Hexagons are the Bestagons:
Design Automation for Silicon Dangling Bond Logic

Marcel Walter  
Technical University of Munich  
Chair for Design Automation  
Munich, Bavaria, Germany  
marcel.walter@tum.de

Konrad Walus  
University of British Columbia  
Department of Electrical and Computer Engineering  
Vancouver, British Columbia, Canada  
konradw@ece.ubc.ca

Samuel Sze Hang Ng  
University of British Columbia  
Department of Electrical and Computer Engineering  
Vancouver, British Columbia, Canada  
samueln@ece.ubc.ca

Robert Wille  
Chair for Design Automation, Technical University of Munich, Germany  
Software Competence Center Hagenberg GmbH, Austria  
robert.wille@tum.de

ABSTRACT
Field-coupled Nanocomputing (FCN) defines a class of post-CMOS nanotechnologies that promises compact layouts, low power operation, and high clock rates. Recent breakthroughs in the fabrication of Silicon Dangling Bonds (SiDBs) acting as quantum dots enabled the demonstration of a sub-30 nm² OR gate and wire segments. This motivated the research community to invest manual labor in the design of additional gates and whole circuits which, however, is currently severely limited by scalability issues. In this work, these limitations are overcome by the introduction of a design automation framework that establishes a flexible topology based on hexagons as well as a corresponding Bestagon gate library for this technology and, additionally, provides automatic methods for physical design. By this, the first design automation solution for the promising SiDB platform is proposed. In an effort to support open research and open data, the resulting framework as well as all design and code files are made publicly available.

CCS CONCEPTS
• Hardware → Quantum dots and cellular automata; Place-ment; Wire routing; Physical synthesis; Clock-network synthesis; Technology-mapping; Combinational synthesis; Circuit optimization; Design databases for EDA; Software tools for EDA; Equivalance checking; Simulation and emulation; Theorem proving and SAT solving; Design rule checking.

ACM Reference Format:

1 INTRODUCTION & MOTIVATION
Post-CMOS Field-coupled Nanocomputing (FCN) [2] technologies promise the possibility to realize nanometer-sized elementary logic-in-memory devices [20, 33], offer energy dissipation capabilities below the Landauer limit [20–22, 25, 42], or clock frequencies in the terahertz regime [27, 41]. For all these endeavors, FCN serves as an umbrella term for a class of emerging technologies that perform computations via the repulsion of physical fields instead of electrical current flow. Recent years have established tremendous advancements in the fabrication of Silicon Dangling Bonds (SiDBs) [1, 15, 18, 19, 31, 53] which exhibit quantum-dot behavior, an application which has been dubbed atomic silicon quantum dots. They can be used to implement logic under the FCN paradigm as proven by experimental demonstrations of a sub-30 nm² SiDB OR gate [18]; ushering in an exciting platform at the limit of scaling.

Motivated by the promise of these experimental demonstrations, the research community has started to show great interest in the SiDB platform; with multiple computational explorations leading to various manually designed gates and circuits [3, 10, 11, 29, 30, 45] along with promises of low-power and high frequency operation in a post-CMOS realm [10, 11, 29]. However, scaling up would greatly benefit from the contributions of the design automation community towards corresponding design frameworks, which do not currently exist for the SiDB platform. This is in contrast to other FCN implementations such as Quantum-dot Cellular Automata (QCA) [23] for which methodologies exist that respect the corresponding physical-level design rules [11, 17, 37], utilize established standard cell libraries [36] as well as technology-specific clocking strategies [5, 9, 26, 44], and provide tailored placement and routing algorithms [13, 43, 46, 49]. Unfortunately, those accomplishments cannot directly be used for SiDB since the corresponding physical basics (as demonstrated in [18]) require a substantially different approach. For example, whereas existing FCN design automation frameworks encode logic in terms of rectangular arrangements of quantum dots laid out on a Cartesian grid, this work shows that the SiDB platform is much more suited for a hexagonal architecture. This topology shift has heavy consequences on the correspondingly needed gate library, the resulting design rules, and the way physical design is supposed to be conducted. In this work, those challenges are addressed by

1. the establishment of a hexagonal floor plan topology,
2. the proposal of the Bestagon gate library, i.e., a set of hexagonal standard component tiles, which is carefully attuned to SiDB’s technological constraints with validation from the platform-specialized CAD tool SiQAD [30],
3. a design rule framework that respects current fabrication technology for clocking electrodes, physical limitations of Coulombic bias, and similar constraints, as well as
4. a physical design flow that uses the proposed hexagonal floor plans, gate library, and design rules to automatically generate dot-accurate SiDB circuit layouts from logical specifications.
which encode bit information in charge locations in rectangular areas of quantum dots as illustrated in Figure 1a [23]; in some cases, additional charge sites are used for null logic states [24]. An alternative implementation is to encode bit-states in pairs of quantum dots, dubbed Binary-dot Logic (BDL) [18], also illustrated in Figure 1a.

Among quantum-dot FCN implementations in the literature, the use of SiDBs on the hydrogen-passivated silicon(100) 2×1 surface is particularly intriguing as they are able to act as atomically-sized quantum dots [15, 18, 32, 35, 53], placing them at the ultimate limit of scaling. The H-Si(100)-2×1 surface has discretely defined sites where SiDBs can be fabricated with atomic precision using the probe of scanning tunneling microscopes [1, 19, 31], as illustrated in Figure 1b. A groundbreaking demonstration by Huff et al. [18] has shown experimentally that careful configurations of SiDBs can be used to construct BDL logic components.

In the demonstrated system setup, SiDBs may possess 0, 1, or 2 electrons, corresponding to positive, neutral, and negative charge states respectively. In simulations throughout this work, SiDBs with cyan fill represent negatively charged sites and hollow SiDBs represent neutral sites; positive charge states are not relevant to the configuration of interest [18, 30]. The charge states can be influenced by environmental factors such as the bulk dopant concentration [34] and the presence of electric fields [32]. In the absence of deliberate excitations, charges in SiDB systems exhibit a tendency to settle to low-energy configurations at both cryogenic [18, 35] and room temperatures [15]—a behavior that is desirable for charge-based ground state FCN logic.

These electronic properties enable the demonstration of BDL wire structures as well as a sub 30 nm² logic OR gate at the nanoscale [18]. A simulated reproduction of the OR gate is shown in Figure 1c. Here, the input bit state is set by the existence of a peripheral SiDB, dubbed a perturber [18, 30], which exerts Coulombic pressure on the input SiDB-pair to emulate the presence of an input BDL wire at logical state 1. An output perturber is present to emulate the presence of an output BDL wire. It can be observed that when one or both of the input SiDB pairs are set to logic 1 by input perturbers, the output also toggles to logic 1 state as expected of an OR gate. It is to be noted that the need for these perturbers will be alleviated upon the future development of I/O devices.

Prospective scaling of SiDB FCN systems requires the introduction of clocking and I/O circuitry. Hereby, clocking is a necessity for any sufficiently large FCN circuit layout—combinational or sequential. Clocking in FCN stabilizes signals and directs the flow of information in a pipeline-like fashion by alternately expressing certain activated regions which are able to hold logic states and carry out computations, and deactivated regions which act as separators and reduce cross-talk [16, 26]. SiDB clocking is expected to be achieved through the modulation of surface charge populations where segments can be deactivated by removing surface charges, creating an electrically neutral region [11, 30, 32]. To enforce information flow in designated directions, past SiDB logic research has proposed the use of a prevalent clocking strategy from FCN literature, i.e., four-phase clocking [11, 30], illustrated in Figure 2. For the benefit of design automation and design rule enforcement,
Hexagons instead of Cartesian ones. This hexagonal topology natively routing of the only experimentally verified SiDB gate type. Thus, established settings and physical design of inputs and outputs cannot properly be fitted into the Cartesian grids as illustrated in Figure 3a—it can be observed that connections such Cartesian grids cannot reasonably accommodate Y-shaped based floor plans for physical design algorithms [17]. However, plus-shape, Cartesian layout structures were established as tile-reviewed in Figure 1c is Y-shaped, which is an unusual shape for the OR gate demonstrated by Huff et al. Layout Topology:

The OR gate demonstrated by Huff et al. and reviewed in Figure 1c is Y-shaped, which is an unusual shape for an FCN gate to have. Since previous FCN gates, especially for QCA, have their inputs and outputs orthogonally aligned due to their plus-shape, Cartesian layout structures were established as tile-based floor plans for physical design algorithms [17]. However, such Cartesian grids cannot reasonably accommodate Y-shaped gates as illustrated in Figure 3a— it can be observed that connections of inputs and outputs cannot properly be fitted into the Cartesian coordinate structure. Thus, established settings and physical design methods for the FCN domain are not suitable for placement & routing of the only experimentally verified SiDB gate type. Hence, in order to respect the Y-shape gate configuration in the layout generation process, this paper proposes to utilize hexagonal grids instead of Cartesian ones. This hexagonal topology natively matches the demonstrated SiDB gate structure, as shown in Figure 3b. In this orientation, the input pins of all gates are directly accessible via the center of tile borders and the gates’ output can be propagated to either of the two bottom directions. Within each hexagonal tile, the gate can naturally be rotated or mirrored. This, however, requires new and correspondingly developed design methods.

Extended Gate Library: Huff et al.’s experimental demonstration of SiDB logic structures physically validated a Y-shaped OR gate (Figure 1c) and BDL wires [18]. However, logic layout generation requires additional components such as inverters, fan-out wires, and wire crossings. Moreover, standard libraries usually offer a broader variety of gate types to enable more cost-efficient technology-mapped logic networks and layouts. Hence, in this work, a corresponding gate library—coined the Bestagon library—is proposed which is designed with this versatility and flexibility in mind. The proposed gate tiles implement various Boolean functions and resemble the established Y-shaped topology as closely as possible.

Clocking Constraints: With a footprint of merely $30 \text{ nm}^2$, Huff et al.’s OR gate operates at the limit of scaling. This impressive achievement, which aces conventional fabrication, is simultaneously a blessing and a curse. While it theoretically allows for ultra area-efficient designs, current technology nodes cannot offer fabrication for clocking electrodes that match the OR gate’s dimensions. This circumstance dictates that realistic clock zone dimensions must be significantly greater than the demonstrated individual gate sizes. Hence, in order to additionally enable clocked designs, this work proposes to group multiple hexagonal tiles together in regions large enough to be driven by the same clocking electrode. All tiles in each resulting super-tile are exposed to the same clock field respectively and, thus, switch simultaneously. Innately, this approach restricts the obtained layouts to specific linear clocking schemes but ensures their physical fabricability.
structures, and the clocking constraints are highly interdependent, they are discussed as one interwoven entity in Section 4.1 with the goal to obtain the Bestagon gate library that conforms to the discussed constraints. Afterward, Section 4.2 presents a corresponding design flow for the obtainsment of dot-accurate SiDB circuit layouts via the utilization of the Bestagon library and adapted logic synthesis and physical design algorithms.

4.1 The Bestagon Gate Library

Section 3 discussed three design challenges and proposed solutions in the process of design automation for SiDBs. This section describes how to holistically address these three points, yielding a set of standard tiles—the Bestagon gate library.

The design of these standard tiles is informed by the consideration of realistic physical limitations of the clocking network and requires to keep the proposed logic layouts in line with Huff et al.’s demonstrated Y-shaped gates [18]. At state-of-the-art 7 nm lithography processes, the minimum metal pitch is 40 nm [54]. In comparison, Huff et al.’s OR gate has dimensions within 5 nm x 6 nm, which constitutes two options in designing the tiles: (1) extend the lengths of input and output BDL wires leading to the logic component at the center of the tile such that each tile is large enough to be individually addressable by a clocking electrode without violating the minimum metal pitch, or (2) maintain the ≈ 10 nm dimension for the core component tiles and group multiple of these tiles inside a super-tile, a collection of tiles large enough to be addressed as a single unit by a clocking electrode.

The former option offers individually addressable standard tiles at the cost of lower logic density; the latter allows the maintaining of logic density at the expense of potential complications in the handling of detailed information flow within the super-tiles. In this work, it is chosen to explore the latter path of higher logic density with the use of smaller standard tiles while managing the complexity of the placement and routing problem by relying on feed-forward clocking floor plans, e.g., Columnar [26] or 2DDWave [44].

Henceforth, each tile in the proposed Bestagon library is of hexagonal shape and consists of input and output BDL wires for connection and a logic design canvas in the center for logic implementation. An example of a 2-in-1-out tile template is illustrated in Figure 4. Input and output wire lengths are chosen such that the logic design canvases of adjacent tiles in all directions have at least 10 nm distance in order to reduce direct interference between logic components when using physical parameters calibrated to [18, 30]. For this study, templates for 1-in-1-out, 1-in-2-out, 2-in-1-out, and 2-in-2-out are proposed, including designed tiles for wires (vertical, diagonal, two parallel verticals), wire crossings, fan-outs, single-tile half adders, inverters (straight and diagonal), and common 2-in-1-out gates including OR, AND, NOR, NAND, XOR, and XNOR. Simulation results of select gate designs using the SimAnneal ground state finder in SiQAD are shown in Figure 5, with the rest available in an open repository [52].

These tiles have been designed with the assistance of a reinforcement learning agent [28] which is allowed to place SiDBs within the logic design canvas and toggle through input combinations to check for logic correctness. The layouts are manually reviewed and edited as needed for inclusion in the library.

The toggling of input logic states to the component tiles is a refinement upon Huff et al.’s methodology [18]; whereas Huff et al. represent a logic 1 input by the existence of an input perturber and logic 0 by the lack of it (see Figure 1c), this work employs the input perturber for both logic 0 and 1 states but created at either farther or closer locations, respectively. This constitutes a more realistic representation of the repulsion exerted by upstream input logic wires because at logic 0, Huff et al.’s method of removing the perturber does not capture any upstream input influence whereas the proposed method does. Ultimately, this approach generates gates that are more robust to disturbances of nearby SiDB structures.

The obtained Bestagon tiles are used in later sections for placement and routing demonstrations. Figure 6 shows such an employment of the tiles in a placed and routed implementation of the par_check benchmark from [43] (design flow detailed in the subsequent section). Logic signals flow from the top to bottom due to the employment of the Columnar clocking scheme [26] rotated by 90° yielding a row-based configuration where tile (x, y) is driven by clock zone y mod 4.

The depicted layout incorporates six different logic gates from the Bestagon library plus two wire configurations, fan-outs, and a crossing, which makes it a prime representative for the benefits of the proposed methods; namely ultra-dense physically sound and fabricable logic structures with a high Boolean expressiveness due to a variety of tile-based components.

4.2 Resulting Physical Design Flow

With the definition of the Bestagon gate library that complies with the hexagonal topology for Y-shaped gates and clocking constraints imposed by physical and fabrication limitations, certain established design automation principles from the QCA domain can be accordingly adjusted. This eventually results in a design flow that uses the Bestagon library and complies with the discussed design rules. The flow starts with specifications at the logic level, e.g., provided by gate-level Verilog or similar files, and automatically generates dot-accurate SiDBs layouts that implement the given functions and can be processed further by simulators, e.g., SiQAD [30] or for physical realization/fabrication.

The proposed flow is as follows:

1. parse a specification file as XOR-AND-Inverter Graph (XAG),
2. perform cut-based logic rewriting with an exact NPN database to reduce the XAG’s size and depth [38],
3. perform technology mapping [8] to structure XAG nodes into gates supported by the proposed Bestagon library,
4. generate a linearly clocked hexagonal gate-level layout from the mapped network via SMT-based exact [46] physical design,
5. perform SAT-based equivalence checking of the input network and the resulting gate-level layout [50],
6. merge adjacent tiles into super-tiles by expanding the clock zone dimensions,
7. apply the Bestagon library to map each gate to a dot-accurate representation yielding an SiDB layout, and
8. generate a design file from the SiDB layout for physical simulation and/or fabrication.

Since the Bestagon gate library supports both AND and XOR standard tiles, XAGs have been picked as the data structure of choice for the logic synthesis part as they offer a potentially more compact representation compared to AND-Inverter-Graphs (AIGs) with only a slight overhead in memory consumption. The optimized XAGs then serve as input to the physical design algorithm from [46] which—via some adjustments—is able to support both hexagonal layout topologies and the Bestagon library. However, since the algorithm does not support super-tiles, the additional step of clock zone expansion becomes necessary.

In order to validate the integrity and verify the logical correctness of the resulting layouts, SAT-based equivalence checking as

---

1. Even though there exist even more expressive data structures, e.g., XOR-Majority-Inverter-Graphs (XMGs), they usually represent logic by the means of Boolean functions that are unsupported by the Bestagon gate library, e.g., Majority.
proposed in [50] is applied first followed by physical simulation with SiQAD to ascertain operational behavior under known physical models.

In order to implement this flow, any algorithms for logic optimization, technology mapping, physical design, and formal verification can be applied as long as they are adjusted to respect the design rules proposed in this paper. Consequently, it is also possible to create a variety of gate libraries following the provided specifications to cover a larger part of the Boolean domain with elementary building blocks.

Therefore, a corresponding design automation framework was implemented that precisely follows the flow outlined above. It is confirmed in the following section that said implementation is in fact applicable for the generation of logically correct and physically feasible dot-accurate SiDB layouts.

5 APPLICATION

Whereas the previous sections provided a detailed description of the obtainment and specifics of the Bestagon library in accordance with physical constraints as well as a resulting design flow, this section illustrates its application and, thereby, supports the conceptual contributions with reproducible and verifiable data.

The proposed design flow was implemented in C++ and integrated into the open-source FCN framework fiction [47]. For logic network representation and optimization, the mockturtle [40] library was utilized. Established QCA benchmarks from [13, 43] were used to generate layouts that are comprehensible by humans for presentation in this work. An excerpt can be found in Table 1 where resulting layout dimensions, the number of needed SiDBs, and area requirements in nm$^2$ are listed. The generated layouts offer the highest possible throughput of $1/3$ because the applied physical design algorithms ensure the balancing of all signal paths.

As an illustrative example, Figure 6 depicts the synthesized layout of the par_check benchmark using the proposed design flow. All obtained design and simulation files as well as the code for their generation are publicly available at [52].

The obtained results demonstrate that the Bestagon library and the proposed design automation flow enable automatic layout generation for the promising SiDB platform. In contrast, previous efforts were limited to manual labor and, therefore, restricted to small designs. Furthermore, physical and fabrication restrictions were barely considered in the literature thus far while they are explicitly addressed in this work.

---

Table 1: Excerpt of generated layout data

<table>
<thead>
<tr>
<th>Name</th>
<th>$w \times h$</th>
<th>A</th>
<th>SiDBs</th>
<th>nm$^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>xor2</td>
<td>$2 \times 3$</td>
<td>6</td>
<td>58</td>
<td>2403.98</td>
</tr>
<tr>
<td>xnor2</td>
<td>$2 \times 3$</td>
<td>6</td>
<td>58</td>
<td>2403.98</td>
</tr>
<tr>
<td>[43] par_gen</td>
<td>$3 \times 4$</td>
<td>12</td>
<td>103</td>
<td>4830.22</td>
</tr>
<tr>
<td>mux21</td>
<td>$3 \times 6$</td>
<td>18</td>
<td>196</td>
<td>7258.52</td>
</tr>
<tr>
<td>[13] par_check</td>
<td>$4 \times 7$</td>
<td>28</td>
<td>284</td>
<td>11312.68</td>
</tr>
<tr>
<td>xor5_r1</td>
<td>$5 \times 6$</td>
<td>30</td>
<td>232</td>
<td>12124.57</td>
</tr>
<tr>
<td>xor5_majority</td>
<td>$5 \times 6$</td>
<td>30</td>
<td>244</td>
<td>12124.57</td>
</tr>
<tr>
<td>t</td>
<td>$5 \times 8$</td>
<td>40</td>
<td>426</td>
<td>16180.79</td>
</tr>
<tr>
<td>t_5</td>
<td>$5 \times 8$</td>
<td>40</td>
<td>448</td>
<td>16180.79</td>
</tr>
<tr>
<td>[13] c17</td>
<td>$5 \times 8$</td>
<td>40</td>
<td>396</td>
<td>16180.79</td>
</tr>
<tr>
<td>majority</td>
<td>$5 \times 11$</td>
<td>55</td>
<td>651</td>
<td>22285.12</td>
</tr>
<tr>
<td>majority_5_r1</td>
<td>$5 \times 12$</td>
<td>60</td>
<td>737</td>
<td>24293.23</td>
</tr>
<tr>
<td>cm82a_5</td>
<td>$5 \times 15$</td>
<td>75</td>
<td>1211</td>
<td>30377.56</td>
</tr>
<tr>
<td>newtag</td>
<td>$8 \times 10$</td>
<td>80</td>
<td>651</td>
<td>32419.82</td>
</tr>
</tbody>
</table>

$w \times h$: Aspect ratio given in hexagonal tiles
A: Layout area in tiles
SiDBs: Number of SiDB quantum dots
nm$^2$: Layout area in nm$^2$

The c17 benchmark was originally published in [7].
6 CONCLUSIONS

This work has touched on every foundational aspect for the en-
ableness of SiDB FCN logic design automation. At the physical
level, design rules were outlined, which are informed by recent
experimental demonstrations, physical models, and realistic fab-
rication constraints. At the gate level, the Bestagon gate library was
presented, which provides hexagonal tiles that serve as the basis for
physical design and future architectural studies. At the design
automation level, a complete design flow was presented that turns
logic networks into dot-accurate physical SiDB layouts. Thereby,
automatic design for this highly-promising contestant in the FCN
domain becomes feasible for the first time.

Since the implementation and all obtained design and simulation
files are publicly available [52], this work will serve to facilitate
various follow-up studies of interest to the FCN design automation
community, including higher-level architectural and application
studies into the platform.

There are also opportunities to improve upon the proposed
framework. Studies of more intricate clocking floor plans such as
USE [9] will require the development of intra-super-tile detailed
routing capabilities. The advancement of a streamlined operational
domain evaluation framework will also be of interest since the
existing work is computationally heavy and not trivially quantifi-
able [30]. Future availability of clocking-informed SiDB logic simu-
lations will also bring us closer to functioning fabricable layouts.

This entails that both the Bestagon gate library and the proposed
software suites will continue to evolve and facilitate the next gen-
eration of SiDB FCN design automation research.

ACKNOWLEDGMENTS

The authors would like to express their gratitude towards Amit
Patel for the wonderful resources on hexagonal grids on his website
Red Blob Games and CGP Grey for the title inspiration.

REFERENCES

devices and memories. Nature Communications 9, 1 (July 2018), 2778.
7 (2005), 619–624.
Circuits and a Target Translator in Fortran. In International Symposium on Circuits
Restructure Logic. (2022).
Scheme for QCA. TCAD 35, 3 (2016), 513–517.
Gates. In ISCAS.
4 (April 2012), 242–246.
Quantum Dots at Room Temperature. Physical Review Letters 102, 4 (2009),
046805.
JETC 1, 3 (2005), 163–185.