

Datapath and Control for Quantum Wires

NEMANJA ISAILOVIC, MARK WHITNEY, YATISH PATEL,
and JOHN KUBIATOWICZ

University of California, Berkeley

DEAN COPSEY and FREDERIC T. CHONG

University of California, Davis

ISAAC L. CHUANG

Massachusetts Institute of Technology

and

MARK OSKIN

University of Washington

As quantum computing moves closer to reality the need for basic architectural studies becomes more pressing. Quantum wires, which transport quantum data, will be a fundamental component in all anticipated silicon quantum architectures. Since they cannot consist of a stream of electrons, as in the classical case, quantum wires must fundamentally be designed differently. In this paper, we present two quantum wire designs: a swap wire, based on swapping of adjacent qubits, and a teleportation wire, based on the quantum teleportation primitive. We characterize the latency and bandwidth of these two alternatives in a device-independent way. Furthermore, unlike classical wires, quantum wires need control signals in order to operate. We explore the complexity of the control mechanisms and the fundamental tension between the scale of quantum effects and the scale of the classical logic needed to control them. This “pitch-matching” problem imposes constraints on minimum wire lengths and wire intersections, leading us to use a SIMD approach for the control mechanisms. We ultimately show that qubit decoherence imposes a basic limit on the maximum communication distance of the swapping wire, while relatively large overhead imposes a basic limit on the minimum communication distance of the teleportation wire.

Categories and Subject Descriptors: B.4.3 [**Input/Output and Data Communications**]: Interconnections (Subsystems)

General Terms: Design, Reliability

Additional Key Words and Phrases: Architecture, Control, Layout

Authors' addresses: N. Isailovic, M. Whitney, Y. Patel, and J. Kubiatowicz, Computer Science Division, University of California, Berkeley, 387 Soda Hall, Berkeley, CA 94720-1776; email: nemanja@cs.berkeley.edu; D. Copsey and F. T. Chong, Department of Computer Science, University of California, Davis, Davis, CA; I. L. Chuang, Department of Physics, Massachusetts Institute of Technology, Cambridge, MA; M. Oskin, Department of Computer Science, University of Washington, Seattle, WA.

Permission to make digital/hard copy of all or part of this material without fee for personal or classroom use provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.

© 2004 ACM 1544-3566/04/0300-0034 \$5.00

1. INTRODUCTION

Many important problems seem to require exponential resources on a classical computer. Quantum computers can solve some of these problems with polynomial resources, leading a great number of researchers to explore quantum information processing technologies [Bennett and DiVincenzo 2000; DiVincenzo 1995; Gershenfeld and Chuang 1998; Goodkind 2000; Lloyd 1995; Nielsen and Chuang 2000; Shor 1997; Skinner et al. 2002]. Early-stage quantum computers have used a small number of components (less than ten) and have utilized molecules in solution and trapped ions [Knill et al. 1999; Vandersypen et al. 2000a; Sackett et al. 2000]. To exploit our tremendous historical investment in silicon, however, solid-state silicon quantum computers are desirable. Promising proposals along these lines have begun to appear [Kane 1998; Vion et al. 2002]. However, as the number of components grows, quantum computing systems will begin to require the same level of engineering as current computing systems.

This paper is about a seemingly mundane subject: a wire. To be clear, we define a wire in the quantum world as a mechanism for moving quantum data from one spatial location to another. Any optimistic view of the future of quantum computing includes enough interacting devices to introduce a spatial extent to the layout of those devices. This spatial dimension introduces a need for wires. As we will show, a quantum wire is a very different creature from a classical one. One of the most important distinctions between quantum and classical wires arises from the fact that quantum information *cannot be copied* [Nielsen and Chuang 2000]. Instead, it must be *transported*—destroying the information at the source and re-creating it at the destination. This no-cloning theorem is discussed further in Section 2.3.

Quantum information can be encoded in a number of ways, such as the spin component of basic particles such as protons or electrons, or in the polarization of photons. Thus, there are multiple ways in which we might transfer information. One option is to physically transport particles from one point to another, using some form of ballistic transport. Unfortunately, this causes several problems (as discussed in Section 2.3). Another option is to pass information along a line of quantum devices. This *swap wire* is, in fact, a viable option for short distances (as discussed in Section 4), but tends to accumulate errors over long distances.

Over longer distances, we need something fundamentally different. We propose to use a technique called *teleportation* [Bennett et al. 1993] and to call the resulting long-distance quantum wire a *teleportation wire* to distinguish it from a swap wire. Teleportation uses an unusual quantum property called *entanglement*, which allows quantum bits to interact instantaneously at a distance.¹

For both wire designs, classical control circuitry must be used to coordinate the many operations necessary to perform a single basic transport step (swap or teleportation). For any scalable technology, classical control must be

¹Although this property sounds suspiciously like as “faster-than-light” communication, we shall see that the interaction is ambiguous without the additional transmission of two bits of classical information, which must travel at a subluminal velocity.

integrated on the same substrate as the quantum datapath. This somewhat obvious statement leads to two immediate difficulties. First, many proposals for quantum computing involve temperatures close to absolute zero. As a result, control circuits must operate at the same temperature, effectively ruling out CMOS transistors (since the impurity carriers become “frozen out” near absolute zero). Bipolar transistors are a possible choice, but the constant current that is on during operation poses huge problems with heat dissipation, particularly at the extremely low temperature that must be maintained. Since *single-electron transistors* (SETs) work optimally at these low temperatures, we explore them here, with the caveat that more work needs to be done to determine the best choice for the technology used in the control circuitry. Second, control circuitry must somehow match the pitch of the quantum datapath to allow scalable control. In both wire designs, the drive transistors closest to the datapath place constraints on the pitch of the quantum bits.

The remainder of this paper is organized as follows. We present a brief introduction to quantum computation as well as to methods of qubit transport in Section 2. We then provide a detailed discussion of the aspects of the Kane model [Skinner et al. 2002], the technology model we have chosen to use, in Section 3. Section 4 presents the datapath and control for a swap wire, while Section 5 presents the same for a teleportation wire. Section 6 concludes.

2. QUANTUM COMPUTING

We begin with a brief overview of the basic terminology of quantum computation. Our purpose is to introduce the language necessary for this paper; in-depth treatments of these subjects are available in the literature [Nielsen and Chuang 2000].

2.1 Quantum States: Qubits

The state of a classical digital system \mathcal{X} can be specified by a binary string \mathbf{x} composed of a number of bits x_i , each of which uniquely characterizes one elementary piece of the system. For n bits, there are 2^n unique possible states. The state of an analogous quantum system ψ is described by a complex-valued vector $|\psi\rangle = \sum_{\mathbf{x}} c_{\mathbf{x}}|\mathbf{x}\rangle$, a weighted combination (a “superposition”) of the basis vectors $|\mathbf{x}\rangle$, where the *probability amplitudes* $c_{\mathbf{x}}$ are complex numbers whose modulus squared sums to one, that is $\sum_{\mathbf{x}} |c_{\mathbf{x}}|^2 = 1$.

A single quantum bit is commonly referred to as a *qubit* and is described by the equation $|\psi\rangle = c_0|0\rangle + c_1|1\rangle$. Legal qubit states include pure states, such as $|0\rangle$ and $|1\rangle$, and states in superposition, such as $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. Also valid are $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ and $\frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$, which are other equal superpositions.

Larger quantum systems can be composed from multiple qubits. For example, $|00\rangle$ is a valid two-qubit state, as is $\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle - \frac{1}{\sqrt{2}}|11\rangle$. An n -qubit state is described by 2^n basis vectors, each with its own complex probability amplitude, so an n -qubit system can exist in an arbitrary superposition of the 2^n classical states.

Unlike the classical case, however, where the total can be completely characterized by its parts, the state of larger quantum systems cannot always be

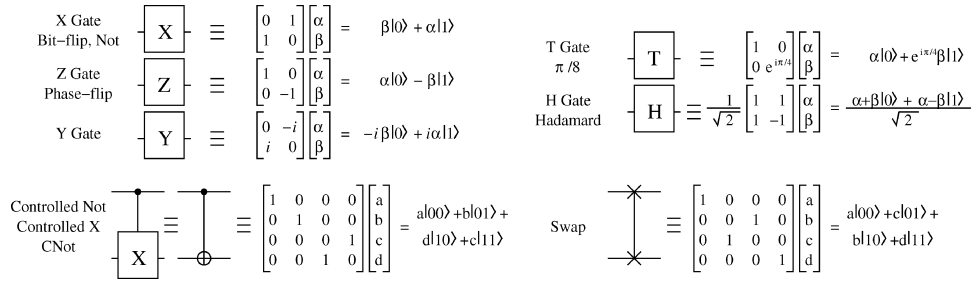


Fig. 1. Basic quantum gates and their matrix representations.

described simply by giving the individual states of its component qubits. This property, known as *entanglement*, is best illustrated with an example: there exist no single qubit states $|\psi_A\rangle$ and $|\psi_B\rangle$ such that the two-qubit state $|\Psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ can be expressed as a composite of $|\psi_A\rangle$ and $|\psi_B\rangle$. Entanglement does not exist classically, and it lies at the heart of many quantum operations.

Another nonintuitive property of quantum states is their behavior when measured. Upon observation, a quantum state collapses into one of a number of possible classical states, the set of possibilities being determined by the measurement apparatus. Specifically, it is conventional to adopt the *computational basis* states $|0\dots 00\rangle, |0\dots 01\rangle, |0\dots 10\rangle, \dots, |1\dots 11\rangle$, and choose measurements to collapse states into this basis. The probability that a particular basis state \mathbf{x} results is $|c_x|^2$, the modulus square of the probability amplitude for the basis vector \mathbf{x} . For example, when $\frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$ is measured, the outcome is $|0\rangle$ or $|1\rangle$ with equal probability. Similarly, when the state $|\Psi\rangle$, above, is measured, the result is either $|00\rangle$ or $|11\rangle$, with equal probability; the outcomes $|01\rangle$ or $|10\rangle$ never occur.

2.2 Quantum Gates and Circuits

Just as bits can be flipped using a NOT gate and interact with each other via multi-bit logic gates such as the XOR, qubits can be operated on by gates such as those shown in Figure 1. In the quantum realm, the role of the classical truth table is played by a unitary operator U . The output state vector is the operator applied to the input vector; that is, $|\psi_{\text{out}}\rangle = U|\psi_{\text{in}}\rangle$. The X gate is analogous to the classical NOT gate: it flips $|0\rangle$ and $|1\rangle$. The Z gate flips the phase of the $|1\rangle$ state, thus exchanging $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. The Hadamard gate H is another unusual single-qubit gate: it turns $|0\rangle$ into $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|1\rangle$ into $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$; it can be thought of as performing a radix-2 Fourier transform. And analogous to the classical XOR gate is the quantum controlled-NOT (or CNOT) gate.

One more important operator is the SWAP gate. SWAP can be implemented using three CNOTS. However, SWAP is often available as a basic gate for a given technology, which is a valuable thing, given its importance to quantum communication.

In quantum circuits, single lines represent qubits, and double lines represent classical bits. A meter is used to represent measurement. By convention, black

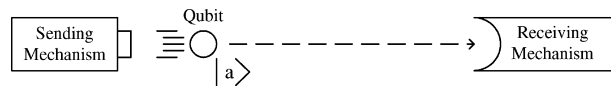


Fig. 2. An abstract view of ballistic transport of qubits. The qubit is “launched” from the source and “caught” at the destination. This approach turns out to be unrealistic in many technologies.

dots represent control terminals for quantum-controlled gates. The \oplus symbol is shorthand for the target qubit of the CNOT gate.

2.3 Quantum Transport

In a classical computer, data are continuously moved, copied, operated upon, and destroyed. The output from a gate may be fanned out to act as input to multiple other gates, may be transported some distance to a memory cell, and so on. Likewise, at the quantum level, qubits need to be transported from place to place in order to be operated upon, and thus quantum wires are a necessary basic component of any quantum architecture. However, due to fundamental laws of quantum physics, qubit transport must be handled very differently from bit transport.

First of all, data in a classical computer consist of large groups of electrons or holes signifying high or low voltage. Data in a quantum computer consist of individual qubits. Thus, classical wires, being just long, thin strips of metal, cannot be used in the quantum domain. More precise means of transporting volatile qubits must be devised. Secondly, the no-cloning theorem [Nielsen and Chuang 2000] states that it is impossible to make an exact duplicate of an arbitrary qubit $a|0\rangle + b|1\rangle$. Note that this does not preclude the possibility of copying a qubit in a known state. For example, a qubit known to be in the state $|0\rangle$ ($a = 1, b = 0$) can be copied at will. However, a qubit in an arbitrary unknown state cannot be copied. The no-cloning theorem has severe consequences on quantum transport. Each qubit must be moved, not copied, from location to location, so there are no backup copies of the data. Thus, reliable wires need to be constructed.

A straightforward first attempt at a means of transporting qubits might be ballistic transport (Figure 2). Simply launch the particle comprising the qubit from the source and “catch” it at the destination. Unfortunately, this method turns out to be impractical in many technologies. The chances of the qubit interacting with other particles on its path or decohering upon launch or arrival is too great.

A second approach to quantum transport is based on the local swap operation, which swaps the state of two *adjacent* qubits. Once the details of a single swap cell are worked out, qubit communication via swap becomes possible. A series of swap cells joined end to end form a wire, and a qubit can be moved along the length of the wire by a series of successive swaps (Figure 3).

The two approaches just discussed use operations available in the classical domain: ballistic movement and swapping. However, the operation known as teleportation is only available in the quantum domain. Quantum teleportation is the re-creation of a quantum state at a distance. Contrary to its science fiction counterpart, quantum teleportation is not instantaneous transmission

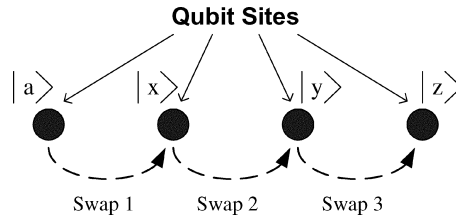


Fig. 3. Multiple swap cells are chained together to form a swap wire. A single qubit is transported down the wire using a series of successive swaps.

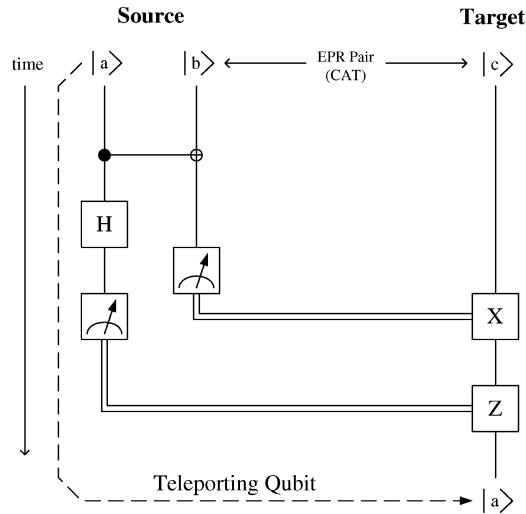


Fig. 4. Quantum teleportation of state $|a\rangle$ over distance. First, *entangled* qubits $|b\rangle$ and $|c\rangle$ are exchanged. Then, $|a\rangle$ is combined with $|b\rangle$ after which measurements produce two *classical* bits of information (double lines). After transport, these bits are used to manipulate $|c\rangle$ to regenerate state $|a\rangle$ at the destination.

of information. Rather, it uses an entangled *EPR pair*, $|\Psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ [Bell 1964].

Figure 4 gives an overview of the teleportation process. We start by generating an EPR pair. We separate the pair, keeping one qubit, $|b\rangle$, at the source and transporting the other, $|c\rangle$, to the destination. When we want to send a qubit, $|a\rangle$, we first interact $|a\rangle$ with $|b\rangle$ using a CNOT gate. We then measure $|a\rangle$ and $|b\rangle$ in the computational basis, and send the two one-bit classical results to the destination, and use those results to re-create the correct phase and amplitude in $|c\rangle$ such that it takes on the original state of $|a\rangle$. The re-creation of phase and amplitude is done with X and Z gates, whose application is contingent on the outcome of the measurements of $|a\rangle$ and $|b\rangle$. Intuitively, since $|c\rangle$ has a special relationship with $|b\rangle$, interacting $|a\rangle$ with $|b\rangle$ makes $|c\rangle$ resemble $|a\rangle$, modulo a phase and/or amplitude error. The two measurements allow us to correct these errors and re-create $|a\rangle$ at the destination. Note that the original state of $|a\rangle$ is destroyed when we take our two measurements. This is consistent with the “no-cloning” theorem, which states that a quantum state cannot be copied.

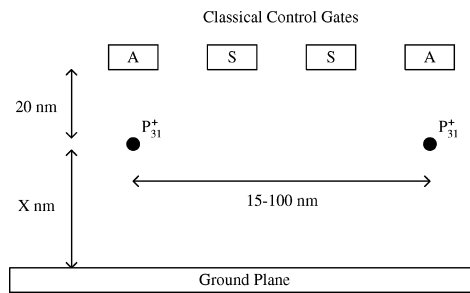


Fig. 5. The basic quantum bit technology proposed by Kane. Qubits are embodied by the nuclear spin of a phosphorus atom coupled with an electron embedded in silicon under high magnetic field at low temperature.

Why bother with teleportation when we end up transporting $|c\rangle$ anyway? Why not just transport $|a\rangle$ directly? First, we can precommunicate EPR pairs with extensive pipelining without stalling computations. Second, it is easier to transport EPR pairs than real data. Since $|b\rangle$ and $|c\rangle$ have known properties, we can employ a specialized procedure known as *purification* to turn a collection of pairs partially damaged from transport into a smaller collection of asymptotically perfect pairs. Third, transmitting the two classical bits resulting from the measurements is more reliable than transmitting quantum data.

The swap wire and the teleportation wire are both viable options for reliable quantum transport, but both have their limitations. Qubits tend to decohere after a certain number of swaps, limiting the maximum length of a basic swap wire. On the other hand, the teleportation primitive involves quite a bit of overhead, putting a limit on the minimum length of a teleportation wire. Swap wires are discussed in detail in Section 4. Teleportation wires are then discussed in Section 5. In the next section, we introduce the technology model that we will be using.

3. AN ARCHITECTURAL MODEL

With some basics of quantum operations in mind, we turn our attention to the technologies available to implement these operations. Experimentalists have examined several technologies for quantum computation, including Josephson junctions [Nakamura et al. 1999; Vion et al. 2002], trapped ions [Monroe et al. 1995], photons [Turchette et al. 1995], bulk spin NMR [Vandersypen et al. 2000b], and phosphorus impurities in silicon [Kane 1998]. Of these proposals, only those building on a solid-state platform are expected to provide the scalability required to achieve a useful computational substrate. The Kane [Kane 1998; Skinner et al. 2002] schemes of phosphorus in silicon build upon modern semiconductor fabrication and transistor design, drawing upon understood physical properties, and we shall be using the second Kane scheme [Skinner et al. 2002] as our technology model in this paper.

Kane proposes that the nuclear spin of a phosphorus atom coupled with an electron embedded in silicon under a high magnetic field and low temperature can be used as a quantum bit. This quantum bit is classically controlled by a local electric field (Figure 5). Two phosphorus atoms spaced 15–100 nm apart

are shown. This interqubit spacing is currently a topic of debate within the physics community. What is being traded off is noise immunity versus difficulty of manufacturing. For our study, we will use a value (60 nm) that lies in the middle. We parametrize our work, however, to generalize for changes in the underlying technology.

Twenty nanometers above the phosphorus atoms lie three classical electrodes that are spaced 20 nm apart. By applying precisely timed pulses to these electrodes, Kane describes how arbitrary one- and two-qubit quantum gates can be realized. The electrodes above the phosphorus atoms are called A-electrodes, while the other ones, which are solely used for transport between nuclei, are called S-electrodes. The details of the pulses and quantum mechanics of this technique are beyond the scope of this paper and are described in Skinner et al. [2002].

3.1 Geometry and Layout Difficulties

The technology model we have chosen presents significant technical challenges. The first hurdle is the placement of the phosphorus atoms themselves. The leading work in this area has involved precise ion implantation through masks and manipulation of single atoms on the surface of silicon [Kane et al. 1999]. While new technologies may be developed to enable precise placement, the key for our work is only the spacing (60 nm) of the phosphorus atoms themselves, and the number of control lines (3) per qubit. The relative scale of quantum interaction and the classical control is what will lead our analysis to fundamental architectural constraints.

A second challenge is the scale of classical control. Each control line into the quantum datapath is roughly 10 nm in width. While such wires are difficult to fabricate, we expect that either electron beam lithography [Anderson et al. 1991], or phase-shifted masks [Sanie et al. 2001] will make such scales possible.

A remaining challenge is the temperature of the device. In order for the quantum bits to remain stable for a reasonable period of time, the device must be cooled to less than 1 K. The cooling itself is straightforward, but the effect of the cooling on the classical logic is a problem. Two issues arise: first conventional transistors stop working as the electrons become trapped near their dopant atoms, which fail to ionize. Second, the 10 nm classical control lines begin to exhibit quantum–mechanical behavior [Ferry and Goodnick 1997].

Fortunately, many researchers are already working on low-temperature transistors. For instance, single-electron transistors (SETs) [Likhareve 1999] are the focus of intense research due to their high-density and low-power properties. SETs have been problematic for conventional computing because they are sensitive to noise and operate best at low temperatures, but this predilection for low temperatures is beneficial for quantum computing. Tucker and Shen [2000] describe this complementary relationship and propose several fabrication methods.

On the other hand, the quantum–mechanical behavior of the control lines presents a subtle challenge that has been mostly ignored to date. At low temperatures, and in narrow wires, the quantum nature of electrons begins to

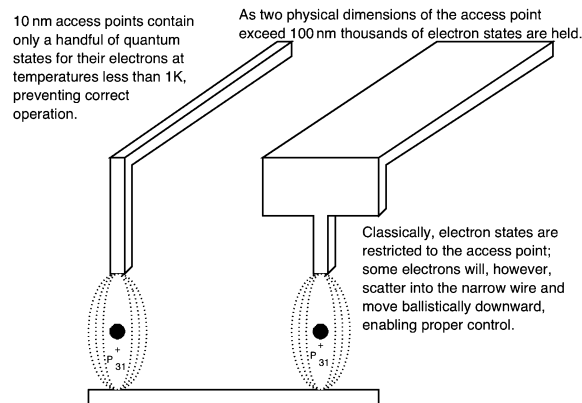


Fig. 6. Quantization of electron states overcome by increasing the physical dimension of the control lines beyond 100 nm. The states propagate quantum-mechanically downward through access vias to control the magnetic field.

dominate over normal classical behavior. For example, in 100 nm wide polysilicon wires at 100 mK, electrons propagate ballistically like waves through only one conductance channel, which has an impedance given by the quantum of resistance, $h/e^2 \approx 25 \text{ k}\Omega$. Impedance mismatches to these and similar metallic wires make it impossible to properly drive the AC current necessary to perform qubit operations.

Avoiding such limitations mandates a geometric design constraint: narrow wires must be short and locally driven by nearby wide wires. Using 100 nm as a rule of thumb² for a minimum metallic wire width sufficient to avoid undesired quantum behavior at these low temperatures, we obtain a control gate structure such as that depicted in Figure 6. Here, wide wires terminate in 10 nm vias that act as local gates above individual phosphorus atoms.

3.2 Pulse-Generation Parameters

The Kane proposal, like all quantum computing proposals, uses classical signals to control the timing and sequence of operations. All known quantum algorithms, including basic error correction for quantum data, require the determinism and reliability of classical control. Without efficient classical control, fundamental results demonstrating the feasibility of quantum computation do not apply (such as the threshold theorem used in Section 4.5.2).

The A- and S-electrodes operate in two states: “on” and “off,” so the control circuitry needs to generate a pulse sequence for each electrode, charging it and discharging it as necessary. The S-electrodes are responsible for movement of the donor electrons, and thus their control sequences consist of single cycle pulses, appropriately spaced, as shown in Figure 7.

A-electrodes are used for performing the hyperfine interaction between a donor electron and a P^+ nucleus. The A-gate pulse sequences consist of a few

²This value is based on typical electron mean free path distances, given known scattering rates and the electron Fermi wavelength in metals.

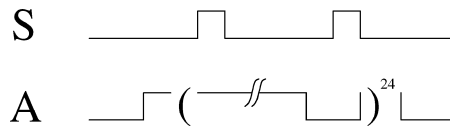


Fig. 7. Example pulse sequences for S- and A-electrodes. S-electrodes aid in electron movement and thus consist of single-cycle pulses appropriately spaced. A-electrodes perform the hyperfine interaction and thus are significantly more complex.

square pulses used for transport interleaved with a much more complicated pulse sequence used to apply the hyperfine interaction. To obtain a “pure” hyperfine evolution (negating the influence of the global magnetic field), as described in Skinner et al. [2002], we must perform the hyperfine evolution in “deltas,” breaking the hyperfine period into 96 pieces. Each of the two-cycle deltas is coupled with 254 cycles of pure magnetic evolution, allowing the magnetic interaction to come back to the beginning of the period. There are only 24 repetitions of the on–off pulse in our sequence because the electron and nucleus need only interact for 1/2 of the period (24 repetitions * 2 cycles per repetition = 48 cycles, which is half of the full 96 cycle interaction).

Figure 7 shows a single hyperfine interaction performed at an A-electrode. We begin with a single cycle high pulse to draw the donor electron over from a nearby S-electrode. The A-electrode is turned off for 2 cycles out of the next 256 to allow the electron to drift down to and interact with the P^+ nucleus. For the other 254 cycles, when the A-electrode is on, the electron is held away from the nucleus, allowing the magnetic field to complete one full period. We then repeat this 23 more times to get the half of a hyperfine evolution that we desire.

From the Skinner–Kane QC proposal [Skinner et al. 2002], we find that the period for the hyperfine interaction (A-gate) of a qubit is 8.5 ns. The period of global magnetic interaction is 22.7 ns under the suggested magnetic field strength of 1.6 mT. The S-gate electron transport step details are ignored in the Skinner–Kane paper but since the interaction is not periodic, it will not affect the clock period. We will show in Section 4.3.3 that a single cycle has plenty of time for a donor electron to move between adjacent electrodes, since this timing *will* still be necessary in the control logic we develop.

3.2.1 Clock Rate. The interaction timings determine the clock rate of the charging of the electrodes. The clock period must clearly be less than the smallest interaction period, which would be the hyperfine interaction delta. Thus, breaking up the hyperfine period into 96 pieces as mentioned earlier, we need a clock period of $8.5 \text{ ns}/96 = 88.5 \text{ ps}$ or a frequency of 11.3 GHz. The high clock rate required by the electron-nucleus hyperfine interaction introduces some interesting constraints on the exact positioning of the components in the Skinner–Kane architecture and will also impose constraints on the circuitry that drives them. We now discuss some effects of two circuit device parameters, voltage swing and drive current.

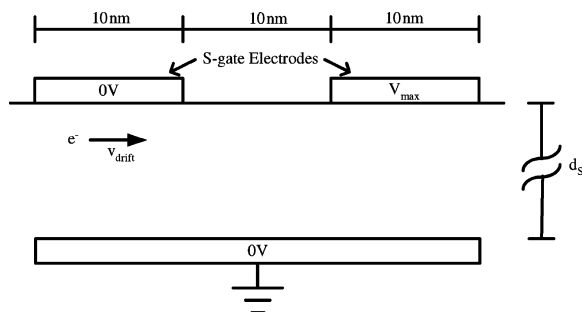


Fig. 8. Setting gate capacitance: Constraints on drive current and maximum slew-rate set a maximum gate capacitance. One option for minimizing this capacitance is to adjust the height relative to the back gate (see text).

3.2.2 Pulse Height. The maximum electrode potential determines the time evolution of the donor electron wavefunction. A lower potential will lead to a slower hyperfine turn-off time because the force exerted on the electron by the electric field will be less. In Skinner et al. [2002], it was assumed that the hyperfine interaction could go from “off” instantaneously to “on.” This assumption may not be valid with such short interaction times and low electrode voltages. Since the focus of this paper is on control circuits and not refinement of the physical model, we will continue on the assumption that the A-gates “just work,” but this issue clearly merits future work.

Error rates in A-gate operations are partially dependent on the differential between the potential at the P ion directly under the electrode and the potential at a P ion adjacent from the same electrode. Greater qubit separation lessens the risk of error from unwanted interaction. However, increased separation results in greater decoherence of the fragile electron spin state as it is transported between A-gates. Thus, we have a trade-off in errors. From Kikkawa and Awschalom [1999], we see that spin coherent electron transport is possible at up to 100 μm . In our analysis, we choose a somewhat arbitrary electrode spacing of 20 nm (for a qubit spacing of 60 nm). We chose this number because electrostatic simulations show that the electron directly under the gate feels an electric field about two orders of magnitude stronger than the electron at an adjacent site.

3.2.3 Slew Rate. Another concern is whether the gate electrodes above the donor sites can be charged in the necessary 88.5 ps. Electrode charge time depends on both the voltage swing and maximum current available.

If we look at the A-gate in Figure 8, the electrode above the donor site is 10 nm in length on a side and 20 nm between electrode centers (electrodes are 10 nm apart). If we model the electrode with a back gate as a parallel plate capacitor, we find the electrode’s capacitance to be $C_{\text{electrode}} \approx \epsilon_0 * \kappa_{\text{Si}} * 100 \text{ nm}^2 / d_{\text{Si}}$, where d_{Si} is the thickness of the Si between the electrode and the metallic back gate. We can calculate the minimum d_{Si} given the constraint that the capacitor must be small enough to be charged in the above mentioned time, with the given

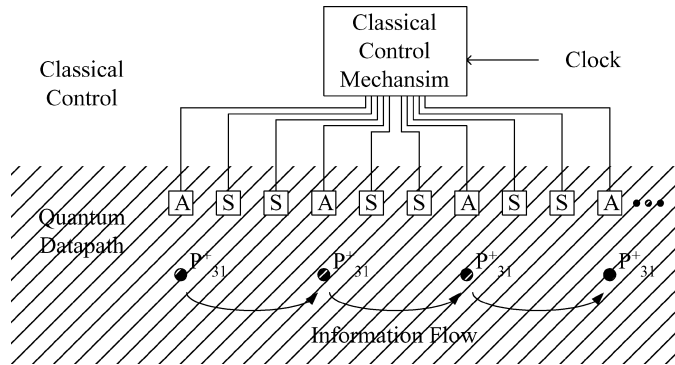


Fig. 9. This is a side view of the swap wire layout, consisting of a classical control mechanism controlling the electrodes in the quantum datapath. The gray area is the substrate in which the ions and electrodes are embedded.

maximum current. If we approximate the electrode/capacitor as charging at a constant rate, the approximate minimum spacing needed is

$$d_{Si} = \frac{V_{\text{electrode}} * \epsilon_0 * \kappa_{Si} * 100 \text{ nm}^2}{\tau_{\text{hyperfine}} * i_{\text{max}}}. \quad (1)$$

From Pakes et al. [2001], we see that the electrode to back gate distance (d_{Si}) affects the probability of erroneous A-gate operation. The closer the back gate is to the electrode, the larger the voltage differential between the active qubit site and the adjacent site, thus a smaller probability of error. This back gate distance would ideally feed into an error analysis that is a function of qubit separation and d_{Si} . This calculation is beyond the scope of this paper, but it will be important future work to produce a physical model of the full A-gate and S-gate operations.

4. SHORT WIRES

In this section, we shall delve into the details of the swap wire discussed earlier, including both the layout of the wire and the control mechanism operating the electrodes. The initial proposal for the architecture [Skinner et al. 2002] makes the simplification that the interactions controlled by the electrodes can be made to be either “on” or “off” using a sequence of precise electrical square pulses to the appropriate electrodes (with no charge and discharge time). We will use this basic concept, bearing in mind the discussion of pulse height and slew rate in Section 3.2.

4.1 Designing the Swap Datapath

For our swap wire architecture, we use the one-dimensional layout proposed in Skinner et al. [2002]. As shown in Figure 9, this layout consists of a classical control mechanism on top of a quantum datapath embedded in substrate (the gray area). The control mechanism must be deterministic and thus operates in the classical realm. The outputs of the control mechanism are bused to a line of electrodes in the quantum datapath. As per the proposal in Skinner et al.

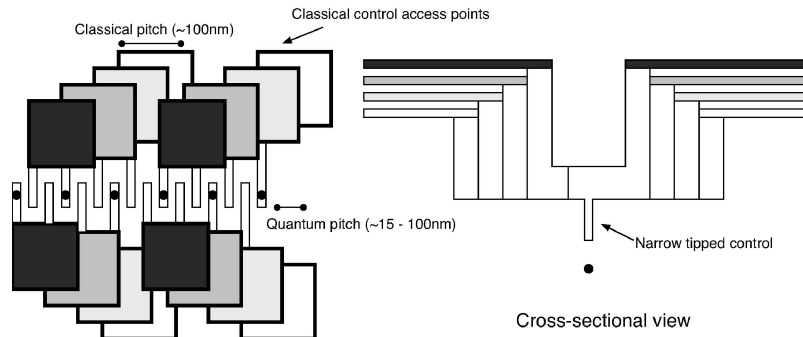


Fig. 10. A linear row of quantum bits: In this figure (not drawn to scale) we depict access control for a line of quantum bits. On the left, we depict a “top down” view. On the right is a vertical cross-section that more clearly depicts the narrow-tipped control lines that quickly expand to classical dimensions.

[2002], there are two S-electrodes between each pair of adjacent A-electrodes. A single swap operation transfers the desired qubit state from one statically placed P^+ ion to the next. Repeated swap operations on successive pairs of ions result in transport of state.

4.2 Pitch Matching

A problem arises from the need to have classical control of our quantum operations. We need a minimum wire width to avoid quantum effects in our classical control lines. Producing a line of quantum bits that overcomes this problem and all of the other ones mentioned in Section 3.1 is possible. We illustrate a design in Figure 10. Note how access lines quickly taper into layers of metal of a classical scale. These control areas can then be routed to access transistors that can gate on and off the frequencies (in the 10s to 100s of MHz) required to apply specific quantum gates.

Of course, any solution for data transport must also support routing, which is not possible without fan-out provided by wire intersections. We can extend our linear row of quantum bits to a four-way intersection capable of supporting sparsely intersecting topologies of qubits. We illustrate the quantum intersection in Figure 11. This configuration is similar to Figure 10 except that the intersection creates a more challenging tapering. Note that each quadrant of our intersection will need to be some minimum size to accommodate access to our control signals.

Recall from Figure 5 that each qubit has three associated control signals (one A- and two S-gates). Each of these control lines must expand from a thin 10 nm tip into a 100 nm access point in an upper metal layer to avoid charge quantization effects at low temperatures (Figure 6). Given this structure, it is possible to analytically derive the minimum width of a line of qubits and its control lines, as well as the size of a four-way intersection. For this minimum size calculation, we assume all classical control lines are routed in parallel, albeit spread across the various metal layers. This parallel nature makes this calculation trivial under normal circumstances (sufficiently “large” lithographic

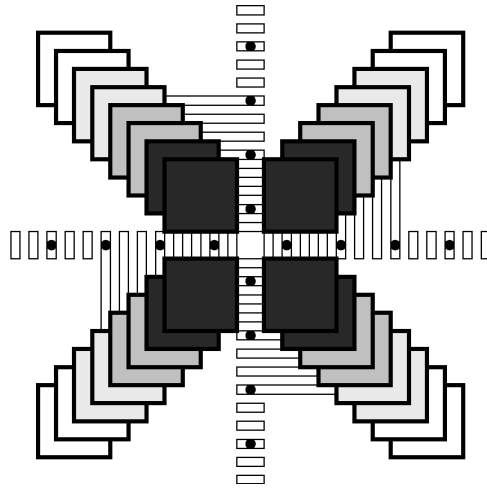


Fig. 11. In this simplified view, we depict a four-way intersection of quantum bits. A diamond-shaped junction is also needed to densely pack junction cells.

feature size λ_c), with the minimum line segment being equal in length to twice the classical pitching, 150 nm in our case, and the junction size equal to four times the classical pitching, 400 nm, in size. However, we illustrate the detailed computation to make the description of the generalization clearer. We begin with a line of qubits.

Let N be the number of qubits along the line segment. Since there are three gates (an A and two S lines) we need to fit in $3N$ classical access points of 100 nm in dimension each, in the line width. We accomplish this by offsetting the access points in the x and y dimensions (Figure 10) by 20 nm. The total size of these offsets will be 100 nm divided by the qubit spacing 60 nm times the number of control lines 3 per qubit, times the offset distance of 20 nm. This number $100 \text{ nm}/60 \text{ nm} \times 3 \times 20 \text{ nm} = 100 \text{ nm}$ is divided by 2 because the access lines are spread out on each side of the wire. Hence, the minimum line segment will be $100 + 50 \text{ nm}$. Shorter line segments within larger, more specialized cells are possible.

Turning our attention to an intersection (Figure 11), let N be the number of qubits along each “spoke” of the junction. We need to fit $3N$ classical access points in a space of $(60 \text{ nm} \times N)^2$, where each access point is at least 100 nm on a side. As with the case of a linear row of bits, a 20 nm x and y shift in access point positioning between layers is used for via access. Starting with a single access pad of 100 nm, we must fit $100 \text{ nm}/60 \text{ nm} \times 3$ additional pads shifted in x and y within the single quadrant of our intersection. This leads to a quadrant size of $100 + 100 \text{ nm}/60 \text{ nm} \times 3 \times 20 \text{ nm} = 200 \text{ nm}$. Therefore, the minimum size four way intersection is 8 (rounding up) qubits in each direction.

In this construction we have assumed a densely packed edge to each spoke, however, this is easily “unpacked” with a specialized line segment, or by joining to another junction that is constructed inversely from that shown in Figure 11. Obviously, the specific sizes will vary according to technological parameters and

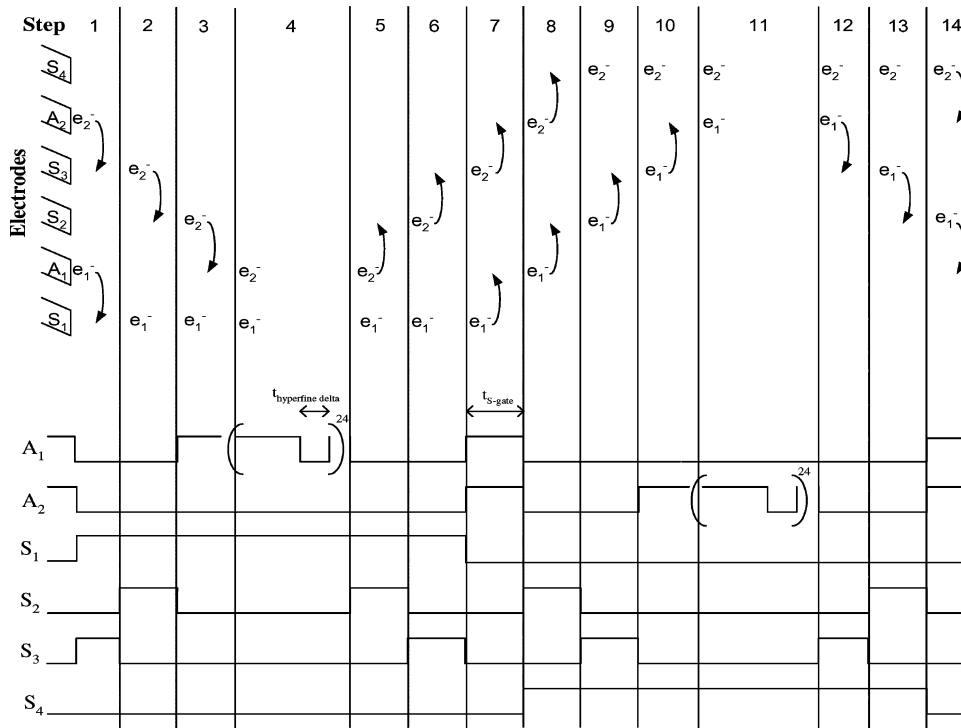


Fig. 12. This is the time sequence of S- and A-gates operations necessary to swap two qubits in the 1D layout. The parallelograms on the left represent A- and S-gates, and each vertical line of electrons represents a time step in the swap. At the bottom are the necessary control signals for each A- and S-gates.

assumptions about control logic, but this calculation illustrates the approximate effect of what appears to be a fundamental tension between quantum operations and the classical signals that control them. A minimum intersection size implies minimum wire lengths, which imply a minimum size for computation units.

4.3 Classical Control

Now that we have the layout and physical characteristics of the swap wire, we next need to design the classical control circuitry needed to operate it. We begin in the next section by investigating the control pulses necessary for the electrodes. We then describe and analyze a gate-level design that generates these pulses.

4.3.1 Control Pulses. Figure 12 illustrates the sequence of steps necessary to perform a single swap operation. We see that a single swap cell consists of six electrodes (two A- and four S-electrodes), and thus six pulse sequences are necessary to perform a swap between adjacent qubits. Each time step is delineated by a dotted line with the corresponding electrode voltage levels below.

The donor electron for the *P* ion at gate A₁ is labeled e₁, while e₂ similarly corresponds to gate A₂. In order to perform a swap, e₁ must execute a hyperfine

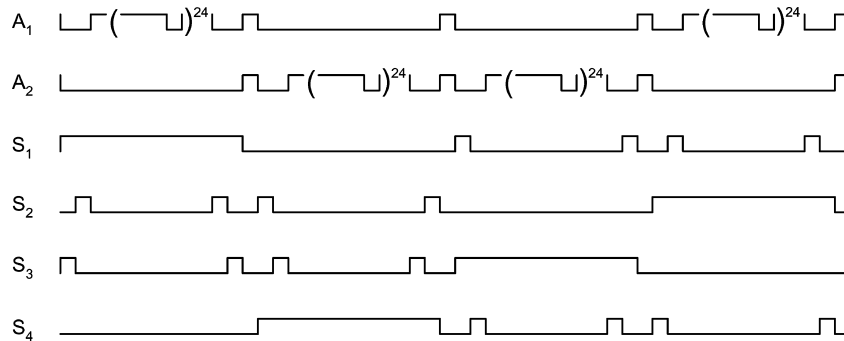


Fig. 13. These are the complete pulse sequences necessary to implement a full swap wire, using the swap cell naming conventions in Figure 12.

interaction at gate A_2 , and e_2 must execute a hyperfine interaction at gate A_1 . The basic procedure is to pull e_1 “aside” (to gate S_1), while e_2 is moved to gate A_1 to perform its hyperfine interaction. Then e_2 is pulled “aside” (to gate S_4), while e_1 is moved to gate A_2 to perform its hyperfine interaction. Finally, both electrons are moved back to their corresponding A-gates, with the qubit values swapped.

The S-gate pulse sequences consist of single-cycle square pulses used for transport and lengthier square pulses used to “hold” the donor electrons. The A-gate pulse sequences consist of a few single-cycle square pulses interleaved with the much more complex sequence necessary to perform a pure hyperfine interaction. We can see this clearly in Figure 12 for the case of a single swap cell. Now we can join many of these swap cells, end to end, to create a swap wire.

The goal is to move one qubit continuously down the line by doing successive swaps, as shown in Figure 3. Consider an abstract swap wire (i.e., for the moment, forget about the underlying technology). Let us label a series of qubits in the wire A, B, C, D, and E. Qubits A and B swap while qubits C and D are also simultaneously swapping. Then qubits B and C swap, while qubits D and E are swapping. This results in the qubit state originally at A to be swapped first to B, then to C. Now we repeat these steps ad infinitum to create the swap wire.

Note that qubits A and C are performing the same operations, while B and D are also identical in their operations. In fact, every qubit in the swap wire fits into one of these two types. Thus, we need only design control signals for A and B, and we will have control signals for the whole wire. So we need only generate pulse sequences for four S-electrodes and two A-electrodes.

In order to extend the pulse sequences shown in Figure 12 to apply to the swap wire, we need to alternate between swapping the two qubits in the swap cell and swapping these two with qubits in adjacent swap cells. Figure 13 shows the full pulse sequences necessary for the complete swap wire. This sequence of operations, repeated ad infinitum, will result in alternating qubits traveling along the wire leftward and alternating qubits traveling along the wire rightward, as desired.

4.3.2 Control Circuitry. Now that we have determined the pulse sequences that we need to implement, we can design the control circuitry necessary to

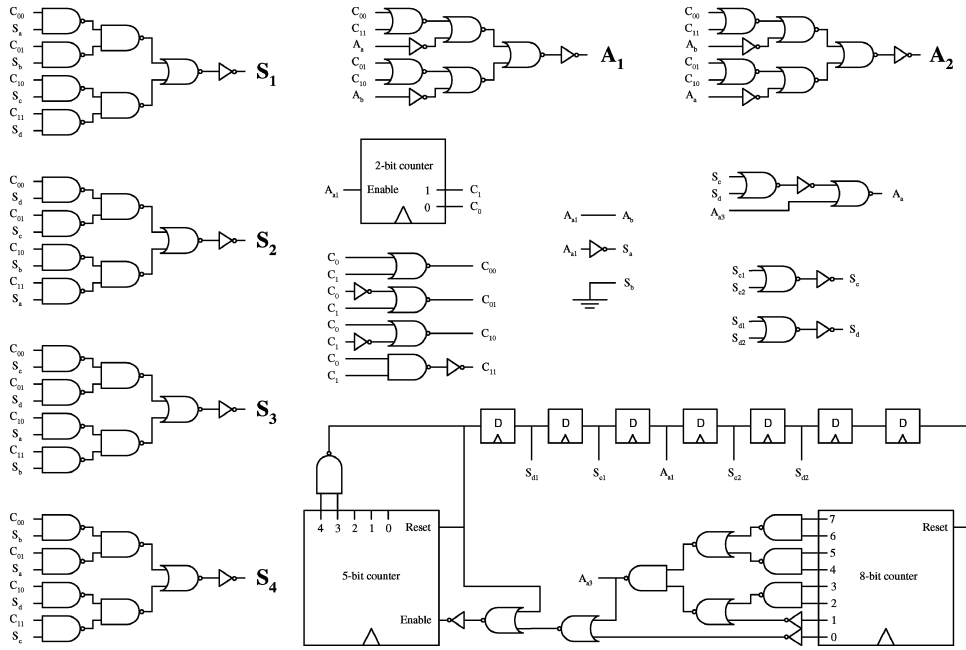


Fig. 14. This control mechanism for a swap wire produces the pulse sequences shown in Figure 13. The primary outputs, four S-gates signals and two A-gates signals, are in bold type and larger than the other text.

produce these pulses. The extremely low temperatures (1 K) required for proper operation of a solid state QC architecture render standard CMOS transistors useless. Fortunately, a good deal of research has been conducted in modeling and building single-electron transistors (SETs) [Kim et al. 2001; Takahashi et al. 2000; Wasshuber 2002], which actually have more favorable characteristics at lower temperatures, making SETs an ideal choice for the drive circuitry and control logic.

Figure 14 shows the global control mechanism that generates the six distinct pulse sequences, one each for the S_1 -, S_2 -, S_3 -, S_4 -, A_1 - and A_2 -gates. The signals are then distributed to many A- and S-gates.

The bulk of the control is used to generate the complicated A-gate pulse sequence. The goal for the hyperfine interaction is to turn the A-gate off for 2 cycles out of every 256, and to do this 24 times. The 8-bit counter is used to count up to 256 and turn the A-gate off for 2 of those cycles. The 5-bit counter is used to count up to 24 to determine the end of the hyperfine interaction. The S-gate control is implemented as a series of D flip-flops. There is one series of flip-flops for each of the two sequences we wish to activate. The T flip-flop is toggled to switch between these two sequences. The D flip-flops are used to keep the appropriate signal high for a cycle. When either sequence is completed, we set the A-gate high and begin the hyperfine interaction sequence again.

In order to determine the feasibility of this design, we approximate its area using modern technology constraints. We shall use a minimum feature size of $10 \text{ nm} \times 10 \text{ nm}$, the same size as our electrodes, for our SET islands. Figure 15

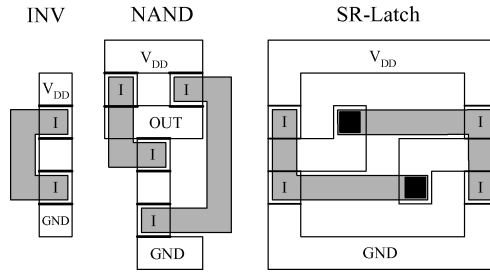


Fig. 15. SET layouts for three basic components: an inverter, a NAND gate, and an SR-latch. Squares labeled “I” are the SET islands. Gray areas (drawn thinner for clarity) are above the SETs, connected to the lower level at the black junctions.

shows the layout of an inverter, a NAND gate, and an SR-latch using SETs. The squares labeled “I” represent $10 \text{ nm} \times 10 \text{ nm}$ islands. Using the fact that each unit square in Figure 15 is $10 \text{ nm} \times 10 \text{ nm}$, we can calculate that the inverter is $20 \text{ nm} \times 50 \text{ nm}$, the NAND gate is $40 \text{ nm} \times 70 \text{ nm}$ (the same as a NOR gate), and the SR-latch is $70 \text{ nm} \times 70 \text{ nm}$. Additionally, an XOR gate in SET technology only uses two SETs, so it uses approximately the same amount of room as an inverter.

In total, the design consists of a grand total of 44 SR latches, 182 NAND or NOR gates, 55 inverters, and 13 XOR gates. This means that the gates take up a total area of $\sim 0.79 \mu\text{m}^2$. Without doing a full optimized layout, we cannot know the actual area of the mechanism. So for the purposes of discussion, we will assume that the space for routing between the gates is approximately 50% of the area of the gates, giving us a total area of $\sim 1.2 \mu\text{m}^2$ for the entire control mechanism.

4.3.3 Drive Circuit Calculations. The introduction of SET buffers as electrode drive circuits leads to a few problems, the primary ones being voltage swing and drive current. All physically implemented SETs so far are capable of handling voltage swings of up to only 40 mV [Uchida et al. 2002], so the A- and S-gate electrodes can only be charged to at most 40 mV. This relatively low voltage constraint affects the operating speed of the electron wavefunction deforming operations, as well as the quantum operation error rates as mentioned in Section 3.2.

Since *single-electron* transistors work on the order of single-electron flow from source to drain, they have low drive current. In fact, in current physical specimens, the maximum current observed is around 5 nA [Kim et al. 2001]. Thus, we have the following parameters: $V_{\text{electrode}} = 0.04 \text{ V}$, $\tau_{\text{hyperfine}} = 88.5 \text{ ps}$, and $i_{\text{max}} = 5 \text{ nA}$. If we plug these into the formula derived in Section 3.2, we obtain $d_{\text{Si}} = 0.09 \mu\text{m}$.

With the electrode distances, voltages, and currents established, we can calculate S-gate timing. Placing d_{Si} into Figure 8 and using an electrostatic simulator, we calculate the electric field to be approximately 300 V/cm at the target electron. We then use the experimental data in Jacobini et al. [1977] to compute the electron drift velocity for that field strength. Using this we find the approximate time for an electron to move from one electrode to the next (a distance of

20 nm) to be $t_{\text{shuttling}} = 0.2$ ps. This justifies our assumption that the electron transport time is significantly less than the hyperfine interaction time. Since the expected time for an electron to move between two adjacent electrodes is ~ 0.2 ps, and our clock period is 88.5 ps, we can expect that each electron will easily reach the next gate within one clock cycle. For this reason, to perform electron transport, each of the S gates need only remain high for one cycle.

4.3.4 Other Technologies. Several problems have been mentioned concerning the use of SETs for the control mechanism, such as low drive and voltage swing. We have assumed the use of SETs in our design because they operate well at low temperature. However, future work must be done to investigate the possibility of using other types of transistors (such as bipolar) or transistors made with different materials (GaAs or perhaps carbon nanotubes) to address our circuit concerns.

4.4 A SIMD Approach

From Section 4.3.2, we see that regardless of the specific layout of SET circuits, the qubit control circuitry for a swap wire is extensive. An issue we must therefore consider is the placement of this control. The control circuitry consumes too much space to allow for a unique control block dedicated to each swap cell. This results in the need for multiple swap cells to share a single control block.

As described in Section 4.3.2, the area of the control circuitry is estimated to be about $1.2 \mu\text{m}^2$. If we naively assume the logic can be laid out with the same width as the swap cells, 10 nm, the minimum wire length allowed will be $120 \mu\text{m}$ in order to fit all the necessary control circuitry. If a swap cell cluster is to be used as our wire to connect quantum logic components that cannot be adjoined without routing, we will be forced to design very sparse quantum circuits.

This large discrepancy in area between the control and datapath layouts suggests the reuse of pulse-generation circuits, or, in other words, a SIMD model of qubit control, where many qubits receive signals simultaneously. In the case of the swap wire, this is straightforward: qubits are transported identically between sets of functional units along sections of swap wire connected to the same control block. Taking this design a step further, the swap wires could be analogous to a standard microprocessor and its clock distribution network. If we place a global instance of the control circuitry on the chip, we can distribute the control signals to all the swap wires. This scheme, coupled with a few SETs above the swap cells to enable and disable the control signals, defines a fully functional swap wire.

4.5 Limits of Short Wires

We now analyze this short wire to derive two important architectural constraints: the classical-quantum interface boundary and the latency/bandwidth characteristics. We strive to achieve a loose lower bound on these constraints for a given quantum device technology. While future quantum technologies

may have different precise numbers, it is almost certain they will continue to be classically controlled, and thus also obey similar constraints based upon this classical-quantum interface.

4.5.1 Technology Independent Limits. Thus far we have focused our discussion on a particular quantum device technology. This has been useful to make the calculations concrete. Nevertheless, it is useful to generalize these calculations to future quantum device technologies. Therefore, we parametrize our discussion based on a few device characteristics:

Assuming two-dimensional devices (i.e., not a cube of quantum bits), let p_c be the classical pitching required, and p_q the quantum one. Furthermore, let R be the ratio p_c/p_q of the classical to quantum distance for the device technology, m be the number of classical control lines required per quantum bit, and finally λ_c be the feature size of the lithographic technology. We use two separate variables p_c and λ_c to characterize the “classical” technology because they arise from different physical constraints. The parameter λ_c comes from the lithographic feature size, while p_c (which is a function of λ_c) is related to the charge quantization effect of electrons in gold. With the Kane technology we assume a spacing p_q of 60 nm between qubits, three control lines per bit of 100 nm (p_c) each, and a λ_c of 5 nm. We can use these to generalize our pitch matching equations. Here we find that the minimum line segment is simply equivalent to $R(1 + 2\lambda_c m/p_q)$ qubits in length.

Examining our junction structure (Figure 11), we note that it is simply four line segments, similar to those calculated above, except that the control lines must be on the same side. Therefore, the minimum crossing size of quantum bits in a two-dimensional device is of size $\approx 2R(1 + 4\lambda_c m/p_q)$ on a side.

4.5.2 Latency and Bandwidth. Calculating the latency and bandwidth of quantum wires is slightly different than it is for classical systems. The primary difficulty is decoherence—i.e., quantum noise. Unlike classical systems, you cannot simply re-send quantum data when an error is detected. The “no-cloning” theorem [Nielsen and Chuang 2000], according to which quantum states cannot be perfectly copied, prohibits transmission by duplication, thereby making it impossible to re-transmit quantum data if they are corrupted. Once the data are destroyed by the noisy channel, you have to start the entire computation over. To avoid this loss, quantum data are encoded in a sufficiently strong error-correcting code that, with high probability, the data will remain coherent for the entire length of the quantum algorithm.

Our goal is to provide a quantum communication layer that sits below higher level error correction schemes. Consequently, we start our calculation by assuming a channel with no error correction. Then we factor in the effects of decoherence and derive a maximum wire length for our line of qubits.

Recall that data traverses the line of qubits with swap gates, each of which takes approximately $1 \mu\text{s}$ to execute in the Kane technology. Thus, a single row of quantum bits has latency

$$\text{latency} = 1 \mu\text{s} \times \text{distance}/60 \text{ nm}. \quad (2)$$

This latency can be quite large. A short $1 \mu\text{m}$ has a latency of $17 \mu\text{s}$! On the plus side, the wire can be fully pipelined and has a sustained bandwidth of $1/17 \mu\text{s} = 1\text{M}$ qbps (quantum bits per second). This may seem small compared to a classical wire, but keep in mind that quantum bits hold an exponential amount of information and can enable algorithms with exponential power.

The number of error-free qubits is actually lower than this physical bandwidth. Noise, or decoherence, degrades quantum state and makes the true bandwidth of our wire less than the physical quantum bits per second. Bits decohere over time, so longer wires will have a lower bandwidth than shorter ones.

The stability of a quantum bit over time decays (exactly like a nonerror corrected classical bit) as a function $e^{-k \times t}$. Usually, a normalized form of this equation is used, $e^{-\lambda \times t}$, where t in this new equation is the number of operations and λ is related to the time per operation and the original k . As quantum bits traverse our wire they arrive with a fidelity proportional to the latency, namely,

$$\text{fidelity} = e^{-k \times \text{latency}}. \quad (3)$$

The true bandwidth is then proportional to the fidelity

$$\text{bandwidth}_{\text{true}} = \text{bandwidth}_{\text{physical}} \times \text{fidelity}. \quad (4)$$

Choosing a reasonable³ value of $\lambda \approx 10^{-6}$, the true bandwidth of a wire is

$$1/17 \mu\text{s} \times e^{-10^{-6} \times \text{distance}/60 \text{ nm}} \quad (5)$$

which for a $1 \mu\text{m}$ wire is close to ideal (999,983 qbps).

This does not seem to be a major effect, until you consider an entire quantum algorithm. Data may traverse back and forth across a quantum wire millions of times. It is currently estimated [Aharonov and Ben-Or 1997] that a degradation of fidelity more than 10^{-4} makes arbitrarily long quantum computation theoretically unsustainable, with the practical limit being far higher [Oskin et al. 2002]. This limit is derived from the threshold theorem, which relates the decoherence of a quantum bit to the complexity of correcting this decoherence [Aharonov and Ben-Or 1997; Knill et al. 1998; Preskill 1998].⁴ Given our assumptions about λ , the maximum theoretical wire distance is about $6 \mu\text{m}$.

4.5.3 Technology Independent Metrics. Our latency and bandwidth calculations require a few more device parameters. Let T be the time per basic swap operation. Let λ be the decoherence rate, which for small λ and T is equivalent to the decoherence a quantum bit undergoes in a unit of operation time T . This makes the latency of a swap wire equal to

$$\text{latency} = T \times D \quad (6)$$

³This value for λ is calculated from a decoherence rate of 10^{-6} per operation, where each operation requires $1 \mu\text{s}$. We refer the interested reader to Nielsen and Chuang [2000].

⁴By “practical” we mean without an undue amount of error correction. The threshold theorem ensures that theoretically we can compute arbitrarily long quantum computations, but practical error correction overhead makes the real limit 2–3 orders of magnitude higher [Oskin et al. 2002].

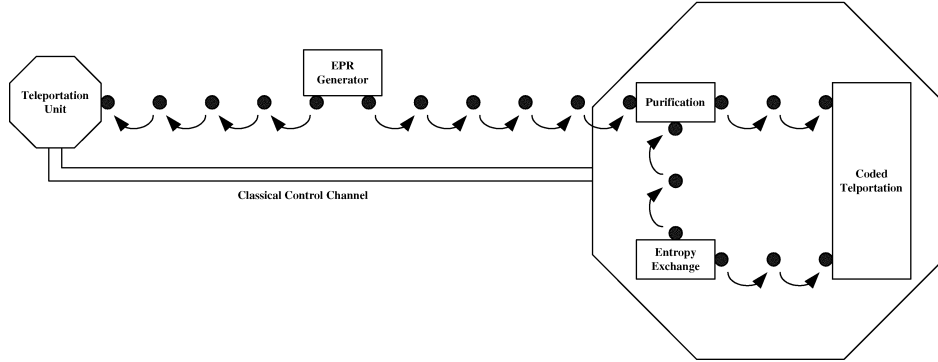


Fig. 16. Teleportation wire: Solid double lines represent classical communication channels, while chained links represented quantum swap wires.

where distance D is expressed in the number of qubits. The bandwidth is proportional to the fidelity or

$$\text{bandwidth}_{\text{true}} = \frac{1}{T} e^{-\lambda D}. \quad (7)$$

This bandwidth calculation is correct so long as the fidelity remains above the critical threshold $C \approx 10^{-4}$ required for fault tolerant computation. Finally, the maximum distance of this swap wire is the distance when the fidelity drops below the critical threshold

$$\text{distance}_{\text{max}} = \log_e(1 - C) / -\lambda. \quad (8)$$

Realize that no amount of error correction will be robust enough to support a longer wire, while still supporting arbitrarily long quantum computation. For this we need a more advanced architecture. One obvious option is to break the wire into segments and insert “repeaters” in the middle. These quantum repeaters are effectively performing state restoration (error correction). However, we can do better, which is the subject of the next section.

5. LONG WIRES

The architecture for a teleportation wire is shown in Figure 16 and implements the design presented in Figure 4. The basic components shall be discussed in further detail in the next section, but we begin by describing the overall operation.

The laws of quantum physics require that qubits neither be created nor destroyed, simply modified. For this reason, most complex quantum operations require one or more “scratch” qubits as input, generally assumed to be in the zero state ($|0\rangle$). These $|0\rangle$ qubits are generated in a steady stream by the entropy exchange unit.

Our teleportation wire also needs a “centralized” EPR generator to generate a steady stream of EPR pairs, which are then transported to either end of the

teleportation wire using the swap wires discussed earlier (with possible repeaters along the way). It is easier to transport EPR pairs than arbitrary qubits along swap wires. Nonetheless, at each end of the teleportation wire, we need a purification unit to refresh uncorrupted EPR pairs and discard corrupted ones.

With all these components, we can now implement the design shown in Figure 4. Once the EPR pair is distributed to either end, we need a few basic gates (a CNOT, a Hadamard, two measurements, a controlled bit-flip and a controlled phase flip) to teleport the state of the source qubit to the target location. In the next sections, we further discuss the core architectural components, followed by a brief discussion of the control requirements, and finally an analysis of the limits of this wire.

5.1 Basic Building Blocks

In this section, we highlight three important architectural building blocks: the *Entropy Exchange Unit*, the *EPR Generator*, and the *Purification Unit*. Note that the description of these blocks is quasi-classical in that it involves input and output ports. Keep in mind, however, that all operations (except measurement) are inherently reversible, and the specification of input and output ports merely provides a convention for understanding the forward direction of computation.

5.1.1 Entropy Exchange Unit. The method used in the generation of $|0\rangle$ states depends heavily on the technology chosen. In general, the process can be viewed as one of thermodynamic cooling. $|0\rangle$ states are very “ordered,” and thus have low entropy. The goal in generating a $|0\rangle$ state is to take an arbitrary qubit and reduce its entropy sufficiently. However, since entropy cannot be destroyed, it is simply transferred elsewhere (hence, the term entropy exchange).

The first step is to take “junk” qubits (in unknown state) and polarize them to all be in some predetermined mixed state. This can be done in a number of ways, for example, through optical pumping [Kane et al. 1999; Verhulst et al. 2001] or using spintronics methods [Kane et al. 1999]. These polarized qubits are then “sorted” to extract as many $|0\rangle$ states as possible, for example, using the method described in Schulman and Vazirani [1999]. The result is that some relatively constant fraction of the incoming junk qubits are turned into $|0\rangle$ qubits, while the rest have more entropy on average as a result of this procedure and are thus recycled.

5.1.2 EPR Generator. The EPR Generator creates a steady stream of EPR pairs. We start with two $|0\rangle$ state qubits from our entropy exchange unit. The first qubit is passed through a Hadamard gate and then used as the control in a CNOT gate with the other qubit as the target. This gives the desired state, the EPR pair $(|00\rangle + |11\rangle)/\sqrt{2}$. Note that a real EPR Generator would likely consist of several of these in parallel in order to increase the speed of EPR pair generation.

5.1.3 Purification Unit. The purification unit takes as input n EPR pairs that have been partially corrupted by errors and outputs nE asymptotically

perfect EPR pairs. E is the entropy of entanglement, a measure of the number of quantum errors which the pairs suffered. The details of this entanglement purification procedure are beyond the scope of this paper but the interested reader can see Bennett et al. [1996a, 1996b, 1996c].

5.2 Classical Control

Each of the three units discussed above requires classical control, as does the process of teleportation illustrated in Figure 4. A full implementation of this control is beyond the scope of this paper (and, in fact, depends heavily on the algorithm and technology chosen for each component), but we would like to get some idea of the scale of the control mechanism relative to the swap wire control.

The EPR Generator consists of a Hadamard and a CNOT. There may be several of these in parallel (in order to produce more EPR pairs per unit time), but they can all use identical control signals, thus we only need control for one such circuit. A Hadamard gate is likely the same order of complexity as a swap operation. The CNOT gate designed in Skinner et al. [2002] takes approximately six times as long as a swap operation (the hyperfine operation is performed for six times as long, and since this is the dominant operation in terms of time, we can make this simplifying assumption). Putting this altogether, we conclude that the control mechanism for the EPR Generator is more complicated than for the swap wire, but we will not venture a guess as to how much more.

Next, we need two swap wires to transport the EPR pairs to the ends of the teleportation wire. These swap wires will use the control mechanism derived earlier.

Many competing proposals exist for the design of the Entropy Exchange Unit. If we assume that some form of “sorting” mechanism is used to compress entropy into a few qubits, then we will need conditional swapping in order to perform this sorting, and thus the Entropy Exchange Unit control mechanism will be at least as complicated as that for the swap wire (likely an order of magnitude more so).

The purification unit detects errors in EPR pairs and discards faulty ones. Though multiple designs for such functionality exist, they all have in common that they include Hadamards, CNOTs, swaps, as well as others. Regardless of the design, this is clearly more complex than the EPR Generator control discussed above, and thus we will restrict ourselves to saying that the control for the Purification Unit is also significantly more complex than the control for the swap wire.

Finally, we have the teleportation circuit in Figure 4. Once again, we make the claim that the control for this is more complex than for the EPR Generator, and thus transitively it is more complex than for the swap wire.

Putting all of these control mechanisms together, we see that the teleportation wire requires an enormous overhead for its classical control, likely at least an order of magnitude more than the swap wire. However, we get a wire that can transmit a qubit reliably over a nearly arbitrary distance in almost negligible time.

5.3 Analysis

Our goal now is to analyze this architecture and derive its bandwidth and latency characteristics. The bandwidth is proportional to the speed with which reliable EPR pairs are communicated. Since we are communicating unreliable pairs we must purify them, so the efficiency of the purification process must be taken into account. Purification has an efficiency roughly proportional to the fidelity of the incoming, unpurified qubits [Schulman and Vazirani 1998]:

$$\text{purification}_{\text{efficiency}} \approx \text{fidelity}^2. \quad (9)$$

Entropy exchange is a sufficiently parallel process that we assume enough zero qubits can always be supplied. Therefore, the overall bandwidth of this wire is

$$1/1 \mu\text{s} \times e^{-2 \times 10^{-6} \times \text{distance}/60 \text{ nm}} \quad (10)$$

which for a 1 μm wire is 999,967 qbps. Note this result is less than for the simple wiring scheme, but the decoherence introduced on the logical quantum bits is only $O(e^{-\lambda \times 10})$. It is this latter number that does not change with wire length that makes an important difference. In the previous short-wire scheme we could not make a wire longer than 6 μm . Here we can make a wire of nearly arbitrary length. For example, a wire that is 10 mm long has a bandwidth of 716,531 qbps, while a swap wire has an effective bandwidth of zero at this length.

The latency result is even better. Pipelined precommunication of EPR pairs allows us to achieve transmission with a constant latency. This latency is roughly the time it takes to perform teleportation, or $\approx 20 \mu\text{s}$.

Using the same constants defined above for the swapping channel, we can generalize our analysis of teleportation channels. The latency is simply

$$\text{latency} \approx 10T. \quad (11)$$

The bandwidth is

$$\text{bandwidth}_{\text{true}} = \frac{1}{T} e^{-2\lambda D}. \quad (12)$$

Unlike the short wire, this bandwidth is *not* constrained by a maximum distance related to the threshold theorem, since teleportation is unaffected by distance. The communication of EPR pairs before teleportation, however, can be affected by distance, but at a very slow rate. While purification must discard more corrupted EPR pairs as distance increases, this effect is orders-of-magnitude smaller than direct data transmission over short wires and is not a factor in an practical silicon of up to 10s of millimeters on a side.

6. CONCLUSION

Our study has focused on a critical aspect of any quantum computing architecture, quantum wires to transport quantum data. Building upon key pieces of quantum technology, we have provided an end-to-end look at a quantum wire architecture. We have shown that our teleportation wire scales with distance and that swap wires do not, but teleportation wires suffer from far greater

overhead than do swap wires. We have also discovered fundamental architectural pressures not previously considered. These pressures arise from the need to co-locate physical phenomena at both the quantum and classical scale. Our analysis indicates that these pressures will force architectures to be sparsely connected, resulting in coarser-grain computational components than generally assumed by previous quantum computing studies. We believe that further architectural studies of this nature will be valuable in identifying the research challenges facing quantum technologies of the future.

ACKNOWLEDGMENTS

Thanks to John Owens and Matt Farrens for their helpful comments on preliminary material for this paper. This work is supported in part by the DARPA Quantum Information Science and Technology Program, by NSF CAREER grants to Fred Chong, Mark Oskin, and John Kubiawicz, an NSF NER grant, and by a UC Davis Chancellor's Fellowship to Fred Chong.

REFERENCES

- AHARONOV, D. AND BEN-OR, M. 1997. Fault tolerant computation with constant error. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing*, 176–188.
- ANDERSON, E., V. BOEGLI, SCHATTENBURG, M., KERN, D., AND SMITH, H. 1991. Metrology of electron beam lithography systems using holographically produced reference samples. *J. Vac. Sci. Technol.*, B 9.
- BELL, J. S. 1964. On the Einstein–Podolsky–Rosen paradox. *Physics 1*, 195–200. Reprinted in J. S. Bell, *Speakable and Unspeakable in Quantum Mechanics*, Cambridge University Press, Cambridge, 1987.
- BENNETT, C. H., BERNSTEIN, H. J., POPESCU, S., AND SCHUMACHER, B. 1996a. Concentrating partial entanglement by local operations. *Phys. Rev. A 53*, 4, 2046–2052. arXiv e-print quant-ph/9511030.
- BENNETT, C. H., BRASSARD, G., CRÉPEAU, C., JOZSA, R., PERES, A., AND WOOTTERS, W. 1993. Teleporting an unknown quantum state via dual classical and EPR channels. *Phys. Rev. Lett. 70*, 1895–1899.
- BENNETT, C. H., BRASSARD, G., POPESCU, S., SCHUMACHER, B., SMOLIN, J. A., AND WOOTTERS, W. K. 1996b. Purification of noisy entanglement and faithful teleportation via noisy channels. *Phys. Rev. Lett. 76*, 722. arXiv e-print quant-ph/9511027.
- BENNETT, C. H. AND DiVINCENZO, D. P. 2000. Quantum information and computation. *Nature 404*, 247–255.
- BENNETT, C. H., DiVINCENZO, D. P., SMOLIN, J. A., AND WOOTTERS, W. K. 1996c. Mixed state entanglement and quantum error correction. *Phys. Rev. A 54*, 3824. arXiv e-print quant-ph/9604024.
- DiVINCENZO, D. P. 1995. Quantum computation. *Science 270*, 5234, 255. arXiv e-print quant-ph/9503016.
- FERRY, D. K. AND GOODNICK, S. M. 1997. *Transport in Nanostructures*. Cambridge Studies in Semiconductor Physics & Microelectronic Engineering, vol. 6. Cambridge University Press, Cambridge, MA.
- GERSHENFELD, N. AND CHUANG, I. 1998. Quantum computing with molecules. *Scientific American*.
- GOODKIND, J. M. 2000. Proposed fabrication of a quantum computer using electrons on helium. In *Second Annual SquInT Workshop*. Poster Abstract.
- JACOBINI, C., CANALI, C., OTTAVIANI, G., AND QUARANTA, A. 1977. A Review of Some Charge Transport Properties of Silicon. *Solid-State Electronics 20*, 77–89.
- KANE, B. 1998. A silicon-based nuclear spin quantum computer. *Nature 393*, 133–137.
- KANE, B. E., McALPINE, N. S., DZURAK, A. S., CLARK, R. G., MILBURN, G. J., SUN, H. B., AND WISEMAN, H. 1999. Single spin measurement using single electron transistors to probe two electron systems. arXiv e-print cond-mat/9903371. *Phys. Rev. B*. submitted.

- KIKKAWA, J. M. AND AWSCHALOM, D. D. 1999. Lateral drag of spin coherence in gallium arsenide. *Nature* 397, 139–141.
- KIM, D. H., SUNG, S., KIM, K. R., CHOI, B. H., HWANG, S. W., AHN, D., LEE, J. D., AND PARK, B. 2001. Si single-electron transistors with sidewall depletion gates and their application to dynamic single-electron transfer logic. In *IEEE International Electron Devices Meeting*.
- KNILL, E., LAFLAMME, R., MARTINEZ, R., AND TSENG, C.-H. 1999. A cat-state benchmark on a seven bit quantum computer. arXiv e-print quant-ph/9908051.
- KNILL, E., LAFLAMME, R., AND ZUREK, W. H. 1998. Resilient quantum computation. *Science* 279, 5349, 342–345. arXiv e-print quant-ph/9702058.
- LIKHAREVE, K. K. 1999. Single-electron devices and their applications. In *Proceedings of the IEEE'87*.
- LLOYD, S. 1995. Quantum-mechanical computers. *Scientific American* 273, 4 (Oct.), 44.
- MONROE, C., MEEKHOF, D. M., KING, B. E., ITANO, W. M., AND WINELAND, D. J. 1995. Demonstration of a fundamental quantum logic gate. *Phys. Rev. Lett.* 75, 4714.
- NAKAMURA, Y., PASHKIN, Y. A., AND TSAI, J. S. 1999. Coherent control of macroscopic quantum states in a single-cooper-pair box. *Nature* 398, 786–788.
- NIELSEN, M. AND CHUANG, I. 2000. *Quantum computation and quantum information*. Cambridge University Press, Cambridge, UK.
- OSKIN, M., CHONG, F., AND CHUANG, I. 2002. Overhead reduction in a architecture for quantum computers. *IEEE Computer* 35, 1, 79–87.
- PAKES, C., WELLARD, C., JAMIESON, D., HOLLENBERG, L., PRAYER, S., DZURAK, A., HAMILTON, A., AND CLARK, R. 2001. Modeling of electrostatic gate operations in the Kane solid state quantum computer. *Proc. SPIE* 4591, 70–77.
- PRESKILL, J. 1998. Fault-tolerant quantum computation. In *Quantum Information and Computation*, H.-K. Lo, T. Spiller, and S. Popescu, eds. World Scientific, Singapore.
- SACKETT, C., KIELPINSKY, D., KING, B., LANGER, C., MEYER, V., MYATT, C., ROWE, M., TURCHETTE, Q., ITANO, W., WINELAND, D., AND MONROE, C. 2000. Experimental entanglement of four particles. *Nature* 404, 256–258.
- SANIE, M., COTE, M., HURAT, P., AND MALHOTRA, V. 2001. Practical application of full-feature alternating phase-shifting technology for a phase-aware standard-cell design flow.
- SCHULMAN, L. J. AND VAZIRANI, U. 1998. Scalable NMR quantum computation. arXiv e-print quant-ph/9804060.
- SCHULMAN, L. J. AND VAZIRANI, U. 1999. Molecular scale heat engines and scalable quantum computation. In *Proceedings 31st Annual ACM Symposium on Theory of Computing (STOC '99)*, 322–329.
- SHOR, P. W. 1997. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comp.* 26, 5, 1484–1509.
- SKINNER, A. ET AL. 2002. Hydrogenic spin quantum computing in silicon: a digital approach. quant-ph/0206159.
- TAKAHASHI, Y., FUJIWARA, A., ONO, Y., AND MURASE, K. 2000. Silicon single-electron devices and their applications. In *IEEE International Symposium on Multiple-Valued Logic*.
- TUCKER, J. R. AND SHEN, T.-C. 2000. Can single-electron integrated circuits and quantum computers be fabricated in silicon? *International Journal of Circuit Theory and Applications* 28, 553–562.
- TURCHETTE, Q. A., HOOD, C. J., LANGE, W., MABUCHI, H., AND KIMBLE, H. J. 1995. Measurement of conditional phase shifts for quantum logic. *Phys. Rev. Lett.* 75, 4710.
- UCHIDA, K., KOGA, J., OHBA, R., AND TORIUMI, A. 2002. Programmable single-electron transistor logic for low-power intelligent Si LSI. In *IEEE International Solid-State Circuits Conference*.
- VANDERSYPEN, L. M., STEFFEN, M., BREYTA, G., YANNONI, C. S., CLEVE, R., AND CHUANG, I. L. 2000a. Experimental realization of order-finding with a quantum computer. *Phys. Rev. Lett.* December 15.
- VANDERSYPEN, L. M., STEFFEN, M., BREYTA, G., YANNONI, C. S., CLEVE, R., AND CHUANG, I. L. 2000b. Experimental realization of order-finding with a quantum computer. *Phys. Rev. Lett.*, to appear.
- VERHULST, A. S., LIVAK, O., SHERWOOD, M. H., HANS-MARTIN-VIETH, AND CHUANG, I. L. 2001. Non-thermal nuclear magnetic resonance quantum computing using hyperpolarized xenon. *Applied Physics Letters*, 79, 15.

- VION, D., AASSIME, A., COTTET, A., JOYEZ, P., POTHIER, H., URBINA, C., ESTEVE, D., AND DEVORET, M. H. 2002. Manipulating the quantum state of an electrical circuit. *Science* 296, 886.
- WASSHUBER, C. 2002. Single-electronics—How it works. How it's used. How it's simulated. In *IEEE International Symposium on Quality Electronic Design*.

Received October 2003; revised January 2004; accepted January 2004