Unlocking Flexible Silicon Dangling Bond Logic Designs on Alternative Silicon Orientations

Samuel S. H. Ng, Jan Drewniok, Marcel Walter, Jacob Retallick, Robert Wille, and Konrad Walus

***Abstract*— With the impending plateau of Moore's Law, the search for novel computational paradigms has intensified. Silicon dangling bond (SiDB) logic emerges as a promising avenue in this quest, leveraging the quantum-dot-like properties of SiDBs and atomically precise fabrication techniques to realize logic functions at the nanometer scale. Advances in computer-aided design (CAD) tools specialized for SiDB logic exploration have also opened the door to novel logic research from the gate- to application-level. This paper introduces a lattice vector formulation for SiDB logic designs on alternative silicon lattice orientations, enabling the exploration of logic gates on arbitrary lattice orientations and addressing the limitations of previous SiDB logic research confined to the H-Si(100)-2×1 surface. A comprehensive workflow for designing standard tile libraries compatible with design automation frameworks is proposed, facilitating the scaling of SiDB layouts to large-scale systems implementation on multiple lattice orientations. We demonstrate the proposed lattice vector representation and the library design workflow through a case study on the H-Si(111)-1×1 surface, showcasing the first logic gates designed for this orientation. This advancement opens new avenues for SiDB logic research, enabling rigorous evaluations of various lattice orientations for future logic design studies and experimental investigations.**

I. Introduction

As the scaling of complementary metal-oxide-semiconductor (CMOS) technology approaches its physical limits, there is a growing need to explore alternative modes of computation that can overcome the challenges of power dissipation and integration density. Field-coupled nanocomputing (FCN) presents an alternative computing paradigm which represents bit states and performs computation via field-effects [1]. Recent advances in the atomically-precise fabrication of silicon dangling bonds (SiDBs) [2], [3] have led to experimental demonstrations of nanoscale logic devices made of SiDBs on the hydrogen-passivated silicon (100) 2×1 surface (H-Si(100)-2×1) [4]–[6], promising high frequency [7] and low-powered operation [8], [9]. Experimental demonstrations have shown the versatility of SiDBs in implementing multiple types of logic unit cells [5], [6]. Most notably, Huff *et al.* have demonstrated a logic OR gate which spans merely $5 \times 6\,\mathrm{nm}^2$ [6], showing tremendous promise for this platform.

This work was supported by the Natural Sciences and Engineering Research Council of Canada under Grant RGPIN-2022-04830.

S. S. H. Ng, J. Retallick, and K. Walus were with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada (e-mail: samueln@ece.ubc.ca; jret@ece.ubc.ca; konradw@ece.ubc.ca); J. Drewniok, M. Walter, and R. Wille were with the Chair for Design Automation, Technical University of Munich, BY, Germany (e-mail: jan.drewniok@tum.de, marcel.walter@tum.de, robert.wille@tum.de). R. Wille was also with the Software Competence Center Hagenberg (SCCH) GmbH, Hagenberg, OÖ, Austria.

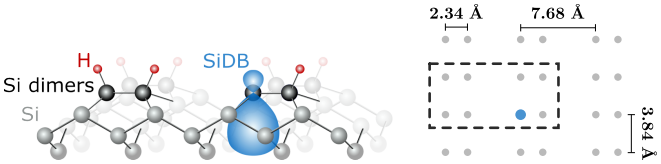
Further fueling research interests, the introduction of computer-aided design (CAD) tools specialized in SiDB design and simulation has enabled rapid prototyping of SiDB logic devices without expensive experimental equipment. SiDB layouts can be designed in *SiQAD* [10] and simulated using specialized physics models [8], [10]–[12]. Automated circuit designers have also emerged to speed up the design of logic gates [13], [14], which have in turn enabled the creation of standard SiDB gate tiles and allowed the *fiction* framework to introduce support for the placement and routing of SiDB layouts [15], [16]. The SiDB logic platform has also shown promise in analog-to-digital conversion [9] and machine learning acceleration [17].

Despite the significant advancements in SiDB technology on the H-Si(100)-2×1 surface, other lattice orientations remain largely unexplored—a gap this work aims to bridge. This study's key contributions are threefold: we propose a lattice representation suitable for generic lattice orientations, establish a universal workflow for designing SiDB logic tiles across various H-Si surfaces, and present a detailed case study on the H-Si(111)-1×1 orientation using the proposed workflow. In Section II, we provide a comprehensive background on SiDB logic design and available CAD platforms. In Section III, we introduce a formulation for defining arbitrary periodic lattice orientations that is portable across multiple open source CAD frameworks, then propose a workflow that helps streamline the design of standard tile libraries on arbitrary lattice orientations. In Section IV, we first explore the merits of H-Si(111)-1×1, an alternative lattice orientation we found to be suitable for logic design, then detail our software implementation of multi-lattice orientation support in prominent open source CAD tools, and lastly design a full suite of H-Si(111)-1×1 standard tiles as a case study; it represents the first logic gates designed for this lattice orientation and forms the basis for design automation frameworks to create large-scale SiDB circuitry on this orientation. Section V concludes the manuscript.

II. Background

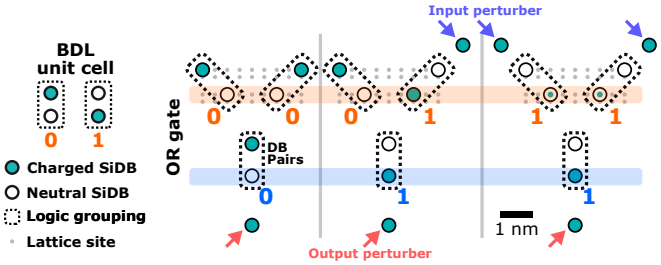
In this section, we provide background information on the use of SiDBs as quantum dots and how they form the basis for the creation of logic devices. Then we review existing CAD capabilities and design automation frameworks that facilitate gate- to system-level SiDB logic design.

The precise fabrication of SiDBs has been extensively demonstrated on the H-Si(100)-2×1 surface [2], [4]. They can be created by the removal of a single hydrogen atom via a scanning probe which leaves behind an unsatisfied



(a) Surface of H-Si(100)-2×1 with an SiDB.

(b) Top view.



(c) BDL unit cell and OR gate from [6].

Fig. 1. (a) Side view of the H-Si(100)-2×1 surface featuring a single SiDB. (b) Top view of the H-Si(100)-2×1 surface with H-passivated lattice sites depicted in gray and a bare SiDB in blue. (c) A BDL unit cell and a BDL OR gate. The unit cell is composed of a pair of SiDBs, capable of representing logic bit state. The OR gate is recreated from [6], composed of SiDB-pairs at each input and output location. Shaded boxes highlight which SiDB to inspect to interpret the logic state of that pin. Reprinted with permission: (a) and (b) from [10], (c) from [15].

dangling bond [2]; they can also be erased by repassivating the dangling bond with a hydrogen atom [3]. The hydrogen atoms are positioned spatially periodically with their locations defined by the lattice orientation, as illustrated in Fig. 1a-1b. These SiDBs have been demonstrated to be capable of holding 0, 1, or 2 electrons, corresponding to positive, neutral, and negative charge states [18]. Given an n-doped silicon bulk, isolated SiDBs tend to take on negative charge states, but pairs of SiDB in close proximity would share a negative charge [5]. This behavior enabled the binary-dot logic (BDL) representation, whereby logic bits are encoded in the position of negative charges shared among pairs of SiDBs [6] as illustrated in Fig. 1c. Using this logic representation, an OR gate at the size of $5 \times 6\,\mathrm{nm}^2$ has been experimentally demonstrated as shown in Fig. 1c [6], ushering in an era of nanoscale logic devices that can be replicated with atomic precision. Input/output (I/O) circuitry is expected to be constructed of biasing electrodes for input [8], [9] and a combination of atomic wires made of contiguous chains of SiDBs [19], [20] and surface contact pads [21] for readout.

The introduction of CAD tools has further propelled the ease and scope of SiDB logic exploration beyond what is currently possible in experiments that require costly equipment and highly specialized professionals. At the physical level of designing logic gates by the assembly of SiDBs, *SiQAD* and specialized physical simulators [8], [10]–[12] have enabled the rapid design and simulation of SiDB gates and circuits with multiple demonstrated topologies for wire and gate arrangement [10], [22]. Higher-level abstraction and design automation called for the proposal of standard gate tiles that specified the input and output wire locations, leaving

TABLE I

LATTICE AND BASIS VECTORS OF MULTIPLE SILICON ORIENTATIONS (UNIT: Å)

	H-Si(100)-2×1		H-Si(111)-1×1 Monoclinic		H-Si(111)-1×1 Orthorhombic	
	$\hat{x}$	$\hat{y}$	$\hat{x}$	$\hat{y}$	$\hat{x}$	$\hat{y}$
$\vec{a}_1$	3.84	0.00	3.3255	1.92	6.65	0.00
$\vec{a}_2$	0.00	7.68	0.00	3.84	0.00	3.84
$\vec{b}_1$	0.00	0.00	0.00	0.00	0.00	0.00
$\vec{b}_2$	0.00	0.25	–	–	3.3255	1.92

a design canvas in the center of the tile where SiDBs can be placed to implement logic functions. The task of placing SiDBs in the canvas can be performed automatically [13], [14] or manually. These innovations have culminated in the *Bestagon* standard tile library which has been successfully used to implement a placement and routing workflow via the *fiction* framework [15], [16].

As mentioned above, all of these achievements through experimental and computational studies of SiDB logic have been focused on the H-Si(100)-2×1 surface. Other lattice orientations are left unexplored, leaving open questions about potential improvements in logic design that are left untapped. In the next section, we propose methodologies for representing arbitrary lattice orientations suitable for latest SiDB CAD tools and workflows for designing circuits from the quantum dot assembly level to the synthesized logic system level.

III. Methodology for H-Si Logic Exploration on Arbitrary Lattice Orientations

In this section, we outline a methodology for exploring H-Si logic on arbitrary silicon lattice orientations. We start with establishing a mathematical representation of the crystalline structure by defining tileable lattice unit cells in Section III-A. We then propose a workflow for creating new SiDB gate libraries from scratch on arbitrary lattice orientations in Section III-B, which can subsequently be used in design automation frameworks.

A. Lattice Representation

We will first establish a mathematical representation of silicon crystalline structures that is portable across CAD frameworks and simulators. Knowing that different silicon lattice orientations and reconstructions are spatially periodic, we can define tileable lattice unit cells, each consisting of a number of discretely defined SiDB locations, to represent arbitrary lattice orientations and reconstructions. The top layer silicon lattice structure can be defined as follows: the 2D-tiling unit cell can be defined with two real-valued vectors, a.k.a. the lattice vectors, $\vec{a}_1$ and $\vec{a}_2$; an array of basis vectors, $\vec{b}_i$, can then be defined to denote the positions of silicon atoms in the unit cell relative to the origin of the cell. The top silicon layer is thus defined by tiling these unit cells on a 2D plane anchored by multiples of $\vec{a}_1$ and $\vec{a}_2$.

A lattice coordinate system, $(n, m, l) \in \mathbb{Z}^2 \times \mathbb{N}$ with l bounded by the count of silicon atoms in the unit cell, can

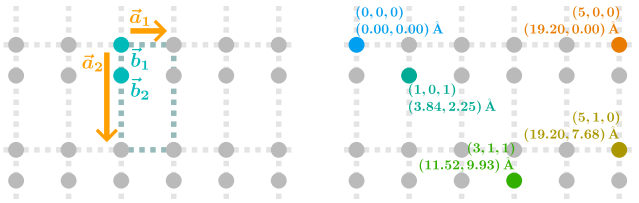


Fig. 2. Example of a lattice vector representation on the H-Si(100)-2×1 surface. (a) An illustration of the tiled surface. Lattice tiles are delimited by gray dotted lines, a single tileable unit cell is highlighted with the tiling separation defined by $\vec{a}_1$ and $\vec{a}_2$ and the SiDB locations within the unit cell defined by $\vec{b}_1$ and $\vec{b}_2$. (b) Several examples of SiDBs locations represented in lattice coordinates as defined in Eq. (1).

then be defined to locate specific silicon atoms on a Cartesian plane at location $\vec{d}$ using the following expression:

$$\vec{d} = n\vec{a}_1 + m\vec{a}_2 + \vec{b}_{l+1} \tag{1}$$

Here, n and m are scalar multipliers of the lattice vectors $\vec{a}_1$ and $\vec{a}_2$ respectively; l is an index for $\vec{b}_i$. Since l is 0-indexed in the coordinate system while $\vec{b}_i$ is 1-indexed, a $+1$ offset is added for conversion.

In Table I, we include lattice vector definitions for the H-Si(100)-2×1 surface which is established in previous works, as well as multiple definitions for H-Si(111)-1×1, including monoclinic and orthorhombic, which we will discuss in Section IV. The H-Si(100)-2×1 surface, composed of unit cells defined by these lattice vectors, is illustrated in Fig. 2a. Graphical examples of the use of lattice coordinates to address exact surface lattice sites, as well as their corresponding Cartesian coordinates, are also provided in Fig. 2b.

B. Physical Design on Generic H-Si Lattices

Although there already exist SiDB gate libraries, some with standardized pin locations such as the *Bestagon* Gate Library [15] and some without [22], there has not yet been a formalized workflow for the creation of new SiDB gate libraries from scratch. In the following, we propose such a workflow, with the expectation that all gates in the library will share standardized pin locations to simplify physical design. Each step in the workflow, from step 1 to 4, corresponds to a labeled figure in Fig. 3.

1) **Decide on the shape of the standard tile:** each tile consists of I/O pins as well as a blank canvas in the center for logic implementation. Current design automation frameworks support rectangular and hexagonal tiles [15], [16], but triangular tiles would also form a uniform grid if support is added to corresponding frameworks. Other shapes will require compound tiling, potentially causing additional physical design overhead.
2) **Define standardized pin locations:** this forms the standard tile template and allows gates to be placed and routed conveniently with design automation frameworks. Current understanding suggests that the pin wires should ideally be long enough to allow screening effects to attenuate electrostatic influences [15]. At the extremities of the pins, peripheral SiDBs—which exist to emulate the existence of input and output wires extending beyond the gate—are referred to as *perturbers* [6]. Input perturbers can be moved closer to the layout to emulate a logic 1, and further to emulate a logic 0 [15].
3) **Define and implement a list of logic functions that form a universal gate set:** the Boolean function of each SiDB gate can be implemented by iteratively populating the canvas in the standard tile template and evaluating the logic correctness of all input combinations via ground-state simulations [10]–[12]. There exist automatic circuit designers that significantly streamline this phase [13], [14], but manual design is also an option, if necessary. If possible, design multiple variants of each Boolean function to accumulate candidates for the library, and evaluate their logic stability metrics [23]–[25].
4) **If required, make modifications to the standard tile template:** upon encountering difficulties in implementing a universal gate set using the standard tile template, modifications to the template may help unblock the issues. Consider adjusting the pin locations and lengths, canvas dimension, and permitted SiDB count in the canvas.

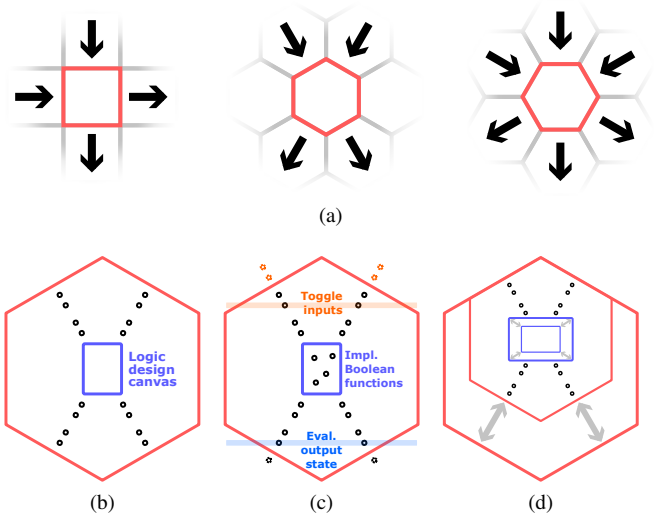


Fig. 3. (a) Illustration of a few possible shapes that a standard tile template can take on. (b) Example of a standard tile template with 2 input pins and 2 output pins. A logic design canvas is included in the center of the template where SiDBs can be added to implement Boolean functions. (c) A standard tile template with SiDBs added to the canvas. All input combinations should be tested by toggling the input perturbers and the logic correctness evaluated by the output SiDB-pair. (d) If necessary, the template and the canvas should be adjusted to account for physical effects specific to the targeted surface.

Upon acquiring a universal set of logic gates, they form the basis for design automation algorithms to synthesize netlists into large-scale SiDB layouts. Although this is by no means the only possible workflow for the creation of new H-Si gate libraries, we believe that designers can save time and maximize compatibility with existing CAD tools specialized for SiDB design [10], [16] if they follow the proposed flow. A gate library need not be restricted to only

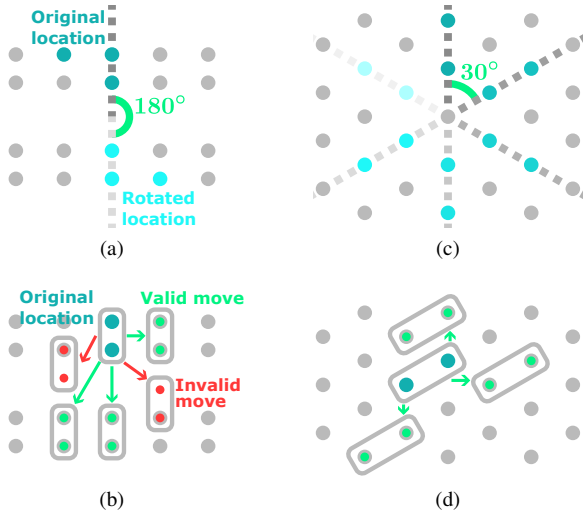


Fig. 4. Illustrations of geometric symmetries for (a-b) H-Si(100)-2×1 and (c-d) H-Si(111)-1×1. (a) and (c) demonstrate rotational symmetries where it is 2-fold for H-Si(100)-2×1 and 6-fold for H-Si(111)-1×1. (b) and (d) demonstrate the translational symmetry where SiDBs in solid full are translated to some example destinations, valid destinations are green-filled while invalid destinations are red-filled; H-Si(100)-2×1 displays limited translational symmetry while H-Si(111)-1×1 allows assemblies of SiDBs to be translated to any destination where lattice sites exist.

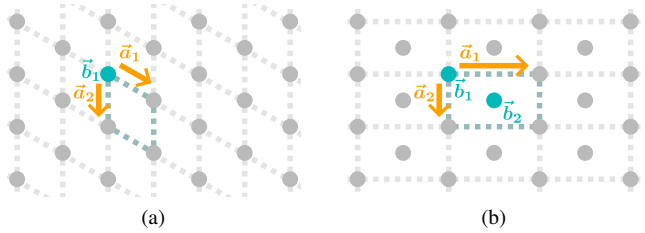


Fig. 5. Lattice vector representation of H-Si(111)-1×1 in (a) monoclinic representation where each unit cell contains 1 SiDB and in (b) orthorhombic representation where each unit cell contains 2 SiDBs.

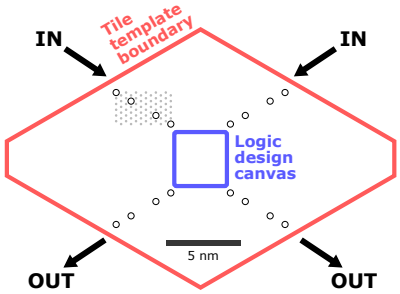


Fig. 6. A standard tile template designed for the H-Si(111)-1×1 surface with ground state simulation parameters in line with previous logic studies [15]: $\epsilon_r = 5.6$, $\lambda_{TF} = 5\,\text{nm}$, $\mu_- = -0.32\,\text{eV}$.

contain one gate candidate for each implemented Boolean function. The existence of multiple gate candidates, each possibly optimized for different logic robustness metrics, allows users to select the best-performing gate optimized for the intended operating environment.

IV. Case Study of Logic Design on the H-Si(111)-1×1 Orientation

There exists a wide selection of H-Si lattice orientations for the research community to explore. As a case study of physical logic design on alternative lattice orientations, we choose H-Si(111)-1×1 due to several geometrical advantages over H-Si(100)-2×1 as discussed in Section IV-A. In Section IV-B, we detail software implementations for arbitrary lattice support in *SiQAD* and *fiction* as well as how they can be used together for the implementation of standard tile libraries. We then finish this section with Section IV-C, where we propose a full suite of standard library tiles implementing a range of 2-in-1-out and 2-in-2-out functions on the H-Si(111)-1×1 surface which form the basis for higher-level design automation exploration.

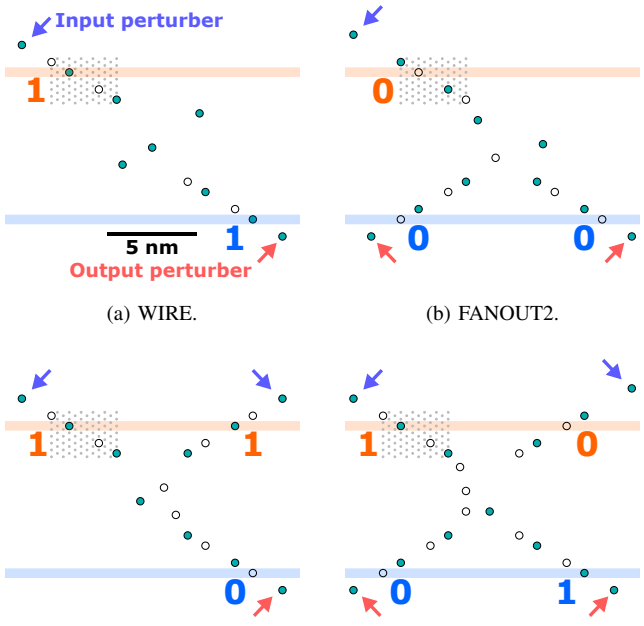
A. Selection of Lattice Orientations

Out of the available H-Si lattice orientations, we are motivated to select one that has capabilities for logic implementation and subsequent system integration superior to those of H-Si(100)-2×1. It can be observed that SiDB layouts on the H-Si(100)-2×1 surface are restricted to 2-fold rotational symmetry (i.e., 180° rotations), as illustrated in Fig. 4a, and possess limited discrete translational symmetry where they can be discretely translated to some, but not all, discrete lattice locations, as shown in Fig. 4b. On the other hand, the H-Si(111)-1×1 surface exhibits 6-fold rotational symmetry (i.e., 60° rotations) and supports the translation of SiDB layouts to any available discrete sites as shown in Fig. 4c–4d. The improved geometric symmetry of H-Si(111)-1×1 makes it an attractive lattice orientation for logic designers to investigate, and we further postulate that the symmetry offers potential advantages in manufacturing defect mitigation (e.g., offsetting entire gates or circuits for defect avoidance).

Looking beyond logic implementation, the H-Si(111)-1×1 surface also offers unique advantages over the H-Si(100)-2×1 surface for the creation of atomic wires made of contiguous chains of SiDBs [19]. On the H-Si(100)-2×1 surface, the inter-SiDB spacing differs in the directions along and across the dimer rows (i.e., the $\pm\vec{a}_1$ and $\pm\vec{a}_2$ directions for H-Si(100)-2×1 from Table I), causing the transport properties to also differ [26]. This poses additional difficulties when designing atomic wires that run in both directions on the 2D surface. On the other hand, the H-Si(111)-1×1 surface can accommodate uniform atomic wires in all six directions using the lattice vectors for H-Si(111)-1×1 Monoclinic from Table I: $\pm\vec{a}_1$, $\pm\vec{a}_2$, and $\pm(\vec{a}_1 - \vec{a}_2)$. Although this work will not directly investigate the incorporation of atomic wires in SiDB logic design, this is an important advantage to keep in mind for future work that considers I/O circuitry for interfacing SiDBs and CMOS, or other novel applications of atomic wires on the H-Si surface.

B. Implementation in CAD

To enable computer-aided exploration of alternative lattice structures, we have modified the coordinate systems of *SiQAD* and *fiction* to support multiple lattice tiling definitions. SiDB layouts are thus defined by the lattice coordinates of each SiDB as well as lattice vectors that define the



(a) WIRE. (b) FANOUT2.

(c) NAND. (d) CROSSOVER.

Fig. 7. Select logic gates designed on the H-Si(111)-1×1 surface, including (a) WIRE, (b) FANOUT2, (c) NAND, and (d) CROSSING. The full set of gates is available at [27], all of which designed to operate at simulation parameters: $\mu_- = -0.32\,\text{eV}$, $\epsilon_r = 5.6$, and $\lambda_{\text{TF}} = 5\,\text{nm}$.

crystalline structure of the top-layer silicon atoms, allowing the recreation of the entire SiDB layout. The Cartesian coordinates of any SiDB on arbitrary lattices defined can thus be computed by Eq. (1) by *SiQAD* and simulators alike. This ensures the portability of simulation results and enables support for other arbitrary lattice structures that are of interest to the research community in the future. Ground-state simulators that come with *SiQAD* and *fiction* have been updated to support multiple lattice vector definitions.

For this case study, we have defined the following lattice vectors in *SiQAD* and *fiction*: H-Si(100)-2×1 due to its status as an established surface orientation of choice, and H-Si(111)-1×1 due to its desirable properties for logic design as detailed in Section IV-A. The lattice vectors can be found in Table I. Notice that two variants of lattice vectors are provided for H-Si(111)-1×1: monoclinic, which defines the surface with a rhomboidal unit cell containing just one SiDB, making it the primitive unit cell; and orthorhombic, which defines the surface with a rectangular unit cell with two SiDBs. Although the monoclinic representation constitutes the primitive unit cell, there exists rendering optimizations in *SiQAD* that only work with orthorhombic unit cells. We therefore employ the orthorhombic definition in this work.

With the above adjustments made to the CAD tools, we now have the tools needed to attempt to create a standard tile library on H-Si(111)-1×1 as a case study.

C. Proposal of a Standard Tile Library for H-Si(111)-1×1

Following the standard tile library design workflow proposed in Section III-B, we present an original standard tile library for H-Si(111)-1×1 which constitutes the first SiDB standard tile library for an alternative surface orientation and serves as a demonstration of the proposed workflow. Below, we outline the decisions made for each step of the workflow, and the tools employed to accomplish them:

1) **Decide on the shape of the standard tile:** for this case study, we have chosen a pointed-top hexagonal shape just like the *Bestagon* gate library [15], albeit with different dimensions. The reason that we have chosen a pointed-top hexagon rather than the flat-top counterpart is due to clocking considerations where we would prefer a full row of tiles to be activated simultaneously in row-wise clocking.
2) **Define standardized pin locations:** we have created multiple standard tile template prototypes using *SiQAD* [10], taking care to align each of the wire pins to one of the axes labeled in Fig. 4c. The resulting standard tile template is included in Fig. 6.
3) **Define and implement a list of logic gates that form a universal gate set:** all 2-in-1-out Boolean functions have been successfully implemented, along with a number of functions with other pin counts, including 1-in-1-out: wires and inverters; 1-in-2-out: fanouts; and 2-in-2-out: crossing and double-wire. They were implemented using the *Automatic Exhaustive Gate Designer* [13].
4) **If required, make modifications to the standard tile template:** leading up to the proposed template, we have made numerous adjustments to the wire spacing and canvas size in response to initial failures to implement a universal gate set, as a template that is too dense or sparse can lead to under- or over-population of surface charges which hinder logic representation.

Select gates from the thusly proposed standard tile library designed with the above flow are illustrated in Fig. 7. All of the created gate tiles are publicly available at [27]. This universal gate set can then be used as the basis for design automation frameworks, such as *fiction* [16], to synthesize large-scale SiDB logic circuits.

Due to the absence of specifically fitted screened Coulomb parameters for the H-Si(111)-1×1 surface, this study has resorted to utilizing parameters experimentally fitted for the H-Si(100)-2×1 surface [6], which have also been employed previously in SiDB logic studies [10], [15]. Despite the potential misalignment of these parameters with the precise conditions of the H-Si(111)-1×1 surface, we postulate that the employed values provide a reasonable approximation, given the fundamental similarity in the bulk properties shared by these surfaces. Moreover, there exists documented variability in parameters even amongst distinct H-Si(100)-2×1 surfaces [6], [28], further justifying the need for approximations for case studies. The proposed methodology still stands nonetheless, in the sense that standard tile libraries can be adapted to different sets of physical parameters by modifying the standard tile template and reimplementing the canvas body by automated designers.

V. Conclusion

In the pursuit of advancing the field of SiDB logic studies, this paper successfully introduces a lattice vector formulation that enables the exploration of arbitrary H-Si lattice orientations beyond the established H-Si(100)-2×1 orientation. Observing a lack of guidelines for the creation of logic tiles on arbitrary lattice orientations, we have also proposed a workflow that designers can follow to design standard tile libraries that can be employed by design automation frameworks to create large-scale SiDB logic systems. To allow the research community to explore alternative lattice orientations, we have added support for multiple lattice orientations to prominent open-sourced tools for SiDB logic research: *SiQAD* [10] and *fiction* [16]; the former facilitates the design and simulation of SiDB layouts, and the latter a design automation framework which have built-in support for SiDBs. To serve as a case study for logic design on alternative lattice orientations, we have chosen the H-Si(111)-1×1 surface due to identified superior symmetric properties for the implementation of logic components and atomic wires, then employed the proposed standard tile library design workflow to create the first H-Si(111)-1×1 standard tile library. Using the Automatic Exhaustive Gate Designer [13], we have designed a universal gate set with all 2-in-1-out Boolean functions as well as a number of other functions with different pin counts. This successful case study demonstrates a promising path towards the exploration of alternative H-Si lattice orientations, thus allowing future work to rigorously compare logic systems designed for different lattice orientations based on geometrical symmetries, robustness metrics [23]–[25], and other physical properties pertaining to each surface. This ultimately informs logic designers and experimental labs alike on which lattice orientations to focus future research efforts on.

References

[1] N. G. Anderson and S. Bhanja, Eds., *Field-Coupled Nanocomputing: Paradigms, Progress, and Perspectives*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, vol. 8280. [Online]. Available: https://link.springer.com/10.1007/978-3-662-43722-3

[2] R. Achal *et al.*, "Lithography for robust and editable atomic-scale silicon devices and memories," vol. 9, no. 1, p. 2778. [Online]. Available: https://doi.org/10.1038/s41467-018-05171-y

[3] T. R. Huff *et al.*, "Atomic white-out: Enabling atomic circuitry through mechanically induced bonding of single hydrogen atoms to a silicon surface," vol. 11, no. 9, pp. 8636–8642. [Online]. Available: https://doi.org/10.1021/acsnano.7b04238

[4] J. Pitters, J. Croshaw, R. Achal, L. Livadaru, S. Ng, R. Lupoiu, T. Chutora, T. Huff, K. Walus, and R. A. Wolkow, "Atomically precise manufacturing of silicon electronics," *ACS Nano*, vol. 18, no. 9, pp. 6766–6816, 2024, pMID: 38376086.

[5] M. B. Haider, J. L. Pitters, G. A. DiLabio, L. Livadaru, J. Y. Mutus, and R. A. Wolkow, "Controlled coupling and occupation of silicon atomic quantum dots at room temperature," vol. 102, no. 4, p. 046805.

[6] T. Huff *et al.*, "Binary atomic silicon logic," vol. 1, no. 12, pp. 636–643. [Online]. Available: https://doi.org/10.1038/s41928-018-0180-3

[7] L. Livadaru *et al.*, "Dangling-bond charge qubit on a silicon surface," vol. 12, no. 8, p. 083018. [Online]. Available: https://doi.org/10.1088/1367-2630/12/8/083018

[8] H. N. Chiu, S. S. H. Ng, J. Retallick, and K. Walus, "PoisSolver: A tool for modelling silicon dangling bond clocking networks," in *2020 IEEE 20th International Conference on Nanotechnology (IEEE NANO)*. IEEE, pp. 134–139. [Online]. Available: https://ieeexplore.ieee.org/document/9183710/

[9] H. N. Chiu, "Simulation and analysis of clocking and control for field-coupled quantum-dot nanostructures." [Online]. Available: https://open.library.ubc.ca/cIRcle/collections/ubctheses/24/items/1.0389918

[10] S. S. H. Ng *et al.*, "SiQAD: A design and simulation tool for atomic silicon quantum dot circuits," vol. 19, pp. 137–146.

[11] J. Drewniok, M. Walter, S. S. H. Ng, K. Walus, and R. Wille, "*QuickSim*: Efficient *and* Accurate Physical Simulation of Silicon Dangling Bond Logic," in *IEEE International Conference on Nanotechnology (IEEE NANO)*, pp. 817–822. [Online]. Available: https://ieeexplore.ieee.org/document/10231266

[12] J. Drewniok, M. Walter, and R. Wille, "The need for speed: Efficient exact simulation of silicon dangling bond logic," in *2024 29th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2024, pp. 576–581.

[13] ——, "Minimal design of sidb gates: An optimal basis for circuits based on silicon dangling bonds," in *Proceedings of the 18th ACM International Symposium on Nanoscale Architectures*, ser. NANOARCH '23. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3611315.3633241

[14] R. Lupoiu, S. S. H. Ng, J. A. Fan, and K. Walus. Automated Atomic Silicon Quantum Dot Circuit Design via Deep Reinforcement Learning. [Online]. Available: http://arxiv.org/abs/2204.06288

[15] M. Walter, S. S. H. Ng, K. Walus, and R. Wille, "Hexagons are the Bestagons: Design automation for silicon dangling bond logic," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, ser. DAC '22. Association for Computing Machinery, p. 6.

[16] M. Walter, R. Wille, F. Sill, D. Große, and R. Drechsler, "Fiction: An open source framework for the design of field-coupled nanocomputing circuits," vol. abs/1905.02477.

[17] S. S. H. Ng, H. N. Chiu, J. Retallick, and K. Walus, "A Blueprint for Machine Learning Accelerators Using Silicon Dangling Bonds," in *2023 IEEE 23rd International Conference on Nanotechnology (IEEE NANO)*. IEEE, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/10231158/

[18] M. Rashidi *et al.*, "Resolving and Tuning Carrier Capture Rates at a Single Silicon Atom Gap State," vol. 11, no. 11, pp. 11 732–11 738. [Online]. Available: https://pubs.acs.org/doi/10.1021/acsnano.7b07068

[19] S. Bohloul, Q. Shi, R. A. Wolkow, and H. Guo, "Quantum transport in gated dangling-bond atomic wires," vol. 17, no. 1, pp. 322–327. [Online]. Available: https://pubs.acs.org/doi/10.1021/acs.nanolett.6b04125

[20] M. Kepenekian, R. Robles, C. Joachim, and N. Lorente, "Surface-state engineering for interconnects on H-passivated Si(100)," vol. 13, no. 3, pp. 1192–1195.

[21] J. L. Pitters, I. A. Dogel, and R. A. Wolkow, "Charge control of surface dangling bonds using nanoscale Schottky contacts," vol. 5, no. 3, pp. 1984–1989. [Online]. Available: https://doi.org/10.1021/nn103042m

[22] M. D. Vieira *et al.*, "Three-Input NPN Class Gate Library for Atomic Silicon Quantum Dots," pp. 1–1.

[23] M. Walter, J. Drewniok, S. S. H. Ng, K. Walus, and R. Wille, "Reducing the complexity of operational domain computation in silicon dangling bond logic," in *Proceedings of the 18th ACM International Symposium on Nanoscale Architectures*, ser. NANOARCH '23. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3611315.3633246

[24] J. Drewniok, M. Walter, and R. Wille, "Temperature behavior of silicon dangling bond logic," in *2023 IEEE 23rd International Conference on Nanotechnology (NANO)*, 2023, pp. 925–930.

[25] S. S. H. Ng, J. Croshaw, M. Walter, R. Wille, R. Wolkow, and K. Walus. Simulating Charged Defects in Silicon Dangling Bond Logic Systems to Evaluate Logic Robustness. [Online]. Available: http://arxiv.org/abs/2211.08698

[26] S. Watanabe, Y. A. Ono, T. Hashizume, and Y. Wada, "Theoretical study of atomic and electronic structures of atomic wires on an H-terminated Si(100)2x1 surface," vol. 54, no. 24, pp. R17 308–R17 311.

[27] S. S. H. Ng, J. Drewniok, M. Walter, J. Retallick, R. Wille, and K. Walus. Supplementary material for paper: Unlocking flexible silicon dangling bond logic designs on alternative silicon orientations. [Online]. Available: https://github.com/samuelngsh/si-111-paper-supplementary

[28] T. R. Huff *et al.*, "Electrostatic landscape of a Hydrogen-terminated silicon surface probed by a moveable quantum dot," vol. 13, no. 9, pp. 10 566–10 575. [Online]. Available: https://doi.org/10.1021/acsnano.9b04653