NanoSCL: A Standard Cell Library for Nanomagnetic Logic

Leandro Lázaro Araújo Vieira[1] Larissa Isabelle Rodrigues e Silva[1] Ruan Evangelista Formigoni[2]
Michael Canesche[2] Ricardo S. Ferreira[2] José Augusto M. Nacif[1]
[1]Science and Technology Institute, Universidade Federal de Viçosa, Florestal, Minas Gerais, Brasil
[2]Informatics Departament, Universidade Federal de Viçosa, Viçosa, Minas Gerais, Brasil
{leandro.lazaro, larissa.isabelle, ruan.formigoni, michael.canesche, ricardo, jnacif}@ufv.br

***Abstract*—The Complementary Metal-Oxide Semiconductor (CMOS) technology is reaching its physical limitations, and several recent works focus on Field-Coupled Nanocomputing technologies such as the Nanomagnetic Logic (NML). In our work, we present NanoSCL, a new standard cell library for this technology. NanoSCL provides logic blocks to simplify NML layout generation and abstract technology intrinsic details. Furthermore, NanoSCL provides automatic placement and routing of logic elements for a complete layout generation.**

I. Introduction

The CMOS (Complementary Metal-Oxide Semiconductor) [1] is the industry-leading technology for the fabrication of semiconductor devices. Over the decades, miniaturization of transistors devices scaled exponentially until the leakage current increase in CMOS technology started imposing several barriers to this shrinking pace [2]. Consequently, research efforts led to alternative beyond-CMOS technologies to improve the area, speed, and power consumption of future devices. Field-Coupled Nanocomputing (FCN) is a class of these candidate technologies, where computation occurs through field interactions among nanoscale elementary cell arrangements.

Nanomagnet Logic (NML) is an attractive FCN-based technology that is under active research and relies on nanomagnets' interaction to perform logic operations. The technology is non-volatile and operates at room temperature with ultra-low energy dissipation [3]. Signal propagation works by cascading signals through its nanomagnets, and this propagation relies on local magnetic interactions, which are enabled by a clocking system. Each nanomagnetic structure holds a magnetization referred to as "up" and ""down", representing the binary logic "0" and "1".

The task of building complex circuits in this novel technology increases with the inputs/outputs (I/O) and logic components. A standard cell library (SCL) provides abstraction, which ranges from simple logic gates to complex components, e.g., a cascade full-adder. Thus, the process does not have to handle internal component related details until later in the process, including an additional step to handle component-related details.

In this paper, we present a standard cell library for NML that features simple and complex building blocks, feedback support for sequential circuits, and a description language to define parameters for fully automated mapping. Furthermore, our NML standard cell library supports the BANCS [4] clocking scheme, which is a flexible and scalable structure to maintain circuit synchronization and enable scalable designs.

We organize this paper as follows: Section II provides a background of NML, Section III discusses previous research related to NML standard cells and automatic mapping. In Section IV, we present our SCL and its description language, followed by the results in Section V, with circuits designed with our SCL for proof of concept, and lastly, Section VI summarizes this paper.

II. Background

In this section, we present a summary of NML. The primary element of the technology, how signal propagation works, the used 3-phase clocking system, and how clocking schemes help with signal synchronization and integrity.

A. Nanomagnetic Logic

NML uses interaction among nanomagnets to perform computation. A magnetization directed to north represents logic state 0 and the opposite, logic state 1. When nanomagnets are placed and routed in a controlled manner, they can behave like wires or logic gates. Figure 1 shows two possible organizations to build wires with the NML technology, Figure 1a shows ferromagnetic wire, which replicates the input signal in a domino-like manner. Figure 1b illustrates an antiferromagnetic wire. In this arrangement, the signal logic value inverts after propagating to each neighbor nanomagnet. Thus an NML wire with an odd number of magnets acts as an inverter.

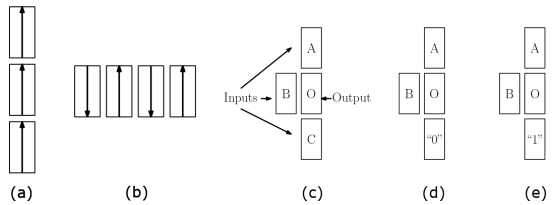

Fig. 1. (a) A ferromagnetic wire. (b) An antiferromagnetic wire. (c) A NML majority gate. (d) AND gate using NML majority gate. (e) OR gate using NML majority gate

Figure 1c shows a majority gate (MG), which is a particular structure for FCN technologies, replicating the most common signal in one of its three inputs. It is possible to reduce an MG to *And* and *Or* elements with a fixed input, to upwards and downwards magnetizations, as shown by Figures 1d-e.

B. NML Clocking

A 3-phase clocking system is used for synchronization in NML circuits [5]. The phases are referred as *Reset*, *Switch* and *Hold*, Figure 2a, shows an input nanomagnet in black, the succeeding elements in dark gray are in the *Reset* state, which means they are prone to be influenced by the magnetization of the input. This happens in Figure 2b, the elements transition to the switch state, and propagate the input signal to the subsequent elements in light gray, which are put in the *Reset* state to propagate the signal further, Figure 2c shows the propagation to the elements in light gray, where the previous elements are in the *Hold* state, to maintain the signal until it is propagated correctly.

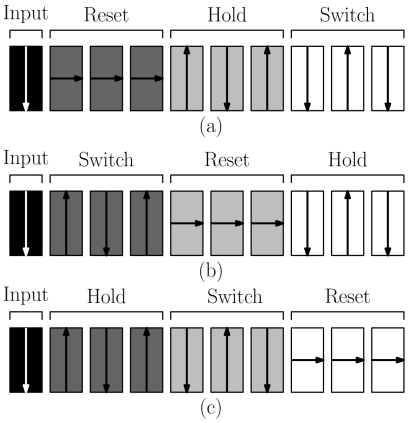

Fig. 2. 3-Phase clocking system for NML. (a) Light gray elements are in the Reset state. (b) The signal from the input transitions to the elements in light gray. (c) The signal from the elements in light gray transitions to the ones in dark gray.

C. Clocking Schemes

A clocking scheme (CS) is a patterned structure that subdivides the layout area into several zones. These zones accommodate a pre-defined number of nanomagnets, and transition between each NML clocking phase. The CS used by our library is BANCS, the basic *cutout* for this CS is shown in Figure 4a. BANCS has alternative signal directions indicated by the edge arrows. Each shade of gray in the *cutout* represents a *tile*, which has $3x3$ capacity of nanomagnets. This signal flow distribution allows developing sequential circuits with feedback paths.

III. Related Work

In this section, we review previous proposals of SCLs, focusing on clarifying points of improvement, which we consider with our design to enhance the results over the proposed works.

MagnetoElastic [6] is an NML SCL that uses a clocking mechanism based on an electric field that effectively forces magnets in the *Reset* state [7]. In this work, the authors claim to reduce the area by a factor of 4 with a 50 times reduction in power consumption. The main issue of the proposal is power losses of NML clocks, the cell width and height ranges from three to five nanomagnets, where the $3x3$ provides more efficiency since it has the shortest critical pattern.

A different design for NML logic circuits has been proposed based on Magnetic RAM, M-RAM [8], which is a well-developed technology that provides an interface regarding the CMOS technologies. It presents Magneto-Tunnel Junctions (MTJs) as fundamental elements. The spin Hall effect is the basis of the clocking mechanism [8], showing the feasibility of complex circuits built with MTJs in NML technology. M-RAM can be used to design magnetic logic circuits. The limitations of M-RAM structures and clocking zones do not lead to optimum results from the area and power consumption. NML circuits based on M-RAM structures [8] are modeled using VHDL language, where it is possible to simulate complex architectures in the NANOcom tool [9]. It is essential to highlight that this work [8] does not propose a complete standard cell library. It presents only an alternative way to design circuits with input/output cells, MTJ cells with and without access transistor, and blank cells.

Our study proposes a complete SCL designed for NML-Sim [10], allowing the user to design nanomagnetic circuits freely, and provides detailed layout exploitation and simulation. We offer an area and delay improvements over current methods [8]. Furthermore, our library provides a greater variety of standard cells than the MagnetoElastic approach [6]. In our research, we did not find any recent work that addressed the design of a Standard Cells library for NML.

IV. NanoSCL

The process for layout generation starts in a hardware description language (HDL), which is translated from the sum-of-products form to a directed acyclic graph (DAG) with the ABC tool [11]. The DAG can be optionally optimized with the EPFL logic and synthesis tools [12]. In the next phase, NanoSCL is used to define the logic blocks' position to build the complete layout, through a reference language offered by the SCL.

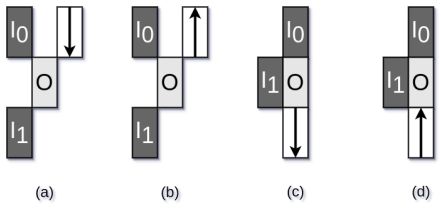

Fig. 3. Basic logic elements. (a) And. (b) Or. (c) Nand. (d) Nor.

NanoSCL logic elements are a three-input majority gate, two-input *And*/*Or* gates and an *Inverter*. With majority gates, we develop the fundamental Boolean logic, as shown in

Figure 3. The constant input one (True) and zero (False) are represented by the ↑ and ↓ symbols, respectively. The output has the label O, and the inputs I_0 and I_1. As shown in Figure 3, if we set one of the inputs to 0 (↓). We get an AND gate and, if it is set to 1 (↑), we will get an OR gate. The same methodology was adopted to create *Nand* and *Nor* gates.

In our approach, each descriptor specifies a simple or complex gate. A simple gate is an element from Figure 3. Other structures are referred to as complex. Our library extends the *QCA One* library [13] which was originally developed for the Quantum-dot Cellular Automata (QCA) technology; as a proof of our concept, we utilize the NMLSIM 1.0 tool [10] for verification. To enforce circuit synchronization and ease the implementation of placement and routing approaches, we utilize the Bidirectional Alternating Nanomagnetic Clocking Scheme (BANCS) [4]. BANCS generates additional area overhead as a trade-off. Figure 4a illustrates BANCS, each arrow indicates the signal flow direction, which provides feedback path support. The simple elements occupy a single tile in the clocking scheme. We present a sample MUX 2:1 circuit with our library. Figure 4b shows the schematics Figure 4c presents the realization.

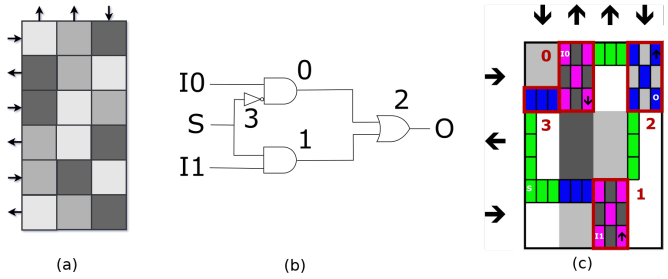

Fig. 4. (a) BANCS cutout. (b) MUX 2:1 schematic. (c) MUX 2:1 standard cell.

We show the respective layout descriptor for the *MUX 2:1* in Figure 5. We declare the circuit name followed by the interface, which includes the name and number of inputs and outputs. Next, we describe the boolean function, this information is useful for logic synthesis and technology mapping tools. Next, the zone dimension specifies one region's width and height in the clock grid in nanomagnets. For the MUX2:1, the layout has 12x9 magnets and 4x3 tiles (the grey color background in Figure 4c).

Figure 6a shows a two-input XOR gate cell. This complex cell has two synchronized inputs I_0 and I_1. Both connect to the clocking zone in blue color. Also, both the input signals traverse two clocking zones (purple and green) to achieve the two gate inputs in the first level at the blue color clocking zone. It is important to highlight that the wire from I_0 propagates through six magnets (one clocking zone) in the purple clocking zone in the vertical direction in the first column and only three magnets (one clocking zone) in the horizontal bottom line to satisfy the BANCS constraints. To demonstrate the feedback capabilities of NanoSCL, we build a *SR-Latch* in Figure 6b.

```
name: mux           | up_l_reference_zone: 2
n_input: 3          | up_r_reference_zone: 3
n_output: 1         | down_l_reference_zone: 1
expression:         | down_r_reference_zone: 2
O=(I0&~S)+(I0&I1)   | Delay table:
zone_dimension: 3   | / I0 I1 S
width: 12           | 0  2  2  4
height: 12          | Port location:
                    | I0 3 0 0
                    | I1 6 8 0
                    | S  0 6 0
                    | O 11 2 0
```

Fig. 5. SCL language to define the input SR-Latch circuit information.

Figure 6b presents an SR-Latch element, in which the signal flow at input I_0 crosses the cell from left to right, i.e., it adheres to the constraints defined by BANCS. This is ensured with four parameters, up_l_reference_zone references the tile in the upper left corner, and up_r_reference_zone in the upper right corner. Similarly, down_l_reference_zone references the tile in lower-left corner, and down_r_reference_zone to the lower right corner. Afterwards, we describe two important data for the placement and routing tools: cell delay and cell in/out positions. The delay_table reports the number of cycles (clocking zones) from each cell from input to output. In our example, a signal from the inputs I_0 and I_1 transverses two clocking zones. It is important to highlight that the S signal propagates through two paths, with the same length in clocking zones for circuit synchronization. Finally, the port_location informs the relative coordinates of each cell input and output.

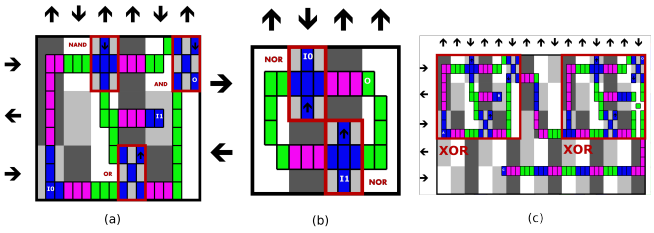

Fig. 6. Standard cells. (a) XOR. (b) SR-Latch. (c) Cascade XOR.

V. Results

In this section, we present the results of circuit generation with our SCL, illustrating the layouts of realized circuits. Figure 6c shows a CASCADE-XOR composed of two standard XOR cells. We moved input C to traverse more clocking zones, thus compensating for the delay caused by inputs A and B concerning output O. Figure 7a shows the layout of circuit C17. In contrast, Figure 7b and Figure 7c show C17 placement and routing with and without our library, respectively. Note that we significantly reduced the area with our library. We outlined and marked the gates in both circuits in red to facilitate the comparison of the circuits.

In Table I, we compare the delay and area of the realized circuits. We measure the area in *tiles* and the delay in clock cycles. For the MUX 2:1, we achieve a single clock cycle

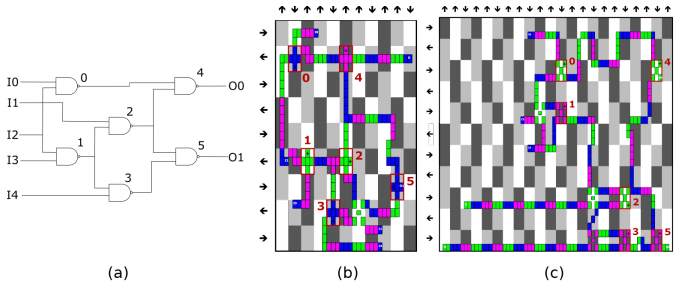

Fig. 7. C17 circuit. (a) Schematic. (b) With standard cells. (c) Without standard cells.

delay to generate a valid output, and the area dimension is rather small for a circuit with four logic gates. We demonstrate in the next entry the design for an SR-Latch, a sequential circuit, which has a higher ratio of gates per area, due to synchronization requirements wires have to follow.

The next analyzed circuit is an XOR gate, which is more complex than the previous combinational circuit. Thus, it yields a more significant area overhead, even with the same number of logic gates. Furthermore, the delay is now two clock cycles. In the 2-bit cascade realization of the XOR gate, the area does not grow in a linear behavior, with an overhead of 4.16% more than the single-bit design. Moreover, we can observe the delay behavior, which increases 3 times when compared to the more straightforward design.

Lastly, we analyze the C17 circuit, the most significant design, with 12 logic gates. The area overhead is not as significant when compared to the cascade XOR because of the external connections of the single-bit elements of the previous layout leading to the smaller design and shorter delay of 5 clock cycles to the outputs.

TABLE I
NANOSCL CIRCUITS COMPARISON

Circuit	Gates	Area	Delay
MUX 2:1	4	3x4	1
SR-Latch	2	2x4	1
XOR	4	3x6	2
Cascade XOR	8	5x15	6
C17	12	9x11	5

VI. CONCLUSIONS

In this article, we introduce NanoSCL, a standard cell library that uses the BANCS clock scheme. As the illustration shows, compared to other standard cell libraries for NML, NanoSCL provides a complete language to describe the circuit to be generated. Thanks to our library, it is possible to abstract from part of the placement and routing process. As future work, we intend to improve our library and expand it further to make it more useful, compact, practical, and effective.

ACKNOWLEDGMENTS

We would like to thank CNPq, CAPES, and FAPEMIG for the financial support.

REFERENCES

[1] F. Wanlass and C. Sah, "Nanowatt logic using field-effect metal-oxide semiconductor triodes," in *1963 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, vol. VI, 1963, pp. 32–33.

[2] S. S. BintiMdSallah, H. Mohamed, M. Mamun, and M. S. Amin, "Cmos downsizing: Present, past and future," *Journal of Applied Sciences Research*, vol. 8, no. 8, pp. 4138–4146, 2012.

[3] N. G. Anderson and S. Bhanja, *Field-Coupled Nanocomputing*, 1st ed. Springer-Verlag Berlin Heidelberg, 2014.

[4] R. E. Formigoni, O. P. V. Neto, and J. A. M. Nacif, "Bancs: Bidirectional alternating nanomagnetic clocking scheme," in *2018 31st Symposium on Integrated Circuits and Systems Design (SBCCI)*. IEEE, 2018, pp. 1–6.

[5] M. Vacca, M. Graziano, A. Chiolerio, A. Lamberti, M. Laurenti, D. Balma, E. Enrico, F. Celegato, P. Tiberto, L. Boarino *et al.*, "Electric clock for nanomagnet logic circuits," in *Field-Coupled Nanocomputing*. Springer, 2014, pp. 73–110.

[6] D. Giri, M. Vacca, G. Causapruno, W. Rao, M. Graziano, and M. Zamboni, "A standard cell approach for magnetoelastic nml circuits," in *2014 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*. IEEE, 2014, pp. 65–70.

[7] M. Vacca, M. Graziano, L. Di Crescenzo, A. Chiolerio, A. Lamberti, D. Balma, G. Canavese, F. Celegato, E. Enrico, P. Tiberto *et al.*, "Magnetoelastic clock system for nanomagnet logic," *IEEE Transactions on Nanotechnology*, vol. 13, no. 5, pp. 963–973, 2014.

[8] M. Bollo, G. Turvani, M. Zamboni, J. Das, S. Bhanja, and M. Graziano, "Design of nml circuits based on m-ram," in *2015 IEEE 15th International Conference on Nanotechnology (IEEE-NANO)*. IEEE, 2015, pp. 1339–1342.

[9] S. Frache, D. Chiabrando, M. Graziano, M. Vacca, L. Boarino, and M. Zamboni, "Enabling design and simulation of massive parallel nanoarchitectures," *Journal of Parallel and Distributed Computing*, vol. 74, no. 6, pp. 2530–2541, 2014.

[10] T. R. Soares, J. G. N. Rahmeier, V. C. De Lima, L. Lascasas, L. G. C. Melo, and O. P. V. Neto, "Nmlsim: a nanomagnetic logic (nml) circuit designer and simulation tool," *Journal of Computational Electronics*, vol. 17, no. 3, pp. 1370–1381, 2018.

[11] R. Brayton and A. Mishchenko, "Abc: An academic industrial-strength verification tool," in *International Conference on Computer Aided Verification*. Springer, 2010, pp. 24–40.

[12] L. Amarù, P.-E. Gaillardon, and G. De Micheli, "The epfl combinational benchmark suite," *Proceedings of the 24th International Workshop on Logic; Synthesis (IWLS)*, 2015. [Online]. Available: http://infoscience.epfl.ch/record/207551

[13] C. A. T. Campos, "A feasible clocking scheme (use) and a standard cells library (qca one) for future quantum-dot cellular automata," 2015.