



Evaluating nanomagnetic logic circuit layouts using different clock schemes

Ruan Evangelista Formigoni¹ · Leandro Lázaro Araújo Vieira¹ · Omar Paranaíba Vilela Neto² · Ricardo Ferreira³ · José Augusto M. Nacif¹

Received: 7 August 2019 / Revised: 26 January 2020 / Accepted: 30 March 2020 / Published online: 22 April 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

The complementary metal oxide semiconductor technology, CMOS, is reaching its physical limitations, as the transistors' feature size decreases. A promising alternative is the nanomagnetic logic technology (NML), a paradigm of field-coupled nanocomputing. This technology applies single domain nanomagnets to implement digital logic with switching energies that are orders of magnitude lower than a CMOS transistor due to the complete absence of static energy dissipation. When designing nanomagnetic circuitry, several challenges arise, such as the design of a clocking system able to avoid signal disruption due to the thermal noise effect. In this paper, we compare four NML clocking schemes: BANCS, USE, RES, and 2DDWave by analyzing scalability and area overhead of combinational and sequential circuits.

Keywords Field-coupled nanocomputing · Nanomagnetic logic · Quantum-dot cellular automata · QCA · Clocking scheme · Bidirectional alternating nanomagnetic clocking scheme · BANCS

1 Introduction

Since the first appearance in a 1963 paper by Wanlass and Sah [6, 33], the complementary metal-oxide semiconductor (CMOS) transistor has become the leading technology used

in digital electronic devices, unfortunately, the previous achievements are coming to a halt. The traditional transistor is reaching its physical limits. At the same time, reliability and power issues are rising at an alarming pace. Even though there is no mature technology available yet, many new devices are considered as a replacement for CMOS transistors, many of which do not even use electron charge as state variables [5].

One attractive alternative to charge-based devices is the field-coupled nanocomputing (FCN) paradigm [1], where circuits can execute all logic operations based on local field interactions between nanoscale building blocks that are organized in patterned arrays. Several FCN paradigms are currently under active investigation, including nanomagnetic logic (NML) [8] and quantum-dot cellular automata (QCA) [19].

NML is known as the magnetic QCA and presents some advantages, such as operating at room temperature. Here, the circuits exploit the magnetic “stray” field produced by one or more (input) nanomagnets to change the magnetization of the neighbor nanomagnets. This influence occurs through the magnetostatic coupling, which depends on the magnetization direction and the relative position/distance between the magnetic particles. The device magnetization

✉ Ruan Evangelista Formigoni
ruan.formigoni@ufv.br

Leandro Lázaro Araújo Vieira
leandro.lazaro@ufv.br

Omar Paranaíba Vilela Neto
omar@dcc.ufmg.br

Ricardo Ferreira
ricardo@ufv.br

José Augusto M. Nacif
jnacif@ufv.br

¹ Science and Technology Institute, Universidade Federal de Viçosa, Campus Florestal, Rodovia LMG 818, km 06., Florestal, MG, Brazil

² Universidade Federal de Minas Gerais, Av. Pres. Antônio Carlos, 6627 - Pampulha, Belo Horizonte, MG 31270-901, Brazil

³ Universidade Federal de Viçosa, Avenida Peter Henry Rolfs, s/n - Campus Universitário, Viçosa, MG 36570-977, Brazil

is associated with ‘0’ and ‘1’ binary logic states, allowing them to perform Boolean logic operations. Some simple NML circuits have been experimentally demonstrated [14, 29, 30].

An external clock enables the correct propagation of a signal in an array of nanomagnets. The clocking system in NML circuits has three purposes: to avoid signal error in long arrays of nanomagnets, to yield an adiabatic change of magnetization, and to ensure signal synchronization. Several efforts have been made to design an efficient clocking system for NML [3, 13, 20, 26]. At a higher abstraction level, we can organize these clocking systems in clocking schemes with the usage of restrictive routing grids.

In this paper, we explain in-depth BANCS [10], a QCA-inspired clocking scheme for NML, which is scalable and flexible enough to enable feedback paths and to route. Moreover, we discuss how these structures are essential to building scalable solutions. Subsequently, we reference the current state-of-the-art frameworks and tools to aid the process of the transposition of a circuit specification onto a clocking scheme. Finally, we compare BANCS with three other QCA clocking schemes [4, 10, 12, 27, 28].

We organize this paper as follows: Sect. 2 reviews the basics of the NML technology, its basic logic elements, and how the clocking system works in the technology. Section 3 presents in detail BANCS along with its design challenges. Section 4 shows and compares circuits implemented in BANCS and three other clocking schemes. Finally, Sect. 5 summarizes the topics presented in the paper.

2 Background

In this section, we present an overview of NML technology. We show how the nanomagnets interact with each other to build the essential logical devices to perform computation. We also show how the clocking system affects a circuit’s stability and synchronization. The technology has no static power dissipation, which is one of the issues with the CMOS technology, the switching energy of a nanomagnetic device can be orders of magnitude lower than a charge-based CMOS transistor.

2.1 Nanomagnetic logic basics

The basic building block of an NML circuit is a rectangular-shape nanomagnet (other geometries are also possible). The nanomagnet must be small enough (around 200 nm long) to present only one magnetic domain. The nanomagnet’s magnetization of an isolated elongated nanomagnet is likely to lie along its longer axis (also known as easy axis), to minimize the shape energy. This

energy has two stable minimum, yielding the magnetization vector to point in any of the two possible directions along the length of the rectangle. We defined the logical values ‘1’ and ‘0’ when the magnetization points “up” and “down”, respectively. An external magnetic field can be applied to bring the targeted nanomagnets into a “null” state [13]; furthermore, the spin-hall effect can be explored to avoid the usage of an external magnetic field at all [3].

Wires in NML can be arranged in two basic configurations, exploiting the coupling between nanomagnets. These are known as ferromagnetic [Fig. 1(a)] or antiferromagnetic [Fig. 1(b)]. The alignment of the magnetization is parallel in the former, while it is antiparallel in the latter. For the configuration shown in Fig. 1(b), it is possible to perceive that in this technology, an inverter can be represented by a wire with an even number of nanomagnets.

The fundamental logic element is the majority gate, shown in Fig. 2. Nanomagnets ‘A’, ‘B’, and ‘C’ are the inputs and nanomagnet ‘O’ is the output. The magnetic coupling between nanomagnets ‘A’ and ‘O’ and between ‘C’ and ‘O’ forces the latter to magnetize ferromagnetically, while the influence of ‘B’ over ‘O’ favors antiferromagnetic coupling. The majority gate takes three inputs and retrieves the majority of the magnetization results of ‘A’, ‘ $\neg B$ ’, and ‘C’.

It is possible to reduce a majority gate to an AND or an OR gate. To this end, we should arbitrarily choose one of the three inputs (‘A’, ‘B’ or ‘C’) and set it equal to ‘0’ (down) or ‘1’ (up), respectively. In this example, we’ve chosen the ‘C’ input as a fixed nanomagnet. By setting ‘C’ to ‘0’, the majority gate is reduced to an AND gate, as shown in Fig. 2(b). This way, the output equals ‘1’ only if ‘A’ and ‘ $\neg B$ ’ are both ‘1’. Similarly, by setting ‘C’ to ‘1’, the majority gate is reduced to an OR gate, as shown in Fig. 2(c). Thus, the output equals ‘1’ if at least ‘A’ or ‘ $\neg B$ ’ are equal to ‘1’.

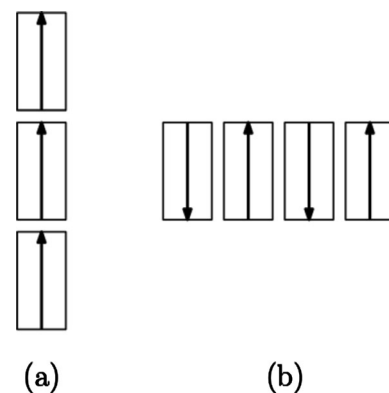


Fig. 1 **a** A ferromagnetic wire, **b** an antiferromagnetic wire

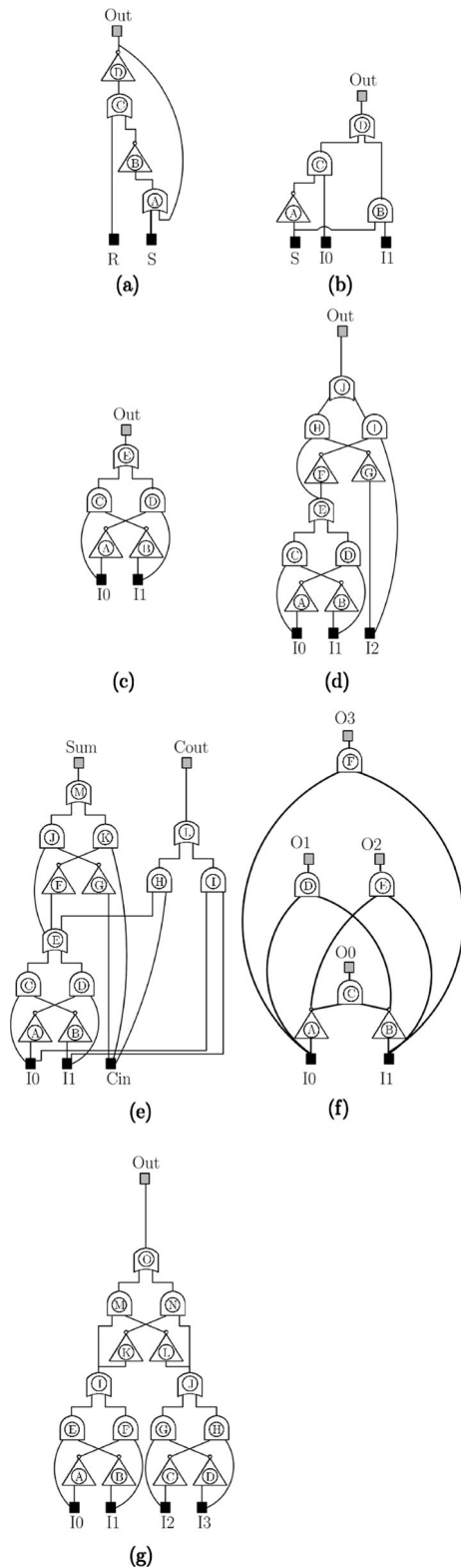


Fig. 2 **a** A majority gate, **b** 1-bit AND gate with 'A' and 'B' as inputs, **c** 1-bit OR gate with 'A' and 'B' as inputs

2.2 Clocking

The clocking system is an important issue in NML circuits. As an example, we consider how the clocking [13] works under the application of an adiabatic clocking field. If we switch the input of an NML array suddenly, the array is momentarily in some combination of exit states [e.g., the wire shown in Fig. 3(a)]. The first nanomagnet does not have a magnetic field strong enough to change the magnetization of its neighbors. To address this issue, an external magnetic field is applied to aid the switching of the nanomagnets by forcing them into a null state [Fig. 3(b)]. The external magnetic field is then slowly removed from all the neighboring magnets simultaneously and the nanomagnet *M1* induces the magnetization of *M2* [Fig. 3(c)], which also influences the magnetization of *M3* [Fig. 3(d)].

Furthermore, the clocking system is applied to avoid signal error in long arrays of nanomagnets. The wire length cannot grow indefinitely without causing ordering errors. As pointed out by Csaba and Porod [7], wires with more than five nanomagnets present a high error rate due to thermal noise. The issue is exemplified in Fig. 4. Figure 4(a), shows an antiferromagnetic wire with the first magnet working as an input; Fig. 4(b) shows the result of applying an external magnetic field on all the magnets to the right of the input, they are now in a null state. Finally, in Fig. 4(c), the magnetic field is gradually removed from all the targeted magnets simultaneously, and the input magnetic field is now enough to cause a domino-like effect, cascading the signal through their consecutive neighbors. Unfortunately, when we remove as the magnetic, thermal noise can switch a magnet prematurely before the signal propagates, therefore causing an ordering error. Thus, we achieve proper signal propagation and synchronization by splitting the circuit into groups called clocked tiles, and by submitting them to different external magnetic fields (clock signals) [13].

In NML, a periodic clock signal controls each tile. Each clock signal is composed of three phases [13] called Hold, Reset, and Switch. In the Hold phase, the external magnetic

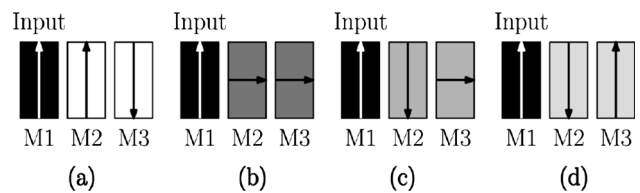


Fig. 3 **a** 'M1' changes magnetization but 'M2' and 'M3' remain unaffected. **b** An external magnetic field is applied to 'M2' and 'M3' (the grayscale represents the magnetic field strength). **c** The magnetic field is slowly removed and 'M1' sets the magnetization state of 'M2'. **d** 'M2' sets the magnetization of 'M3'

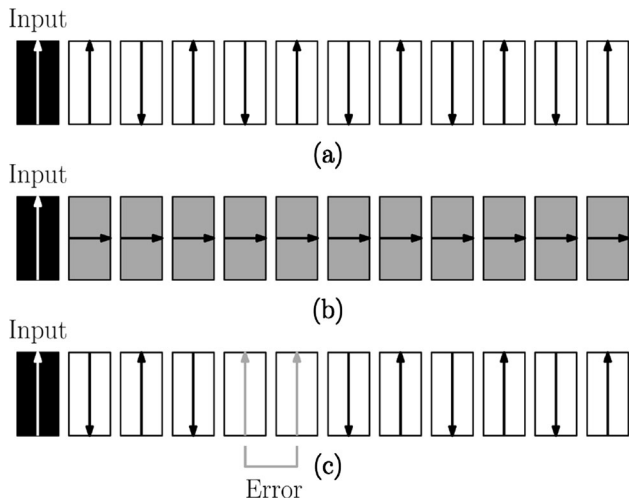


Fig. 4 Ordering error example. **a** The magnetization of the input magnet is inverted. **b** The remaining magnets are submitted to a clocking field and induced to a RESET state. **c** The input magnet cascades the signal, but the nanomagnet six has its magnetization set by thermal noise before the propagation reaches it

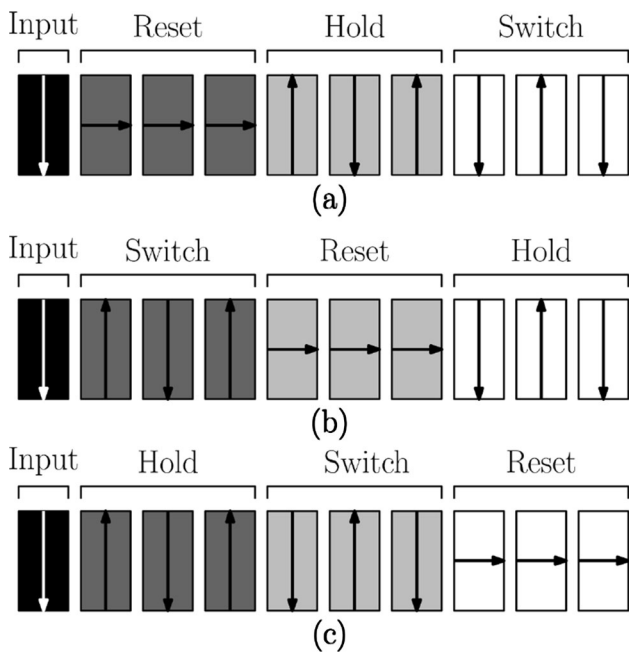


Fig. 5 Use of clocked tiles to solve the ordering errors in magnet arrays. An input nanomagnet is highlighted in green. **a** The tile closest to the input are in the *reset* phase, highlighted in dark gray. **b** The signal of the input propagates when the nanomagnets in the first tile are transitioned to the *switch* phase. **c** The signal propagates to the following tiles, highlighted in medium gray and light gray using the same idea

field is zero. Thus the magnetization of the nanomagnets remains unchanged. In the Reset phase, the magnetic field is applied, inducing the nanomagnets into a “null” magnetization state. In the Switch phase, we gradually remove

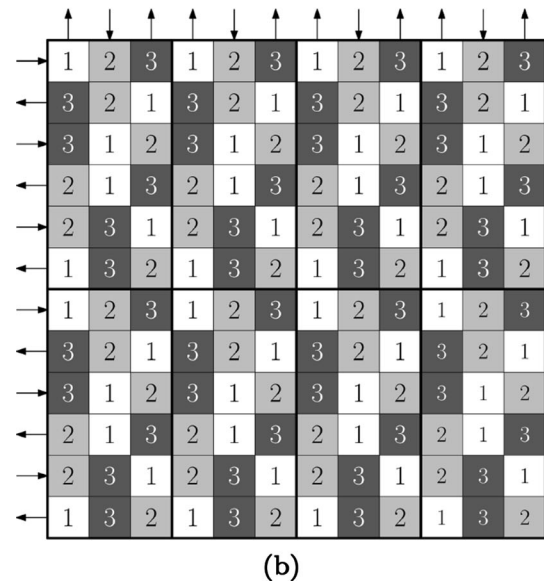
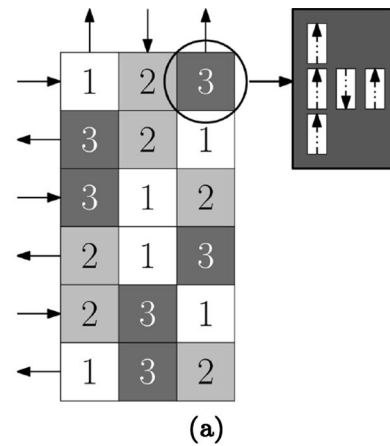


Fig. 6 **a** The BANCS cutout; **b** the BANCS clocking scheme

the magnetic field, allowing the nanomagnets to change their magnetizations according to their neighbors’ influences. In a circuit cutout, the magnetic fields will act upon each tile independently, thus eliminating errors if the number of nanomagnets within a tile does not exceed its limits. A clock cycle in NML is the time a tile needs to pass through all the aforementioned three phases.

Figure 5 presents an example with an antiferromagnetic wire. In this case, the nanomagnet with a black background represents the input; the grayscale colors are clocked tiles with nanomagnets within them. These are subjected to the same clock phase. Figure 5(a) shows an input with the downwards direction, and three arrays of three magnets on the *reset*, *hold*, and *switch* states, respectively. In Fig. 5(b), the first clocked tile is in the switch phase, to propagate the signal. Subsequently, in Fig. 5(c), the first clocked zone enters the hold state to propagate its signal to the second clocked zone. This process can happen indefinitely.

Fig. 7 Creating a stable path to avoid signal disruption

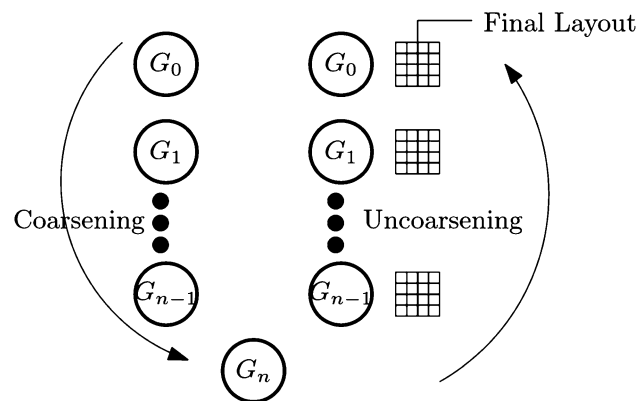
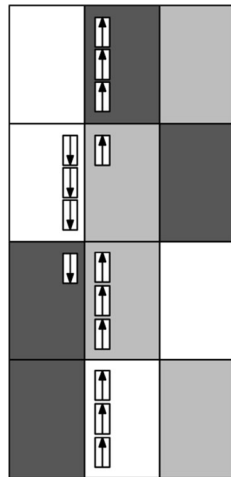


Fig. 8 The algorithm used for the placement and routing of the circuits

3 Bidirectional alternating nanomagnetic clocking scheme

A clocking scheme is a structure to standardize the arrangement of the tiles of a circuit. It defines the size of the tiles and determines immutable clock phase arrangements to allow only valid transitions to be performed during the circuit layout transposition onto the scheme, which can be accomplished by placement and routing algorithms [9, 24]. The main concerns when designing a novel proposal are scalability and regularity. The former defines how well the clocking scheme can scale without generating an excessive area overhead, although this is highly dependent of the chosen placement and routing algorithm. As a downside, irregular clocking schemes can increase the complexity of placement and routing algorithms; they can indefinitely scale just as regular clocking schemes.

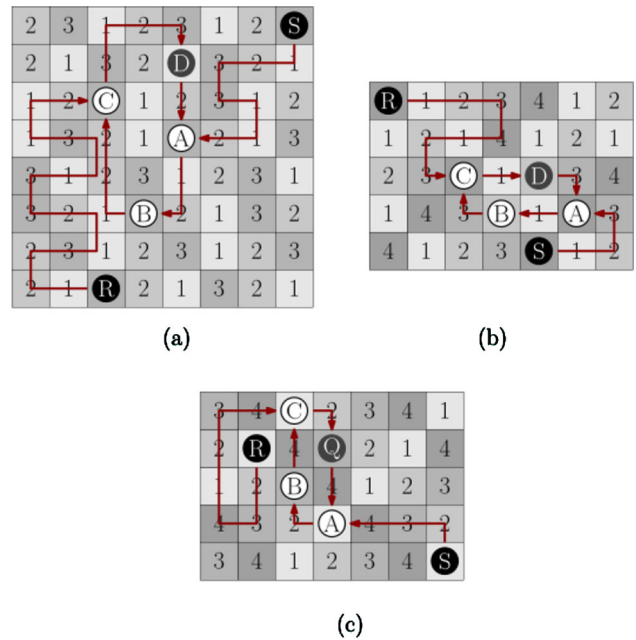


Fig. 9 The figures represent graphs of the circuits to be analyzed. **a** SR-Latch. **b** 2:1 MUX. **c** XOR logic gate. **d** Parity generator. **e** 1-bit ripple carry adder. **f** Decoder. **g** Parity checker

BANCS cutout is shown in Fig. 6(a). We refer to each numbered area as a *tile* and our building block is composed of 18 tiles, where all have the same dimensions, 3×3 nanomagnets. This choice was made to avoid signal disruption by the thermal noise effect as was previously shown in Sect. 2.2 [7].

Figure 6(b) illustrates how to create larger BANCS routing grids using the basic cutout. The building block is vertically stacked and horizontally paired with copies of itself to create an arbitrarily larger grid, i.e., a clocking scheme. It scales indefinitely and, at the same time, conforms with the restrictions for signal flow consistency.

The arrows in Fig. 6(a, b) indicate the direction in which the signal propagates. BANCS presents alternating signal flow directions when considering the rows (left and right). On the other hand, the vertical flow is characterized by two columns in the upward direction followed by one in the downward direction. This is a consequence of our design choices to deal with the challenges of designing a 3-phase clocking scheme.

The arrows in Fig. 6(a, b) indicate the direction in which the signal propagates. BANCS presents alternating signal flow directions when considering the rows (left and right). On the other hand, we represent the vertical flow by two columns in the upward direction, followed by one in the downward direction. This is a consequence of our design choices to deal with the challenges of designing a 3-phase clocking scheme. There are several efforts to minimize and analyze the power cost of clocking a nanomagnetic

circuit [2, 20], applying a magnetic field to the magnets, and most recently exploring the spin hall effect to avoid the usage of the field above, thus, reducing power consumption [3].

The tiles 1, 2 and 3 in Fig. 6(a) are always submitted to different clock phases at some point during the clock cycle, e.g., when the tiles labeled as 1 are all on the *reset* state, the ones labeled 2 are all in the *hold* state and those labeled 3 are all in the *switch* state. Considering the phase ordering mentioned in Sect. 2.2, the tiles labeled 3 should never be at the *reset* state and the tiles labeled 1 in the *switch* state because that would mean using nanomagnets in a *reset* state to set the values of nanomagnets in the *switch* state, thus, leading the resulting signal to be defined exclusively by thermal noise. BANCS eliminates this issue with its stable path generation and the design of its cutout.

Figure 7 shows how BANCS addresses the thermal noise effect problem when the 5-magnet limit is not respected. This happens because BANCS has two consecutive tiles on the same clock phase when the signal flows in the vertically upward direction. As a direct consequence, it is possible to consecutively position six nanomagnets side-by-side, thus, overstepping the five consecutive nanomagnets limitation. The solution is to use the tiles in an adjacent column to create a new stable path for the signal flow. The signal is reversed, due to antiferromagnetic coupling, when transitioned to the adjacent tile and reversed back to its original value when transitioned to a tile in its original column. The only drawback of this solution is the addition of one more clock cycle to the wire delay for each time we apply this technique.

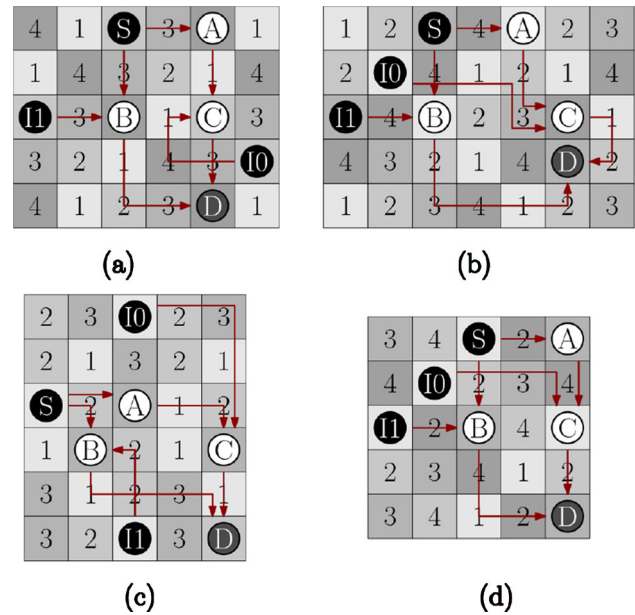


Fig. 10 The transposition of the sr-latch circuit, onto the clocking schemes. a BANCS. b RES. c USE

4 Results

The placement and routing problem (P&R) in Field-Coupled Nano computing is NP-Complete [31]. Therefore, one strategy is to find approximate solutions based on heuristics. Some important works are Fiction [32] and Ropper [11], which use the EPFL logic synthesis libraries [23]. Another initiative is The Torino Politecnico Nanotechnology (ToPoliNano) [25] framework, which generates layouts based on hardware description languages, performs simulation and logic verification.

For the comparison methodology, we have added a multilevel graph partitioning algorithm [16–18, 22]; as

Table 1 Area comparison of the chosen clocking schemes

Circuit	Gates	BANCS	USE	RES	2DDWave
SR-latch	4	64 Tiles ²	35 Tiles ²	35 Tiles ²	–
MUX 2:1	7	30 Tiles ²	30 Tiles ²	35 Tiles ²	25 Tiles ²
XOR gate	7	42 Tiles ²	63 Tiles ²	64 Tiles ²	64 Tiles ²
Decoder	8	30 Tiles ²	30 Tiles ²	25 Tiles ²	25 Tiles ²
Parity generator	13	180 Tiles ²	112 Tiles ²	99 Tiles ²	64 Tiles ²
1-Bit adder	16	156 Tiles ²	130 Tiles ²	289 Tiles ²	228 Tiles ²
Parity checker	19	180 Tiles ²	176 Tiles ²	225 Tiles ²	120 Tiles ²

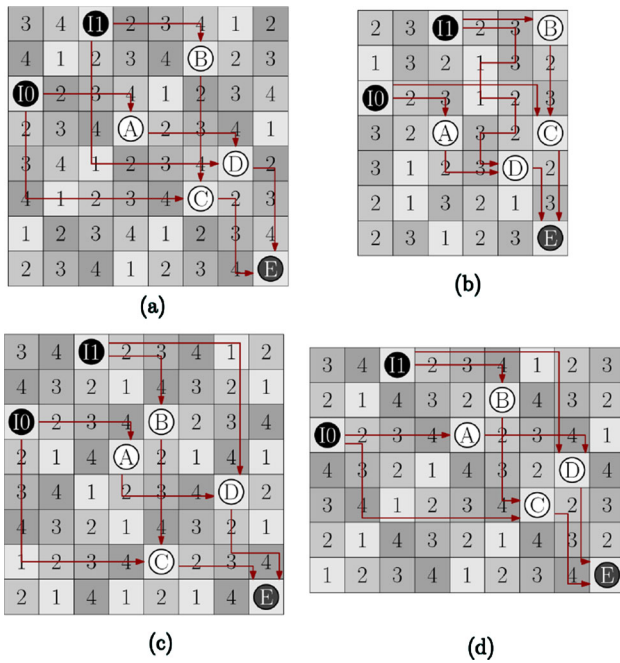


Fig. 11 The transposition of the 2:1 multiplexer, onto the clocking schemes. **a** USE. **b** RES. **c** BANCS. **d** 4-Phase 2DDWave

shown in Fig. 8. We generate several hypergraphs, indexed from 0 to n , with a graph maximal matching algorithm [15] as criteria to collapse adjacent graph vertices. The algorithm repeats this process until it creates a hypergraph composed of a single vertex. For the second phase of our strategy, we perform the placement of the hypergraph indexed by n , followed by on-grid uncoarsening for layout expansion. The algorithm generates the final layout when the process expands the original base graph G_0 . Area overhead translates to the number of generated hypergraphs, which the algorithm should minimize. We present the coarsening step in Algorithm 1 and the uncoarsening step in Algorithm 2.

Algorithm 1 Coarsening a graph with a multilevel graph partitioning technique.

- 1: **procedure** COARSENING(*root*, *get_adjacent*, *matching*, *collapse*)
- 2: *edges* \leftarrow *matching*(*root*, *get_adjacent*)
- 3: $G_i \leftarrow$ *collapse*(*edges*)
- 4: **return** G_i
 \rightarrow Return the hypergraph just created.
- 5: **end procedure**

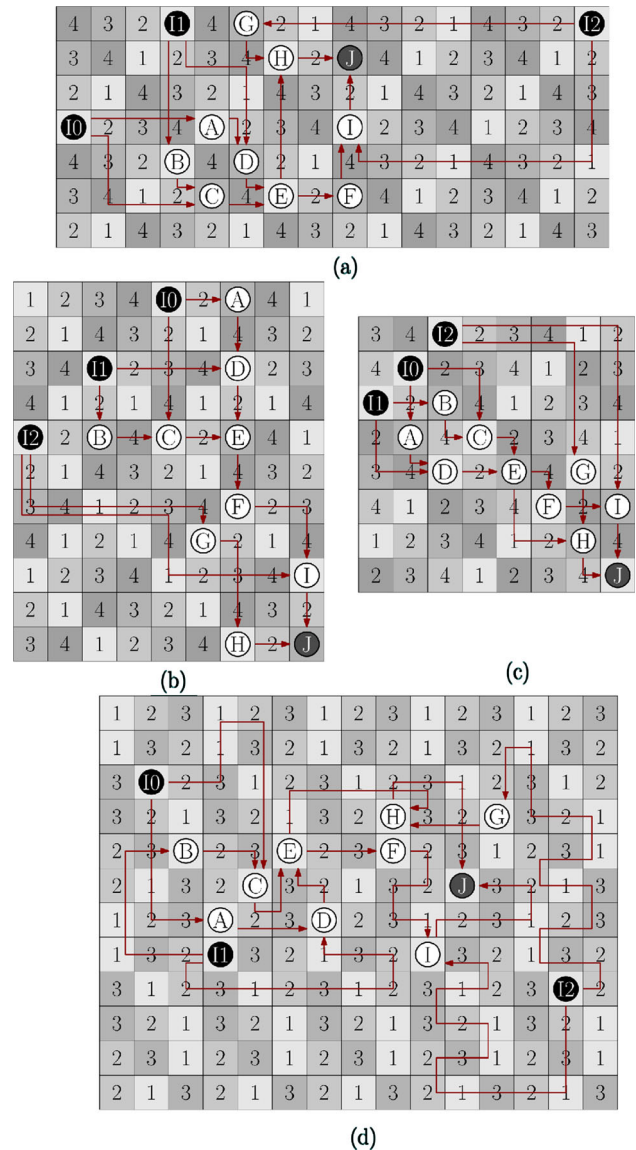


Fig. 12 The transposition of the XOR logic gate, onto the clocking schemes. **a** 2DDWave. **b** BANCS. **c** RES. **d** USE

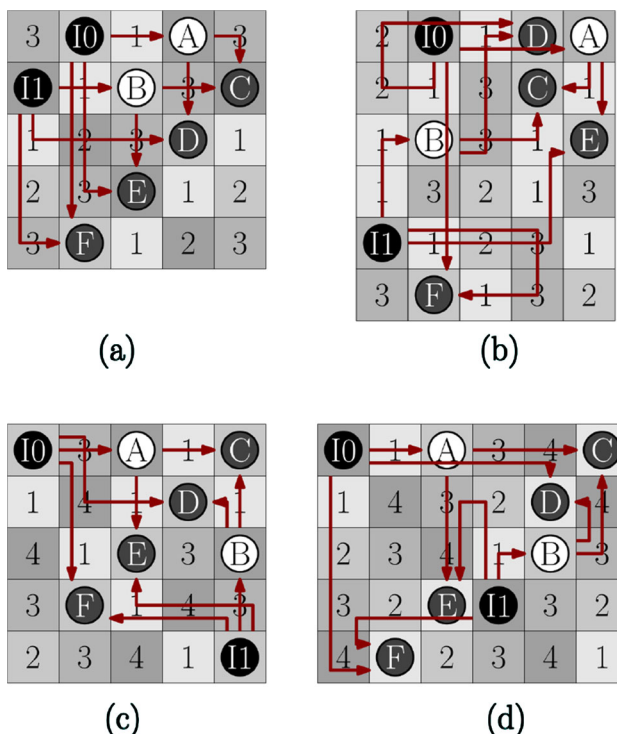


Fig. 13 The transposition of the parity generator circuit, onto the clocking schemes. **a** USE. **b** RES. **c** 4-Phase 2DDWave. **d** BANCS

Algorithm 2 Uncoarsening phase with a strategy based on the multilevel graph partitioning technique.

```

1: front ← graph_queue.front()
2: back ← graph_queue.back()
3: procedure UNCOARSENING(graph_queue, place, route,
    expand_grid)
4:   if graph_queue.front() == front then
5:     place(front, 0, 0); graph_queue.pop_front()
6:   else
7:     G1 ← graph_queue.front(); graph_queue.pop_front()
8:     G2 ← graph_queue.front()
9:     Try
10:    place(G1, G2)
11:    route(G1, G2)
12:   Catch
13:    expand_grid(G1, G2)
14:    place(G1, G2)
15:    route(G1, G2)
16:   EndTry
17:   end if
18: end procedure
    
```

We show a comparison of BANCS with the robust efficient and scalable (RES) clocking scheme [12], the universal scalable and efficient (USE) clocking scheme [4], and the two dimensional diagonal wave (2DDWave) clocking scheme [27, 28]. RES, USE, and 2DDWave specifically support QCA technology. We have added the design constraints for the layout where a tile can have up to three wires or a vertex and a wire or just one vertex. These assumptions are realistic considered recent NML technology advances, where multilayer crossings have been proposed [21].

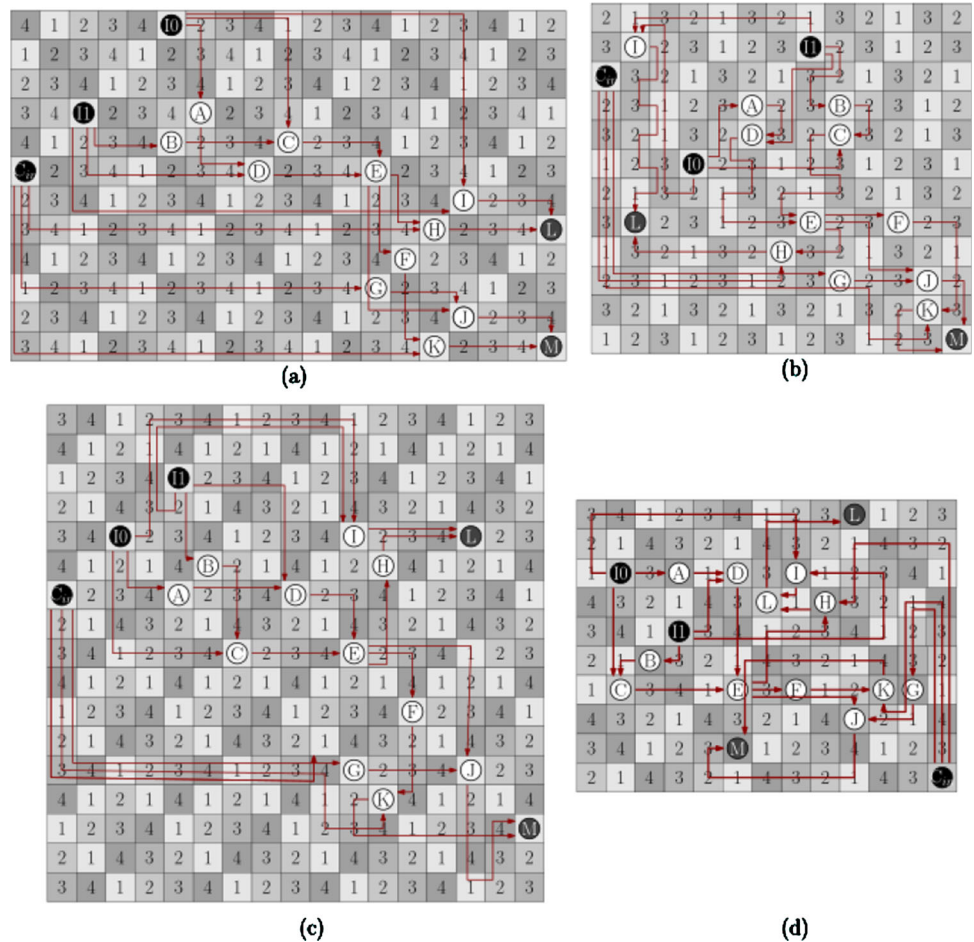
In Fig. 9, we present the seven circuits for this comparison. The circuits are a SR-Latch [Fig. 9(a)], 2:1 multiplexer [Fig. 9(b)], an XOR gate [Fig. 9(c)], a parity generator [Fig. 9(d)], a 1-bit full adder [Fig. 9(e)], a decoder circuit [Fig. 9(f)], a parity checker [Fig. 9(g)]. The first column of Table 1 shows the circuit name, followed by the number of logic gates. The last four columns refer to the final area after the process of placement and routing onto the clocking schemes above.

Figure 10 shows the SR-Latch circuit layouts. The P&R process was not possible to perform using our algorithm due to its sequential nature. Therefore we have used an *ad hoc* methodology to perform the P&R. The final area of the circuit has the lesser overhead in the BANCS clocking scheme, followed by RES and finally the USE clocking scheme. The 2DDWave clocking scheme does not support the feedback path, thus making it impossible to perform the SR-Latch P&R.

Figure 11 shows the layouts for the 2:1 multiplexer. As expected for a circuit with a small number of logic gates, the results are similar across the tested clocking schemes, where RES imposed the most significant area overhead, the 2DDWave clocking scheme presented the minimum area overhead for the mapping of this circuit.

Figure 12 shows the layouts for the XOR logic gate. In this case, the RES and 2DDWave clocking schemes imposed the most considerable area overhead for the mapping of the circuit. The BANCS clocking

Fig. 14 The transposition of the decoder circuit, onto the clocking schemes. **a** 4-Phase 2DDWave. **b** BANCS. **c** RES. **d** USE



scheme achieves the best area compaction, followed by the USE clocking scheme with a difference of roughly 33.3%.

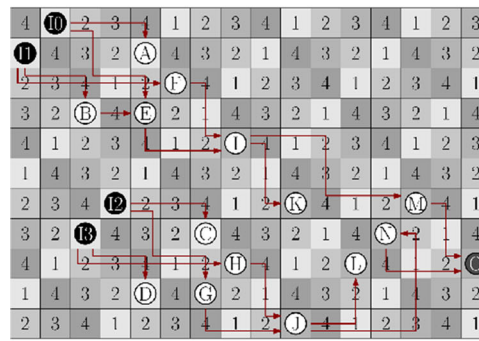
Figure 13 shows the layouts for the parity generator. This circuit has ten logic gates and three inputs. As shown in Fig. 13(a), the circuit mapping does not always tend to a regular quadrilateral geometry. The average area overhead imposed for all circuits had a significant increase of 48.8%, in comparison to the previous XOR logic gate circuit. BANCS imposes the most significant area overhead, and the 2DDWave clocking scheme achieved the best area compaction.

Figure 14 shows the result for the decoder circuit, the 2DDWave and the RES clocking schemes achieve a tie in terms of area overhead. Whereas the difference is not substantial when compared with the other two clocking schemes, BANCS and USE have a 16% increase.

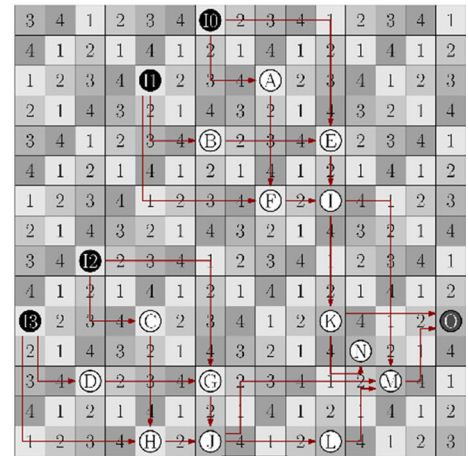
Figure 15 shows the final layouts for the 1-bit ripple carry adder circuit. In this case, the USE clocking scheme delivers the best area compaction. The circuit grows around 13.8% when compared to the previous circuit, and with the increase of three logic gates. BANCS presents a layout 16.6% larger than USE, whereas the 2DDWave clocking scheme a 43% overhead and, RES a 55% overhead.

Figure 16 shows the mapping for the parity checker circuit, which has six additional logic gates in contrast to the parity generator circuit. Here the BANCS clocking scheme has its area dimensions unfazed in comparison with the parity generator circuit, the RES clocking scheme imposes the most considerable area overhead, and the 2DDWave clocking scheme offered the smallest area dimensions for the circuit.

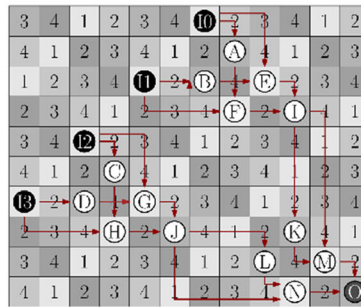
Fig. 15 The transposition of the 1-bit ripple carry adder circuit, onto the clocking schemes. **a** 2DDWave. **b** BANCS. **c** RES. **d** USE



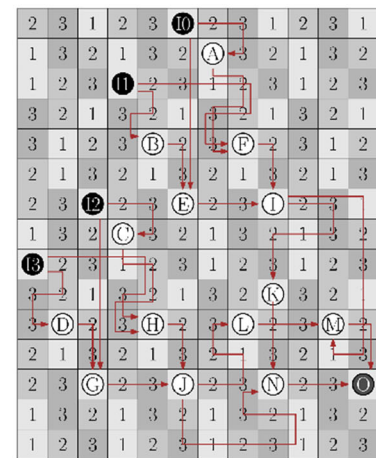
(a)



(b)



(c)



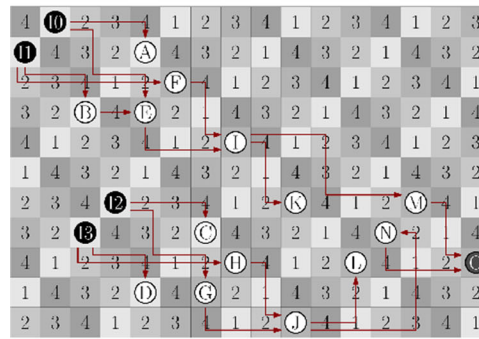
(d)

To show an example of the synthesis process, Fig. 17 shows the layouts for the presented clocking schemes. The area remains the same, the main challenges with the process are inherited technology structures.

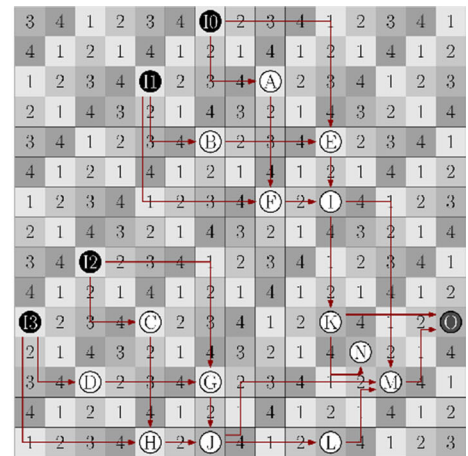
Our tests have shown that the 2DDWave clocking scheme scaled with more area compaction efficiency for our algorithm, the USE and RES clocking schemes achieved similar results, interchangeably, and the BANCS clocking scheme resulted in a higher overhead for the parity generator circuit and kept the compaction unfazed when mapping the parity checker circuit.

Although the 2DDWave presented better outcomes for the chosen combinational circuits, allowing three and four phases layouts, it does not support sequential circuits as USE, RES, and BANCS. Also, the BANCS clocking scheme supports a three phase layout and sequential circuits.

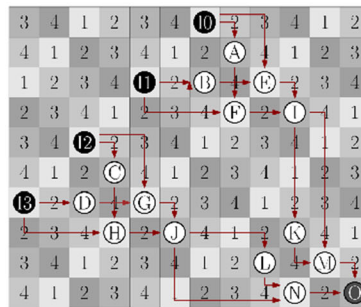
Fig. 16 The transposition of the parity checker circuit, onto the clocking schemes. **a** USE. **b** RES. **c** 4-Phase 2DDWave. **d** BANCS



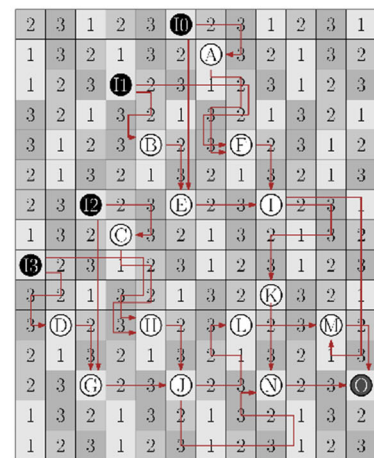
(a)



(b)



(c)



(d)

5 Conclusion

In this paper, we reviewed the basics of nanomagnetic logic technology. Moreover, we explained in more depth the concepts of a clocking scheme and how our design addresses the issues of thermal noise signal disruptions and scalability. Also, we discussed the details of the BANCS Clocking Scheme. To provide a better overview of the purpose of the design of efficient clocking scheme designs, we briefly explained the placement and routing problem in

field-coupled nano computing technologies. We also presented a discussion of the state-of-the-art frameworks and tools.

Finally, we compare the area compaction across three other clocking scheme designs proposed for the quantum-dot cellular automata technology. For the chosen circuits, the 2DDWave clocking scheme presents the best results, but it does not support sequential circuits.

With regards to our future work, we aim to explore further the usage of clocking scheme designs in the

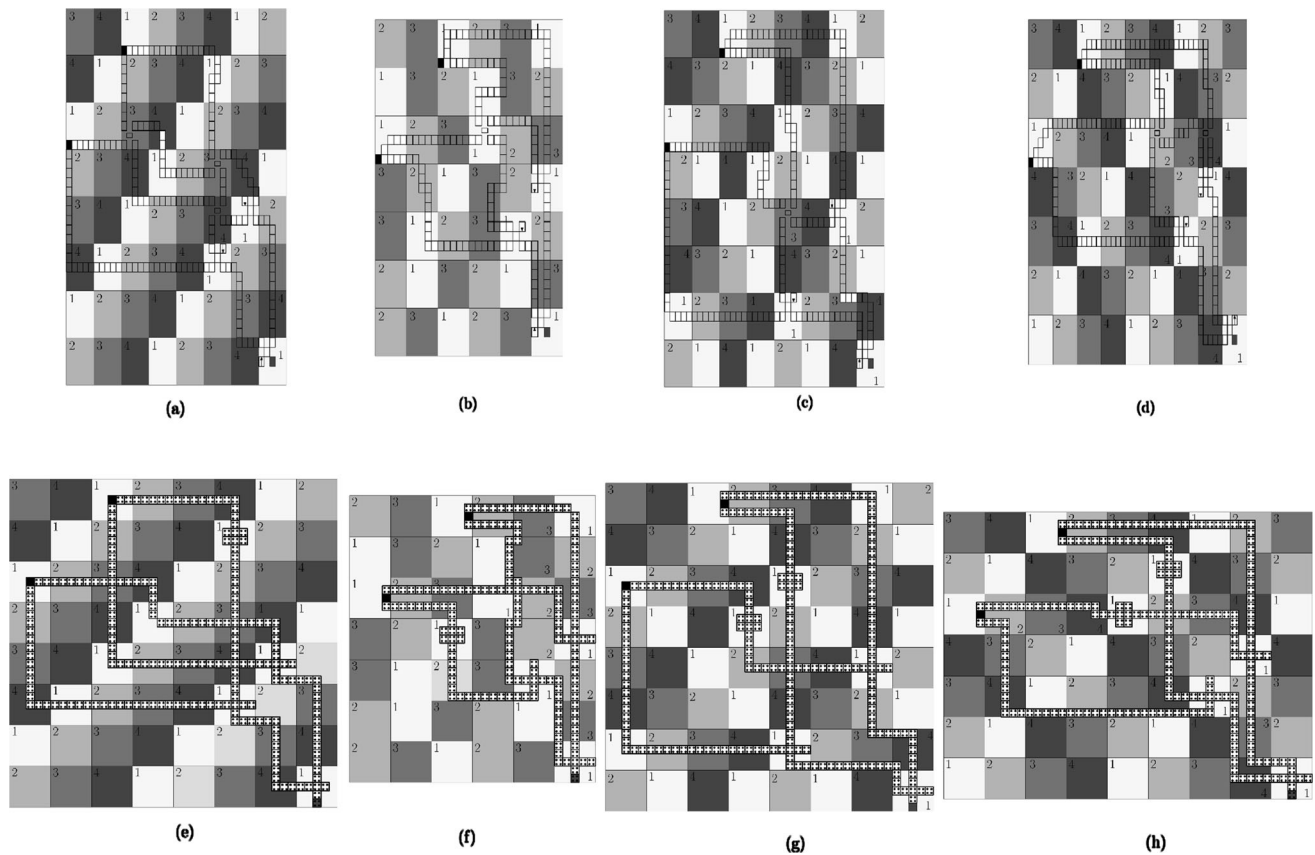


Fig. 17 The layouts of the xor logic gate for the NML technology, for the clocking schemes **a** 2DDWave. **b** BANCS. **c** RES. **d** USE. And for the QCA technology, with the usage of the clocking schemes **e** 2DDWave. **f** BANCS. **g** RES. **h** USE

nanomagnetic logic technology, also, design more efficient P&R algorithms.

References

- Anderson, N. G., & Bhanja, S. (2014). *Field-coupled nanocomputing* (1st ed., Vol. 8280). Berlin: Springer.
- Atulasimha, J., & Bandyopadhyay, S. (2010). Bennett clocking of nanomagnetic logic using multiferroic single-domain nanomagnets. *Applied Physics Letters*, *97*, 173105–173105. <https://doi.org/10.1063/1.3506690>.
- Bhowmik, D., You, L., & Salahuddin, S. (2013). Spin hall effect clocking of nanomagnetic logic without magnetic field. *Nature Nanotechnology*, <https://doi.org/10.1038/nnano.2013.241>.
- Campos, C. A. T., Marciano, A. L., Neto, O. P. V., & Torres, F. S. (2016). Use: a universal, scalable, and efficient clocking scheme for QCA. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *35*(3), 513–517.
- Cavin, R. K., Lugli, P., & Zhirnov, V. V. (2012). Science and engineering beyond Moore's law. In *Proceedings of the IEEE 100 (Special Centennial Issue)* (pp. 1720–1749).
- Chih-Tang, S. (1988). Evolution of the mos transistor—from conception to VLSI. *Proceedings of the IEEE*, *76*, 1280–1326. <https://doi.org/10.1109/5.16328>.
- Csaba, G., & Porod, W. (2010). Behavior of nanomagnet logic in the presence of thermal noise. In *14th International Workshop on Computational Electronics* (Vol. 75). <https://doi.org/10.1109/IWCE.2010.5677954>.
- Csaba, G., Porod, W., & Csurgay, Á. I. (2003). A computing architecture composed of field-coupled single domain nanomagnets clocked by magnetic field. *International Journal of Circuit Theory and Applications*, *31*, 67–82. <https://doi.org/10.1002/cta.226>.
- Fontes, G., Silva, P. A. R., Nacif, J. A. M., Neto, O. P. V., & Ferreira, R. (2018). Placement and routing by overlapping and merging QCA gates. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1–5). IEEE
- Formigoni, R. E., Vilela Neto, O. P., & Nacif, J. A. M. (2018). BANCS: Bidirectional alternating nanomagnetic clocking scheme. In *2018 31st symposium on integrated circuits and systems design (SBCCI)* (pp. 1–6).
- Formigoni, R. E., Ferreira, R. S., & Nacif, J. A. M. (2019) Ropper: A placement and routing framework for field-coupled nanotechnologies. In *32nd symposium on integrated circuits and systems design (SBCCI '19)*, August 26–30, 2019, Sao Paulo, Brazil, *10*(1145/3338852), 3339838.
- Goswami, M., Mondal, A., Mahalat, M. H., Sen, B., & Sikdar, B. K. (2019). An efficient clocking scheme for quantum-dot cellular automata. *International Journal of Electronics Letters*, <https://doi.org/10.1080/21681724.2019.1570551>.
- Graziano, M., Vacca, M., Chiolerio, A., & Zamboni, M. (2011). An NCL-HDL snake-clock-based magnetic QCA architecture. *IEEE Transactions on Nanotechnology*, *10*(5), 1141–1149. <https://doi.org/10.1109/TNANO.2011.2118229>.

14. Imre, A., Csaba, G., Ji, L., Orlov, A., Bernstein, G. H., & Porod, W. (2006). Majority logic gate for magnetic quantum-dot cellular automata. *Science*, *311*(5758), 205–208. <https://doi.org/10.1126/science.1120506>.
15. Israeli, A., & Itai, A. (1986). A fast and simple randomized parallel algorithm for maximal matching. *Information Processing Letters*, *22*(2), 77–80.
16. Karypis, G., & Kumar, V. (1995). Multilevel graph partitioning schemes. In *ICPP* (Vol. 3, pp. 113–122).
17. Karypis, G., & Kumar, V. (1996). Parallel multilevel graph partitioning. In *Proceedings of international conference on parallel processing* (pp. 314–319). IEEE.
18. Karypis, G., & Kumar, V. (1998). A parallel algorithm for multilevel graph partitioning and sparse matrix ordering. *Journal of Parallel and Distributed Computing*, *48*(1), 71–95.
19. Lent, C. S., & Tougaw, P. D. (1997). A device architecture for computing with quantum dots. *Proceedings of the IEEE*, *85*, 541–557. <https://doi.org/10.1109/5.573740>.
20. Niemier, M. T., Hu, X. S., Alam, M., Bernstein, G., Porod, W., Putney, M., & DeAngelis, J. (2007). Clocking structures and power analysis for nanomagnet-based logic devices. In *International symposium on low power electronics and design (ISLPED)*, 2007 (pp. 26–31). ACM/IEEE. <https://doi.org/10.1145/1283780.1283787>.
21. Santoro, G., Vacca, M., Bollo, M., Riente, F., Graziano, M., & Zamboni, M. (2018). Exploration of multilayer field-coupled nanomagnetic circuits. *Microelectronics Journal*, *79*, 46–56.
22. Schloegel, K., Karypis, G., & Kumar, V. (2003). *Graph partitioning for high-performance scientific simulations* (pp. 491–541). San Francisco, CA: Morgan Kaufmann Publishers Inc.
23. Soeken, M., Riener, H., Haaswijk, W., & De Micheli, G. (2018). The EPFL logic synthesis libraries. [arXiv:1805.05121](https://arxiv.org/abs/1805.05121)
24. Trindade, A., Ferreira, R., Nacif, J. A. M., Sales, D., & Neto, O. P. V. (2016). A placement and routing algorithm for quantum-dot cellular automata. In *2016 29th symposium on integrated circuits and systems design (SBCCI)* (pp. 1–6). IEEE.
25. Vacca, M., Frache, S., Graziano, M., Riente, F., Turvani, G., Ruo Roch, M., & Zamboni, M. (2014a). *Topolinano: Nanomagnet logic circuits design and simulation. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* (Vol. 8280, pp. 274–306). https://doi.org/10.1007/978-3-662-45908-9_12.
26. Vacca, M., Graziano, M., Chiolerio, A., Lamberti, A., Laurenti, M., Balma, D., et al. (2014b). *Electric clock for nanomagnet logic circuits* (pp. 73–110). Berlin: Springer. https://doi.org/10.1007/978-3-662-43722-3_5.
27. Vankamamidi, V., Ottavi, M., & Lombardi, F. (2006). Clocking and cell placement for QCA. In *2006 Sixth IEEE Conference on Nanotechnology* (Vol. 1, pp. 343–346). <https://doi.org/10.1109/NANO.2006.247647>.
28. Vankamamidi, V., Ottavi, M., & Lombardi, F. (2008). Two-dimensional schemes for clocking/timing of QCA circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *27*, 34–44. <https://doi.org/10.1109/TCAD.2007.907020>.
29. Varga, E., Liu, S., Niemier, M. T., Porod, W., Hu, X. S., Bernstein, G. H., et al. (2010). Experimental demonstration of fanout for nanomagnet logic. In *Device research conference (DRC)* (Vol. 2010, pp. 95–96). <https://doi.org/10.1109/DRC.2010.5551852>.
30. Varga, E., Csaba, G., Bernstein, G.H., & Porod, W. (2011). Implementation of a nanomagnetic full adder circuit. In *2011 11th IEEE Conference on Nanotechnology (IEEE-NANO)* (pp. 1244–1247). <https://doi.org/10.1109/NANO.2011.6144445>.
31. Walter, M., Wille, R., Große, D., Sill Torres, F., & Drechsler, R. (2019a). Placement and routing for tile-based field-coupled nanocomputing circuits is np-complete (research note). *ACM Journal on Emerging Technologies in Computing Systems*, *15*, 1–10. <https://doi.org/10.1145/3312661>.
32. Walter, M., Wille, R., Torres, F.S., Große, D., & Drechsler, R. (2019b). fiction: An open source framework for the design of field-coupled nanocomputing circuits.
33. Wanlass, F., & Sah, C. (1963). Nanowatt logic using field-effect metal-oxide semiconductor triodes. In *Solid-state circuits conference. Digest of technical papers. 1963 IEEE international* (pp. 1280–1326). <https://doi.org/10.1109/ISSCC.1963.1157450>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Ruan Evangelista Formigoni is a computer science undergraduate student at Universidade Federal de Viçosa, Brazil. His research interests include Electronic Design Automation and Field-Coupled Nanocomputing. He is a member of SBC.



Leandro Lázaro Araújo Vieira is a computer science undergraduate student at Universidade Federal de Viçosa, Brazil. His research interests include Electronic Design Automation, especially Nanomagnetic Logic.



Omar Paranaíba Vilela Neto Omar P. Vilela Neto is an Assistant Professor of Computer Science at the Universidade Federal de Minas Gerais (UFMG). He received his undergraduate in computer engineering, M.S., and Ph.D. degrees in electrical engineering from the Pontifícia Universidade Católica do Rio de Janeiro. His research interest are computational nanotechnology and nanocomputation.



Ricardo Ferreira was born in Belo Horizonte, Brazil, in 1969. He received the B.E. degree in Physics and M.Sc. in Computer Science from the Federal University of Minas Gerais, Brazil, in 1991 and 1994, respectively, and the Ph.D degree in Applied Sciences (Microelectronics) from the Universite Catholique de Louvain, Louvain-la-Neuve, Belgium, in 1999. In 1992, he joined the Department of Computer Science, Federal University

of Vicoso, as a Lecturer, and currently he is an Associate Professor. His current research interests include reconfigurable computing, FPGAs, GPUs, placement and routing, embedded systems, and run-time approaches.



José Augusto M. Nacif is a Professor in the Science and Technology Institute, UFV-Florestal Campus, Universidade Federal de Viçosa. He received his B.S. (2001) from PUC/MG and his M.S. (2004) and Ph.D. (2011) degrees in Computer Science from the Universidade Federal de Minas Gerais (UFMG). His research interests include Internet of Things, Reconfigurable Computing, and Electronic Design Automation. Dr. Nacif is member of IEEE, SBC, and

SBMICRO.