Building Quantum Wires: The Long and the Short of it

Mark Oskin[†], Frederic T. Chong[‡], Isaac L. Chuang[*], John Kubiatowicz[°]
[†]University of Washington, [‡]University of California, Davis
[*]Massachusetts Institute of Technology, [°]University of California, Berkeley

Abstract

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. In this paper, we introduce a quantum wire architecture based upon quantum teleportation. We compare this teleportation channel with the traditional approach to transporting quantum data, which we refer to as the swapping channel. We characterize the latency and bandwidth of these two alternatives in a device-independent way and describe how the advanced architecture of the teleportation channel overcomes a basic limit to the maximum communication distance of the swapping channel. In addition, we discover a 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, which in turn imply a sparsely connected architecture of coarse-grained quantum computational elements. This is in direct contrast to the "sea of gates" architectures presently assumed by most quantum computing studies.

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 [28, 31, 13, 15, 41, 9, 17, 42]. Early-stage quantum computers have involved a small number of components (less than 10) and have utilized molecules in solution and trapped ions [47, 25, 35]. 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 [22, 50]; these even include ideas that merge atomic physics and silicon micromachining[24]. However, as the number of components grows, quantum computing systems will begin to require the same level of engineering as current computing systems. The same process we as computer architects do for classical silicon-based systems, of building abstractions and optimizing structure, needs to be applied to quantum technologies.

Even at this early stage, a general architectural study of quantum computation is important. By investigating the potential costs and fundamental challenges of quantum devices, we can help illuminate previously unforeseen obstacles of constructing a scalable quantum processor. We may also anticipate and specify important subsystems and techniques common to all implementations. Identifying these practical challenges early will help focus the ongoing development of fabrication and device technology. Developing abstractions for quantum technology and basic architectural concepts for it has proven to be quite fascinating.

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, in turn, 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 (composed of quantum bits or qubits) *cannot be copied* [31]. Instead, it must be *transported* from source to destination – destroying the information at the source and re-creating it at the destination. This fact changes our normal intuitions about the use of buffers to drive wires, repeaters to amplify signals, and fanout to distribute information. In particular, all wires must be *point-to-point* and can only *protect* information rather that amplifying it.

Quantum information can be encoded in a number of ways, such as the spin component of basic particles like protons or electrons, or in the polarization of photons. Thus, there are several ways in which we might transfer information. First, we might physically transport particles from one point to another. In a large solid-state system, the logical candidate for information carriers would be electrons, since they are highly mobile. Unfortunately, electrons are also highly interactive with the environment and hence subject to corruption of their quantum state, a process known as *decoherence*. Second, we might consider passing information along a line of quantum devices. This *swapping channel* is, in fact, a viable option for short distances (as discussed in Section 4), but tends to accumulate errors over long distances. In some ways, this solution resembles a quantum-cellular automata (QCA) [32] wire, except without duplication of data capabilities.

Proceedings of the 30th Annual International Symposium on Computer Architecture (ISCA'03)
1063-6897/03 $17.00 © 2003 IEEE

Over longer distances, we need something fundamentally different. We propose to use a technique called *teleportation* [7] and to call the resulting long-distance quantum wire a *teleportation channel* to distinguish from a swapping channel. Teleportation uses an unusual quantum property called *entanglement*, which allows quantum bits to interact instantaneously at a distance[1]. To understand the mathematical details and practical implications of teleportation, we will need to cover some background and prior art before returning to the subject in Section 2.3.

In the remainder of this paper, we will quantify the advantages and disadvantages of swapping channels versus teleportation channels. Realistic concerns such as quantum error-correction [43] for protecting information data errors and entropy exchange [37] for generating zeros and entangled pairs, greatly complicate things. Also important is an often-neglected facet of quantum computing systems — the fact that they depend upon classical signals for control of quantum operations. We will explore the fundamental tension between the scale at which quantum effects occur and the scale at which classical signals can be reliably routed. The architectural implications of this tension manifest themselves as a pitch-matching problem.

Overall, the contributions of this research are:

- We define the basic building blocks required to construct long and short quantum wires.
- We discover that the interface between classical control and quantum devices requires minimum wire lengths between fanout sites. We generalize these limitations in terms of the ratio of quantum and classical devices in a given technology and discuss the architectural implications of these limitations.
- We find that the latency and bandwidth of swapping channels are extremely sensitive to the length of the channel, but that teleportation channels do not exhibit the same sensitivity.

The remainder of this paper continues with a brief introduction to quantum computing in Sections 2 and 3. Section 4 introduces the swapping channels that can be constructed from solid-state technologies and presents an analysis of the scalability problems with these channels. Section 5 presents teleportation channels, our architectural solution to scalable quantum data transport. Section 6 discusses our future work in system bandwidth issues and in Section 7 we conclude.

2 Quantum Computing

We begin with a brief overview of the basic terminology and constructs of quantum computation. Our purpose is to introduce the language necessary for subsequent sections; in-depth treatments of these subjects are available in the literature [31].

[1] Although this property sounds suspiciously like "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.

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 $|\psi\rangle = \sum_x c_x|\mathbf{x}\rangle$, a weighted combination (a "superposition") of the basis vectors $|\mathbf{x}\rangle$, where the *probability amplitudes* c_x are complex numbers whose modulus squared sums to one, i.e. $\sum_x |c_x|^2 = 1$.

A single quantum bit is commonly referred to as a *qubit* and is described by the equation $|\psi\rangle = c_0|0\rangle + c_1|1\rangle$. Such a qubit might be represented, for example, by the nuclear spin of an atom. 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, but with different *relative phases* between the basis states.

Larger quantum systems can be composed from multiple qubits. For example, $|00\rangle$ is a valid two-qubit state, and so 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 possible 2^n classical states of the system. To compose multiple independent quantum systems together, the tensor product operator $\otimes$ is used, e.g., $a \otimes b$.

Unlike the classical case, however, where the total can be completely characterized by its parts, the state of larger quantum systems cannot be 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 the composite state $|\psi_A\rangle \otimes |\psi_B\rangle$. Entanglement does not exist classically, and the unique properties of entangled states are widely believed to be at the heart of what gives quantum computers their computational powers.

Another non-intuitive 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 (in the quantum computation and quantum information community) to adopt the *computational basis* states $|0\ldots00\rangle$, $|0\ldots01\rangle$, $|0\ldots10\rangle$, …, $|1\ldots11\rangle$, 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 $|\Psi\rangle$,

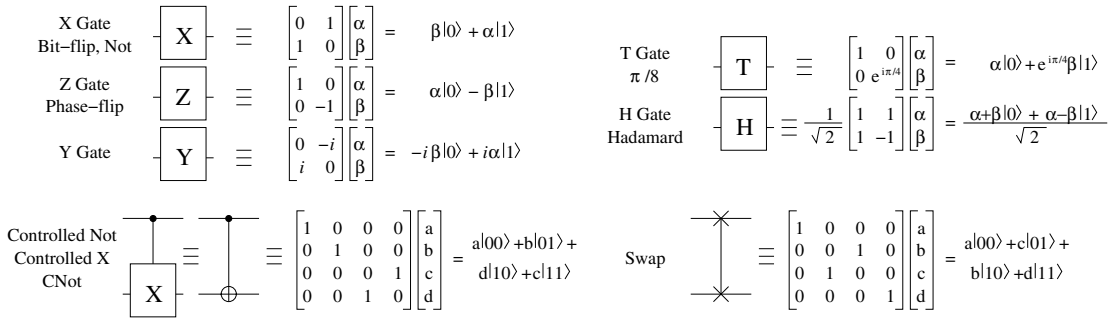

Figure 1. Basic Quantum Gates and their matrix representations.

above, is measured, the result is either $|00\rangle$ or $|11\rangle$, with equal probability; the outcomes $|01\rangle$ or $|10\rangle$ never occur.

Due to the probabilistic nature of measurement, designers of quantum algorithms must be very clever about how to get useful answers out of their computations. One method is to iteratively skew probability amplitudes in a qubit vector until the desired value is near $|1\rangle$ and the other values are close to $|0\rangle$. This technique is used in Grover's algorithm for searching an unordered list of n elements [18]. The algorithm goes through $\sqrt{n}$ iterations, at which point a qubit vector representing the keys can be measured. The desired element is found with high probability.

Another option in a quantum algorithm is to arrange the computation such that it does not matter which of many random results is measured from a qubit vector. This method is used in Shor's algorithm for factoring the product of two large primes [40], which is built upon the quantum Fourier transform, an exponentially fast version of the classical discrete Fourier transform. Essentially, the factorization is encoded within the period of a set of highly probable values, from which the desired result can be obtained no matter what value is measured. Since the tractability of factoring the product of two large primes is the basis of nearly all public-key cryptographic security systems, Shor's algorithm has received much attention.

For the interested reader, quantum algorithms for a variety of problems other than search and factoring have been developed: adiabatic solution of optimization problems (the quantum analogue of simulated annealing) [11], precise clock synchronization (using EPR pairs to synchronize GPS satellites) [21, 12], quantum key distribution (provably secure distribution of classical cryptographic keys) [6], and very recently, Gauss sums [46], testing of matrix multiplication (in $O(n^{1.75})$ steps versus the $O(n^2)$ required classically) [20], and Pell's equation [19].

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

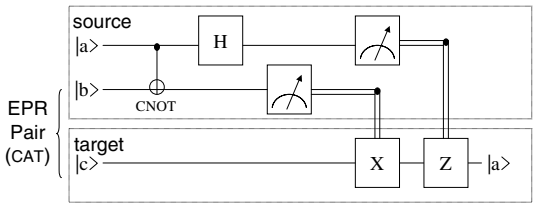

Figure 2. Quantum Teleportation of state $|a\rangle$ over distance. First, *entangled* qubits $|b\rangle$ and $|c\rangle$ are exchanged. Then, $|a\rangle$ is combined with $|b\rangle$ after which measurements produce two *classical* bits of information (double lines). After transport, these bits are used to manipulate $|c\rangle$ to regenerate state $|a\rangle$ at destination.

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 is something new to the quantum realm: it 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. Another important single-qubit gate, T, leaves $|0\rangle$ unchanged but multiplies $|1\rangle$ by $\sqrt{i}$. And analogous to the classical XOR gate is the quantum controlled-NOT (or CNOT) gate.

Together, these gates form a *universal set*: just as any Boolean circuit can be composed from AND and NOT gates, any polynomially describable multi-qubit quantum transform U can be efficiently approximated by composing these quantum gates into a circuit. In addition to these universal gates, one more important operator is the SWAP gate. SWAP can be implemented as 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, time goes from left to right, where single lines represent qubits, and double lines represent classical bits. A meter is used to represent measurement. By convention, black dots represent control terminals for quantum-controlled gates. The $\oplus$ symbol is shorthand for the target qubit of the CNOT gate (Figure 2).

2.3 Quantum teleportation

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 of information. Rather, it uses an entangled *EPR pair*, $|\Psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ [4].

Figure 2 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.

Why bother with teleportation when we end up transporting $|c\rangle$ anyway? Why not just transport $|a\rangle$ directly? First, we can pre-communicate 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.

3 Solid-State Technologies

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 [30, 50], trapped ions [29], photons [45], bulk spin NMR [48], and phosphorus impurities in silicon [22]. 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 [22, 42] schemes of phosphorus in silicon builds upon modern semiconductor fabrication and transistor design, drawing upon understood physical properties. To focus the presentation in this paper we begin our calculations with the Kane proposal, and then generalize to consider limits imposed by any solid-state technology. This quantum analysis proceeds in precisely the same manner that it would in the classical domain—by characterizing device technologies with a few underlying parameters.

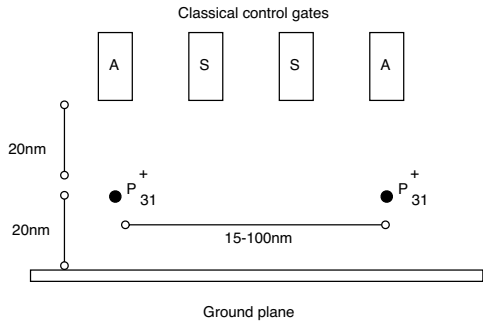

Figure 3. The basic quantum bit technology proposed by Kane [42]. Qubits are embodied by the nuclear spin of a phosphorus atom coupled with an electron embedded in silicon under high magnetic field at low temperature.

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, much as nuclear spins in molecules have been shown to be good quantum bits for quantum computation with nuclear magnetic resonance [15]. This quantum bit is classically controlled by a local electric field. The process is illustrated in Figure 3. Shown are two phosphorus atoms spaced 15-100 nm apart. This inter-qubit spacing is currently a topic of debate within the physics community, with conservative estimates of 15nm, and more aggressive estimations of 100nm. What is being traded off is noise immunity versus difficulty of manufacturing. For our study, we will use a figure (60nm) that lies between these two. We parameterize our work, however, to generalize for changes in the underlying technology.

Twenty nanometers above the phosphorus atoms lie three classical wires 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. Four different sets of pulse signals must be routed to each electrode to implement a universal set of quantum operations. The details of the pulses and quantum mechanics of this technique are beyond the scope of this paper and are described in [42].

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 4.2.3).

Quantum computing systems display a characteristic tension between computation and communication. Fundamentally, technologies that transport data well do so be-

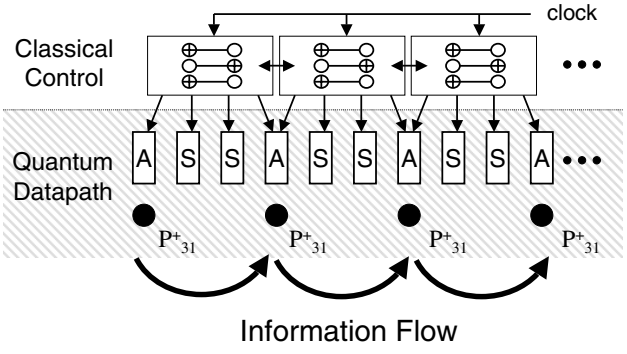

Figure 4. Short wires are constructed from successive qubits (phosphorus atoms). Information in the quantum data path is swapped from atom to atom by classical control. This localized control produces swapping behavior through a repeated series of three back-to-back CNOT operations.

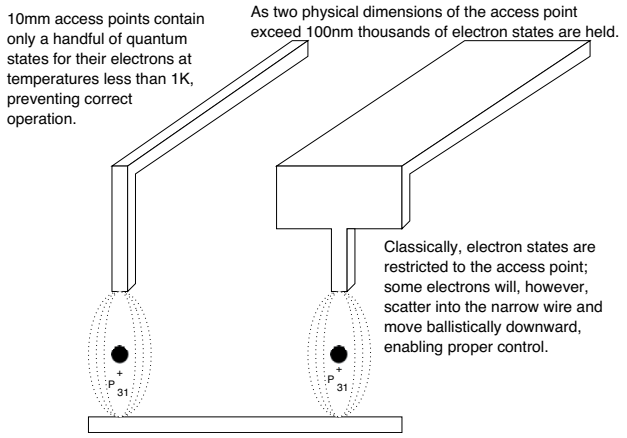

Figure 5. Quantization of electron states overcome by increasing the physical dimension of the control lines beyond 100 nm. The states propagate quantum-mechanically downward through access vias to control the magnetic field around the phosphorus atoms.

cause they are resistant to interaction with the environment or other quantum bits; on the other hand technologies that compute well do so precisely because they *do* interact. Thus, computation and communication are somewhat at odds.

In particular, atomic-based solid-state technologies are good at providing scalable computation but complicate communication, because their information carriers have nonzero mass. The Kane proposal, for example, represents a quantum bit with the nuclear spin of a phosphorus atom implanted in silicon. The phosphorus atom does not move, thus transporting this state to another part of the chip is laborious and requires carefully controlled swapping of the states of neighboring atoms. In contrast, photon-based proposals that use polarization to represent quantum states can easily transport data over long distances through fiber. It is very difficult, however, to get photons to interact and achieve any useful computation. Further, transferring quantum states between atomic and photon-based technologies is extremely difficult.

Optimizing these tensions, between communication and computation, between classical control and quantum effects, imply a structure to quantum systems. Rather than cover the gamut of quantum architecture we instead will focus on a very crucial architectural concept: a wire. Specifically, we begin by examining a short wire.

4 Short Wires

We begin by examining a "short" quantum wire. Section 4.2 shows that the basic short wire does not scale well, hence a more scalable approach appears Section 5.

In solid-state technologies, a line of qubits is one plausible approach to transporting quantum data. Figure 4 provides a schematic of a *swapping channel* in which information is progressively swapped between pairs of qubits in the *quantum datapath*—somewhat like a bubble sort[2]. Swapping channels require active control from classical logic, illustrated by the *classical control* plane of Figure 4.

4.1 Technical Challenges

As simple as it might appear, a quantum swapping channel 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 [23]. For applications where substantial monetary investment is not an issue, slowly placing a few hundred thousand phosphorus atoms with a probe device [16] may be possible. For bulk manufacturing the advancement of DNA or other chemical self-assembly techniques [1] may need to be developed. Note, 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 of these interactions is what will lead our analysis to the fundamental constraints on quantum computing architectures.

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 [3], or phase-shifted masks [36] will make such scales possible.

[2]For technologies that do not have an intrinsic swap operation, one can be implemented by three controlled-not gates performed in succession. This is a widely known result in the quantum computing field and we refer the interested reader to [31].

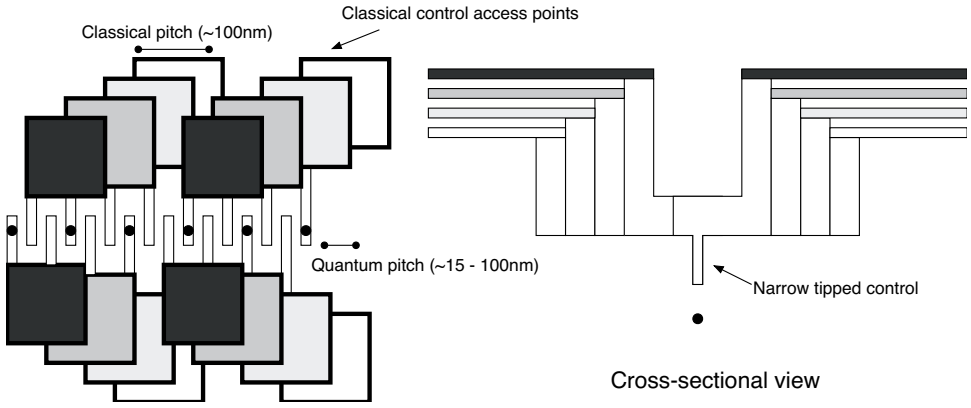

Figure 6. A linear row of quantum bits: In this figure (not drawn to scale) we depict access control for a line of quantum bits. On the left, we depict a "top down" view. On the right is a vertical cross-section which more clearly depicts the narrow-tipped control lines that quickly expand to classical dimensions.

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 one degree Kelvin. 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 such as conductance quantization and interference from ballistic transport [14].

Fortunately, many researchers are already working on low-temperature transistors. For instance, single-electron transistors (SET's) [27] are the focus of intense research due to their high density and low power properties. SET's, however, have been problematic for conventional computing because they are sensitive to noise and operate best at low temperatures. For quantum computing, this predilection for low temperatures is exactly what is needed! Tucker and Shen describe this complementary relationship and propose several fabrication methods in [44].

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 dominate over normal classical behavior. For example, in 100 nm wide polysilicon wires at 100 millikelvin, electrons propagate ballistically like waves, through only one conductance channel, which has an impedance given by the quantum of resistance, $h/e^2 \approx 25\ \mathrm{k}\Omega$. 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 thumb[3] 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 5. Here, wide wires terminate in 10 nm vias that act as local gates above individual phosphorus atoms.

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

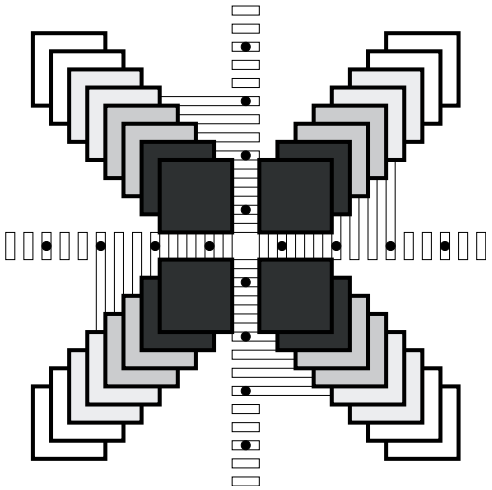

Figure 7. Intersection of quantum bits. In this simplified view, we depict a four-way intersection of quantum bits. An diamond shaped junction is also needed to densely pack junction cells.

Producing a line of quantum bits that overcomes all of the above challenges is possible. We illustrate a design in Figure 6. Note how access lines quickly taper into upper layers of metal and into control areas of a classical scale. These control areas can then be routed to access transistors that can gate on and off the frequencies (in the 10's to 100's of MHz) required to apply specific quantum gates.

Of course, any solution for data transport must also support routing. Routing is not possible without fanout 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 quantum bits. We illustrate the quantum intersection in Figure 7. This configuration is similar to Figure 6 except that the intersection creates a more challenging tapering.

4.2 Analysis

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.

4.2.1 Pitch Matching

Our first constraint is derived from the need to have classical control of our quantum operations. As previously discussed, we need a minimum wire width to avoid quantum effects in our classical control lines. Referring back to Figure 7, we can see that each quadrant of our four-way intersection will need to be some minimum size to accommodate access to our control signals.

Recall from Figure 3 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 5). Given this structure, it is possible to analytically derive the minimum width of a line 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 feature size λ_c), with the minimum line segment being equal in length to twice the classical pitching, $150nm$ in our case, and the junction size equal to four times the classical pitching, $400nm$, 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 6) by 20nm. The total size of these offsets will be $100nm$ divided by the qubit spacing $60nm$ times the number of control lines 3 per qubit, times the offset distance of $20nm$. This number $100nm/60nm \times 3 \times 20nm = 100nm$ is divided by 2 because the access lines lines are spread out on each side of the wire. Hence, the minimum line segment will be $100 + 50nm$. Shorter line segments within larger, more specialized cells are possible.

Turning our attention to an intersection (Figure 7), 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 $100nm$, we must fit $100nm/60nm \times 3$ additional pads shifted in x and y within the single quadrant of our intersection. This leads to a quadrant size of $100 + 100nm/60nm \times 3 \times 20nm = 200nm$. 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 7. Obviously, the specific sizes will vary according to technological parameters and 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.

4.2.2 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 parameterize 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 $60nm$ between qubits, three control lines per bit of $100nm$ (p_c) each, and a λ_c of 5nm. 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 7), 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.

4.2.3 Latency and Bandwidth

Calculating the latency and bandwidth of quantum wires is similar but slightly different than it is for classical systems. The primary difficulty is decoherence—i.e. quan-

tum noise. Unlike classical systems, if you want to perform a quantum computation, you cannot simply re-send quantum data when an error is detected. The "no-cloning" theorem [31], according to which quantum states cannot be perfectly copied, prohibits transmission by duplication, thereby making it impossible to re-transmit quantum data if it is corrupted. Once the data is destroyed by the noisy channel, you have to start the entire computation over. To avoid this loss, quantum data is 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. Unfortunately, quantum systems will be so error-prone that they will execute right at the limits of their error tolerance [33].

Our goal is to provide a quantum communication layer which sits below higher level error correction schemes. We will discuss our future work, which is the interaction of this layer with quantum error correction and algorithms in Section 6. 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 μs to execute in the Kane technology. Thus, a single row of quantum bits has latency:

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

This latency can be quite large. A short 1 μm has a latency of 0.000017 seconds! On the plus side, the wire can be fully pipelined and has a sustained bandwidth of $1/1\ \mu\text{s} = 1\text{M}$ 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 an un-error 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 through our wire they arrive with a fidelity proportional to the latency, namely:

$$fidelity = e^{-k\times latency} \quad (2)$$

The true bandwidth is then proportional to the fidelity:

$$bandwidth_{true} = bandwidth_{physical} \times fidelity \quad (3)$$

Choosing a reasonable [4] value of $\lambda \approx 10^{-6}$, we find the true bandwidth of a wire to be:

$$1/1\ \mu\text{s} \times e^{-10^{-6}\times distance/60\ \text{nm}} \quad (4)$$

which for a 1 μm 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 [2] that a degradation of fidelity more than 10^{-4} makes arbitrarily long quantum computation theoretically unsustainable, with the practical limit being far higher [33]. This limit is derived from the Threshold Theorem, which relates the decoherence of a quantum bit to the complexity of correcting this decoherence [26, 34, 2].[5] Given our assumptions about λ, the maximum theoretical wire distance is about 6μm, and again the practical wire distance is about two orders of magnitude less than this.

4.2.4 Technology Independent Metrics

Our latency and bandwidth calculations require slightly more device parameters. Let T be the time per basic swap operation. Some technologies will have an intrinsic SWAP, and others will require synthesizing the swap from 3 CNOT operations. 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 swapping channel wire equal to:

$$latency = T \times D \quad (5)$$

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

$$bandwidth_{true} = \frac{1}{T}e^{-\lambda D} \quad (6)$$

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 swapping channel is the distance when the fidelity drops below the critical threshold:

$$distance_{max} = log_e(1 - C)/ - \lambda \quad (7)$$

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.

[4] This value for λ is calculated from a decoherence rate of 10^{-6} per operation, where each operation requires 1 μs. It is aggressive, but not too unreasonable for phosphorus atoms in silicon. We refer the interested reader to [31].

[5] By "practical" we mean without an undue amount of error correction. The threshold theorem ensures that theoretically we can compute arbitrarily long quantum computations, but the practical overhead of error correction makes the real limit 2-3 orders of magnitude higher [33].

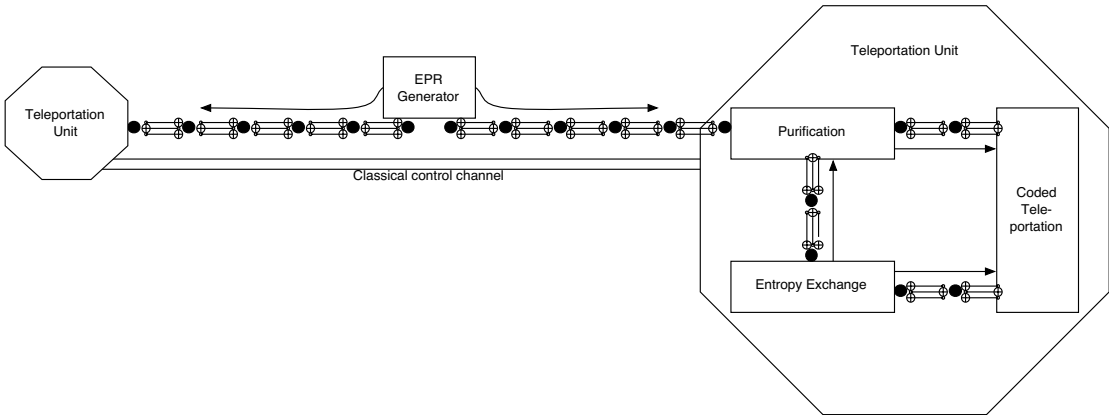

Figure 8. Architecture for a Quantum Wire: Solid double lines represent classical communication channels, while chained links represented a quantum swapping channel. Single lines depict the direction in which the swapping channel is being used for transport.

5 Long Wires

In this section, we introduce an architecture for long quantum wires, shown in Figure 8. These wires make use of the quantum primitive of teleportation. Teleportation involves pre-communication of EPR pairs, followed by a combination of quantum measurement and classical communication to destroy a quantum state at one end of a wire and re-create it on the other end. The key is that the pre-communication can be pipelined. Furthermore, teleportation allows quantum wires to convert quantum data between components that use different error correction codes, a conversion that is impractical without teleportation. In the next few sections, we provide a brief introduction to the core architectural components of this wire.

5.1 Basic Building Blocks

Although teleportation and the mechanisms described in this section are known in the literature, what has been missing is the identification and analysis of which mechanisms form fundamental building blocks of a realistic system. 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 theses 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 physics of quantum computation requires that operations be reversible and conserve energy. The initial state of the system, however, must be created somehow. We need to be able to create zero states, denoted as "$|0\rangle$". Furthermore, errors cause qubits to become randomized; stated equivalently, entropy enters the system through decoherence caused by coupling with the external environment.

Where do these zero states come from? The process can be viewed as one of thermodynamic cooling. Distributed throughout a quantum processor are "cool" quantum bits in a nearly zero state. These can be created by pulling spin-polarized electrons (created, for example, using a standard technique known as optical pumping [23] [49] or directly using spintronics methods, with ferromagnetic materials and spin filters [23]) over the phosphorus atoms.

To arbitrarily increase this probability (and make an extremely cold zero state) we can use a variant of the purification technique described in Section 5.1.3. Specifically, we employ an efficient algorithm for data compression [38] [39] that gathers entropy across a number of qubits into a small subset of highly random qubits. As a result, the remaining quantum bits are reinitialized to the desired pure zero state $|0\rangle$.

5.1.2 EPR Generator

Constructing an EPR pair of quantum bits is straightforward. We start with two $|0\rangle$ state bits from our entropy exchange unit. A Hadamard gate is applied to the first of these quantum bits. We then take this transformed quantum bit that is in a half-way superposition of a zero and a one state and use it as the control bit for a controlled-NOT gate. The target bit that is to be inverted is the other fresh $|0\rangle$ quantum bit from the entropy exchange unit. A controlled-NOT gate is a bit like a classical inverter except the target bit is inverted if the control bit is in the $|1\rangle$ state. Using a control bit of $(|0\rangle + |1\rangle)/\sqrt{2}$ and a target bit of $|0\rangle$ we end up with a two bit entangled state of $(|00\rangle + |11\rangle)/\sqrt{2}$. The quantum bits in this state are called an EPR pair.

The overall process of EPR generation is depicted in Figure 9. Schematically the EPR generator has a single

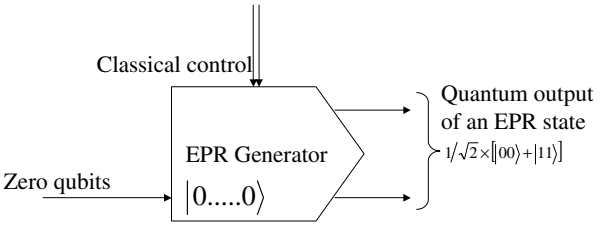

Figure 9. Quantum EPR generator: Solid double lines represent classical communication (or control), while single lines depict quantum wires.

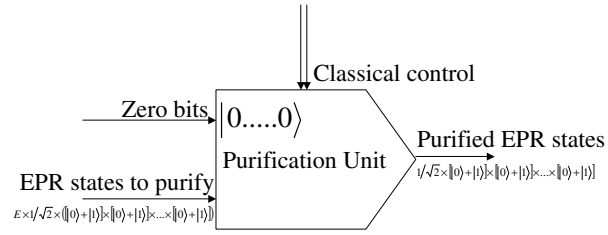

Figure 10. Quantum purification unit: EPR States are sufficiently regular that they can be purified at the ends of a teleportation channel.

quantum input and two quantum outputs. The input is directly piped from the entropy exchange unit and the output is the entangled EPR pair.

5.1.3 Purification unit

The final building block we require is the purification unit. This unit takes as input n EPR pairs which 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 [10, 5, 8].

Figure 10 depicts a purification block. The quantum inputs to this block are the input EPR states and a supply of $|0\rangle$ bits. The outputs are pure EPR states. Note that the block is carefully designed to correct only up to a certain number of errors; if more errors than this threshold occur, then the unit fails with increasing probability.

5.2 Analysis

Figure 8 illustrates how we use these basic building blocks and protocols for constructing a long wire. The EPR generator is placed in the middle of the wire and "pumps" entangled quantum bits to each end (via a pipelined swapping channel). These bits are then purified such that only the error-free qubits remain. Purification and teleportation consume zero-state qubits that are supplied by the entropy exchange unit. Finally, the coded-teleportation unit transmits quantum data from one end of the wire to the other using the protocol described in Section 2.3. 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 [38]:

$$\text{purification}_{\text{efficiency}} \approx \text{fidelity}^2 \tag{8}$$

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

$$1/1\ \mu\text{s} \times e^{-2\times10^{-6}\times\text{distance}/60\ \text{nm}} \tag{9}$$

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 $O(e^{-\lambda\times10})$. It is this latter number that does not change with wire length which makes an important difference. In the previous short-wire scheme we could not make a wire longer than 6μm. 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 simple wire has an effective bandwidth of zero at this length (for computational purposes).

The situation is even better when we consider latency. Unlike the simple wire, the wire architecture we propose allows for the pre-communication of EPR pairs at the sustainable bandwidth of the wire. These pre-communicated EPR pairs can then be used for transmission with a constant latency. This latency is roughly the time it takes to perform teleportation, or about $\approx$ 20 μs. Note this latency is much improved compared to the distance-dependent simple wiring scheme.

5.2.1 Technology Independent Metrics

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

$$latency \approx 10T \tag{10}$$

The bandwidth is:

$$bandwidth_{true} = \frac{1}{T}e^{-2\lambda D} \tag{11}$$

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 10's of millimeters on a side.

6 System Bandwidth

Our goal has been to design a reliable, scalable quantum communication layer that will support higher-level quantum error correction and algorithms functioning on top of this layer. A full description of error correction and quantum algorithms is beyond the scope of this paper. A key issue for future evaluation, however, is that the lower latency of our teleportation channel actually translates to even higher bandwidth when the upper layers of a quantum computation are considered. It is for this reason that long wires should not be constructed from chained swapping-channels and quantum "repeaters".

The intuition behind this phenomenon is as follows. Quantum computations are less reliable than any computation technology that we are accustomed to. In fact, quantum error correction consumes an enormous amount of overhead both in terms of redundant qubits and time spent correcting errors. This overhead is so large that the reliability of a computation must be tailored specifically to the run length of an algorithm. The key is that, the longer a computation runs, the stronger the error correction needed to allow the data to survive to the end of the computation. The stronger the error correction, the more bandwidth consumed transporting redundant qubits. Thus, lower latency on each quantum wire translates directly into greater effective bandwidth of logical quantum bits. For more information on quantum error correction and algorithms, we refer the reader to [31].

7 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 channel scales with distance and that swapping channels do not. 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.

Acknowledgements

Thanks to Dean Copsey, 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 Kubiatowicz, an NSF NER grant, and by a UC Davis Chancellor's Fellowship to Fred Chong.

References

[1] L. Adleman. Toward a mathematical theory of self-assembly. USC Tech Report, 2000.

[2] D. Aharonov and M. Ben-Or. Fault tolerant computation with constant error. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing*, pages 176–188, 1997.

[3] E. Anderson, V.Boegli, M. Schattenburg, D. Kern, and H. Smith. Metrology of electron beam lithography systems using holographically produced reference samples. *J. Vac. Sci. Technol.*, B-9, 1991.

[4] J. S. Bell. On the Einstein-Podolsy-Rosen paradox. *Physics*, 1:195–200, 1964. Reprinted in J. S. Bell, *Speakable and Unspeakable in Quantum Mechanics*, Cambridge University Press, Cambridge, 1987.

[5] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher. Concentrating partial entanglement by local operations. *Phys. Rev. A*, 53(4):2046–2052, 1996. arXive e-print quant-ph/9511030.

[6] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.

[7] C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. Wootters. Teleporting an unknown quantum state via dual classical and EPR channels. *Phys. Rev. Lett.*, 70:1895–1899, 1993.

[8] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin, and W. K. Wootters. Purification of noisy entanglement and faithful teleportation via noisy channels. *Phys. Rev. Lett.*, 76:722, 1996. arXive e-print quant-ph/9511027.

[9] C. H. Bennett and D. P. DiVincenzo. Quantum information and computation. *Nature*, 404:247–55, 2000.

[10] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters. Mixed state entanglement and quantum error correction. *Phys. Rev. A*, 54:3824, 1996. arXive e-print quant-ph/9604024.

[11] A. M. Childs, E. Farhi, and J. Preskill. Robustness of adiabatic quantum computation. *Phys. Rev. A*, (65), 2002.

[12] I. L. Chuang. Quantum algorithm for clock synchronization. *Phys. Rev. Lett.*, 85:2006, Aug 2000.

[13] D. P. DiVincenzo. Quantum computation. *Science*, 270(5234):255, 1995. arXive e-print quant-ph/9503016.

[14] D. K. Ferry and S. M. Goodnick. *Transport in Nanostructures*. Cambridge Studies in Semiconductor Physics & Microelectronic Engineering, 6. Cambridge University Press, Cambridge, 1997.

[15] N. Gershenfeld and I. Chuang. Quantum computing with molecules. *Scientific American*, June 1998.

[16] A. Globus, D. Bailey, J. Han, R. Jaffe, C. Levit, R. Merkle, and D. Srivastava. Nasa applications of molecular nanotechnology. *Journal of the British Interplanetary Society*, 51, 1998.

[17] J. M. Goodkind. Proposed fabrication of a quantum computer using electrons on helium. In *Second Annual SQuInT Workshop*, 2000. Poster Abstract.
[18] L. Grover. In *Proc. 28^{th} Annual ACM Symposium on the Theory of Computation*, pages 212–219, New York, 1996. ACM Press.
[19] S. Hallgren. *Quantum Information Processing '02 Workshop*, 2002.
[20] P. Hoyer. *Banff workshop on quantum algorithms*, 2002.
[21] R. Jozsa, D. Abrams, J. Dowling, and C. Williams. Quantum atomic clock synchronization based on shared prior entanglement. *Phys. Rev. Lett.*, pages 2010–2013, August 2000.
[22] B. Kane. A silicon-based nuclear spin quantum computer. *Nature*, 393:133–137, 1998.
[23] B. E. Kane, N. S. McAlpine, A. S. Dzurak, R. G. Clark, G. J. Milburn, H. B. Sun, and H. Wiseman. Single spin measurement using single electron transistors to probe two electron systems. *arXive e-print cond-mat/9903371*, 1999. Submitted to Phys. Rev. B.
[24] D. Kielpinsky, C. Monroe, and D. Wineland. Architecture for a large-scale ion trap quantum computer. *Nature*, 417:709, 2002.
[25] E. Knill, R. Laflamme, R. Martinez, and C.-H. Tseng. A cat-state benchmark on a seven bit quantum computer. *arXive e-print quant-ph/9908051*, 1999.
[26] E. Knill, R. Laflamme, and W. H. Zurek. Resilient quantum computation. *Science*, 279(5349):342–345, 1998. arXive e-print quant-ph/9702058.
[27] K. K. Likhareve. Single-eletron devices and their applications. *Proceedings of the IEEE*, 87, 1999.
[28] S. Lloyd. Quantum-mechanical computers. *Scientific American*, 273(4):44, Oct. 1995.
[29] C. Monroe, D. M. Meekhof, B. E. King, W. M. Itano, and D. J. Wineland. Demonstration of a fundamental quantum logic gate. *Phys. Rev. Lett.*, 75:4714, 1995.
[30] Y. Nakamura, Y. A. Pashkin, and J. S. Tsai. Coherent control of macroscopic quantum states in a single-cooper-pair box. *Nature*, 398:786–788, 1999.
[31] M. Nielsen and I. Chuang. *Quantum computation and quantum information*. Cambridge University Press, Cambridge, England, 2000.
[32] M. T. Niemier and P. M. Kogge. Exploring and exploiting wire-level pipelining in emerging technologies. In *International Symposium on Computer Architecture*, 2001.
[33] M. Oskin, F. Chong, and I. Chuang. Overhead reduction in a architecture for quantum computers. *IEEE Computer*, 35(1):79–87, 2002.
[34] J. Preskill. Fault-tolerant quantum computation. In H.-K. Lo, T. Spiller, and S. Popescu, editors, *Quantum information and computation*. World Scientific, Singapore, 1998.
[35] C. Sackett, D. Kielpinsky, B. King, C. Langer, V. Meyer, C. Myatt, M. Rowe, Q. Turchette, W. Itano, D. Wineland, and C. Monroe. Experimental entanglement of four particles. *Nature*, 404:256–258, 2000.
[36] M. Sanie, M. Cote, P. Hurat, and V. Malhotra. Practical application of full-feature alternating phase-shifting technology for a phase-aware standard-cell design flow. 2001.
[37] L. Schulman and U. Vazirani. Molecular scale heat engines and scalable quantum computation. In *31st STOC*, 1999.
[38] L. J. Schulman and U. Vazirani. Scalable NMR quantum computation. *arXive e-print quant-ph/9804060*, 1998.
[39] L. J. Schulman and U. Vazirani. Molecular scale heat engines and scalable quantum computation. *Proc. 31st Ann. ACM Symp. on Theory of Computing (STOC '99)*, pages 322–329, 1999.
[40] P. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proc. 35^{th} Annual Symposium on Foundations of Computer Science*, page 124, Los Alamitos, CA, 1994. IEEE Press.
[41] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comp.*, 26(5):1484–1509, 1997.
[42] A. Skinner et al. Hydrogenic spin quantum computing in silicon: a digital approach. *quant-ph/0206159*, 2002.
[43] A. Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77, 1996.
[44] J. R. Tucker and T.-C. Shen. Can single-electron integrated circuits and quantum computers be fabricated in silicon? *International Journal of Circuit Theory and Applications*, 28:553–562, 2000.
[45] Q. A. Turchette, C. J. Hood, W. Lange, H. Mabuchi, and H. J. Kimble. Measurement of conditional phase shifts for quantum logic. *Phys. Rev. Lett.*, 75:4710, 1995.
[46] W. van Dam and G. Seroussi. Efficient quantum algorithms for estimating gauss sums. *quant-ph*, page 0207131, 2002.
[47] L. M. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, R. Cleve, and I. L. Chuang. Experimental realization of order-finding with a quantum computer. *Phys. Rev. Lett.*, December 15, 2000.
[48] L. M. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, R. Cleve, and I. L. Chuang. Experimental realization of order-finding with a quantum computer. *Phys. Rev. Lett.*, to appear, 2000.
[49] A. S. Verhulst, O. Liivak, M. H. Sherwood, Hans-Martin-Vieth, and I. L. Chuang. Non-thermal nuclear magnetic resonance quantum computing using hyperpolarized xenon. *Applied Physics Letters*, 79-15, 2001.
[50] D. Vion, A. Aassime, A. Cottet, P. Joyez, H. Pothier, C. Urbina, D. Esteve, and M. H. Devoret. Maniplating the quantum state of an electrical circuit. *Science*, 296:886, 2002.