Scalable Addressing Circuits for a Surface Code Silicon-based Quantum Computer

Rubaya Absar, *Member, IEEE,* Zach D. Merino, Hazem Elgabra, *Member, IEEE,* Xuesong Chen, *Member, IEEE,* Jonathan Baugh and Lan Wei, *Member, IEEE*

***Abstract*—Practical quantum computers require a coordinated operation on large numbers of quantum bits (qubits), posing significant challenges including large-scale integration, precise control of individual qubits, and large error-correction overhead. Among various quantum technologies, the combination of silicon (Si) quantum dot (QD) spin qubits and CMOS control electronics offers scalable solutions due to their large-scale integration potential using established semiconductor processes. In this paper, we propose a control circuit scheme for QD spin qubits operating on a node network architecture, which greatly eases the area constraints for control signal routing compared to traditional two-dimensional array architectures. A scalable circuit is designed to provide control signals for local and global operation of surface quantum error correction codes through a modular design of tiered switches controlled by demultiplexers. Critical power, performance, and area (PPA) merits are reported through a TSMC 65 nm technology implementation. The internode distance is found to be 44 µm for the 65 nm technology node, which scales down considerably using more advanced technology nodes (e.g., 8 µm internode distance for 7 nm technology). The total power consumption at 4 K for 10^4 number of nodes is calculated as 220 µW with negligible delay. We have also established a logarithmic relationship between the number of control lines and the number of nodes. The proposed design is an important step toward the implementation of scalable solid-state quantum processors with integrated cryo-electronics.**

***Index Terms*—electron spin qubits, quantum dots, network architecture, CMOS control circuits, cryogenic CMOS**

I. Introduction

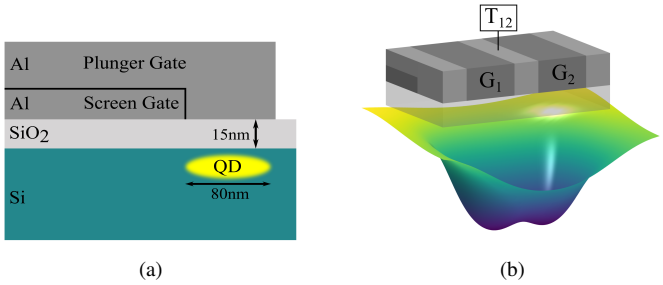
QUANTUM information processors have the potential to achieve significant computational speedups for certain problems, such as prime number factorization [1], quantum chemistry simulations [2], database search [3], financial modelling and forecasting [4], linear algebra problems [5], and more. Physicists, materials scientists, and engineers are combining efforts to solve the formidable challenges inherent in developing large-scale quantum processors. Quantum computers operate on qubits, i.e. two-level quantum systems that exhibit non-classical properties. A qubit can exist in a superposition of eigenstates and can be entangled with other qubits. These non-classical properties can be exploited using *quantum* algorithms, leading to exponential speedups for certain classically hard problems. Many physical systems have been explored for realizing qubits, including superconducting circuits [6], ion trap systems [7], electron spins in quantum dots [8], nuclear magnetic resonance (NMR) [9], photonic systems [10] and others. In this paper, we focus on developing a control addressing circuit for a scalable architecture based on electron spin qubits in silicon [11].

A major challenge for implementing a large-scale quantum processor is the sensitivity of quantum systems to noise and perturbations. The information stored in a quantum system is readily lost to decoherence, i.e. by uncontrolled interactions between the system and its environment. Thermal radiation, charge noise, control errors, and qubit cross-talk are a few important sources of decoherence in solid-state devices. Hence, quantum error correction (QEC) is a primary requirement for a fault-tolerant quantum processor. QEC relies on encoding the information into a larger Hilbert space, requiring additional resources in terms of qubits and logic operations. Surface codes are a family of QEC codes that provide a relatively high tolerance to errors and map to a two-dimensional (2D) device layout [12], [13]. Such codes underlie commercial efforts to build scalable processors based on superconducting qubits. However, the error correction overhead for achieving a truly fault-tolerant device is very large, requiring $\sim 10^4$ physical qubits for each logical qubit. Thus, millions of physical qubits will be necessary to build fault-tolerant systems with enough logical qubits ($\sim$ hundreds) to solve problems that are intractable on classical processors. Integrating, connecting, and controlling such a large number of qubits is extremely challenging, regardless of the chosen platform.

Electron spins in silicon quantum dots have several key advantages as a scalable quantum computing platform [14]–[21]. Due to their mesoscopic footprint (50-100 nm), quantum dots can yield a very high qubit density. Secondly, the use of isotopically enriched silicon (^{28}Si) enables qubit coherence times up to $\sim$ 1 second, with control fidelities as high as 99.5% [17]–[19]. A long coherence time reflects a low probability for qubit errors, in turn requiring less error correction overhead. Another significant advantage is that spin qubits in silicon can be operated at high temperatures, up to $\sim$2K, relative to superconducting qubits which require < 100 mK temperatures [22], [23]. This paves the way for the direct integration of classical control and readout electronics with the quantum processor. Finally, the widespread use of silicon technology in modern integrated circuits promises high precision, large-scale fabrication, with low manufacturing costs.

Quantum processors require high-fidelity qubits working with a control circuit interface that maintains smooth communication between the qubits and high-precision controlling electronics. In current laboratory experiments, the quantum processor is connected to room temperature (RT) electronics, requiring many wires to feed into the cryostat, creating a thermal load that severely limits the number of control lines [24]. With one or more control lines per qubit, this approach

is clearly not scalable. Developing a scalable control solution is one of the critical hurdles to advancing quantum computing. Operating the primary control electronics at cryogenic temperatures, i.e. cryo-electronics, is key to such a solution [25]–[28]. One can envision a Field Programmable Gate Array (FPGA)-based source of control signals located at a high cooling power cryostat stage (∼4 K), connected by a small number of lines to low-power multiplexing circuits located near the quantum processor at $\lesssim 1$ K temperatures. In this paper, we develop a cryogenic CMOS circuit for scalable addressing of spin qubits in a nodal network surface code architecture [11]. The circuit is modular and consists of tiered switches, shift registers, and demultiplexers. We analyze power consumption, performance, and scaling metrics, and find that the proposed circuits are feasible for integration with a quantum chip and operation at sub-Kelvin temperatures.



(a) (b)

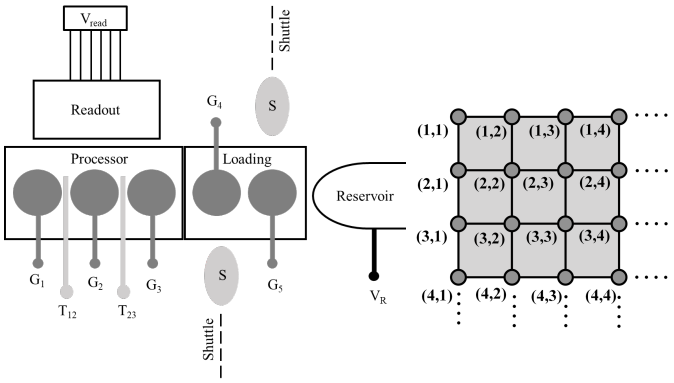
Fig. 1. (a) A schematic side view of an accumulation-mode quantum dot formed at the Si/SiO_2 interface by applying a DC voltage to the metal plunger gate, while keeping the screen gate grounded. A thin layer of insulating Al_2O_3 (black line) separates the plunger and screen gates. Dimensions shown are approximate to an order of magnitude. (b) A schematic representation of a double quantum dot, with surface gate electrodes and a corresponding double well potential energy profile. Here, the plunger gates G_1 and G_2 form the dots, while interdot tunneling is controlled by central gate T_{12}. The gate layout can be extended to form a linear dot array.

II. System Architecture

A scalable architecture for a quantum processor based on spin qubits in silicon was proposed by Buonacorsi *et al.* [11]. It consists of a 2D network of few-qubit nodes, with internode separation on the order of microns, and neighboring nodes linked by electron shuttling lines. A pair of electrons can be locally entangled, and then transported to separate nodes to share internode entanglement. This non-local entanglement is a necessary resource for implementing surface code error correction in such a network [29]. A node consists of a small linear array of quantum dots, with each dot confining a single electron. Figure 1a shows a schematic side view of a lateral silicon MOS quantum dot. By applying a positive DC voltage to the plunger gate electrode with the screen gate grounded, an accumulation-mode quantum dot forms at the Si/SiO_2 interface. Electrons populate the quantum dot via tunnel coupling to a nearby reservoir, and the charge state can be tuned by Coulomb blockade physics to single-electron occupancy. The quantum dot is pancake-shaped, with a lateral dimension $\sim 50-100$ nm and a vertical height of a few nm. Figure 1b is a schematic representation of a double quantum dot, the building block for a linear dot array. A double-well potential is formed using plunger gates G_1 and G_2, with interdot tunneling controlled by the central gate electrode T_{12}.

Figure 2a illustrates a possible layout within a single node. The processor section includes a data qubit controlled by G_1, and two ancilla qubits controlled by G_2 and G_3. Interdot tunneling, necessary for two-qubit logic operations, is controlled by gates T_{12} and T_{23}. A loading section includes two additional quantum dots for loading electrons from the reservoir and shuttling, controlled by G_4 and G_5. Focusing only on the processor and loading functions, and ignoring readout and shuttling, the node requires $k = 7$ control lines. Nodes are arranged in a square lattice array to form a network, as shown in Figure 2b, where nodes are labelled by row and column indices. Neighboring nodes are separated by several microns and are linked by linear, nominally empty dot arrays that enable internode electron shuttling [30]. The space between nodes significantly relaxes constraints on wiring densities, i.e. interconnects and vias, for connecting control lines to each node. In contrast, proposals for close-packed qubit arrays [15], [31]–[33] require wiring and interconnect densities difficult to achieve with present-day technology.

The surface code is based on defining a set of data qubits and measurement qubits, linked in a 2D array, and implementing repeated stabilizer measurements that do not perturb the data qubits when no error has occurred, but reveal an error when a measured eigenvalue changes [13]. When mapped to a nodal network [29], the X and Z stabilizer operations must be carried out sequentially, and each of those is split into two steps, as shown in Figure 3. In each step, plaquettes of four neighboring nodes (indicated by dark squares in Figure 3) are entangled and subsequently stabilized. Performing these operations globally on the whole network selects and stabilizes a quiescent state, equivalent to maintaining a single logical qubit in an arbitrary quantum state [13]. For such



(a) (b)

Fig. 2. (a) A possible layout within a node. The processor consists of three quantum dots controlled by plunger gates labelled G_1, G_2, G_3 and tunneling gates labelled T_{12}, T_{23}. The loading section contains two dots controlled by gates G_4, G_5 to load electrons into the processor from the reservoir or shuttle electrons to/from neighboring nodes. Internode shuttling dots are labelled S. A readout circuit is capacitively coupled to the processor dots. In the present work, we focus on control of the processor and loading sections, ignoring shuttling and readout. (b) A square array of nodes forms a network. Nodes are numbered according to row and column indices. Physically, the internode separation is on the scale of several microns.

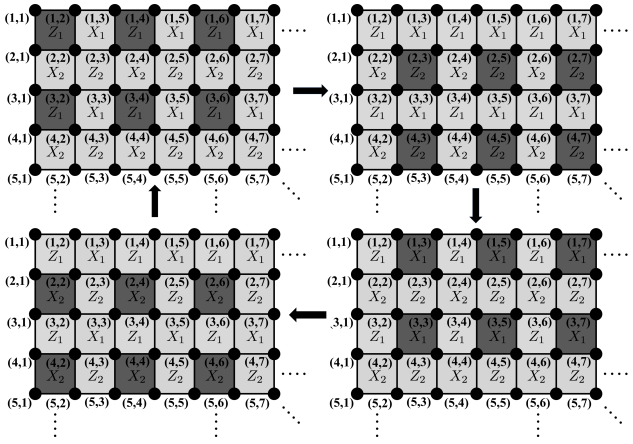


Fig. 3. A full stabilizer cycle consists of four steps. Starting at the upper left, the Z stabilizer is split into two steps, followed by the two-step X stabilizer.

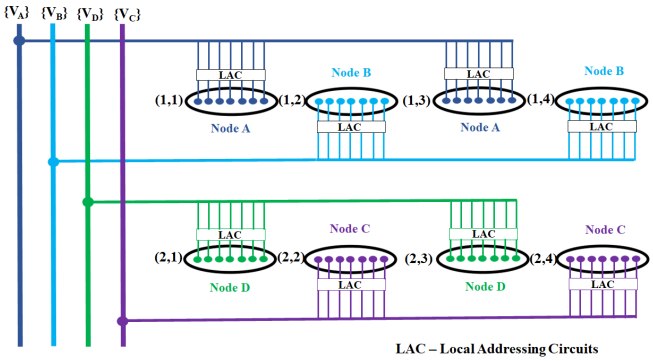


Fig. 4. Four sets of inputs are routed for global control. Each coloured thick line (labelled $\{V_A\}$,$\{V_B\}$,$\{V_D\}$,$\{V_C\}$) represents a set of k input signals associated with each of the four nodes (A, B, D, C) in a unit cell. Local addressing circuits are bypassed for nodes participating in global operations.

global operations, all nodes (n, m) and $(n + 2a, m + 2b)$ are equivalent, for integers a and b, so there are four inequivalent lattice sites. However, on top of the *global* addressing, *local* addressing of nodes is required in order to correct errors, define multiple logic qubits, and implement logical operations. For example, creating local inactive nodes, or 'holes', is needed to define logical qubits and gate operations [13]. When errors are detected, corrective pulses must also be applied locally, leaving all other nodes untouched. The remainder of the paper describes a scalable circuit for implementing both global and local addressing of nodes.

III. RTL Design for Control Signal Routing

The surface code sequence can be decomposed into single-qubit and two-qubit unitary gate operations. Single-qubit unitaries (rotations) are performed by electron spin resonance (ESR) using a global microwave field [16], [21]. Qubit addressability is achieved by electric control of the electronic g-factor via the Stark effect. These manipulations are performed by sending fast voltage pulses to the plunger gates [34], e.g. G_1, G_2, G_3 in Figure 2a. Two-qubit operations are implemented by the exchange interaction [35], again realized by appropriate voltage signals sent to the plunger and tunneling gates (G_1, G_2, G_3 and T_{12}, T_{23} respectively in Figure 2a). The voltages required for single- and two-qubit operations typically have a constant offset in the range of 0.5-3 V, combined with short pulses of ∼1-100 mV in amplitude and duration on the scale of nanoseconds to microseconds.

As described in Section II, the network architecture has spatial symmetry, with operations applied to identical 4-node unit cells, and repetition of the basic stabilizer cycle in time. As a result, a small number of node-specific gate voltage signals can be generated and multiplexed to all identical nodes for global operations. Local node addressability is needed for certain operations, such as correcting errors and performing operations on encoded qubits. In this section, we present the register-transfer-level (RTL) design to route gate signals for both global and local addressing. To minimize wiring in the cryostat (and thus heat load, pick-up noise, etc.), the routing control circuits are intended to be placed at sub-Kelvin temperatures in close proximity to the quantum device. It is well known that CMOS transistors and circuits can function at cryogenic temperatures, typically with an increase in drain current by approximately 20-30 % [36], [37] and improvement in sub-threshold slope (SS) from $>$ 60 mV/ dec at RT to as low as 10 mV/ dec [36], [38].

A. Global Control

For global operations, all nodes (n, m) and $(n+2a, m+2b)$ are equivalent for integers a and b, resulting in a 4-node unit cell. Consequently, the network requires 4 groups of k gate signal sequences to be shared spatially among unit cells ($k = 7$ in the example of Figure 2a). Figure 4 shows the routing of global control signals in a 2x4 array of nodes. We label the four nodes in a unit cell by A, B, C, and D, with A as the upper left node and proceeding clockwise in alphabetical order. Thick lines indicate a bundle of k input signals intended for the k gates in each node. Circuits that enable local addressing (labelled LAC in Figure 4) are bypassed for global operations. Each step of the stabilizer cycle indicated in Figure 3 would correspond to a distinct set of voltage sequences on the four nodes A-D. Note that global control assumes identical behavior of quantum dots across all nodes, which is not expected to be the case in practice. We assume that local calibrations can be performed *a priori*, e.g., using floating gates to fine-tune the chemical potentials in each dot so that their behaviour under global voltage pulses is approximately uniform.

B. Local Addressing

The ability to address any individual node in the network is necessary for creating holes while sending correction pulses requires individual access to specific gates. As illustrated in Figure 5a, the input signal to each gate passes through two switches, one to enable hole creation (SW_1) and one for error correction (SW_2). SW_1 and SW_2 are turned on or off through their own address demultiplexer (deMUX) tree for individual gate access. The control signals for the deMUX trees are generated by the cryogenic FPGA and then supplied through shift registers.

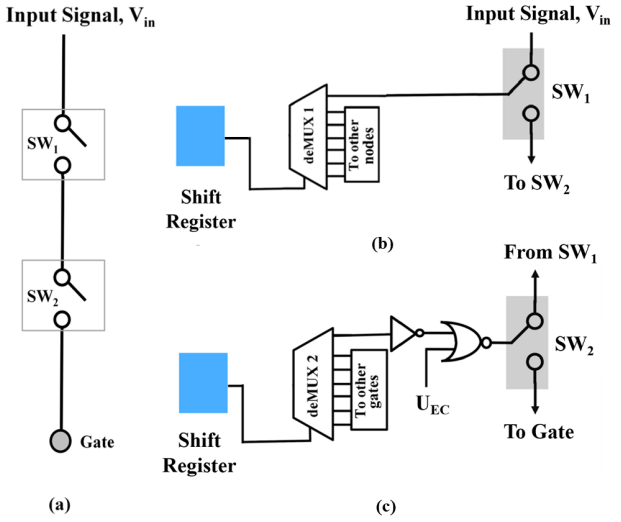


Fig. 5. (a) Each gate inside a node is connected to two switches in series (SW_1, SW_2) for local addressing. The control signal of SW_1 is shared between k gates (k = 7 as shown in Fig. 2) inside a particular node. (b) SW_1 controls the local operation of hole creation and is connected to deMUX 1, which will turn on or turn off the switch. (c) SW_2 controls the data correction process and is connected to deMUX 2 through the logic gate nor and inverter, which will turn on or turn off the switch depending on the control signal from the deMUX 2 and the universal signal, U_{EC}. The control signals of the deMUX 1, and deMUX 2 are supplied from an FPGA through shift registers.

A hole is defined as a node that does not receive any pulsed signals during some computational step. This is implemented by opening SW_1 to disconnect the gates of the chosen node from the global signal path. Figure 5b shows the addressing circuitry connected to SW_1. In this example LAC, one of the outputs of a demultiplexing tree, deMUX 1, is used for addressing the chosen node.

At the end of each surface code cycle, measurements of ancilla qubits are used to identify data qubits that have experienced errors, along with the type of error. Thus, correction pulses must be addressed to individual nodes. All nodes in the network must be disconnected from gate voltage signals, except for the particular ones requiring correction. Inside that particular node, each gate might require a different correction pulse depending on the type of error. In this case, all switches SW_1 are connected and all switches SW_2 will be initialized as open by setting a universal control signal, U_{EC}, to 0 (Figure 5c). Through deMUX 2, only SW_2 switches for those gates inside the nodes that require correction will be connected. SW_2 for the rest of the nodes remains open, as summarized in Table I. According to the error type, a pre-defined gate voltage sequence supplied by the FPGA will be sent to the erroneous nodes. Different error types are corrected in series.

TABLE I
MODES OF OPERATION FOR THE ERROR CORRECTION CIRCUITRY

deMUX 2 output	U_{EC} signal	Nor output	SW_2 state	Mode of Operation
0	0	0	Closed	Correction pulse to specific node
1	0	1	Open	All node disconnected
1	1	0	Closed	Normal operation

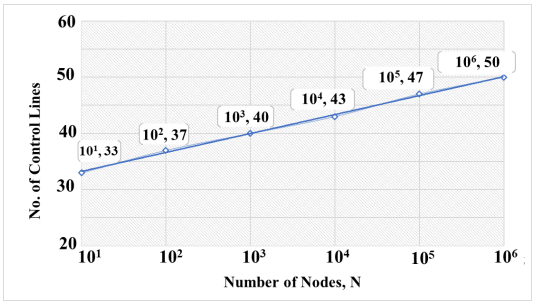


Fig. 6. Logarithmic dependence of the no. of control lines on no. of nodes (N), for $k = 7$, $M = 8$, and $C = 4$ (see text for parameter definitions).

Present-day, small-scale implementations of qubit arrays require at least 2 coaxial cables per qubit [39], i.e. a linear dependence on the number of qubits (or nodes), which is not scalable. Here, a scalable *logarithmic* dependence is achieved for the number of control lines connecting the FPGA (few-Kelvin stage) to the quantum chip (sub-Kelvin stage) versus the number of nodes, N, for our multiplexed signal routing scheme. Defining C as the number of control lines to each M bit serial-in parallel-out shift register, we obtain the total number of local and global control lines as $CX + 4k$, where $X = \frac{\log(N^2 k)}{M \log(2)}$ is the number of shift registers. For example, when $N = 10^4, k = 7, M = 8$, and $C = 4$, the total number of control lines is ≈ 43. Figure 6 shows the logarithmic scaling of the number of control lines with the number of nodes.

IV. CMOS-GATE LEVEL CIRCUIT DESIGN AND PERFORMANCE ANALYSIS

In this section, several gate-level circuit designs for implementing the switches and demultiplexers are described. Results for power-performance-area (PPA) are obtained and analyzed.

A. Gate-level design for signal routing

The single-qubit or two-qubit operations require fast DC voltage pulses combined with a constant (or slow) DC offset applied to the plunger and tunneling gates. The range of the DC offsets is typically between 0 - 3 V and is assumed to remain constant throughout global and local control operations. As a result, the fast pulses and offset voltages can be routed separately and combined through a DC-level restorer. Figure 7a shows the routing path of these signals, where the pulses are passed through the switches SW_1 and SW_2, then combined with the offset voltage through a DC level restorer. SW_1 and SW_2 are designed using pass gate transistors with an NMOS feeder as shown in Figure 7b. Figure 7c shows a 1:4 deMUX circuit. The DC restorer circuit is shown in Figure 7d. These circuits were implemented and taped out using the TSMC65 nm GP process.

B. Power-Performance-Area (PPA) Results and Projections

Quantum processor requires sub-Kelvin temperatures, so it is important to ensure that the heat power dissipated during the process is within the cooling power of the cryostat [23].

The total static and dynamic power dissipated in a unit cell at RT are extracted from simulations using Cadence Virtuoso.

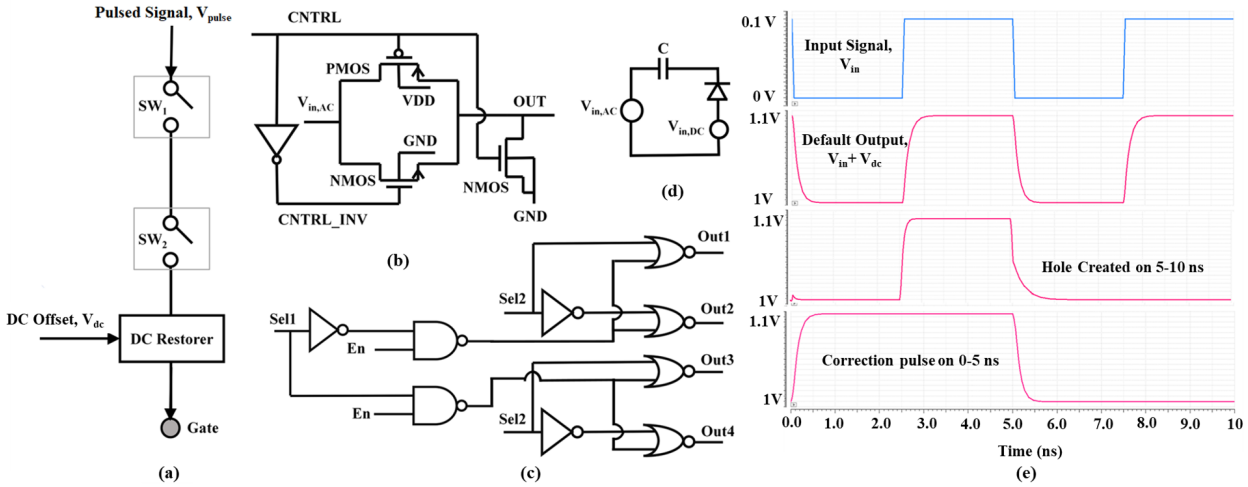


Fig. 7. (a) The layout of a gate inside nodes consisting of two series switches (SW_1, SW_2) to pass/block the fast pulse signal (V_{pulse}) and a DC level restorer to combine V_{pulse} and constant offset voltage (V_{dc}). (b) The gate level implementation of SW_1 and SW_2 using a pass gate transistor and an NMOS feeder. (c) A 1:4 deMuX tree is designed based on NAND and NOR gates. (d) A DC restorer circuit. (e) Three distinct cases of outputs including default output (V_{pulse} + V_{dc}), hole creation, and sending correction pulse are shown with V_{pulse} = 100 mV, f_{vpulse}= 1 GHz and V_{dc} = 1 V as the inputs. The delay between V_{pulse} and the output at the QD gates is 35 ps.

The room temperature value is projected to cryogenic temperatures by experimentally measuring the factor by which static and dynamic power are altered at low temperature. Measurements from NMOS and PMOS transistors fabricated using TSMC 65 nm technology indicate that the static power dissipation is reduced by a factor of 4 at cryogenic temperatures due to the leakage current reduction, while the dynamic power remains the same.

Figure 8 shows the total power dissipated by the switching circuits and deMUX trees (power dissipated by DC level restorer is negligible) during normal operation and hole creation/error correction at cryogenic temperature for an input periodic pulse of 10 mV amplitude and 1 GHz frequency with an activity factor of 0.1 and 0.001 for normal operation and hole creation/error correction respectively. During normal operation, the deMUX tree is not activated and it dissipates static power while the switches dissipate dynamic power. On the other hand, in the case of hole creation and error correction, the deMUX trees are activated and dissipate dynamic power while the switches dissipate static power.

The total power dissipated in a network is dominated by the contribution from the switches and is roughly 220 μW for $N = 10^4$. In practice, the power consumption will be much lower since the technology used to determine the power consumption is an older one, and the power consumption per gate is reduced by 40% between technology nodes while moving towards the newer technologies [40]. Interconnect capacitance from various metal layers of signal routing wires contributes significantly to the dynamic power. Following good layout practices is necessary to minimize these capacitances.

The total area occupied by a unit cell is as large as 1986 μm^2, where the individual sub-circuits deMUXes, capacitors, and switches occupy 1045 μm^2, 28 μm^2 and 896 μm^{2}respectively. This results in a node-to-node distance of 44 μm using TSMC 65 nm technology. Moving to a more

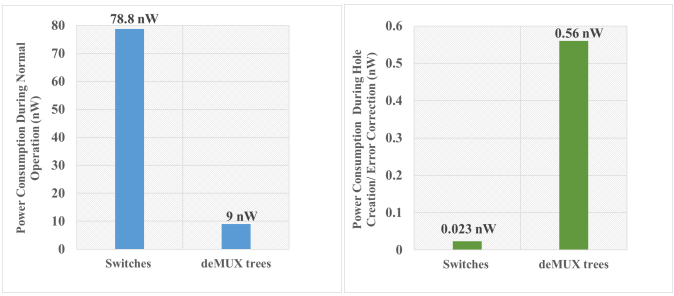


Fig. 8. The total power dissipated from switches and deMUXes in a unit cell at $T = 4$ K for a 1 GHz, 10 mV input during (a) Normal operation (activity factor = 0.1) and (b) Hole creation/ error correction (activity factor = 0.001).

advanced technology node, such as 7 nm, the area reduces by a factor of 25 [41]. In that case, the internode distance becomes 8 μm, which is feasible to implement using the network architecture and electron shuttling [11].

Lastly, we will analyze the worst-case propagation delay. The delay between the input to the tiered switches and the output at the QD gates is 35 ps. The control signals for the switches are supplied using deMUX trees. A 1:4 deMUX requires 10 ps to propagate the control signal to the switch after being addressed. Given that the shortest qubit control pulses are expected to be $\sim$ 1 ns in duration, the propagation delay is not a fundamental obstacle and can be accounted for in the relative timing of signals generated by the FPGA. Figure 7e shows the input and output pulses for 3 different modes of operation including the default mode where the output is the summation of input DC pulse signal (V_{pulse}) and a dc offset signal (V_{dc}), the hole creation mode which disconnects V_{pulse} signal and only passes V_{dc} and finally sending correction signal to the gate where error correction is required.

V. Conclusion

A challenging aspect of building a scalable quantum processor is to avoid the interconnect bottleneck. In this work, an approach based on tiered switches and an array of demultiplexers is introduced to scalably route control signals in the context of a QD spin qubit network architecture. Our method creates a low-power interface between the spin qubits at sub-Kelvin temperature and controls cryo-electronics at a higher temperature ($\sim$ 4 K) stage. We have determined the key characteristics in terms of scalability, area, and power consumption, for the proposed sub-circuits. We have established the inter-node distance to be 44 µm while the total power consumption at 4 K for $N = 10^4$ is 220 µW using TSMC 65 nm technology. The purpose of this work is to give insights into the feasibility of implementing a quantum processor with integrated cryo-electronics to achieve local and global control, an important step towards reaching a large-scale, semiconductor-based quantum computer.

References

[1] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*. IEEE Comput. Soc. Press, 1994, pp. 124–134.

[2] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, "Simulating chemistry using quantum computers," *Annual review of physical chemistry*, vol. 62, no. 1, pp. 185–207, 2011.

[3] L. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the twenty-eighth annual ACM symposium on theory of computing*, ser. STOC '96, vol. 129452. ACM, 1996, pp. 212–219.

[4] R. Orús, S. Mugel, and E. Lizaso, "Quantum computing for finance: Overview and prospects," *Reviews in Physics*, vol. 4, p. 100028, Nov. 2019. [Online]. Available: https://doi.org/10.1016/j.revip.2019.100028

[5] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Physical Review Letters*, vol. 103, no. 15, p. 150502, 2009. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.103.150502

[6] M. Kjaergaard *et al.*, "Superconducting qubits: Current state of play," *Annual Review of Condensed Matter Physics*, vol. 11, no. 1, pp. 369–395, 2020. [Online]. Available: https://doi.org/10.1146/annurev-conmatphys-031119-050605

[7] D. Kielpinski, C. Monroe, and D. J. Wineland, "Architecture for a large-scale ion-trap quantum computer," *Nature*, vol. 417, no. 6890, pp. 709–711, 2002. [Online]. Available: https://doi.org/10.1038/nature00784

[8] C. Kloeffel and D. Loss, "Prospects for spin-based quantum computing in quantum dots," *Annual review of condensed matter physics*, vol. 4, no. 1, pp. 51–81, 2013.

[9] D. Suter and T. S. Mahesh, "Spins as qubits: quantum information processing by nuclear magnetic resonance," *The Journal of chemical physics*, vol. 128, no. 5, pp. 052 206–052 206, 2008.

[10] P. Kok, W. J. Munro, K. Nemoto, T. C. Ralph, J. P. Dowling, and G. J. Milburn, "Linear optical quantum computing with photonic qubits," *Reviews of modern physics*, vol. 79, no. 1, pp. 135–174, 2007.

[11] B. Buonacorsi *et al.*, "Network architecture for a topological quantum computer in silicon," *Quantum Science and Technology*, vol. 4, no. 2, p. 25003, 2019.

[12] N. C. Jones *et al.*, "Layered architecture for quantum computing," *Physical review. X*, vol. 2, no. 3, p. 031007, 2012.

[13] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, "Surface codes: Towards practical large-scale quantum computation," *Physical review. A, Atomic, molecular, and optical physics*, vol. 86, no. 3, 2012.

[14] M. Friesen *et al.*, "Practical design and simulation of silicon-based quantum-dot qubits," *Physical Review B*, vol. 67, no. 12, Mar. 2003. [Online]. Available: https://doi.org/10.1103/physrevb.67.121301

[15] M. Veldhorst, H. G. J. Eenink, C. H. Yang, and A. S. Dzurak, "Silicon cmos architecture for a spin-based quantum computer," *Nature communications*, vol. 8, no. 1, pp. 1766–8, 2017.

[16] M. Veldhorst *et al.*, "An addressable quantum dot qubit with fault-tolerant control-fidelity," *Nature nanotechnology*, vol. 9, no. 12, pp. 981–985, 2014.

[17] P. Cerfontaine *et al.*, "Closed-loop control of a gaas-based singlet-triplet spin qubit with 99.5% gate fidelity and low leakage," *Nature communications*, vol. 11, no. 1, pp. 4144–4144, 2020.

[18] A. Noiri *et al.*, "Fast universal quantum gate above the fault-tolerance threshold in silicon," *Nature (London)*, vol. 601, no. 7893, pp. 338–342, 2022.

[19] X. Xue *et al.*, "Quantum logic with spin qubits crossing the surface code threshold," *Nature*, vol. 601, no. 7893, pp. 343–347, Jan. 2022. [Online]. Available: https://doi.org/10.1038/s41586-021-04273-w

[20] L. M. K. Vandersypen and M. A. Eriksson, "Quantum computing with semiconductor spins," *Physics today*, vol. 72, no. 8, pp. 38–45, 2019.

[21] R. Maurand *et al.*, "A cmos silicon spin qubit," *Nature communications*, vol. 7, no. 1, pp. 13 575–13 575, 2016.

[22] C. H. Yang *et al.*, "Operation of a silicon quantum processor unit cell above one kelvin," *Nature (London)*, vol. 580, no. 7803, pp. 350–354, 2020.

[23] L. M. K. Vandersypen *et al.*, "Interfacing spin qubits in quantum dots and donors—hot, dense, and coherent," *npj Quantum Information*, vol. 3, no. 1, p. 34, Sep 2017. [Online]. Available: https://doi.org/10.1038/s41534-017-0038-y

[24] F. Arute *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature (London)*, vol. 574, no. 7779, pp. 505–510, 2019.

[25] J. Park *et al.*, "A fully integrated cryo-cmos soc for state manipulation, readout, and high-speed gate pulsing of spin qubits," *IEEE journal of solid-state circuits*, vol. 56, no. 11, pp. 3289–3306, 2021.

[26] F. Jazaeri, A. Beckers, A. Tajalli, and J.-M. Sallese, "A review on quantum computing: Qubits, cryogenic electronics and cryogenic mosfet physics," 2019. [Online]. Available: https://arxiv.org/abs/1908.02656

[27] E. Charbon *et al.*, "Cryo-cmos for quantum computing," in *2016 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2016, pp. 13.5.1–13.5.4.

[28] B. P. Wuetz *et al.*, "Multiplexed quantum transport using commercial off-the-shelf CMOS at sub-kelvin temperatures," *npj Quantum Information*, vol. 6, no. 1, May 2020. [Online]. Available: https://doi.org/10.1038/s41534-020-0274-4

[29] N. H. Nickerson, Y. Li, and S. C. Benjamin, "Topological quantum computing with a very noisy network and local error rates approaching one percent," *Nature communications*, vol. 4, no. 1, pp. 1756–1756, 2013.

[30] B. Buonacorsi, B. Shaw, and J. Baugh, "Simulated coherent electron shuttling in silicon quantum dots," *Physical Review B*, vol. 102, no. 12, p. 125406, 2020.

[31] C. D. Hill *et al.*, "A surface code quantum computer in silicon," *Science Advances*, vol. 1, no. 9, pp. e1 500 707–e1 500 707, 2015.

[32] J. O'Gorman, N. H. Nickerson, P. Ross, J. J. L. Morton, and S. C. Benjamin, "A silicon-based surface code quantum computer," *npj Quantum Information*, vol. 2, no. 1, p. 15019, 2016.

[33] G. Pica, B. W. Lovett, R. N. Bhatt, T. Schenkel, and S. A. Lyon, "Surface code architecture for donors and dots in silicon with imprecise and nonuniform qubit couplings," *Phys. Rev. B*, vol. 93, p. 035306, Jan 2016. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.93.035306

[34] C. H. Yang *et al.*, "Silicon qubit fidelities approaching incoherent noise limits via pulse engineering," *Nature Electronics*, vol. 2, no. 4, pp. 151–158, Apr. 2019. [Online]. Available: https://doi.org/10.1038/s41928-019-0234-1

[35] M. Veldhorst *et al.*, "A two-qubit logic gate in silicon," *Nature*, vol. 526, no. 7573, pp. 410–414, 2015.

[36] A. Beckers, F. Jazaeri, and C. Enz, "Characterization and modeling of 28-nm bulk cmos technology down to 4.2 k," *IEEE Journal of the Electron Devices Society*, vol. 6, pp. 1007–1018, 2018.

[37] B. Xie *et al.*, "Effect of cryogenic temperature characteristics on 0.18-μm silicon-on-insulator devices," *Chinese Physics B*, vol. 25, no. 7, p. 078501, 2016.

[38] H. Elgabra, B. Buonacorsi, C. Chen, J. Watt, J. Baugh, and L. Wei, "Virtual source based iv model for cryogenic cmos devices," in *2019 International Symposium on VLSI Technology, Systems and Application (VLSI-TSA)*. IEEE, 2019, pp. 1–2.

[39] J. C. Bardin *et al.*, "29.1 a 28nm bulk-cmos 4-to-8ghz ¡2mw cryogenic pulse modulator for scalable quantum computing," in *2019 IEEE International Solid- State Circuits Conference - (ISSCC)*, 2019, pp. 456–458.

[40] E. Sicard and A. Boyer, "Impact of technological trends and electromagnetic compatibility of integrated circuits," in *EMC Compo 2019*.

[41] A. Stillmaker and B. Baas, "Scaling equations for the accurate prediction of cmos device performance from 180 nm to 7 nm," *Integration*, vol. 58, pp. 74–81, 2017.