Entanglement and Metrology With Singlet-Triplet Qubits

Entanglement and Metrology with Singlet-Triplet Qubits

A DISSERTATION PRESENTED
BY
MICHAEL DEAN SHULMAN
TO
THE DEPARTMENT OF PHYSICS

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
PHYSICS

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS
MAY 2015

Thesis advisor: Professor Amir Yacoby Michael Dean Shulman

Entanglement and Metrology with Singlet-Triplet Qubits

Abstract

Electron spins confined in semiconductor quantum dots are emerging as a promising system to study quantum information science and to perform sensitive metrology. Their weak interaction with the environment leads to long coherence times and robust storage for quantum information, and the intrinsic tunability of semiconductors allows for controllable operations, initialization, and readout of their quantum state. These spin qubits are also promising candidates for the building block for a scalable quantum information processor due to their prospects for scalability and miniaturization. However, several obstacles limit the performance of quantum information experiments in these systems. For example, the weak coupling to the environment makes inter-qubit operations challenging, and a fluctuating nuclear magnetic field limits the performance of single-qubit operations.

The focus of this thesis will be several experiments which address some of the outstanding problems in semiconductor spin qubits, in particular, singlet- triplet (S-T_0) qubits. We use these qubits to probe both the electric field and magnetic field noise that limit the performance of these qubits. The magnetic noise bath is probed with high bandwidth and precision using novel techniques borrowed from the field of Hamiltonian learning, which are effective due to the rapid control and readout available in S-T_0 qubits. These findings allow us to effectively undo the undesired effects of the fluctuating nuclear magnetic field by tracking them in real-time, and we demonstrate a 30-fold improvement in the coherence time T_2^*.

We probe the voltage noise environment of the qubit using coherent qubit oscillations, which is partially enabled by control of the nuclear magnetic field. We find that the voltage noise bath is frequency-dependent, even at frequencies as high as 1MHz, and it shows surprising and, as of yet, unexplained temperature dependence. We leverage this knowledge of the voltage noise environment, the nuclear magnetic field control, as well as new techniques for calibrated measurement of the density matrix in a singlet- triplet qubit to entangle two adjacent single-triplet qubits. We fully characterize the generated entangled states and prove that they are, indeed, entangled.

This work opens new opportunities to use qubits as sensors for improved metrological capabilities, as well as for improved quantum information processing. The singlet-triplet qubit is unique in that it can be used to probe *two* fundamentally different noise baths, which are important for a large variety of solid state qubits. More specifically, this work establishes the singlet-triplet qubit as a viable candidate for the building block of a scalable quantum information processor.

Contents

Listing of figures

Dedicated to my parents, for their never-ending support, and for encouraging my curiousity, even if it leaves the cat simultanously dead and alive.

Acknowledgments

I AM TOLD THAT THE ACKNOWLEDGEMENTS will be the only part of this document that gets read, which is a pitty since there are far too many people to thank, and I am sure sure that I will forget people. Many people contributed to the work in this thesis, as well as work not presented, and I therefore apologize in advance for forgetting many of the people responsible for this work.

First, and most importantly, I thank my advisor, Amir Yacoby, because, as a fellow labmate said, everywhere you go from this lab will be a step down. Amir is an incredible physicist and problem solver, who created a physical and intellectual environment where it is easy to thrive. I have learned a lot from Amir, most importantly the appetite to to tackle hard and interesting problems, even if they are out of one's area of expertise.

Amir paired me with some of the most capable and brightest people I have met. Hendrik Bluhm gave me an introduction into spin qubits and taught me the importance of being able to produce[*] (even theory!). I have also really enjoyed staying in touch and collaborating with Hendrik as he establishes his laboratory in Germany. I am also very fortunate to have worked with Oliver Dial. Oliver taught me far too much to list here, including how to program, how to effectively prank lab members, how to decorate my desk, and how to strike the delicate balance between making immediate progress and doing things thoroughly. Though I was sad[†] when Oliver left the lab, I was happy to see him stay in the quantum information science community and to continue to see and interact with him. I am very thankful for the immense patience that both Oliver and Hendrik demonstrated when working with me.

Vivek Venkatachalam taught me how to make quantum dots (albeit with unconventional teaching methods) and was a great source of advice for me when I was a new graduate student. Vivek is also a fantastic running partner, is always up for a workout, and therefore made graduate school much more pleasant. I was obviously thrilled to see him continue in a lab across the street upon graduating. Sandra Foletti and Gilad Barak were great influences on me as senior graduate students when I joined that lab. Sandra taught me how to hold a transfer tube (and that it is important to be taught how to transfer helium), and Gilad, the happiest physicist ever, left me some of the most useful advice of graduate school: study for quals more than me (Gilad) and less than Sandra.

[*] also called getting sh*t done

[†] and incredibly nervous

For the last two thirds of graduate school I worked with Shannon Harvey, who is a great physicist and lab member. Working with a teammate who cares not only about physics and progress but also about the state of the lab, people, and equipment made the experience much more enjoyable. Shannon and I were joined by Dr. John Nichol PhD, who brought a Scandinavian disposition, FPGA programming skills, and aversion to a harsh office environment to team qubit. It was a true pleasure to work with Shannon and John, and I will consider myself lucky to find myself in a team like this in the future. I am honored to be co-authors and friends with Shannon and John, and I know that team spewbit, with the addition of Lucas Orona, is in good hands.

Patrick Maletinsky taught be how beautiful Switzerland is and how to treat a fume hood with respect[‡]. Sungkun Hong taught me how to approach nanofabrication with a similar cavalier attitude, and Mike Grinolds taught me a healthy skeptical attitude, as well as how to "settle down" when things get out of hand. And hey! It's Ben Feldman, with whom I was lucky to briefly work on the new Frosatti refrigerator.

The yacobyserver is in the extremely competent hands of Yuliya Dovzhenko, who, more than anyone I have worked with, is an expert learner by doing. My path happily crossed that of Marc Warner, who is not only an excellent table tennis partner, but also a great friend. Happily, the NV experiments are very difficult, and require a lot of smart, capable people with whom I was fortunate to interact, including Toeno van der Sar, Francesco Casola, Lan Luan, and Pat Gumann.

I was fortunate to be exposed to team graphene, a group of talented researchers including Thomas Weitz[§], Jens Martin, who is an experimental physics guru and legend, and Monica Allen. Team graphene will hold a special place in my heart after sharing an apartment with Andrei Levin and sharing a desk with Didi Wei. I really enjoyed working on the Frosatti dilution refrigerators with Didi, and should probably take the opportunity to apologize for the hostile office environment that team qubit (mostly me) created.

Graduate school would have been impossible without help from people outside the lab, many of whom were in the Marcus Lab. Charlie was a great source of advice, I was unhappy to see him leave Harvard. Jim Medford was always happy to talk qubits (or life), and Hugh Churchill was a great source of physics, fabrication, and cryogenic knowledge (and an incredibly nice guy). Andrew Higginbotham is a great friend and scientist, and I am fortunate to have had the opportunity to frequently invade his office. I found my true snack twin in Angela Kou[¶], who punctuated graduate school with delicious snack adventures, impatience for stupidity, and a sharp eye for interesting physics. Lastly, but certainly not least, I must acknowledge Willy Chang for dealing with me for eight years- enough said.

The Columbia University north campus continued to grow with Hechen Ren, who was an excellent addition to the lab, and eventually with Philip Kim, who is a real Mensch, and who was a great source of advice for me. I am sad to finish so soon after Philip arrived.

[‡]The reader is directed the the lab website for photo proof

[§]inventor of the eponymous Thomas method of beer drinking

[¶]who fortunately joined the Yacoby lab for a year

Sean Hart likely delayed graduation by a year[‖], but I must forgive him since he is an adept Mario Kart player, as well as the only other graduate of the Vivek school of nanofabrication.

I owe many thanks to Jim MacArthur for dealing with my incessant nagging and unreasonable needs, and to Giorgio Frosatti for graciously handling numerous phone calls as well as teaching me the care-free attitude that all should have when dealing with cryogenics. I must also acknowledge Stephen Bartlett and Andrew Doherty for a fruitful collaboration. I am told that such a pleasant collaborative experience is rare and should not be taken for granted.

A very special thank you to Carolyn Moore who is the backbone of the group and who graciously took care of so many things over the past years.

Lastly, none of the work presented here would be possible without my family, particularly my parents Vivian and Yale, who offer endless encouragement and support.

[‖]joke

If the Boeing 747 obeyed Moore's law, it would travel a million miles an hour...and a trip to New York would cost five dollars.

Nathan Myhrvold

0 Introduction

0.1 History and Motivation

Quantum computing is no longer an obscure concept known only to a few physicists, computer scientists, and several people at the NSA. The New York Times runs an article or blog post about quantum computing almost every month. But what is this mysterious machine called a quantum

computer? It is a computer whose building blocks are governed by and take advantage of the strange laws of quantum mechanics. Thankfully, it is not difficult to prove that computers made of quantum systems can perform computation at least as fast as those made of classical parts. That, however, is not reason enough to justify the immense effort to build a quantum computer. Rather, we pursue this undertaking because there are certain problems where quantum mechanical algorithms are exponentially faster than the most efficient known classical ones. Some particularly well known examples include Shor's algorithm for prime factorization of numbers[57] and Grover's algorithm for searching a database[26].

It is difficult to accurately describe *why* quantum computers are faster than classical ones. As a heuristic, we can consider two special properties about quantum states that do not have classical counterparts: quantum superposition and entanglement. Quantum mechanics allows for a state, $|\Psi\rangle$, not only to be in the state $|0\rangle$ or $|1\rangle$, but also in a superposition of $|0\rangle$ and $|1\rangle$. In other words, we can have valid quantum states that are written as $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$ for two complex numbers α and β, whereas a classical system would be constrained to take one of two values, either $|0\rangle$ or $|1\rangle$. There is, in essence, quantum parallelism[45] which is exploited in quantum computation and is derived from the availability of superpositions. The second property of quantum systems for which there is no classical analog is the availability of entangled states. These states exhibit classically forbidden correlations between the particles, and can therefore not be completely described by specifying only the state of *each* particle. For example, we can consider $|\Psi\rangle = \alpha|01\rangle + \beta|10\rangle$, or generally we can have $|\Psi\rangle \neq \prod_i |\psi_i\rangle$. The existence of non- classical correlations mean that operations on one part of a quantum system can have instantaneous, non-local consequences on another part, and also gives quantum computation its power. These strange deviations from classical physics (and most of observable reality) make quantum mechanics difficult to comprehend and left great physicists such as Einstein uneasy about their "spooky" consequences. They are, nevertheless, the laws that govern reality, and we use them to build new, useful technologies.

It is far easier to describe what is difficult about quantum computing than why it is so powerful. Quantum systems are disrupted by measurement, i.e. measuring a quantum system collapses or changes the state*, effectively making the system classical. Moreover, this collapse of a quantum system occurs not only from an intentional measurement (for example, by a human), but also from an unintentional measurement from the environment. As such, isolating the computer from its environment is paramount to building a quantum computer. Therefore, the requirements of a quantum computer are contradictory- on the one hand we must isolate a quantum system completely so that its environment does not destroy its delicate quantum nature, but on the other hand we must also be able to interact with it to perform operations.

One of the first people to think about quantum computation was the American physicist, Richard Feynmann. In his famous lecture at MIT, "Simulating physics with computers"[21], Feynmann wondered whether quantum mechanical systems could be efficiently simulated with classical computer, since the dimensionality of the problem (the number of quantum states available) scales exponentially with the number of particles in the system. Instead, Feynmann posited that quantum systems would need to be simulated on a computer that behaves quantum mechanically. In fact, this exponential scaling with the number of quantum states available also sheds light on the origin of the power of quantum computation.

0.2 Physical Implementation

Since Feynmann, countless others have contributed to the field, and there are now several known uses for a quantum computer, as well as several viable candidates for a feasible architecture. Some particularly promising examples are superconducting elements (Josephson junctions), nuclear spins, electron spin, trapped atoms, and trapped ions, each with its own advantages and disadvantages.

*this may not be the most accurate description of reality, but it will suffice

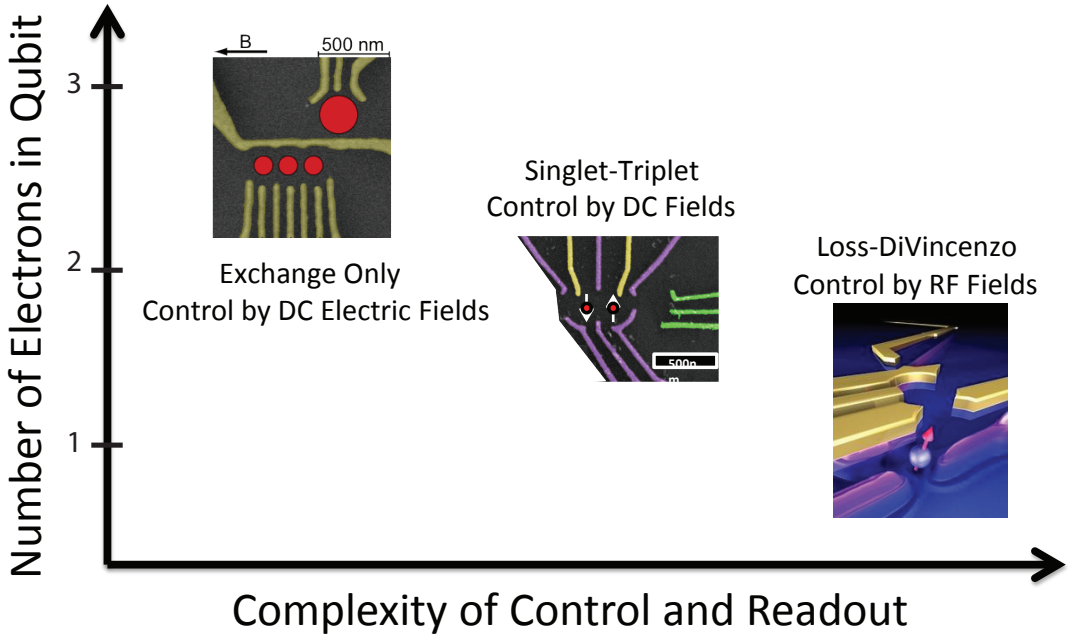

Figure 1: There are many kinds of gate-defined semiconductor spin qubits. In general, as one adds more electrons to define the qubit, qubit readout and control become easier.

As such, the state of quantum computing is similar to the state of classical computing in the 1930s- there are several promising candidates for the building block of the computer, and a clear favorite has yet to emerge.

Our group has historically been interested in qubits made of electron spins in gate-defined quantum dots (QDs). First proposed by Loss and DiVincenzo in 1998[38], these qubits have great potential for scalability and miniaturization and have spawned several "cousin" qubits (Figure 1) which are also promising. One such variant, and the focus of this work, is the singlet-triplet qubit [37, 49], which is made of two electron spins in a gate-defined double quantum dot (DQD).

0.3 Organization of this Document

All of the work presented here will deal with singlet-triplet (S-T0) qubits in gate-defined Gallium Arsenide DQDs. We are partial to these qubits for several reasons in addition to those stated above. First, they are easily manipulated with DC electric fields, which simplifies qubit control. Second, fabrication is straightforward and makes use of standard semiconductor techniques. Third, owing to the fact that the singlet-triplet is similar to many other spin qubits (Figure 1) and can be made sensitive to both electric and magnetic fields, it is a good testbed for studying the environments of the other spin qubit cousins. This work builds on the contribution of much previous work in S-T0 qubits, and hopefully demonstrates the viability of the S-T0 qubit as a platform for studying quantum information and condensed matter physics.

Chapter 1 will provide background on qubits, such as the basics of two-level quantum systems, operation, measurement, etc.

We provide background information on singlet-triplet qubits in chapter 2. This chapter will motivate the S-T_0 qubit from an easily-understood "cartoon" approach, instead of a historical and rigorous theoretical one that can be found elsewhere.

Chapter 3 will describe an experiment to entangle two adjacent singlet-triplet qubits. This work uses the difference in charge configuration between the qubit states to generate coherent coupling between qubits. It uses a dynamically decoupled two-qubit gate in order to isolate each qubit from its fluctuating voltage environment, and, using two-qubit state tomography, verifies that an entangled state is created.

In chapter 4 we describe the procedure for calibrating state tomography, which is necessary to accurately measure the density matrix of a quantum state. This procedure is an integral part of entanglement verification in chapter 3 and provides a useful tool for many future quantum information experiments. This tomography calibration is different from existing calibration sequences in that it determines the axes on to which a qubit is projected, rather than tuning these axes to be an orthonormal set.

Chapter 5 describes an experiment in which coherent qubit oscillations are used to probe the voltage noise bath to which the singlet- triplet qubit couples. The spectrum is probed from near-DC up to 1MHz, and we find that the voltage noise spectrum is frequency dependent even at these high frequencies. Though we are unable to definitively identify the source of this noise, the surprising temperature dependence of the spectrum provides a starting point to begin looking for it.

We shift gears from voltage noise to nuclear magnetic field noise in chapter 6. We develop techniques to accurately and rapidly measure the nuclear magnetic field, which draw on the rich fields of Hamiltonian learning and Bayesian inference. We then leverage these tools in real-time to improve the coherence of the qubit by a factor of 30.

Additional useful (but peripheral) information can be found in the appendices.

Computers are useless. They can only give you answers.

Pablo Picasso

1 Qubits

This work is far from the first thesis ever written on semiconductor spin qubits. There have been many well-written works detailing the background of semiconductor spin qubits[22], most of which begin with two dimensional electron gases (2DEGs), quantum point contacts (QPCs), quantum dots (QDs) and many other great strides in mesoscopic semiconductor physics that have enabled the subsequent scientists to realize spin qubits. These works have had monumental effects

on both quantum information and fundamental condensed matter physics. However, here I will present a different background to singlet-triplet qubits for three reasons: first, as the field progresses, the participants (grad students) become more removed from these works and therefore understand them with less depth. As a result, I am certain that I cannot describe these topics better than they are described, for example, in Sandra Foletti's thesis[22], and would rather direct the reader to its introduction. Second, the historical progression of events is not necessarily the best order in which to learn and understand a field*. Lastly, in many respects, learning to think about singlet- triplet qubits as qubits, instead of as mesoscopic QDs, was crucial to much of the work presented here and to furthering our understanding of the field, and a quantum information- focused approach is therefore more relevant.

1.1 Qubit Basics

The basic building block of a modern computer is the transistor, which can be either on (1) or off (0) and therefore stores one bit of information. The quantum analog is a two-state quantum system or quantum bit, which we call qubit for short. A general quantum state can therefore be written as

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

where α and β are complex numbers that obey $|\alpha|^2 + |\beta|^2 = 1$. This condition allows us to rewrite the qubit state as

$$|\Psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle \tag{1.1}$$

where θ and ϕ are real numbers. This representation of a qubit resembles the coordinates describ-

*for example, first courses in quantum mechanics which begin with the ultraviolet catastrophe

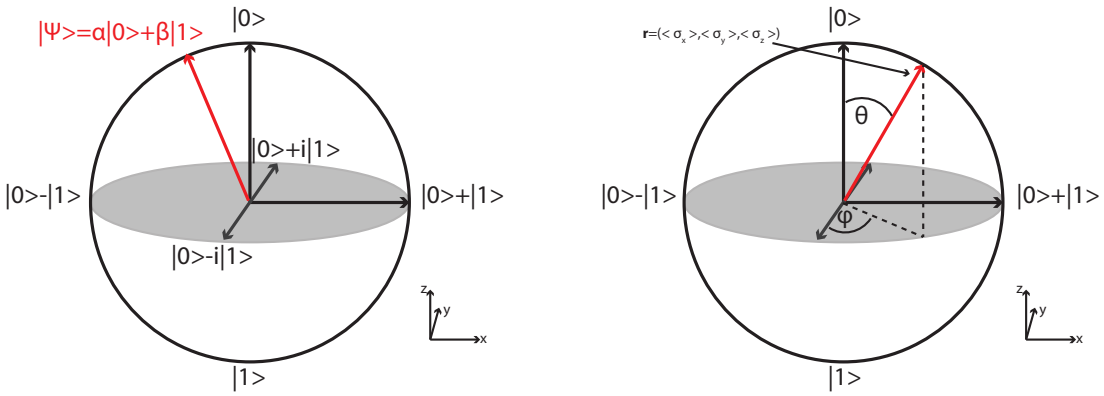

Figure 1.1: Left: The Bloch sphere representation of a pure two-level quantum system, with one qubit state at the north pole and one qubit state at the south pole. Right: The Bloch sphere representation of a qubit is analogous to the surface of a sphere.

ing the surface of a sphere, where θ and ϕ represent latitude and longitude, respectively. We call this representation the Bloch sphere (Figure 1.1 left), and it is powerful because it allows scientists studying disparate qubits to have a common language for describing their work. Additionally, as we will see, the Bloch sphere gives a natural representation for qubit evolution and mixed states.

We will want to control the qubit, so the next ingredient we need is to describe qubit evolution. If we restrict ourselves only to pure quantum states then qubit evolutions are unitary operations on a quantum state. Given the Hamiltonian $\mathcal{H}(t)$ we can solve the Schrodinger equation for the time evolution of the state $|\Psi\rangle$:

$$i\hbar\frac{d}{dt}|\Psi\rangle = \mathcal{H}|\Psi\rangle$$

which will uniquely describe the system given an initial condition $|\Psi(t=0)\rangle$. As a concrete example, we consider the quantum state $|\Psi(t=0)\rangle = |0\rangle$ and imagine it interacts with a Hamiltonian

$$\mathcal{H} = \frac{\Delta}{2}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \frac{\Delta}{2}\sigma_x \tag{1.2}$$

We can solve the Schrodiger equation to find

$$|\Psi(t)\rangle = \cos(\Delta t)\,|0\rangle + \sin(\Delta t)\,|1\rangle\,.$$

If we plot the trajectory on the Bloch sphere, we see that the solution begins at the north pole and rotates around the x-axis. Perhaps unsurprisingly, this relation between σ_x and rotations around the x-axis hold for the y and z Pauli matrices as well. Moreover, simple calculations show that the coordinates of the quantum state on the Bloch sphere (often called the Bloch vector), (x, y, z) are simply given by the expectation values $(\langle\sigma_x\rangle, \langle\sigma_y\rangle, \langle\sigma_z\rangle)$ (Figure 1.1 right). So now we can represent pure quantum states on the surface of the Bloch sphere and can represent evolution of states by rotations around the sphere. More precisely, the axis of rotation around the Bloch sphere is defined by the eigenstates of the Hamiltonian. If, for example, the eigenstates are $|0\rangle$ and $|1\rangle$, then we will have rotations around the axis that connects those two states (in this case the z-axis).

1.2 Mixed States

It may appear that we have all of the machinery we need for describing two- level quantum systems: we can describe any pure state and any evolution between pure states with a simple depiction on the Bloch sphere. Sadly, the world is imperfect, and we need more machinery to deal with non- pure (mixed quantum states). It should be noted, however, that need to describe mixed states in modern quantum information experiments is, in some sense, encouraging; the field has progressed to the point where losses and interactions with a bath (the environment) are interesting and relevant. Happily, the Bloch sphere can easily be extended to describe mixed states.

First we need to understand what a mixed state is. To understand this strange beast, we compare two scenarios:

1. $\frac{1}{\sqrt{2}}\left[|0\rangle + |1\rangle\right]$

2. a state with 50% probability $|0\rangle$ and 50% probability $|1\rangle$

The first, is a state that is *truly* in both $|0\rangle$ and $|1\rangle$ at the same time, whereas the second is either in $|0\rangle$ or in $|1\rangle$, but *never* in both. To illustrate the difference between these, imagine rotating these states by $\frac{\pi}{2}$ around the y-axis of the Bloch sphere. For the first item in the list we would obtain the simple pure state $|1\rangle$, but for the second (it require a few lines of algebra) the state would remain unchanged! It is in this way that a superposition state and mixed state differ.

One effective way to describe a mixed state is to keep track of all of the comprising pure states and their relative probabilities. We do this with the density matrix:

$$\rho = \sum_i p_i \left|\psi_i\right\rangle \left\langle\psi_i\right| \tag{1.3}$$

where p_i is the probability that we find the system in the state $|\psi_i\rangle$. For example, for the two systems mentioned above, we can write the density matricies in the $|0\rangle$,$|1\rangle$ basis as:

$$\frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \text{ and } \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{1.4}$$

The density matrix (sometimes called the density operator) has many useful properties. Some handy examples include

1. $\mathrm{Tr}(\rho) = 1$
2. $\rho^2 = \rho$ for pure states
3. $\langle O\rangle = \mathrm{Tr}(O\rho)$.

The last property is particularly useful, since the x, y, and z coordinates of a quantum state on the Bloch sphere are related to the expectation values of the Pauli matrices. We can therefore represent a

mixed state on the Bloch sphere with the Bloch vector ($\vec{r}$)

$$\vec{r} = (x, y, z) = (\langle\sigma_x\rangle, \langle\sigma_y\rangle, \langle\sigma_z\rangle). \tag{1.5}$$

As we will see shortly, pure states are points on the surface of the sphere, while mixed states are points in the interior of the sphere, and all fully mixed states located at the origin.

Both the density matrix and the Bloch sphere representations have distinct advantages. For example, it is easy to see how the density matrix evolves under unitary operations, while the Bloch vector is only three real numbers (compared to four complex numbers constrained by symmetry for the density matrix). We have already seen how to obtain the Bloch vector from ρ (equation 1.5), and a little math gives the (tremendously useful) reverse formula:

$$\rho = \frac{1}{2}\left(\mathbb{I} + \vec{r}\cdot\sigma\right) \tag{1.6}$$

where σ is shorthand for $(\sigma_x, \sigma_y, \sigma_z)$.

1.3 Interaction with the Environment and Decoherence

So how does a quantum state become mixed? Were we simply too lazy to keep track of its state? Purity is generally lost when a quantum system interacts with its environment. Recall, that when a quantum system is measured it is irreversibly collapsed or projected into an eigenstate. This applies not only for intentional measurements, but also for unintentional measurements. As a heuristic, we can therefore think of interactions between a quantum system and the environment as the environment accidentally measuring the qubit. Since we do not have precise knowledge or control of the environment, we do not know whether this measurement has occurred, so it causes us to lose information about the quantum state. This process is generally called decoherence.

Put more precisely, we can group the processes by which a quantum state becomes mixed into three categories:

1. Energy relaxation: if the qubit exchanges energy with the environment, it will alter the quantum state. There will often be a characteristic time (denoted T_1) that describes this interaction.

2. Loss of phase coherence: The phase of a quantum state can also be altered by interaction with the environment. An example of such a "phase flip" would be if a qubit switched from $|0\rangle + |1\rangle$ to $|0\rangle - |1\rangle$. This, too, often has a characteristic timescale, usually denoted T_2.

3. Inhomogeneous broadening: In practical situations, experimentalists will usually deal with ensembles of qubits. For example, in liquid NMR we might perform experiments on many nominally identical molecules in order to increase signal to noise, and in semiconductor spin qubits we will often repeat experiments many times to obtain statistics. Even if each *individual* qubit preserves its phase coherence, due to slight inhomogeneities between qubits, ensemble averages will appear to lose phase coherence. This "inhomogeneous broadening" (a term borrowed from the NMR community[†]) is described by a characteristic time T_2^*.

As we can see, there are numerous ways in which a qubit interacts with the environment and its state becomes mixed. However, as we will see, these three types of interaction are fundamentally different from one another. Additionally, these characteristic times, T_1, T_2 and T_2^*, are a measure of how well isolated a qubit is from its environment, and for better or for worse, since decoherence is arguably the biggest obstacle towards quantum information processing, these are the metrics by which desperate qubits are compared.

[†]Borrowing terms from the NMR community will be an unfortunate recurring theme, for which the author claims no responsibility

1.4 Measuring a Qubit

Once we know how to describe and control a qubit, we also need to know how to measure it. Quantum measurement is a rich field of study, with far too much information to convey here[71]. The salient point for this work is that measuring a qubit will irreversibly alter its state. For all the work presented here, measurement will extract the projection of the qubit onto an axis of the Bloch sphere (usually z-axis) and we will measure either $|0\rangle$ or $|1\rangle$, regardless of the original qubit state.

This peculiar property of quantum systems leads to an obvious question: how do we extract the full state of a qubit (i.e. ρ or $\vec{r}$) if we can only perform one measurement before destroying the qubit state? The process of extracting a full qubit state, called state tomography, is an active area of theoretical and experimental research, and will be important for understanding chapters 3 and 4. State tomography relies on repeated measurements, projecting identical copies of a qubit onto the three Cartiesian axes. As we have seen, once we know $(\langle\sigma_x\rangle, \langle\sigma_y\rangle, \langle\sigma_z\rangle)$ we can reconstruct the entire density matrix using equation 1.6.

1.5 Viable Qubits

Before we begin describing singlet-triplet qubits (after all, that *is* the focus of this work), we pause to consider what is required to build a functioning quantum computer. The necessary criteria were originally laid out by David DiVincenzo[18], which we summarize here.

A scalable physical system with well characterized qubits

> i.e. we need a system in which it is easy to build many qubits, and we need to understand each qubit precisely in order to control the system.

The ability to initialize the state of the qubit to a simple fiducial state

We must begin a computation in a known, well-defined state. Additionally, all quantum error correction schemes rely on having a continuous supply of qubits in known states.

A universal set of quantum gates

We need to access all possible quantum states in order to perform effective computation. A particularly feasible way to realize this requirement is two axes of control on the Bloch sphere of *each* individual qubit, as well as two-qubit operations (for example CNOT) between pairs of qubits. Interestingly, for N qubits, one can show that given sufficiently complicated operations, only two unitary operations are needed for universal control of the system.

A qubit specific measurement capability

We cannot extract answers from our computation unless we can accurately measure at least some of the qubits.

Decoherence time, much longer than the gate operation time

We cannot hope to perform quantum computation if the computation time is longer than the lifetime of the quantum information.

The final criterion is likely the hardest one to fulfill and also arguably the biggest problem standing in the way of quantum computation. As we increasingly isolate qubits from their environments in order to prolong coherence, they become more challenging to control, and operation times often suffer. We will see this issue in chapter 5 when considering the effects of charge noise on singlet-triplet qubits.

What's a cubit?.

Conversation between Noah and God

2

The Singlet-Triplet Qubit

A SINGLE SPIN is a natural choice for a qubit. After all, the implicit analogy of all qubits (and the Bloch sphere) is to a spin. Practically speaking, spins are attractive candidates for quantum information experiments because they interact weakly with the environment (and usually everything else) and therefore have long coherence times. In their seminal paper, Loss and DiVincenzo[38] proposed

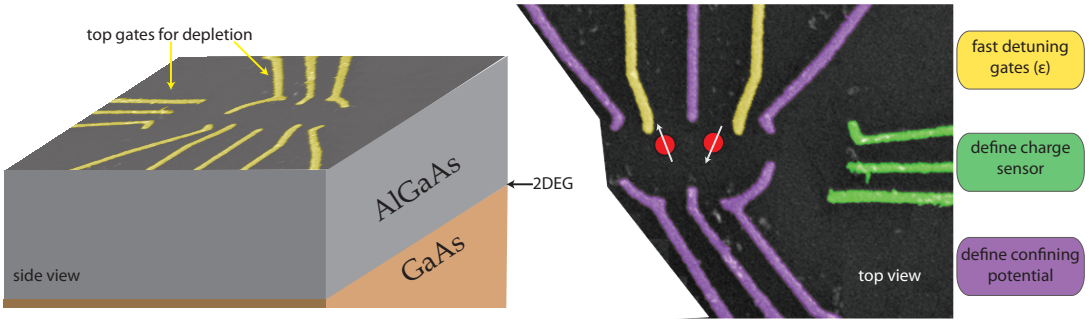

Figure 2.1: Left: a side view of a GaAs-AlGaAs heterostructure with a two dimensional electron gas (2DEG) at the interface between GaAs and AlGaAs, and the surface gates used to deplete the 2DEG. Right: A top view of a device with the various top gates used to define, control, and read the charge state of the quantum dots.

using single spins in semiconductor quantum dots as the building blocks for a quantum computer[*]. Though much great work has been done on Loss-Divincenzo (LD) qubits[38], due to some practical limitations, we choose to build our qubits from *two* electron spins. The singlet-triplet qubit is attractive because it is easy to initialize and measure, is imuned to certain fluctuations in the magnetic field, and can be controlled with DC electric fields. Here, we will explain the basics, energetics, and operations of the singlet-triplet qubit[†]. Importantly, we hope that this treatment will be accessible and intuitive to a wide readership. A more rigorous derivation of the Hamiltonian can be found in references [22, 62].

2.1 Two-Particle Physics

Condensed matter physics and technology have triumphed in two opposing ways; our control and understanding of many-body physics allow us to understand systems such as complex oxides and high-temperature superconductors, while our ability to make systems pristine allows us to isolate and control simple one- and two-particle systems. The latter are particularly exciting since they are among the only exactly solvable problems in quantum mechanics.

[*] see figure 1 of the paper ([38]) for one of my favorite figures in all of the literature
[†] Thanks to Oliver Dial for suggesting this as a relatively straightforward way to motivate this material

The singlet-triplet qubit is made of a two-electron state in a double quantum dot, like the one shown in Figure 2.1 left). These quantum dots are made in a GaAs-AlGaAs heterostructure, and a good description of the system can be found in ref [22]. Simply put, this system is two pockets of electrons whose charge occupancy can be tuned with DC voltages on the top gates (Figure 2.1 right). If we have a charge occupancy of n_l electrons in the left dot and n_r electrons in the right dot, we denote the charge state of the system as (n_l, n_r). We restrict ourselves to two- electron charge states, specifically to the states (0,2) and (1,1).

We begin by constructing a charge qubit, and we consider the lowest energy levels. We control the system with a uniform electric field that can push one electron between the two dots and thus energetically favor either (0,2) or (1,1). Throughout this work, we will refer to this parameter as ϵ, or detuning, and it is the difference in energy between (0,2) and (1,1). The Hamiltonian for such a system is simply $\mathcal{H} = \frac{\epsilon}{2}\sigma_z$, and the spectrum of the lowest two configurations of the (0,2)-(1,1) system as a function of ϵ is shown in Figure 2.2a. The system becomes far more interesting if there is a tunnel barrier between the dots, so an electron can coherently tunnel between them at an energy cost of T_c. This addition allows for admixtures of (0,2) and (1,1), and changes the spectrum to one of a simple avoided crossing (Figure 2.2b).

Charge qubits are interesting, but we prefer to study spin qubits. There are four two-electron spin states, which, when we write them as eigenstates of the exchange operator are

1. $|S\rangle = \frac{1}{\sqrt{2}}\left(|\uparrow\ \downarrow\rangle - |\downarrow\ \uparrow\rangle\right)$
2. $|T_0\rangle = \frac{1}{\sqrt{2}}\left(|\uparrow\ \downarrow\rangle + |\downarrow\ \uparrow\rangle\right)$
3. $|T_+\rangle = |\uparrow\ \uparrow\rangle$
4. $|T_-\rangle = |\downarrow\ \downarrow\rangle$

We can now identify the the spin states on the energy diagram. Since electrons are Fermions, we

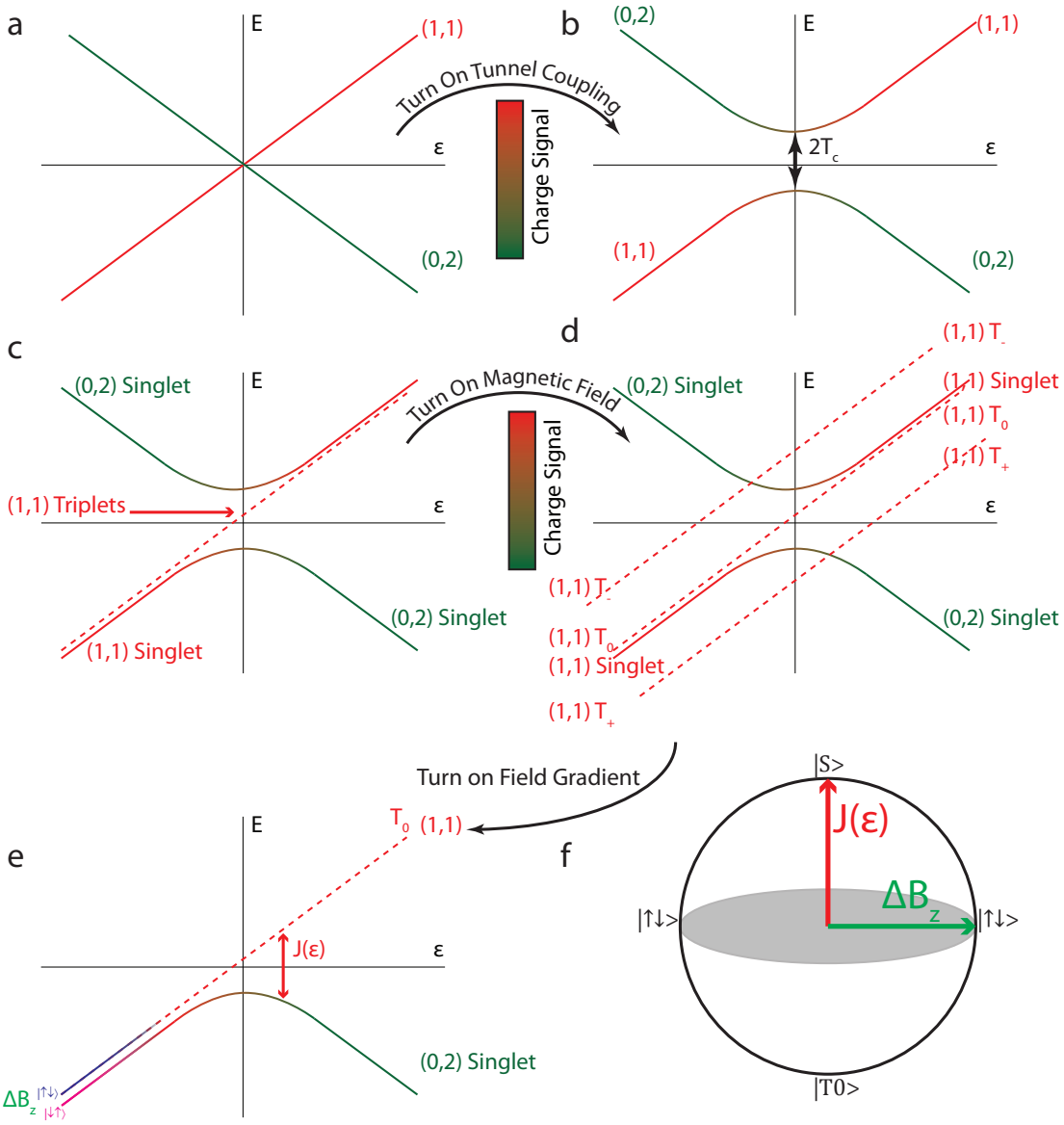

Figure 2.2: **a**. The energy diagram for a single charge qubit with two levels, (0,2) and (1,1) that do not mix. **b**. Adding a tunnel coupling T_c between the two quantum dots opens an avoided crossing between the qubit state. **c**. The spin states can be identified on the charge diagram. The singlet state can occupy any charge configuration, but the triplet states are blockaded from the lowest energy configuration of (0,2) since triplets are symmetric under exchange. **d**. Turning on a static, uniform magnetic field splits the $|T_+\rangle$ and $|T_-\rangle$ by the Zeeman energy. **e**. After turning on a magnetic field gradient, we obtain the final energy diagram for the two energy levels S-T_0 qubit. **f**. The Bloch sphere representation of the S-T_o qubit with the two axes of control, J and ΔB_z.

require that the total wavefunction (spin and orbital) be antisymmetric upon exchanging the two particles. The $|S\rangle$ therefore, can occupy any charge state, since we can construct ground state orbital wavefunctions that are symmetric for both (0,2) and (1,1). However, the spin- symmetric triplet states ($|T_{0,\pm}\rangle$) cannot occupy the (0,2) charge configuration, since the orbital ground state of (0,2) is symmetric (both electrons in the orbital ground state). Put simply, since the state $(0, 2)$ $|\text{triplet}\rangle$ is symmetric under exchange of the two particles, and is therefore not a valid quantum state, the triplet is "blocked" from the (0,2) charge configuration. In the literature, this effect is commonly called Pauli blockade or spin blockade. If we now ignore the excited branches in the spectrum (they are inaccessible at dilution refrigerator temperatures) we have a nice two level system whose splitting is tunable with a DC voltage (ϵ) from very small (<1MHz) to very large (>20GHz)[‡]. We call this splitting the exchange splitting (J) because it arises due to the exchange energy difference between the symmetric and antisymmetric states (Figure 2.2c). We can describe the resulting Hamiltonian as

$$\mathcal{H}_J = \frac{J(\epsilon)}{2}\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = \frac{J(\epsilon)}{2}\sigma_z \tag{2.1}$$

We do not yet have a qubit because there are still four spin states available. We eliminate two using a uniform magnetic field, B, which splits the $|T_\pm\rangle$ away from $|T_0\rangle$ by an amount $g^*\mu_B B$, where μ_B is the Bohr magneton, and $g^* \approx -0.4$ is the effective gyromagnetic ratio in GaAs (Figure 2.2d).

The last ingredient we need for the S-T_0 qubit is universal quantum control, since the exchange interaction J can only drive rotations around one axis (z). Some careful thought reveals that a magnetic field *gradient*, ΔB_z, will break the degeneracy between $|\uparrow\ \downarrow\rangle$ and $|\downarrow\ \uparrow\rangle$ and therefore drive rotations around the x-axis of the Bloch sphere between $|S\rangle$ and $|T_0\rangle$. The Hamiltonian for ΔB_z

[‡]We will attempt, whenever possible to measure energies in frequency. Though it may see odd and theorist-esque, this convention is extremely useful since a qubit, when under the influence of a certain energy splitting, will precess at this frequency

can therefore be written as

$$\mathcal{H}_{\Delta B_z} = \frac{\Delta B_z}{2} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \frac{\Delta B_z}{2} \sigma_x \tag{2.2}$$

which allows us to write the complete Hamiltonian of the system, $\mathcal{H} = \mathcal{H}_J + \mathcal{H}_{\Delta B_z}$ as

$$\mathcal{H} = \frac{1}{2} \begin{pmatrix} J(\epsilon) & \Delta B_z \\ \Delta B_z & J(\epsilon) \end{pmatrix} = \frac{J(\epsilon)}{2} \sigma_z + \frac{\Delta B_z}{2} \sigma_x \tag{2.3}$$

The final qubit spectrum is depicted in Figure 2.2e, with the Bloch sphere depicted in Figure 2.2f.

In practice, though we do not understand why, one the (1,1) side of the charge transition, we find that J depends on ϵ in an exponential manner, namely $J = J_0 e^{\epsilon/\epsilon_0}$. This peculiar behavior, which is studied in chapter 5, leads to convenient operation: To drive x-rotations we tune $\epsilon \ll 0$ to have $\Delta B_z \gg J$, and to drive z-rotations we tune ϵ such that $J \gg \Delta B_z$. This effectively gives two orthogonal axes of control.

2.2 Basic Operation of S-T_0 Qubits

Now that we have described the energy spectrum of the S-T_0 qubit, we can describe typical operation. We do so, framing the S-T_0 qubit in terms of the DiVincenzo criteria (1.5).

2.2.1 Initialization

We initialize the S-T_0 qubit in the region of the energy diagram where the J is much larger than temperature, deep in (0,2). By bringing the chemical potential of the quantum dot close to the Fermi energy of the lead to the dot, an electron is exchanged with the leads and the resulting state will be a (0,2)$|S\rangle$. This operation can be completed quickly ($\approx$1ns) by making the tunnel barriers

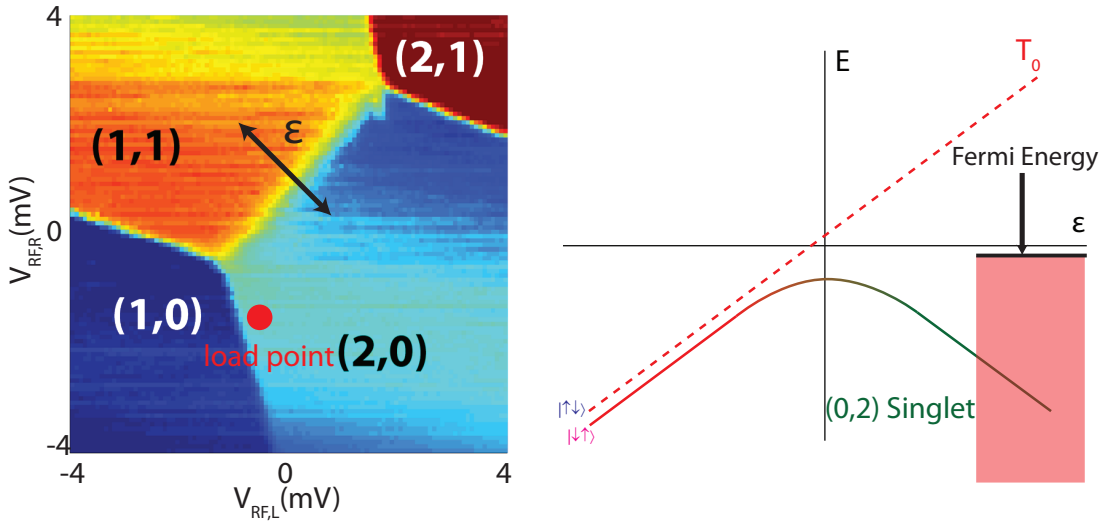

Figure 2.3: Left: a typical honeycomb charging diagram with the location of singlet-load. Right: a depiction of the energy levels with the position of the Fermi level in the leads of the quantum dot. By placing the Fermi level in between singlet and triplet, a singlet can be rapidly and deterministically loaded with high fidelity.

to the leads sufficiently transparent, and if so, qubit initialization will be limited by thermal occupation of the T_0 state, which will scale as $e^{\frac{-J_{load}}{k_B T}}$, where k is Boltzmann's constant and J_{load} is the singlet-triplet splitting at the load point, which is typically $\approx$ 1 Kelvin. (Figure 2.3).

2.2.2 Characterization

We have demonstrated, in theory, that the S-T_0 qubit is well-described by a simple model above. In practice, the model will be different from qubit to qubit due to different tunings and disorder. However, in practice, these qubits are quite stable, allowing them to be characterized with great precision. Standard methods can be employed to characterize S-T_0 qubits, but in chapter 6 we will draw on sophisticated methods in Hamiltonian learning in order to greatly improve qubit characterization.

2.2.3 Control

The qubit control is done in the region of the energy diagram where the splittings are easily measured and are within the bandwidth of our waveform generators (1GHz)[§]. To drive rotations around the z (singlet-triplet) axis of the Bloch sphere, ϵ is tuned to a region where $J \gg \Delta B_z$, and to drive rotations around the x-axis of the Bloch sphere, J is switched off so the qubit precesses around ΔB_z. In chapter 6 we will introduce an AC method of control of the S-T_0 qubit, but for the rest of the work we will use this DC operation.

One disadvantage of spin-based quantum information processing is that it is difficult to entangle two spins, since they interact only very weakly with one another. Chapter 3 describes an experiment that generates a provably entangled state, the first time with S-T_0 qubits.

2.2.4 Measurement

Spin blockade is the basis for many spin-qubit measurement schemes, and is most straightforward for S-T_0 qubits. To measure the qubit projection on the singlet-triplet axes, ϵ is switched to the region of the energy diagram where $|S\rangle$ occupies the (0,2) charge configuration and $|T_0\rangle$ is blockaded in (1,1). This "spin to charge" conversion enables the qubit spin state to be inferred from its charge state, which is determined by the conductance of a proximal charge sensor. For this work we use another QD tuned to the shoulder of a Coulomb blockade peak as the charge sensor[2].

To read the state of the qubit in a single shot with good fidelity, it is necessary to rapidly determine the conductance of the charge sensor before the qubit charge state can relax. Thankfully, this has become commonplace using standard RF-reflectomtry techniques [53, 4], and a good treatment of S-T_0 qubit readout can be found in Christian Barthel's thesis[2].

[§]we will relax this assumption in chapter 3 and measure fast J ($\approx$20GHz) oscillations. This region of the energy diagram holds great potential for precise quantum control of the system, however such schemes are currently impractical due to bandwidth limitations.

2.2.5 Coherence

We require that the coherence time of the qubit be much longer than operation times. This is often complicated to determine, because there are many kinds of decoherence (1.3). For the S-T_0 qubit, the question is further complicated by the fact that there are two distinct controls (J and ΔB_z), which have different coherence times, and by the fact that one of these controls (J) and the resulting operation times can be tuned over several orders of magnitude. Much of this work will deal with measuring and improving the coherence of the S-T_0 qubit, and we will therefore defer the discussion of this requirement to the later chapters.

2.3 Hyperfine Interaction and Dynamic Nuclear Polarization

The exchange interaction, J, is straightforward to control with a DC voltage, but we require easy control over *both* axes for realistic quantum computation. Therefore, we need a way to create and control a ΔB_z between the quantum dots. One approach, which has been adopted by several groups, is to evaporate a micromagnet on to of the quantum dots in order to generate a static ΔB_z[50]. We prefer to use the host nuclei of the GaAs, which, if fully polarized, generate a hyperfine magnetic field of 4-5T≈25GHz[62]. These nuclei, however, are a doubled edged sword: on the one hand, they offer a strong, intrinsic, fabrication-free method for generating a ΔB_z. On the other hand, fluctuations in the nuclear field cause ΔB_z to randomly change, and without some method for polarizing the nuclei, these fluctuations will have mean $\langle \Delta B_z \rangle = 0$ [¶]. As a result, coherence is quickly lost for ΔB_z operations, with a "bare" $T_2^* \approx 10$ns.

The interaction between a pair of electron spins and a bath of approximately a million nuclear spins is not fully understood and is the subject of active research. Though not perfectly understood, there is a dynamical nuclear polarization (DNP) scheme that is a useful tool for S-T_0 qubits.

[¶] We note that these fluctuations are also problematic for devices with micromagnets

We note, that there is point on the energy diagram where the $|S\rangle$ and the $|T_+\rangle$ states cross (Figure 2.2d). In reality, due to the hyperfine interaction[‖], there is a matrix element between these two states $\Delta B_x \, |S\rangle \, \langle T_+| + \text{h.c.}$ that creates an avoided crossing at this transition. Therefore, if we traverse this transition adiabatically, we can transition from $|S\rangle$ to $|T_+\rangle$ or vice versa, which changes the total spin of the electron system $|\Delta m_s| = 1$. Since the transition is driven by the hyperfine field, in order to conserve angular momentum, when an electron spin is flipped, it is accompanied by a nuclear spin flop. Therefore, by repeating cycles where the electrons are reloaded (either in $|S\rangle$ or $|T_+\rangle$) and swept through the S-T_+ transition, a net polarization may be pumped into the nuclei. Such a scheme will, in practice, generate a net B_z as well as a ΔB_z, presumably due to asymmetries in the shapes of the two quantum dots[27]. This ΔB_z has been used to demonstrate universal quantum control in S-T_0 qubits [23], and is an integral part of this work.

We can go one step further. This DNP scheme will set the mean ΔB_z to be nonzero, which allows us to drive, but not prolong, the coherence of ΔB_z operations. To do so, we employ a feedback scheme that sets the mean ΔB_z to a desired value as well and stabilizes the fluctuations to prolong coherence. This feedback is well describe in reference [6], and for completeness we outline it here. If we load a $|S\rangle$ and evolve the qubit around ΔB_z for a time τ, we can write the $|S\rangle$ return probability as $P(|S\rangle) = \cos^2(\Delta B_z \tau)$. If we subsequently sweep the qubit through the S-T_+ anticrossing we will only flip a nuclear spin if we return as a singlet. We can therefore write the probability of flipping a nuclear spin as $P_{flip}(\Delta B_z, \tau) \propto \cos^2(\Delta B_z \tau)$. If we continuously repeat such a cycle, we create a situation when the pumping efficiency depends on the size of ΔB_z, which is necessary to create a feedback loop. This feedback, however, is unstable, since it only polarizes the nuclei in one direction, and to make this loop stable, we interlace these "feedback" pump cycles with unconditional pumping in the opposite direction (electrons transitioning from $|S\rangle$ to $|T_+\rangle$).

[‖]the spin orbit interaction can contribute as well, depending on the orientation of the external magnetic field

Using this nuclear feedback scheme, T_2^* of ΔB_z is prolonged from $\approx$ 10ns to over 100ns. Despite the fact that the microscopic mechanism and limits are not well understood, this feedback technique is an integral part of this work. In practice, we run the feedback cycle every $\approx$ 100ms before continuing experiments in order to reset and re-stabilize ΔB_z.

Let's just say we'd like to avoid any Imperial entanglements.

Obi-Wan Kanobe

3

Demonstration of Entanglement of Electrostatically Coupled Singlet-Triplet Qubits

Quantum computers have the potential to solve certain interesting problems significantly faster than classical computers. To exploit the power of a quantum computation it is necessary to

This chapter is adapted from reference [58]

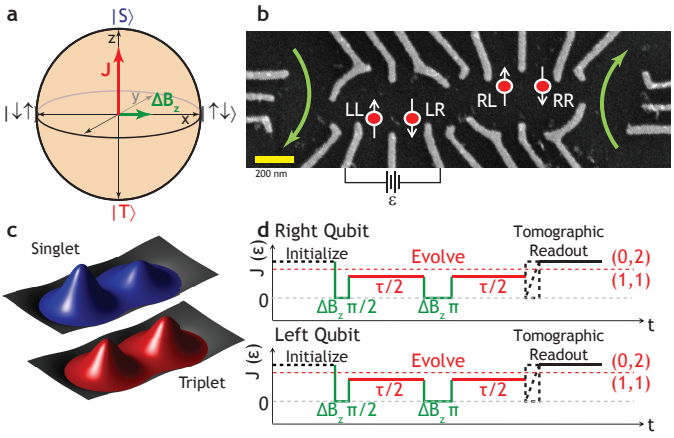

Figure 3.1: Two-qubit coupling scheme. a, A Bloch sphere can be used to describe the states of the effective two-level system defined by the singlet and triplet states of the qubit, with the z-axis along the S-T_0 axis and the x-axis along the $|\uparrow\downarrow\rangle/|\downarrow\uparrow\rangle$ axes. **b**, An SEM image of the top of the device used shows gates used to define the S-T_0 qubits (white), dedicated ns control leads, the approximate locations of the electrons in the two qubits (red), and current paths for the sensing dots (green arrows). The left qubit uses the LR and LL electrons, while the right qubit uses the RL and RR electrons. **c**, A schematic of the electronic charge configurations for the $|S\rangle$ (blue) and the $|T_0\rangle$ (red) at non-zero J. This difference in charge configuration is the basis for the electrostatic coupling between the qubits **d**, The pulse sequence used to entangle the qubits: initialize each qubit in a $|S\rangle$, perform a $\pi/2$ rotation around the x-axis, allow the qubits to evolve under exchange for a time $\tau/2$, perform a π-rotation around the x-axis, thereby decoupling the qubits from the environment but not each-other, evolve under exchange for $\tau/2$, and perform state tomography to determine the resulting density matrix (see chapter 4.)

perform inter-qubit operations and generate entangled states. Spin qubits are a promising candidate for implementing a quantum processor due to their potential for scalability and miniaturization. However, their weak interactions with the environment, which leads to their long coherence times, makes inter-qubit operations challenging. We perform a controlled two-qubit operation between singlet-triplet qubits using a dynamically decoupled sequence that maintains the two-qubit coupling while decoupling each qubit from its fluctuating environment. Using state tomography we measure the full density matrix of the system and determine the concurrence and the fidelity of the generated state, providing proof of entanglement.

3.1 Introduction

Singlet-triplet (S-T_0) qubits, a particular realization of spin qubits[38, 34, 50, 46, 51, 12, 44], store quantum information in the joint spin state of two electrons[37, 49, 62].The basis states for the S-T_0 qubit can be constructed from the eigenstates of a single electron spin, $|\uparrow\rangle$ and $|\downarrow\rangle$. We choose $|S\rangle = \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle)$ and $|T_0\rangle = \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle)$ for the logical subspace of the S-T_0 qubit because these states are insensitive to uniform fluctuations in the magnetic field. The qubit can then be described as a two level system with a representation on a Bloch sphere shown in Figure 3.1a. Universal quantum control is achieved using two physically distinct operations that drive rotations around the x and z-axes of the Bloch sphere [23]. Rotations around the z-axis of the Bloch sphere are driven by the exchange splitting, J, between $|S\rangle$ and $|T_0\rangle$, and rotations around the x-axis are driven by a magnetic field gradient, ΔB_z between the electrons.

We implement the S-T_0 qubit by confining two electrons to a double quantum dot (QD) in a two dimensional electron gas (2DEG) located 91nm below the surface of a GaAs-AlGaAs heterostructure. We deposit local top gates using standard electron beam lithography techniques in order to locally deplete the 2DEG and form the QDs. We operate between the states (0,2) and (1,1) where (n_L,n_R) describes the state with $n_L(n_R)$ electrons in the left (right) QD. The $|S\rangle$ and $|T_0\rangle$ states, the logical subspace for the qubit, are isolated by applying an external magnetic field of B =700mT in the plane of the device such that the Zeeman splitting makes $T_+ = |\uparrow\uparrow\rangle$, and $T_- = |\downarrow\downarrow\rangle$ energetically inaccessible. The exchange splitting, J, is a function of the difference in energy, ϵ, between the levels of the left and right QDs. Pulsed DC electric fields rapidly change ϵ, allowing us to switch J on, which drives rotations around the z-axis. When J is off the qubit precesses around the x-axis due to a fixed ΔB_z, which is stabilized to $^{\Delta B_z}/_{2\pi}$ =30MHz by operating the qubit as a feedback loop between interations of the experiment[6]. Dephasing of the qubit rotations reflects fluctuations in the magnitude of the two control axes, J and ΔB_z, caused by electrical

noise and variation in the magnetic field gradient, respectively. The qubit is rapidly (<50ns) initialized in $|S\rangle$ by exchanging an electron with the nearby Fermi sea of the leads of the QD in a region of (0,2) where only $|S\rangle$ is accessible, and the qubit state is read out using standard Pauli blockade techniques, where ϵ is quickly tuned to the regime where S occupies (0,2) and T_0 occupies (1,1), allowing the qubit state to be determined by the proximal charge sensor. The charge state of the qubit is rapidly read ($\sim 1\mu s$) using standard RF- techniques [53, 4] on an adjacent sensing QD.

3.2 Two-Qubit Coupling

In order to make use of the power of quantum information processing it is necessary to perform two qubit operations in which the state of one qubit is conditioned on the state of the other[45]. To investigate two- qubit operations we fabricate two adjacent S-T_0 qubits such that they are capacitively coupled, but tunneling between them is suppressed (Figure 3.1b). A charge sensing QD next to each qubit allows for simultaneous and independent projective measurement of each qubit (see chapter 4). We use the electrostatic coupling between the qubits to generate the two-qubit operation[61]. When J is nonzero, the S and T_0 states have different charge configurations in the two QDs due to the Pauli exclusion principle (Fig. 1c). This charge difference, which is a function of ϵ, causes the $|S\rangle$ and $|T_0\rangle$ states in one qubit to impose different electric fields on the other qubit. Since J is a function of the electric field, the change imposed by the first qubit causes a shift in the precession frequency of the second qubit. In this way the state of the second qubit may be conditioned on the state of the first qubit. More precisely, when a single qubit evolves under exchange, there exists a state- dependent dipole moment, $\vec{d}$, between $|S\rangle$ and $|T_0\rangle$ resulting from their difference in charge occupation of the QDs. Therefore, when simultaneously evolving both qubits under exchange, they experience a capacitively mediated, dipole-dipole coupling that can generate an entangled state. The two-qubit Hamiltonian is therefore given by:

$$\mathcal{H}_{2-qubit} = \frac{\hbar}{2}\left(J_1(\sigma_z \otimes I) + J_2(I \otimes \sigma_z) + \frac{J_{12}}{2}\left((\sigma_z - I) \otimes (\sigma_z - I)\right)\right) + \frac{\hbar}{2}\left(\Delta B_{z,1}(\sigma_x \otimes I) + \Delta B_{z,2}(I \otimes \sigma_x)\right) \quad (3.1)$$

where $\sigma_{x,y,z}$ are the Pauli matrices, I is the identity operator, $\Delta B_{z,i}$ and J_i are the magnetic field gradients and the exchange splittings (i=1,2 for the two qubits), and J_{12} is the two-qubit coupling, which is proportional to the product of the dipole moments in each qubit. For a two level system with constant tunnel coupling, the dipole moment scales as $\vec{d_i} \propto \frac{\partial J_i}{\partial \epsilon_i}$. Empirically, we find that for experimentally relevant values of J_i, $\frac{\partial J_i}{\partial \epsilon_i} \propto J_i(\epsilon)$, so that $J_{12} \propto J_1 J_2$. As with the single qubit operations, this two-qubit operation requires only pulsed DC electric fields.

3.3 Dynamically Decoupled Gate

In principle, evolving both qubits under exchange produces an entangling gate. However, the time to produce this maximally entangled state exceeds the inhomogeneously broadened coherence times of each individual qubit, rendering this simple implementation of the two-qubit gate ineffective. To mitigate this we use a dynamically decoupled entangling sequence[67, 36] (Figure 3.1d). In this sequence, each qubit is prepared in $|S\rangle$ and is then rotated by $\frac{\pi}{2}$ around the x-axis ($J_i = 0$, $\Delta B_{z,i}/2\pi \approx 30$MHz) to prepare a state in the x-y plane. The two qubits are subsequently both evolved under a large exchange splitting ($J_1/2\pi \approx 280\text{MHz},\ J_2/2\pi \approx 32\text{MHz} \gg \Delta B_z$) for a time $\frac{\tau}{2}$, during which the qubits begin to entangle and disentangle. A π-pulse around the x-axis (ΔB_z) is then applied simultaneously to both qubits, after which the qubits are again allowed to exchange for a time $\frac{\tau}{2}$. This Hahn echo-like sequence[28] removes the dephasing effect of noise that is low frequency compared to $1/\tau$, while the π-pulses preserve the sign of the two-qubit interaction.

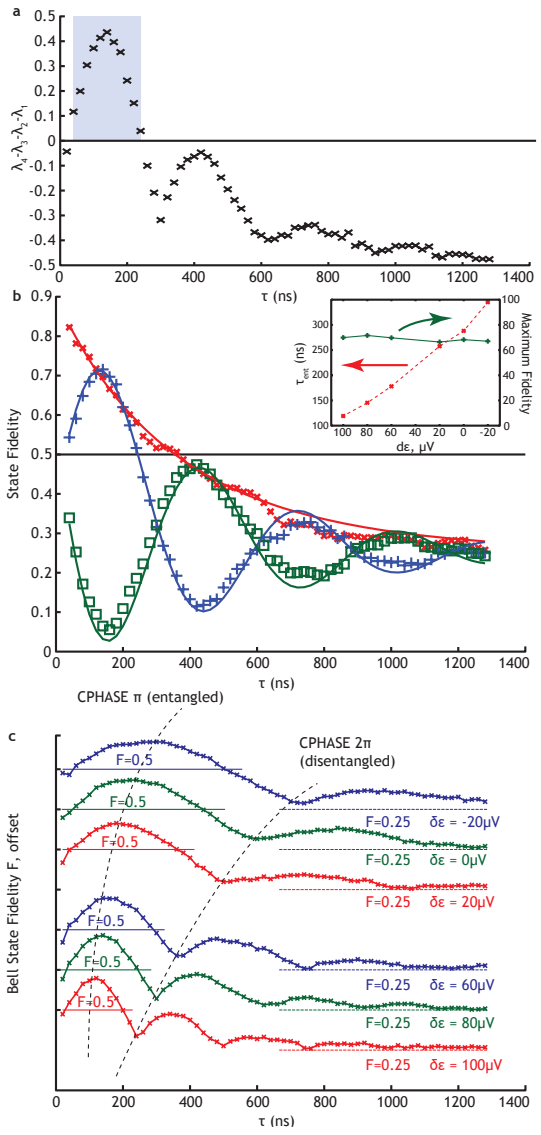

Figure 3.2: Proof of entanglement: concurrence and state fidelity. a, A plot of the difference of the sorted eigenvalues of the matrix R, which for positive values is equal to the concurrence $c\left(\rho\right)$. States with a concurrence greater than zero (shaded region) are necessarily entangled. **b**, The fidelity with which the measured state approximates the target $\left|\Psi_{ent}\right\rangle$ (blue), and $e^{i\pi(\sigma_y\otimes I+I\otimes\sigma_y)/4}\left|\Psi_{ent}\right\rangle$ (green), which differs from $\left|\Psi_{ent}\right\rangle$ by single qubit rotations and is the expected state for $\tau = {}^{3\pi}/_{2J_{12}}$. The fidelity with which the measured state approximates a dephasing-free model of the entangling operation (red) shows smooth decay due to decoherence. The solid lines are fits to the data. Inset: The time to produce a maximally entangled state as a function of the change in ϵ (and therefore J) in the two qubits. As J increases τ_{ent} decreases (red), but the maximum attainable fidelity (green) is approximately constant. Arrows indicate which y-axis is to be used. **c**, The Bell state fidelity as a function of time for different values of J (offset) with guides to show where the fidelity exceeds ${}^1/_2$ for each curve. As J increases in the two qubits the time to produce an entangled state, τ_{ent}, decreases.

The resulting operation produces a CPHASE gate, which, in a basis of $\{|SS\rangle, |T_0S\rangle, |ST_0\rangle, |T_0T_0\rangle\}$, is an operation described by a matrix with $e^{-i\theta/2}$,1,1,$e^{-i\theta/2}$ on the diagonals. For $\tau = \tau_{ent} = \frac{\pi}{2J_{12}}$ ($\theta = \pi$) the resulting state is a maximally entangled generalized Bell state $|\Psi_{ent}\rangle = e^{i\pi(I\otimes\sigma_y+\sigma_y\otimes I)/8}|\Psi_-\rangle$ which differs from the Bell state $|\Psi_-\rangle = \frac{1}{\sqrt{2}}(|SS\rangle - |T_0T_0\rangle)$ by single qubit rotations.

3.4 Entanglement Verification

In order to characterize our two-qubit gate and verify that we produce an entangled state we perform two-qubit state tomography and extract the density matrix and appropriate entanglement measures. The tomographic procedure is carefully calibrated with minimal assumptions in order to avoid adding spurious correlations to the data that may artificially increase the measured degree of entanglement (chapter 4). We choose the Pauli set representation of the density matrix[31, 45, 11], where we measure and plot the 16 two-qubit correlators $\langle ij\rangle = \langle\sigma_i\sigma_j\rangle$ where σ_i are the Pauli matrices and $i, j \in \{I, X, Y, Z\}$. As a first measure of entanglement, we evaluate the concurrence[30] (Figure 3.2a), $C(\rho) = \max\{0,\ \lambda_4 - \lambda_3 - \lambda_2 - \lambda_1\}$ for different τ, where ρ is the experimentally measured density matrix, and λ_i are the eigenvalues, sorted from largest to smallest, of the matrix $R = \sqrt{\sqrt{\rho}\tilde{\rho}\sqrt{\rho}}$, and $\tilde{\rho} = (\sigma_y \otimes \sigma_y)\rho^*(\sigma_y \otimes \sigma_y)$, and ρ^* is the complex conjugate of ρ. A positive value of the concurrence is a necessary and sufficient condition for demonstration of entanglement [30]. For τ =140ns we extract a maximum concurrence of 0.44.

While a positive value of the concurrence is a definitive proof of entanglement, it alone does not verify that the two-qubit operation produces the intended entangled state. In order to better characterize the generated quantum state, we evaluate another measure of entanglement, the Bell state fidelity, $F \equiv \langle\Psi_{ent}|\rho|\Psi_{ent}\rangle$. This may be interpreted as the probability of measuring our two-qubit state in desired $|\Psi_{ent}\rangle$. Additionally, for all non-entangled states one can show that $F \leq 0.5$ [54, 5]. In terms of the Pauli basis, the Bell state fidelity takes the simple form $F = \frac{1}{4}\vec{P}_{ent} \cdot \vec{P}_{experiment}$

where $\vec{P}_{ent}$ and $\vec{P}_{experiment}$ are the Pauli sets of a pure target Bell state and of the experimentally measured state, respectively. For our target state $|\Psi_{ent}\rangle$, the resulting Pauli set is given by $\langle XZ\rangle = \langle ZX\rangle = \langle YY\rangle$ =1, with all other elements equal to zero (Figure 3.3a).

In an idealized, dephasing-free version of the experiment, as τ increases and the qubits entangle and disentangle, we expect the nonzero elements of the Pauli set for the resulting state to be

$$\langle YI\rangle = \langle IY\rangle = \cos\left(J_{12}\tau\right),\ \langle XZ\rangle = \langle ZX\rangle = \sin\left(J_{12}\tau\right),\ \langle YY\rangle = 1 \tag{3.2}$$

Dephasing due to electrical noise causes the amplitudes of the Pauli set to decay. However, the two-qubit Hamiltonian (equation 3.1) includes rapid single-qubit rotations around the S-T_0 axis (J_1,J_2 $\gg J_{12}/2\pi$ $\approx$ 1MHz) that change with τ due to imperfect pulse rise times in the experiment. These add additional single-qubit rotations around the S-T_0 axis of each qubit, which are not accounted for in equation 3.2. We determine the angle of the single qubit rotations by performing a least-squares fit of the experimental data to modified form of equation 3.2 that accounts for these rotations and dephasing. The decays due to dephasing are fit by calculating $\rho\left(t\right)$ in the presence of noise on J_1 and J_2, which leads to decay of certain terms in the density matrix [72, 15]. For the present case where $J_{12} \ll J_1,\ J_2$, we neglect the two-qubit dephasing, which is smaller than single-qubit dephasings by a factor of $\frac{J_1}{J_{12}}$, $\frac{J_2}{J_{12}}$ $\approx$300, and we extract a separate dephasing time for each individual qubit. We remove the single-qubit rotations numerically in order to simplify the presentation of the data (Figure 3.3e). The extracted angles exhibit a smooth monotonic behavior which is consistent with their underlying origin (see 3.6.1).

In the absence of dephasing we would expect the Bell state fidelity to oscillate between 0.5 for an unentangled state and 1 for an entangled state as a function of τ. This oscillation is caused by the phase accumulated by a CPHASE gate between the two qubits. However, the qubits dephase as the state becomes increasingly mixed, and this oscillation decays to 0.25. Indeed, this behavior is

observed (Figure 3.2b): for very short τ there is very little dephasing present, and the qubits are not entangled. As τ increases the Bell state fidelity increases as the qubits entangle, reaching a maximum value of 0.72 at τ =140ns. As τ is increased further, we continue to see oscillations in the Bell state fidelity , though due to dephasing, they do not again rise above 0.5.

Figure 3.2c shows these oscillations in Bell state fidelity as a function of τ for several different values of J as ϵ is changed symmetrically in the two qubits. We see that as the value of J increases in the two qubits, the time required to produce a maximally entangled state, τ_{ent}, decreases, but the maximum attainable fidelity is approximately constant. This is consistent with the theory that $J_{12} \propto \frac{\partial J_1}{\partial \varepsilon_1} \cdot \frac{\partial J_2}{\partial \varepsilon_2} \propto J_1 \cdot J_2$.

3.5 Entangled States

To further understand the evolution of the quantum state, we focus on one value of J and compare the measured Pauli set to that expected from single-qubit dephasing rates and J_{12} (3.6.2). Figure 3.3a shows the Pauli set for the measured and expected quantum states for τ =40ns, which shows three large bars in the $\langle YI \rangle$,$\langle IY \rangle$, and $\langle YY \rangle$ components of the Pauli set. This is a nearly unentangled state. At τ =140ns, we see weight in the in the $\langle XZ \rangle$, $\langle ZX \rangle$, $\langle YY \rangle$ components of the Pauli set (Figure 3.3b), and we extract a Bell state fidelity of 0.72, which demonstrates the production of an entangled state. For $\tau = \tau_{ent} = \frac{\pi}{2J_{12}}$= 160ns (Figure 3.3c) we see a similar state to τ =140ns, but with less weight in the single qubit components of the Pauli set. This state corresponds to the intended CPHASE of π, though it has a slightly lower fidelity than the state at τ =140ns due to additional decoherence. Finally, at $\tau = \frac{\pi}{J_{12}}$ =320ns (Figure 3.3d), where we expect the state to be unentangled, we again see large weight in the $\langle YI \rangle$, $\langle IY \rangle$, and $\langle YY \rangle$ components of the Pauli set, though the bars are shorter than the Pauli set for τ =40ns, due to dephasing of the qubits. We plot the entire Pauli set as a function of time (Figure 3.3e), which clearly shows the predicted oscillation

(equation3.2) between $\langle YI\rangle$,$\langle IY\rangle$ and $\langle XZ\rangle$, $\langle ZX\rangle$, with decays due to decoherence.

The two-qubit gate that we have demonstrated is an important step toward establishing a scalable architecture for quantum information processing in S-T_0 qubits. State fidelity is lost to dephasing from electrical noise, and decreasing the ratio $\frac{\tau_{ent}}{T_2^{echo}}$, where T_2^{echo} is the single-qubit coherence time with an echo pulse, is therefore paramount to generating high-fidelity Bell states. Large improvements can be made by introducing an electrostatic coupler between the two qubits [63] in order to increase the two-qubit coupling (J_{12}) and reduce τ_{ent}. We estimate that in the absence of other losses, if an electrostatic coupler were used, a Bell state with fidelity exceeding 90% could be produced. Additional improvements can be made by studying and mitigating the origins of charge noise to increase T_2^{echo}. This would allow future tests of complex quantum operations including quantum algorithms and quantum error correction. The generation of entangled also states opens the possibility of studying the complex dynamics of the nuclear environment, which is a fundamental, quantum, many body problem.

3.6 Supplementary Information

3.6.1 Determining Single Qubit Rotations

During the entangling sequence the two qubits rotate very rapidly around the S-T_0 axis compared to the speed of the CPHASE gate ($J_1/2\pi \sim J_2/2\pi \sim 300MHz$, $J_{12}/2\pi \sim 1MHz$). These single qubit rotations are not perfectly canceled out by the π-pulses in the dynamically decoupled sequence due to pulse distortions, consistent with pulse rise time effects at short times and capacitive coupling to RC-filtered DC gates at long times. Moreover, the angles by which the qubits are rotated change as a function of the evolution time τ. In order to undo these rotations, we perform a least-square fit of the data to the expected form of the Pauli set (see equation3.3 below), restricting the rotation to be around the S-T_0 axis because $J_1, J_2 \gg \Delta B_z$. These angles are shown in Figure 3.4d, and

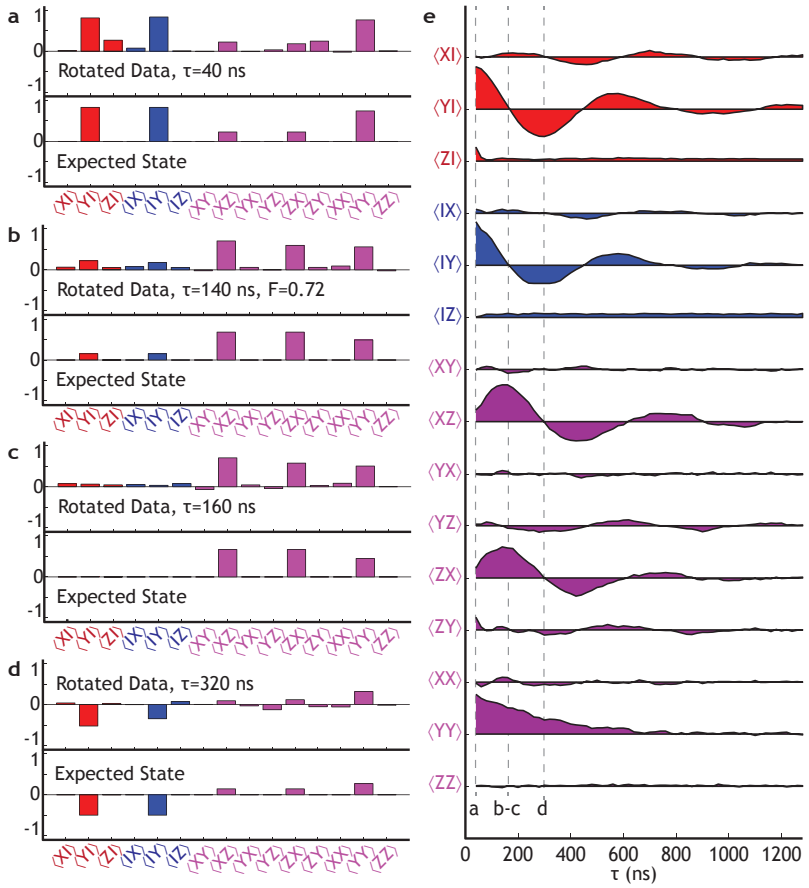

Figure 3.3: Pauli set representation. a, The elements of the Pauli set for the measured density matrix and the state expected from the entangling and dephasing rates for small τ (unentangled). **b**, The Pauli set of the measured and expected states for τ =140ns, which produces a maximum Bell state fidelity of 0.72. **c**, The Pauli set of the measured and expected states for $\tau = \tau_{ent}$ =160ns, which is a CPHASE of π but does not the highest fidelity due to dephasing. **d**, The Pauli set for the measured and expected states for τ =320ns, which is an unentangled state. **e**, The full measured Pauli set as a function of τ, which shows the expected behavior for a CPHASE gate. The the y-axes of adjacent elements in the Pauli set are offset by 1.

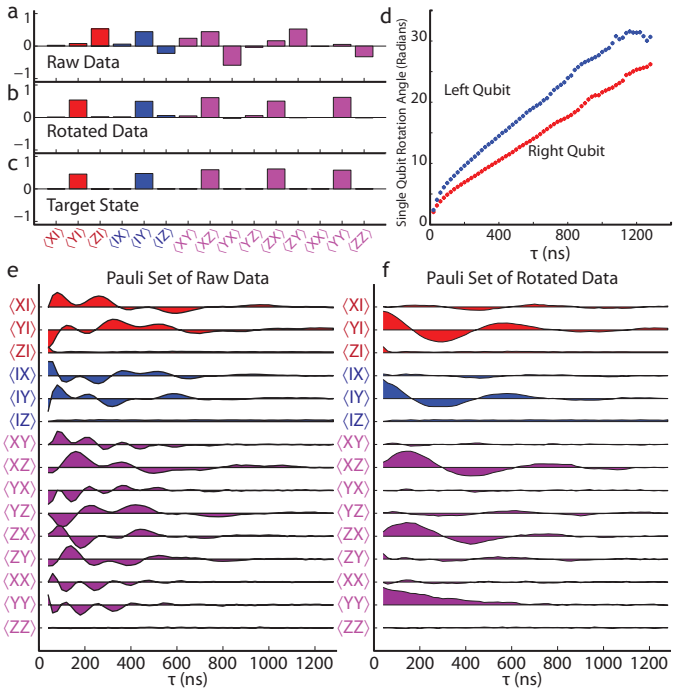

Figure 3.4: Single-qubit rotations: **a**, The Pauli set for τ =100ns as measured is complicated by single qubit rotations. **b**, Numerically rotating each qubit around the S-T_0-axis simplifies presentation and analysis. **c**, The expected state for τ =100ns. **d**, The single qubit rotation angles for both qubits as a function of τ are smooth and monotonic functions. **e-f** The entire Pauli set as a function of τ for the raw and rotated data equation 3.3. The the y-axes of adjacent elements in the Pauli set are offset by 1.

exhibit a smooth, monotonic behavior. The angles increase quickly for small τ, which is consistent with pulse rise time effects, and display linear behavior for long τ, which is consistent with long time RC filtering. For comparison, we plot the entire Pauli set for both the rotated and unrotated data in Figure 3.4e-f.

3.6.2 Fit Form for a Dephased Bell State

In order to fit the experimentally measured Pauli sets we calculate $\rho(t)$ in the presence of fluctuations on J_1, and J_2, and average over all these fluctuations[25,26]. For the present case where $J_{12} \ll J_1,\ J_2$, we neglect the two-qubit dephasing, which is smaller than single-qubit dephasings by a

factor of $\frac{J_1}{J_{12}}$, $\frac{J_2}{J_{12}} \approx 300$. This yields the following non-zero elements of the Pauli set

$$\begin{aligned}
\langle YI \rangle &= e^{-\tau/T_{2,1}} \cos(J_{12}\tau) \\
\langle IY \rangle &= e^{-\tau/T_{2,2}} \cos(J_{12}\tau) \\
\langle XZ \rangle &= e^{-\tau/T_{2,1}} \sin(J_{12}\tau) \\
\langle ZX \rangle &= e^{-\tau/T_{2,2}} \sin(J_{12}\tau) \\
\langle YY \rangle &= e^{-\tau/T_{2,1}} e^{-\tau/T_{2,2}}
\end{aligned} \tag{3.3}$$

where $T_{2,i}$ are the single-qubit coherence times of the two qubits. From fits of the data to this form we extract for $\delta\epsilon$ =0: $T_{2,1}$ =420ns, $T_{2,2}$ =510ns, and $J_{12}/2\pi$ =0.87MHz. Imperfect state preparation, in which the $\pi/2$-pulse does not leave the state of the qubits in the x-y-plane causes mixing between terms, which is visible in the form of small amplitude oscillations in the $\langle XI \rangle$,$\langle IX \rangle$, and $\langle XX \rangle$ components of the Pauli set (Figure 3.4f).

Acknowledgments

This work is supported through the ARO, "Precision Quantum Control and Error-Suppressing Quantum Firmware for Robust Quantum Computing" and IARPA "Multi-Qubit Coherent Operations (MQCO) Program." This work was partially supported by the US Army Research Office under Contract Number W911NF-11-1-0068. This work was performed in part at the Center for Nanoscale Systems (CNS), a member of the National Nanotechnology Infrastructure Network (NNIN), which is supported by the National Science Foundation under NSF award no. ECS-0335765. CNS is part of Harvard University.

Measure what is measurable, and make measurable what is not so.

Galileo Galilei

4

Calibration of State Tomography in Singlet-Triplet Qubits

As quantum information science progresses , and experimental capabilities advance, it becomes necessary to develop techniques to measure new quantities. The process of measuring the full

Oliver Dial must be credited with the idea for this tomography calibration

density matrix of a system (quantum state tomography), which has now become standard practice in many systems, determines the elements of the density matrix by repeated projective measurements of different axes of the system.

Accurate and calibrated state tomography is important for characterizing quantum systems, and it is particularly important in entanglement verification experiments, where inaccurate tomography can give rise to spurious correlations, which artificially enhance the measured entanglement. However, there is a fundamental problem in calibrating state tomography: most qubits have measurement capabilities along one axes, and to measure the projection around the other two, it is necessary to rotate the qubit before projecting it. However, how can these rotations themselves be calibrated without a calibrated state tomography procedure? Put differently, calibrated rotations are needed to calibrate state tomography, and calibrated state tomography is needed to calibrate qubit rotations. Fortunately, established bootstrap and tune-up procedures[19] have been devised to skirt this issue.

Though it may be standard in other systems, state tomography is relatively new in gate-defined semiconductor spin qubits, since these qubits are relatively less-established. Though spin qubits are able to borrow a lot of technology from other qubits, unfortunately, the standard tune-up procedures are difficult in semiconductor spin qubits, especially in singlet-triplet qubits. First, due to the limited bandwidth in the cables that deliver signals from room temperature down to milliKelvin temperatures, pulse risetimes make signals dependent on the signals that preceded them. The unwanted consequence of this effect is that the tomographic procedure would affect different rotations depending on the experiment that proceeded them. Second, the x-axis readout in S-T_0 qubits involves adiabatically tuning on J, in order to map the x-axis into the x-axis. Though this rotation is robust, it cannot be tuned or altered, which renders the standard tomography calibration schemes ineffective.

Here we describe a novel technique for calibrating one-qubit and two-qubit state tomography in S-T_0 qubits, which was an integral part of entanglement verification in these qubits (chapter 3).

Instead of tuning the tomographic rotations to perfectly map the x and y axes onto the z-axis, this procedure does not tune the rotation at all, but rather precisely measures the actual axes onto which the quantum state is projected. The density matrix is therefore determined in a strange basis, but can trivially be transformed into the standard basis. The procedure makes minimal assumptions about the tools that are available, and is therefore general and may be used for many types of qubits. We first detail the calibration of the sensor and the elimination of readout crosstalk, and then we describe the single-qubit tomography calibration procedure.

4.1 Calibration of RF Sensor Response

In order to quantitatively interpret sensor values for state tomography, it is important to precisely determine the RF sensor response that corresponds to a $|S\rangle$ or a $|T_0\rangle$ state. Because the state preparation is imperfect, it is in general difficult to accurately measure these values experimentally. To provide exact calibrations for $|S\rangle$ and $|T_0\rangle$, we exploit the fact that our sensor is capable of single shot readout. Histograms of sensor values for typical measurements yield a double-peaked curve- one peak corresponds to $|T_0\rangle$ and one to $|S\rangle$ (Figure 4.1a). In order to calibrate the sensor we first measure T_1 at the measurement point by preparing a state that is majority $|T_0\rangle$ (done with a π-pulse around the x-axis) and fitting the sensor signal to a decaying exponential function of time elapsed before measurement (Figure 4.1a). We note that the measured value of T_1 is a strong function of the power of the RF excitation used to read the conductance of the sensing QD. With prior knowledge of T_1, we use a procedure similar to that described in Barthel et. al[4] to optimize the measurement time given our signal to noise ratios and T_1. This process is repeated several times per day to check for drift. We recalibrate the sensor signals that correspond to $|S\rangle$ and $|T_0\rangle$ for each dataset (typically 10 minutes of acquisition time). For each set, we prepare a histogram of all observed sensor values. The presence of several reference measurements in each dataset guarantees that there will be a signif-

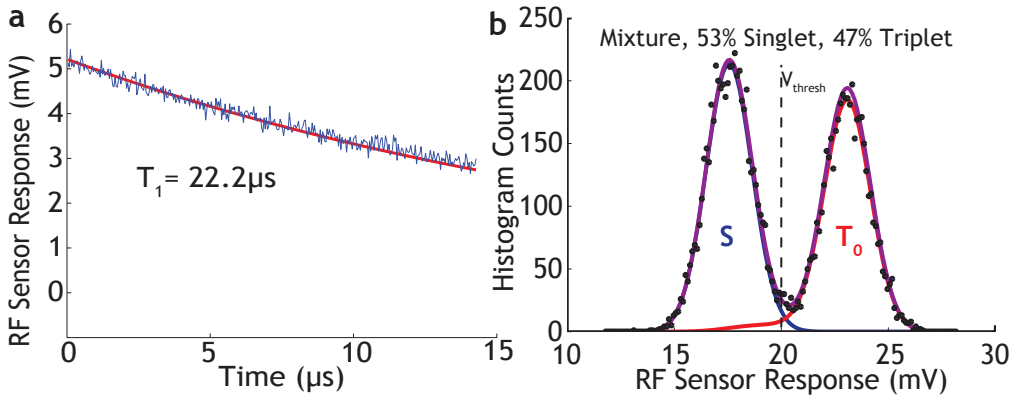

Figure 4.1: Singleshot Readout: a, The difference in sensor signal between $|S\rangle$ and $|T_0\rangle$ is fit to a decaying exponential to determine T_1, which is used in calibration of sensor values. **b,** The histograms of a mixture of $|S\rangle$ and $|T_0\rangle$ states used to calibrate the sensor values. If we choose a threshold V_{thresh} to distinguish between $|S\rangle$ and $|T_0\rangle$ we see a readout fidelity of 97%. Purple: fit to noisy distribution including T_1 decay from $|T_0\rangle$ to $|S\rangle$. The deduced distribution for $|S\rangle$ (blue) is a Gaussian, while that for $|T_0\rangle$ (red) has a tail due to T_1 decay.

icant fraction of both $|S\rangle$ and $|T_0\rangle$. We then fit this double peaked curve to an analytic expression corresponding to a weighted sum of two Gaussians with some filling in due to T_1 decay during measurement (Figure 4.1b, purple line) as in ref. 14. From this, we extract the expected sensor distributions for $|S\rangle$ and $|T_0\rangle$ (blue and red lines in Figure 4.1b, respectively), as well as the fractions of $|S\rangle$ and $|T_0\rangle$ present. The centers of the two distributions correspond to the sensor signals that will be measured for pure $|S\rangle$ and pure $|T_0\rangle$, and using these values we can accurately scale the tomography data. We note that this procedure is insensitive to the percentages of $|S\rangle$ and $|T_0\rangle$. In our state tomography only expectation values are needed, so the single-shot capability of our readout is not necessary beyond this calibration. Nonetheless, we note that for the data presented, we measured readout fidelities of 97% and 98% for the left and right qubits, respectively.

4.2 Readout Crosstalk

For accurate state tomography it is important that the readout of the two qubits be independent. The two-qubit coupling relies on the fact that the $|S\rangle$ and $|T_0\rangle$ in one qubit electrostatically gate the

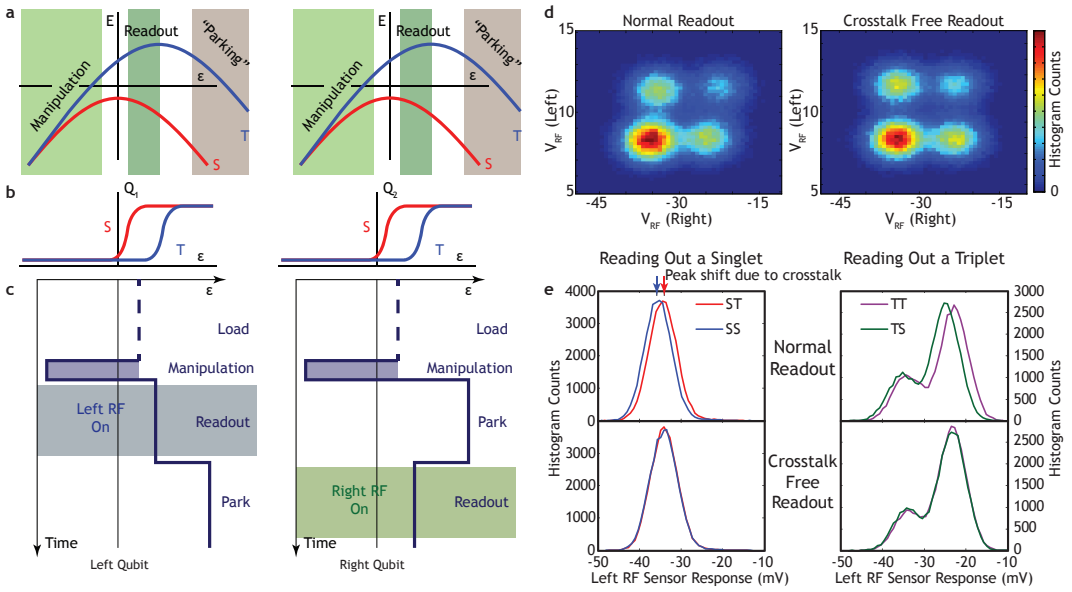

Figure 4.2: Readout Crosstalk: a, A schematic of the energy diagram as a function of ϵ that describes the two qubits and shows the regions of ϵ where different operations are carried out. **b,** A schematic of the signal from the RF charge sensor as a function of ϵ for the two qubits. This signal reflects the charge distribution of the two qubit states. For large positive ϵ there is a region where $|S\rangle$ and $|T_0\rangle$ have the same sensor signal (charge distribution), which is the foundation of the crosstalk-free readout scheme. **c,** A schematic of the readout scheme that eliminates crosstalk. First, the left qubit is read while the right qubit is "parked" in (0,2), and then the right qubit is read while the left qubit is "parked" in (0,2). **d,** A two dimensional histogram of the RF sensor responses without (left panel) and with (right panel) this crosstalk-free readout scheme. **e,** Histograms of sensor values without (top) and with (bottom) the crosstalk-free readout. Without the crosstalk-free readout the sensor signal of one qubit depends on the state of the other qubit.

other qubit, and reading out the two qubits simultaneously leads to readout crosstalk; the left-qubit sensor value will be different for $|S\rangle|T_0\rangle$ and $|S\rangle|S\rangle$, and similarly for the right qubit (Figure 4.2d-e). We avoid this problem by reading out the qubits sequentially: while one qubit is read out the other qubit is "parked" deep inside of (0,2) where the (0,2)$|T_0\rangle$ is lower in energy than the (1,1)$|T_0\rangle$ so that both the $|S\rangle$ and $|T_0\rangle$ occupy (0,2) (Figure 4.2a-c). In this way each qubit is read out while the other qubit has the same charge distribution for the both qubit states. T_1 in this region (with the RF excitation off) is large, so no measurable degradation of the qubit being stored occurs (dephasing around the S-T_0 axis preserves readout fidelity).

4.3 Calibration of State Tomography

State tomography involves reading the projection of a qubit on to the three Cartesian axes of the Bloch sphere[45]. However, the charge sensor next to each qubit allows us to determine only the projection on to the z-axis (S-T_0) of the qubit. Therefore, in order to perform state tomography, we apply rotations that map the x and y axes of the qubit to the z-axis[23]. The y-axis component is mapped on to the minus z-axis by a $\pi/2$-pulse around the x-axis, which is driven by ΔB_z, and the x-axis component is mapped on to the z-axis by adiabatically turning on J, which maps the eigenstate $|\uparrow\downarrow\rangle(|\downarrow\uparrow\rangle)$ into $|S\rangle$ ($|T_0\rangle$) (Figure 4.2a).

Extreme care must be taken when performing two-qubit state tomography, as errors in tomography can introduce spurious correlations which might inflate the measured degree of entanglement. However, in S-T_0 qubits, traditional methods for calibration are difficult to implement. The two control axes, J and ΔB_z, are not orthogonal, and precise 90 degree rotations are problematic due to the timing resolution available on our signal generators as well as pulse rise times. Pulse rise times in the experimental apparatus prevent instantaneous changes in J, which changes both the axis of rotation and the total angle rotated. These rise times also cause sudden changes in J to become

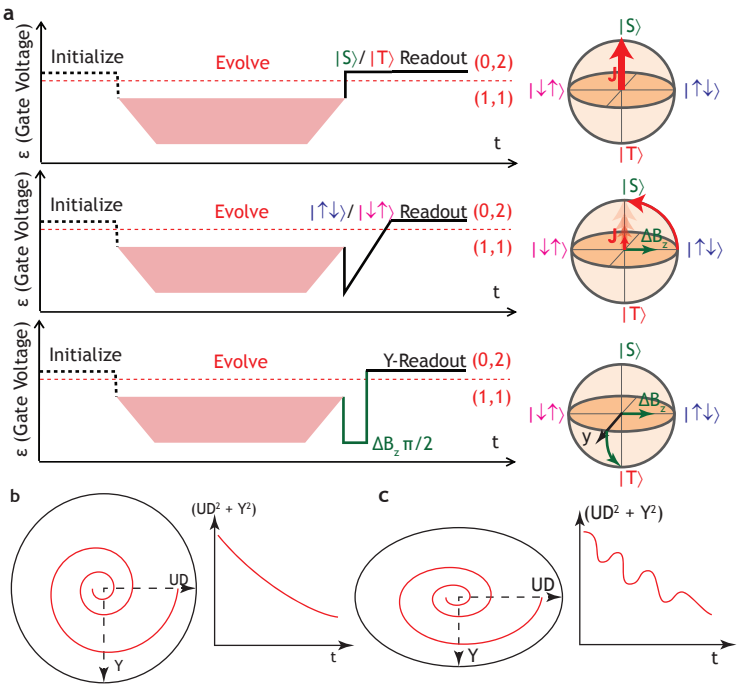

Figure 4.3: Calibration of state tomography: a, A schematic of how state tomography is performed in S-T_0 qubits[23]. The S-T_0 component is read out by charge sensing in a region where $|S\rangle$ occupies (0,2) and $|T_0\rangle$ occupies (1,1). The x (y) component is read adiabatically turning on J (rotating by $\pi/2$ around the x-axis), followed by charge sensing. **b**, A schematic of the length of the Bloch vector for perfect state tomography. **c**, A schematic of the "ripples" in the length of the Bloch vector if state tomography is flawed. The tomographic axes are determined by minimizing these ripples.

somewhat adiabatic. Furthermore, the effects of these pulse rise times depend both on the starting and ending point of the an individual pulse. Therefore, we perform careful measurements in order to determine the three axes on to which the qubit is projected, and apply a transformation in order to map these axes onto the traditional Cartesian axes. Dephasing during readout rotations is represented in this process as readout axes that are longer than the radius of Bloch sphere.

The procedure for determining the the tomographic axes makes minimal assumptions. We first calibrate the the S-T_0 readout using the singleshot histograms (see above). We then assume that any path of the state of the qubit around the Bloch sphere in a free-induction-decay experiment should smoothly dephase, i.e., there should not be oscillations in the amplitude of the Bloch vector. No

assumptions are made about the axis or frequency of any rotation. If either the length or the angle of one of the tomography axes is incorrect, we expect to see "ripples" in the length of the measured Bloch vector as a function of evolution time (Figure 4.3b-c). If we simultaneously consider many paths of evolution around the Bloch sphere, errors in the angles and lengths of the different read-out axes become distinguishable due to varying phases, periods, and amplitudes of the ripples in the lengths of the measured vectors (Figure 4.4a). Therefore, to calibrate our axes, we gather data on many different evolutions around the Bloch sphere by evolving from many different starting points at many values of ϵ (Figure 4.4b,c,e,f). We determine the axes on to which we project our state by finding the axes that minimize the amplitude of the ripples in the length of the Bloch vectors (Figure 4.4d). Based on our measurement procedure, we define the S-T_0 axis to lie along the z-axis. We allow the y-axis to lie anywhere on the Bloch sphere because a rotation around the x-axis can suffer from over/under rotation as well as adiabaticity issues with switching J on and off instantly. We constrain the x-axis to lie in the x-z-plane because the only expected error is due to adiabaticity turning J on and off. The typical tomographic axes are shown in Figure 4.4d, and the signs of the errors are consistent with their origins. The variation from calibration to calibration is $\sim$1% on the axis lengths and angles.

4.4 Process Tomography

State tomography is a powerful tool, which enables many capabilities beyond entanglement verification. One such application is process tomography, which is a procedure for completely characterizing a quantum process. A quantum process, which need not be unitary, transforms one quantum state (density matrix) into another.

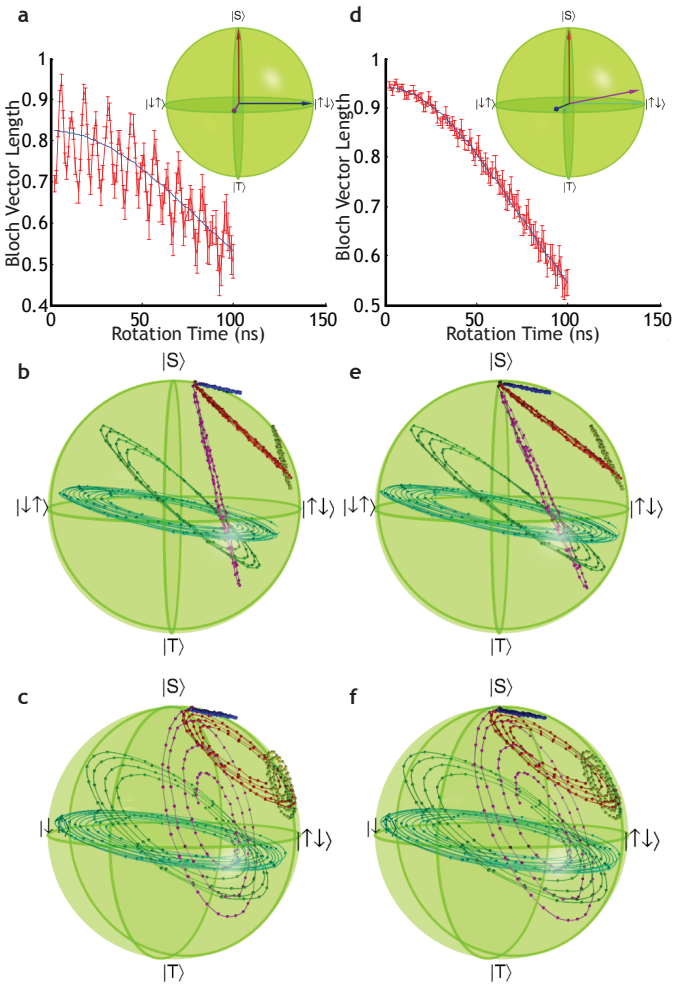

Figure 4.4: Calibrated vs uncalibrated state tomography: a, Data taken to calibrate the tomography shows ripples in the length of the Bloch vector if we assume that the tomography projects the quantum state on to Cartesian axes (inset). **b-c,** The paths around the Bloch sphere for the different evolutions that are used for tomography calibration. If the tomography is assumed to project on to the Cartesian axes there are points that lay outside the Bloch sphere, and the pure $|S\rangle$ states are not at the north pole, which is indicative of flawed state tomography. **d,** The ripples in the length of the Bloch vector are diminished (compared to panel a) if the axes deduced from state tomography (inset) are used. **e-f,** The paths around the Bloch sphere for the different evolutions that are used for state tomography. When the correct axes are used, all the points lie inside the Bloch sphere and the pure $|S\rangle$ are at the north pole.

There's a basic satisfaction in seeing that we are able to make something a little more coherent by the end of the day.

Alainn de Botton

5

Charge noise spectroscopy using coherent exchange oscillations in a singlet-triplet qubit

Two level systems that can be reliably controlled and measured hold promise in both metrology [10, 39] and as qubits for quantum information science (QIS) [18]. When prepared in a superposition of two states and allowed to evolve freely, the state of the system precesses with a frequency

this chapter is adapted from reference [17]

proportional to the splitting between the states. In QIS, this precession forms the basis for universal control of the qubit, and in metrology the frequency of the precession provides a sensitive measurement of the splitting. However, on a timescale of the coherence time, T_2, the qubit loses its quantum information due to interactions with its noisy environment, causing qubit oscillations to decay and setting a limit on the fidelity of quantum control and the precision of qubit-based measurements. Understanding how the qubit couples to its environment and the dynamics of the noise in the environment are therefore key to effective QIS experiments and metrology. Here we show measurements of the level splitting and dephasing due to voltage noise of a GaAs singlet-triplet qubit [29] during exchange oscillations. Using free evolution and Hahn echo [28] experiments we probe the low frequency and high frequency environmental fluctuations, respectively. The measured fluctuations at high frequencies are small, allowing the qubit to be used as a charge sensor with a sensitivity of $2 \times 10^{-8} e/\sqrt{\text{Hz}}$, two orders of magnitude better than the quantum limit for an RF single electron transistor (RF-SET) [16]. We find that the dephasing is due to non-Markovian voltage fluctuations in both regimes and exhibits an unexpected temperature dependence. Based on these measurements we provide recommendations for improving T_2 in future experiments, allowing for higher fidelity operations and improved charge sensitivity.

The singlet-triplet (S-T_0) qubit studied in this work is formed by two gate-defined lateral quantum dots (QD) in a GaAs/AlGaAs heterostructure (Figure 5.1a). The QDs are depleted until there is exactly one electron left in each, so that the system occupies the so-called $(1, 1)$ charge configuration. Here (n_L, n_R) describes a double QD with n_L electrons in the left dot and n_R electrons in the right dot. This two-electron system has four possible spin states: $|S\rangle$, $|T_+\rangle$, $|T_0\rangle$, and $|T_-\rangle$. The $|S\rangle$,$|T_0\rangle$ subspace is used as the logical subspace for this qubit because it is insensitive to homogeneous magnetic field fluctuations and is manipulable using only pulsed DC electric fields [49, 61, 37]. The relevant low-lying energy levels of this qubit are shown in Figure 5.1c. Two distinct rotations are possible in these devices: rotations around the x-axis of the Bloch sphere driven by difference in

magnetic field between the QDs, ΔB_z(provided in this experiment by feedback-stabilized hyperfine interactions[6]), and rotations around the z-axis driven by the exchange interaction, J (Figure 5.1b) [23]. A $|S\rangle$ can be prepared quickly with high fidelity by exchanging an electron with the QD leads, and the projection of the state of the qubit along the z-axis can be measured using RF reflectometery with an adjacent sensing QD (green arrow in Figure 5.1a)[53, 4].

Previous work on S-T_0 qubits focused almost entirely on x (ΔB_Z) rotations, which are dephased by fluctuations in the nuclear bath [7, 3, 42]. In this work, we focus on the exchange interaction, which creates a splitting, J, between the $|S\rangle$ and $|T_0\rangle$ states once the $(1,1)$ and $(0,2)|S\rangle$ states of the double QD are brought near resonance (Figure 5.1c). The value of J depends on the energy detuning, ϵ, between the QDs. The exchange interaction drives single qubit rotations in S-T_0 [49] and exchange-only [35, 24] qubits and is the foundation for two-qubit operations in single spin, S-T_0, and exchange-only qubits ([58, 66, 38, 8, 47]). Exchange oscillations are dephased by fluctuations in J (Figure 5.1c) driven, for example, by ϵ (voltage) fluctuations between the dots with a tunable sensitivity proportional to $dJ/d\epsilon$ (Figure 5.1d) [14]. We will show that this controllable sensitivity is a useful experimental tool for probing the noise bath dynamics. Previous studies have shown the decay of exchange oscillations within a few π rotations [49, 40], but a detailed study of the nature of the noise bath giving rise to this decay is still lacking. In this work, using nuclear feedback to control x-rotations, we systematically explore the low frequency noise portion of the voltage noise bath and its temperature dependence, as well as introduce a new Hahn-echo based measurement of the high frequency components of the voltage noise and its temperature dependence.

5.1 Free Induction Decay

The simplest probe of J and its fluctuations is a free induction decay (FID) experiment, in which the qubit is allowed to freely precess for a time t under the influence of the exchange splitting. For

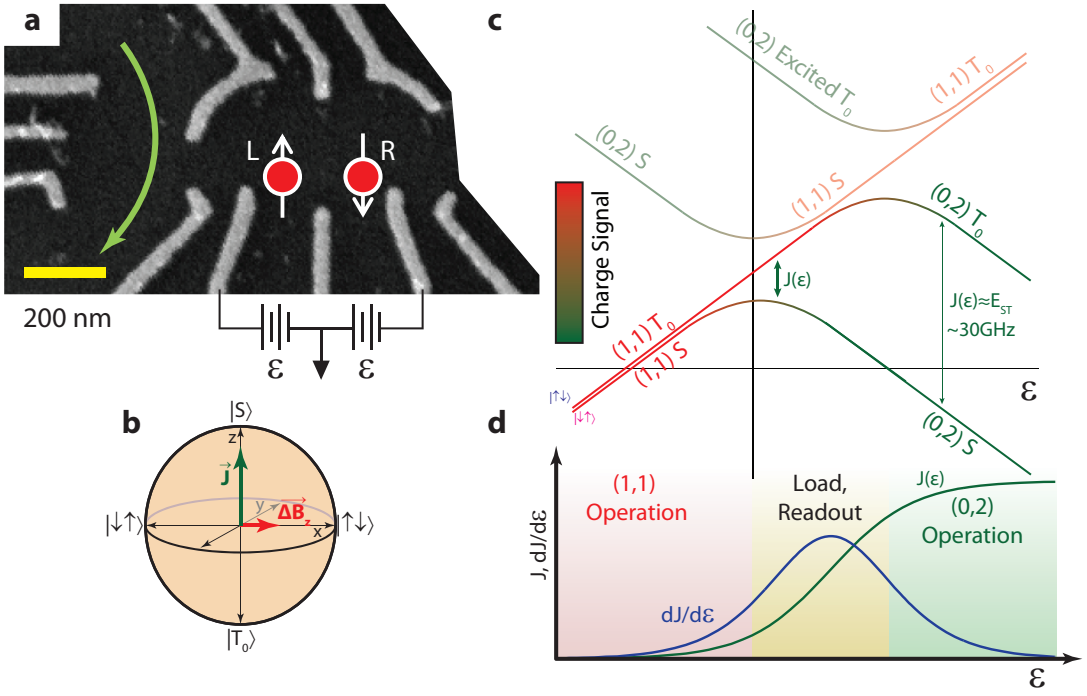

Figure 5.1: **a**, The device used in these measurements is a gate-defined S-T_0 qubit with an integrated RF sensing dot. The detuning ϵ is the voltage applied to the dedicated high-frequency control leads pictured. **b**, The Bloch sphere that describes the logical subspace of this device features two rotation axes (J and ΔB_Z) both controlled with DC voltage pulses. **c**, An energy diagram of the relevant low-lying states as a function of ϵ. States outside of the logical subspace of the qubit are grayed out. **d**, $J(\epsilon)$ and $dJ/d\epsilon$ in three regions; the $(1,1)$ region where J and $dJ/d\epsilon$ are both small and S-T_0 qubits are typically operated, the transitional region where J and $dJ/d\epsilon$ are both large where the qubit is loaded and measured, and the $(0,2)$ region where J is large but $dJ/d\epsilon$ is small and large quality oscillations are possible.

FID measurements, we use a $\pi/2$ pulse around the x-axis to prepare and readout the state of the qubit along the y-axis (Figure 5.2a, Fig. Figure 5.5). Figure 5.2b shows qubit oscillations as a function of t for many different values of ϵ. By measuring the period of these oscillations with t we extract $J(\epsilon)$, and we calculate $dJ/d\epsilon$ by fitting $J(\epsilon)$ to a smooth function and differentiating (Figure 5.2c). For negative ϵ (small J), we empirically find across many devices and tunings that J is well described by $J(\epsilon) \simeq J_0 + J_1 exp(-\epsilon/\epsilon_0)$.

The oscillations in these FID experiments decay due to voltage noise from DC up to a frequency of approximately $1/t$. As the relaxation time, T_1 is in excess of 100μs in this regime, T_1 decay is not an important source of decoherence (Figure 5.8). The shape of the decay envelope and the scaling of coherence time with $dJ/d\epsilon$ (which effectively changes the magnitude of the noise) reveal information about the underlying noise spectrum. White (Markovian) noise, for example, results in an exponential decay of e^{-t/T_2^*} where $T_2^* \propto (dJ/d\epsilon)^{-2}$ is the inhomogeneously broadened coherence time [15]. However, we find that the decay is Gaussian (Figure 5.2d) and that T_2^* (black line in Figure 5.2e) is proportional to $(dJ/d\epsilon)^{-1}$ (red solid line in Figure 5.2e) across two orders of magnitude of T_2^*. Both of these findings can be explained by quasistatic noise, which is low frequency compared to $1/T_2^*$. In such a case, one expects an amplitude decay of the form $\exp\left[-(t/T_2^*)^2\right]$, where $T_2^* = \frac{\sqrt{2}2\pi}{(dJ/d\epsilon)\epsilon_{RMS}}$ and ϵ_{RMS} is the root-mean-squared fluctuation in ϵ (Eq. 5.3). From the ratio of T_2^* to $(dJ/d\epsilon)^{-1}$, we calculate $\epsilon_{RMS} = 8\mu$V in our device. At very negative ϵ, J becomes smaller than ΔB_z, and nuclear noise limits T_2^* to approximately 90ns, which is consistent with previous work [6]. We confirm that this effect explains deviations of T_2^* from $(dJ/d\epsilon)^{-1}$ by using a model that includes the independently measured $T^*_{2,nuclear}$ and ΔB_z (Eq. 5.1) and observe that it agrees well with measured T_2^* at large negative ϵ (dashed red line in Figure 5.2e).

Since we observe J to be approximately an exponential function of ϵ, $(dJ/d\epsilon \sim J)$, we expect and observe the quality (number of coherent oscillations) of these FID oscillations, $Q \equiv JT_2^*/2\pi \sim J(dJ/d\epsilon)^{-1}$, to be approximately constant regardless of ϵ. However, when ϵ is made

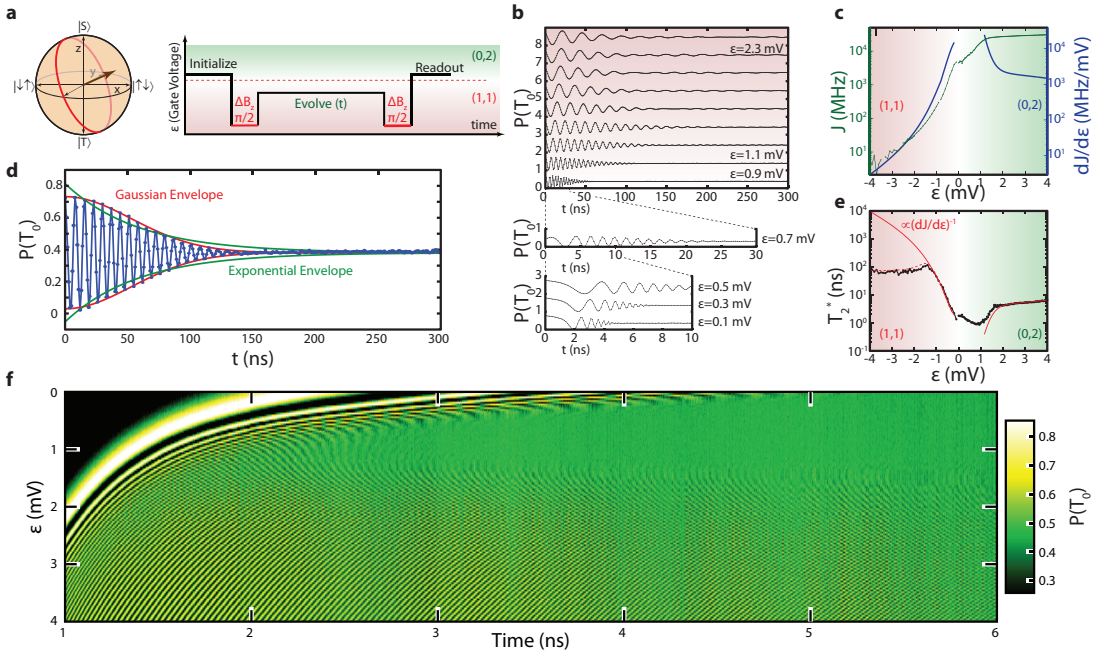

Figure 5.2: **a**, The pulse sequence used to measure exchange oscillations uses a stabilized nuclear gradient to prepare and readout the qubit and gives good contrast over a wide range of J. **b**, Exchange oscillations measured over a variety of detunings ϵ and timescales consistently show larger T_2^* as $dJ/d\epsilon$ shrinks until dephasing due to nuclear fluctuations sets in at very negative ϵ. **c**, Extracted values of J and $dJ/d\epsilon$ as a function of ϵ. **d**, The decay curve of FID exchange oscillations shows Gaussian decay. **e**, Extracted values of T_2^* and $dJ/d\epsilon$ as a function of ϵ. T_2^* is proportional to $(dJ/d\epsilon)^{-1}$, indicating that voltage noise is the cause of dephasing of charge oscillations. **f**, Charge oscillations measured in $(0, 2)$. This figure portrays the three basic regions we can operate our device in: a region of low frequency oscillations and small $dJ/d\epsilon$, a region of large frequency oscillations and large $dJ/d\epsilon$, and a region where oscillations are fast but $dJ/d\epsilon$ is comparatively small.

very positive and J is large, an avoided crossing occurs between the $(1,1)|T_0\rangle$ and the $(0,2)|T_0\rangle$ state, making the $(0,2)|S\rangle$ and $(0,2)|T_0\rangle$ states electrostatically virtually identical. Here, as ϵ is increased, J increases but $dJ/d\epsilon$ decreases(Figure 5.1d), allowing us to probe high quality exchange rotations and test our charge noise model in a regime that has never before been explored.

5.2 Exchange oscillations in (0,2)

Using a modified pulse sequence that changes the clock frequency of our waveform generators to achieve picosecond timing resolution (Figure 5.5)), we measure exchange oscillations in $(0,2)$ as a function of ϵ and time (Figure 5.2e) and we extract both J (Figure 5.2c) and T_2^* (Figure 5.2d) as a function of ϵ. Indeed, the predicted behavior is observed: for moderate ϵ we see fast oscillations that decay after a few ns, and for the largest ϵ we see even faster oscillations that decay slowly. Here, too, we observe that $T_2^* \propto (\frac{dJ}{d\epsilon})^{-1}$ (Figure 5.2d), which indicates that FID oscillations in $(0,2)$ are also primarily dephased by low frequency voltage noise. We note, however, that we extract a different constant of proportionality between T_2^* and $(dJ/d\epsilon)^{-1}$ for $(1,1)$ and $(0,2)$. This is expected given that the charge distributions associated with the qubit states are very different in these two regimes and thus have different sensitivities to applied electric fields. We note that in the regions of largest $dJ/d\epsilon$ (near $\epsilon = 0$), T_2^* is shorter than the rise time of our signal generator and we systematically underestimate J and overestimate T_2^* (Figure 5.5).

5.3 Exchange Echo

The above measurements indicate that the dephasing during FID experiments in both $(1,1)$ and $(0,2)$ arises overwhelmingly due to low frequency (non-Markovian) noise, and the observed linear dependence of T_2^* on $(dJ/d\epsilon)^{-1}$ strongly suggests that ϵ noise is indeed responsible for the observed dephasing, as these data rule out dephasing from other mechanisms in most realistic sit-

uations (see 5.5.5). In the presence of such low frequency noise, the addition of a π-pulse half-way through the free evolution can partially decouple the qubit from its noisy environment. Such a "Hahn-echo" [28] sequence prolongs coherence, which is useful for complex quantum operations [58], sensitive detection [41], and probing higher-frequency portions of the voltage noise bath. Rather than being sensitive to noise from DC to $1/\tau$ where τ is the total evolution time, these echo sequences have a noise sensitivity peaked at $f \approx 1/\tau$ and a reduced sensitivity at lower frequencies.

In our echo measurements, we select a fixed ϵ inside $(1, 1)$ for the free evolutions, and we sweep the length of the evolution following the π-pulse time by small increments δt to reveal an echo envelope (Figure 5.3a-b). The maximum amplitude of this observed envelope reveals the extent to which the state has dephased during the echo process, while the Gaussian shape and width of the envelope arise from an effective single-qubit rotation for a time δt, and thus reflect the same T_2^* and low frequency noise measured in FID experiments. We note that this exchange echo is distinct from the echo measurements previously performed in singlet-triplet qubits[7, 3, 42] in that we use ΔB_z rotations to echo away voltage noise, rather than J rotations to echo away noise in the nuclear bath.

The use of Hahn echo dramatically improves coherence times, with T_2^{echo} (the τ at which the observed echo amplitude has decayed by $1/e$) as large as 9 μs, corresponding to qualities ($Q \equiv T_2^{echo} J/2\pi$) larger than 600 (Figure 5.3c). If at high frequencies (50kHz-1MHz) the voltage noise were white (Markovian), we would observe exponential decay of the echo amplitude with τ. However, we find that the decay of the echo signal is non-exponential (Figure 5.3d), indicating that even in this relatively high-frequency band being probed by this measurement, the noise bath is not white.

A simple noise model that can account for this decay includes a mixture of white and $1/f$ noise, $S_\epsilon(f) = A + B/f$, which leads to an echo amplitude decay $exp(-\tau/C_0 - \tau^2/C_1)$ [15], where $C_{0,1}$ are functions of the noise power. Since $C_{0,1}$ are both proportional to $(dJ/d\epsilon)^{-2}$, we expect the ratio $C_0/C_1 \propto A/B$ to be independent of $dJ/d\epsilon$. While this decay accurately describes the decay

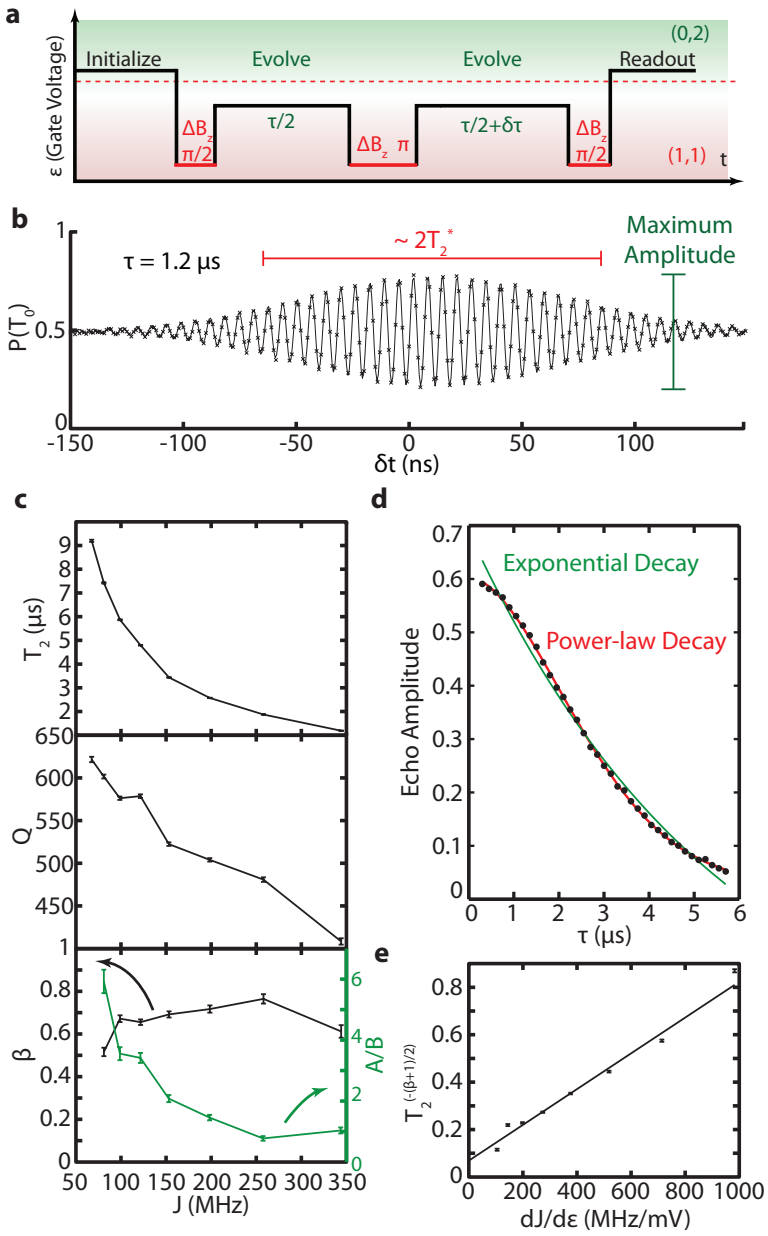

Figure 5.3: **a**, The pulse sequence used to measure exchange echo rotations. **b**, A typical echo signal. The overall shape of the envelope reflects T_2^*, while the amplitude of the envelope as a function of τ (not pictured) reflects T_2^{echo}. **c**, T_2^{echo} and $Q \equiv JT_2^{echo}/2\pi$ as a function of J. A comparison of the two noise models: power law and a mixture of white and $1/f$ noise. Noise with a power law spectrum fits over a wide range of frequencies (constant β), but the relative contributions of white and $1/f$ noise change as a function of ϵ. **d**, A typical echo decay is non-exponential but is well fit by $\exp(-(\tau/T_2^{echo})^{\beta+1})$. **e**, T_2^{echo} varies with $dJ/d\epsilon$ in a fashion consistent with dephasing due to power law voltage fluctuations.

for a single value of ϵ, as we change ϵ (and therefore $dJ/d\epsilon$), we find that the ratio A/B changes, indicating that this model is inconsistent with our data because the relative contributions of white and $1/f$ noise change (Figure 5.3c). Alternatively, we consider a power law noise model $S_\epsilon(f) = \frac{S_0}{f^\beta}$, which leads to an echo amplitude decay $exp(-(\tau/T_2^{echo})^{\beta+1})$. With this model we expect β to be independent of $dJ/d\epsilon$, and we indeed observe $\beta \approx 0.7$ for all values of ϵ (Figure 5.3c), indicating that this model can adequately describe our observed noise from approximately 50 kHz to 1 MHz. We further confirm that the observed dephasing is consistent with voltage noise by checking T_2^{echo} has the expected dependence on $dJ/d\epsilon$, namely, $T_2^{echo} \propto (dJ/d\epsilon)^{\frac{-2}{\beta+1}}$ (Figure 5.3e). From the scale factor, we deduce that the noise is well approximated by $S_\epsilon(f) = 8 \times 10^{-16} \frac{\mathrm{V}^2}{\mathrm{Hz}} \left(\frac{1\mathrm{Hz}}{f}\right)^{0.7}$ from approximately 50 kHz to 1 MHz, corresponding to ϵ noise of 0.2 nV/$\sqrt{\mathrm{Hz}}$ at 1 MHz. We note that this noise exceeds that accounted for by known sources of noise present in the experiment, including instrumental noise on the device gates and Johnson noise of the wiring. The RMS noise deduced from our FID measurements exceeds that expected from this power-law noise; there is excess noise at very low frequencies in the device.

5.4 Temperature Dependence

Thus far, we have explored the voltage noise bath at the base temperature of our dilution refrigerator ($\mathcal{T}\approx$50 mK). We gain additional insight into the properties of the voltage noise by studying its temperature dependence. For the nuclear bath, attainable temperatures in dilution refrigerators are much larger than the nuclear Zeeman splitting and no temperature dependence is expected. This is confirmed by measuring $T^*_{2,nuclear}$, the dephasing time for FID rotations in the stabilized nuclear gradient field (at J=0) as a function of $\mathcal{T}$(Figure 5.4a).

By contrast, T_2^* in $(1,1)$ and $(0,2)$ show unexpected temperature dependences (Figure 5.4b). These have the same scaled temperature dependence: $T^*_{2,(1,1)}(\mathcal{T}) \propto T^*_{2,(0,2)}(\mathcal{T})$, suggesting the

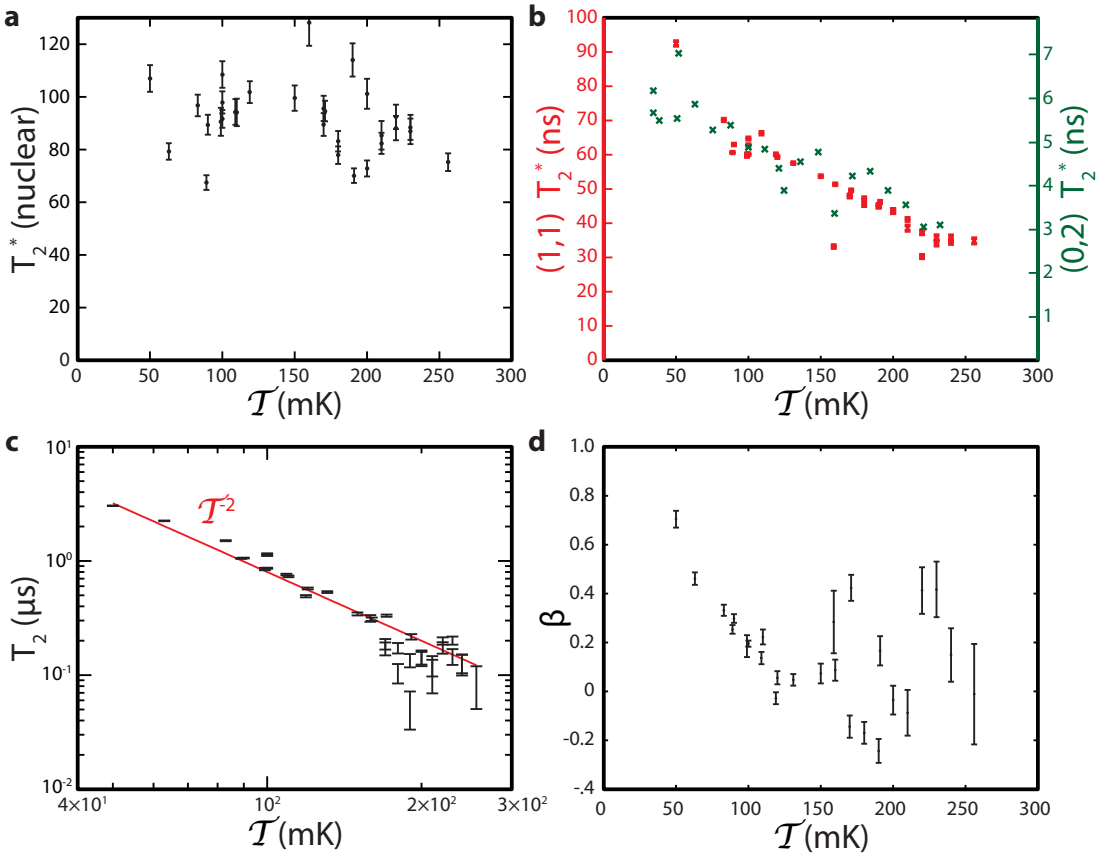

Figure 5.4: **a**, $T^*_{2,nuclear}$ does not depend significantly on $\mathcal{T}$. **b**, T_2^* in $(1,1)$ (red squares) and $(0,2)$ (green crosses) has the same, weak, scaled dependence on $\mathcal{T}$. **c**, T_2^{echo} shows a strong temperature dependence near $\mathcal{T}^2$ (red line) across over an order of magnitude in coherence times. **e**, As $\mathcal{T}$ is increased, β approaches zero indicating the noise leading to decoherence becomes nearly white. Uncertainties in β are larger at higher temperatures due to the fast decay of the echo.

loss of coherence is due to the same mechanism, presumably increased voltage noise, in both instances. In both cases T_2^* is roughly linear with $\mathcal{T}$, indicating that only small gains are likely to be made in the quality of FID-based rotations by reducing $\mathcal{T}$. By comparison, T_2^{echo} shows a strong dependence of $T_2^{echo} \propto \mathcal{T}^{-2}$ (Figure 5.4c). As $\mathcal{T}$ is increased, the observed noise becomes increasingly white (frequency independent) (Figure 5.4d), though measurements of β become inaccurate at large $\mathcal{T}$ where T_2^{echo} is small. We note that the underlying mechanism of this temperature dependence is currently unknown, however the dependence of the observed dephasing on temperature strongly suggests that the noise originates within the device rather than the experimental apparatus. Lower temperatures carry a double benefit for echo coherence; the noise becomes both smaller and more non-Markovian, thereby increasing coherence times and extending the potential for multi-pulse dynamical decoupling sequences to mitigate the effects of the noise [9, 64]. This trend shows no indication of saturating at low temperatures; it appears likely much longer coherence times are attainable by reducing temperatures with more effective refrigeration (Figure 5.7).

Operating at base temperature and using the Hahn echo sequence described above, we observe a voltage sensitivity of 0.2 nV/$\sqrt{\text{Hz}}$ at 1 MHz, which suggests that the qubit can be used as a sensitive electrometer at this frequency. In order to compare to other electrometers, we convert our voltage sensitivity into a charge sensitivity of 2×10^{-8}e/$\sqrt{\text{Hz}}$ by dividing by the Coulomb blockade peak spacing, 10mV. This value is nearly two orders of magnitude better than the theoretical limits for RF-SETs[16], and is limited only by T_2^{echo}, which the data suggest can be improved.

Using both FID and echo measurements, we have characterized the exchange interaction and presented experimental evidence that exchange rotations in S-T_0 qubits dephase due to voltage noise. These measurements reveal that the voltage noise bath that couples to the qubit is non-Markovian and establish baseline noise levels for S-T_0 qubits. We suggest that further improvements in operation fidelity and charge sensitivity are possible by reducing $\mathcal{T}$, using more complex pulse sequences such as CPMG [9] and UDD [64], and performing operations at larger J and $dJ/d\epsilon$ to move to

a higher frequency portion of the noise spectrum with potentially lower noise. In particular, because two-qubit operations in S-T_0 qubits rely heavily on exchange echo [58], our data show a path forward for increasing gate fidelities in these devices and generating higher quality entangled states. Lastly, the metrological capabilities of the S-T_0 qubit may be further improved by harnessing the power of entanglement and measuring simultaneously with many qubits[25].

5.5 Supplementary Information

5.5.1 Measuring $J(\epsilon)$

Previously [49, 40], exchange oscillations were measured inside of (1,1) by preparing a $|S\rangle$, adiabatically turning J off to prepare a state on the x-axis of the Bloch sphere and then turning J on again to allow the qubit state to rotate around the z-axis. Finally, to map the x-axis of the Bloch sphere back to the z-axis for readout J is turned off suddenly and then adiabatically turned back on again (Figure 5.5a). This allows measurements without a stabilized nuclear gradient. For $J \gg \Delta B_Z$, J drives rotations around the z-axis leading to oscillations with large visibility, but as J becomes comparable to $\Delta B_Z/2\pi$, ~30 MHz in this work, the axis of rotation tilts towards the x-axis and the visibility of exchange oscillations becomes small (Bloch sphere in Figure 5.5a). In order to measure the value of J when it is small, we use a modified pulse sequence where we load the qubit in $|S\rangle$, perform a $\pi/2-$pulse around the x-axis to prepare a state on the y-axis, then turn on J for a time t, and finally perform a $\pi/2-$pulse to project onto $|S\rangle$ (Figure 5.5b). We use $\pi/2-$pulses around the x-axis because it guarantees good contrast by preparing the qubit in a state that is perpendicular to the axis of rotation for all values of J. We note that this sequence requires a stabilized hyperfine gradient. The observed oscillation is driven by a combination of J and ΔB_Z In order to extract only the component of the splitting due to the exchange interaction, we measure the magnetic field gradient ΔB_Z with each measurement of the splitting and calculate $J(\epsilon) = \sqrt{J_{tot}^2 - (\Delta B_Z)^2}$ where

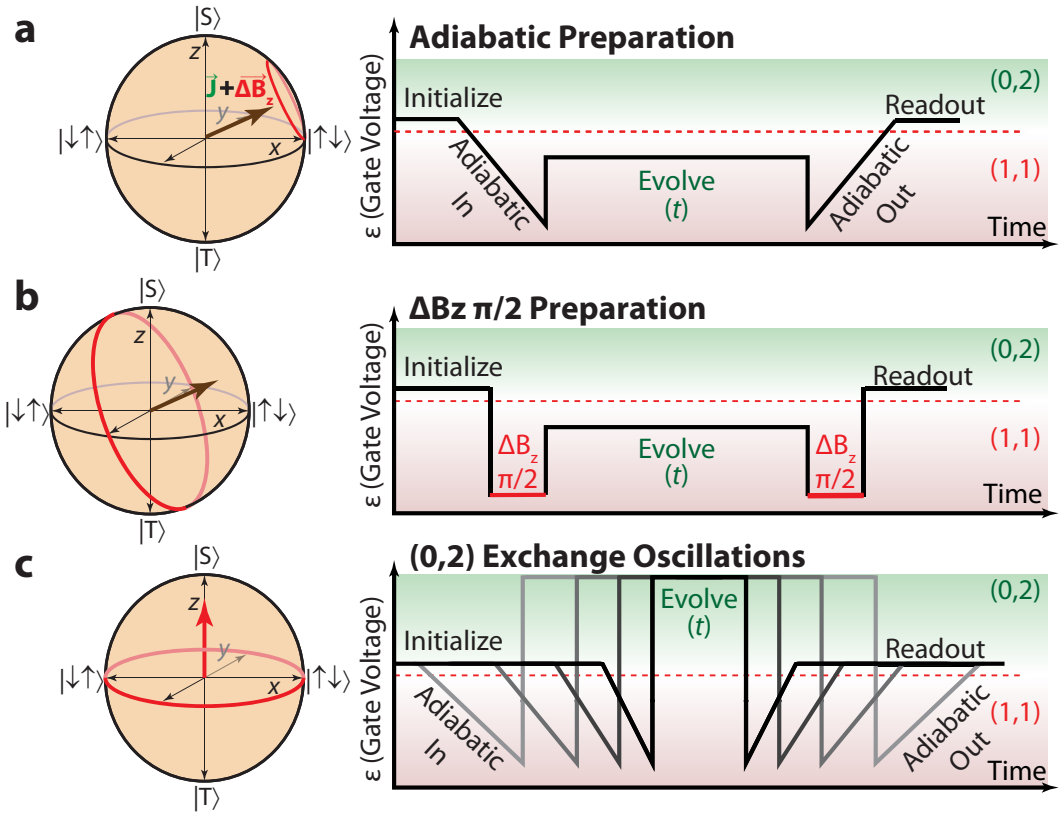

Figure 5.5: a, The pulse sequence traditionally used to measure exchange oscillations in $(1, 1)$ employs adiabatic preparation and readout but suffers reduced visibility when J becomes comparable to ΔB_Z. Red circles show the path of the state vector on the Bloch sphere during the exchange rotation for each sequence. **b**, We use a feedback-controlled nuclear gradient to prepare and readout on the y-axis, which provides good visibility for all values of J. **c**, The pulse sequence used to measure exchange oscillations in $(0, 2)$ uses adiabatic preparation and readout and "stretches" the pulse by changing the waveform generator clock rate.

J_{tot} is the observed total frequency.

With an AWG clock frequency of 12 GHz, we can directly measure J_{tot} up to a few GHz. To measure faster oscillations we change the clock rate of the waveform generator, which stretches the pulse, changing the evolution time in small increments. We use adiabatic preparation and readout instead of $\pi/2-$pulses around the x-axis because they do not change when the pulse is stretched by changing the waveform generator clock and do not require a stabilized nuclear gradient (Figure 5.5c). Using this method, we are able to resolve exchange oscillations exceeding 30GHz. The axis error (see below) associated with using an adiabatic preparation is negligible when measuring such large values of exchange since the angle between the z-axis and axis of revolution $\theta \sim .01 - .001$ radians. We note that for these measurements we use Minicircuits SBLP-467+ low pass filters on the fast detuning gates to prevent Zener tunneling when crossing the $(0,2)$-$(1,1)$ charge transition.

For an exchange echo experiment, there are two ways in which axis errors lead to a reduction in visibility. As in FID, if an adiabatic preparation is used, the axis of rotation is not perpendicular to the prepared state, and we avoid this problem by using $\pi/2$-pulses around the x-axis to prepare and read out the qubit. The second axis error arises from the fact that the axis around which we apply the π-pulse for decoupling is not perpendicular to the axis of rotation. It is possible to mitigate this adverse effect by using a π rotation around the y-axis, although for simplicity in this work we do not. The expected visibility as a function of θ for different sequences is summarized in Table 5.1.

Pulse risetimes in the experimental setup cause the frequency of exchange oscillations to slowly chirp up to a steady value over several nanoseconds. This chirp causes us to underestimate $J(\epsilon)$ in the region where T_2^* is comparable to the time required for J to saturate. Consequently, we are unable to assign unbiased values for $J(\epsilon)$ or T_2^* in the regions of the largest $\frac{dJ}{d\epsilon}$, in this work roughly $-1\text{mV} < \epsilon < 0mV$. Unlike some previous works, ϵ=0 is the detuning at which we measure the state of the qubit and not the degeneracy point between $(0,2)$ and $(1,1)$.

Rotation Sequence	Ideal Visibility
FID, adiabatic prep	$\cos^2(\theta)$
FID, x-$\pi/2$-prep	1
Echo, adiabatic prep, x-π	$\cos^4(\theta)$
Echo, adiabatic prep, y-π	$\cos^2(\theta)$
Echo, x-$\pi/2$ prep, x-π	$\cos^2(\theta)$
Echo, x-$\pi/2$ prep, y-π	1

Table 5.1: The visibility loss associated with the different preparations and π-pulses.

5.5.2 Saturation of T_2^*

When $J(\epsilon)$ is small compared to ΔB_Z, T_2^* no longer scales with $dJ/d\epsilon$, but rather saturates at $T^*_{2,nuclear}$ which is limited by the quality of the nuclear feedback. Geometric considerations give

$$(T_2^*)^{-2} = \left(\frac{J}{J_{tot}}\right)^2 \left(T^*_{2,voltage}\right)^{-2} + \left(\frac{\Delta B_Z}{J_{tot}}\right)^2 \left(T^*_{2,nuclear}\right)^{-2} \tag{5.1}$$

where J is the exchange splitting, $J_{tot} = \sqrt{J^2 + \Delta B_Z^2}$ is the observed frequency, and $T^*_{2,voltage}$ and $T^*_{2,nuclear}$ are the independent coherence times for J and ΔB_Z noise, respectively. We use this form to plot the expected T_2^* and note that this fit has no free parameters since we measure J, ΔB_Z, and $T^*_{2,nuclear}$.

5.5.3 T_2^* and T_{meas}

Nominally, an FID measurement is sensitive to charge noise from DC to J. However, any given experiment only runs over a period of time T_{meas}, and this introduces and additional low frequency cutoff; noise with a frequency below $1/T_{meas}$ will appear as an offset to J and will not contribute

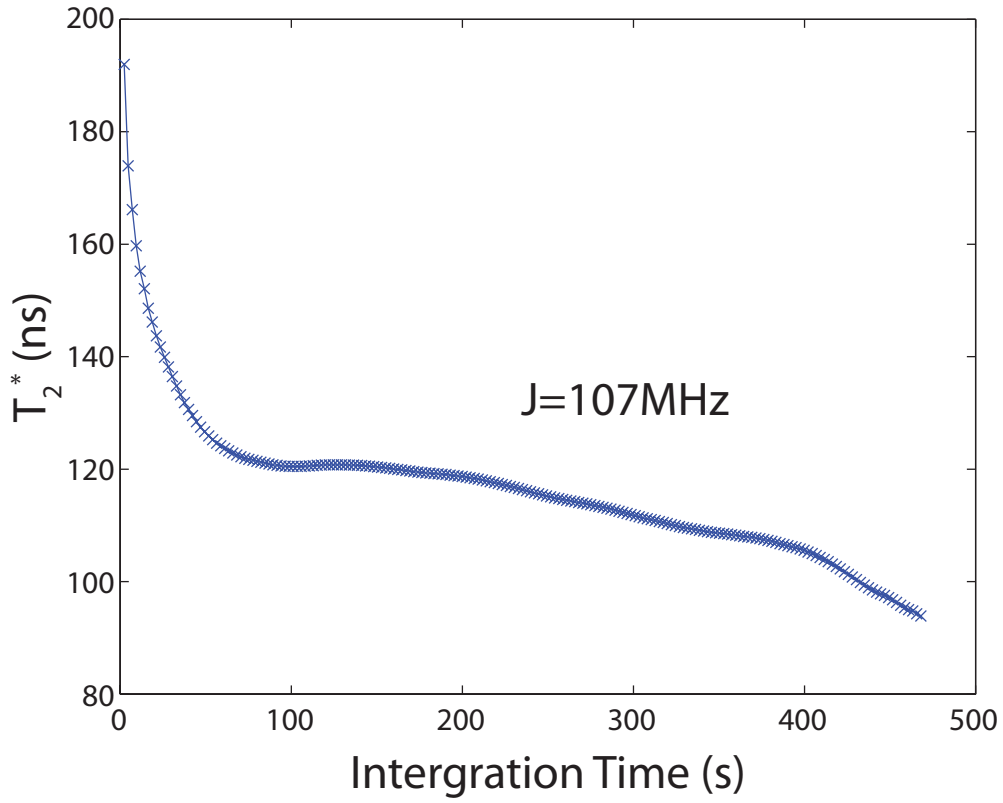

Figure 5.6: As we measure exchange oscillations for longer, increasingly low frequency noise contributes to dephasing and T_2^* decreases.

to dephasing. This is particularly relevant for $1/f$ noise, which diverges at DC. In general, as the measurement time increases, the low-frequency cutoff decreases and the observed T_2^* becomes smaller (Figure 5.6). This makes it essential to normalize for total measurement time in comparing results between different groups and experiments.

5.5.4 Fitting Echo Data

In a typical Hahn echo experiment, we measure the probability of finding a singlet as a function of total evolution time τ and change in length of the second exchange interval δt. In order to self-

consistently and reliably fit the data, we simultaneously fit all the echo data for a given value of ϵ. For the power-law noise model, we use the form

$$P_S(\tau, \delta t) = A_0 \cos\left[\omega(\tau)\delta t + \phi(\tau)\right] \times \exp\left[-(\tau/T_2)^{\beta+1} - ((\delta t - t_0(\tau))/T_2^*)^2\right]$$

Allowing ω, ϕ, and the echo center, t_0, to vary with τ allows us to fit in the presence of finite pulse rise times which gives small phase and frequency shifts as a function of τ. Fitting the overall envelope globally through A_0, β, T_2, and T_2^* across many values of τ greatly reduces the probability of encountering local minima during fitting.

In some instances if the $\pi/2$ and π pulses are not tuned perfectly, there may be some "un-echoed" signal at short total evolution times such that $T_2^* \gtrsim \tau$. In this case, we add an additional term to $P_S(\tau, t)$

$$P_S'(\tau, \delta t) = P_S(\tau, \delta t) + A_1 \exp\left[-(\tau/T_2^*)^2\right] \cos\left[\omega(\tau)\delta t + \theta(\tau)\right] \tag{5.2}$$

Here, the new τ-dependent fit paraemter $\theta(\tau)$ accounts for the phase shift of the unechoed signal due to pulse rise times[58], while A_1 accounts for the magnitude of the un-echoed signal.

5.5.5 Dephasing from Other Fluctuating Parameters

There is no reason, a priori, to assume that the dephasing present in FID oscillations is due only to fluctuations in ϵ. For example, in a simple avoided crossing model we expect J to depend on the tunnel coupling, T_C, between the QDs. However, the observed linear dependence of T_2^* on $(dJ/d\epsilon)^{-1}$ strongly suggests that fluctuations in ϵ are responsible for the observed dephasing. For T_C to dominate dephasing, we would need $dJ/dT_C \propto dJ/d\epsilon$ for all ϵ, a condition which is not true for the simple avoided crossing model and would generally seem unlikely. The same argument

can be repeated for any other parameter that couples to J.

5.5.6 Extracting Noise Power

As per Cywinski *et al.*[15], we calculate the amplitude of qubit oscillations by integrating the noise spectrum multiplied by an experiment-specific filter function over all frequencies. For example, for a Hahn echo experiment in the presence of power-law noise $S_\phi(\omega) = \frac{S}{\omega^\beta}$ we find

$$\left(T_2^{echo}\right)^{1+\beta} = \frac{\pi}{2^{-\beta}(-2+2^\beta)S\Gamma(-1-\beta)\sin(\pi\beta/2)} \tag{5.3}$$

We note that $S_\phi = \frac{1}{2}S_\epsilon(\frac{dJ}{d\epsilon})^2$, where S_ϵ is the ϵ noise spectrum for positive frequencies only. From this we find that $(T_2^{echo})^{-(\beta+1)/2} \propto \frac{dJ}{d\epsilon}$.

For FID experiments, which are dephased by low frequency noise, we assume the noise is quasistatic (much slower than $1/T_2^*$) and normally distributed with variance σ^2. For a given experiment we find the probability of measuring a singlet $P(S) = \int_{-\infty}^{\infty} d\omega e^{-(\omega-\omega_0)^2/2\sigma^2} e^{i\omega t}$, which can be integrated to obtain $P(S) = e^{i\omega_0 t}e^{-t^2\sigma^2/2}$. From here we find $T_2^{*2} = \frac{2}{\sigma^2} = \frac{2}{\epsilon_{RMS}^2(dJ/d\epsilon)^2}$.

Alternatively, we can consider dephasing from $1/f$ noise during FID experiments, which leads to nearly Gaussian decay. More precisely, for noise described by a noise power $S_\epsilon(\omega) = S_1/\omega$, the decay envelope of FID oscillations is given by $\exp[-\frac{S_1}{4\pi}\left(\frac{dJ}{d\epsilon}\right)^2 t^2 ln\frac{T_{meas}}{2\pi t}]$ where the total measurement time $T_{meas} \approx 5$ minutes provides a low frequeny cutoff to the noise sensitivity[15] (see above). For these measurements, relevant values of t for decay are of order 100 nanoseconds, so we neglect the weak t dependence of the logarithmic term and approximate it by 20. We fit the decay envelope to the form $\exp\left[-(t/T_2^*)^2\right]$, where $T_2^{*2} = \frac{4\pi}{S_1(dJ/d\epsilon)^2 ln\frac{T_{meas}}{2\pi t}}$ and extract $\sqrt{S_\epsilon} = 2\mu\mathrm{V}/\sqrt{f(\mathrm{Hz})}$. We note, however, that the observed dependence of T_2^* on T_{meas} (Figure 5.6) is not consistent with simple $1/f$ noise.

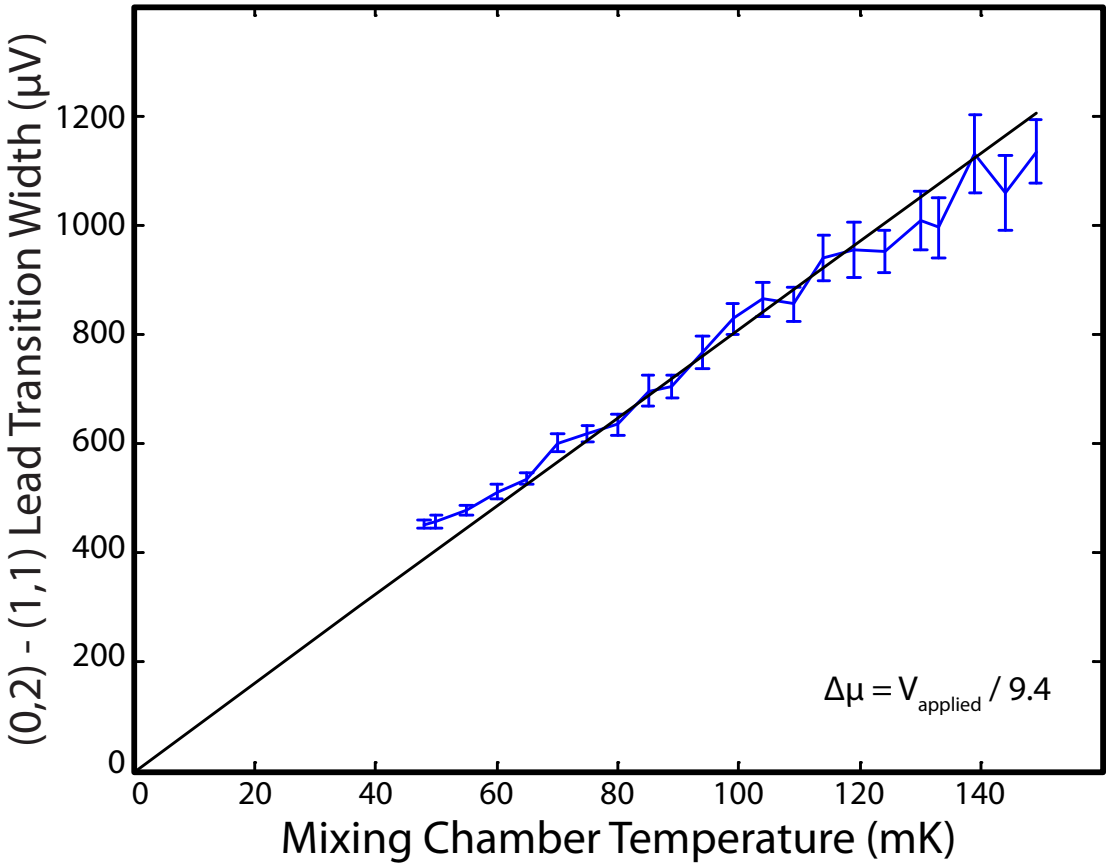

Figure 5.7: The $(0,1)$-$(0,2)$ charge transition is thermally broadened, allowing us to determine the electron temperature. The width is calibrated using the mixing chamber temperature at high temperatures. The black line is a best fit, forced through zero width at zero temperature. The "lever-arm" for the fast-RF gates is seen to be 9.4, allowing conversion from gate voltage noise to effective noise at the qubit if desired.

5.5.7 Temperature Dependence

All experiments presented are performed in a dilution refrigerator with a base temperature of approximately 50mK. All temperature dependence data are taken by setting the mixing chamber temperature of the dilution refrigerator and waiting sufficient time (30 min) for the system to equilbrate. We monitor the electron temperature as the mixing chamber temperature is changed by measuring the width of a charge transition of the double QD and assuming that the electronic temper-

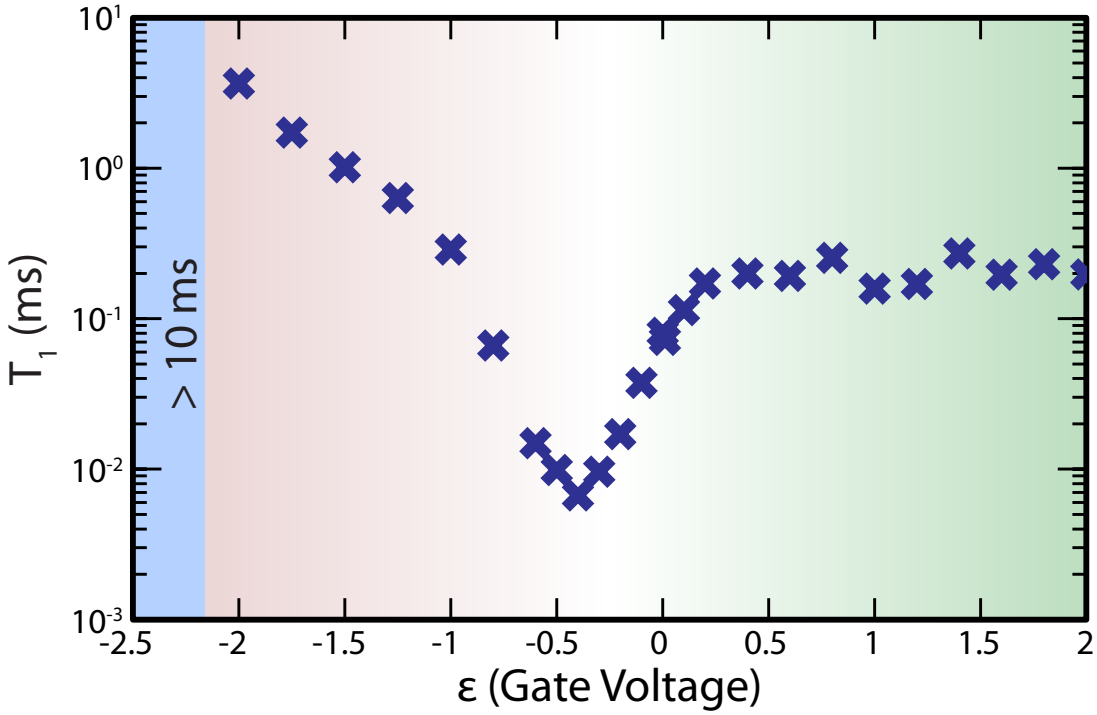

Figure 5.8: Although T_1 is a complex function of ϵ, it is much longer than T_2^* and T_2 everywhere.

ature and mixing chamber temperatures are the same at high temperatures (Figure 5.7). We find the electronic temperature and lattice temperature (assumed to be the same as that of the mixture) are similar over the range studied, and we are therefore unable to differentiate between noise sources that depend on electron temperature and those that depend on lattice temperature (for example, phonons).

5.5.8 Measuring T_1 at finite ϵ

We measure T_1 as a function of ϵ in two steps. First we prepare a $|S\rangle$, adiabatically ramp to ϵ and wait for a time t, then measure the probability of the resulting state being a singlet, $P_{S|S}$. We then prepare a $|T_0\rangle$ and repeat the process to measure $P_{S|T_0}$. We finally fit $P_{S|S} - P_{S|T_0}$ to an exponential function of t to extract T_1. As seen in Figure 5.8, T_1 is much longer than T_2. The nuclear

gradient ΔB_Z is stabilized to 30 MHz for this measurement.

Acknowledgments

This work is supported through the ARO, "Precision Quantum Control and Error-Suppressing Quantum Firmware for Robust Quantum Computing" and the IARPA "Multi-Qubit Coherent Operations (MQCO) Program". This work was partially supported by the US Army Research Office under Contract Number W911NF-11-1-0068. This work is sponsored by the United States Department of Defence. The views and conclusions contained in the document are those of the authors and should not be interpreted as representing the official policies, either expressly or implied, of the U. S. Government. This work was performed in part at the Center for Nanoscale Systems (CNS), a member of the National Nanotechnology Infrastructure Network (NNIN), which is supported by the National Science Foundation under NSF award no. ECS-0335765. CNS is part of Harvard University.

Learning never exhausts the mind.

Leonardo da Vinci

6

Suppressing qubit dephasing using real-time Hamiltonian estimation

Unwanted interaction between a quantum system and its fluctuating environment leads to decoherence and is the primary obstacle to establishing a scalable quantum information processing

this chapter is adapted from reference [59]

architecture. Strategies such as environmental and materials engineering, quantum error correction and dynamical decoupling can mitigate decoherence, but generally increase experimental complexity. Here we improve coherence in a qubit using real-time Hamiltonian parameter estimation. Using a rapidly converging Bayesian approach, we precisely measure the splitting in a singlet-triplet spin qubit faster than the surrounding nuclear bath fluctuates. We continuously adjust qubit control parameters based on this information, thereby improving the inhomogenously broadened coherence time (T_2^*) from tens of nanoseconds to above 2 μs. Because the technique demonstrated here is compatible with arbitrary qubit operations, it is a natural complement to quantum error correction and can be used to improve the performance of a wide variety of qubits in both metrological and quantum-information-processing applications.

6.1 Introduction

Hamiltonian parameter estimation is a rich field of active experimental and theoretical research that enables precise characterization and control of quantum systems [71]. For example, magnetometry schemes employing Hamiltonian learning have demonstrated dynamic range and sensitivities exceeding those of standard methods [68, 48]. Such applications focused on estimating parameters that are quasistatic on experimental timescales. However, the effectiveness of Hamiltonian learning also offers exciting prospects for estimating fluctuating parameters responsible for decoherence in quantum systems.

The quantum system that we study is a singlet-triplet (S-T_0) qubit [49, 40] which is formed by two gate-defined lateral quantum dots (QDs) in a GaAs/AlGaAs heterostructure (Figure 6.1a,Figure 6.5), similar to that of refs. [17, 58]. The qubit can be rapidly initialized in the singlet state $|S\rangle$ in $\approx$ 20 ns and read out with 98% fidelity in $\approx 1\ \mu s$ [4, 53] (Figure 6.6). Universal quantum control is provided by two distinct drives [23]: the exchange splitting, J, between $|S\rangle$ and $|T_0\rangle$, and the mag-

netic field gradient, ΔB_z, due to the hyperfine interaction with host Ga and As nuclei. The Bloch sphere representation for this qubit can be seen in Figure 6.1b. In this work, we focus on qubit evolution around ΔB_z (Figure 6.2a). Due to statistical fluctuations of the nuclei, ΔB_z varies randomly in time, and consequently oscillations around this field gradient decay in a time $T_2^* \approx 10$ ns [49]. A nuclear feedback scheme relying on dynamic nuclear polarization [6] can be employed to set the mean gradient, ($g^*\mu_B \Delta B_z/h \approx 60$ MHz in this work) as well as reduce the variance of the fluctuations. Here, $g^* \approx -0.44$ is the effective gyromagnetic ratio in GaAs, μ_B is the Bohr magneton and h is Planck's constant. In what follows, we adopt units where $g^*\mu_B/h = 1$. The nuclear feedback relies on the avoided crossing between the $|S\rangle$ and $|T_+\rangle$ states. When the electrons are brought adiabatically through this crossing, their total spin changes by $\Delta m_s = \pm 1$, which is accompanied by a nuclear spin flip in order to conserve angular momentum. With the use of this feedback, the coherence time improves to $T_2^* \approx 100$ ns [6] (Figure 6.2b), limited by the low nuclear pumping efficiency [23]. Crucially, the residual fluctuations are considerably slower than the timescale of qubit operations [7].

In this work we employ techniques from Hamiltonian estimation to prolong the coherence of a qubit by more than a factor of 30. Importantly, our estimation protocol, which is based on recent theoretical work [56], requires relatively few measurements (≈100) which we perform rapidly enough (total time $\approx 100\mu s$) to resolve the qubit splitting faster than its characteristic fluctuation time. We adopt a paradigm in which we separate experiments into "estimation" and "operation" segments, and we use information from the former to optimize control parameters for the latter in real-time. Our method dramatically prolongs coherence without using complex pulse sequences such as those required for non-identity dynamically decoupled operations [32].

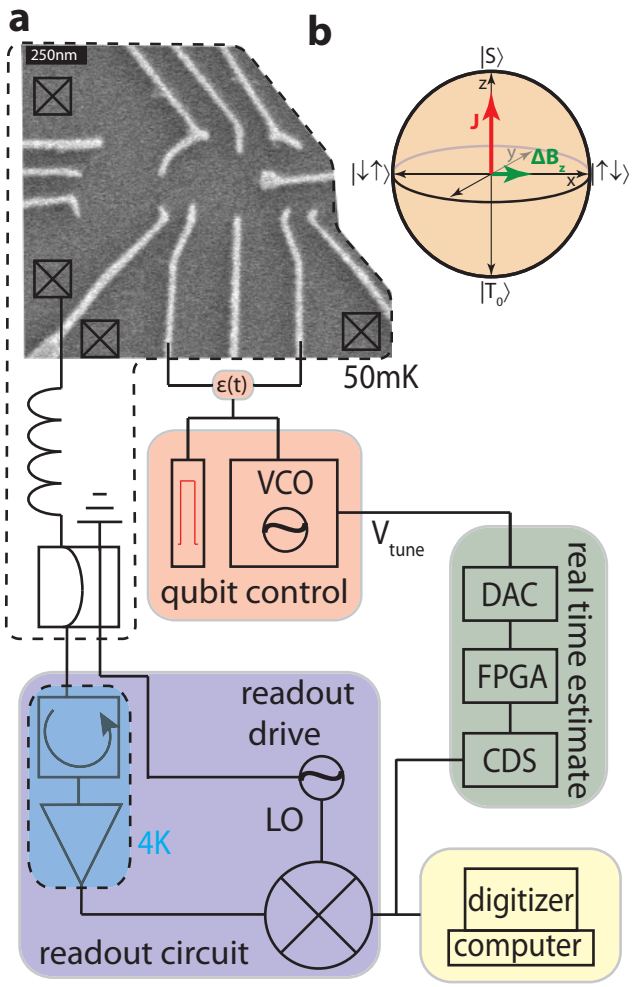

Figure 6.1: Experimental apparatus. **a,** A scanning electron microscope image of the double QD with a schematic of the apparatus used for adaptive qubit control. A floating metal gate protruding from the right can be seen which increases the capacitance between the qubit and an adjacent qubit (not pictured), which is left inactive for this work. The reflected readout drive signal is demodulated to DC, digitized by a correlated double sampler (CDS), and ΔB_z is estimated in real time by the field programmable gate array (FPGA). The FPGA updates the digital to analog converter (DAC) in order to keep the voltage controlled oscillator (VCO) resonant with the estimated value of ΔB_z. The VCO controls the voltage detuning, $\epsilon(t)$ between the QDs, which, in turn, modulates J at Ω_J. **b,** The Bloch sphere representation for the S-T_0 qubit showing the two axes of control, J and ΔB_z.

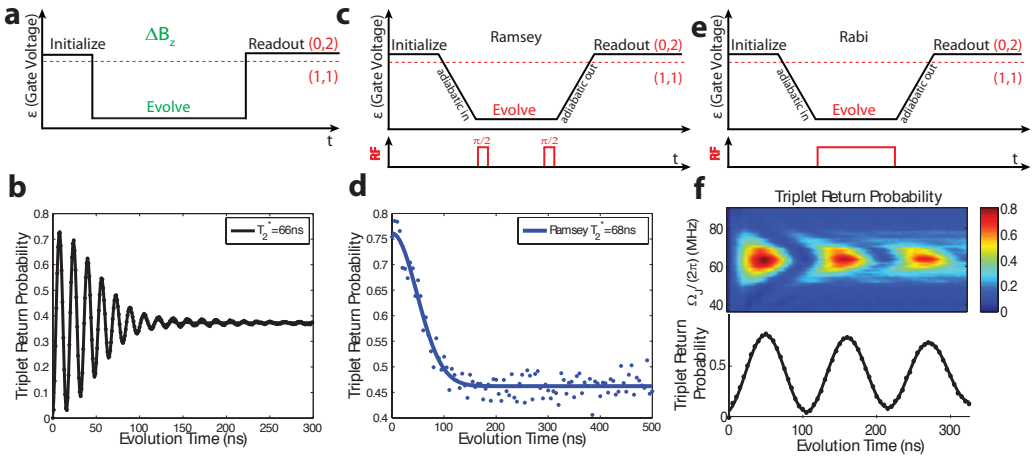

Figure 6.2: ΔB_z oscillations. **a,** The pulse sequence used to estimate ΔB_z. **b,** Using nuclear feedback, ΔB_z oscillations decay in a coherence time $T_2^* = 66$ns due to residual slow fluctuations in ΔB_z. **c,** The Ramsey sequence used to operate the S-T_0 qubit in the rotating frame. **d,** The Ramsey contrast (blue dots) decays in a characteristic time (solid line fit $T_2^* = 68$ns) similarly to the oscillations in **(c)** due to the same residual slow fluctuations in ΔB_z. **e,** The Rabi pulse sequence used to drive the qubit in the rotating frame. **f,** The rotating frame S-T_0 qubit exhibits the typical behavior when sweeping drive frequency and time (top). When driven on resonance (bottom), the qubit undergoes Rabi oscillations, demonstrating control in the rotating frame.

6.2 Results

6.2.1 Rotating frame $S - T_0$ qubit

To take advantage of the slow nuclear dynamics, we introduce a method that measures the fluctuations and manipulates the qubit based on precise knowledge but not precise control of the environment. We operate the qubit in the rotating frame of ΔB_z, where qubit rotations are driven by modulating J at the frequency $\frac{\Omega_J}{2\pi} = \Delta B_z$ Figure 6.5 [13, 33]. This is in contrast to traditional modes of opeation of the S-T_0 qubit, which rely on DC voltage pulses. To measure Rabi oscillations, the qubit is adiabatically prepared in the ground state of ΔB_z ($|\psi\rangle$=$|\uparrow\ \downarrow\rangle$), and an oscillating J is switched on (Figure 6.2e), causing the qubit to precess around J in the rotating frame. Additionally, we perform a Ramsey experiment (Figure 6.2c) to determine T_2^*, and as expected, we ob-

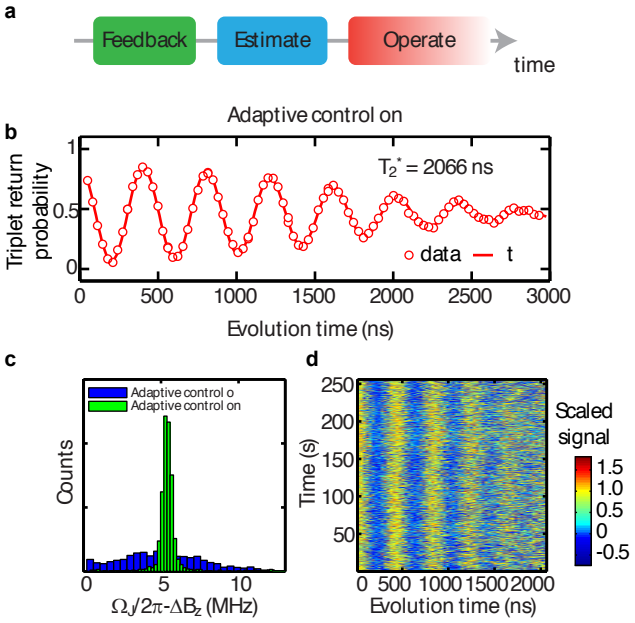

Figure 6.3: Adaptive control. **a,** For these measurements we first perform our standard nuclear feedback, then quickly estimate ΔB_z and update the qubit control, then operate the qubit at the correct driving frequency. **b,** Using adaptive control, we perform a Ramsey experiment (deliberately detuned to see oscillations) and obtain coherence times of $T_2^* = 2066$ ns. **c,** Histograms of measured Ramsey detunings with (green) and without (blue) adaptive control. For clarity, these data were taken with a different mean detuning than those in **(b)**. **d,** Raw data for 1024 consecutive Ramsey experiments with adaptive control lasting 250 s in total. A value of 1 corresponds to $|T_0\rangle$ and 0 corresponds to $|S\rangle$. Stabilized oscillations are clearly visible in the data, demonstrating the effect of adaptive control.

serve the same decay (Figure 6.2d) as Figure 6.2b. More precisely, the data in Figure 6.2d represent the average of 1024 experimental repetitions of the same qubit operation sequence immediately following nuclear feedback. The feedback cycle resets ΔB_z to its mean value (60MHz) with residual fluctuations of $(\sqrt{2}\pi T_2^*)^{-1} \approx 10$MHz between experimental repetitions. However, within a given experimental repetition, ΔB_z is approximately constant. Therefore we present an adaptive control scheme where, following nuclear feedback, we quickly estimate ΔB_z and tune $\frac{\Omega_J}{2\pi} = \Delta B_z$ in order to prolong qubit coherence (Figure 6.3a).

6.2.2 Bayesian Estimation

To estimate ΔB_z, we repeatedly perform a series of singleshot measurements after allowing the qubit to evolve around ΔB_z (using DC pulses) for some amount of time (Figure 6.2a). Rather than fixing this evolution time to be constant for all trials, we make use of recent theoretical results in Hamiltonian parameter estimation [56, 20, 33] and choose linearly increasing evolution times, $t_k = kt_{samp}$, where $k = 1, 2, \cdots, N$. We choose the sampling time t_{samp} such that the estimation bandwidth $\mathcal{B} = \frac{1}{2t_{samp}}$ is several times larger than the magnitude of the residual fluctuations in ΔB_z, roughly 10 MHz. With a Bayesian approach to estimate ΔB_z in real-time, the longer evolution times (large k) leverage the increased precision obtained from earlier measurements to provide improved sensitivity, allowing the estimate to outperform the standard limit associated with repeating measurements at a single evolution time. Denoting the outcome of the k^{th} measurement as m_k (either $|S\rangle$ or $|T_0\rangle$), we define $P(m_k|\Delta B_z)$ as the conditional probability for m_k given a value ΔB_z. We write

$$P(m_k|\Delta B_z) = \frac{1}{2}\left[1 + r_k\left(\alpha + \beta\cos\left(2\pi\Delta B_z t_k\right)\right)\right], \tag{6.1}$$

where r_k=1 (-1) for m_k=$|S\rangle$ ($|T_0\rangle$), and α =0.25 and β =0.67 are parameters determined by the measurement error and axis of rotation on the Bloch sphere (see 6.4.1). Since we assume that earlier measurement outcomes do not affect later ones (i.e. that there is no measurement back-action), we write the conditional probability for ΔB_z given the results of N measurements as:

$$\begin{aligned} P(\Delta B_z|m_N, m_{N-1}, ...m_1) &= P(\Delta B_z|m_{N-1}, ...m_1)\cdot P(\Delta B_z|m_N) &(6.2)\\ &= \prod_{k=1}^{N} P(\Delta B_z|m_k). &(6.3)\end{aligned}$$

Using Bayes' rule, i.e., $P(\Delta B_z|m_k) = P(m_k|\Delta B_z)P(\Delta B_z)/P(m_k)$, and eq. 6.13, we can rewrite eq. 6.3 as:

$$P(\Delta B_z|m_N, m_{N-1}, ...m_1) = P_0(\Delta B_z)\mathcal{N}\prod_{k=1}^{N}\left(1 + r_k\left(\alpha + \beta\cos\left(2\pi\Delta B_z t_k\right)\right)\right], \qquad (6.4)$$

where $\mathcal{N}$ is a normalization constant and $P_0(\Delta B_z)$ is a prior distribution to which the algorithm is empirically insensitive*, and which we take to be a constant over the estimation bandwidth †. After the last measurement, we find the value of ΔB_z that maximizes the posterior distribution $P(\Delta B_z|m_N, m_{N-1}, ...m_1)$.

6.2.3 Adaptive Control

We implement this algorithm in real-time on a field-programmable gate array (FPGA), computing $P(\Delta B_z)$ for 256 values of ΔB_zbetween 50 and 70MHz. With each measurement m_k, the read-out signal is digitized and passed to the FPGA, which computes $P(\Delta B_z)$ and updates an analog voltage that tunes the frequency of a voltage controlled oscillator (Figure 6.1a). Following the N^{th} sample, $\frac{\Omega_J}{2\pi}$ nearly matches ΔB_z, and since the nuclear dynamics are slow, the qubit can be operated with long coherence without any additional complexity. To quantify how well the FPGA estimate matches ΔB_z, we perform a Ramsey experiment (deliberately detuned to observe oscillations) with this real-time tracking of ΔB_z and find optimal performance for $N \approx 120$, with a maximum experimental repetition rate, limited by the FPGA, of 250kHz and a sampling time $t_{samp} = 12$ ns. Under these conditions, and making a new estimate after every 42 Ramsey experiments, we observe $T_2^* = 2066$ ns, a 30-fold increase in coherence (Figure 6.3b). We note that these data are taken with the same pulse sequence as those in Figure 6.2d. To further compare qubit operations with

*within reason, of course

†A Gaussian prior of width $\sim \frac{1}{T_2^*}$ yields similar results

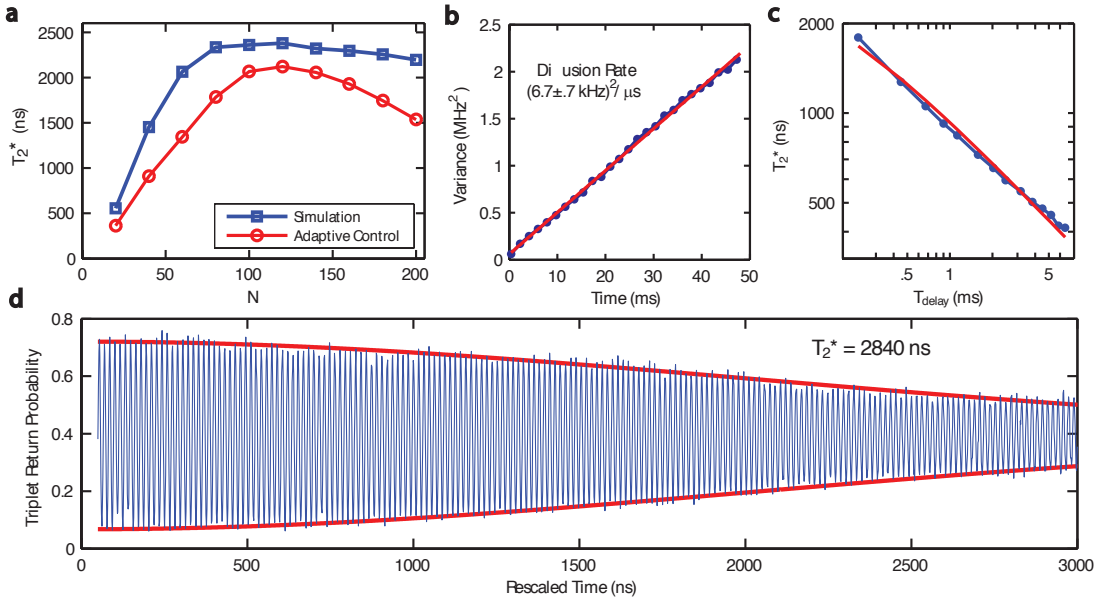

Figure 6.4: ΔB_z diffusion. **a,** The coherence time, T_2^* using the adaptive control and for a simulation show a peak, indicating that there is an optimal number of measurements to make when estimating ΔB_z. **b,** When many time traces of ΔB_z are considered, their variance grows linearly with time, indicating a diffusion process. **c,** The scaling of T_2^* as a function of T_{delay} for software scaled data is consistent with diffusion of ΔB_z. The red line is a fit to a diffusion model. **d,** The performance of the Bayesian estimate of ΔB_z can be estimated using software post processing, giving $T_2^* = 284\text{ns}$, which corresponds to a precision of $\sigma_{\Delta B_z} = 80\text{kHz}$.

and without this technique, we measure Ramsey fringes for $\approx 250\text{s}$ (Figure 6.3d), and histogram the observed Ramsey detunings. With adaptive control we observe a stark narrowing of the observed frequency distribution, consistent with this improved coherence (Figure 6.3c).

6.3 Discussion

Although the estimation scheme employed here is theoretically predicted to improve monotonically with N [56], we find that there is an optimum ($N \approx 120$), after which T_2^* slowly decreases with increasing N (Figure 6.4a). A possible explanation for this trend is fluctuation of the nuclear gradient during the estimation period. To investigate this, we obtain time records of ΔB_z using the Bayesian estimate and find that its variance increases linearly in time at the rate of $(6.7 \pm .7\text{kHz})^2 \mu s^{-1}$ (Fig-

ure 6.4c). The observed linear behavior suggests a model where the nuclear gradient diffuses, which can arise, for example, from dipolar coupling between adjacent nuclei. Using the measured diffusion of ΔB_z, we simulate the performance of the Bayesian estimate as a function of N(see 6.4.2). Given that the simulation has no free parameters, we find good agreement with the observed T_2^*, indicating that indeed, diffusion limits the accuracy with which we can measure ΔB_z (Figure 6.4a).

This model suggests that increasing the rate of measurements during estimation will improve the accuracy of the Bayesian estimate. Because our FPGA limits the repetition rate of qubit operations to 250 kHz, we demonstrate the effect of faster measurements through software post-processing with the same Bayesian estimate. To do so, we first use the same estimation sequence, but for the operation segment, we measure the outcome after evolving around ΔB_z for a single evolution time, t_{evo}, rather than performing a rotating frame Ramsey experiment, and we repeat this experiment a total of N_{tot} times. In processing, we perform the Bayesian estimate of each $\Delta B_{z,i}$, sort the data by adjusted time $\tau_i = \frac{\Delta B_{z,i} t_{evo,i}}{\langle \Delta B_z \rangle}$ (for $i = 1, 2, \cdots, N_{tot}$), and average together points of similar τ in order to observe oscillations (see 6.4.3). We fit the decay of these oscillations to extract T_2^* and the precision of the Bayesian estimate, $\sigma_{\Delta B_z} = \left(\sqrt{2}\pi T_2^*\right)^{-1}$. For the same operation and estimation parameters, we find that T_2^* extracted from software post-processing agrees with that extracted from adaptive control (Figure 6.4a). Using a repetition rate as high as 667kHz, we show coherence times above 2800 ns, corresponding to an error of $\sigma_{\Delta B_z}$=80kHz (Figure 6.4d), indicating that improvements are easily attainable by using faster (commercially available) FPGAs.

Additionally, we use this post-processin to examine the effect of this technique on the duty cycle of experiments as well as the stability of the ΔB_z estimate. To do so we introduce a delay T_{delay} between the estimation of ΔB_z and the single evolution measurement performed in place of the operation. We find $T_2^* = (a + bT_{delay}^c)^{-0.5}$, where c = 0.99 (Figure 6.4c), consistent with diffusion of ΔB_z. Indeed, this dependence underscores the potential of adaptive control, since it demonstrates that after a single estimation sequence, the qubit can be operated for $>$ 1ms with $T_2^* > 1\mu$s. Thus,

adaptive control need not significantly reduce the experimental duty cycle.

In this work, we have used real-time adaptive control based on Hamiltonian parameter estimation of a $S - T_0$ spin qubit to prolong T_2^* from 70ns to more than 2 μs. Dephasing due to nuclear spins has long been considered a significant obstacle to quantum information processing using semiconductor spin qubits [55], and elimination of nuclear spins is an active and fruitful area of research [1, 70, 43]. However, here we have shown that with a combination of nuclear feedback, rotating frame S-T_0 spin resonance, and real-time Hamiltonian estimation, we are able to achieve ratios of coherence times to operation times in excess of 200 without recourse to dynamical decoupling [28, 65, 7]. If the same adaptive control techniques were applied to gradients as high as 1 GHz [23], ratios exceeding 4000 would be possible, and longer coherence times may be attainable with more sophisticated techniques [56]. Though the observed coherence times are still smaller than the Hahn echo time, T_2^{echo} [7], the method we have presented is straightforward to implement, compatible with arbitrary qubit operations, and general to all qubits that suffer from non-Markovian noise. Looking ahead, it is likely, therefore, to play a key role in realistic quantum error correction efforts [45, 52, 69, 60], where even modest improvements in baseline error rate greatly diminish experimental complexity and enhance prospects for a scalable quantum information processing architecture.

6.4 Methods

6.4.1 Bayesian Estimate

We wish to calculate the probability that the nuclear magnetic field gradient has a certain value, ΔB_z, given a particular measurement record comprising N measurements. We follow the technique in Sergeevich *et. al.*[56] with slight modifications. Writing the outcome of the k^{th} measure-

ment as m_k, we write this probability distribution as

$$P\left(\Delta B_z | m_N, m_{N-1}, ...m_1\right). \tag{6.5}$$

To arrive at an expression for this distribution, we will write down a model for the dynamics of the system, i.e. $P\left(m_N, m_{N-1}, ...m_1 | \Delta B_z\right)$. Using Bayes' rule we can relate the two equations as

$$P\left(\Delta B_z | m_N, m_{N-1}, ...m_1\right) \cdot P\left(m_N, m_{N-1}, ...m_1\right) = P\left(m_N, m_{N-1}, ...m_1 | \Delta B_z\right) \cdot P\left(\Delta B_z\right). \tag{6.6}$$

First, we seek a model that can quantify $P\left(m_N, m_{N-1}, ...m_1 | \Delta B_z\right)$ that accounts for realistic errors in the system, namely measurement error, imperfect state preparation, and error in the axis of rotation around the Bloch sphere. For simplicity, we begin with a model that accounts only for measurement error. Denoting the error associated with measuring a $|S\rangle$ ($|T_0\rangle$) as η_S (η_T), we write

$$P\left(S | \Delta B_z\right) = \left(1 - \eta_S\right) \cos^2\left(2\pi \Delta B_z t_k / 2\right) + \eta_T \sin^2\left(2\pi \Delta B_z t_k / 2\right) \tag{6.7}$$

$$P\left(T_0 | \Delta B_z\right) = \left(1 - \eta_T\right) \sin^2\left(2\pi \Delta B_z t_k / 2\right) + \eta_S \cos^2\left(2\pi \Delta B_z t_k / 2\right) \tag{6.8}$$

We combine these two equations and write

$$P\left(m_k | \Delta B_z\right) = \frac{1}{2}\left[1 + r_k\left(\alpha + \beta \cos\left(2\pi \Delta B_z t_k\right)\right)\right] \tag{6.9}$$

where r_k=1 (-1) for $m_k = |S\rangle(|T_0\rangle)$ and α and β are given by

$$\alpha = \left(\eta_T - \eta_S\right), \quad \beta = \left(1 - \eta_S - \eta_T\right). \tag{6.10}$$

Next, we generalize the model to include the effects of imperfect state preparation, and the presence of nonzero J during evolution, which renders the initial state non-orthogonal to the axis of rotation around the Bloch sphere (see above). We assume that the angle of rotation around the Bloch sphere lies somewhere in the x-z plane and makes an angle θ with the z-axis. We define $\delta = \cos^2(\theta)$. Next, we include imperfect state preparation by writing the density matrix $\rho_{init} = (1-\epsilon)\left|S\right\rangle\left\langle S\right| + \epsilon\left|T_0\right\rangle\left\langle T_0\right|$. With this in hand, we can write down the model

$$P\left(S|\Delta B_z\right) = \eta_T + \frac{1}{2}(1-\eta_S-\eta_T)\left\{1+(1-2\epsilon)\left[\delta\ +(1-\delta)\cos\ (2\pi\Delta B_z t_k)\right]\right\}, \quad (6.11)$$

$$P\left(T_0|\Delta B_z\right) = \eta_S + \frac{1}{2}(1-\eta_S-\eta_T)\left\{1-(1-2\epsilon)\left[\delta\ +(1-\delta)\cos\ (2\pi\Delta B_z t_k)\right]\right\}. \quad (6.12)$$

Using the same notation for r_k=1 (-1) for $m_k = |S\rangle(|T_0\rangle)$, we rewrite this in one equation as

$$P\left(m_k|\Delta B_z\right) = \frac{1}{2}\left[1+r_k\left(\alpha+\beta\cos\left(2\pi\Delta B_z t_k\right)\right)\right], \quad (6.13)$$

where we now have

$$\alpha = \eta_T - \eta_S + (1-\eta_S-\eta_T)(\delta-2\epsilon\delta) \quad (6.14)$$

$$\beta = (1-\eta_S-\eta_T)(1-\delta)(1-2\epsilon). \quad (6.15)$$

We find the best performance for $\alpha = 0.25$ and $\beta = 0.67$, which is consistent with known values for qubit errors.

We next turn our attention to implementing Bayes' rule to turn this model into a probability dis-

tribution for ΔB_z. First, we assume that all measurements are statistically independent[‡], allowing us to write

$$\begin{aligned} P\left(\Delta B_z | m_N, m_{N-1}, ...m_1\right) &= P\left(\Delta B_z | m_N\right) \cdot P\left(\Delta B_z | m_{N-1}, ...m_1\right) \\ &= \prod_{k=1}^{N} P\left(\Delta B_z | m_k\right). \end{aligned} \tag{6.16}$$

We next use Bayes rule (6.6) and rewrite this equation as

$$P\left(\Delta B_z | m_N, m_{N-1}, ...m_1\right) = \prod_{k=1}^{N} P\left(m_k | \Delta B_z\right) \frac{P(\Delta B_z)}{P(m_k)}. \tag{6.17}$$

Using our model (6.13) we can rewrite this as

$$P\left(\Delta B_z | m_N, m_{N-1}, ...m_1\right) = \mathcal{N} P_0(\Delta B_z) \prod_{k=1}^{N} \left[1 + r_k\left(\alpha + \beta \cos\left(2\pi \Delta B_z t_k\right)\right)\right], \tag{6.18}$$

where $\mathcal{N}$ is a normalization constant, and $P_0(\Delta B_z)$ is a prior distribution for ΔB_z which we take to be a constant over the estimation bandwidth, and to which the estimator is empirically insensitive. With this formula, it is simple to see that the posterior distribution for ΔB_z can be updated in real time with each successive measurement. After the N^{th} measurement, we choose the value for ΔB_z which maximizes the posterior distribution (6.18).

6.4.2 Simulation with Diffusion

We simulate the performance of our software scaling and hardware (FPGA) estimates of ΔB_z using the measured value of the diffusion rate. We assume that ΔB_z obeys a random walk, but assume

[‡]this is often called the naive Bayes estimate

that during a single evolution time t_k, ΔB_z is static. This assumption is valid when $\sqrt{t_N \mathcal{D}} T_2^* \ll 1$, where $\mathcal{D}$ is the diffusion rate of ΔB_z. For an estimation of ΔB_z with N different measurements, we generate a random walk of N different values for ΔB_z (using the measured diffusion), simulate the outcome of each measurement, and compute the Bayesian estimate of ΔB_z using the simulated outcomes. By repeating this procedure 4096 times, and using the mean squared error, MSE $= \langle \left(\Delta B_z - \Delta {B_z}^{estimated}\right)^2 \rangle$ as a metric for performance, we can find the optimal number of measurements to perform. To include the entire error budget of the FPGA apparatus, we add to this MSE the error from the phase noise of the VCO, the measured voltage noise on the analog output controlling the VCO, and the diffusion of ΔB_z during the "operation" period of the experiment.

6.4.3 Software Post Processing

The estimate of ΔB_z can be independently verified using software analysis. In this experiment, we use the same method to estimate ΔB_z as in the adaptive control experiment, but in the operation segment perform oscillations around ΔB_z for verification. We choose m different evolution times and measure each n times for a total of $N_{tot} = m \times n$ measurements of ΔB_z. In the i^{th} experiment ($i = 1, 2, \ldots N_{tot}$), we evolve for a time $t_{evo,i}$, accumulating phase $\phi_i = \Delta B_{z,i} t_{evo,i}$. Because we make a precise measurement of ΔB_z at the start of each experiment, we can employ it to rescale the time, $t_{evo,i}$, so that the phase accumulated for a given time is constant using the equation,

$$\tau_i \equiv t_{evo,i} \frac{\Delta B_{z,i}}{\langle \Delta B_z \rangle}$$

This sets $\phi_i(\tau_i) = \langle \Delta B_z \rangle \tau_i$, with residual error arising from inaccuracy in the estimate of $\Delta B_{z,i}$. The data are then sorted by τ, and points of similar τ are averaged using a Gaussian window with $\sigma_\tau = 0.5$ ns $\ll T \approx 16$ ns, where T is the period of the oscillations.

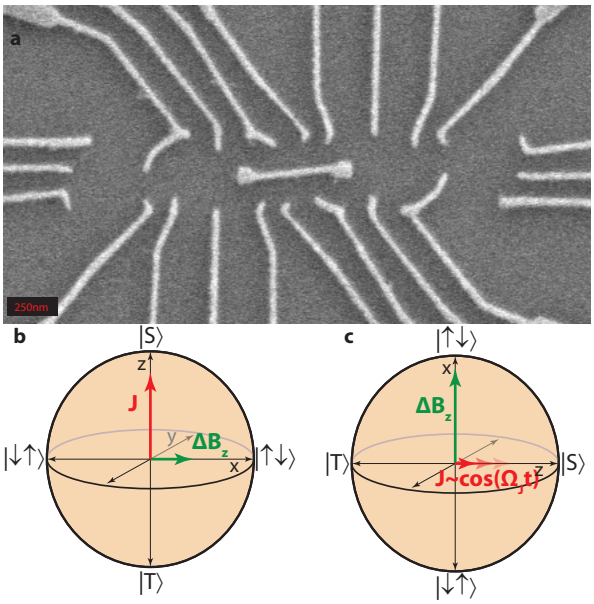

Figure 6.5: a. An electron microscope image of the device used. Two qubits each comprising a double quantum dot and an additional quantum dot for charge sensing are fabricated in close proximity. A floating metal gate is fabricated between the qubits to increase the inter-qubit capacitance. The right qubit is left inactive for this work. **b**. The Bloch sphere representation for the S-T_0 qubit. **c**. The Bloch sphere representation for the S-T_0 qubit in the rotating frame, where rotations are driven my modulating J.

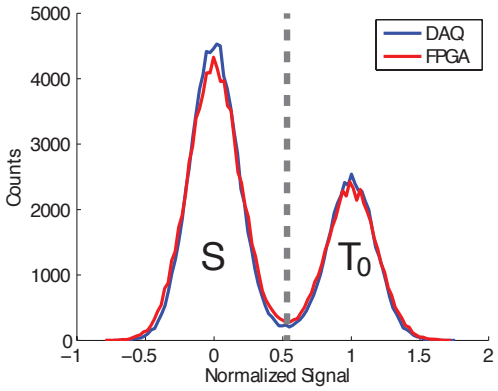

Figure 6.6: a. A histogram of values measured by the data acquisition card (DAQ) and the FPGA/CDS show nearly identical double peaked structures, indicating that they are capable of consistent singleshot readout. The difference in the heights of the two peaks is caused by residual exchange (J) during evolution, which causes the axis of evolution around the Bloch sphere to be non-orthogonal to the initial state. For the Bayesian estimate, which requires discretized data ($r_k = \pm 1$), we choose a threshold (grey dashed line) corresponding to the minimum between the peaks for the adaptive control on the FPGA.The dashed line is chosen as the threshold for estimating ΔB_z with the FPGA.

6.4.4 FPGA and Experimental Apparatus

The reflected readout drive signal returns to room temperature through a cryogenic circulator and amplifier at 4K. The signal is amplified again at room temperature before being demodulated to DC. This DC signal is split and sent to a digitizing card (AlazarTech 660) in a computer and a home built correlated double sampler (CDS). The CDS digitizes the signal and performs a local reference subtraction to reject low frequency noise. The resulting 16 bit signal is converted to a low voltage digital signal and sent to the FPGA for processing. The FPGA is a National Instruments model PXI-7841R and is clocked at 40MHz to maximize processing speed. The probability $P(\Delta B_z|m_k)$ is computed for 256 consecutive frequencies in the estimation bandwidth, $\mathcal{B}$, in two parallel processes on the FPGA to decrease calculation time. Since $\mathcal{B} \approx$ 40MHz is larger than the residual fluctuations of ΔB_z, we increase the frequency resolution by computing the Bayesian estimate of ΔB_z for the the middle 256 frequencies inside of $\mathcal{B}$. For these parameters, the minimum calculation time is $3.7\mu s$ for a single t_k. The probability distributions are stored and updated as single-precision floating-point numbers, since we find that single-precision improves the accuracy of the estimator over fixed-point numbers.

After estimating ΔB_z, the FPGA returns the index (an integer between 1 and 256) of the most probable frequency, which must be converted to a voltage to control the VCO. To do so, we apply a linear transformation to the index, $V = G \times index + O$, where the O controls the detuning of the driving frequency. We tune the G to maximize T_2^* using adaptive control (Figure 6.7a).

6.4.5 Software Post Processing

To compare post-processing with adaptive control, we first perform the same estimation sequence for both software post-processing and adaptive control, with a 250 kHz repetition rate, $t_{samp} =$ 12 ns and $N =$120, followed by an operation sequence of 30 measurements. We find $T_2^* = 2148 \pm$

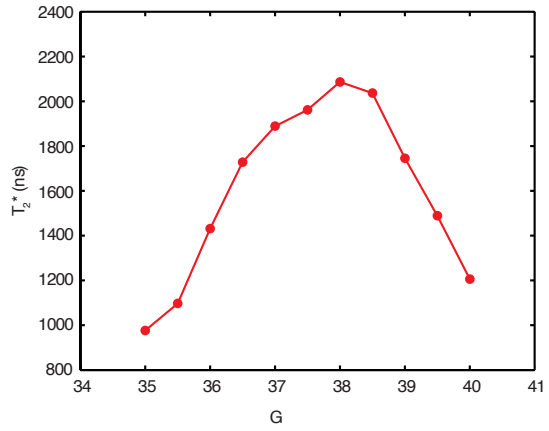

Figure 6.7: a. T_2^* changes with the gain, G, converting a frequency index into a control voltage for the VCO. This allows for the optimal gain to be found.

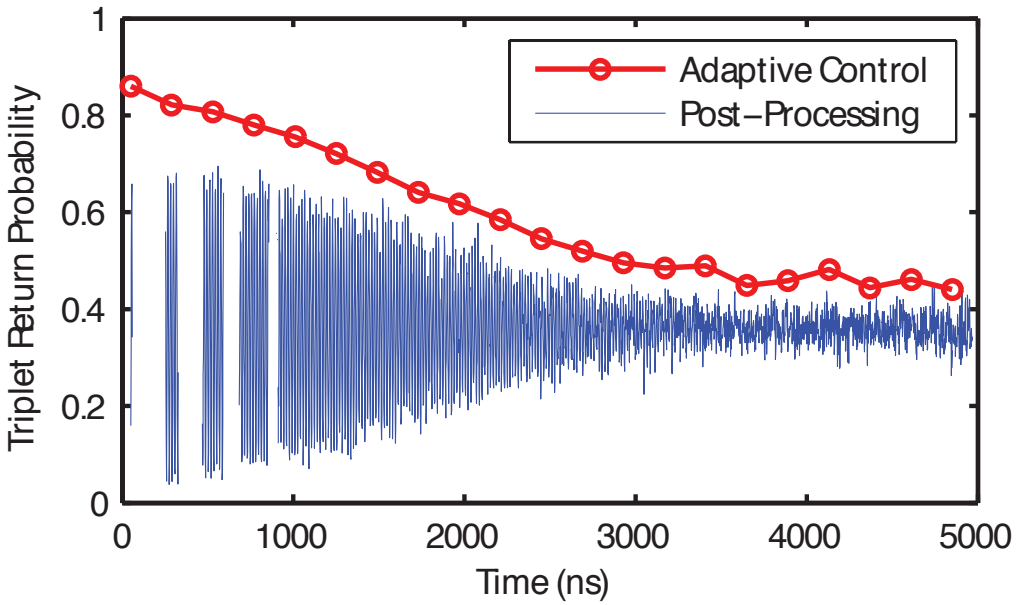

Figure 6.8: a. When using the same estimation sequence, post-processed oscillations (blue) and data taken using adaptive control (red) show the same decay, indicating similar performance of the estimation. The post-processing technique allows us to explore estimation sequences that are too fast for the FPGA.

30 ns with software and $T_2^* = 2066$ ns with adaptive control, showing good agreement between the two approaches (Figure 6.8a).

For the software post-processing, we can reduce the amount of diffusion that occurs during the operation sequence by performing only one verification measurement following the same estimation sequence, enhancing T_2^*, to 2580 ± 40 ns. For the software rescaling in Fig. 4d, the 109 estimations were performed in 225 μs instead of the 440 μs used by the FPGA, yielding $T_2^* = 2840 \pm 30$ ns. This is likely limited by diffusion and the precision of the estimator with N=109.

Acknowledgments

We acknowledge James MacArthur for building the correlated double sampler. This research was funded by the United States Department of Defense, the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), and the Army Research Office grant (W911NF-11-1-0068 and W911NF-11-1-0068). All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies either expressly or implied of of IARPA, the ODNI, or the U.S. Government. S.P.H was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program. ACD acknowledges discussions with Matthew Wadrop regarding extracting diffusion constants. ACD and SDB acknowledge support from the ARC via the Centre of Excellence in Engineering Quantum Systems (EQuS) project number CE110001013. This work was performed in part at the Center for Nanoscale Systems (CNS), a member of the National Nanotechnology Infrastructure Network (NNIN), which is supported by the National Science Foundation under NSF award no. ECS-0335765. CNS is part of Harvard University.

Begin thus from the first act, and proceed; and, in conclusion, at the ill which thou hast done, be troubled, and rejoice for the good.

Pythagoras

7

Conclusion and Outlook

The preceding chapters have described recent progress in quantum information science for singlet-triplet qubits in GaAs quantum dots and hopefully indicated that there are myriad directions for future study as well as outstanding challenges. We have demonstrated progress both for single-qubit and two-qubit operations. For scalable quantum information processing using S-T_0 qubits, the biggest challenge will be to improve the two-qubit operations, since, while provably entangled, a

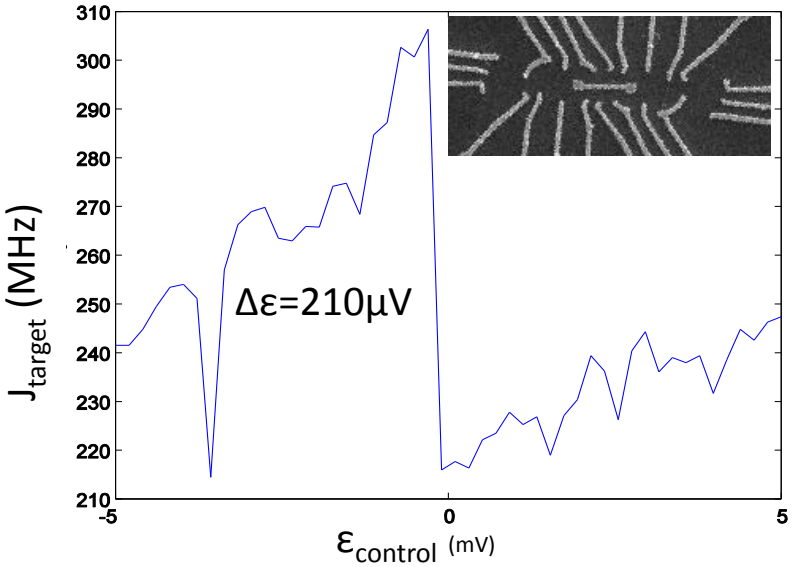

Figure 7.1: The coupling strength measured on a device with a floating metal "dogbone" is approximately twice as large as the coupling without such a gate. Inset: an SEM micrograph of the device in question

Bell state with fidelity of 72% is not good enough to perform quantum computation. For example, if we consider the simplest error correcting code, where the target qubit is entangled with two ancilla qubits, after two entangling operations the target qubit will be approximately 50% mixed.

The most straightforward way to improve the two-qubit operations is two improve the ratio $\frac{\tau_{ent}}{T_2}$, since the fidelity will scale roughly as $e^{-\frac{\tau_{ent}}{T_2}}$. We can improve τ_{ent} byx increasing the inter-qubit capacitance, for example with a floating metal coupler (colloquially called a dogbone) between the two qubits[63]. At the same inter-qubit distance as the device in chapter 3, such a metal gate would decrease T_{ent} by a factor of $\approx$3 and would also allow the qubits to be spatially separated (which is good for scalability) without a great loss of coupling strength. Indeed, the device used in chapter 6 had a such gate, though one of the two qubits on this device did not work properly (possi-

bly due to a broken coaxial cable), and we were were therefore unable to measure a Bell state fidelity. Nevertheless, we were able to measure the coupling strength on this device in a simpler experiment: The exchange frequency in one qubit is monitored as a function of the detuning in the other qubit. When the control qubit is brought through the charge transition, the target qubit sees an effective change in ϵ of 210μV, compared to 90μV without a coupler. Though not quite as large as expected (a factor of 3 was expected in the simulations of reference [63]), the increase in coupling of a factor of two will improve two-qubit operations. What remains to be seen is whether such a floating piece of metal increases the noise or makes tuning very difficult. Indeed, tuning devices with couplers in somewhat more difficult than those without couplers, although it is not impossible.

The ratio $\frac{\tau_{ent}}{T_2}$ can also be improved by increasing T_2 by studying and eliminating charge noise. Such studies would also increase the fidelity of single qubit exchange operations. Though we have yet to definitively identify the source of the charge noise measured in chapter 5, the temperature dependance gives some clues as to the origin of this noise. One likely candidate is the dopant layer of silicon in the heterostructure. To this end, we have started a collaboration with the Manfra group at Purdue University in order to create doping-free materials. It should be noted that although these devices increase the complexity of fabrication, much of the progress in gate-defined Si quantum dots was enabled by switching to doping-free structures. Additionally, removing the dopants is likely to improve tunability and yield of devices.

The results of chapter 5 suggest another avenue to improve two-qubit coupling, namely, to shift the sensitivity to higher-frequency, lower-noise portion of the voltage spectrum. A simple way to do this would be to increase the number of π-pulses applied to the qubits during the entangling operation, though the gains from such a scheme would diminish when the time spent during π-pulses becomes comparable to the entangling time. A natural extention is to operate both qubits in the rotating frame of ΔB_z and to apply an oscillatory J to each qubit. Combined with the results from chapter 6 to prolong the coherence in the rotating frame, there exist several schemes for resonantly

entagling two qubits, for example, by modulating J in each qubit at a frequency $\frac{\Delta B_{z1}-\Delta B_{z2}}{2}$. Such a resonant two-qubit coupling could also be improved by means of a high-quality L-C resonator, perhaps, for example, in the form of a NbN kinetic inductor fabricated on-chip.

Beyond improving two-qubit operations, these techniques can also be used to improve single-qubit operations. One promising avenue to explore is to operate the qubit in the rotating frame of ΔB_z, but to modulate J very deep in the (1,1) charge state, symmetrically around the point where (0,2) and (2,0) are degenerate. Though such a control scheme involves very large voltage pulses, it is, in principle, insensitive to first order to voltage noise.

Before concluding the singlet-triplet qubit prospects for quantum information processing, we remark on the prospects for scalability. These spin qubits hold promise for scalability due to their potential for miniaturization and mass- production, and the biggest challenge for scalability is likely a two-qubit coupling which works over moderate ($\approx$1 micron) distance. Many skeptics cite the large number of DC control lines that must be delivered to the qubit and precisely tuned. We note that the devices measured in this work were always tunable and stable for years[*]. As for tuning an array of 100 qubits, given recent advances in the field of (classical) machine learning, we are confident that adjusting approximately 700 gate voltages would not constitute a significant problem, and the tuning programs used for this work (see appendix C) are certainly a good start.

This work also establishes the singlet-triplet as a good platform for sensitive metrology. In particular, it can be used to measure both electric and magnetic fields. As a voltage probe, it has unprecedented sensitivity of $2 \times 10^{-8} \frac{\mathrm{e}}{\sqrt{\mathrm{Hz}}}$, which is 100 times better than an RF-SET. This sensor has extremely large bandwidth (>20GHz) and retains the capability to slowly initialize and read the qubit[†]. For example, one could imagine adiabatically preparing the qubit in the ground state of ΔB_z(which is stable for more than 10ms), turning on J for some time to be sensitive to a high-

[*] one wonders when dilution refrigerator stability becomes an issue

[†] thanks to Oliver Dial for pointing out this capability

bandwidth signal, and finally reading the qubit slowly. The challenge remains to find a system to study with such a sensor.

The magnetic field sensing capabilities of the S-T_0 qubit are also unique. Owing to the large contact hyperfine coupling in GaAs, the qubit can detect changes in the nuclear magnetic field gradient of only a few nuclear spins. However, this sensitivity would be much lower for detecting systems which are external to the nuclei. Fortunately, the nuclear system is full of new, rich physics, for example spin diffusion. One can also imagine probing the effects of many body localization in this system, since the electrons can detect relatively few spin flips. This system becomes even more interesting when one introduces a coil to coherently drive the nuclear spins.

Ultimately, if some kind of spin qubit is to be established as the building block of a quantum information processor, the singlet-triplet qubit will be invaluable as a sensitive probe of the environment. Moreover, we hope we have established the S-T_0 qubit as a fun and worthy platform of study, external to the pursuit of scalable quantum information processing, because the author certainly thinks that they are.

Nanotechnology is an idea that most people simply didn't believe.

Ralf Merkle

A

Quantum dot fabrication recipe

Gallium-Arsenide nanofabrication has, fortunately, become standand and straightfoward. Here we detail the recipes:

A.1 Wafer

All of the work presented here was done on wafer grown at the Weizmann Institute of Science by Vladimir Umansky. Considerable work and iterations were done at Weizmann by Sandra Foletti

Process Name: mbe8-177.mbe
Description:: CAlibr Al Ga +2DEG 600sp RHEED:
Substrate Type : Semi-Insulating
Run Time : 02:13:40.4
Creation Date: 29-06-2006 14:30:10 Date of Last Change 02-07-2006 15:39:46

#	Material		X%	Y%	N(Si)	(A/min)	Thick	Time	Temp	SL	To
001	Ga4AS4					160.0	0800.0	00:05:00.0	600		
002	Ga4AS4:Si				3.0e+17	160.0	0090.0	00:00:33.7	600		
003	Ga4AS4					160.0	0250.0	00:01:33.7	600		
004	AlGa4AS4		36.0			250.0	0060.0	00:00:14.3	600		
005	Ga4AS4					160.0	0070.0	00:00:26.2	600		
006	AS4	(Ga4)				160.0	0060.0	00:00:22.5	600	012	004
007	Ga4AS4					160.0	1800.0	00:11:15.0	600		
008	AS4	(Ga4)				160.0	0050.0	00:00:18.7	600	004	007
009	AlGa4AS4		8.0			173.9	0010.0	00:00:03.4	600		
010	Ga4AS4					160.0	0020.0	00:00:07.5	600		
011	AS3	(Ga3)				100.0	0100.0	00:01:00.0	600		
012	AlGa3AS3		36.0			156.2	0200.0	00:01:16.7	600		
013	Ga3AS3					100.0	0080.0	00:00:48.0	600		
014	AS3	(Ga3)				100.0	0030.0	00:00:18.0	600	020	012
015	Ga3AS3					100.0	1000.0	00:10:00.0	600		
016	AlGa3AS3		36.0			156.2	0600.0	00:03:50.3	600		
017	Ga3AS3					100.0	0002.8	00:00:01.6	600		
018	AS2	(Ga2)				60.0	0080.0	00:01:20.0	600		
019	AlGa2AS2:Si		36.0		3.1e+18	93.8	0143.0	00:01:31.5	600		
020	AlGa2AS2		36.0			93.8	0100.0	00:01:04.0	600		
021	Ga2AS2					60.0	0070.0	00:01:09.9	600		

Figure A.1: The growth sheet for the material used in this work

in order to find a recipe for a good stable material. We note, that having such "magical" wafer was invaluable for making progress, instead of spending time characterizing wafer or thermal cycling untunable dots, and the author is forever indebted to Sandra and Vladimir for their efforts. The growth sheet for this wafer can be found in Figure A.1. More recently, we have begun to collaborate with the Manfra group at Purdue University. We have received clones of the Weizmann material which are very stable and promising.

A.2 Four-solvent clean

This process should be done before starting processing.

1. TCE, 4 min, sonicating
2. Acetone, 4 min, sonicating
3. IPA, 4 min, sonicating
4. DI water, 4 min, sonicating
5. Bake, 180^{o}C, 4min.

A.3 Mesas

Use this procedure to define the regions where you want to remove the 2DEG. This recipe will not make undercut, so do not try to do liftoff. The four-solvent clean should be done right before.

1. Spin Shipley 1805 at 5000 RPM for 40 seconds
2. Bake the wafer at 80^{o}C for 5 minutes.
3. Expose for 4 seconds on MJB4 in hard contact mode.

4. Develop in MIF 319[*] for 9 seconds

5. Bake the wafer at 100^{o}C for 5 minutes[†].

6. Inspect the chip in the microscope to make sure the pattern came out.

7. Etch in H_3PO_4 : H_2o_2 : H_2o (1:1:25) for approximately 30s. To be thorough, it is best to calibrate the etch for a minute on a junk chip and profile it. Be sure to mix the etchant thoroughly. Rinse in DI water.

8. Remove the resist by sonicating in acetone and IPA.

9. Measure the mesa height on the profilometer. You must go past the dopants (I usually aim for the 2DEG just to be sure).

A.4 Ohmic Contacts

This recipe will make undercut so that evaporation and liftoff work.

1. Spin AZ5214IR at 5000 RPM for 40 seconds.

2. Bake at 100^{o}C for 45 seconds.

3. Expose for 0.4 seconds in MJB4 in hard contact mode.

4. Immediately bake at 120^{o}C for 55 seconds. [‡]

5. Floor expose (no mask) in MJB4 for 45 seconds

6. Develop in AZ726 for 22 seconds[§].

[*] This has TMAH, so be careful

[†] I accidentally skipped this once with no problems

[‡] have the hotplate ready at this temperature before exposing

[§] you can always repeat the flood expose and develop once if the features do not appear to have good undercut

7. Evaporate Ni/Au/Ge 6/30/60nm ¶

8. Evaporate 20nm Ni, followed by desired amount of gold to make sure the metal is taller than the mesa.

9. Liftoff in hot acetone. Inspect in the microscope to make sure the features came out.

10. Anneal: 120^{o}C for 1 minute, 340^{o}C for 1 minute, ramp from 340^{o}C to 460^{o}C in 30 seconds, hold at 460^{o}C for 2 seconds, stop the heat.

A.5 Small Gates and Electron Beam Lithography

1. Spin PMMA 950 A2 at 4000 RPM for 45 seconds (should be around 80nm).

2. Expose on the Elionix-7000 (100kV acceleration voltage): For small features use a 75 micron chip, 60000 dots, approximate dose of 1900 $\mu C/cm^2$. For larger features a 600 micron chip and 60000 dots.

3. Develop in cold MIBK:IPA (1:3) for 60 seconds. The developer should be kept on ice water for several minutes to make sure its cold. Rinse in IPA.

4. Evaporate PdAu (60:40 by weight) 25nm.

5. Liftoff in hot acetone.

A.6 Large Optical Gates

1. Do the standard AZ5214 lithography as in A.4

¶ In the past people have cleaned in UV ozone and dipped the sample in a weak acid directly before evaporation in order to remove resist and oxide, which can spoil Ohmic contacts. We have not seen this to help, so we skip this step. Waiting more than approximately one day between defining the mesas and making Ohmics *can* spoil the quality of the Ohmics.

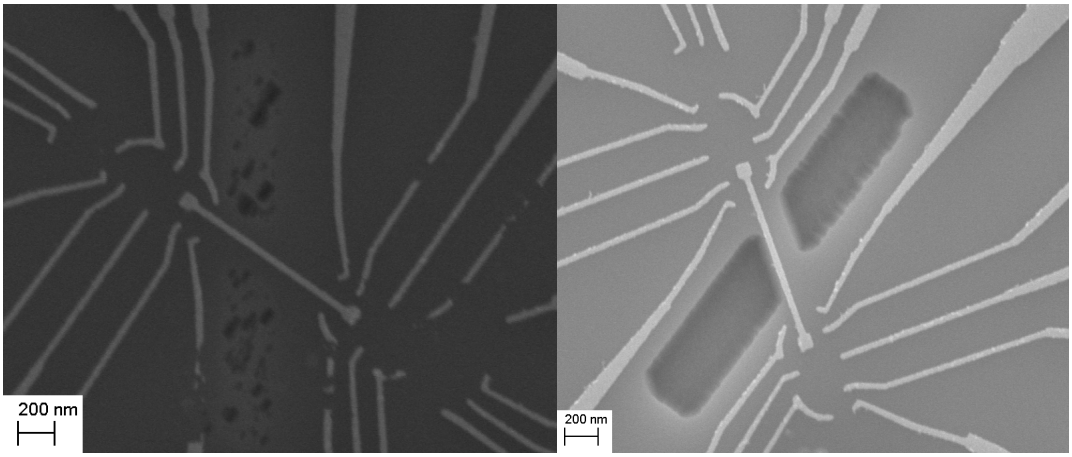

Figure A.2: Left: Annealing the Ohmic contacts prevents the subsequent wet etch from reliably starting, leaving a "swiss cheese" pattern instead. Right: If the etch is performed before annealing it can be done reliably.

2. Evaporate 20nm Ti and enough Au to climb the mesa.

3. Liftoff in hot acetone.

A.7 Trench Warfare

New devices might, in the future, require etching nanoscale trenches in the GaAs to remove small section of 2DEG and decrease the parasitic capacitance of components. Standard reactive ion etches are very effective at defining nanometer-sized structures since they etch non-isotropically, but these etches are known to degrade the quality of the 2DEG and to leave deposits in the trenches. Instead, we rely on isotropic wet etches, exactly like those described in A.3. Through much soul destroying experimentation[‖], we determined that the wet etches do not reliably start after Ohmic contacts (defined with AZ5214) have been annealed. Instead, "swiss cheese" patters form Figure A.2left, which is unsurprising given that this resist is known to leave residues, and is usually lifted off with more aggressive chemicals (which attack GaAs). Instead, it is necessary to first perform the Ohmic

[‖]the leading theory for the cause of our fabrication troubles was that fab works for Oliver and not for Mikey

lithography, define alignment marks, use electron-beam lithography to define the trenches, etch the trenches, and only after etching the trenches anneal the Ohmic contacts. After this, fab can proceed as usual (small gates, large gates, etc). Like this, we can define qubits which separated by micron-scale distances, are coupled with a floating metal gate, and have the GaAs in between them removed Figure A.2right.

God hangs the greatest weights upon the smallest wires.

Francis Bacon

B

Experimental Wiring

Spin qubit experiments in semiconductor quantum dots require a unique set of measurment conditions and apparatus. At the very least, we require that the temperature of the system be significantly less than the charging energy of the quantum dot, so that the number of electrons in the quantum dot is a well- defined quantity*. Furthermore, we require that the temperature be smaller

*In fact, without this requirement, we cannot even call the system a quantum dot. One definition of a quantum dot is a system where the number of particles in a good quantum number

than the singlet-triplet splitting in the quantum dot (≈1K) so that we can load and read the state of the qubits. However, in light of the temperature-dependent noise found in chapter 5 (though these findings were obviously not known at the onset of this work), and in order to have an efficient way to load the $|T_+\rangle$ state, lower temperatures lead to better performance, and we therefore perform these spin qubit experiments in a dilution refrigerator (Oxford MX50). A schematic of the wiring can be found in Figure B.1, and we also outline the wiring that is in the dilution refrigerator.

B.1 Low-bandwidth wiring

The low-bandwidth wiring is used to supply voltages to the surface electrodes that defining the confining potential of the quantum dots and trap electrons. We use constantin (a Cu-Ni alloy) loom from room temperature down to the mixing chamber, wrapping and anchoring the wires to copper posts at the various stages of the dilution refrigerator with GE varnish. In order to filter these lines we use single pole RC filters, 1/11 voltage dividers, and 1.9MHz π-filters (minicircuits BLP1.9) at room temperature. On the cold finger we use a filtered microD connector (critek FMD-1D-0N5CH) and 50Ω resistors that are stycast into the plates of the cold finger, and on the sample board, we use an additional 2kΩ resistors and 100nF capacitors to further filter these lines.

B.2 High-bandwidth control

In order to deliver high-bandwidth (DC-≈1GHz) control pulses to the qubits we use coaxial cables. We choose UT85 SS-SS coax (with crimp-on sma connectors) from room temperature to 4K since SS provides a good balance between low thermal conductivity and high electrical conductivity[†]. The skin- effect losses of these coaxes from room temperature to the mixing chamber stage of the refrigerator is approximately 7dB at 1GHz. To thermalize the outer conductor of the coaxial cable we use

[†]CuNi UT47 would likely be better

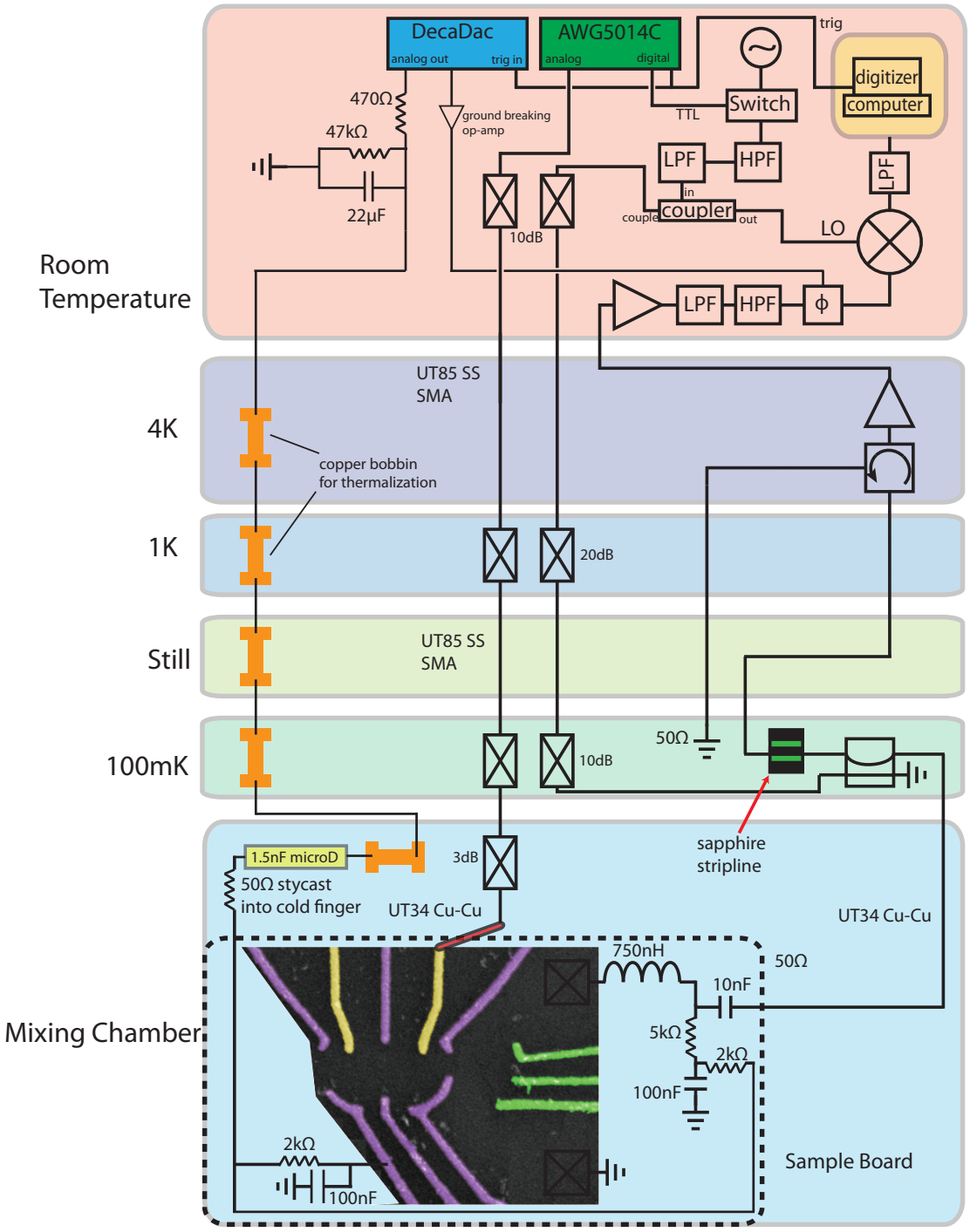

Figure B.1: Left: A schematic of the wiring in the dilution refrigerator used for spin-qubit experiments. Right: a schematic of the frequency multiplexing setup used to read two qubits on one circuit.

a bulkhead feedthrough at every plate of the refrigerator. Thermalizing the inner conductor is more difficult, since it cannot be thermally anchored to the plates of the refrigerator. We use attenuators at the 1K, coldplate, and mixing chamber stages.

From the mixing chamber stage down to the sample, we use UT34 Cu-Cu coax with smm connectors (solder on). These coax are soldered to the sample board and bring signals up to the wire bonds which connect to the device. In contrast to previous experiments in this refrigerator, we did not use bias-Ts to add the high frequency signals instead of adding them to DC gates. This greatly simplifies qubit operations at the cost of having extra gates which are held near 0 DC voltage.

B.3 Qubit readout

In order to read the charge state of the quantum dot in approximately 1μs, we use radio-frequency (RF) signals to determine the impedance of a proximal charge sensor (another quantum dot in this work). We cannot, however, measure the transmission of RF through the charge sensor, since the RC time of the sensor and fridge wiring $t \approx .1\frac{h}{e^2} \times 500\text{pF} \approx 12\mu$s is too long and the sensor cannot respond fast enough. Instead, we employ RF reflectometry techniques, which infer the impedance of the sensor from the amount of reflected RF power. We defer the reader to Christian Barthels's thesis[2] for a nice treatment of RF reflectometry.

The input RF signal travels down the refrigerator through a SS-SS UT85 coax, attenuated at the various stages of the refrigerator like the high-bandwidth control lines. The signal enters a directional coupler at the mixing chamber stage, with most of the signal going to a 50Ω terminated port, and a small portion going to the tank circuit and charge sensor. The reflected signal passed through the directional coupler. Unlike the signal entering the dilution refrigerator, the reflected signal returning from the device cannot be attenuated since this would hurt the signal to noise ratio. Instead, the inner conductor is thermalized by a copper stripline on sapphire, which is thermally anchored

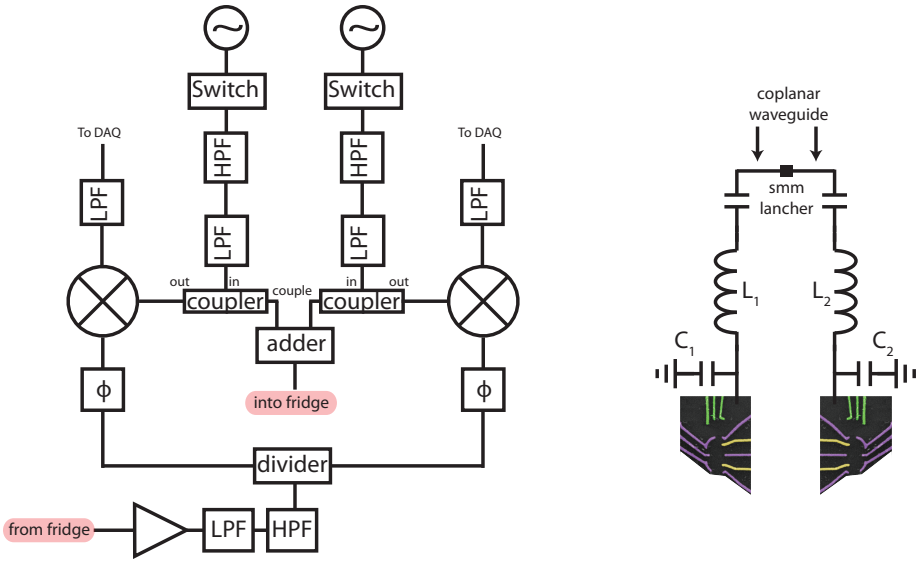

Figure B.2: Left: A schematic of the multiplexing scheme used to read two qubits on one circuit. Right: The sample-board portion of the circuit.

to the mixing chamber. Sapphire is a reaonably good thermal conductor and an electrical insulator, which makes it uniquely suited for this task. More information on the design can be found in Sandra Foletti's thesis[22].

After being thermalized by the sapphire stripline, the reflected signal passes through a cryogenic amplifier and circulator. The cryogenic amplifier is used since a room temperature amplifier would add excess noise to the signal. The circulator is used to make sure the signal which reflected from the amplifier, which are at 4K, do not reflect to the sample and cause unwanted heating and noise, but are instead directed to a cold (100mK ground). In turn, the sample only sees Johnson noise arising from a 50Ω resistor at 100mK $\approx 17 \frac{\mathrm{pV}}{\sqrt{\mathrm{Hz}}}$.

At room temperature, the signal is amplified once more by a low noise amplifier, phase shifted, and demodulated to baseband before being digitized by a data acquisition card in the computer (Figure B.1).

B.4 Multiplexed readout

The readout electronics described above enable singleshot readout of a single qubit with an integration time of $\approx 1\mu s$. However, since the dilution fridge is equipped with only six coaxial cables, one need to multiplex two qubit readouts onto one readout circuit to enable fast readout in two- qubit experiments. The wiring inside the dilution refrigerator needed for two- qubit experiments is identical to that needed for single qubit experiments, but the room temperature wiring is somewhat more complicated. Two RF signal generators are used, and the signals are added by a resistive adder before entering the dilution refrigerator. In this way, two RF frequencies are delivered to the sample board, and selectivity is achieved by having two tank circuits with sufficiently different resonant frequencies (see below B.5). The returning signal with both RF frequencies is amplified both at 4K and at room temperature. The qubits are individually selected by splitting the returning signal with a resistive divider, and individually phase shifting and demodulating the signals with the appropriate frequency. The bassband signal for each qubit is then directed to different channels on the acquisition card. A schematic of the wiring can be seen in Figure B.2.

This multiplexing scheme delivers two RF frequencies to the sample board. Having two tank circuits whose resonant frequencies are farther apart than the resonance widths ($\approx$20MHz) insures that only the correct RF frequency reaches each sensor. In principle, the two frequencies could be very far apart, however, the cryogenic circulator has a bandwidth of approximately 20MHz (220-240MHZ), which requires the use of 220MHZ and 240MHz for the two tank circuits. These frequencies can easily be obtained using slightly different inductors (for example 760 and 620nH) for the two tank circuits, as well as gimmicks (a piece of ground wire) that can be bent to change the parasitic capacitance in the circuit. Variable capacitors (veractors) could be use, although there require an additional control line and can add unnecessary electrical noise.

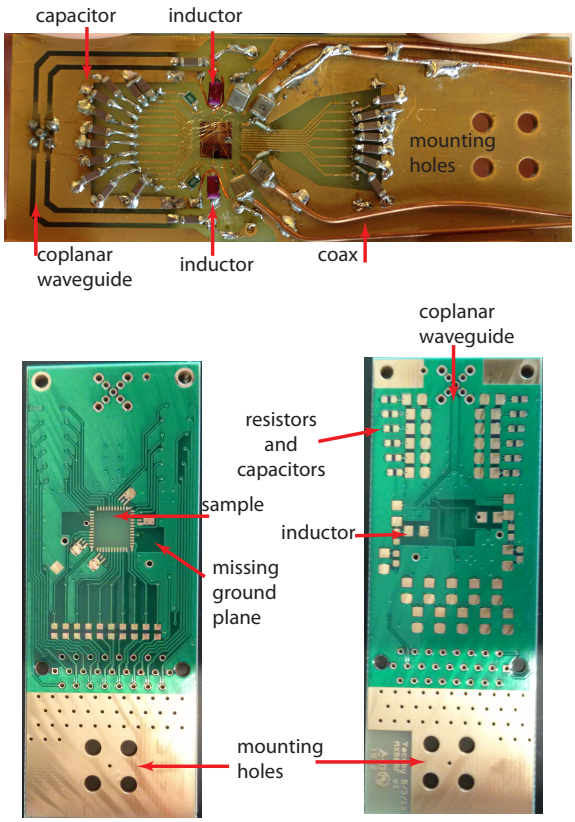

Figure B.3: Top: The old sample board used in chapters 3,5. Bottom: The new sample board used in chapter 6

B.5 Sample board

The sample board is the final link between control signals and the device, and delivers both low frequency and high frequency signals to the quantum dots. The low-frequency signals are delivered through a 25-pin microD connector and pass through on-board RC filters before reaching the bond pads. The high frequency control signals travel through UT34 Cu-Cu coax, which are soldered directly to the bond pads, which are wire bonded to the sample. The readout signal is delivered to the board through an smm launcher and travels through an on-board coplanar waveguide before reaching the tank circuits.

The experiments performed in chapters 3 and 5 used a sample board designed by David Reilly. The RC filters were comprised of leaded resistors which held the microD connector in place and surface mount capacitors which were soldered in place. This board presented some challenges in board preparation and wire bonding[‡] (Figure B.3top). The experiments in chapter 6 we performed on a sample board that was redesigned by Oliver Dial. Improvements included the use of solder mask, specific locations to place components, strain relief for the coax, as well as a device-specific bonding map for improved ease of bonding (Figure B.3bottom).

[‡]but brought the "ninja-bond" into popularity

The first rule of any technology used in a business is that automation applied to an efficient operation will magnify the efficiency. The second is that automation applied to an inefficient operation will magnify the inefficiency.

Bill Gates

C

Tuning Qubits: The idiot's guide to autotune

As mentioned eariler in this thesis, much of the progress made in this work was enabled by thinking of these qubits as qubits, instead of complicated mesoscopic systems. Tuning of the quantum dots to behave like qubits is no exception to this.

Many quantum dot parameters must be tuned in order to operate singlet-triplet qubits. One

must control the chemical potential (charge occupancy) in each of the two quantum dots, the tunnel coupling to each lead (the"transparancy" of the leads), and the tunnel coupling between the quantum dots. Happily, thanks to talented growers and a tunable semiconductor environment, these parameters can be tuned with reasonable ease. The seminal quantum dot experiments inferred relevant parameters from more primitive measurements. For example, the "lead speeds", or characteristic tunneling time of the leads, was inferred by the DC conductance of the double quantum dot system. However, such a measurement measures a combination of *both* lead conductances and the tunnel coupling between the two quantum dots. We prefer a more direct measurement of the lead conductance: we measure pulse from one charge state to the other, and we measure the time domain response of the charge sensor.

Here we describe autotune, a program written to aide in tuning up quantum dots for spin qubits. It is written in matlab, and is an add on to the matlab data acquisition software called special measure. Autotune will allow you to directly measure and tune the various parameters of a double quantum dot, with an emphasis on the parameters relevant for tuning a singlet-triplet qubit.

Before beginning autotune, we assume that the quantum dot is tuned to the correct charge transition (assumed to be (0,2)-(1,1)). The procedure for finding this transition is usually straightforward, and are outlined in Sandra Foletti's thesis[22]. Transport measurements are used to find Coulomb blockade, and charge sensing measurements are used to find the last electron in the quantum dots. Empirically, we have found that early switching from transport to charge sensing is advantageous- one should use charge sensing as soon as it works because it is a more direct measurement of the relevant quantities.

Lastly, we note some reasonable values for the relevant dot parameters. The dots should be tuned to the (0,2)-(1,1) charge transition. The tunnel coupling should be approximately $1\mu V$. One lead is required to be fast, which means that the characteristic tunneling time should be less than 50ns. The

other lead can be left fast or slow. If it is very slow (>100μs) one can use the (2,1)-readout trick*. We have seen no change in charge noise (both from FID and echo measurement) by tuning the slow lead conductance.

C.1 TUNEDATA

Tunedata is a struct that has all of the information needed to tune the dot. It keeps a history of everything that has happened and is able to adapt to changes in the parameters of the dot. A few concepts:

run: A "run" represents all of the parameters of the dot for a given tuning (i.e. all of the voltages on the gates are the same).

set: A set represents a quantum dot, so a two qubit device might have two "sets" named, for example, "left" and "right." A set, therefore, will be comprised of an array of "runs" which will accumulate as the tuning of the dot changes.

As we will see, tunedata is structured as follows: tunedata.(somefield) (e.g. tunedata.stp) has information for measuring some infomation about the active qubit, and tunedata.runs(xx).(somefield) will hold the history of the evolution of the parameters that are extracted from these scans. For example, tunedata.runs(end).stp will hold the last know information about the S-Tplus peak of the ACTIVE qubit. tunedata.sets is essentially an array of tunedatas for the inactive qubits. This means that tunedata.sets(set_ind).stp has scan information for measuring the stp of the set that corresponds to set_ind, and tunedata.sets(set_ind).runs(end) has the last know S-Tplus peak location

*In this trick, one (right) lead is very fast and the other (left) lead is very slow. To read the qubit, the levels are detuned so that the (1,2) charge state is between the (1,1)$|T_0\rangle$ and the (0,2)$|S\rangle$. Therefore, if the qubit returns as a $|T_0\rangle$ it is shelved to the (1,2) charge state through the fast lead, and cannot relax to the (0,2) charge state since the left lead is slow. If the qubit returns as a $|S\rangle$ the charge state will be (0,2). Therefore, the state of the qubit can be determined by distinguishing between the (1,2) and (0,2) charge states. This readout scheme offers two advantages: it can work at large ΔB_z if the tunneling rate $\gg \Delta B_z$, and there is increased SNR since there is an entire electron distinguishing the two change states, instead of an electric dipole in the convention (1,1)-(0,2) readout

of that set. When we change which qubit we want to tune (i.e. change which qubit is active) we will swap information in and out of tunedata.sets, but this will be handled by atswap (see below). A schematic of tunedata can be seen in Figure C.1.

The typical workflow for a single qubit (set), might be as follows: Measure the parameters of the dot and populate a single "run," realize that one parameter is not ideal, tweak the gate voltages to have the desired change, and start a new "run" to measure the parameters again.

In a nutshell, the fields of tunedata hold the information of the last know parameters of the active dot (the last "run" or the current "set") , and they will be swapped into other places as the tuning of the dot evolves.

Let's look at the fields in tunedata:

dir: the directory to save all scans (referenced from the current data directory)

runs: a struct containing all of the history of the current (active) qubits (set).

chrg: the information for the charge scan.

lead: the information from the lead scan.

meap: the location of the measurement point in the current run.

offset: the default offset to find the measurement point from teh charge diagram.

load: the information for the load scan.

read: the information for the read.

gatechan: the numbers of the channels in smdata that correspond to the gate voltages to track while tuning occurs.

cntr: , the center to aim for. Usually [0,0]

line: the information for the line scan.

plscal: deprecated, for calibrating biasTs

basenames : a cell array of the names of "directions" in gate space the tunedata knows about. These will be used with atchg. For example, atch('XL',4e-3), will attempt to move in the 'XL' direction (move the charge transition of the left qubit in the X direction) by 4mV.

stp: the information for the spt scan.

tl: the information for the tl scan.

fine_ind: the index for tracking and naming scans for runs that will allow some fileds to have multiple entries in the same run. For example, we might have more than one stp scan within a run, and fine _ind will track these.

stp_tgt: The target to have the location of the stp peak. Used for centering. (This should probably be put in the stp struct)

name: The name of the current set. For example, ''right'.

tl _tgt: The target to have the location of the tl peak. Used for centering. (This should probably be put in the tl struct).

sets : A struct array of the different sets in tunedata. When changing which qubit is actively tuned the relevant set is swapped into tunedata, and the other sets will be swapped back into tunedata.sets(xx) for safe keeping. This will be handled by atswap (see below).

resp: the information from the resp scan.

xychan: a cell array of the names of the "directions" (from tunedata.basenames) that will be used to center the dot.

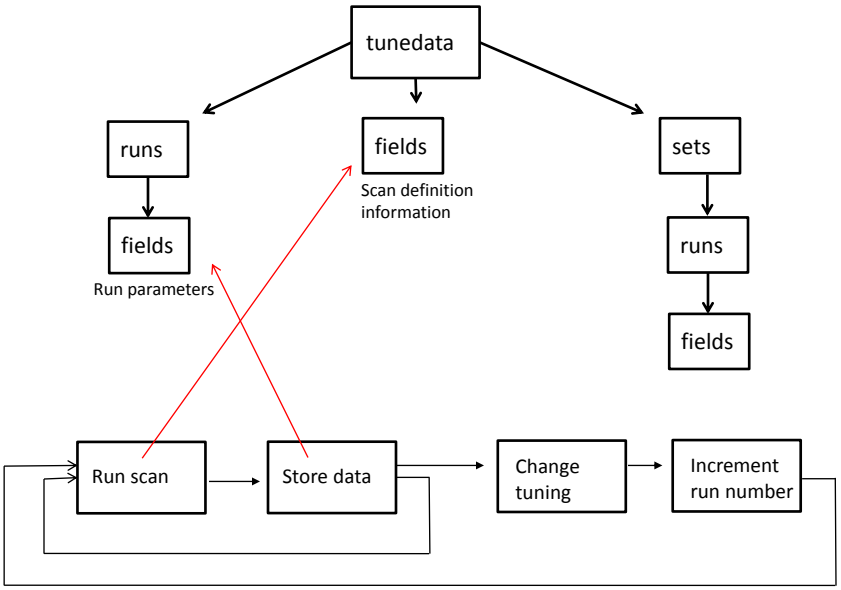

Figure C.1: The structure of tunedata and a typical workflow for using autotune

gradchan: the indecies of the channels (in smdata.channels) that correspond to the channels that will compose the gradient matrix that will make tuning the dot easier (see below).

tmp: a struct with information used in the tmp scan (Tune Measurement Point).

T1: the information for the T1 scan.

tldir : the direction of the top lead (in gate space).

vis: the information for the vis scan.

basis: the gradient matrix that holds information with cross capacitance that will allow for easy tuning.

The previous section listed a lot of different scans that autotune can do. Let's describe them a little better.

C.2 CHRG

The charge scan of autotune. It will take a small honeycomb charge diagram. It can be called with:

```
autotune('chrg')
```

The charge scan is usually the first scan when starting a new "run." This will cause tunedata to ask you what to do for the gradient matrix for the new run. Options:

a: analyze a new gradient matrix

c: copy the gradient matrix from the previous run

i: ignore

r: repeat the last run: i.e. do not start a new run. This will cause you to overwrite (after confirming) all of the scans in the previous run.

The scan stored in tunedata.chrg.scan will be run. It can be modified by passing autotune

```
autotune('chrg fine')
```

or

```
autotune('chrg wide')
```

to make the scan have more points or a wider range. Analysis will happen in chargeana2.m. The analysis will extract the locations of the triple points and the slopes of the leads in tunedata.runs(end).chrg.

The autoanalysis in chargeana2 will attempt to find the cross correlation of the data taken with than in tunedata.chrg.imgl or tunedata.chrg.imgr.

As it will be many of the autotune scan, analysis can be run without taken data (say, if you changed the analysis code) by passing 'ana' to tunedata. For example

```
autotune('chrg ana')
```

The autoanalysis can be disabled (you'll need to click on the triple points)

```
autotune('chrg ana man')
```

and super manual mode (finding the triple points and leads manually) by then hitting the "m" key.

C.3 zoom

The zoom scan will zoom in on the charge scan and will apply a pulse to find Pauli spin blockade. It can be called with:

```
autotune('zoom')
```

The scan will be the same as tunedata.chrg.scan, but will have a range of tunedata.chrg,scan.zoomrng, and a resolution of tunedata.chrg.zoomres. The pulse applied will be tunedata.chrg.zoompls. The analysis will prompt you for input. It will guess where the measurement point is, but you can hit "m" and select it yourself. This will update tunedata.measp! As in the charge scan, the zoom scan can be analysed manually by running

```
autotune('zoom ana')
```

which will again prompt you for input. Like the charge scan, the zoom scan will put the awg into amp mode, as it is needed to run the scan.

C.4 Lead

The lead scan will measure the tunneling rate of electrons onto and off of the quantum dot. It can be called with:

```
autotune('lead')
```

It will apply a square pulse across the lead and subtract a background of the same pulse applied away from the lead. The analysis will fit an exponential to the difference in order to find the tunneling time. The results will be stored in tunedata.runs(end).lead.

C.5 line

The line scan is the poor man's way to measure the tunnel coupling. It can be called with:

```
autotune('line')
```

It will scan across the (0,2)-(1,1) charge transition and fit the modified Fermi-function to extract the width. The results will be saved in tunedata.runs(end).line.

C.6 stp

The stp (S-Tplus) scan will locate the location (ϵ) of the S-Tplus transition. It can be called with:

```
autotune('stp')
```

The struct in tunedata.stp has information (pulsegroup, nrep, nloop, etc) for running the scan. The analysis will locate the peak and store the result in tunedata.runs(end).stp.

C.7 TL

The tl (top lead) scan will locate the location (ϵ) of the top lead. This will be used for loading Tplus for use for feedback. . It can be called with:

```
autotune('tl')
```

The struct in tunedata.tl has information (pulsegroup, nrep, nloop, etc) for running the scan. The analysis will locate the peak and store the result in tunedata.runs(end).tl.

C.8 TMP

The tmp will tune the measurement point. It can be called with

```
autotune('tmp')
```

It will attempt to pick the measurement point such that the stp peak and tl peak will be at their targets (tunedata.stp_tgt and tunedata.tl_tgt). It will confirm the location of the new measurement point by raising figure(2) (the zoom figure) and placing a triangle on the new measurement point.

C.9 T1

The T1 scan will measure the T1 DURING MEASUREMENT. It can be called with:

```
autotune('t1')
```

The scan will prepare singlets and triplets and measure them while oversampling during measurement. This will allow you to measure T1 and find the optimal measurement time.

Other useful awgcommands: awggroups; print the awggroups that are in memory with some meta data atplschk(grps, inds, opts); a nice graphical way to check pulses

C.10 atcenter

atcenter is the function that will attempt to move tunedata.measp to (0,0). It uses the gradient and the basis in tunedata to get this right.

C.11 atchg

It is often useful to try to change one parameter of the dot tuning (e.g. the coupling to the lead of the quantum dot) and leave the rest of the tuning, (in particular the location of the charge transition) the same. However, simply changing one of the gate voltages that defines the tunnel barter to the lead will change both the lead coupling and the location of the charge transition. Instead, we use atchg, which can be called with:

```
atchg('Lead1',2e-3);
```

which will change the gate that couples to the lead of dot 2 ('2b') by 2mV and adjust other gates to keep the charge transition centered. All of the basis vectors in tunedata.basis can be passed to atch.

C.12 atxyfix

Often, after much tuning, the X and Y basis vectors change. Accuracy of these vectors is important to make sure we can center the quantum dots. For this, we use atxyfix. The general procedure will be to locate the ST+ crossing and TL location, and measure the change in these locations as different gates are changed.

C.13 atswap

Once we know how to tune a qubit, we might want to tune more than one. To change which qubit we tune, we use atswap. The syntax is atswap(side), where side is the name of one of the sets in tune-

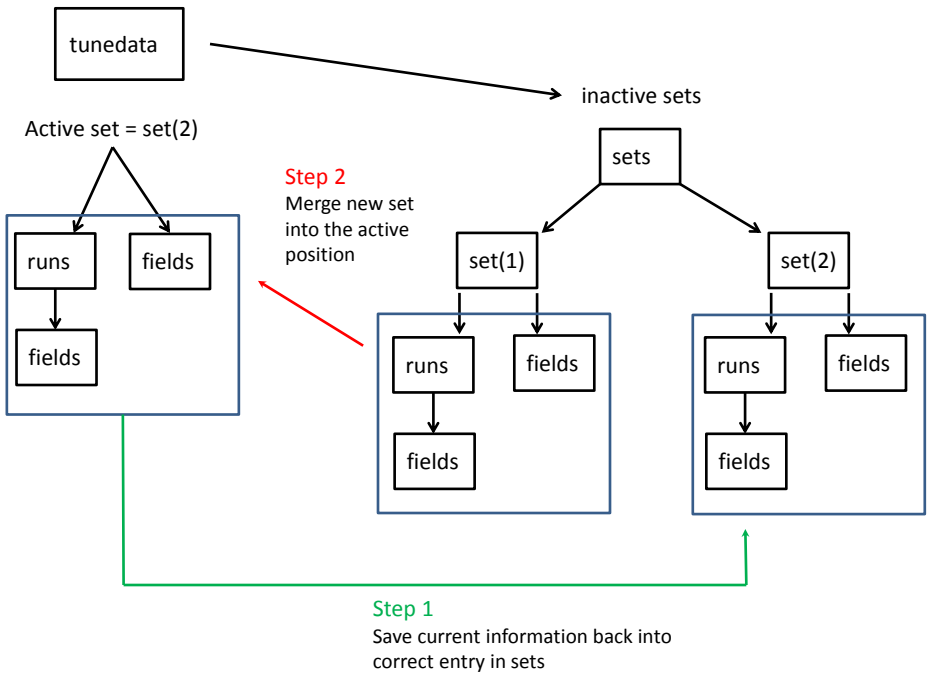

Figure C.2: A schematic of how atswap will allow you to tune different qubits. In this example, set(2) is active, and we swap set(1) into the active position.

data. For example, we might try:

```
atswap('right')
```

which will swamp to the set named "right." The program will first merge the current or active fields of tunedata (i.e. tunedata.(fields) and tunedata.runs, into the appropriate entry in tunedata.sets, and then will take the new active set from tunedata.sets, and use it to populate the active fields of tunedata (tunedata.(fields) and tunedata.runs). A schematic of this operation can be seen in Figure C.2.

C.14 Typical Work Flows

In the beginning, you will want to get the approximate shape of the dot correct. This will likely be iterating charge, lead, line scans. You might, for example, run:

```
autotune('chrg lead line')
```

Perhaps the lead for dot 2 is too slow, and the line scan shows that the junction is too narrow (need more tunnel coupling. You might then run:

```
atchg('Lead2',5e-3);
atchg('N12',5e-3);
```

You would then repeat and run

```
autotune('chrg lead line')
```

Each charge scan will create a new run, and hopefully the basis will become accurate. At the beginning, the gradient will not be accurate, and you might want to use

```
atcenter;
```

after each iteration to keep things centered.

After things get close, you might add the zoom scan to the mix to make sure you have the right charge transition and hope that the S-T splitting remains large.

```
autotune('chrg lead line zoom')
```

This will also allow you to locate a reasonable measurement point. Once the zoom region looks nice and the charge scans analyze properly, you will want to run

```
autotune('updateoff')
```

to update the offset between the triple points and the default measurement point.

Once things look reasonable you will want to begin locating the stp and tl locations. First, start with:

```
autotune('chrg lead line zoom')
```

Pick a nice measurement point. Then run

```
autotune('stp tl')
```

and update the stp and tl targets and the pulsegroups to reflect the locations of the stp tl. Then re-run the scans:

```
autotune('stp tl')
```

(which will require overwriting them). Now the locations and targets should be close. We can then run

```
autotune('tmp')
```

to tune the measurement point based on the new information. Now the workflow might look like:

```
autotune('stp tl tmp');
atcenter;
```

This sequence can be repeated until it converges. This will NOT require a new run each time because of the use of tunedata.fine_ind. Once the sequence converges, it can be repeated it awg raw mode:

```
awgcntrl('on start wait raw');
autotune('stp tl tmp');
atcenter;
```

Once things converge again, you're ready for nuclear feedback and experiments! You can see an example workflow in Figure C.1

If you think your users are idiots, only idiots will use it.

Linus Torvalds

D

The Idiot's Guide to Plscontrol

Spin qubit experiments are not like other physics experiments, in that we can repeat experiments approximately every microseconds. It is rare, therefore, that one finds experiments that run over night, and instead, human input is usually the limiting factor in progressing. As such, it is important to develop tools to allow the experimentalist to quickly progress for conceiving of an experiment to executing it. Here we describe plscontrol- a set of tools for thinking of quantum information experiments at a high level, while still being flexible enough to allow for fast prototyping of new

features and experiments. The original version of plscontrol was written by Hendrik Bluhm, and numerous people have contributed to its revisions.

D.1 Core Concepts

awg: the machine that makes the voltages that we send to the qubit. pulse: a sequence of voltages that can be made/saved/sent to AWG, etc. A pulse is generally an entire experiment, i.e. reload, evolve, measure.

pulsegroup: a number of pulses that generally have something in common, and therefore are meaningful together (maybe, measuring some evolution for 100 different times).

Our Pulse Control software has two global structure that contain most of the relevant information plsdata: a global structure that contains information about the pulses we make. awgdata: a global structure that contains knowledge of the awg and its parameters.

D.2 PLSDATA

Plsdata is a global struct that handles everyting to do with pulses. It has fields:

plsdata.datafile: where plsdata gets saved

plsdata.grpdir: where the pulsegroups will ultimately get saved

plsdata.tbase: usually 1000, which means there will be 1000 awg time steps per unit of time used in the pulseing software (1 microsecond). This means the awg has 1ns resolution and we units of microseconds.

plsdata.pulses: is a struct array of templates for pulses. it has fields: data: struct array of elements from which to build the pulse (i.e. 'meas'). Each field has a type (e.g. '@wait'), time, and val.

The 'type' field will tell the software how to interpret the time and val fields to construct the pulse.

xval: not often used, but a flag for the pulse

taurc deprecated, used when bias-Ts were in place

name: human readable name

pardef: an array that describes which elements from pulse.data are controlled. The first element in each row signifies which element in pulse.data is controlled, and the second value designates what property is controlled. Negative values mean data(x).time will be set and positive mean data(x).val For example, pardef = [5 -1; 5 2; 6 3] means that we will control data(5).time(1), data(5).val(2), data(6).val(3)

trafofn: a function that transforms human readable parameters into the pardef. More precisely, it transforms the pulsegroup.params into the pardef (see below). For example, if pardef = [5 -1; 5 2; 6 3], the trafofn might be pinf.trafofn = @(x) [x(2)*1e-3, x(1),x(1)] which would mean that there are two human readable parameters: the first will get passed to data(5).val(2) and data(6).val(3), and the second will get multiplied by 1e-3 and passed to data(5).time(1).

format: a string that says which format the pulse is in. usually 'elem'

D.3 Defining a new pulse

Do not simply append a pulse to the end of plsdata.pulses. We use a function called plsreg to handle that (along with nice bookkeeping and error checking). plsreg is passed a struct with fields: data, pardef, trafofn,name, and is also passed a plsnum (which index in plsdata.pulses should be assigned the new pulse). the 'data' field is a struct array with fields 'type','time','val'. The syntax is

```
plsreg(pinf,plsnum)
```

See any of the pulsedef files for loads of examples.

Once a pulse template is made and is save in plsdata, we can construct a pulsegroup from it. A pulsegroup is s combination of pulses that have something in common and will usually be run together. We do not add a single pulse to the awg, but rather we make pulsegroups.

D.4 Dictionaries

It is often useful to have a dictionary of pulse elements that can be called upon to create a pulse. In fact, most of the time, this is all we do. For example, we might want a pulse.data.type to be generically a preparation, and we can decide how to prepare the qubit for different scenarios. If pulse.data.type starts with '@' it signifies that this element should be taken from a dictionary. When the pulse/pulsegroup is actually made, the dictionaries will be called upon to fill in the blanks. Each qubit can have its own dictionary. We load the dictionaries with the command pdload. For example:

```
r = pdload('right')
```

will load the dictionary for the right qubit into the variable r in memory. If changes are made, they can be saved with:

```
pdsave('right',r).
```

Do not fear, the history of the dictionaries is save. At the time when a group is made we will assign a pulse group a dictionary from which it can draw elements. The dictionary can be simply

```
group.dict ='right',
```

or it can be more complex if there are special elements in the pulse. In the example above, we might have

```
pulse.data.type ='@prep',
```

and at the time that the group is made we might say

```
group.dict = {'right',struct('prep','@adprep')}
```

which will generically assign all unknown elements from the 'right' dictionary, and will use the right dictionary's 'adprep' in place of the 'prep' field in the pulse.

D.5 Pulse Groups

Now it's time to make a pulse group. A simple pulsegroup is a struct with fields:

pg.name: a name that will also serve as the filename when the group is save pg.pulses: a number that says which number pulse from plsdata.pulses should be used as a template.

pg.xval: a parameter that can be used to signify something special about the pulsegroup. For example, if this group is to do a ramsey experiment at epsilon= 3, then pg.xval might = 3.

pg.params: the values that should get filled into the parameters of a given pulse (probably with the aid of pulse.trafofn. pg.dict: the dictionary from which pulse elements should be filled. It can be a string (e.g. 'right') or a cell array (see example above).

pg.chan: the channels (usually just 2) of the awg that the pulse will be sent to.

pg.varpar: a column vector of the parameter to be varied. If this is single n-element column, then n pulses will be created in the group by replacing the last pg.params with each of the pg.varpar. pg.varpar can also be a m x n column vector, which will still create an n pulse group, but replace the last m elements in params with the elements in pg.varpar.

pg.cntl: a string that tells the software how to interpret the pulse. it can be: 'loop': upon finishing a group the awg should return to the first pulse in the group 'pack': pack all the pulses in the group into one pulse for uploading. This speeds up awg load time but may break complex pulses.

Other controls can be: 'multi', 'grp', 'seq'. See below for a description of weird controls.

One a pulsegroup is defined, it is saved, much in the same way a pulse is saved by plsreg. A new group is created using

```
plsdefgroup(pg),
```

where pg is a struct with all of the fields above. This will simply save the metadata of the pulsegroup. It knows nothing about the awg. If at any time you want to change something in a group (e.g. pg.params, pg.varpar) then you use

```
plsupdate(pg)
```

There are some things, for example, the number of pulses in the group (determined by the size of pg.varpar) that cannot be changed, and plsupdate will throw an error. There is an inherent problem with this approach: A group is often made in a cell in the pulsedef file. If the cell is changed, the pulsegroup is usually forgotten about. This has lead to unneeded proliferation of pulsegroups.

D.6 Adding pulses to the AWG

Let's look at the fields in awgdata:

chans: a list of channels that the awg has. e.g. [1 2 3 4]

scale: the full amplitude (in mV) for each channel.

offset: an offset (in mV) applied to each channel

bits: the resolution of the awg

quiet: a flag used to suppress unnecessary output to the command line during waveform uploading.

slave: usually empty. This will signify whether two awgs are tethered together.

alternates: a struct of other awgdatas that can be swapped in and out. For example, the alternate might hold the awgdata with an awg5k and an awg7k tied together.

clk: usually 1e9. The clock frequency.

waveforms: a cell array of the names of the waveforms that are in memory. There are a lot because they are different for every channel, but only the names are stored here.

pulsegroups: a struct array of the pulsegroups that are in memory with lots of nice metadata.

zeropls: a list of all of the lengths of all of the different pulses. This is used to fill empty channels with zeros (if, e.g. you only want to pulse 2 of the 4 channels) in order to save memory.

In order to understand how we add pulses to the awg, one must understand the internals of the 5014C. Much of the software has been designed around it. It is a good idea to read through the programming manual.

The awg has two kinds of memory. waveform memory: memory used to store voltage values to output sequence memory: memory used to store information about sequences of pulses- in what order should the pulses be executed. The software will take care of both of these, but it's a good idea to understand how they work in case you want to change things:

Generically, we add pulses to the awg using

```
awgadd('name_of_some_group').
```

This will check if the group is in memory, and if not, it will make it, add the voltages to the waveform memory, and add information to the sequence memory. This will also store meta data about the pulsegroup in awgdata.pulsegroups. awgadd also works on cell arrays of names.

to remove a pulsegroup from the SEQUENCE memory use:

```
awgrm('some_name'), or awgrm(number),
```

where number referes to awgdata.pulsegroups(number). One should note that when removing by number (e.g. awgrm(12)), the software will remove everything from pulsegroup 12 and after, and then readd the necessary groups. This is why a much better idea would be to use

```
awgrm(12,'after')
```

which will not add back the groups after 12, or to use

```
awgrm('all')
```

which will remove all groups from memory. Note, that awgrm will NOT remove things from waveform memory. This makes operations like

```
awgrm('the_last_group'); awgadd('the_last_group');
```

very fast since the actual waveforms are not manipulated.

To manipulate waveform memory we use awgclear. Common uses are:

```
awgclear('unused');
```

which will clear the unused groups

```
awgclear('all');
```

which will clear all waveforms

```
awgclear('pack');
```

which will check which groups are unused or stale and readd them.

```
awgclear('pack','paranoid');
```

which will remove all waveforms and add them back. Check for no ways to speed up the process.

Other useful awgcommands: awggroups; print the awggroups that are in memory with some meta data atplschk(grps, inds, opts); a nice graphical way to check pulses

D.7 Future Pulsecontrol

Want a flexible, easy-to-use system for loading, using, and tracking awg pulses. It should be easy to prototype new ideas while still allowing for high-level use.

Particular problems with the current implementation:

- Ability to include different combos of pulses in groups without dirty hacks
  - different pulses in one group
  - multiple pulse lengths in one group
- Capable of working with multiple AWGs
- All information about the group is saved with the scan that runs it – no more plsinfo.
- No more need to retrieve pulses from a database or directory
- More human readable, to reduce stupid errors: plsdata, pardef and trafofn / grp params have clearer formatting / daq readout / string to describe group? / good plotting, good documentation of how each function is used, what it is capable of.

- Doesn't put information required for analysis in structs w/ visa objects (trafofns in plsdata, awgdata awg visa)
- Does anything get faster or slower doing this?

D.7.1 Implementation

The Pls package: has everything that is needed for creating and manipulating pulses and pulsegroups. This does not know about AWGs. The basic building block is a

```
pls.pls_elem
```

All pulse elements will inherit from this. All pulse elements (e.g.

```
pls.reload<pls.pls_elem
```

will need to overwite a

```
to_tab
```

method. This way, all elements know how to turn themselves into pulsetab. In turn, pulses (comprised of elems) turn themselves into pulsetabs, and in turn, groups (comprised of pulses) turn themselves into tabs. This implementation allows straightforward extensions by addition elements with complex behavior instead of having to change plstotab.

The various elements that inherit from

```
pls.pls_elem
```

For example: pls.meas will have properties:

```
time, start_delay,end_delay, measpt,
```

etc, and the other classes that inherit from

```
pls.pls_elem
```

properties. This will allow for pulses to reference these element properties by name. For example,

```
pulse.pardef(x).elem_num = 4; pulse.pardef(x).par = 'measpt'
```

Pulses are made of elems, groups are made of pulses. Pulses manipulate individual elems through the pardef, groups manipulate pulses through the varpar.

Having pls be a package (folder named

```
+pls
```

give nice namespace and tab completion properties.

Notes on package folders: http://www.mathworks.com/help/matlab/matlab_oop/scoping-classes-with-packages.html

Notes on heterogeneous arrays: http://www.mathworks.com/help/matlab/ref/matlab.mixin.heterogeneous-class.html

notes on copyable class http://www.mathworks.com/help/matlab/ref/matlab.mixin.copyable-class.html

AWG: The big change here is that groups are disposable. Groups will be made, uploaded to the AWGs, and stored in awgdata.memory. There will be an awgdata.queue where we put groups that we want in memory. Calling

```
awgdata.sync()
```

Everything needed will be stored in awgdata, and scans will save the relevant information so that there is never need to look up what groups used to look like. It is EXTREMELY rare in our current workflow to partially update groups without needed to remake and reload waveforms (only happens for changing feedback ratio), so we see no downside to this.

The user will interact with awgdata, not with awgs. This way, there can be nice ways of changing awg.scale, etc. emitting messages to repack wfs (or automatically repacking them). Similar things could be done with dictionary changes.

The complete design document can be found at: https://docs.google.com/document/d/1K_SszoH7TnVtuXcaFX6OEEwqc3Ifa9AIvTJVf2SyH00/edit and the code for the

+pls

found at https://github.com/mikeyshulman/pulsecontrol

E

Notes on Charge Noise

Charge noise in semiconductors has been an active area of resarch for decades. Here are some handy notes with some useful calculations regarding charge noise and its effect on singlet-triplet qubits. We will analyze the one-qubit and two qubit cases here.

E.1 Introduction

We begin with several assumptions. First, we assume and exponential model for $J(\epsilon)$ so that

$$\frac{\partial J}{\partial \epsilon} = \frac{-J}{\epsilon_0}$$

which will be the basis for many of the following calculations. We will also measure J (the exchange operation) in radians/sec, and therefore usually denote it as ω_J. Additionally, we must define noise power or noise spectral density. We define $\tilde{S}(f)$ as the noise power in $\frac{V^2}{Hz}$. We have

$$\tilde{S}(f) = 2S(\omega) = 2S(2\pi f)$$

where S is the noise power as defined in the Das Sarma paper of 2008[15].

We begin with the calculations for dephasing due to white noise in a spin echo sequence. We define the decoherence function χ, such that

$$W = e^{-\chi(t)}$$

where W is the echo amplitude. As per reference [15]. We define the white noise spectrum as

$$\tilde{S}_\nu = S_0$$

For pure dephasing from white noise we have

$$\chi = \frac{S_\phi t}{2}$$

Where S_ϕ is the phase noise power. We have $S_\phi = \frac{1}{2}\tilde{S_\nu}\left(\frac{\partial J}{\partial \epsilon}\right)^2 = \frac{1}{2}\tilde{S_\nu}\left(\frac{J}{\epsilon_0}\right)^2$. We therefore find

$$\chi = \frac{St}{2} = \frac{S_0}{4}\left(\frac{\omega_J}{\epsilon_0}\right)^2 t$$

From this we extract

$$T_2 \equiv \frac{2}{S_\phi} = \frac{4\epsilon_0^2}{\omega_J^2 \tilde{S_\nu}}$$

We define the period, $T \equiv \frac{2\pi}{\omega_J}$, and the quality factor of oscillations,

$$Q \equiv \frac{T_2}{T} = \frac{2\epsilon_0^2}{\pi \tilde{S_\nu} \omega_J}$$

Solving for the noise power we find

$$\tilde{S_\nu} = \frac{2\epsilon_0^2}{\pi Q \omega_J}$$

Plugging in realistic numbers, $Q = 400$, $J = 70MHz = 440\frac{Mrad}{sec}$, and $\epsilon_0 = 250\mu V$ we find

$$\tilde{S_\nu} = 2.7x10^{-19}\frac{V^2}{Hz}$$

$$\sqrt{\tilde{S_\nu}} = 4.8x10^{-10}\frac{V}{\sqrt{Hz}}$$

The AWG5014 has a white noise floor of $1^{nV}/\sqrt{Hz}$ at room temperature. Taking into account the 43dB of attenuation, the unterminated load (factor of 2) and the 2 uncorrelated channels that affect the qubit (factor of $\sqrt{2}$) we find that the AWG contributes $\sim\ 20^{pV}/\sqrt{Hz}$, which is significantly lower than the power necessary to explain the observed decay. This is not surprising given the temperature dependence results of chapter 5, which suggest that the noise is intrinsic to the device.

E.2 1/f NOISE WITH ECHO

Moving on to 1/f like noise we define the noise spectrum

$$\tilde{S}_\nu(f) = \frac{\tilde{S}_1}{f} = \frac{2\pi\tilde{S}_1}{\omega}$$

For this noise spectrum we have

$$\chi = \frac{S_\phi ln(2) t^2}{2\pi}$$

where

$$S_\phi = \frac{1}{2}\tilde{S}_\phi = \frac{1}{2}S_\nu(f)\left(\frac{\partial J}{\partial \epsilon}\right)^2 = \pi\tilde{S}_1\left(\frac{\omega_J}{\epsilon_0}\right)^2$$

where we have left off the factor of $\frac{1}{\omega}$ as it is implicit in the definition. We therefore have

$$\chi = \frac{\tilde{S}_1 ln(2)\omega_J^2 t^2}{2\epsilon_0^2}$$

defining T and T_2 the same way as above we find

$$T_2 = \frac{\epsilon_0}{\omega_J}\sqrt{\frac{2}{\tilde{S}_1 ln(2)}}$$

In this case, we solve for $\tilde{S}_1$ to find

$$\tilde{S}_1 = \frac{2\epsilon_0^2}{T_2^2\omega_J^2 ln(2)}$$

Experimentally, we have found that $T_2 \approx 7.5\mu s$. Using the same J and ϵ_0 we find

$$\tilde{S}_\nu = 1.7x10^{-14}\frac{V^2}{f(Hz)}$$

$$\sqrt{\tilde{S}_\nu(f)} = 130\frac{nV}{\sqrt{f(Hz)}}$$

E.3 Free Induction Decay

We now move on to a free induction decay, where we are dominated by 1/f-like charge noise, and we have the result

$$\chi = \frac{A_0^2 t^2}{2\pi}\ln(\frac{1}{\omega_{IR}t})$$

where ω_{IR} is the infarred cutoff frequency, above which the $1/f$ noise does not contribute. For the following calculations, we take it to be the inverse of the time over which the data is taken, whis is $\approx 5min$. We next assume a functional form for the noise power

$$S(\omega) = \frac{A_0^2}{\omega}$$

Using the same notation where

$$\tilde{S}_\phi = 2S(\omega) = 2S(2\pi f) = \frac{2A_0^2}{2\pi f} = \frac{A_0^2}{\pi f}$$

we obtain the measured noise power

$$\tilde{S}_\nu = \frac{A_0^2}{\pi f}\left(\frac{\partial J}{\partial \epsilon}\right)^{-2}$$

Using $\tilde{S}_\nu = \frac{\tilde{S}_1}{f}$ as the functional form of the noise power we obtain

$$A_0^2 = \pi\tilde{S}_1\left(\frac{\partial J}{\partial \epsilon}\right)^2$$

which gives

$$\chi = \frac{\tilde{S}_1 \omega_J^2 t^2}{2\epsilon_0^2} ln\left(\frac{1}{\omega_{IR} t}\right) = \frac{t^2}{T_2^2}$$

solving for $\tilde{S}_1$ we find

$$\tilde{S}_1 = \frac{2\epsilon_0^2}{\omega_J^2 T_2^2 ln\left(\frac{1}{\omega_{IR} t}\right)} = \frac{\epsilon_0^2}{2\pi^2 Q^2 ln\left(\frac{1}{\omega_{IR} t}\right)}$$

where $\tilde{S}_1$ is measured in $\frac{V^2}{f(Hz)}$, and $Q = {}^{\omega T_2}/_{2\pi}$ is the quality factor of the oscilaltions. Using $ln\left(\frac{1}{\omega_{IR} t}\right) \approx 21$ and $Q = 10$ we find

$$\sqrt{\tilde{S}_\nu(f)} = 1.2 \frac{\mu V}{\sqrt{f(Hz)}}$$

E.4 Entagling and Dephasing

We would like to express the full, two-qubit density matrix as a function of time given our standard, capacitively mediated two-qubit coupling. We begin by preparing both qubits in a superposition of $|S>$ and $|T_0>$ with a $\frac{\pi}{2}$ pulse around δB_Z. For now, however, we will be more general and assume that the initial state has the form

$$|\Psi_i> = (\alpha|S> + \beta|T>) \otimes (\gamma|S> + \delta|T>) \tag{E.1}$$

We assume that the Hamiltonian has the form

$$H_{2-qubit} = J_1 + J_2 + J_{12} \tag{E.2}$$

ie, we assume that states accumulate phase at a rate

$\|SS>$	0
$\|TS>$	J_1
$\|ST>$	J_2
$\|TT>$	J_{12}

Therefore, for example, upon evolving the qubit for a time T we have

$$|\Psi>=\alpha\gamma|SS>+\alpha\delta e^{i\int_0^T J_2 dt}|ST>+\beta\gamma e^{i\int_0^T J_1 dt}|TS>+\beta\delta e^{i\int_0^T J_1+J_2+J_{12}dt}|TT> \tag{E.3}$$

We must also keep track of the noise terms assosiated with J_1, J_2, and J_{12}. We therefore write $J_i = J_i + \delta J_i$, where i = 1,2,12 . Therefore, upon evolving the qubit for time T, applying a π-pulse which swaps S and T, and evolving for another time T, we have

$$\begin{aligned}|\Psi>= e^{i(J_1+J_2)T} \quad \Big\{ e^{iJ_{12}T}\Big(\alpha\gamma e^{i\int_T^{2T}\delta J_1+\delta J_2+\delta J_{12}}|TT>+\beta\delta e^{i\int_0^T \delta J_1+\delta J_2+\delta J_{12}}|SS>\Big)+ \\ \alpha\delta e^{i\int_0^T \delta J_2} e^{i\int_T^{2T}\delta J_1}|TS>+\beta\gamma e^{i\int_0^T \delta J_1} e^{i\int_T^{2T}\delta J2_2}|ST>\Big\}\end{aligned}$$

We now substitute in for $\alpha, \gamma = \frac{1}{\sqrt{2}}$, and $\beta, \delta = \frac{i}{\sqrt{2}}$, and we write the full density matrix as

$$\rho = \frac{1}{4}\begin{pmatrix} 1 & -ie^{iJ_{12}T}\eta_{12}^1\chi_2 & -ie^{iJ_{12}T}\eta_{12}^1\chi_1 & -\chi_1\chi_2\chi_{12} \\ & 1 & \chi_1\chi_2 & -ie^{-iJ_{12}T}\eta_{12}^2\chi_1 \\ & & 1 & -ie^{-iJ_{12}T}\eta_{12}^2\chi_2 \\ & & & 1 \end{pmatrix} \tag{E.4}$$

where

$$\chi_i =< e^{i\int_0^T \delta J_i - i\int_T^{2T}\delta J_i} > \tag{E.5}$$

which is a spin echo filter function,

$$\eta_i^1 =< e^{i\int_0^T \delta J_i} >,\ \eta_i^2 =< e^{i\int_T^{2T} \delta J_i} > \tag{E.6}$$

which are free-induction decay filter functions. As the density matrix above stands, it is correct, only if the noises on J_1, J_2, and J_{12} are assumed to be uncorrelated. If we assume that the noises are correlated, then we must change terms like $\chi_1\eta_{12}^1 \equiv< e^{i\int_0^T \delta J_{12}+\int_0^T \delta J_1-\int_T^{2T} \delta J_1} >$, because the terms are now correlated.

We proceed with a simple example: In simple models we assume $J_{12} = MJ_1J_2$ where M is a constant. Therefore we proceed by assuming

$$\delta J_{12} = M\left(J_1\delta J_2 + J_2\delta J_1\right) = \left(\frac{J_{12}}{J_1}\delta J_1\right) + \left(\frac{J_{12}}{J_2}\delta J_2\right) \tag{E.7}$$

We therefore find

$$\chi_1\eta_{12}^1 \equiv< e^{i\int_0^T \delta J_{12}+\int_0^T \delta J_1-\int_T^{2T} \delta J_1} >= \tag{E.8}$$

$$< e^{\int_0^T \left(1+\frac{J_{12}}{J_1}\right)\delta J_1-\int_T^{2T} \delta J_1+\int_0^T \left(1+\frac{J_{12}}{J_2}\right)\delta J_2-\int_T^{2T} \delta J_2} > \tag{E.9}$$

We may rewrite this as

$$\chi_1\eta_{12}^1 = \eta_{M_1}^1\eta_{M_2}^1\chi_1 \tag{E.10}$$

where $\eta_{M_1}^1 =< e^{i\int_0^T \frac{J_{12}}{J_1}\delta J_1} >$. The other term that we must evaluate is

$$\chi_1\chi_2\chi_{12} \equiv< e^{i\int_0^T \delta J_1+\delta J_2+\delta J_{12}-i\int_T^{2T} \delta J_1+\delta J_2+\delta J_{12}} > \tag{E.11}$$

These integrals are easily rescalled by $T \to T\left(1+\frac{J_{12}}{J_1}\right)$ or $T \to T\left(1+\frac{J_{12}}{J_2}\right)$ to give

$$\chi_1(T)\chi_2(T)\chi_{12}(T) = \chi_1\left(T\left(1+\frac{J_{12}}{J_1}\right)\right)\chi_2\left(T\left(1+\frac{J_{12}}{J_2}\right)\right) \tag{E.12}$$

Under these assumptions we may rewrite the density matrix as

$$\rho = \frac{1}{4}\begin{pmatrix} 1 & -ie^{iJ_{12}T}\eta^1_{M_1}\eta^1_{M_2}\chi_2 & -ie^{iJ_{12}T}\eta^1_{M_1}\eta^1_{M_2}\chi_1 & -\chi_1\left(T\left(1+\frac{J_{12}}{J_1}\right)\right)\chi_2\left(T\left(1+\frac{J_{12}}{J_2}\right)\right) \\ & 1 & \chi_1\chi_2 & -ie^{-iJ_{12}T}\eta^2_{M_1}\eta^2_{M_2}\chi_1 \\ & & 1 & -ie^{-iJ_{12}T}\eta^2_{M_1}\eta^2_{M_2}\chi_2 \\ & & & 1 \end{pmatrix} \tag{E.13}$$

We may further assume that the FID time for J_{12} is the same before and after the echo pulse, such that

$$\eta^2_{M_i} = \eta^1_{M_i} = \eta_{M_i}$$

Which gives the density matrix a slightly simpler form

$$\rho = \frac{1}{4}\begin{pmatrix} 1 & -ie^{iJ_{12}T}\eta_{M_1}\eta_{M_2}\chi_2 & -ie^{iJ_{12}T}\eta_{M_1}\eta_{M_2}\chi_1 & -\chi_1\left(T\left(1+\frac{J_{12}}{J_1}\right)\right)\chi_2\left(T\left(1+\frac{J_{12}}{J_2}\right)\right) \\ & 1 & \chi_1\chi_2 & -ie^{-iJ_{12}T}\eta_{M_1}\eta_{M_2}\chi_1 \\ & & 1 & -ie^{-iJ_{12}T}\eta_{M_1}\eta_{M_2}\chi_2 \\ & & & 1 \end{pmatrix} \tag{E.14}$$

Finally, we compute the Bell state fidelity as

$$F = \frac{1}{4}\left\{1+\frac{1}{2}\left(\chi_1\chi_2+\chi_1\left(T(1+\frac{J_{12}}{J_1})\right)\chi_2\left(T(1+\frac{J_{12}}{J_2})\right)\right)+\eta_{M_1}\eta_{M_2}sin\left(J_{12}T\right)\left(\chi_1+\chi_2\right)\right\} \tag{E.15}$$

One should note, that technically, all decay functions χ, η should be evalutated at time 2T, because

the qubit was evolved for a time 2T. The non-zero elements of the Pauli set are

$$< YI >= -\eta_{M_1}\eta_{M_2}\chi_1 cos\left(J_{12}T\right)$$

$$< YI >= -\eta_{M_1}\eta_{M_2}\chi_2 cos\left(J_{12}T\right)$$

$$< XZ >= \eta_{M_1}\eta_{M_2}\chi_1 sin\left(J_{12}T\right)$$

$$< XZ >= \eta_{M_1}\eta_{M_2}\chi_2 sin\left(J_{12}T\right)$$

$$< XX >= \frac{1}{2}\left(\chi_1\chi_2 - \chi_1\left(T(1+\frac{J_{12}}{J_1})\right)\chi_2\left(T(1+\frac{J_{12}}{J_2})\right)\right)$$

$$< YY >= \frac{1}{2}\left(\chi_1\chi_2 + \chi\left(T(1+\frac{J_{12}}{J_1})\right)\chi_2\left(T(1+\frac{J_{12}}{J_2})\right)\right)$$

References

[1] Balasubramanian, G., Neumann, P., Twitchen, D., Markham, M., Kolesov, R., amd J. Isoya, N. M., Achard, J., Beck, J., Tissler, J., Jacques, V., Hemmer, P. R., Jelezko, F., and Wrachtrup, J. (2009). Ultralong spin coherence time in isotopically engineered diamond. *Nature Materials*, 8:383–387.

[2] Barthel, C. (2010). *Control and fast Measurement of Spin Qubits*. PhD thesis, Harvard University.

[3] Barthel, C., Medford, J., Marcus, C. M., Hanson, M. P., and Gossard, A. C. (2010). Interlaced dynamical decoupling and coherent operation of a singlet-triplet qubit. *Phys. Rev. Lett.*, 105:266808.

[4] Barthel, C., Reilly, D. J., Marcus, C. M., Hanson, M. P., and Gossard, A. C. (2009). Rapid single-shot measurement of a singlet-triplet qubit. *Phys. Rev. Lett.*, 103:160503.

[5] Bennet, C. H., Brassard, G., Popescu, S., Schumacher, B., Smolin, J. A., and Wooters, W. K. (1996). Purification of noisy entanglement and faithful teleportation via noisy channels. *Phys. Rev. Lett.*, 76:722.

[6] Bluhm, H., Foletti, S., Mahalu, D., Umansky, V., and Yacoby, A. (2010). Enhancing the coherence of a spin qubit by operating it as a feedback loop that controls its nuclear spin bath. *Phys. Rev. Lett.*, 105:216803.

[7] Bluhm, H., Foletti, S., Neder, I., Rudner, M. S., Mahalu, D., Umansky, V., and Yacoby, A. (2011). Dephasing time of GaAs electron-spin qubits coupled to a nuclear bath exceeding 200 μs. *Nature Physics*, 7:109.

[8] Brunner, R., Shin, Y.-S., Obata, T., Pioro-Ladrière, M., Kubo, T., Yoshida, K., Taniyama, T., Tokura, Y., and Tarucha, S. (2011). Two-qubit gate of combined single-spin rotation and interdot spin exchange in a double quantum dot. *Phys. Rev. Lett.*, 107:146801.

[9] Carr, H. Y. and Purcell, E. M. (1954). Effects of diffusion on free precession in nuclear magnetic resonance experiments. *Phys. Rev.*, 94:630.

[10] Chernobrod, B. M. and Brerman, G. P. (2005). Spin microscope based on optically detected magnetic resonance. *Journal of Applied Physics*, 97:014903.

[11] Chow, J. M., DiCarlo, L., Gambetta, J. M., Nunnenkamp, A., Bishop, L. S., Frunzio, L., Devoret, M. H., Girvin, S. M., and Schoelkopf, R. J. (2010). Detecting highly entangled states with a joint qubit readout. *Phys. Rev. A*, 81:062325.

[12] Churchill, H. O. H., Kuemmeth, F., Harlow, J. W., Bestwick, A. J., Rashba, E. I., Flensberg, K., Stwertka, C. H., Taychatanapat, T., Watson, S. K., and Marcus, C. M. (2009). Relaxation and dephasing in a two-electron ^{13}C nanotube double quantum dot. *Phys. Rev. Lett.*, 102:166802.

[13] Coish, W. A. and Loss, D. (2004). Hyperfine interaction in a quantum dot: Non-markovian electron spin dynamics. *Phys. Rev. B*, 70:195340.

[14] Culcer, D., Hu, X., and Das Sarma, S. (2009). Dephasing of Si spin quits due to charge noise. *Appl. Phys. Lett*, 95:073102.

[15] Cywinski, L., Lutchyn, R. M., Nave, C. P., and Das Sarma, S. (2008). How to enhance dephasing time in superconducting qubits. *Phys. Rev. B*, 77:174509.

[16] Devoret, M. H. and Schoelkopf, R. J. (2000). Amplifying quantum signals with the single-electron-transistor. *Nature*, 406:1039–1046.

[17] Dial, O. E., Shulman, M. D., Harvey, S. P., Bluhm, H., Umansky, V., and Yacoby, A. (2013). Charge noise spectroscopy using coherent exchange oscillations in a singlet-triplet qubit. *Phys. Rev. Lett.*, 110:146804.

[18] DiVincenzo, D. (2000). The physical implementation of quantum computation. *Fortschr. Phys.*, 48:771.

[19] Dobrovitski, V., de Lange, G., Ristè, D., and Hanson, R. (2010). Bootstrap tomography of the pulses for quantum control. *Phys. Rev. Lett.*, 105:077601.

[20] Ferrie, C., Granade, C. E., and Cory, D. G. (2013). How to best sample a periodic probability distribution, or on the accuracy of hamiltonian finding strategies. *Quantum Information Processing*, 12:611.

[21] Feynman, R. P. (1982). Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467.

[22] Foletti, S. (2010). *Manipulation and coherent control of a two-electron logical spin qubit using GaAs double quantum dots*. PhD thesis, Harvard University.

[23] Foletti, S., Bluhm, H., Mahalu, D., Umansky, V., and Yacoby, A. (2009). Universal quantum control in two-electron spin quantum bits using dynamic nuclear polarization. *Nature Physics*, 5:903.

[24] Gaudreau, L., Granger, G., Kam, A., Aers, G. C., andP. Zawadzki, S. A. S., Pioro-Ladrière, M., Wasilewski, Z. R., and Sachrajda, A. S. (2011). Coherent control of three-spin states in a triple quantum dot. *Nature Physics*, 8:54–58.

[25] Giovannetti, V., Lloyd, S., and Maccone, L. (2011). Advances in quantum metrology. *Nature Photonics*, 5:222–229.

[26] Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *Proceedings of 28th Annual ACM Symposium on Theory of Computing (STOC)*, page 212.

[27] Gullans, M., Krich, J. J., Taylor, J. M., Bluhm, H., Halperin, B. I., Marcus, C. M., Stopa, M., Yacoby, A., and Lukin, M. D. (2010). Dynamic nuclear polarization in double quantum dots. arXiv:1003.4508.

[28] Hahn, E. L. (1950). Spin echoes. *Phys. Rev.*, 80:580.

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

[30] Hill, S. and Wootters, W. K. (1997). Entanglement of a pair of quantum bits. *Phys. Rev. Lett.*, 78:5022–5025.

[31] James, D. F. V., Kwiat, P. G., Munro, W. J., and White, A. G. (2001). On the measurement of qubits. *Phys. Rev. A*, 64:052312.

[32] Kestner, J. P., Wang, X., Bishop, L. S., Barnes, E., and Das Sarma, S. (2013). Noise-resistant control for a spin qubit array. *Phys. Rev. Lett.*, 110:140502.

[33] Klauser, D., Coish, W. A., and Loss, D. (2006). Nuclear spin state narrowing via gate-controlled Rabi oscillations in a double quantum dot. *Phys. Rev. B*, 73:205302.

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

[35] Laird, E. A., Taylor, J. M., DiVincenzo, D. P., Marcus, C. M., Hanson, M. P., and Gossard, A. C. (2010). Coherent spin manipulation in an exchange-only qubit. *Phys. Rev. B*, 82:075403.

[36] Leibfried, D., Demarco, B., Meyer, V., Lucas, D., Barrett, M., Britton, J., Itano, W. M., Jelenkovic, B., Langer, C., Rosenband, T., and Wineland, D. J. (2003). Experimental demonstration of a robust, high-fidelity geometric two ion-qubit phase gate. *Nature*, 422:412.

[37] Levy, J. (2002). Universal quantum computation with spin-1/2 pairs and Heisenberg exchange. *Phys. Rev. Lett.*, 89:147902.

[38] Loss, D. and DiVincenzo, D. P. (1998). Quantum computation with quantum dots. *Phys. Rev. A*, 57:120.

[39] Maletinsky, P., Hong, S., Grinolds, M. S., Hausmann, B., Lukin, M. D., Walsworth, R. L., Loncar, M., and Yacoby, A. (2012). A robust scanning diamond sensor for nanoscale imaging with single nitrogen-vacancy centres. *Nature Nanotechnology*, 7:320–324.

[40] Maune, B. M., Borselli, M. G., Huang, B., Ladd, T. D., Deelman, P. W., Holabird, K. S., Kiselev, A. A., Alvarado-Rodriguez, I., Ross, R. S., Schmitz, A. E., Sokolich, M., Watson, C. A., Gyure, M. F., and Hunter, A. T. (2011). Coherent singlet-triplet oscillations in a silicon-based double quantum dot. *Nature*, 481:344–347.

[41] Maze, J. R., Stanwix, P. L., Hodges, J. S., Hong, S., Taylor, J. M., Cappellaro, P., Jian, L., Dutt, M. V. G., Togan, E., Zibrov, A. S., Yacoby, A., Walsworth, R. L., and Lukin, M. D. (2008). Nanoscale magnetic sensing with an individual electronic spin in diamond. *Nature*, 455:644–647.

[42] Medford, J., Cywiński, L., Barthel, C., Marcus, C. M., Hanson, M. P., and Gossard, A. C. (2012). Scaling of dynamical decoupling for spin qubits. *Phys. Rev. Lett.*, 108:086802.

[43] Muhonen, J. T., Dehollain, J. P., Laucht, A., Hudson, F. E., Sekiguchi, T., Itoh, K. M., Jamieson, D. N., McCallum, J. C., Dzurak, A. S., and Morello, A. (2014). Storing quantum information for 30 seconds in a nanoelectronic device. preprint at http://arxiv.org/abs/1402.7140.

[44] Nadj-Perfe, S., Frolov, S. M., Bkkers, E. P. A. M., and Kouwenhoven, L. P. (2010). Spin-orbit qubit in a semiconductor nanowire. *Nature*, 468:1084.

[45] Nielsen, M. and Chuang, I. (2000). *Quantum Computation and Quantum Information*. Cambridge University Press.

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

[47] Nowack, K. C., Shafiei, M., Laforest, M., Prawiroatmodjo, G. E. D. K., Schreiber, L. R., Reichl, C., Wegscheider, W., and Vandersypen, L. M. K. (2011). Single-shot correlations and two-qubit gate of solid-state spins. *Science*, 333:1269.

[48] Nusran, N. M., Ummal, M., and Dutt, M. G. (2012). High dynamic range magnetometry with a single electronic spin in diamon. *Nature Nanotechnology*, 7:109.

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

[50] Pioro-Ladriere, M., Obata, T., Tokura, Y., Shin, Y. S., Kubo, T., Yoshida, K., Taniyama, T., and Tarucha, S. (2008). Electrically driven single electron spin resonance in a slanting Zeeman field. *Nature Physics*, 4:776.

[51] Pioro-Ladriere, M., Tokura, Y., Kubo, T., and Tarucha, S. (2007). Micromagnets for coherent control of spin-charge qubit in lateral quantum dots. *Appl. Phys. Lett.*, 90:024105.

[52] Reed, M., DiCarlo, L., Nigg, S., Sun, L., Frunzio, L., Girvin, S. M., and Schoelkopf, R. J. (2011). Realization of a three-qubit quantum error correction with superconducting circuits. *Nature*, 482:382.

[53] Reilly, D. J., Marcus, C. M., Hanson, M. P., and Gossard, A. C. (2007). Fast single-charge sensing with a RF quantum point contact. *Appl. Phys. Lett.*, 91:162101.

[54] Sackett, C. A., Kielpinski, D., King, B. E., Langer, C., Meyer, V., Myatt, C. J., Rowe, M., Turchette, Q. A., Itano, W. M., Wineland, D. J., and Monroe, C. (2000). Experimental entanglement of four particles. *Nature*, 404:6775.

[55] Schroer, M. D. and Petta, J. R. (2008). Quantum dots: time to get the nukes out. *Nature Physics.*, 4:1745.

[56] Sergeevich, A., Chandran, A., Combes, J., Bartlett, S. D., and Wiseman, H. M. (2011). Characterization of a qubit hamiltonian using adaptive measurements in a fixed basis. *Phys. Rev. A*, 84:052315.

[57] Shor, P. W. (1994). Algorithms for quantum computation: Discrete log and factoring. *Proceedings of the 25th Annual Symposium on Foundations of Computer Science*.

[58] Shulman, M. D., Dial, O. E., Harvey, S. P., Bluhm, H., Umansky, V., and Yacoby, A. (2012). Demonstration of entanglement of electrostatically coupled singlet-triplet qubits. *Science*, 336:202.

[59] Shulman, M. D., Harvey, S. P., Nichol, J. M., Bartlett, S. D., Doherty, A. C., Umansky, V., and Yacoby, A. (2014). Suppressing qubit dephasing using real-time Hamiltonian estimation. *Nat Commun*, 5:5156.

[60] Taminiau, T. H., Cramer, J., der Sar, T. V., Dobrovitski, V. V., and Hanson, R. (2014). Universal control and error correction in multi-qubit spin registers in diamond. *Nature Nanotechnology*, 9:171–176.

[61] Taylor, J. M., Engel, H.-A., Dür, W., Yacoby, A., Marcus, C. M., Zoller, P., and Lukin, M. D. (2005). Fault-tolerant architecture for quantum computation using electrically controlled semiconductor spins. *Nature Physics*, 1:177.

[62] Taylor, J. M., Petta, J. R., Johnson, A. C., Yacoby, A., Marcus, C. M., and Lukin, M. D. (2007). Relaxation, dephasing, and quantum control of electron spins in double quantum dots. *Phys. Rev. B*, 76:035315.

[63] Trifunovic, L., Dial, O., Trif, M., Wootton, J., Abebe, R., Yacoby, A., and Loss, D. (2012). Long-distance spin-spin coupling via floating gates. *Phys. Rev. X*, 2:011006.

[64] Uhrig, G. S. (2007). Keeping a quantum bit alive by optimizing π-pulse sequences. *Phys. Rev. Lett.*, 98:100504.

[65] Uhrig, G. S. (2008). Exact results on dynamical decoupling by *pi* pulses in quantum inforamtion processes. *New J. Phys.*, 10:083024.

[66] van Weperen, I., Armstrong, B. D., Laird, E. A., Medford, J., Marcus, C. M., Hanson, M. P., and Gossard, A. C. (2011). Charge-state conditional operation of a spin qubit. *Phys. Rev. Lett.*, 107:030506.

[67] Viola, L., Knill, E., and Lloyd, S. (1999). Dynamical decoupling of open quantum systems. *Phys. Rev. Lett.*, 82:2417.

[68] Waldherr, G., Beck, J., Neumann, P., Said, R. S., Nitsche, M., Markham, M. L., Twitchen, D. J., Twamley, J., Jelezko, F., and Wrachtrup, J. (2012). High-dynamic-range magnetometry with a single nuclear spin in diamon. *Nature Nanotechnology*, 7:105–108.

[69] Waldherr, G., Wang, Y., Zaiser, S., Jamali, M., Schulte-Herbruggen, T., Abe, H., Ohshima, T., and Wrachtrup, J. (2014). Quantum error correction in a solid-state hybrid spin register. *Nature*, 506:204–207.

[70] Wild, A., Kierig, J., Sailer, J., Ager, J. W., Haller, E. E., Abstreiter, G., Ludwig, S., and Bougeard, D. (2012). Few electron double quantum dot in an isotopically purified 28Si quantum wel. *Appl. Phys. Lett*, 100:143110.

[71] Wiseman, H. M. and Milburn, G. J. (2010). *Quantum Measurement and Control*. Cambridge University Press.

[72] Yu, T. and Eberly, J. (2006). Sudden death of entanglment: Classical noise effects. *Opt. Commun*, 264:393.

This thesis was typeset using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. The body text is set in 11 point Egenolff-Berner Garamond, a revival of Claude Garamont's humanist typeface. The above illustration, "Science Experiment 02", was created by Ben Schlitter and released under CC BY-NC-ND 3.0. A template that can be used to format a PhD thesis with this look and feel has been released under the permissive MIT (XII) license, and can be found online at github.com/suchow/Dissertate or from its author, Jordan Suchow, at suchow@post.harvard.edu.