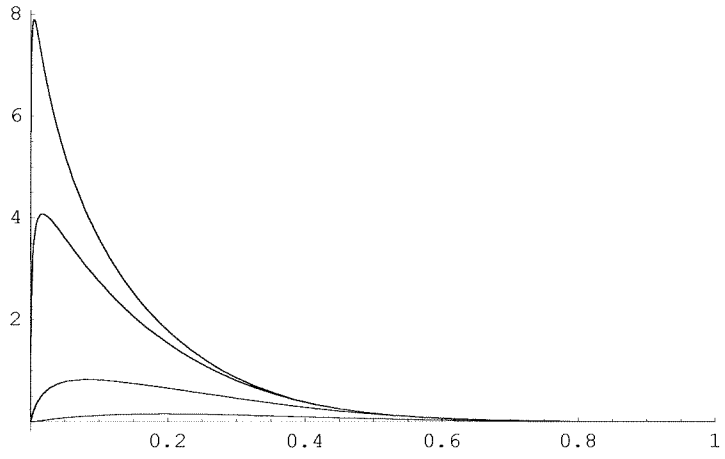
```
uSignal1 = Table[N[0], {iu, 2 * znpts - 1}, {i, 2}];
uSignal2 = Table[N[0], {iu, 2 * znpts - 1}, {i, 2}];
uSignal3 = Table[N[0], {iu, 2 * znpts - 1}, {i, 2}];
uSignal4 = Table[N[0], {iu, 2 * znpts - 1}, {i, 2}];
Do[
 z = N[iz / izRes1];
 u = -Exp[-2 * z];
 uSignal1[[iz, 1]] = u;
 uSignal2[[iz, 1]] = u;
 uSignal3[[iz, 1]] = u;
 uSignal4[[iz, 1]] = u;
, {iz, 1, znpts - 1}];
Do[
 z = N[(iz - 1) / izRes1];
 u = Exp[-2 * z];
```

```
uSignal1[[2 * znpts - iz, 1]] = u;
uSignal2[[2 * znpts - iz, 1]] = u;
uSignal3[[2 * znpts - iz, 1]] = u;
uSignal4[[2 * znpts - iz, 1]] = u;
uSignal1[[2 * znpts - iz, 2]] =
 Re[PolarizePz1[[iz, 2]]] *
  Log[u]^2 * T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]);
uSignal2[[2 * znpts - iz, 2]] =
 Re[PolarizePz2[[iz, 2]]] *
  Log[u]^2 * T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]);
uSignal3[[2 * znpts - iz, 2]] =
 Re[PolarizePz3[[iz, 2]]] *
  Log[u]^2 * T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]);
uSignal4[[2 * znpts - iz, 2]] =
 Re[PolarizePz4[[iz, 2]]] *
  Log[u]^2 * T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]);
, {iz, znpts, 1, -1}]
```

```
Area1 = N[0];
Area2 = N[0];
Area3 = N[0];
Area4 = N[0];
Do[
 Area1 = Area1 + (uSignal1[[iz, 2]] + uSignal1[[iz + 1, 2]]) *
    (uSignal1[[iz + 1, 1]] - uSignal1[[iz, 1]]);
 Area2 = Area2 + (uSignal2[[iz, 2]] + uSignal2[[iz + 1, 2]]) *
    (uSignal2[[iz + 1, 1]] - uSignal2[[iz, 1]]);
 Area3 = Area3 + (uSignal3[[iz, 2]] + uSignal3[[iz + 1, 2]]) *
    (uSignal3[[iz + 1, 1]] - uSignal3[[iz, 1]]);
 Area4 = Area4 + (uSignal4[[iz, 2]] + uSignal4[[iz + 1, 2]]) *
    (uSignal4[[iz + 1, 1]] - uSignal4[[iz, 1]]);
, {iz, 1, Length[uSignal1] - 1}];
```

```
plot1 = ListPlot[Re[uSignal1], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot2 = ListPlot[Re[uSignal2], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot3 = ListPlot[Re[uSignal3], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot4 = ListPlot[Re[uSignal4], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot11 = Plot[Log[u]^2 (1 - Exp[(-tonp1 * u^2)/T1at0])
    T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1},
  PlotPoints -> znpts, DisplayFunction -> Identity];
plot12 = Plot[Log[u]^2 (1 - Exp[(-tonp2 * u^2)/T1at0])
    T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1},
  PlotPoints -> znpts, DisplayFunction -> Identity];
plot13 = Plot[Log[u]^2 (1 - Exp[(-tonp3 * u^2)/T1at0])
    T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1},
  PlotPoints -> znpts, DisplayFunction -> Identity];
plot14 = Plot[Log[u]^2 (1 - Exp[(-tonp4 * u^2)/T1at0]) T1at0/u^2
    (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1}, PlotPoints -> znpts,
  DisplayFunction -> Identity];
```

```
Show[plot1, plot2, plot3,
 plot4, DisplayFunction -> $DisplayFunction,
 PlotRange -> {{0, 1}, All}]
```



- Graphics -

```
Max[Transpose[uSignal1][[2]]] /
 Max[Transpose[uSignal4][[2]]]
Max[Transpose[uSignal2][[2]]] /
 Max[Transpose[uSignal4][[2]]]
Max[Transpose[uSignal3][[2]]] /
 Max[Transpose[uSignal4][[2]]]
Max[Transpose[uSignal4][[2]]] /
 Max[Transpose[uSignal4][[2]]]
```

```
0.0190525

0.10472

0.516737

1.
```

Area1 / Area4

Area2 / Area4

Area3 / Area4

Area4 / Area4

```
0.0610652

0.266193

0.747857

1.
```

$$a0 = N\left[\frac{100}{10^{10}}, ndigit\right];$$

$$LatticeSpacingGaAs = N\left[\frac{565}{100\,10^{10}\sqrt{2}}, ndigit\right];$$

$$scale = 1;$$

$$\Delta r = N[LatticeSpacingGaAs * scale, ndigit];$$

$$izRes1 = a0 / \Delta r;$$

$$izRes = a0 / \Delta r;$$

$$Wisotropic = N\left[\left(\frac{396}{1000}\right)^2 \frac{973}{10}\left(\frac{1}{scale}\right)^2, ndigit\right];$$

$$Wisotropic = N\left[\left(\frac{396}{1000}\right)^2 \frac{1903}{10}\left(\frac{1}{scale}\right)^2, ndigit\right];$$

$$Wisotropic = N\left[\left(\frac{396}{1000}\right)^2 473\left(\frac{1}{scale}\right)^2, ndigit\right];$$

$$Wisotropic = N\left[\left(\frac{396}{1000}\right)^2 230\left(\frac{1}{scale}\right)^2, ndigit\right];$$

$$W = Wisotropic;$$

$$zMaxTarget = 6;$$

$$zmax = Ceiling[zMaxTarget * izRes] / izRes;$$

$$izMin = 2;$$

$$t2 = 2;$$

$$tonp1 = N\left[\frac{250}{1000}, ndigit\right];$$

$$tonp2 = N[1, ndigit];$$

$$tonp3 = N[3, ndigit];$$

$$tonp4 = N[5, ndigit];$$

$$T1at0 = N\left[\frac{200}{1000}, ndigit\right];$$

$$tmin = N\left[\frac{40}{1000000}, ndigit\right];$$

$$tRes = \frac{1}{tmin};$$

$$tnpts1 = tonp1 * tRes + 1;$$

$$tnpts2 = tonp2 * tRes + 1;$$

$$tnpts3 = tonp3 * tRes + 1;$$

$$tnpts4 = tonp4 * tRes + 1;$$

$$znpts = Round[zmax * izRes + 1];$$

$$ElectronPolarization = N[30 / 100];$$

$$C2 = N\left[\frac{565}{100\ 10^{10}\ 2\ a0}, ndigit\right];$$

$$twoC2cubedoverPi = \frac{2\ C2^3}{\pi};$$

$$KSconst = ElectronPolarization\ hyperFineGa71\ twoC2cubedoverPi;$$

$$alpha = \left(\frac{1 + ElectronPolarization}{1 - ElectronPolarization}\right); \qquad (* \text{ alpha = nplus/nminus } *)$$

$$norm1 = 1 + alpha + alpha^2 + alpha^3;$$

$$ONPinfplus3half = \frac{alpha^3}{norm1};$$

$$ONPinfplus1half = \frac{alpha^2}{norm1};$$

$$ONPinfminus1half = \frac{alpha^1}{norm1};$$

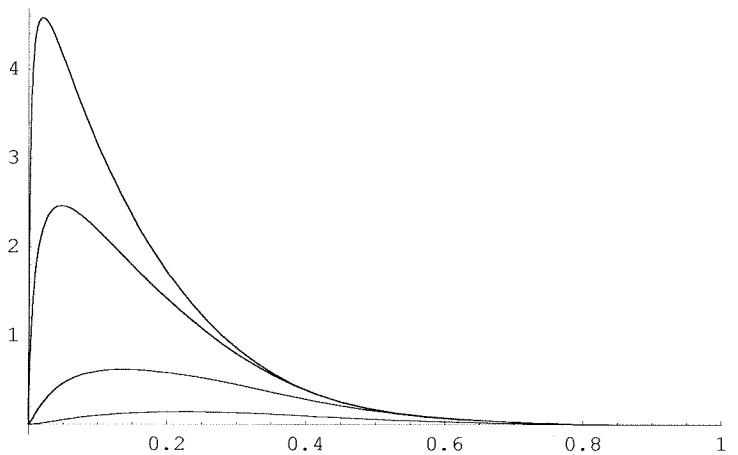
$$ONPinfminus3half = \frac{1}{norm1};$$

$$ONPinfPz = ONPinfplus3half + \frac{1}{3} ONPinfplus1half - \frac{1}{3} ONPinfminus1half - ONPinfminus3half;$$

$$ONPinfPq = ONPinfplus3half - ONPinfplus1half - ONPinfminus1half + ONPinfminus3half;$$

$$ONPinfPo = \frac{1}{3} ONPinfplus3half - ONPinfplus1half + ONPinfminus1half - \frac{1}{3} ONPinfminus3half;$$

```
Show[plot1, plot2, plot3,
 plot4, DisplayFunction -> $DisplayFunction,
 PlotRange -> {{0, 1}, All}]
```



- Graphics -

```
Max[Transpose[uSignal1][[2]]] /
 Max[Transpose[uSignal4][[2]]]
Max[Transpose[uSignal2][[2]]] /
 Max[Transpose[uSignal4][[2]]]
Max[Transpose[uSignal3][[2]]] /
 Max[Transpose[uSignal4][[2]]]
Max[Transpose[uSignal4][[2]]] /
 Max[Transpose[uSignal4][[2]]]
```

```
0.0314094

0.134967

0.537933

1.
```

```
Area1 / Area4
Area2 / Area4
Area3 / Area4
Area4 / Area4
```

```
0.0689294

0.266988

0.711371

1.
```

Real CLSW-16 Experiment - Max. Value

```
{{0.24, 0.048111}, {1.032, 0.22117}, {3.024, 0.60195},
 {4.824, 1}}
```

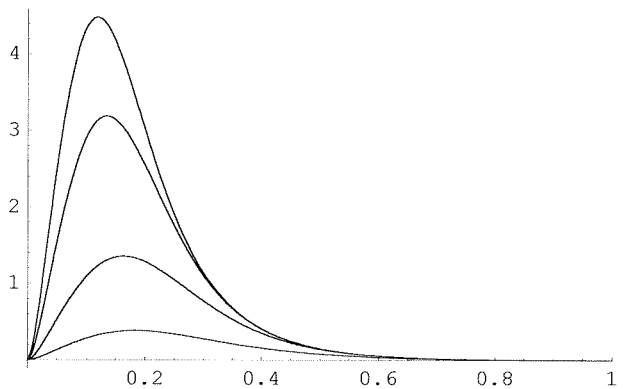
Real CLSW-16 Experiment - Intergral

```
{{0.24, 0.11671}, {1.032, 0.36044}, {3.024, 0.76886},
 {4.824, 1}}
```

ONP Curve

```
{{0.24, 0.12800}, {1.032, 0.40290}, {3.024, 0.79147},
 {4.824, 1}}
```

```
Show[plot11, plot12, plot13,
 plot14, DisplayFunction -> $DisplayFunction,
 PlotRange -> {{0, 1}, All}]
```



- Graphics -

```
zfill = 4;
nzfill = zfill * izRes - 1;
uuSignal1 =
 Table[N[0], {iu, Length[uSignal1] + nzfill}, {i, 2}];
uuSignal2 =
 Table[N[0], {iu, Length[uSignal1] + nzfill}, {i, 2}];
uuSignal3 =
 Table[N[0], {iu, Length[uSignal1] + nzfill}, {i, 2}];
uuSignal4 =
 Table[N[0], {iu, Length[uSignal1] + nzfill}, {i, 2}];
count1 = N[0];
Do[
 uuSignal1[[iz]] =
  uSignal1[[Length[uSignal1] - count1]];
 uuSignal2[[iz]] =
  uSignal2[[Length[uSignal1] - count1]];
 uuSignal3[[iz]] =
  uSignal3[[Length[uSignal1] - count1]];
 uuSignal4[[iz]] =
  uSignal4[[Length[uSignal1] - count1]];
```

```
 count1 = count1 + 1;
, {iz, Length[uuSignal1],
  Length[uuSignal1] - (2 * nzfill + 1), -2}];
Do[
 uuSignal1[[iz]] =
  (uuSignal1[[iz - 1]] + uuSignal1[[iz + 1]]) / 2;
 uuSignal2[[iz]] =
  (uuSignal2[[iz - 1]] + uuSignal2[[iz + 1]]) / 2;
 uuSignal3[[iz]] =
  (uuSignal3[[iz - 1]] + uuSignal3[[iz + 1]]) / 2;
 uuSignal4[[iz]] =
  (uuSignal4[[iz - 1]] + uuSignal4[[iz + 1]]) / 2;
, {iz, Length[uuSignal1] - 1,
  Length[uuSignal1] - 2 * nzfill, -2}];
Do[
 uuSignal1[[iz]] = uSignal1[[iz]];
 uuSignal2[[iz]] = uSignal2[[iz]];
 uuSignal3[[iz]] = uSignal3[[iz]];
 uuSignal4[[iz]] = uSignal4[[iz]];
, {iz, 1, Length[uuSignal1] - 2 * nzfill}];

z3fill = zfill - 1;
nz3fill = z3fill * 2 * izRes - 1;
u3Signal1 =
 Table[N[0], {iu, Length[uuSignal1] + nz3fill}, {i, 2}];
u3Signal2 =
 Table[N[0], {iu, Length[uuSignal1] + nz3fill}, {i, 2}];
u3Signal3 =
 Table[N[0], {iu, Length[uuSignal1] + nz3fill}, {i, 2}];
u3Signal4 =
 Table[N[0], {iu, Length[uuSignal1] + nz3fill}, {i, 2}];
count1 = N[0];
Do[
 u3Signal1[[iz]] =
  uuSignal1[[Length[uuSignal1] - count1]];
```

```
 u3Signal2[[iz]] =
  uuSignal2[[Length[uuSignal1] - count1]];
 u3Signal3[[iz]] =
  uuSignal3[[Length[uuSignal1] - count1]];
 u3Signal4[[iz]] =
  uuSignal4[[Length[uuSignal1] - count1]];
 count1 = count1 + 1;
, {iz, Length[u3Signal1],
  Length[u3Signal1] - (2 * nz3fill + 1), -2}];
Do[
 u3Signal1[[iz]] =
  (u3Signal1[[iz - 1]] + u3Signal1[[iz + 1]]) / 2;
 u3Signal2[[iz]] =
  (u3Signal2[[iz - 1]] + u3Signal2[[iz + 1]]) / 2;
 u3Signal3[[iz]] =
  (u3Signal3[[iz - 1]] + u3Signal3[[iz + 1]]) / 2;
 u3Signal4[[iz]] =
  (u3Signal4[[iz - 1]] + u3Signal4[[iz + 1]]) / 2;
, {iz, Length[u3Signal1] - 1,
  Length[u3Signal1] - 2 * nz3fill, -2}];
Do[
 u3Signal1[[iz]] = uuSignal1[[iz]];
 u3Signal2[[iz]] = uuSignal2[[iz]];
 u3Signal3[[iz]] = uuSignal3[[iz]];
 u3Signal4[[iz]] = uuSignal4[[iz]];
, {iz, 1, Length[u3Signal1] - 2 * nz3fill}];

z4fill = z3fill - 1;
nz4fill = z4fill * 4 * izRes - 1;
u4Signal1 =
 Table[N[0], {iu, Length[u3Signal1] + nz4fill}, {i, 2}];
u4Signal2 =
 Table[N[0], {iu, Length[u3Signal1] + nz4fill}, {i, 2}];
u4Signal3 =
 Table[N[0], {iu, Length[u3Signal1] + nz4fill}, {i, 2}];
```

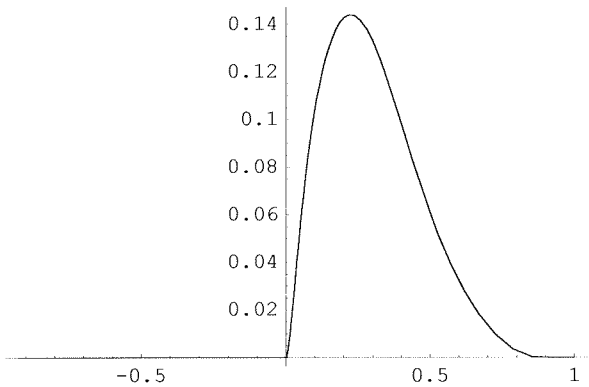
```
u4Signal4 =
 Table[N[0], {iu, Length[u3Signal1] + nz4fill}, {i, 2}];
count1 = N[0];
Do[
 u4Signal1[[iz]] =
  u3Signal1[[Length[u3Signal1] - count1]];
 u4Signal2[[iz]] =
  u3Signal2[[Length[u3Signal1] - count1]];
 u4Signal3[[iz]] =
  u3Signal3[[Length[u3Signal1] - count1]];
 u4Signal4[[iz]] =
  u3Signal4[[Length[u3Signal1] - count1]];
 count1 = count1 + 1;
, {iz, Length[u4Signal1],
  Length[u4Signal1] - (2 * nz4fill + 1), -2}];
Do[
 u4Signal1[[iz]] =
  (u4Signal1[[iz - 1]] + u4Signal1[[iz + 1]]) / 2;
 u4Signal2[[iz]] =
  (u4Signal2[[iz - 1]] + u4Signal2[[iz + 1]]) / 2;
 u4Signal3[[iz]] =
  (u4Signal3[[iz - 1]] + u4Signal3[[iz + 1]]) / 2;
 u4Signal4[[iz]] =
  (u4Signal4[[iz - 1]] + u4Signal4[[iz + 1]]) / 2;
, {iz, Length[u4Signal1] - 1,
  Length[u4Signal1] - 2 * nz4fill, -2}];
Do[
 u4Signal1[[iz]] = u3Signal1[[iz]];
 u4Signal2[[iz]] = u3Signal2[[iz]];
 u4Signal3[[iz]] = u3Signal3[[iz]];
 u4Signal4[[iz]] = u3Signal4[[iz]];
, {iz, 1, Length[u4Signal1] - 2 * nz4fill}];

z5fill = z4fill - 1;
nz5fill = z5fill * 8 * izRes - 1;
```

```
u5Signal1 =
 Table[N[0], {iu, Length[u4Signal1] + nz5fill}, {i, 2}];
u5Signal2 =
 Table[N[0], {iu, Length[u4Signal1] + nz5fill}, {i, 2}];
u5Signal3 =
 Table[N[0], {iu, Length[u4Signal1] + nz5fill}, {i, 2}];
u5Signal4 =
 Table[N[0], {iu, Length[u4Signal1] + nz5fill}, {i, 2}];
count1 = N[0];
Do[
 u5Signal1[[iz]] =
  u4Signal1[[Length[u4Signal1] - count1]];
 u5Signal2[[iz]] =
  u4Signal2[[Length[u4Signal1] - count1]];
 u5Signal3[[iz]] =
  u4Signal3[[Length[u4Signal1] - count1]];
 u5Signal4[[iz]] =
  u4Signal4[[Length[u4Signal1] - count1]];
 count1 = count1 + 1;
, {iz, Length[u5Signal1],
  Length[u5Signal1] - (2 * nz5fill + 1), -2}];
Do[
 u5Signal1[[iz]] =
  (u5Signal1[[iz - 1]] + u5Signal1[[iz + 1]]) / 2;
 u5Signal2[[iz]] =
  (u5Signal2[[iz - 1]] + u5Signal2[[iz + 1]]) / 2;
 u5Signal3[[iz]] =
  (u5Signal3[[iz - 1]] + u5Signal3[[iz + 1]]) / 2;
 u5Signal4[[iz]] =
  (u5Signal4[[iz - 1]] + u5Signal4[[iz + 1]]) / 2;
, {iz, Length[u5Signal1] - 1,
  Length[u5Signal1] - 2 * nz5fill, -2}];
Do[
 u5Signal1[[iz]] = u4Signal1[[iz]];
 u5Signal2[[iz]] = u4Signal2[[iz]];
```

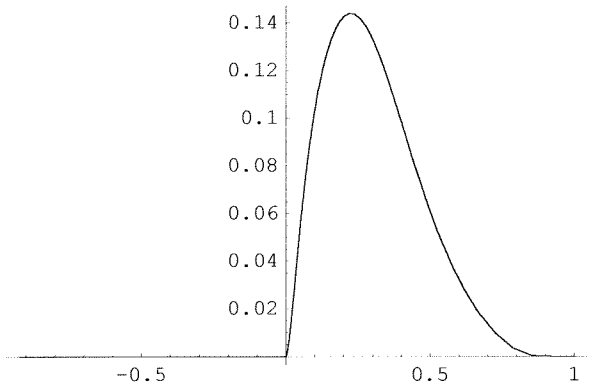
```
 u5Signal3[[iz]] = u4Signal3[[iz]];
 u5Signal4[[iz]] = u4Signal4[[iz]];
, {iz, 1, Length[u5Signal1] - 2 * nz5fill}];

ListPlot[Re[uSignal1], PlotJoined -> True,
 PlotRange -> All]
```



- Graphics -

```
ListPlot[u5Signal1, PlotJoined -> True, PlotRange -> All]
```



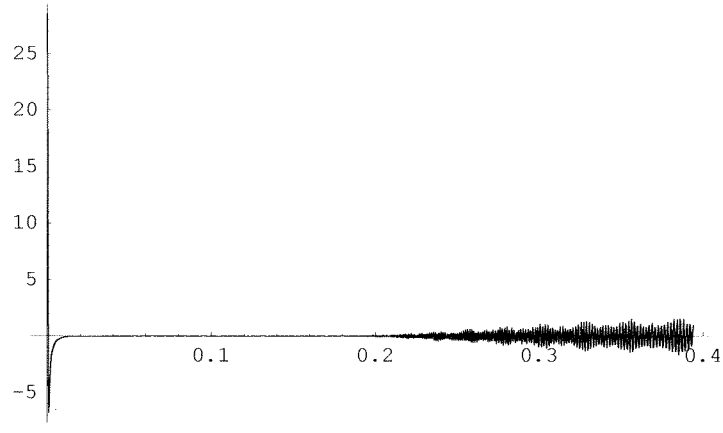
- Graphics -

```
Length[u5Signal1]
```

```
759
```

```
SW = N[KSconst * DutyCycle];
unpts = Length[u5Signal1];
tSignal1 = Table[N[0], {tt, unpts}, {i, 2}];
tSignal2 = Table[N[0], {tt, unpts}, {i, 2}];
tSignal3 = Table[N[0], {tt, unpts}, {i, 2}];
tSignal4 = Table[N[0], {tt, unpts}, {i, 2}];
Do[
 t = (itt - 1) / SW;
 tSignal1[[itt, 1]] = t;
 tSignal2[[itt, 1]] = t;
 tSignal3[[itt, 1]] = t;
 tSignal4[[itt, 1]] = t;
 upast = u5Signal1[[1, 1]];
 u = u5Signal1[[1, 1]];
 Do[
  unext = u5Signal1[[iz + 1 - Floor[iz / unpts], 1]];
  w = (SW / 2) * u;
  wdiff = (SW / 2) * Abs[unext - upast] / 2;
  ftw = 1/Sqrt[2 π] Exp[2 π i w t];
  tSignal1[[itt, 2]] =
   tSignal1[[itt, 2]] + wdiff * ftw * u5Signal1[[iz, 2]];
  tSignal2[[itt, 2]] =
   tSignal2[[itt, 2]] + wdiff * ftw * u5Signal2[[iz, 2]];
  tSignal3[[itt, 2]] =
   tSignal3[[itt, 2]] + wdiff * ftw * u5Signal3[[iz, 2]];
  tSignal4[[itt, 2]] =
   tSignal4[[itt, 2]] + wdiff * ftw * u5Signal4[[iz, 2]];
  upast = u;
  u = unext;
 , {iz, 1, unpts}];
, {itt, 1, unpts}];
```

```
ListPlot[Re[tSignal1], PlotJoined -> True,
 PlotRange -> All]
```



- Graphics -

```
ScalarCoupling = N[60 * DutyCycle];
A = Table[0, {i, 13}, {j, 82}];
Inten = Table[
  (2 - i) (ScalarCoupling * (j - 41) / 2), {i, 2}, {j, 82}];
A[[1, 41]] = 1;
Do[{A[[i, j]] = 1/4 (A[[i - 1, j - 3]] +
      A[[i - 1, j - 1]] + A[[i - 1, j + 1]] + A[[i - 1, j + 3]])},
 {i, 2, 13, 1}, {j, 4, 78, 1}];
Do[{Inten[[2, j]] = Inten[[2, j]] +
    A[[i, j]] Binomial[12, i - 1] 0.604^(13-i) 0.396^(i-1)},
 {j, 80}, {i, 13}];
Clear[A]
```

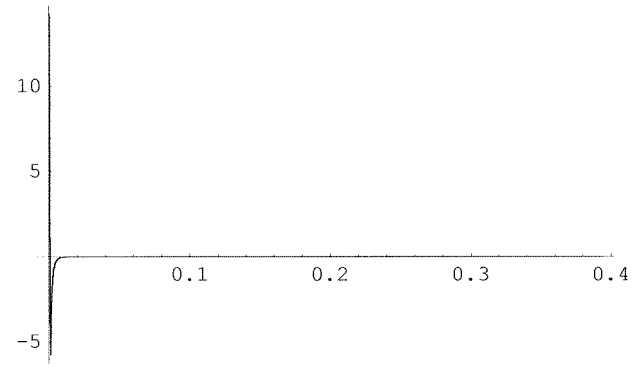
$$\frac{1239}{3} * \text{DutyCycle}$$

```
39.4644
```

```
ttSignal1 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
ttSignal2 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
ttSignal3 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
ttSignal4 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
linewidth = N[40];
ttSignal1[[1, 1]] = tSignal1[[1, 1]];
ttSignal2[[1, 1]] = tSignal2[[1, 1]];
ttSignal3[[1, 1]] = tSignal3[[1, 1]];
ttSignal4[[1, 1]] = tSignal4[[1, 1]];
ttSignal1[[1, 2]] = tSignal1[[1, 2]] / 2;
ttSignal2[[1, 2]] = tSignal2[[1, 2]] / 2;
ttSignal3[[1, 2]] = tSignal3[[1, 2]] / 2;
ttSignal4[[1, 2]] = tSignal4[[1, 2]] / 2;
Do[
 t = (i - 1) / SW;
 lb = Exp[-(π linewidth t)^2 / (4 Log[2])];
 sum1 = N[0];
 Do[
  sum1 = sum1 + Inten[[2, i]] Exp[I * 2 * Pi * Inten[[1, i]] t],
  {i, 5, 77, 1}];
 ttSignal1[[i, 1]] = tSignal1[[i, 1]];
 ttSignal2[[i, 1]] = tSignal2[[i, 1]];
 ttSignal3[[i, 1]] = tSignal3[[i, 1]];
 ttSignal4[[i, 1]] = tSignal4[[i, 1]];
 ttSignal1[[i, 2]] = sum1 * lb * tSignal1[[i, 2]];
 ttSignal2[[i, 2]] = sum1 * lb * tSignal2[[i, 2]];
 ttSignal3[[i, 2]] = sum1 * lb * tSignal3[[i, 2]];
 ttSignal4[[i, 2]] = sum1 * lb * tSignal4[[i, 2]];
, {i, 2, Length[tSignal1]}];
```

```
ttSignal1 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
ttSignal2 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
ttSignal3 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
ttSignal4 = Table[N[0], {tt, Length[tSignal1]}, {i, 2}];
linewidth = N[40];
ttSignal1[[1, 1]] = tSignal1[[1, 1]];
ttSignal2[[1, 1]] = tSignal2[[1, 1]];
ttSignal3[[1, 1]] = tSignal3[[1, 1]];
ttSignal4[[1, 1]] = tSignal4[[1, 1]];
ttSignal1[[1, 2]] = tSignal1[[1, 2]] / 2;
ttSignal2[[1, 2]] = tSignal2[[1, 2]] / 2;
ttSignal3[[1, 2]] = tSignal3[[1, 2]] / 2;
ttSignal4[[1, 2]] = tSignal4[[1, 2]] / 2;
Do[
 t = (i - 1) / SW;
 lb = Exp[-π linewidth t];
 sum1 = N[0];
 Do[
  sum1 = sum1 + Inten[[2, i]] Exp[I * 2 * Pi * Inten[[1, i]] t],
  {i, 5, 77, 1}];
 ttSignal1[[i, 1]] = tSignal1[[i, 1]];
 ttSignal2[[i, 1]] = tSignal2[[i, 1]];
 ttSignal3[[i, 1]] = tSignal3[[i, 1]];
 ttSignal4[[i, 1]] = tSignal4[[i, 1]];
 ttSignal1[[i, 2]] = sum1 * lb * tSignal1[[i, 2]];
 ttSignal2[[i, 2]] = sum1 * lb * tSignal2[[i, 2]];
 ttSignal3[[i, 2]] = sum1 * lb * tSignal3[[i, 2]];
 ttSignal4[[i, 2]] = sum1 * lb * tSignal4[[i, 2]];
, {i, 2, Length[tSignal1]}];
```

```
ListPlot[Re[ttSignal1], PlotJoined -> True,
 PlotRange -> All]
```



- Graphics -

```
wData1 = Table[N[0], {tt, Length[ttSignal1]}, {i, 2}];
wData2 = Table[N[0], {tt, Length[ttSignal1]}, {i, 2}];
wData3 = Table[N[0], {tt, Length[ttSignal1]}, {i, 2}];
wData4 = Table[N[0], {tt, Length[ttSignal1]}, {i, 2}];
Do[
 w = SW/2 (iw - 1 - (Length[ttSignal1] - 1) / 2) /
     (Length[ttSignal1] - 1) / 2;
 wData1[[iw, 1]] = w;
 wData2[[iw, 1]] = w;
 wData3[[iw, 1]] = w;
 wData4[[iw, 1]] = w;
 Do[
  t = (iz - 1) / SW;
  ftw = 1/Sqrt[2 π] Exp[-2 π i w t];
  wData1[[iw, 2]] =
   wData1[[iw, 2]] + ftw/SW * ttSignal1[[iz, 2]];
  wData2[[iw, 2]] =
   wData2[[iw, 2]] + ftw/SW * ttSignal2[[iz, 2]];
  wData3[[iw, 2]] =
   wData3[[iw, 2]] + ftw/SW * ttSignal3[[iz, 2]];
  wData4[[iw, 2]] =
   wData4[[iw, 2]] + ftw/SW * ttSignal4[[iz, 2]];
 , {iz, 1, Length[ttSignal1]}]
, {iw, 1, Length[ttSignal1]}];
```

```
Area1 = N[0];
Area2 = N[0];
Area3 = N[0];
Area4 = N[0];
Do[
 Area1 = Area1 + Re[(wData1[[iz, 2]] + wData1[[iz + 1, 2]]) *
      (wData1[[iz + 1, 1]] - wData1[[iz, 1]])];
 Area2 = Area2 + Re[(wData2[[iz, 2]] + wData2[[iz + 1, 2]]) *
      (wData2[[iz + 1, 1]] - wData2[[iz, 1]])];
 Area3 = Area3 + Re[(wData3[[iz, 2]] + wData3[[iz + 1, 2]]) *
      (wData3[[iz + 1, 1]] - wData3[[iz, 1]])];
 Area4 = Area4 + Re[(wData4[[iz, 2]] + wData4[[iz + 1, 2]]) *
      (wData4[[iz + 1, 1]] - wData4[[iz, 1]])];
, {iz, 1, Length[wData1] - 1}];

Area1 / Area4
Area2 / Area4
Area3 / Area4
Area4 / Area4
```

```
0.0689294

0.266988

0.711371

1.
```

```
Max[Re[Transpose[wData1][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
Max[Re[Transpose[wData2][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
Max[Re[Transpose[wData3][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
Max[Re[Transpose[wData4][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
```

```
0.0390355
```

```
0.165313
```

```
0.601229
```

```
1.
```

```
Max[Re[Transpose[wData4][[2]]]
```

```
0.271592
```

```
wData4[[497]]
```

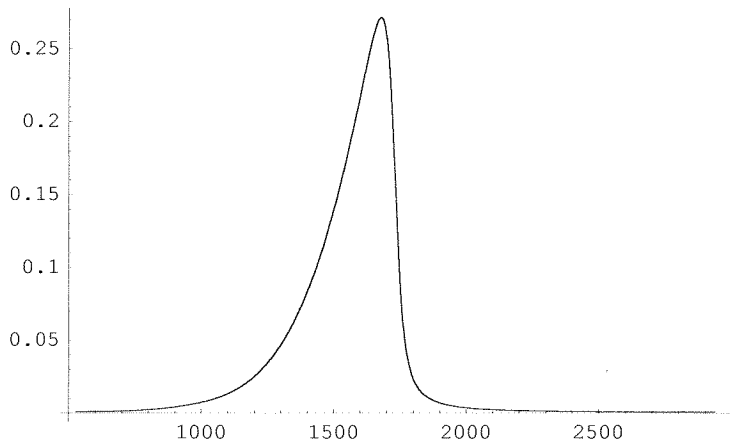
```
{55.8662, 0.271592 + 0.0845847 I}
```

```
55.8662108762756037` + 1676.43
```

```
1732.3
```

```
wwData1 = wData1;
wwData2 = wData2;
wwData3 = wData3;
wwData4 = wData4;
Do[
  wwData1[[iw, 1]] = -wwData1[[iw, 1]] + 1732.3;
  wwData2[[iw, 1]] = -wwData2[[iw, 1]] + 1732.3;
  wwData3[[iw, 1]] = -wwData3[[iw, 1]] + 1732.3;
  wwData4[[iw, 1]] = -wwData4[[iw, 1]] + 1732.3;
, {iw, 1, Length[wData1]}];
```

```
ListPlot[Re[wwData4], PlotJoined -> True,
 PlotRange -> All]
```




```
- Graphics -
```

```
0.69*0.35
```

```
0.2415
```

```
SetDirectory["C:\My Documents\Caltech\Thesis\math"];
```

```
unit =
OpenWrite["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A2
00ms30percent_tonp250ms.out"];
```

```
Do[
Write[unit,Re[wwData1[[i]]]]
,{i,1,Length[wwData1]}]
```

```
Close["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A200ms
30percent_tonp250ms.out"];
```

```
unit =
OpenWrite["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A2
00ms30percent_tonp1s.out"];
```

```
Do[
Write[unit,Re[wwData2[[i]]]]
,{i,1,Length[wwData2]}]
```

```
Close["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A200ms
30percent_tonp1s.out"];
```

```
unit =
OpenWrite["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A2
00ms30percent_tonp3s.out"];
```

```
Do[
Write[unit,Re[wwData3[[i]]]]
,{i,1,Length[wwData3]}]
```

```
Close["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A200ms
30percent_tonp3s.out"];
```

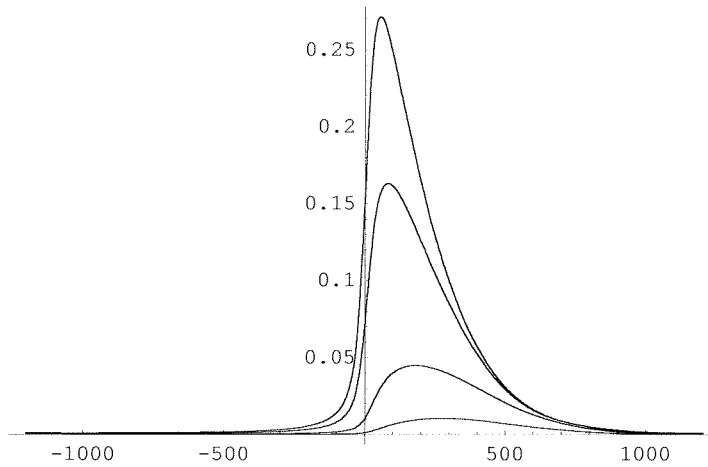
```
unit =
OpenWrite["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A2
00ms30percent_tonp5s.out"];
```

```
Do[
Write[unit,Re[wwData4[[i]]]]
,{i,1,Length[wwData4]}]

Close["KSspindiff3halfs-LineAvrgIsotrpc3_230W100A200ms
30percent_tonp5s.out"];
```

```
plot1 = ListPlot[Re[wData1], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot2 = ListPlot[Re[wData2], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot3 = ListPlot[Re[wData3], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot4 = ListPlot[Re[wData4], PlotRange -> All,
  PlotJoined -> True, DisplayFunction -> Identity];
plot11 = Plot[Log[u]^2 (1 - Exp[(-tonp1 * u^2)/T1at0])
    T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1},
  PlotPoints -> znpts, DisplayFunction -> Identity];
plot12 = Plot[Log[u]^2 (1 - Exp[(-tonp2 * u^2)/T1at0])
    T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1},
  PlotPoints -> znpts, DisplayFunction -> Identity];
plot13 = Plot[Log[u]^2 (1 - Exp[(-tonp3 * u^2)/T1at0])
    T1at0/u^2 (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1},
  PlotPoints -> znpts, DisplayFunction -> Identity];
plot14 = Plot[Log[u]^2 (1 - Exp[(-tonp4 * u^2)/T1at0]) T1at0/u^2
    (1 - Exp[(-t2 * u^2)/T1at0]), {u, 0, 1}, PlotPoints -> znpts,
  DisplayFunction -> Identity];
```

```
Show[plot1, plot2, plot3, plot4,
 DisplayFunction -> $DisplayFunction];
```




```
Area1 / Area4
Area2 / Area4
Area3 / Area4
Area4 / Area4
```

```
0.0689294

0.266988

0.711371

1.
```

```
Max[Re[Transpose[wData1][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
Max[Re[Transpose[wData2][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
Max[Re[Transpose[wData3][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
Max[Re[Transpose[wData4][[2]]] /
 Max[Re[Transpose[wData4][[2]]]
```

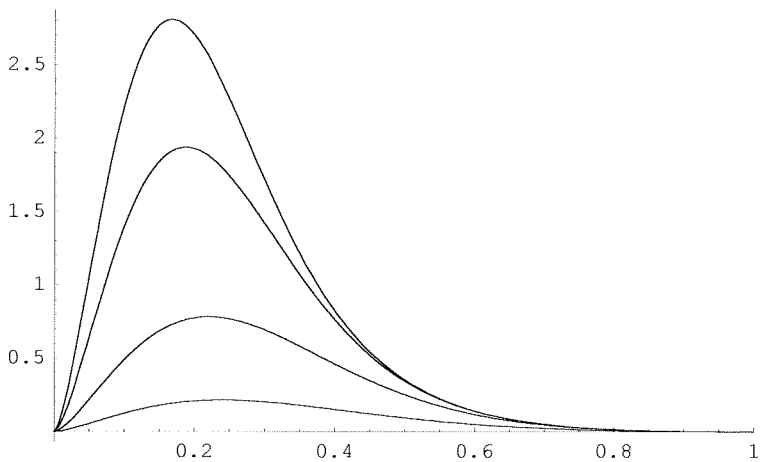
```
0.0390355
```

```
0.165313
```

```
0.601229
```

```
1.
```

```
Show[plot11, plot12, plot13,
 plot14, DisplayFunction -> $DisplayFunction,
 PlotRange -> {{0, 1}, All}]
```




```
- Graphics -
```