Quantum Wires: Architectural Implications of Quantum Data Transport in Silicon

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

Computing with quantum states has become an increasingly intriguing technological possibility. Prototype quantum computers using 4, 5 and 7 quantum bits have begun to appear using molecules in solution and trapped ions[1, 2, 3]. For true scalability and to exploit our tremendous historical investment in silicon, however, solid-state silicon quantum implementations are desirable.

Quantum wires which transport quantum data will be a fundamental system component in all anticipated silicon quantum architectures. Here, we propose a novel approach to low-latency, reliable communication through teleportation of error-coded quantum bits. By examining an end-to-end collection of quantum technologies to implement this approach in a quantum wire architecture, we are able to identify key challenges in solid-state quantum computing. In particular, we discover a fundamental tension between the scale of quantum effects and 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

There are many important problems for which all presently known solutions require exponential resources on a classical computer. Quantum computers can solve some of these problems with polynomial resources. This has motivated a great number of researchers to explore quantum information processing technologies [4, 5, 6, 7, 8, 9, 10, 11]. Once these systems have been demonstrated in the laboratory a considerable challenge remains in constructing a useful computational system. It is this engineering process that computer architects can contribute to, in the development of quantum computing architectures. As architects we work with and develop abstractions for technology and applications. Targeting this process at quantum computers is at first foreign, but ultimately fascinating.

Computer architects have a great deal to offer to quantum computation at this early stage. By investigating the areas, costs, and challenges of quantum devices we can help illuminate potential pitfalls along the way towards a complete quantum processor. We may also anticipate and specify important subsystems common to all implementations, thus fostering inter-operability. Identifying these practical challenges early will help focus research efforts into solutions.

Rather than start with a complete quantum processor design we introduce our architectural study with an investigation of a seemingly mundane subject: a wire. A quantum wire is a very different creature than a classical one. In some ways it resembles a quantum-cellular automata (QCA) [12] wire, with deep component-wise pipelining. This resemblance quickly fades as we consider the challenges of error-correction [13], entropy-exchange [14], and state-purification [15]. We put these technologies together with teleportation [16] and single-electron transistors [17] to construct a reliable, low-latency quantum wire.

An often-neglected facet of quantum computations is that they crucially depend upon classical signals for control of the quantum gates, particularly for fault-tolerant system constructions. We discover a 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, essentially a pitch-matching problem, are a primary focus of this work.

Overall, the contributions of this research are:

- An end-to-end study of the technologies required to construct a quantum wire.
- Identification of key components for silicon-based quantum computers and research challenges to achieve these components.
- Identification of design constraints arising from the quantum-classical interface and the error rate of quantum operations.

The remainder of this paper continues with a brief introduction to quantum computing in Section 2 and then an overview of our quantum wire architecture in Section 3. We then describe basic quantum gates and protocols in Section 4, followed in Section 5 by the building blocks necessary to implement our wire architecture. We then derive our design constraints in Section 6, followed by a discussion of alternative technologies in Section 7, research challenges in Section 8 and conclusions in Section 9.

2 Background

Quantum information systems can be a mathematically-intense subject. However, a great deal can be accomplished with just a simple model of several abstract building blocks, which include quantum bits (qubits), how they transform via quantum gates, the role of quantum algorithms, and the available implementation technologies and their imperfections. Here, we shall follow this approach; for in-depth treatments of the material see for example [18, 9, 19].

Quantum computation seeks to exploit the physics of quantum phenomena to achieve computation that scales exponentially with data size. The basic building block is a quantum bit, referred to as a *qubit*, that is represented by nanoscale physical properties such as nuclear spin. While a bit in a classical computer represents either zero or one, a qubit can be thought to simultaneously represent both states. More precisely, the state of a qubit is described by *probability amplitudes*, which only turn into probabilities upon external observation. Unlike classical probabilistic computation, probabilities for different computational pathways can cancel each other out through a kind of interference, because the amplitudes can be negative, and the probabilities are determined by the square of the amplitudes.

The key is that quantum computers directly manipulate probability amplitudes to perform a computation, and multiple qubits form vectors that can represent 2^n amplitudes with n qubits. In other words, a two-qubit vector simultaneously represents the states 00, 01, 10, and 11 – each with some probability when measured. Each additional qubit in a qubit vector doubles the number of amplitudes represented. The work of a quantum computer is to manipulate these qubit vectors and their associated amplitudes in a useful manner. Manipulation of these qubit vectors leads to what is often called *quantum parallelism*, a useful way to begin thinking about what gives quantum computers such high potential speedups over classical computers. The difficulty is that we generally cannot look at the answer until the end of a computation, and then we only get a random value from the vector! More precisely, measuring a qubit vector collapses it into a probabilistic classical bit vector, yielding a single state randomly selected from the exponential set of possible states. For this reason, quantum computers are best at "promise" problems – applications where the answer can be easily verified.

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 and the other values are close to 0. This is used in Grover's algorithm for searching an unordered list of n elements [20]. The algorithm goes through $\sqrt{n}$ iterations, at which point a qubit vector representing the keys can be measured. The desired key 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 prime factorization of large numbers [21], 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 prime factorization of large numbers is the basis of nearly all modern cryptographic security systems, Shor's algorithm has received much attention.

3 Overview

While quantum algorithms motivate interest in quantum computation, any architecture to support such computation will require a mechanism for transporting quantum data. This is the focus of our study. We introduce a novel architecture for quantum wires, shown in Figure 1. These wires make use of the quantum primitive of teleportation. Teleportation allows us to pre-communicate EPR pairs (a special pair of quantum states described later), and then use classical communication and quantum measurement to destroy a quantum state on one end of a wire and re-create it on the other end using the EPR pair for transport. The key is that the pre-communication can be done in a pipelined manner. The queues in the figure illustrate where this pipelining occurs throughout the architecture. Furthermore, teleporation allows our 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 quantum operations and protocols, and then describe the building blocks needed to implement our quantum wire architecture.

4 Basic Quantum Operations and Protocols

In this section, we introduce a few basic quantum operations and use these to describe two important quantum protocols: teleporation and error correction.

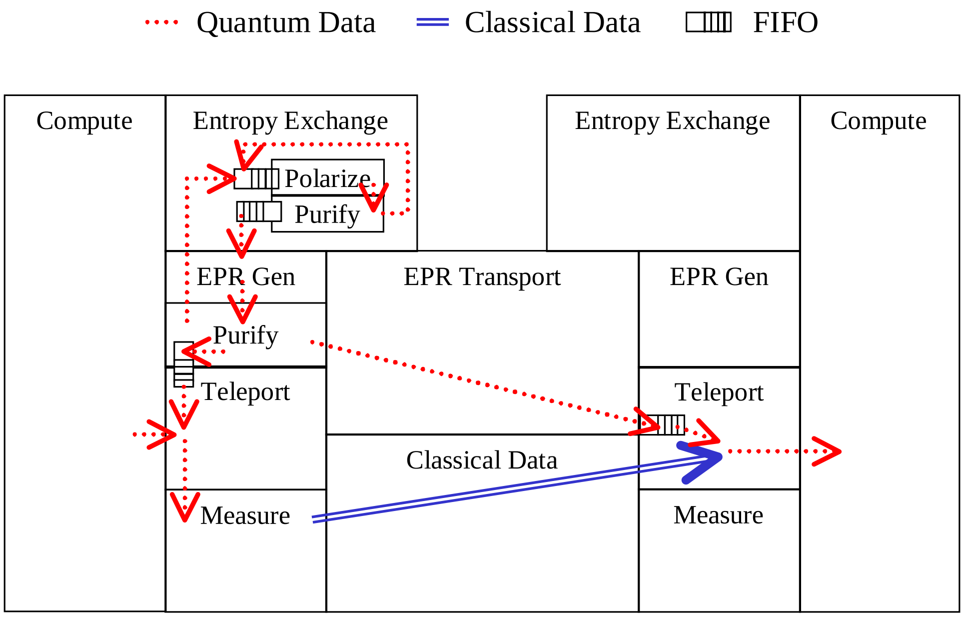

Figure 1: Architecture for a Quantum Wire

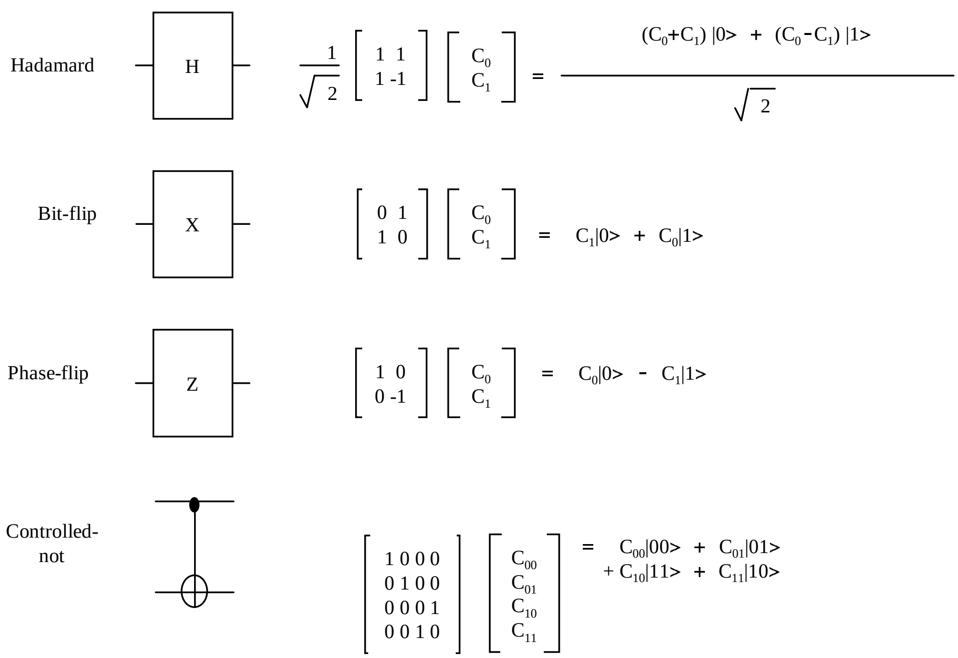

Figure 2: Basic quantum operations

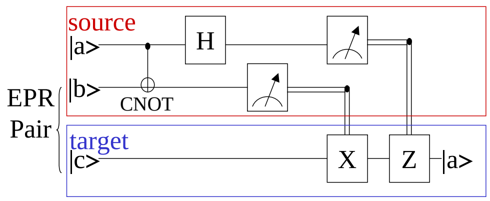

Figure 3: Quantum Teleportation

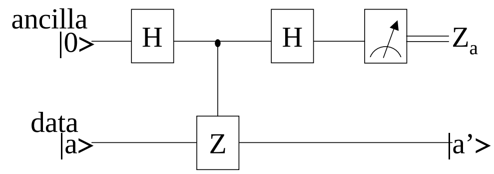

Figure 4: Syndrome measurement for error correction

4.1 Quantum Operations

Figure 2 gives a few basic quantum operations that we will use in our quantum wire architecture. These include one-bit operations such as the Hadamard, bit-flip, and phase-flip; as well as the two-bit controlled-not. These are given in both their circuit representation and their matrix representation. The matrix representation involves multiplying the operator times the amplitude vector of the quantum states. c_i^2 gives the probability of state i (commonly denoted as $|i\rangle$ in quantum information science) and the sum of all these probabilities must equal one. Preserving this sum is equivalent to conserving energy and requires that all operations be reversible. c_i are probability amplitudes, and valid states include not only $|0\rangle$ and $|1\rangle$ but also

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \tag{1}$$

which denotes an equal superposition of $|0\rangle$ and $|1\rangle$.

The bit-flip gate X exchanges the probabilities of the two states $|0\rangle$ and $|1\rangle$ (analogous to the classical NOT), while the phase flip gate Z changes the sign between them. The Hadamard takes the two states and mixes them to produce an equal superposition state. The controlled-not does a bit-flip iff the control qubit is in the $|1\rangle$ state. These basic gates, along with the measurement of qubits, form the basic operations we will use for data transport. For further discussion of quantum gates and circuits the reader is referred to [5].

4.2 Teleportation

Contrary to its science fiction counterparts, quantum teleportation is not the instantaneous transmission of information. Rather, it is the re-creation of a quantum state at a destination using some classical bits that must be communicated along conventional wires or other mediums. In order for this to work, we need to pre-communicate an EPR pair (named for Einstein, Podolsky and Rosen), in which the state of a pair of qubits is $(|00\rangle + |11\rangle)/\sqrt{2}$ [22]. This is known as an *entangled* state because statistics from measuring this state after performing various operations to it give results unobtainable by classical correlated bits. The non-classical properties of entanglement lies at the heart of teleportation.

Figure 3 gives an overview of the teleporation process. We start with an EPR pair at the source end of the wire. We separate the pair, keeping one qubit (b) at the source and transporting the other (c) to the destination (quantum data transport is denoted by the solid lines). When we want to send a quantum bit of data (a), we first interact a with b using a Hadamard gate. We then measure the phase and the amplitude a (measurement is denoted by the meter symbols), send the two results (each one bit) to the destination classically (classical communication is denoted by the double lines), and use those results to re-create the correct phase and amplitude in c such that it takes on the state of a. The re-creation of phase and amplitude is done with controlled-X and Z gates, which perform the same function as the gates described in Figure 2 but contingent on a classical control bit (the measurements of a and b).

Note that the original state of a is destroyed once we take our two measurements. This is consistent with the "no-cloning" theorem, which states that a quantum state can not be copied. Intuitively, since c has a special relationship with b, interacting a with b makes c resemble a, modulo a phase and/or amplitude error. The two measurements allow us to correct these errors and recreate a at the destination.

Why bother with teleporation when we end up transporting c anyway? Why not transport a directly? There are three reasons. 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 c has known properties, we can use the purification unit described in Section 5.4.4 to remove pairs that were damaged during transport. Third, we can use teleportation to convert data to different error coding schemes.

4.3 Quantum Error Correction

Without error correction to keep quantum states coherent, the timescales of quantum phenomena would make useful computation and communication impossible. The problem is that noise causes quantum superposition states to collapse quickly; this is known as the decoherence problem[23]). In fact, only recently have usable codes been developed and sustainable quantum computation been shown to be possible. The key is to allow the basic operations of quantum computing to be applied directly to coded data without decoding and re-encoding

Z_1Z_2	Z_2Z_3	Error Type	Action
+1	+1	no error	no action
+1	-1	bit 3 flipped	flip bit 3
-1	+1	bit 1 flipped	flip bit 1
-1	-1	bit 2 flipped	flip bit 2

Table 1: Phase correction for a 3-qubit code

the data.

Quantum error correction has much in common with its classical counterpart. A logical qubit is encoded redundantly in a number of physical qubits. The syndrome of the physical qubits can be measured to determine whether an error may need to be corrected. One difficulty in quantum codes, however, is that the physical qubits can not be measured without destroying their state! We can get around this by using an ancilliary zero qubit (shown in Figure 4) which is interacted with the original physical qubit (a). We can then measure the ancilla to determine the syndrome of a. What is interesting is that a is not unaffected by this interaction and measurement. The state of a actually becomes a', in which a' is actually the correct value or the value with the opposite phase. So syndrome measurement actually accomplishes two tasks: the syndrome is measured and the *continuous* errors in a are transformed into *discrete* errors which are easy to fix. Table 1 gives a simple example of how these fixes are done for a 3-qubit Shor code [24], where Z_n is the syndrome for the phase of the n-th physical qubit. Since errors can occur in both phase and amplitude, we perform a similar operation using the X gate to measure and correct amplitude.

The error code we expect to use is the 7-bit Steane code [13], which encodes 1 logical qubit in 7 physical qubits. The code is actually applied recursively to achieve adequate reliability in a given technology. The Steane code is the smallest code in which it is easy to directly perform basic quantum operations. Other codes, however, may be more efficient for tasks such as storing quantum data in memory. Unfortunately, it can be quite difficult to convert data between codes. Decoding and re-encoding the data will randomly distribute errors in the new code and compromise reliability by propagating errors between physical bits. Teleporation, however, can take data from source bits encoded in one code and re-create that data in destination bits encoded in another code. This process avoids undesirable propagation of errors and gives us an excellent translation process between codes.

5 Basic Building Blocks

We now turn our attention to the basic building blocks necessary to implement the operations and protocols used in our quantum wire architecture. We base our work on the quantum bit device technology proposed by Kane et al [10]. Our goal

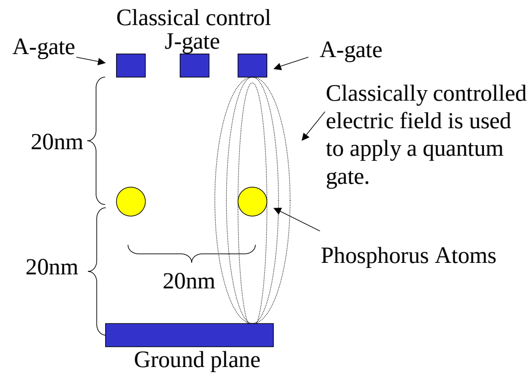

Figure 5: The basic quantum bit technology proposed by Kane et. al [10].

in this section is to build up a set of useful building blocks, cells if you will, that can be put together to form a complete computing system. We start our discussion with two universally useful cells: a line of qubits and an intersection of such lines. We use these basic quantum cells to construct larger macro blocks of useful functions including: an entropy exchange unit, a measurement device, an EPR generator, and a purification unit.

5.1 Phosphorus Atoms for Quantum Bits — Kane

In order to focus the discussion of quantum architecture around a technology we utilize the quantum bit proposal from Kane et al [10]. This device proposal is one of the most widely discussed within the quantum physics community as a potential candidate for silicon-based quantum computing. While this research generalizes beyond this single device technology, it helps to focus the discussion if we narrow in on a candidate device proposal.

Kane proposed that the nuclear spin of a phosphorus atom 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[7]. This quantum bit is classically controlled by a local electric field. The process is illustrated in Figure 5. Shown are two phosphorus atoms spaced 20 nm apart. Twenty nanometers above the phosphorus atoms lie three classical wires that are spaced 10 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 these universal quantum operations.

The details of the pulses and quantum mechanics of this technique are beyond the scope of this paper and clearly described in [10]. Our goal is to describe this technology into a useful set of abstractions. We start with the most basic of architectural components: a row of qubits.

5.2 A line of quantum bits

A line of qubits allows us to transport quantum state by progressively swapping state between pairs of qubits (a swap can be implemented by 3 controlled-not's). By classical standards the idea of a row of bits seems trivial. However, in the quantum domain it is an enormous technical challenge. The first hurdle is the actual placement of the phosphorus atoms themselves. The leading work in this[25] has involved precise ion implantation through masks, and manipulation of single atoms on the surface of silicon. For applications where substantial monetary investment is not an issue, slowly placing a few hundred thousand phosphorus atoms with a probe device [26] may be possible. For bulk manufacturing the advancement of DNA or other chemical self-assembly techniques [27] may need to be developed.

The second challenge is the scale of the classical control lines. Each wire lead above the phosphorus atoms is only 5 nm in width. While this is difficult, we expect with either electron beam lithography [28], or phase-shifted masks [29] such scales will be possible.

The third 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 5 nm classical control lines begin to exhibit quantum-mechanical behavior such as conductance quantization and interference from ballistic transport[30].

A solution already exists for the transistors. Reliable, small classical switching devices can be constructed from single-electron transistors (SET's) [17]. The control lines are a more difficult issue. In order to minimize the quantum effects in the control lines they will likely have to quickly taper into a large enough dimension in two axes such that quantum effects will not dominate. Dimensions exceeding 100 nm, estimated from the Fermi wavelength electrons in gold (at low temperature), will likely be sufficient.

Constructing a line of quantum bits that overcomes these challenges is possible. We illustrate a design in Figure 6. Note how the access lines quickly taper into the upper layers of metal into reasonably sized control areas. These control areas can then be routed to access transistors that can gate on and off the frequencies required to apply specific quantum gates.

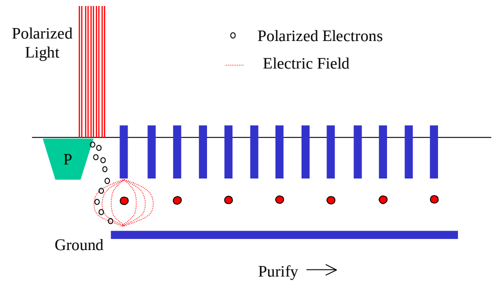

Figure 8: Entropy exchange with optical pumping

5.3 An intersection of quantum bits

An important capability for data transport is fanout. Fanout is not possible without intersections. Here, we extend our linear qubit discussion to a four-way intersection capable of supporting sparsely intersecting topologies of quantum bits. We illustrate the quantum intersection in Figure 7. This is very similar to Figure 6 except the intersection creates a slightly more challenging tapering. In Section 6, we shall see that the control lines for these intersections creates a pitch-matching problem that constrains minimum wire length.

5.4 Quantum Macro-blocks

The previously described line and intersection of qubits form the basis for a universal set of cells that can be put together to form a quantum processor. There are, however, four additional structures that are used frequently enough that they can be considered macro-blocks of a quantum computer. These macro-blocks are constructed from the linear and intersection structures previously described. Note that the description of the macro blocks takes on a classical feel at times with varying numbers of input and output ports. The reader should note, however, that all operations (except measurement) are inherently reversible, and the varying relationship of the input / output ports reflects only the logical meanings of the macro block. We use this as a descriptive technique.

5.4.1 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. Error correction, described later in Section 4.3, can be viewed as a process that consumes a constant stream of zero states and outputs both corrected data and a stream of high entropy waste states.

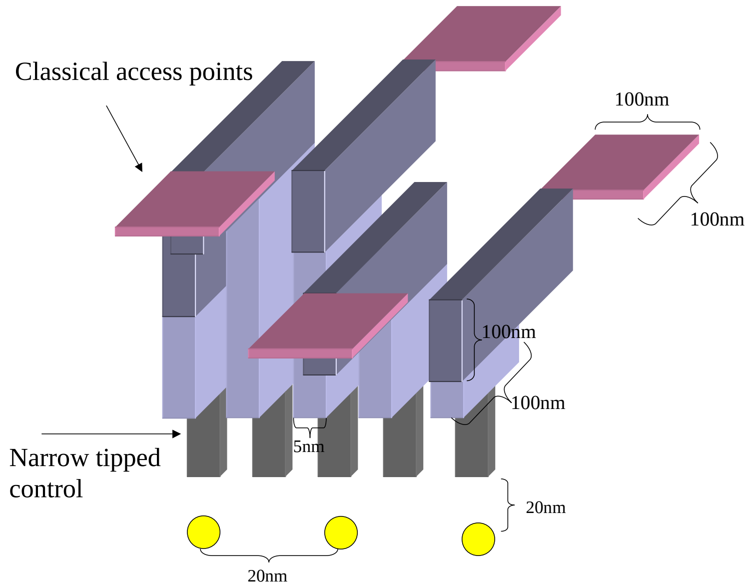

Figure 6: A Linear row of quantum bits. In this 3D perspective, the narrow-tipped classical control lines are shown expanding to classical dimensions. Once expanded they are routed off to the side to access vias.

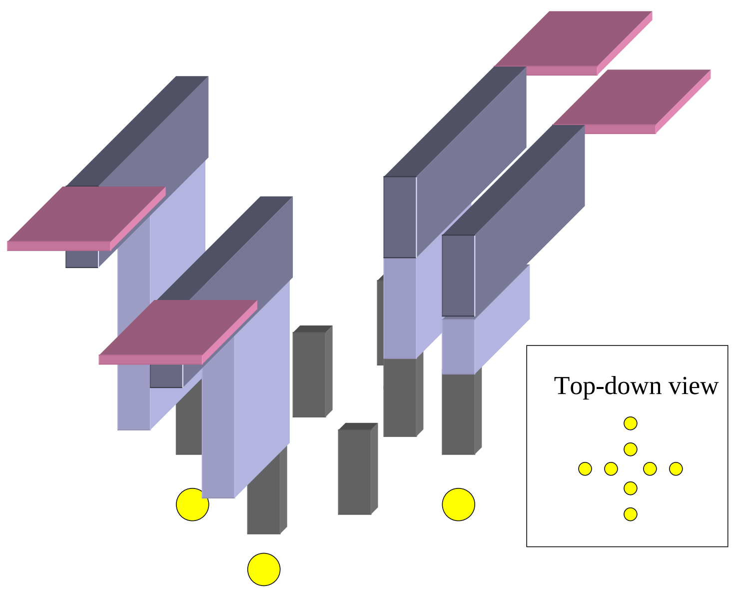

Figure 7: Intersection of quantum bits. In this simplified 3D view, we depict a four-way intersection of quantum bits. Half of the control lines are omitted for clarity. Off to the right is the top-down view of the phosphorus atoms themselves.

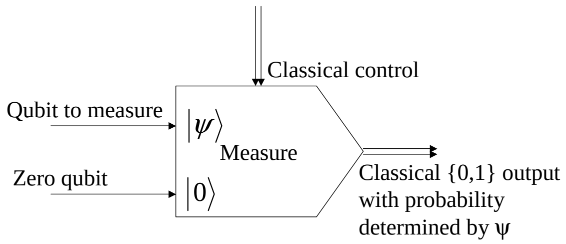

Figure 9: Schematic for a quantum measuring circuit component.

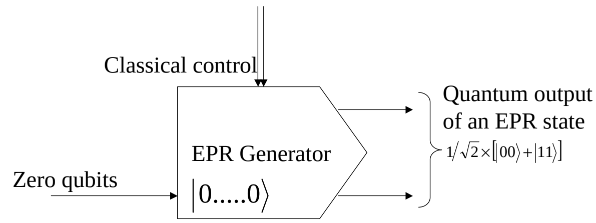

Figure 10: Schematic for a quantum EPR generator.

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 are created by pulling spin polarized electrons (created, for example, using a standard technique known as optical pumping[25, 31] (Figure 8) or directly using spintronics methods, with ferromagnetic materials and spin filters[25]) over the phosphorus atoms.

The electrons can transfer their polarized state to the phosphorus atoms with high probability. To increase this probability arbitrarily (and thus make a really cold zero state) we use a variant of the purification technique described in Section 5.4.4. Specifically, we employ an efficient algorithm for data compression[32, 33] which gathers entropy across a number of qubits into a small subset of highly random quantum bits. As a result, the remaining quantum bits are reinitialized to the desired pure zero state $|0\rangle$.

5.4.2 Measurement unit

Perfect measurement in quantum systems is a challenge. Measurement probabilistically reduces quantum superpositions (e.g. states simultaneously in a balance of $|0\rangle$ and $|1\rangle$) to specific classical states (e.g. 0 and 1). This works by coupling a large classical system to the small quantum system, but in reality making ideal coupling is complicated by practical topological and temporal constraints. Thus, errors occur in measurement. Error correction techniques, which are the basis for sustainable quantum computation require a reliable measurement process. Fortunately there are methods for increasing measurement reliability through repetition.

Measurement of a single qubit can be performed by first entangling it with a second reference qubit in the $|0\rangle$ state (freshly supplied from our entropy exchange unit). We then directly measure this new bit and obtain a result. This result has some error; say this is p. The act of performing the measurement collapses the entanglement between the new quantum bit and the actual bit we want to know the state of. More significantly it also collapses any superposition or entanglement that quantum bit had. We can then repeat the process starting with another $|0\rangle$ quantum bit. Each time we repeat this process the accuracy of the overall measurement is increased. Eventually after k repetitions the measurement result has an error rate of only p^k. Usually only 2-3 repetitions are required.

This process is used throughout any quantum computer and we construct a dedicated node to perform it. Besides classical control, this dedicated node has two quantum inputs and one classical output. The first quantum input is the quantum bit that is to be measured, and the second quantum input is a direct pipe from the entropy exchange unit. The classical output is the binary $\{0, 1\}$ measurement result. This measurement cell is depicted in Figure 9.

The construction of a perfect measurement device along these lines is an open problem for solid state quantum computers. The most likely scenario involves placing a quantum bit in a zero state nearby the bit to be measured[25]. Polarized electrons are then pulled past these two phosphorus atoms and collected by two specially located metal charge collectors. In this setup the collector with the largest charge is indicative of the most likely measurement result.

5.4.3 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.

This process can be generalized to form arbitrarily large CAT states. To do this we use another fresh $|0\rangle$ quantum bit as a target and one of the bits from the EPR pair as the control for another control-NOT gate. After applying this gate we end up with the three bit entangled state of $1(|000\rangle + |111\rangle)/\sqrt{2}$. This process can be extended to create a CAT state of any length.

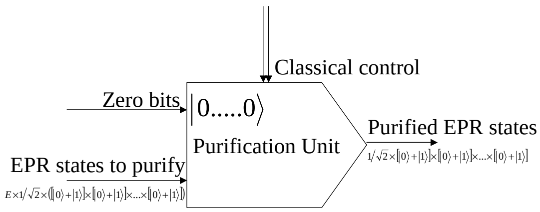

Figure 11: Schematic for a quantum purification unit.

The overall process of EPR generation is depicted in Figure 10. Schematically the EPR generator has a single quantum input and two quantum outputs. The input is directly piped from the entropy exchange unit and the output is the entangled EPR pair.

5.4.4 Purification unit

The final macro 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 [15, 34, 35].

Figure 11 depicts a purification macro block. The quantum inputs to this macro block are the input EPR states, and a fresh supply of $|0\rangle$ bits. The outputs of the macro block are the 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.

6 Design Constraints

Given our building blocks, quantum wire architecture, and implementation technologies; we can derive two important design constraints. These constraints are the classical-quantum interface boundary and the latency/bandwidth charactoristics of quantum wires.

6.1 Pitch Matching

Our first constraint is derived from the need to have classical control of our quantum operations. As discussed in Section 5.2, 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 accomodate access to our control signals.

Each qubit has two control lines associated with it, an A-gate and a J-gate [10]. Each of these control lines must quickly expand from a thin $5nm$ narrow tip into a $100nm$ access point in an upper metal layer. Analytically it is possible to derive the minimum width of a wire of quantum bits, and the size of a four-way intersection. We begin with a line of qubits:

Let N be the number of qubits along the line segment. Then we need to fit in $2N$ classical access points of $100nm$ in demension. Hence the minimum line segment is atleast $200nm$ (10 qubits) to attiqeuetly space the access points in the upper metal layers. The width of the line segment is determined by the number of control lines ($2N = 20$) and the spacing of the access points. The access points are offset by $10nm$ between metal layers to allow for vias. We estimate the minimum width of a quantum wire to be $20 \cdot 10nm + 100nm = 300nm$. Shorter line segments within larger specialized cells are possible.

Turning our attention to an intersection, let N be the number of qubits along each "spoke" of the junction. We need to fit $2N$ classical access points in a space of $(20nm \cdot N)^2$, where each access point is atleast $100nm$ on a side. As with the case of a linear row of bits, within a single metal layer these access points are spaced $100nm$ apart, with a $10nm$ shift between layers for via access. For this minimum size calculation we assume all classical control lines are routed in parallel, albit spread across the various metal layers. Let L be the number of metal layers, and $H_{spacing}$ to be the number of access pad groups per quadrant in the intersection primitive. Then the length along one dimension is:

$$(20\ \text{nm} \cdot N) = 50\ \text{nm} + 2 \times 100\ \text{nm} \times H_{spacing} \qquad (2)$$

and along the other dimension it is:

$$(20\ \text{nm} \cdot N) = 50\ \text{nm} + 100\ \text{nm} + 10\ \text{nm} \times L \qquad (3)$$

while the total area is:

$$(2 \cdot N) = H_{spacing} \times L\,. \qquad (4)$$

This implies that $H_{spacing} = 2$, $L = 23$ and $N = 22$. Therefore, the minimum size four way intersection is 44 quantum bits in each direction. 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.

6.2 EPR Bandwidth

Our second constraint involves the quantum and classical bandwidth requirements to support quantum data transport. Transporting classical data is easy, so transporting EPR pairs will be the limiting factor.

Our EPR pairs will be recursively encoded in the 7-bit Steane code for error correction. Let k be the number of levels of recursion necessary for an error rate per operation of $p = 1 - e^{-\lambda}$. This error rate is determined by the decoherence rate of the quantum bits. We start our calculation from the celebrated threshold theorem for fault-tolerant quantum computation, a key result which uses recursive error correction to show that reliable quantum computation can be performed using unreliable components as long as the error rate p is below a certain threshold [36, 37, 38]. The principal bound at the heart of this theorem is:

$$\frac{(c \cdot p)^{2^k}}{c} \leq \frac{\epsilon}{p(n)} \tag{5}$$

where $c \approx 4000$ is a constant related to the overhead and complexity of the error correction process, ϵ is the desired probability of obtaining a correct answer from the algorithm, $p(n)$ is the space-time complexity of the algorithm, and n describes the problem size. Assuming reasonable values, say $\epsilon = 0.95$, $p \approx 10^{-8}$, and Shor's algorithm for factoring on a 1024 bit number we find $k = 3$. This implies that the Steane code we use will have to recursively encode its error bits three times. So the total number of physical qubits used to store one logical quantum bit is $7^k = 343$. Note that this bound, and the threshold theorem, also assume optimal parallelism in the classical computation assisting the quantum information processing!

Let B be the desired bandwidth of the quantum wire and D be its distance in qubits. The efficiency of transmitting EPR pairs across a wire of length D is estimated to be:

$$e^{-\lambda \cdot 3 \cdot D}, \tag{6}$$

where the time of a quantum *swap* operation is 3 controlled-NOT gates[18]. Assuming that 7^k valid EPR pairs are needed for transmission we need to send:

$$7^k = B \times e^{-\lambda \cdot 3 \cdot D} \tag{7}$$

which given our error rate assumption $p \approx 10^{-8}$ implies that:

$$B \approx 7^3 / e^{-10^{-8} \cdot 3 \cdot D}. \tag{8}$$

Calculating this for some reasonable wire lengths we find that a length of 500 nm (1000 qubits) requires a bandwidth of 344 qubits and for 1 mm, 364 qubits are needed. Interpreting this result the bandwidth of the wire is proportional to the error coding scheme $\approx 7^k$ and the extra overhead for purification is minimal.

Of course supplying this bandwidth is a challenge and requires either bit-serial transmission (effectivelly adding 343 qubits to the logical end of the wire, and decreasing the aggregate bandwidth by a factor of 343) or parallel transmission. We estimate 343 parallel wires would require roughly $343 \times 300nm = 102um$, or put into persecptive it would be as wide as a 5145 quantum bits.

Placing these bandwidth calculations in a classical light, the speed at which a two qubit operation occurs is approximetly 100us. By using the wiring technique we propose in this paper we can largely discount distance to find that a bit-serial wire of qubits has a sustained transmission rate of $1/(3 \cdot 100\ \mu\text{s} \cdot 343) = 9.7$ lqbps (logical quantum bits per second). A parallel wire of $102um$ width has a sustained rate of transmission of approximetly $1/(3 \cdot 100\ \mu\text{s}) = 3333$ lqbps. The latency of the bit-serial quantum wire we propose is approximetly $(343 \cdot 3 + 4) \times 100us = 0.1s$ while the latency of the parallel wiring technique is $4 \cdot 100us = 0.0004s$.

It is significant to point out that a direct approach to wiring that does not follow the techniques described in this paper has dramatically different characteristics. While the bandwidth of the wire would be about the same, the latency would be large and distance dependent. For example, a parallel wire of only $1mm$ in length would require over 150 seconds to work (not counting error correction which could multiply this by up to a factor of 5)! Our pre-communication teleportation technique has clear latency advantages. Latency translates to increased decoherence of quantum data, which translates to increased overhead for error corrrection. Furthermore, while EPR pairs are easy to pipeline and purify, directly pipelining quantum data requires significantly more error correction to preserve that data.

7 Other Technologies

A myriad of technologies are under development for quantum computation. We have chosen a basic set of devices with which to explore a scalable, solid state architecture. More speculative technologies, however, have some interesting implications that we shall discuss here.

An interesting alternative to nuclear spin is using electron spin to represent qubits. The proposals are similar to the basic Kane proposal, but inter-qubit spacing can be much larger (100 nm) [39]. This larger spacing makes fitting control devices near qubits easier. Also, it is possible to avoid the J gate and only use variations on the A gate to interact qubits. Unfortunately, the decoherence time of electron spin is significantly smaller than nuclear spin, requiring control frequencies in the 10 GHz range, for typical magnetic fields used in electron spin resonance. These frequencies will be considerably harder to manage, even at larger pitch spacing.

Several optical devices provide some interesting alternatives to transporting quantum state using the Kane proposal. In particular, it is possible to produce a pair of photons which represent an EPR pair [40]. These pairs might then be distributed over a chip using optically-routed, polysilicon light pipes. Once they arrive at their destination, a photo-detector must be used to convert photon polarization to electron spin [41]. Note that spin representations are still necessary in order to perform any computations.

Yet another technology is the direct transport of electron spin over distance [42]. This technology, which involves ballistic transport of electrons, could avoid laborious qubit swaps or conversion from photons to electron spin.

8 Research Challenges

Our study of practical constraints has led us to identify several key areas of technology that could dramatically simplify the challenges building scalable quantum architectures, specifically focused on realizing the quantum macro-blocks identified in Section 5.

- **Efficient, fast, single-electron spin state to single atomic spin state conversion:** This would greatly reduce the size and increase the performance of the entropy exchange unit, by providing a superior method for preparing $|0\rangle$ states, which are essential to fault-tolerant quantum computation. It could also serve as a component in the photon-based EPR distribution network.
- **Wider pitch quantum technologies:** Device technologies with quantum interactions greater than 100 nm apart and sufficiently reliable decoherence rates will greatly ease the classical-quantum interface. The narrow 20 nm spacing of the Kane proposal balloons the classical support circuitry.
- **Measurement without zeros:** The currently proposed measurement technique requires a "cold" zero qubit to function correctly. This essentially doubles the output requirements of the entropy exchange unit. Since entropy exchange will consume a large fraction of a quantum computer it is vital to make efficient use of zero states.
- **Optimized fault-tolerant Schulman-Vazirani scheme:** The current purification and entropy compression processes are susceptible to errors. While fault-tolerant constructions could be used, they would consume zeros and come at the cost of significant overhead, as described above. Such a general construction is not required for specific, special-purpose macro-blocks such as these two, which we expect can be optimized dramatically to be fault-tolerant with little overhead.
- **Decoherence estimates:** small classical electrodes coming up to the qubits will have imperfect classical behavior and may cause additional decoherence. How much is not known, and determining this factor through experimental measurements and theoretical modeling will improve the accuracy of the calculation of the error-correction overhead.

The basic quantum architectural components described in Section 5 are largely universal among anticipated silicon-based quantum device technologies. While the precise size, shape and efficiency of these building blocks will evolve as new discoveries are made, the underlying functionality will not. Using these basic constructions architects can begin to think about classic architectural questions, some of which we address in this paper such as latency and bandwidth. Our future work will be in the construction and evaluation of quantum computing organizations. The basic blocks described in Section 5 are the roadmap for that process.

9 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 exploited quantum teleportation to enable pipelining and flexible error correction. Most importantly, we have 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.

References

[1] 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.*, vol. December 15, 2000.

[2] 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.

[3] 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*, vol. 404, pp. 256–258, 2000.

[4] S. Lloyd, "Quantum-mechanical computers," *Scientific American*, vol. 273, p. 44, Oct. 1995.

[5] M. Nielsen and I. Chuang, *Quantum computation and quantum information*. Cambridge, England: Cambridge University Press, 2000.

[6] D. P. DiVincenzo, "Quantum computation," *Science*, vol. 270, no. 5234, p. 255, 1995. arXive e-print quant-ph/9503016.

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

[8] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM J. Comp.*, vol. 26, no. 5, pp. 1484–1509, 1997.

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

[10] B. Kane, "A silicon-based nuclear spin quantum computer," *Nature*, vol. 393, pp. 133–137, 1998.

[11] J. M. Goodkind, "Proposed fabrication of a quantum computer using electrons on helium," in *Second Annual SQuInT Workshop*, 2000. Poster Abstract.

[12] M. T. Niemier and P. M. Kogge, "Exploring and exploiting wire-level pipelining in emerging technologies," in *International Symposium on Computer Architecture*, 2001.

[13] A. Steane, "Error correcting codes in quantum theory," *Phys. Rev. Lett.*, vol. 77, 1996.

[14] L. Schulman and U. Vazirani, "Molecular scale heat engines and scalable quantum computation," in *31st STOC*, 1999.

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

[16] 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.*, vol. 70, pp. 1895–1899, 1993.

[17] K. K. Likhareve, "Single-eletron devices and their applications," *Proceedings of the IEEE*, vol. 87, 1999.

[18] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, UK: Cambridge University Press, 2000.

[19] C. H. Bennett and P. W. Shor, "Quantum information theory," *itit*, vol. 44, no. 6, pp. 2724–42, 1998.

[20] L. Grover in *Proc. 28th Annual ACM Symposium on the Theory of Computation*, (New York), pp. 212–219, ACM Press, 1996.

[21] P. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35th Annual Symposium on Foundations of Computer Science*, (Los Alamitos, CA), p. 124, IEEE Press, 1994.

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

[23] I. L. Chuang, R. Laflamme, P. Shor, and W. H. Zurek, "Quantum computers, factoring, and decoherence," *Science*, vol. 270, p. 1633, Dec 1995. arXive e-print quant-ph/9503007.

[24] P. Shor, "Fault-tolerant quantum computation," in *37th FOCS*, 1994.

[25] 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.

[26] 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*, vol. 51, 1998.

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

[28] 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.*, vol. B-9, 1991.

[29] 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.

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

[31] 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*, vol. 79-15, 2001.

[32] L. J. Schulman and U. Vazirani, "Scalable NMR quantum computation," *arXive e-print quant-ph/9804060*, 1998.

[33] 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)*, pp. 322–329, 1999.

[34] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher, "Concentrating partial entanglement by local operations," *Phys. Rev. A*, vol. 53, no. 4, pp. 2046–2052, 1996. arXive e-print quant-ph/9511030.

[35] 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.*, vol. 76, p. 722, 1996. arXive e-print quant-ph/9511027.

[36] E. Knill, R. Laflamme, and W. H. Zurek, "Resilient quantum computation," *Science*, vol. 279, no. 5349, pp. 342–345, 1998. arXive e-print quant-ph/9702058.

[37] J. Preskill, "Fault-tolerant quantum computation," in *Quantum information and computation* (H.-K. Lo, T. Spiller, and S. Popescu, eds.), Singapore: World Scientific, 1998.

[38] 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*, pp. 176–188, 1997.

[39] R. Vrijen, E. Yablonovitch, K. Wang, H. W. Jiang, A. Balandin, V. Roychowdhury, T. Mor, and D. DiVincenzo, "Electron spin resonance transistors for quantum computing in silicon-germanium heterostructures," *arXive e-print quant-ph/9905096*, 1999.

[40] P. Kwiat *et al.*, "New high-intensity source of polarization-entangled photon pairs," *Phys. Rev. Lett.*, vol. 75, no. 24, 1995.

[41] R. Vrijen and E. Yablonovitch, "A solid-state spin coherent photo-detector for quantum communication," *arXive e-print quant-ph/0004078*, 2000.

[42] J. Kikkawa *et al.*, "Spin coherence in semiconductors: storage, transport and reduced dimensionality," *Physica E*, vol. 9, 2001.