Spiderweb Array: A Sparse Spin-Qubit Array

Jelmer M. Boter,[1,2] Juan P. Dehollain,[1,2,3] Jeroen P.G. van Dijk,[1,2,4] Yuanxing Xu,[1,2]
Toivo Hensgens,[1,2] Richard Versluis,[1,5] Henricus W.L. Naus,[1,5] James S. Clarke,[6] Menno
Veldhorst,[1,2] Fabio Sebastiano,[1,4] and Lieven M.K. Vandersypen[1,2,6,*]

[1] *QuTech, Delft University of Technology, Lorentzweg 1, Delft 2628 CJ, Netherlands*
[2] *Kavli Institute of Nanoscience, Delft University of Technology, Lorentzweg 1, Delft 2628 CJ, Netherlands*
[3] *School of Mathematical and Physical Sciences, University of Technology Sydney, Ultimo, NSW 2007, Australia*
[4] *Department of Quantum and Computer Engineering, Delft University of Technology, Delft 2628 CJ, Netherlands*
[5] *Netherlands Organization for Applied Scientific Research (TNO), P.O. Box 155, Delft 2600 AD, Netherlands*
[6] *Components Research, Intel Corporation, 2501 NE Century Blvd, Hillsboro, Oregon 97124, USA*

(Received 6 October 2021; revised 19 January 2022; accepted 5 May 2022; published 19 August 2022)

One of the main bottlenecks in the pursuit of a large-scale–chip-based quantum computer is the large number of control signals needed to operate qubit systems. As system sizes scale up, the number of terminals required to connect to off-chip control electronics quickly becomes unmanageable. Here, we discuss a quantum-dot spin-qubit architecture that integrates on-chip control electronics, allowing for a significant reduction in the number of signal connections at the chip boundary. By arranging the qubits in a two-dimensional array with about 12 μm pitch, we create space to implement locally integrated sample-and-hold circuits. This allows us to offset the inhomogeneities in the potential landscape across the array and to globally share the majority of the control signals for qubit operations. We make use of advanced circuit modeling software to go beyond conceptual drawings of the component layout, to assess the feasibility of the scheme through a concrete floor plan, including estimates of footprints for quantum and classical electronics, as well as routing of signal lines across the chip using different interconnect layers. We make use of local demultiplexing circuits to achieve an efficient signal-connection scaling, leading to a Rent's exponent as low as $p = 0.43$. Furthermore, we use available data from state-of-the-art spin qubit and microelectronics technology development, as well as circuit models and simulations, to estimate the operation frequencies and power consumption of a million-qubit array. This work presents a complementary approach to previously proposed architectures, focusing on a feasible scheme to integrating quantum and classical hardware, and identifying remaining challenges for achieving full fault-tolerant quantum computation. It thereby significantly closes the gap towards a fully CMOS-compatible quantum computer implementation.

DOI: 10.1103/PhysRevApplied.18.024053

I. INTRODUCTION

Semiconductor quantum dots [1], particularly in silicon [2], are attractive hosts for spin qubits in large-scale quantum computation applications, because of their assumed compatibility with conventional CMOS integration processes. In addition, the approximate 100 nm spatial dimensions intrinsic to semiconductor spin qubits provide the potential to pack many millions of qubits inside a single quantum processor chip. The last several years have seen significant progress in spin-qubit research that resulted in the demonstration of long coherence times [3], high-fidelity single- [3–5] and two-qubit gates [6,7], quantum algorithms [8], quantum nondemolition measurements [9,10], and electron spin [11] and charge [12,13] transfer.

Scaling to millions of qubits has many technological hurdles that will need to be cleared. Among them, the wiring bottleneck is a very challenging one for any solid-state qubit controlled by electrical signals [14–16]: currently at least one wire must run from the control electronics to each and every qubit, but chips have strict interconnect wiring density limitations, making a one-wire-per-qubit scheme unfeasible in the large scale. This was already an issue when classical processors started to scale up, leading to the development of techniques such

[*] l.m.k.vandersypen@tudelft.nl

as crossbar addressing, that enabled the number of output pins T to increase at a rate much slower than the number of transistor unit cells U on the chip. This was formally captured by Rent's rule: $T = c\ U^p$, where c is the number of connections per unit cell and p is Rent's exponent, a measure of optimization in the wiring scheme [15].

One approach to reducing Rent's exponent in quantum-dot-based spin qubits [17,18] involves addressing two-dimensional (2D) quantum-dot arrays of N^2 qubits using about N word and bit lines. These crossbar addressing schemes impose strong requirements on the homogeneity of the quantum-dot potentials across the array, which in itself is a great technological hurdle. Furthermore, they are inefficient in the sense that many operations can only be executed sequentially.

A second approach is to decrease qubit density by using long-range quantum links that connect modules of qubits [14,19]. Semiconductor spin qubits provide mechanisms to control the length scales required for qubit-qubit interactions over a wide range, allowing one to configure the desired density while still maintaining very large qubit arrays on chip. With this scheme, the distribution of the space on the chip can be customized for the integration of local, classical control electronics aimed at solving the homogeneity and operation efficiency issues.

Expanding on this potential route to solving the wiring bottleneck, we present and analyze the *spiderweb array*, a sparse 2D spin qubit array with single-qubit nodes separated by gate-based shuttling channels [20]. In this analogy, the qubit array resembles the open structure of a spiderweb, with vacant space between spiders (spin qubits) that move along the threads of their web (shuttling channels). We focus on how to use the open area of the sparse array to integrate classical control electronics with quantum hardware, in order to minimize the need for off-chip interconnects, resulting in a scalable Rent's exponent. Different from many proposal papers, we make a rather extensive initial effort to assess the feasibility of this classical-quantum integration, by considering several relevant practical aspects, without aiming or claiming to be exhaustive in this assessment. Another aim of this work is to identify immediate technological development opportunities that can be prioritized in order to operate a large-scale spiderweb array as a quantum processor.

We first describe in Sec. II the basic physical architecture of the qubit array and the implementations of the operations required to sustain the surface code. In Sec. III we describe the integration of on-chip classical electronics at different layers within the architecture. This leads to a logarithmic reduction in the number of control and measurement lines from the qubit region to the chip boundaries, which is discussed in Sec. IV. We then consider in Sec. V the footprint of the local control circuits and the array sparsity required to accommodate them. Section VI discusses the power dissipation in different sections of the array, assuming about 1 K operation [21–23], and the effects this has on the operation. Finally, we put all these ingredients together to describe a feasible implementation of a million-qubit array in Sec. VII.

II. ARRAY DESIGN AND OPERATION

The overarching concept of the spiderweb array is presented in Fig. 1. A large 2D square lattice of spin qubits constitutes the *quantum plane*. The spin qubits consist of single electrons confined in electrostatically defined quantum dots in silicon. The large 2D array of qubits can be used to implement the surface code [24], by assigning qubits as data qubits or surface code (SC) ancilla qubits in a checkerboard fashion, and allowing single-qubit operations as well as nearest-neighbor two-qubit operations. In contrast to most other spin-qubit architectures, this array is sparse, with a qubit pitch d of the order of 10 μm. This has two major implications.

1. It facilitates the local integration of classical control electronics, consisting here of sample-and-hold circuits that provide independent dc biasing of each quantum-dot gate electrode, which effectively offsets inhomogeneities in the potential landscape across the array. Consequently, the array can be considered fully homogeneous, allowing the majority of qubit control signals to be shared across the entire array, which significantly reduces the number of control lines at the quantum plane boundary.
2. It requires a means to transfer quantum information over distances of about 10 μm, which we implement by shuttling the qubits to operation regions—where single- and two-qubit operations are performed along with readout.

We define a *unit cell* as the smallest operational set of elements, which can be concatenated with identical unit cells to form the large 2D array. This unit cell [Fig. 1(a)] contains four qubits, and has an area of $(2d)^2$ and a perimeter of $8d$. We define *modules* [Fig. 1(b)] consisting of $N \times N$ unit cells with an area and a perimeter of $(2dN)^2$ and $8dN$, respectively. Modules define sections of the quantum plane where dc biasing and readout occur sequentially across the qubits in the module. All other operations that are part of the surface code cycle occur simultaneously in all unit cells in the entire array. Completing the array, the *quantum plane* [Fig. 1(c)] consists of $M \times M$ modules (M^2N^2 unit cells and $4M^2N^2$ qubits), covering an area and a perimeter of $(2dNM)^2$ and $8dNM$, respectively. As seen from Fig. 1(d), the quantum plane is designed to occupy a section of the die, with the remaining space to be used to reduce the off-chip wire count even further by adding additional on-chip electronics. Module sizes for dc biasing and readout (N_b and N_r, respectively) may be different, as

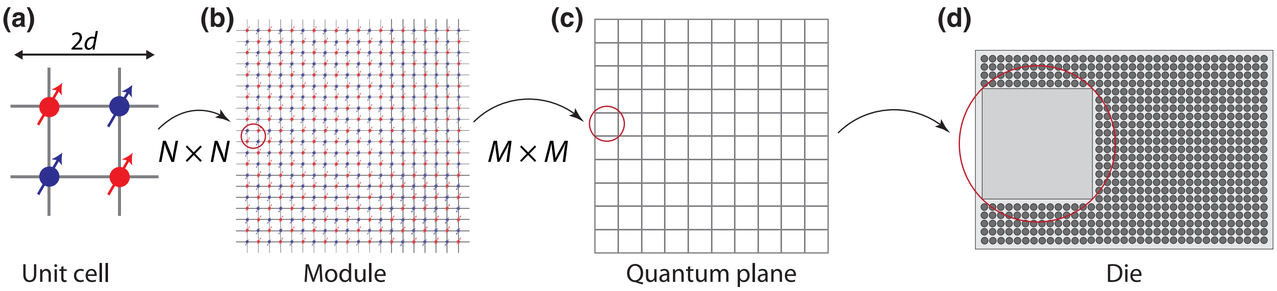

FIG. 1. Overview of the *spiderweb* architecture with a schematic breakdown of its components, as described in the main text. (a) A unit cell contains four qubits that are separated by the qubit pitch d. Qubits are color coded to distinguish data qubits (blue) and SC-ancilla qubits (red), as defined in the surface code. (b) A module consists of $N \times N$ unit cells, and in turn $M \times M$ modules together form (c) the quantum plane. Generally, the quantum plane occupies only part of (d) the die and the remaining area can be used to further reduce the off-chip wire count by adding additional on-chip electronics.

long as $N_bM_b = N_rM_r$, where M_b and M_r relate to the numbers of dc biasing modules and readout modules in the full array, respectively.

A detailed schematic of the components of a unit cell is presented in Fig. 2. The unit cell mainly consists of large linear arrays of electrostatic gates, connecting the vertices of the qubit array to the operation regions [see Fig. 2(a)]. The bulk of the gates in these linear arrays makes up the shuttling channels (blue gates in Fig. 2). Four phase-shifted sinusoidal signals are applied to four consecutive gates (shades of blue in Fig. 2), repeating the set of signals along the linear array to create a traveling-wave potential to trap and shuttle an electron [13,25]. The sign of the phase difference between adjacent gates defines the shuttling direction.

Four gates at the vertices of the spiderweb array [red gates in Fig. 2(b)] are used to control both the confinement potential that keeps the electrons at the vertex while idle, as well as the tunneling in and out of the shuttling channels. The qubits are shuttled to and from the operation regions between the vertices [Figs. 2(c) and 2(d)] in order to perform single- and two-qubit operations, as well as readout and initialization. Figure 2(c) shows a schematic of the gate structure used in the operation regions.

Single-qubit operations are performed via electric dipole spin resonance (EDSR) in a transverse magnetic field gradient provided by a pair of micromagnets [26]. Alternatively, the micromagnets can be omitted in qubit systems with large spin-orbit interaction [27]. After an electron is shuttled to the operation region and confined under the bottom red gates in Fig. 2(c), a microwave pulse is applied to the control gate labeled MW, to drive spin rotations. Qubit phase rotations can be achieved via geometric phase rotations [28] or by pulsing on the gate labeled J to temporarily detune the electron-spin energy via the Stark effect [3].

For two-qubit operations, two electrons from the vertices adjacent to the operation region are shuttled and confined under the red gates in Figs. 2(c) and 2(d), and a pulsed signal on the gate labeled J activates an exchange interaction between the two electron spins [1,29,30].

Most state-of-the-art spin-qubit systems use one of two spin-to-charge conversion mechanisms to read out a qubit state: (1) spin-selective tunneling to a nearby electron reservoir (Elzerman readout) [31]; or (2) based on Pauli spin blockade (PSB readout) [2], using a second quantum dot with an ancilla electron [17,32]. The charge state of the single or double quantum dot is then identified using a quantum dot tuned as a charge sensor, connected to source and drain Ohmic contacts. Although both readout techniques are compatible with this architecture, we focus here on PSB readout, due to its shorter demonstrated readout times [33,34] and its compatibility with higher temperature operation [21–23].

PSB readout can be used to measure the parity of a spin pair [32]. If prior to the measurement, one of the qubits is prepared in a known eigenstate, the parity measurement will reveal the state of the unknown spin. Parallel spins are identified by their (1,1) charge state, while antiparallel states are identified by their quick relaxation to the (0,2) singlet. The singlet can be adiabatically detuned and separated into a known product state ($|\uparrow\downarrow\rangle$ or $|\downarrow\uparrow\rangle$). After this measurement protocol, the target qubit is left in its projected state and the readout (RO) ancilla qubit RO-ancilla qubit remains in its original state. In the surface code protocol, the measured state of the SC-ancilla qubits can be tracked to perform the required decoding of errors. Alternatively, the target spin can be initialized after measurement by real-time feedback [14] or by letting the state thermalize to the (0,2) singlet after every measurement, then adiabatically detuning as described above. In the spiderweb array, the target electron-spin qubit to be read out is shuttled to the operation region and confined under the bottom red gates in Fig. 2(c), alongside an ancilla electron (RO ancilla) prepared in a known state on the neighboring dot. A PSB readout is performed to obtain a measurement of the target qubit, then the sensing dot can be tuned to

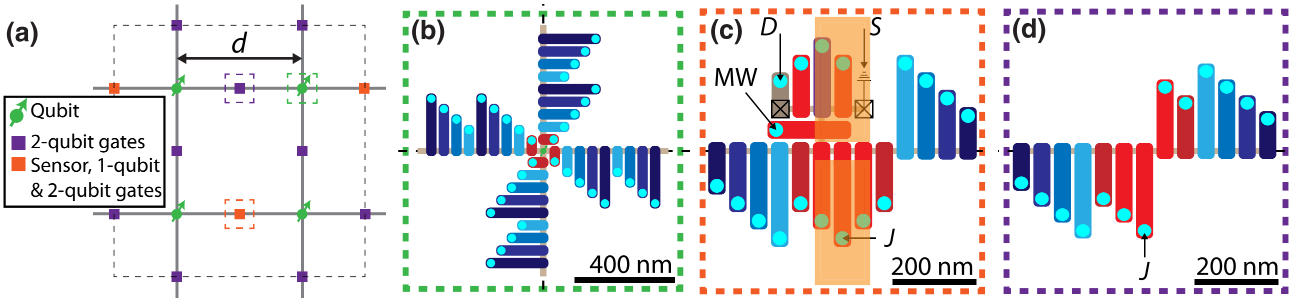

FIG. 2. (a) Schematic (not to scale) of a unit cell containing four spin qubits (green), operation regions (purple and orange), connected via shuttling channels (gray lines). The dashed square indicates the repeating unit cell. (b) Qubit idling region. Four barrier gate electrodes (red) define the confinement potential and allow qubits into the shuttling channels (defined by the blue gate electrodes). Cyan circles represent vias to higher interconnect layers. The vias are arranged in a staggered configuration to enable interconnect routing. (c) Qubit operation region including control gates (red), sensing dot plunger (purple), source (S) and drain (D) Ohmic contacts (squares), and micromagnets (orange rectangles). The gate electrodes labeled MW and J are used for single- and two-qubit operations (see the main text), respectively. (d) Two-qubit operation only regions. Dark red gates in (b)–(d) require coarse-resolution dc biasing, while the light red gates and the purple gate require a fine resolution.

store the RO ancilla, and the two spins can be adiabatically separated with deterministic knowledge of their spin state, provided by the field gradient between the readout and sensing dot regions.

Qubits are able to share an operation region to perform single-qubit gates and only SC-ancilla qubits have to be read out. This allows us to reduce the number of gates in the unit cell by reducing some operation regions to only contain the required gates to perform two-qubit gates [see Fig. 2(d)]. The arrangement of qubits and operation regions shown in Fig. 2(a) constitute a unit cell that can be tiled to obtain a fully operational surface code lattice.

Before the start of a computation, the spiderweb array needs to be initialized by loading electrons with a known spin state onto each of the qubit idling regions. This is achieved by initializing electrons in the operation regions using the two-spin thermalization and adiabatic detuning method described above, and shuttling the electrons to the nearest idling regions. Each operation region will initialize a data qubit first (discarding the second electron), followed by the SC- and RO-ancilla electrons. After these two steps, the entire array is prepared to begin the surface code cycles, which we describe below.

We have designed the unit cell in the spiderweb array such that a cyclic sequence of pulsed signals can be used to perform the required operations to sustain an efficient surface code implementation [35], with the same sequence performed in parallel across all unit cells to sustain it in an arbitrarily large array.

Figure 3(a) shows the circuit diagram of a single surface code cycle in a unit cell (see Appendix D for details of the circuit). The circuit contains single-qubit gates denoted as $R_{\{y,z\}}(\theta)$, where θ is the angle of rotation and the subscript identifies the rotation axis on the Bloch sphere. The two-qubit operation in the schematic is a type of phase gate S_p, decomposed as $-iS_p = \sqrt{S_w}(R_z^{[1]}(\pi) \otimes I^{[2]})\sqrt{S_w}$, where $\sqrt{S_w}$ is the square-root-SWAP operation that is native to the exchange interaction, I is the identity operator, and the superscript in the single-qubit operator indicates which qubit the operation is applied to (see Appendix D for details). Each step in the circuit is implemented by shuttling the participating qubits from their idle position to the operation region to undergo the required single- or two-qubit gate, before being shuttled back to their idle position. Note that each two-qubit gate, as shown in the schematic, consists of three rounds of shuttling to achieve the required steps in the S_p gate. Figure 3(b) shows the required qubit shuttling scheme appropriate to each step of the surface code cycle. The R_y rotations are achieved by tuning the amplitude, phase, and duration of the EDSR microwave pulse, while R_z rotations are phase rotations achieved by using one of the methods described in Sec. II. The $\sqrt{S_w}$ gate is achieved by calibrating the amplitude and duration of the pulse applied to the J gate that controls the exchange interaction between the qubits [see Figs. 2(c) and 2(d)]. At the end of the cycle, the SC-ancilla qubits are measured in the operation regions. The total operation time of a surface code cycle will be

$$t_{\mathrm{SC}} = 22\, t_{\mathrm{sh}} + 14\, t_{1q} + 8\, t_{sw} + t_r, \tag{1}$$

where t_{sh} is the time required to shuttle an electron from the vertex to the operation region and back, t_{1q} is the single-qubit gate duration, t_{sw} is the duration of a $\sqrt{S_w}$ operation, and t_r is the time required to read out the SC-ancilla qubits.

State-of-the-art surface code protocols follow one of two approaches to implementing logic gates. The first, known as *defect qubits* [36], requires disabling a subset of SC-ancilla qubits moving these defects around the lattice while the rounds of surface code error correction continue all around them. The process of moving the defects not only involves disabling and enabling qubits in succession,

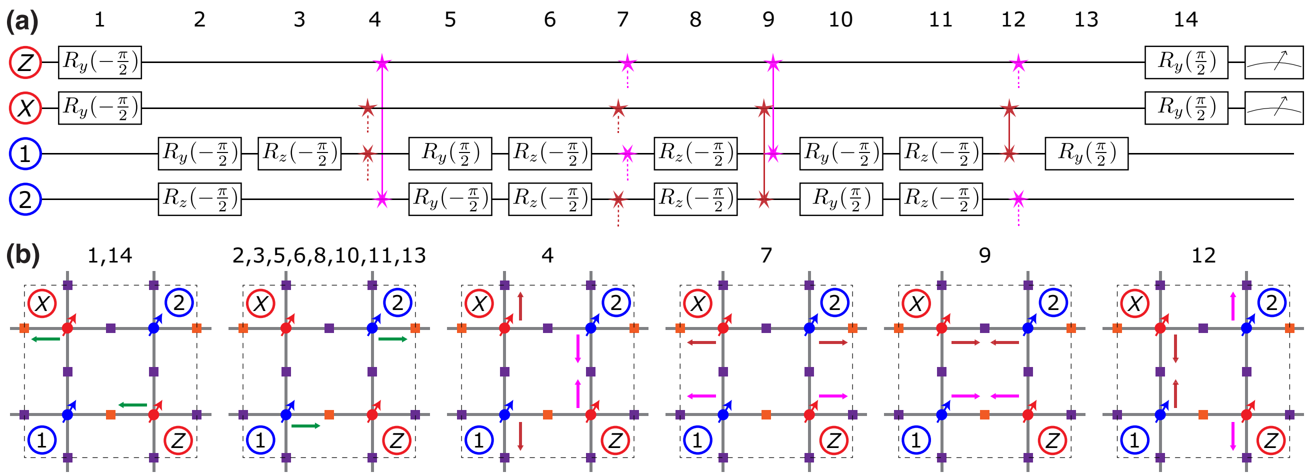

FIG. 3. Surface code operation of the spiderweb array. (a) Circuit diagram showing the surface code cycle for the four-qubit unit cell. Boxes represent single-qubit gates, solid (dashed) lines represent intra- (inter-)unit-cell two-qubit gates. (b) Schematics of qubit shuttling required to perform each of the steps in (a). Note that the two-qubit gates consist of three substeps (as described in the main text), and only the implementation of the $\sqrt{S_w}$ is shown. Also, note that steps 3 and 13 only require shuttling of data qubit 1.

but also requires measurements on individual data qubits. The other implementation, known as *lattice surgery* [37], divides the array into patches, separated by an idle row of interstitial data qubits. Patches are created by splitting a section of the lattice, which requires performing a measurement of the interstitial qubits and subsequently disabling them from the surface code cycles. Patches can also be merged, by initializing the interstitial qubits and reincorporating them into the surface code cycles. The spiderweb array architecture can be designed to accommodate either type of logical qubit implementation. Patches of qubits can be disabled by preventing shuttling of a subset of qubits, while the rest of the surface code operations on all remaining qubits continue. Selective qubit measurement and initialization is more complex to implement, since it will require interruptions of the regular surface code cycle.

III. LOCAL CONTROL ELECTRONICS

The spiderweb array has been designed with the main intention of providing space within the qubit plane to integrate local control electronics, with the ultimate purpose of obtaining a feasible scaling factor between the number of qubits and external control and measurement signals. In this section we describe in detail the implementation and function of these circuits, along with the corresponding routing of the signal lines between the signal-generating source and gates, as summarized in Table I.

A. Biasing and control signals

In principle, the array only requires the generation of control signals for a single unit cell, and that set of signals can be replicated in parallel across the entire array to sustain the surface code. In practice, setting a quantum-dot array to a state where several qubits can be controlled simultaneously with a single set of control signals requires careful calibration of the dc voltages of each gate in the array [14]. Inhomogeneities in the materials and fabrication properties will cause these calibration voltages to vary over a wide range across the lattice, requiring each qubit to have independent biasing.

To mitigate this effect, we design a sample-and-hold scheme to apply a local dc bias to the gate electrodes [38]. We define two bias voltage resolutions ΔV to accommodate different gate functionalities. For gates acting as barriers to shuttling channels (dark red gates in Fig. 2), only a resolution sufficient to maintain an electron in a quantum dot is required and therefore we can afford a coarse resolution $\Delta V_c = 1$ mV. A fine resolution $\Delta V_f = 1\ \mu$V is required for all other plunger and barrier gates [14]. The need for dc biasing of the shuttling gates is eliminated by making the traveling-wave potential large enough to overcome potential landscape inhomogeneities. As shown in Table II, there are a total of 64 gates requiring

TABLE I. Signal routing scheme for the four different types of control lines in the array design. Demux and *S*/*H* abbreviate demultiplexer and sample-and-hold circuitry, respectively.

Gates	Routing
Shuttling (blue)	Source → gate
Pulsed (red)	dc: source → local demux and *S*/*H* → gate ac: source → gate
Sensing dot Plunger (purple)	dc: source → local demux and *S*/*H* → gate ac: source → global demux → gate
Drain contacts	Measurement device ← Ohmic

TABLE II. Number of gates for each region of a unit cell with the associated type of biasing and pulsed signals.

Region	Regions in unit cell	Fine (1 μV)	Coarse (1 mV)	Pulsed gates
Qubit idling	4	0	4	4
Qubit operation	2	7	2	6
Two-qubit only	6	3	2	5
Total per unit cell		32	32	58

dc biasing, with an equal split between gates requiring 1 mV and 1 μV resolutions.

The minimum hold capacitance required to achieve coarse (C_c) and fine (C_f) resolutions is $C_c \approx 0.16$ fF (limited by the electron charge $e/\Delta V_c$) and $C_f \approx 14$ pF (limited by thermal noise $k_B T/\Delta V_f^2$, assuming that power dissipation from the local electronics requires the operating temperature to be raised to 1 K).

The integrated electronics required to implement dc biasing consists of demultiplexers and capacitors that together form sample-and-hold circuits, as schematically depicted in Fig. 4. Local demultiplexers distribute dc voltages generated remotely (i.e., outside the quantum plane) by voltage sources that we have labeled dcDAC, to local capacitors connected to the gate electrodes. The local control electronics for a unit cell needs to be distributed within the d^2 area regions between the qubits, with a total of $4d^2$ open footprints available per unit cell [see Fig. 2(a)]. Therefore, in order to bias 64 gates per unit cell (as per Table II), we fit 1-to-16 demultiplexers in each open region between the qubits, which implies four demultiplexers per unit cell. The demultiplexers are implemented using 4-bit digital decoders, with each output activated using the 4-bit address line.

Figure 5 shows the dc biasing scheme on the unit cell, module, and quantum plane levels. All demultiplexers within a module share the same input dc biasing signal [Fig. 5(b)], and all demultiplexers in the quantum plane share the same address bus [Fig. 5(c)]. The demultiplexers in a module are enabled sequentially by crossbar addressing and in turn sequentially (one by one) update each gate. This way, all modules are updated in parallel and therefore one module refresh cycle is required to refresh the entire qubit array.

The ac signals (MW and pulsed) are generated remotely by sources that we have labeled acDAC. Each signal is distributed throughout the array to their respective gates and the complementary switching circuit (see φ_{ac} and $\overline{\varphi_{\mathrm{ac}}}$) shown in Fig. 6 is used to combine the ac and dc components of the gate voltages.

To implement logical qubits within the surface code lattice, we design an additional scheme to control local patches of data and SC-ancilla qubits. Switches connected to the barrier gates surrounding the idle regions control the tunneling of the qubits into the shuttling channel. When one of these switches is activated, the corresponding qubit will remain in the idle region while the rest of the operations in the array carry on. The signals used to control these switches are arranged in a crossbar fashion across the entire quantum plane, as schematically depicted in Fig. 7. This scheme enables the implementation of either defect qubits or lattice surgery. Defects or interstitial boundaries can be created by disabling patches or rows of qubits, respectively. Selective data qubit measurement or initialization can be performed by temporarily disabling the rest of the array and interrupting the surface code signal to perform the required operations. The crossbar addressing scheme limits the shape of the defects and patches to rectangles of arbitrary size, but meets basic surface code requirements and within those limits allows for universal control.

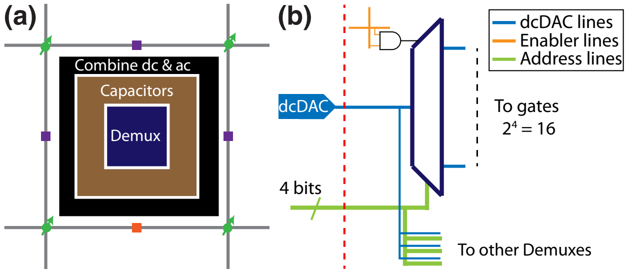

FIG. 4. (a) Schematic of a unit cell containing locally integrated classical electronics. Demultiplexers in combination with capacitors form sample-and-hold circuits to provide dc bias voltages. Additionally, circuits that combine the dc bias with ac control signals are required. (b) Input-output schematic of the demultiplexers. A demultiplexer (dark blue), once enabled via crossbar addressing (orange), ports the dc voltages coming from the dcDAC (light blue) to the output selected by the 4-bit address bus (green). The dashed red line indicates the quantum plane boundary. The color coding in (a) and the legend in (b) represent the same components in both panels as well as in Figs. 5 and 6.

B. Readout signals

It is difficult to realize a favorable Rent's exponent on the measurement circuit because each SC-ancilla qubit needs to be read out independently. In order to use a single line outside the quantum plane to measure more than one qubit, the readout protocol will require some form of multiplexing. With this in mind, we define a *readout module* consisting of N_r^2 unit cells that share a single readout signal line for multiplexed measurements.

The simplest form of multiplexing is to read out qubits sequentially. This is done by connecting the drain contacts of all sensor dots in a readout module to a single line at the quantum plane boundary and consecutively pulsing their plungers to bring them to the low-impedance, electrostatically sensitive regime, while all other sensor dots in the

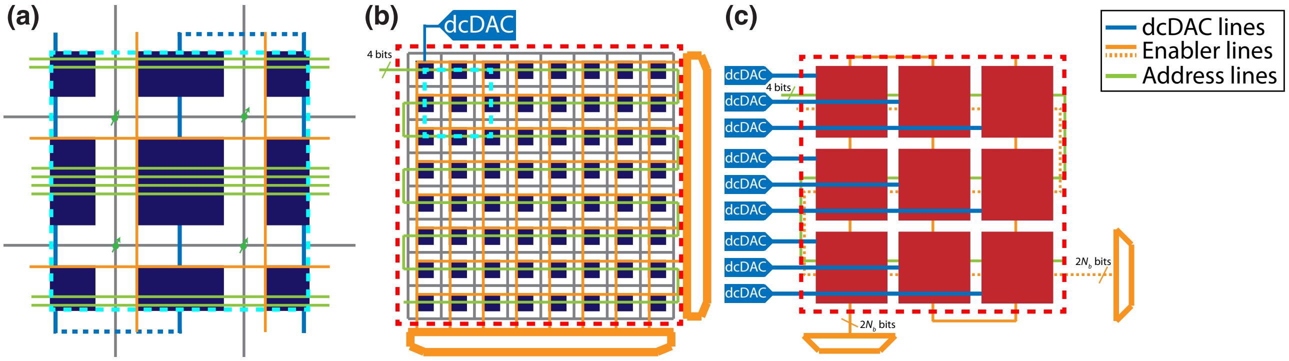

FIG. 5. Schematics of (a) a unit cell and (b) a module. Demultiplexers are sequentially enabled by crossbar addressing controlled by multiplexers (orange blocks). (c) Schematic of the array of modules completing the quantum plane. Dashed red lines in (b) and (c) denote the quantum plane boundary.

readout module are in Coulomb blockade (i.e., in the high-impedance regime). A global readout demultiplexer is used for the sequential control of the sensor plungers in a readout module. This demultiplexer can be shared between all readout modules across the entire array and can be located outside the quantum plane. This method is technically simple to implement, but will be limited by the data qubit coherence time, since it will increase the total surface code cycle time [i.e., the last term in Eq. (1) becomes $N_r t_r$].

Simultaneous readout of a number of qubits can also be achieved using other multiplexing techniques, such as amplitude, frequency, and/or phase modulation. The plungers of the qubits to be read out in parallel can be connected to a single pulsed signal line that activates all sensors simultaneously. The measured signal from the common Ohmic line needs to then be demodulated to extract each individual qubit measurement.

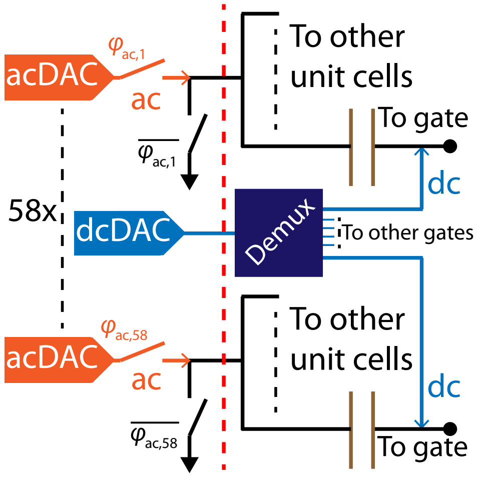

FIG. 6. Circuit schematic to combine ac and dc signals as described in the main text. The acDAC (dcDAC) label denotes voltage sources for pulsed signals (dc biasing). The dashed red line indicates the quantum plane boundary.

Amplitude modulation is achieved by tuning the bias voltages of the plungers from the sensors that are read out simultaneously, such that each sensor response will result in distinct current amplitudes. The currents from all the sensors can then be added into a single output line and the total current amplitude can be used to decode the responses of all the sensors. The number of parallel readouts using this technique will be limited by the signal-to-noise ratio of the sensor response.

Frequency and phase modulation are achieved by applying rf reflectometry [39]. This technique requires the design of resonant circuits that connect to the Ohmics of the charge sensor. The challenge with this multiplexing strategy is that in order to use multiple frequencies combined into a single output line at the qubit plane boundary, the resonant circuits will need to be implemented locally, significantly increasing the footprint requirements of the control electronics.

The evolution in state-of-the-art spin-qubit measurement techniques will determine the most feasible

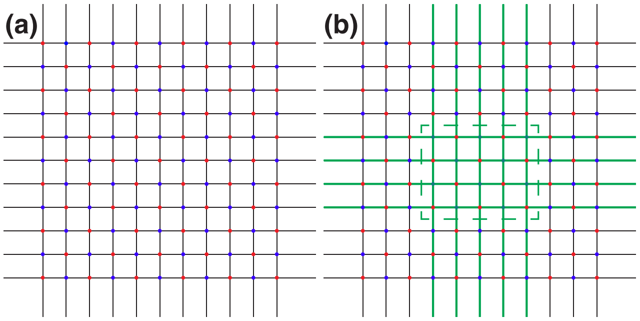

FIG. 7. Crossbar scheme to control local patches of qubits. (a) Crossbars connect all columns and rows of qubits in the array. None of the crossbar lines are activated and the surface code runs on the entire array. (b) Example of the creation of a defect, where the crossbars in green are activated, disabling the qubits inside the dashed green rectangle from tunneling onto the shuttling channel.

readout multiplexing strategy that can be applied in this architecture.

IV. LINE SCALING

The local electronic circuits described above allow for significant sharing of control lines, which results in a very efficient scaling of the ratio of the number of interconnects at the quantum plane boundary to the number of lines at the unit-cell level. The line scaling factors obtained for each type of control and measurement signal are summarized in Table III. We note that some additional interconnects will be required at the unit-cell and module levels in order to route connections to adjacent cells.

The sample-and-hold circuits for independent dc biasing of the gates require $O(M_b^2 + N_b)$ lines at the quantum plane boundary. Concretely, at the unit-cell level, four digital address lines [green lines in Fig. 5(a)] and four enabler lines (orange lines) select a specific output of one demultiplexer, which is set to the correct voltage via a single connection to a voltage source (i.e., dcDAC) outside the quantum plane. This makes a total of nine lines required for dc biasing. At the module level only the enabler lines scale with the number of unit cells as $4N_b$. Meanwhile, the four digital address lines and the connection to the voltage source is shared between all unit cells in a module. Every module is served by one dcDAC, so at the quantum plane level M_b^2 connections to dcDACs are needed, while both the digital address lines and the enabler lines are shared between all modules.

Shuttling of electrons across the array requires four signals that can be fully shared over the entire array.

Since all pulsed and microwave control signals can be shared across all unit cells in the array, a constant number of 58 of these lines (see Table II) are required at the quantum plane boundary to sustain the surface code, irrespective of the total number of qubits.

To control the switches that disable shuttling of qubits from the idle region, we propose to use x crossbars, in order to allow for x qubit patches to be simultaneously created and manipulated, therefore accommodating approximately x logical qubits. To obtain an estimate of x, we first consider that the number of physical qubits required to implement a logical qubit will depend on the maximum error correction distance d_c, measured as the number of neighboring physical qubits that can have errors before the logical qubit produces an error. The maximum number of logical qubits that will fit in the array will then depend on the chosen logical qubit implementation [40]:

$$\begin{aligned} x_{\max,\mathrm{DQ}} &\approx \frac{2U}{3d_c^2} \quad \text{(defect qubits)}, \\ x_{\max,\mathrm{LS}} &\approx \frac{U}{d_c^2} \quad \text{(lattice surgery)}, \end{aligned} \tag{2}$$

with $U = M_b^2 N_b^2$ the number of unit cells in the array. In this line-counting exercise, we consider that each of the crossbars span the entire array, which requires two horizontal and two vertical lines per unit cell. The number of crossbar lines then scales with N_b and $N_b M_b$ at the module and quantum plane levels, respectively. This is a worst-case estimate, since we envision that it will be good enough to define crossbars over partial sections of the array, therefore reducing the total line count. In addition, the number of lines required per crossbar for lattice surgery will likely be significantly less than for defect qubits, since only the interstitial qubits need to be locally addressed.

For the readout signals, a unit cell contains two charge sensors that both require a connection to their plunger and share a single Ohmic line. A module requires a single Ohmic line, and the plunger line scaling will depend on the multiplexing technique used. Performing q sequential readouts will require $\log_2 q$ lines to address the readout decoders. Simultaneous readout of r unit cells will just require a single line that connects all the plungers together. Therefore, a readout module consisting of $N_r^2 = qr$ unit cells will require $2\log_2 N_r - \log_2 r + 1$ lines in total for the readout scheme. To read out the full quantum plane, all M_r^2 modules require their own Ohmic line, while the address lines for the readout demultiplexers are shared.

A. Module size considerations

From the total achievable signal connections at the quantum plane boundary (see Table III), it is clear that Rent's exponent will be minimized if the module size is made as large as possible (i.e., maximize N and thus minimize M).

The main consideration for the dc bias module size N_b is the required refresh rate of the sample-and-hold circuits. Leakage current from the hold capacitors will cause a drift in the dc bias voltage on the gates dV/dt [38,41]. The ratio between the voltage drift and the required voltage stability (which we assume equals the voltage resolution ΔV) will determine the minimum refresh rate

$$f_c = \frac{dV/dt}{\Delta V}. \tag{3}$$

The leakage current is very technology dependent and values are not readily available for state-of-the-art integrated capacitor technology. To date, there have only been two proof-of-principle demonstrations of sample-and-hold circuits integrated with qubit device gates, with reported dV/dt values ranging from 2 μV/s [41] to 0.1 V/s [38]. Using these values, we obtain minimum refresh rates ranging from $f_c = 2$ Hz to 100 kHz, limited by the sample-and-hold circuits with fine resolution $\Delta V_f = 1$ μV. The module size will then set the minimum clock frequency required to run the dc biasing demultiplexers $f_b = 64N_b^2 f_c$. Therefore, the maximum module size will be limited by

TABLE III. Scaling of the number of local connections at different levels of the array.

	Connection level		
Type of line	Unit cell	Module	Quantum plane
dc biasing	9	$4N_b + 5$	$M_b^2 + 4N_b + 4$
Shuttling	4	4	4
Pulsed signals and MW	58	58	58
Logical operations	$4x$	$4N_b x$	$4N_b M_b x$
Readout	3	$2\log_2 N_r - \log_2 r + 1$	$M_r^2 + 2\log_2 N_r - \log_2 r$
Total	$74 + 4x$	$4N_b(1+x) + 2\log_2 N_r$ $- \log_2 r + 68$	$M_b^2 + M_r^2 + 4N_b(1 + M_b x)$ $+2\log_2 N_r - \log_2 r + 66$

the feasibility of distributing the dcDAC signals across the array, which becomes more difficult as f_b increases. This issue will be discussed in more detail below.

The readout module size has two limiting factors. The number of parallel readouts r through a single line will depend on the feasibility of applying readout multiplexing techniques. As readout is generally the operation that takes the most amount of time, q will need to be kept to a minimum in order to restrain t_{SC} to within the appropriate bounds of coherence that will achieve a good quantum memory.

V. FOOTPRINT

We now consider the footprint requirements of the control electronics that need to be locally integrated in the quantum plane. This is the minimum area that needs to be available adjacent to the qubits, and therefore sets the minimum qubit pitch d. The most significant contribution to the footprint comes from the capacitors that are required for the sample-and-hold scheme. As summarized in Table II, 32 gate electrodes per unit cell require a fine voltage resolution and another 32 gates require coarse resolution, which comprise a total capacitance per unit cell of about 450 pF. Assuming about 1 pF/μm^2 (using state-of-the-art deep-trench capacitor technology [42]), we estimate a total capacitor footprint $A_c \approx 450$ μm^2. In addition, we model a decoder circuit using 40-nm technology (see Appendix A for details) and obtain an estimate of the total footprint of the required demultiplexers $A_d \approx 180$ μm^2 per unit cell. This adds to a total footprint per unit cell of $A_u \approx 630$ μm^2, which allows us to set the qubit pitch to $d \gtrsim 13$ μm. Assuming a 50-nm pitch between gate electrodes, this would require linear arrays of 260 gate electrodes per lattice arm and would set the unit cell area to $4\,d^2 \approx 676$ μm^2.

The required unit-cell footprint sets the qubit pitch and consequently the perimeter through which the required interconnects have to enter and leave the unit cell. To evaluate the feasibility of implementing the spiderweb array gate structure integrated with the local control electronics, we construct a circuit model in CADENCE (see Appendix B for details) with realistic parameters that includes all the essential components and connection routing scheme described here, using a total of four metal layers. To obtain an estimate of the maximum number $x_{\mathrm{max,fab}}$ of crossbars that can be added for the implementation of logical qubits, we need to first estimate the maximum number of interconnect lines that can be routed across the perimeter of a unit cell $N_{\mathrm{lines}} = 8dN_{\mathrm{layers}}/\Delta_i$, where N_{layers} is the number of metallic layers that can be used for routing and Δ_i is the pitch of the interconnect lines. Then $x_{\mathrm{max,fab}} = N_{\mathrm{lines}}/8$, since each of the four crossbar lines will cross the unit-cell perimeter twice. Assuming current numbers from the latest device roadmap report [43] $N_{\mathrm{layers}} = 12$ and $\Delta_i = 80$ nm, we estimate $x_{\mathrm{max,fab}} \approx 2000$.

VI. HEAT DISSIPATION

As with all quantum processors that will operate at cryogenic temperatures, it is necessary to ensure that the heat dissipated during the operation is kept to within the stringent requirements set by the cooling power available [14]. In this section we discuss some of the main sources of heat dissipation in the spiderweb array.

Routing of the signal lines across the chip requires a high density of metallic lines at the different interconnect levels, which results in parasitic capacitances between the lines. Any oscillating or pulsed signal on these lines will dissipate energy due to the charging and discharging of these parasitic capacitances. Using the interconnect circuit model drawn in CADENCE as a guide, we consider the two layers with the highest line density as a regular grid of metal lines and use this to obtain an estimate of $C_p = 700$ fF for the total parasitic capacitance of the signal lines in a unit cell (see Appendix C 1 for details). When a capacitor C_p is charged by a voltage source v_p, the energy stored in the capacitor ($0.5C_p v_p^2$) is half the energy supplied by the source ($Qv_p = C_p v_p^2$). The other half is dissipated by the circuit as heat by the parasitic resistance between the voltage source and the capacitor, independent of the resistance value. This is known as the *dynamic power* dissipation from the parasitic capacitance and can be expressed as

$$P_p = \tfrac{1}{2} C_p v_p^2 f_p, \tag{4}$$

where v_p and f_p are the amplitude and frequency of the pulses applied to the lines. We note that this estimate includes power dissipated both by the signal lines and the driving circuit—which will be outside the quantum plane—so it should be taken as a worst-case estimate.

Next, we estimate the dissipation for the sample-and-hold circuits, which comes mainly from the dynamic power consumption of the network of transistor switches driving the hold capacitors. We use the decoder model introduced in Sec. V and described in Appendix A to estimate a maximum energy dissipation of 0.35 pJ for a 4-bit decoder cycling through all 16 outputs. Using the worst-case 100 kHz gate refresh rate previously estimated, we obtain the transient power dissipation $P_d < 140$ nW for four demultiplexers in a unit cell, noting that capacitor leakage rates in state-of-the-art technologies will likely make this dissipation orders of magnitude lower. Additional capacitive-load power from charging and discharging the array of hold capacitors is negligible (about a femtowatt per unit cell).

Another important consideration is the power dissipated due to the finite resistance of lengthy lines carrying ac signals. We model the signal lines as transmission lines with finite resistance and parasitic capacitance to ground (see Appendix C 2 for details), to obtain the following expression for the power dissipation of a line running above a unit cell of length $2d = 26$ μm:

$$P_t = 1.1 \frac{\text{nW ns}^2}{\text{V}^2} (v_t f_t)^2 \qquad (5)$$

with v_t and f_t the amplitude and frequency of the signal, respectively.

The total power that will be dissipated by the operation of the entire array will then be

$$P_T = U(P_p + P_d + P_t). \qquad (6)$$

VII. EXAMPLE: A MILLION-QUBIT ARRAY

To illustrate the advantages of implementing this architecture on a large scale, let us consider a spiderweb array of 2^{20} (approximately 10^6) qubits, or $U = 2^{18}$ unit cells.

To make a concrete assessment of the line scaling, we assume a dc biasing module size $N_b^2 = 1024$, with $M_b^2 = 256$ modules to complete the array. This sets the maximum clock frequency of the biasing demultiplexers to $f_b \approx 6$ GHz. We assume a measurement multiplexing strategy with $q = 4$ sequential readouts of sets of eight SC-ancilla qubits read out in parallel using amplitude, frequency, and/or phase multiplexing ($r = 4$). This implies a readout module size $N_r^2 = 16$ and $M_r^2 = 16\,384$. Omitting for the moment the crossbars required to implement logical qubits, and using the result in Table III, the array contains a total of $c = 74$ connections per unit cell and $T = 16\,836$ connections at the quantum plane boundary. Using the formula for Rent's rule $T = cU^p$, we extract a Rent's exponent $p = 0.43$. Adding crossbar circuits will increase Rent's exponent to a maximum $p = 0.5$ for $x \gtrsim 200$ (see Appendix E for further discussion).

Considering that state-of-the-art qubit fidelities require a code distance $d_c \approx 16$ to perform complex quantum algorithms [40], we can estimate upper bounds of x from Eq. (2): $x_{\text{max,DQ}} \approx 700$ and $x_{\text{max,LS}} \approx 1000$.

The total area covered by the quantum plane of this million-qubit array with local control electronics will be $(2dNM)^2 \approx 177$ mm^2. The remaining area on, for example, a 22×33 mm^2 (726 mm^2) die is about 550 mm^2, and can be used to implement classical control circuits, i.e., among others the pulsed voltage sources we have described. In addition, additional levels of multiplexing can be employed to bring the off-chip wire count, typically being the real bottleneck for Rent's rule, to well below the wire count at the quantum plane boundary.

We can evaluate the total duration of a surface code cycle by estimating the operation times in Eq. (1). From recent modeling of a shuttling protocol similar to that used here [44], we estimate that with a 50-nm gate pitch we can achieve $t_{\text{sh}} \approx 50$ ns, maintaining $> 99.9\%$ fidelity. We note that a recent experimental demonstration of this shuttling protocol [13] showed shuttling fidelities $> 90\%$ for distances of about 420 nm, an encouraging initial result towards the feasibility of this protocol at larger scales. State-of-the-art quantum-dot qubit systems with operation mechanisms similar to those proposed here have demonstrated Rabi frequencies and exchange couplings of about 10 MHz [5,8,45], which correspond in our system to $t_{1q}, t_{sw} \approx 25$ ns. In addition, those same systems exhibit dephasing times as long as $T_2^* \approx 20$ μs. High-fidelity readout using Pauli spin blockade has been achieved on timescales $t_r \approx 1$ μs [33,34]. Assuming these operation times, we estimate that an entire surface code cycle will take $t_{\text{SC}} \approx 6$ μs, more than 3 times shorter than the coherence time.

Finally, we calculate the power dissipation from the sources described in Sec. VI. To estimate the power dissipated from the parasitic capacitance of the interconnects, we first note from Table III that the bulk of lines that need to be routed above the unit cell corresponds to pulsed signals for qubit operations. These signal lines will need to be activated on average about 6 times per surface code cycle, from which we estimate $f_p \approx 1$ MHz. Assuming that $v_p \approx 1$ V of pulse amplitude on these lines, we can use Eq. (4) to estimate $UP_p \lesssim 90$ mW for these lines. The biasing multiplexers will dissipate $UP_d \lesssim 40$ mW for the entire array. We use Eq. (5) to calculate the power dissipation of the lossy transmission lines carrying the higher-frequency signals. The MW signals used for single-qubit control are relatively low amplitude and pulsed with very short duty cycle, so their power dissipation will be negligible. A larger contribution will come from the lines that carry the signals for the shuttling channels, since they will be nearly continuously activated. Assuming a voltage amplitude $v_t = 1$ V and frequency $f_t = 1$ GHz, we estimate $UP_t \lesssim 0.3$ mW. These estimates suggest that the parasitic capacitance and the local demultiplexers will be the main contributors to the total power dissipation in the array, with a million-qubit spiderweb array expected to dissipate power of the order of 100 mW. This is well within the cooling power capabilities for 1 K operation using state-of-the-art cryogenic technologies. We can also estimate how well this power can be transferred from the chip to the cryogenic environment with currently used heat sinking techniques. Recent work [46] studying a CMOS integrated chip in a cryogenic environment (3 K) found that a circuit dissipating 200 mW increased the chip temperature by 7 K over an area of about 4 mm^2. Considering that the qubit plane area is 40 times larger than that, we can roughly estimate about 0.2 K of self-heating in the qubit plane of the spiderweb array.

VIII. DISCUSSION

The goal of this work is to provide insight into the feasibility of implementing a quantum hardware architecture with integrated local control electronics. We show that, with reasonable assumptions regarding quantum and classical fabrication and measurement techniques, a million-qubit array made from spins in quantum dots is achievable. We use the state of the art in quantum-dot qubit development to consider a range of technical implementation aspects, and now discuss some of the open questions that will need to be addressed based on how the technology evolves. As with most other electrically controlled solid-state qubit architectures, the distribution of high-frequency signals is not trivial, with potential issues such as standing waves and signal synchronization across the array. Some additional local circuitry will likely be needed to solve these issues.

Furthermore, qubit operation performance could be significantly improved with significant restructuring of the array topology, to place micromagnets at the qubit idle regions and add a fast gate to perform single-qubit gates. This would reduce the number of shuttles per surface code cycle, allow for dynamical decoupling during idle times to extend coherence times, and correct phase errors that may arise, e.g., from the shuttling process.

If the number of crossbars becomes the limiting factor to Rent's exponent, it would be beneficial to move the dc biasing and readout demultiplexers outside the quantum plane. This would reduce the footprint and in turn relax the shuttling performance requirements.

This work builds on previous large-scale qubit architectures based on quantum dots, focusing on the implementation of integrated classical and quantum electronics. We believe it will help identify the key areas of technological development required to shortcut the path towards building the future generation of quantum computers.

ACKNOWLEDGMENTS

We would like acknowledge Simon Devitt and Ramon W. J. Overwater for useful discussions. This work is supported by Intel Corporation and the Early Research Programme of the Netherlands Organisation for Applied Scientific Research (TNO) with additional support from the Top Sector High Tech Systems and Materials.

APPENDIX A: DEMULTIPLEXER DESIGN

In order to estimate the footprint and the power consumption of the demultiplexer used to route the dc biasing to the gates in a unit cell, we model a 1-to-16 demultiplexer using a 4-bit digital decoder driving an array of 16 switches. As explained in the main text, each unit cell accommodates four 4-bit demultiplexers to bias 64 gates, so that the demultiplexers can be easily placed in the four free areas within the quantum-dot grid.

The switches can be implemented as a simple transistor switch (NMOS or PMOS) or by a pass gate (NMOS and PMOS in parallel) depending on the voltage levels expected by the gates, the demultiplexer supply, and the threshold voltage of the transistors in the employed technology. Because of the increased threshold voltage at cryogenic temperatures with respect to room temperature, a dead zone for voltage biasing around mid supply may appear, in which even the pass-gate impedance could be too high to allow proper biasing of the gates in the unit cell [47]. Possible alternatives could then be the well-known bootstrapped switches or the use for thick-oxide transistors driven by level shifters (or eventually driven by a thick-oxide decoder). Noting that the abovementioned design choices strongly depend on the required dc biasing levels, we assume approximate 1 V outputs and design a standard thin-oxide decoder in a nanometer CMOS technology driving single-transistor switches. Independent of the design choice, we expect the effective footprint and energy to be well within an order of magnitude of the reported estimates.

A 4-bit decoder enabled by the combination of a row and column address (see Fig. 4) has been designed in VERILOG and synthesized using a commercial TSMC 40-nm CMOS process. After place and route, the decoder occupies an area of 36 μm^2. By budgeting 25% extra area for the switch array, each demultiplexer would occupy 45 μm^2 in a 40-nm CMOS, and the total footprint required to fit four demultiplexers in a unit cell will be $A_d = 180$ μm^2.

To estimate the power dissipation, the decoder binary input has been swept from 0000 to 1111, leading to an energy dissipation between 0.2 and 0.35 pJ from a 1-V supply for the whole 16-phase cycle for a decoder load ranging from 1 to 10 fF to emulate the switch input

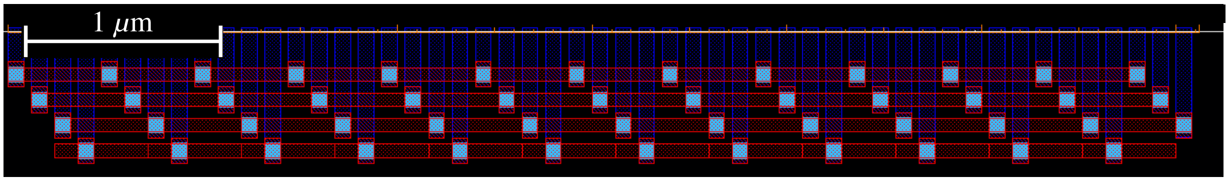

FIG. 8. Shuttling arm of the spiderweb array (> 6 μm) consisting of concatenated sets of four-gate electrodes residing in the bottom metal layer (blue). Every fifth gate electrode is connected together via the second metal layer (red). Vias (cyan) connect the two metal layers and are arranged in a staggered configuration to enable interconnect routing. All gate electrodes carry a sinusoidal signal with equal frequency and amplitude, with a phase shift of $\frac{\pi}{2}$ between neighboring gates in order to create a traveling-wave potential.

capacitance. A unit cell containing four demultiplexers will then dissipate a maximum energy of 1.4 pJ. This estimate uses the standard room-temperature device models and includes the parasitics extracted from the layout, but excludes the power required to drive the demultiplexer inputs.

APPENDIX B: CADENCE DRAWINGS

In order assess more concretely the feasibility of implementing the local classical control electronics described in the spiderweb array, we draw a unit cell with all its elements using CADENCE circuit design software. Although we use realistic gate pitch and width dimensions (80 nm) for the densest part of the array, the drawings do not strictly follow design rules and are not optimized, as they are intended as an illustrative feasibility exercise. Additionally, we do not implement the connection that will be required for readout (these will depend on the exact implementation) or the lines connecting the dcDACs outside of the quantum plane to the local demultiplexers. Footprint is reserved for the local demultiplexers themselves based on the design described above, but they are not implemented explicitly. We do not draw the two-dimensional electron gas (2DEG) channels that run underneath the end of the gate electrodes, which are electrostatically depleted to create quantum dots.

For all the drawings presented, we use the following color convention: metal layers from lowest to highest are blue, red, green, and pink, while vias connecting different layers are represented by squares colored cyan, orange, and purple, connecting to the second, third, and fourth layers, respectively. White areas represent doped regions.

Figure 8 shows the drawing of an approximate 6.5 μm shuttling arm. Identical copies of these shuttling arms link every qubit idling and operation regions to form the spiderweb array. The 2DEG channel runs horizontally in the top of the image below the tips of the blue gate electrodes.

Figure 9 show the two types of operation regions, as shown schematically in Fig. 2 of the main text. In these images, the 2DEG channels run below the tips of the blue gate electrodes, along the horizontal white line in the center. To combine dc biasing signals with ac pulses, the quantum-dot gates are connected to dc biasing capacitors in a sample-and-hold scheme, as explained in detail in the main text. Coarse-resolution capacitors are visible in Fig. 9 (yellowish structures). For simplicity, we have represented these capacitors here by parallel plate capacitors, although we envision using more advanced capacitor types that have a higher capacitance per unit area. The capacitor footprint used in these drawings (and not the parallel plate capacitance that it represents) is in agreement with our estimates in the main text. The gate electrodes for which their capacitor is not seen in the image connect to larger, fine-resolution capacitors that are not shown here, but are visible (as larger greenish hatched squares) in Fig. 12 below. The green wires from the third metal layer carry the ac signals and are routed to the corresponding gates in neighboring unit cells. The top part of Fig. 9(a) shows two doped square regions in white that serve as source and drain reservoirs for the sensing dot, as well as electron reservoirs for initialization of the spiderweb array. The other doped regions in Fig. 9 are part of the capacitor structures. Connections to the boundary of the quantum plane for any of these doped regions are not implemented. In the bottom part of Fig. 9(a) a short connection via the second metal layer (red) is required to bypass another metal line before connecting to the third metal layer (green).

The qubit idle region is surrounded by four shuttling arms, as shown in Fig. 10. The four gates directly adjacent to the qubit idle region act as gate keepers that control the transfer of electrons into the shuttling arms. They are activated through a transistor structure arranged in an AND configuration. Two of these transistors are visible in the top right and bottom left of the images, while the remaining two transistors are outside the field of view. The fourth metal layer (pink) connects the transistors to the gate electrodes, as is visible in Fig. 10(c). The second ring of gate electrodes around the qubit idling region (i.e., the first set of shuttling gates) connect to a line in the second metal layer (red), visible in Fig. 10(a). The subsequent three rings of gate electrodes that make up the rest of the shuttling set connect to three lines from the third metal layer (green), visible in Fig. 10(b). The horizontal green wires that are visible in Fig. 10(b) connect the shuttling

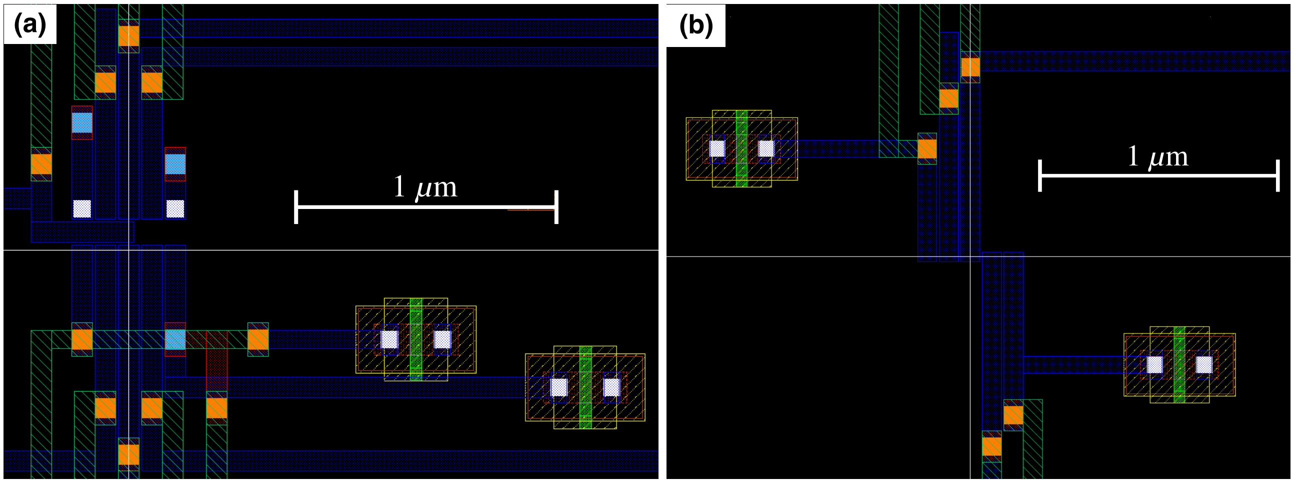

FIG. 9. The two types of qubit operation regions, showing gate configurations for (a) single- and two-qubit operations, as well as readout, and (b) two-qubit operations only. The rulers indicate a length of 1 μm.

signals to the next qubit idle regions, and the vertical pink wires in Fig. 10(c) connect the ac signals in adjacent unit cells together. The idle regions for ancilla and data qubits are very similar, with minor differences to allow for the crossbar addressing to prevent data qubits from being shuttled in the surface code operation. The images shown here correspond to the ancilla qubits.

Figure 11 progressively shows larger areas of the CADENCE drawing, leading to the drawing of the full unit cell in Fig. 12. All layers are included in both figures. Both panels of Fig. 11 are centered on the qubit idling region. The horizontal (red and pink) and vertical (green and pink) wires connect the ac signals in adjacent unit cells together. The open areas visible in the corners of Fig. 11(b) and as nine green open squares in Fig. 12 are reserved footprints for the local demultiplexers used for dc biasing. The greenish hatched squares on the perimeter of the full unit-cell image are fine-resolution capacitors for the sample-and-hold scheme.

We draw the unit cell, with a qubit pitch of about 13 μm, in such a way that it can be tiled horizontally and vertically to form modules, with lines ending at each edge of the unit cell aligned to connect to their respective counterparts at the opposite edge of the adjacent unit cells, and the ancillary structures (such as the fine-resolution capacitors) neatly complement each other.

APPENDIX C: HEAT DISSIPATION ESTIMATES

1. Parasitic capacitance

To assess the heat dissipation of the signal lines, we consider the dominant contribution from the parasitic capacitance caused by having metallic lines next to each other, as well as overlapping across layers. Here we estimate this capacitance based on simplified models, noting that a more accurate estimate can be obtained using capacitance simulation software. We consider a grid of metallic lines such as that shown schematically in Fig. 13, consisting of a set of parallel lines in one layer, overlapped by another set of orthogonally placed lines placed on a layer above. The two main contributors to the total parasitic capacitance will be the capacitance between parallel lines in the same layer C_1 and the capacitance caused by overlapping metal regions across two layers C_2. The total parasitic capacitance of a metallic grid with N_l lines per layer is then $C_p = 2N_lC_1 + N_l^2C_2$. We consider both the parallel plate capacitances and the fringe capacitance between metal edges, to obtain the capacitance model [48,49]

$$C_1 = \epsilon L\left\{\frac{H}{d_1} + \left[377\pi v_0 \ln\left(-2\frac{\sqrt{\alpha_1+1}}{\sqrt{\alpha_1-1}}\right)\right]^{-1}\right\},$$
$$C_2 = \epsilon W\left(3.285\frac{W}{d_2} + 9.01\alpha_2 - 8.696\alpha_2^2\right),$$

with

$$\alpha_1 = \frac{d_1}{d_1+2W} \quad \text{and} \quad \alpha_2 = \frac{H}{H+0.2d_2},$$

where L, W, and H are the length, width, and height of a metal line; d_1 is the distance between neighboring metal lines in the same layer; d_2 is the thickness of the dielectric separating two layers; ϵ is the permittivity of the dielectric between metals; and v_0 is the speed of light in vacuum.

To estimate the C_p of a unit cell, we consider the bottom two layers as the main contributors, since they contain the highest density of signal lines. Metal layers higher up would have signal lines separated widely, thus bringing in less parasitic capacitance. Vias that make connections between layers introduce a parasitic capacitance as well, while it would be reduced through proper layout design

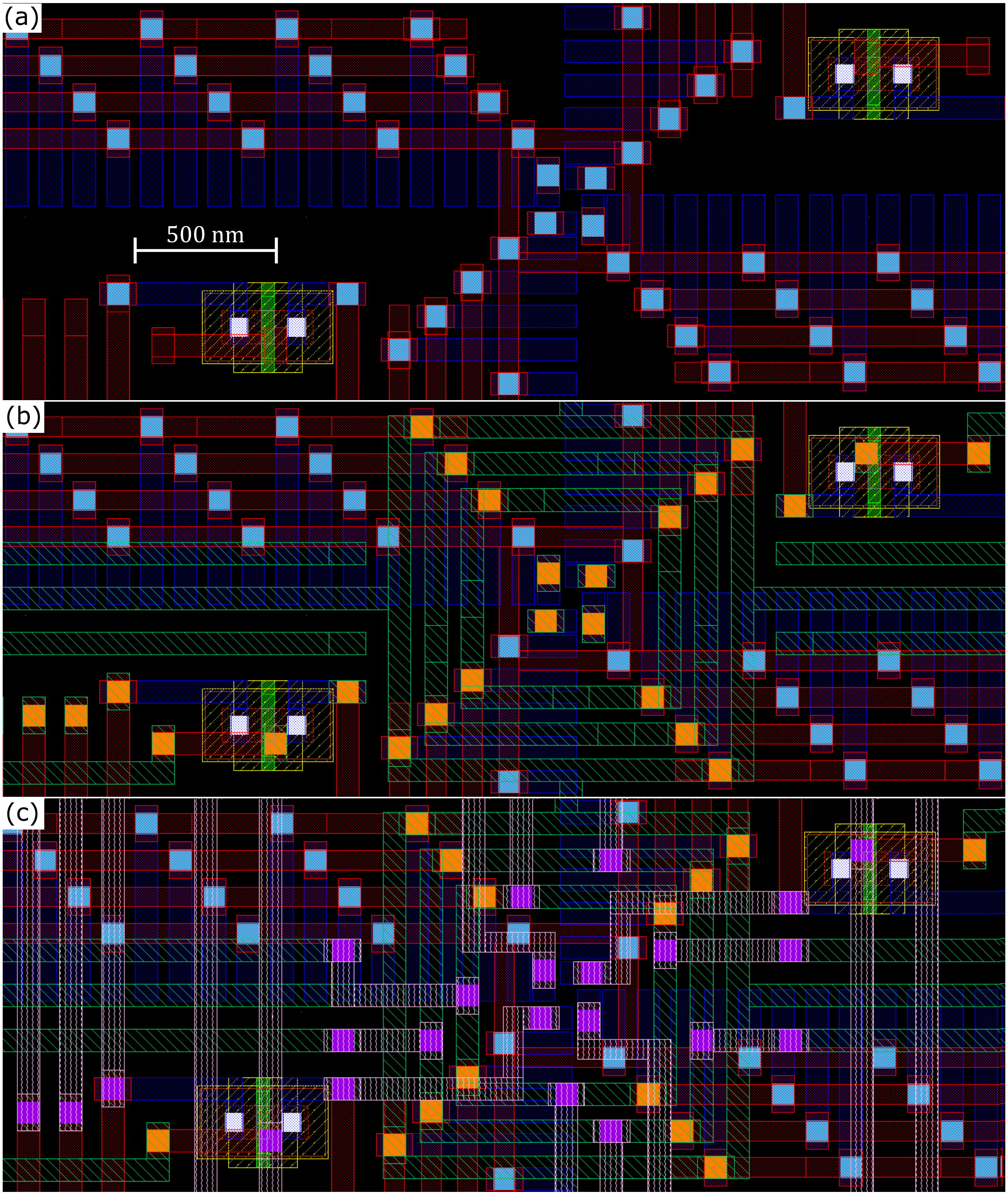

FIG. 10. The qubit idling region, showing (a) the first two, (b) three, and (c) four metal layers.

that avoids placing vias close together. Using the drawings in Appendix B as a guide, we assume that $N_l = 150$, $L = 24$ μm, $W = 80$ nm, $H = 50$ nm, $d1 = 80$ nm, and $d_2 = 500$ nm. The dielectric is assumed to be SiO$_2$, making $\epsilon = 3.9\epsilon_0$, where ϵ_0 is the vacuum permittivity. From this we obtain the estimate $C_p \approx 700$ fF used to

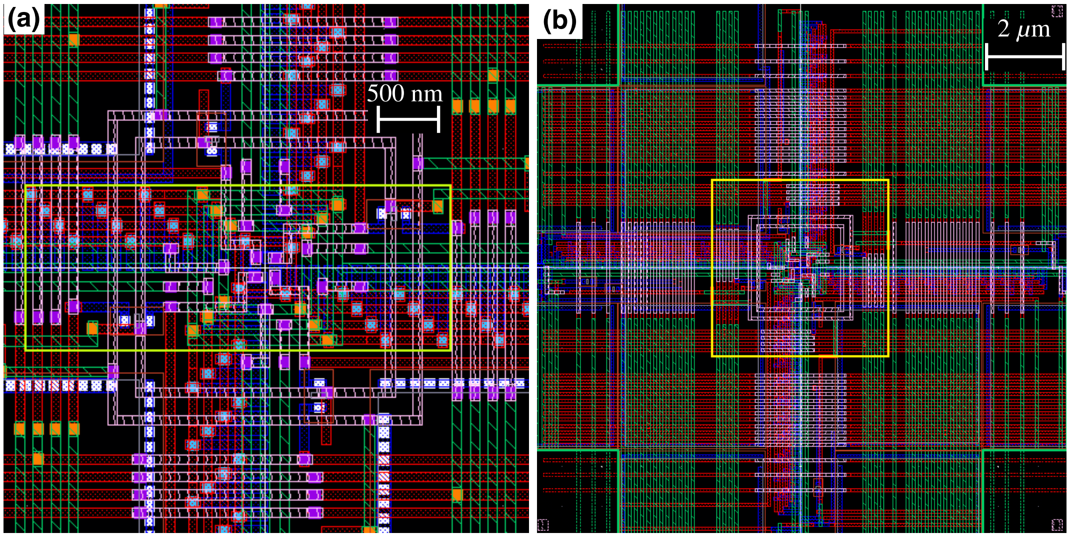

FIG. 11. Zoomed-out images of the qubit idle region. The light green rectangle in (a) corresponds to the area shown in Fig. 10. The yellow square in (b) has a dimension of about 4.4 μm and indicates the area shown in (a).

calculate this portion of the power dissipation in the main text.

2. ac transmission lines

To estimate the power dissipated by lossy transmission lines, we model a segment of transmission line running above a unit cell, as a resistor R_t in series with a capacitor C_t to ground. The amplitude of the current drawn by the capacitor while sustaining a signal of amplitude v_t and frequency f_t is $i_c = 2\pi v_t f_t C_t$. This current flowing through the finite resistance will cause a power dissipation $P_t = i_c^2 R_t/2 = 2R_t(\pi v_t f_t C_t)^2$. Assuming transmission lines with capacitance to ground of about 0.2 fF/μm and a sheet resistance of about 0.1 $\Omega/\square$ [50] with a width of about 1 μm, we obtain the expression in Eq. (5) used in the calculation in the main text. This is a conservative estimate for our application, since the resistance of the lines is expected to reduce at cryogenic temperatures [51].

APPENDIX D: USING THE SQUARE-ROOT-SWAP OPERATION IN THE SURFACE CODE CYCLE

A scalable scheme for implementing error-correction cycles in a surface code for superconducting qubits has recently been presented [35], where an effective controlled-NOT (CNOT) gate is applied by combining a conditional phase gate with appropriate single-qubit Hadamard (**H**) operations. Here we implement a similar scheme, adapted to efficiently accommodate the square-root-SWAP ($\sqrt{S_w}$) operation, i.e., the native two-qubit gate of the spiderweb array, defined as

$$\sqrt{S_w} = \frac{1}{2}\begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 1+i & 1-i & 0 \\ 0 & 1-i & 1+i & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}$$

$$= \frac{\sqrt{2}}{2}\begin{pmatrix} \sqrt{2} & 0 & 0 & 0 \\ 0 & e^{i\pi/4} & e^{-i\pi/4} & 0 \\ 0 & e^{-i\pi/4} & e^{i\pi/4} & 0 \\ 0 & 0 & 0 & \sqrt{2} \end{pmatrix}.$$

We propose $\sqrt{S_w}$ modifications of the X and Z plaquettes that are the building blocks of larger error-correction schemes.

1. Entangling phase gate

An explicit calculation of an operator product proposed in Refs. [1,52] yields

$$\sqrt{S_w}(R_z^{[1]}(\pi) \otimes I^{[2]})\sqrt{S_w} = -iS_p, \tag{D1}$$

where we have defined the phase gate as

$$S_p = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & -i & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}.$$

This relation is pictorially shown in Fig. 14(a) and in what follows the S_p gate will be depicted as in Fig. 14(b).

Note that S_p is an entangling gate, in the sense that it can map a product state into an entangled state. This is

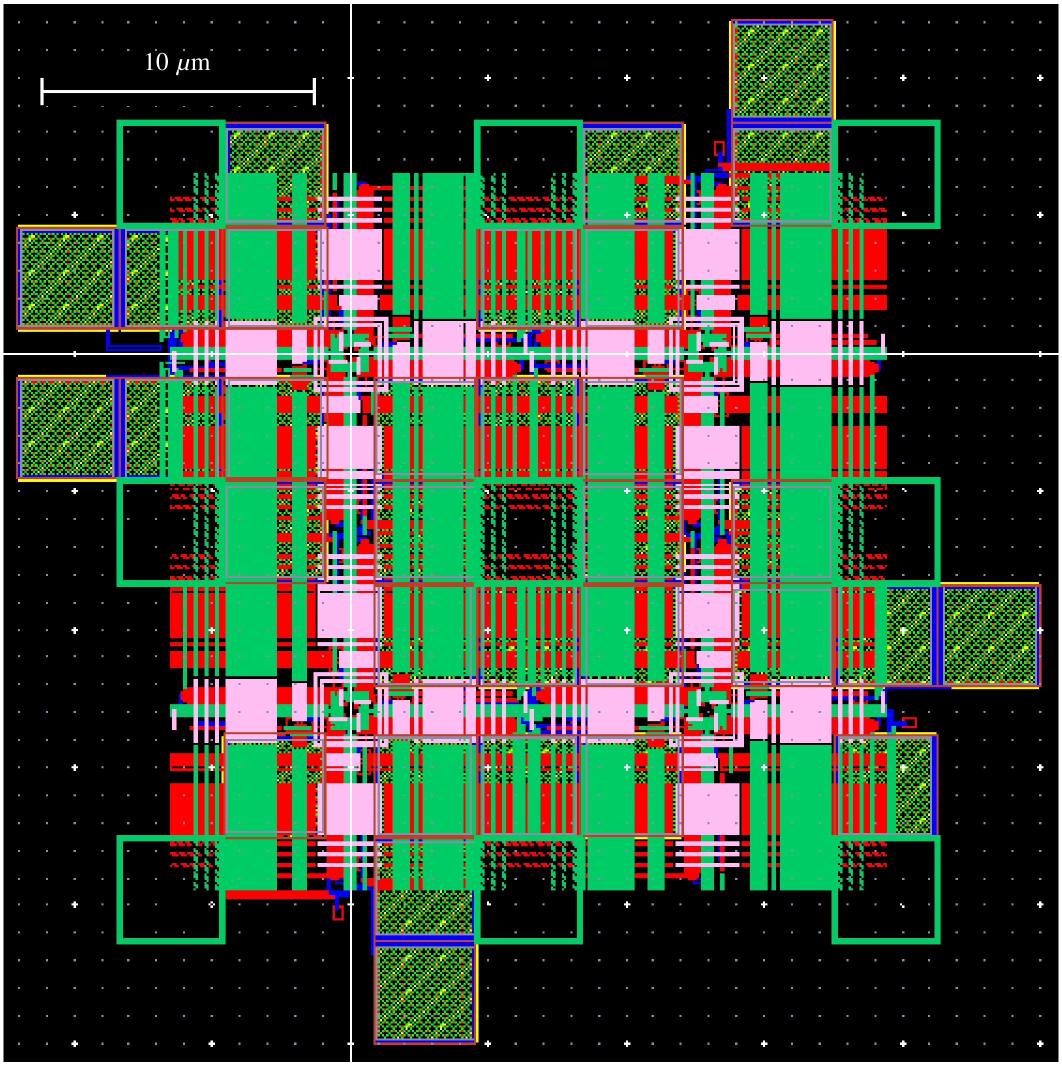

FIG. 12. Full unit cell of the spiderweb array. The distance between two qubit idle regions is about 13 μm.

most easily verified by using the concurrence of a two-qubit state. Its Hermitian conjugate $S_p^\dagger$ can be written as $-iS_p^\dagger = \sqrt{S_w}[I^{[1]} \otimes R_z^{[2]}(\pi)]\sqrt{S_w}$.

The controlled-phase gate C_Z may be used as the primitive data-ancilla interaction in a surface code [35]. With only two single-qubit rotations, this C_Z gate is constructed from S_p as

$$\begin{aligned} C_Z &= [R_z^{[1]}(\pi/2) \otimes I^{[2]}][I^{[1]} \otimes R_z^{[2]}(-\pi/2)]S_p \\ &= [R_z^{[1]}(\pi/2) \otimes R_z^{[2]}(-\pi/2)]S_p. \end{aligned} \tag{D2}$$

Since the three matrices involved are diagonal, they commute and the order of applying the operations is immaterial. This can be advantageous in actually implementing error-correction cycles. To this end, the following relation with the Hermitian conjugate is convenient as well:

$$\begin{aligned} C_Z &= [R_z^{[1]}(-\pi/2) \otimes I^{[2]}][I^{[1]} \otimes R_z^{[2]}(\pi/2)]S_p^\dagger \\ &= [R_z^{[1]}(-\pi/2) \otimes R_z^{[2]}(\pi/2)]S_p^\dagger. \end{aligned} \tag{D3}$$

If one alternatively desires the CNOT gate as the primitive data-ancilla interaction [35], two additional Hadamard gates are necessary [30]. The result can be written as

$$\begin{aligned} C_N &= i[R_z^{[1]}(\pi/2) \otimes I^{[2]}][I^{[1]} \otimes R_z^{[2]}(\pi/2)] \\ &\quad \times [I^{[1]} \otimes R_x^{[2]}(\pi/2)]S_p(I^{[1]} \otimes \mathbf{H}^{[2]}). \end{aligned} \tag{D4}$$

Here, noncommuting operators have to be applied in the right order. A similar relation involving $S_p^\dagger$ can be readily be derived.

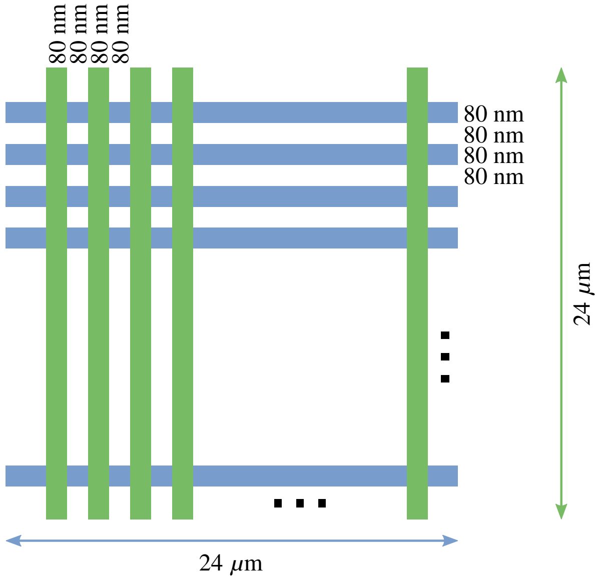

FIG. 13. Schematic of signal lines in a 24 × 24 μm^2 unit cell. The line width and lateral distance in between are both 80 nm. The first layer (blue) and second layer (green) are separated by presumably 500-nm SiO_2 (not shown in this top-view figure).

2. Surface code cycles

We adapt the pipeline-based error-correction scheme [35] to use it with the S_p operator described above. To achieve this, we are only required to reanalyze the X-type and Z-type plaquettes using the C_Z ancilla-data interaction.

For the X-type plaquette, we insert $\mathbf{H} = R_y(\pi/2)Z = ZR_y(-\pi/2)$ for all Hadamard operations. The C_Z gates are to be implemented as in Eq. (D2); here the actual realization of S_p as in Eq. (D1) is implicitly assumed but irrelevant for the remaining analysis. Because of the earlier mentioned commutativity of the diagonal operators, the circuit can be readily simplified. In particular, the Z operators stemming from the initial and final Hadamard gates combine to the identity. The Z rotations can be combined as well. All final Hadamard gates are replaced by $R_y(\pi/2)$. The C_Z ancilla-data operations are replaced by S_p gates without changing the time ordering.

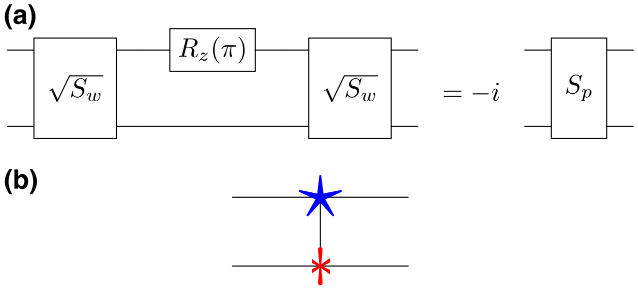

FIG. 14. The S_p gate. (a) Decomposition in a quantum circuit diagram. (b) Simplified circuit diagram representation.

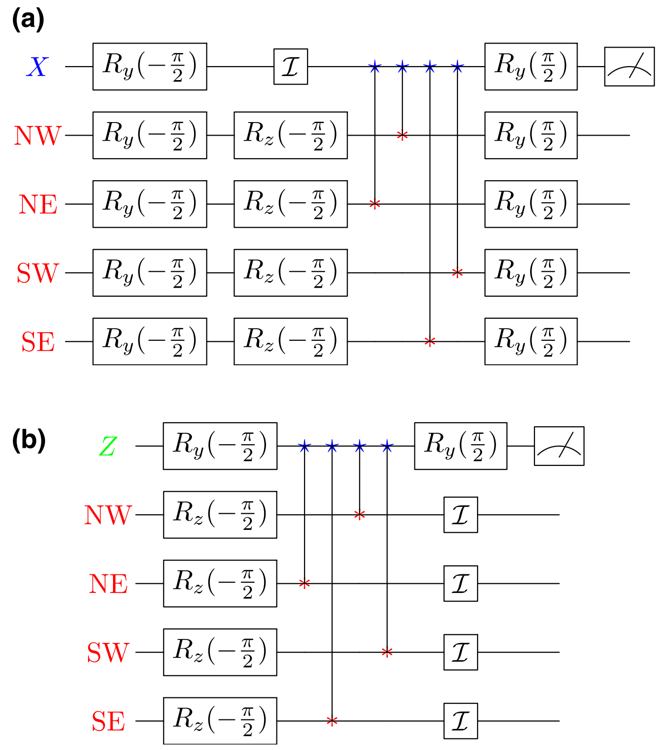

FIG. 15. Quantum circuits for surface code cycles. (a) Type-X plaquette. (b) Type-Z plaquette.

The initial Hadamard operation on the ancilla is replaced by $[R_z(\pi/2)]^4 R_y(-\pi/2) = -R_y(-\pi/2)$, where the minus sign can be omitted. The initial Hadamard data-qubit gates are replaced by the product $R_z(-\pi/2)R_y(-\pi/2)$. The resulting circuit diagram for the X-plaquette cycle is depicted in Fig. 15(a).

The Z plaquette is modified analogously. The final Hadamard gate on the ancilla is again replaced by $R_y(\pi/2)$, whereas the initial Hadamard operation is replaced by $-R_y(-\pi/2)$. Once more, C_Z ancilla-data operations are replaced by S_p gates at the same instant in time. For the data qubits, rotations $R_z(-\pi/2)$ have to be added at some instant in time. In the circuit shown in Fig. 15(b), it is done at the onset.

These building blocks can be used to construct depth-nine quantum circuits for parallel as well as pipeline-based, quantum-error-correction cycles of a surface code; the proposed scheme is scalable.

APPENDIX E: RENT'S EXPONENT CONSIDERATIONS

We calculate Rent's exponent using

$$p = \log_U \left(\frac{T}{c}\right),$$

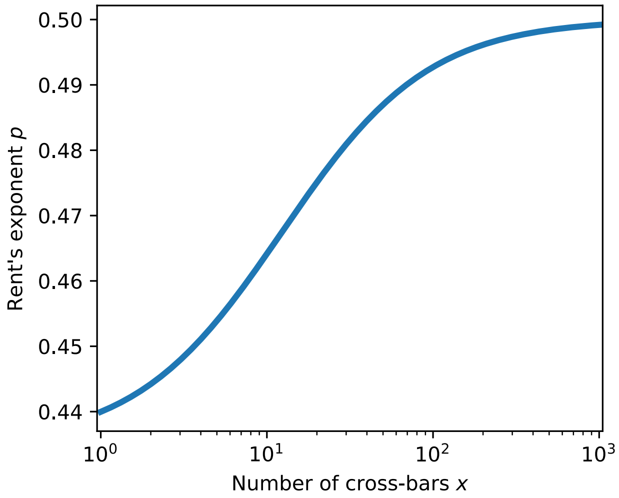

FIG. 16. Rent's exponent as a function of the number of cross-bars used for logical operations in the spiderweb array. All other counts in parameters that affect Rent's exponent are as per the example in Sec. VII.

where, as per Table III, $T = M_b^2 + M_r^2 + 4N_b(1 + M_b x) + 2\log_2 N_r - \log_2 r + 66$ is the total number of connections on the quantum plane, $c = 74 + 4x$ is the number of connections in a unit cell, and U is the number of unit cells.

Using the numbers presented in the example in Sec. VII, we obtain the quoted value $p = 0.43$ for an array without any cross bar circuits ($x = 0$, i.e., a single-qubit memory). As we add crossbar circuits for the implementation of logical operations, Rent's exponent will increase as per Fig. 16 and saturate at a value of 0.5.

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

[2] F. A. Zwanenburg, A. S. Dzurak, A. Morello, M. Y. Simmons, L. C. L. Hollenberg, G. Klimeck, S. Rogge, S. N. Coppersmith, and M. A. Eriksson, Silicon quantum electronics, Rev. Mod. Phys. **85**, 961 (2013).

[3] M. Veldhorst, J. C. C. Hwang, C. H. Yang, A. W. Leenstra, B. de Ronde, J. P. Dehollain, J. T. Muhonen, F. E. Hudson, K. M. Itoh, A. Morello, and A. S. Dzurak, An addressable quantum dot qubit with fault-tolerant control fidelity, Nat. Nanotechnol. **9**, 981 (2014).

[4] E. Kawakami, T. Jullien, P. Scarlino, D. R. Ward, D. E. Savage, M. G. Lagally, V. V. Dobrovitski, M. Friesen, S. N. Coppersmith, M. A. Eriksson, and L. M. K. Vandersypen, Gate fidelity and coherence of an electron spin in a Si/SiGe quantum dot with micromagnet, PNAS **113**, 11738 (2016).

[5] J. Yoneda, K. Takeda, T. Otsuka, T. Nakajima, M. R. Delbecq, G. Allison, T. Honda, T. Kodera, S. Oda, Y. Hoshi, N. Usami, K. M. Itoh, and S. Tarucha, A quantum-dot spin qubit with coherence limited by charge noise and fidelity higher than 99.9%, Nat. Nanotechnol. **13**, 102 (2018).

[6] X. Xue, T. F. Watson, J. Helsen, D. R. Ward, D. E. Savage, M. G. Lagally, S. N. Coppersmith, M. A. Eriksson, S. Wehner, and L. M. K. Vandersypen, Benchmarking gate fidelities in a Si/SiGe two-qubit device, Phys. Rev. X **9**, 021011 (2019).

[7] W. Huang, C. H. Yang, K. W. Chan, T. Tanttu, B. Hensen, R. C. C. Leon, M. A. Fogarty, J. C. C. Hwang, F. E. Hudson, K. M. Itoh, A. Morello, A. Laucht, and A. S. Dzurak, Fidelity benchmarks for two-qubit gates in silicon, Nature **569**, 532 (2019).

[8] T. F. Watson, S. G. J. Philips, E. Kawakami, D. R. Ward, P. Scarlino, M. Veldhorst, D. E. Savage, M. G. Lagally, M. Friesen, S. N. Coppersmith, M. A. Eriksson, and L. M. K. Vandersypen, A programmable two-qubit quantum processor in silicon, Nature **555**, 633 (2018).

[9] J. Yoneda, K. Takeda, A. Noiri, T. Nakajima, S. Li, J. Kamioka, T. Kodera, and S. Tarucha, Quantum non-demolition readout of an electron spin in silicon, Nat. Commun. **11**, 1144 (2020).

[10] X. Xue, B. D'Anjou, T. F. Watson, D. R. Ward, D. E. Savage, M. G. Lagally, M. Friesen, S. N. Coppersmith, M. A. Eriksson, W. A. Coish, and L. M. K. Vandersypen, Repetitive quantum nondemolition measurement and soft decoding of a silicon spin qubit, Phys. Rev. X **10**, 021006 (2020).

[11] J. Yoneda, W. Huang, M. Feng, C. H. Yang, K. W. Chan, T. Tanttu, W. Gilbert, R. C. C. Leon, F. E. Hudson, K. M. Itoh, A. Morello, S. D. Bartlett, A. Laucht, A. Saraiva, and A. S. Dzurak, Coherent spin qubit transport in silicon, Nat. Commun. **12**, 4114 (2021).

[12] T. A. Baart, M. Shafiei, T. Fujita, C. Reichl, W. Wegscheider, and L. M. K. Vandersypen, Single-spin CCD, Nat. Nanotechnol. **11**, 330 (2016).

[13] I. Seidler, T. Struck, R. Xue, N. Focke, S. Trellenkamp, H. Bluhm, and L. R. Schreiber, Conveyor-mode single-electron shuttling in Si/SiGe for a scalable quantum computing architecture, arXiv:2108.00879 [cond-mat.mes-hall] (2021).

[14] L. M. K. Vandersypen, H. Bluhm, J. S. Clarke, A. S. Dzurak, R. Ishihara, A. Morello, D. J. Reilly, L. R. Schreiber, and M. Veldhorst, Interfacing spin qubits in quantum dots and donors—hot, dense and coherent, Npj Quantum Inf. **3**, 34 (2017).

[15] D. P. Franke, J. S. Clarke, L. M. K. Vandersypen, and M. Veldhorst, Rent's rule and extensibility in quantum computing, Microprocess. Microsy. **67**, 1 (2019).

[16] Y. Alexeev *et al.*, Quantum Computer Systems for Scientific Discovery, PRX Quantum **2**, 017001 (2021).

[17] M. Veldhorst, H. G. J. Eenink, C. H. Yang, and A. S. Dzurak, Silicon CMOS architecture for a spin-based quantum computer, Nat. Commun. **8**, 1766 (2017).

[18] R. Li, L. Petit, D. P. Franke, J. P. Dehollain, J. Helsen, M. Steudtner, N. K. Thomas, Z. R. Yoscovits, K. J. Singh, S. Wehner, L. M. K. Vandersypen, J. S. Clarke, and M. Veldhorst, A crossbar network for silicon quantum dot qubits, Sci. Adv. **4**, eaar3960 (2018).

[19] B. Buonacorsi, Z. Cai, E. B. Ramirez, K. S. Willick, S. M. Walker, J. Li, B. D. Shaw, X. Xu, S. C. Benjamin, and J. Baugh, Network architecture for a topological quantum computer in silicon, Quantum Sci. Technol. **4**, 025003 (2019).

[20] J. M. Boter, J. P. Dehollain, J. P. G. van Dijk, T. Hensgens, R. Versluis, J. S. Clarke, M. Veldhorst, F. Sebastiano, and L. M. K. Vandersypen, in *2019 IEEE International Electron Devices Meeting (IEDM)* (IEEE, New York, 2019), p. 1.

[21] L. Petit, H. G. J. Eenink, M. Russ, W. I. L. Lawrie, N. W. Hendrickx, S. G. J. Philips, J. S. Clarke, L. M. K. Vandersypen, and M. Veldhorst, Universal quantum logic in hot silicon qubits, Nature **580**, 355 (2020).

[22] L. Petit, M. Russ, H. G. J. Eenink, W. I. L. Lawrie, J. S. Clarke, L. M. K. Vandersypen, and M. Veldhorst, High-fidelity two-qubit gates in silicon above one kelvin, arXiv:2007.09034 [cond-mat.mes-hall] (2020).

[23] C. H. Yang, R. C. C. Leon, J. C. C. Hwang, A. Saraiva, T. Tanttu, W. Huang, J. Camirand Lemyre, K. W. Chan, K. Y. Tan, F. E. Hudson, K. M. Itoh, A. Morello, M. Pioro-Ladrière, A. Laucht, and A. S. Dzurak, Operation of a silicon quantum processor unit cell above one kelvin, Nature **580**, 350 (2020).

[24] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, J. Math. Phys. **43**, 4452 (2002).

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

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

[27] N. W. Hendrickx, W. I. L. Lawrie, L. Petit, A. Sammak, G. Scappucci, and M. Veldhorst, A single-hole spin qubit, Nat. Commun. **11**, 3478 (2020).

[28] Y. Aharonov and J. Anandan, Phase Change during a Cyclic Quantum Evolution, Phys. Rev. Lett. **58**, 1593 (1987).

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

[30] I. L. Chuang and Michael A. Nielsen, *Quantum Computation and Quantum Information* (Cambridge University Pr., Cambridge, 2010).

[31] J. M. Elzerman, R. Hanson, L. H. Willems van Beveren, B. Witkamp, L. M. K. Vandersypen, and L. P. Kouwenhoven, Single-shot read-out of an individual electron spin in a quantum dot, Nature **430**, 431 (2004).

[32] A. E. Seedhouse, T. Tanttu, R. C. C. Leon, R. Zhao, K. Y. Tan, B. Hensen, F. E. Hudson, K. M. Itoh, J. Yoneda, C. H. Yang, A. Morello, A. Laucht, S. N. Coppersmith, A. Saraiva, and A. S. Dzurak, Pauli Blockade in Silicon Quantum Dots with Spin-Orbit Control, PRX Quantum **2**, 010303 (2021).

[33] A. M. Jones, E. J. Pritchett, E. H. Chen, T. E. Keating, R. W. Andrews, J. Z. Blumoff, L. A. De Lorenzo, K. Eng, S. D. Ha, A. A. Kiselev, S. M. Meenehan, S. T. Merkel, J. A. Wright, L. F. Edge, R. S. Ross, M. T. Rakher, M. G. Borselli, and A. Hunter, Spin-Blockade Spectroscopy of Si/Si-Ge Quantum Dots, Phys. Rev. Appl. **12**, 014026 (2019).

[34] E. J. Connors, J. J. Nelson, and J. M. Nichol, Rapid High-Fidelity Spin-State Readout in Si/Si-Ge Quantum Dots via rf Reflectometry, Phys. Rev. Appl. **13**, 024019 (2020).

[35] R. Versluis, S. Poletto, N. Khammassi, B. Tarasinski, N. Haider, D. J. Michalak, A. Bruno, K. Bertels, L. DiCarlo, and A. Bruno, Scalable Quantum Circuit and Control for a Superconducting Surface Code, Phys. Rev. Appl. **8**, 034021 (2017).

[36] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A **86**, 032324 (2012).

[37] C. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, Surface code quantum computing by lattice surgery, New J. Phys. **14**, 123011 (2012).

[38] Y. Xu, F. K. Unseld, A. Corna, A. M. J. Zwerver, A. Sammak, D. Brousse, N. Samkharadze, S. V. Amitonov, M. Veldhorst, G. Scappucci, R. Ishihara, and L. M. K. Vandersypen, On-chip integration of Si/SiGe-based quantum dots and switched-capacitor circuits, Appl. Phys. Lett. **117**, 144002 (2020).

[39] R. J. Schoelkopf, P. Wahlgren, A. A. Kozhevnikov, P. Delsing, and D. E. Prober, The radio-frequency single-electron transistor (RF-SET): A fast and ultrasensitive electrometer, Science **280**, 1238 (1998).

[40] S. J. Devitt, A. G. Fowler, A. M. Stephens, A. D. Greentree, L. C. L. Hollenberg, W. J. Munro, and K. Nemoto, Architectural design for a topological cluster state quantum computer, New J. Phys. **11**, 083032 (2009).

[41] R. K. Puddy, L. W. Smith, H. Al-Taie, C. H. Chong, I. Farrer, J. P. Griffiths, D. A. Ritchie, M. J. Kelly, M. Pepper, and C. G. Smith, Multiplexed charge-locking device for large arrays of quantum devices, Appl. Phys. Lett. **107**, 143501 (2015).

[42] S. Y. Hou, C. H. Wu, D. Yu, H. Hsia, C. H. Tsai, K. C. Ting, T. H. Yu, Y. W. Lee, F. C. Chen, W. C. Chiou, and C. T. Wang, in *2019 IEEE International Electron Devices Meeting (IEDM)* (IEEE, San Francisco, 2019).

[43] International Roadmap for Devices and Systems (IRDS$^{\mathrm{TM}}$) 2020 Edition - More Moore (2020).

[44] B. Buonacorsi, B. Shaw, and J. Baugh, Simulated coherent electron shuttling in silicon quantum dots, Phys. Rev. B **102**, 125406 (2020).

[45] R. C. C. Leon, C. H. Yang, J. C. C. Hwang, J. C. Lemyre, T. Tanttu, W. Huang, K. W. Chan, K. Y. Tan, F. E. Hudson, K. M. Itoh, A. Morello, A. Laucht, M. Pioro-Ladrière, A. Saraiva, and A. S. Dzurak, Coherent spin control of s-, p-, d- and f-electrons in a silicon quantum dot, Nat. Commun. **11**, 797 (2020).

[46] J. P. G. Van Dijk *et al.*, A scalable cryo-CMOS controller for the wideband frequency-multiplexed control of spin qubits and transmons, IEEE J. Solid-State Circuits **55**, 2930 (2020).

[47] J. Van Dijk, P. Hart, G. Kiene, R. Overwater, P. Padalia, J. Van Staveren, M. Babaie, A. Vladimirescu, E. Charbon, and F. Sebastiano, in *Proceedings of the Custom Integrated Circuits Conference* (IEEE, Boston, 2020), p. 1.

[48] C. R. Paul, *Analysis of Multiconductor Transmission Lines* (Wiley-IEEE Press, Hoboken, 2008).

[49] S.-C. Wong, P. S. Liu, J.-W. Ru, and S.-T. Lin, Interconnection capacitance models for VLSI circuits, Solid-State Electron. **42**, 969 (1998).

[50] B. Razavi, *Design of Analog CMOS Integrated Circuits* (McGraw-Hill Education, New York, 2016), 2nd ed.

[51] B. Patra, M. Mehrpoo, A. Ruffino, F. Sebastiano, E. Charbon, and M. Babaie, Characterization and analysis of on-chip microwave passive components at cryogenic temperatures, IEEE J. Electron Devices Soc. **8**, 448 (2020).

[52] N. Schuch and J. Siewert, Natural two-qubit gate for quantum computation using the XY interaction, Phys. Rev. A **67**, 032301 (2003).