Spin Diode Network Synthesis using Functional Composition

Mayler G. A. Martins[1], Felipe S. Marranghello[1], Joseph S. Friedman[2], Alan V. Sahakian[2], Renato P. Ribas[1], Andre I. Reis[1]
PGMICRO[1] / Electrical Engineering & Computer Science[2]
UFRGS[1] / Northwestern University[2]
Porto Alegre, RS, Brazil[1] / Evanston, IL, USA[2]
{mgamartins, fsmarranghello, rpribas, andreis}@inf.ufrgs.br[1] {jf@u, sahakian@eecs}.northwestern.edu[2]

***Abstract* — This paper proposes an algorithm to synthesize combinational circuits based on spin diode logic technology. Spin diode is a magnetoresistive semiconductor heterojunction device which allows for a binary current based logic. The proposed algorithm takes the advantages of the functional composition (FC) approach to obtain fanout free network implementations with the minimum number of spin diodes. Experimental results have shown that the new proposal obtains better results when compared to the state-of-the-art algorithms that focus on traditional CMOS technology adapted to this new approach.**

Keywords — Digital circuit, logic synthesis, spin diode, functional composition, CAD.

I. INTRODUCTION

Continuous scaling of MOS transistor is the key strategy for improving circuit performance. However, as MOS dimensions shrink, manufacturing imperfections and quantum effects become more critical and threaten to stop the CMOS scaling [1]. As a result, much effort has been done in order to develop new devices that may allow further progress in computation capability [2-26]. Among these emerging technologies are carbon nanotubes and related graphene structures [3-5], single electron transistors (SET) [6-8], nanowires [9-10], quantum cellular automata (QCA) [11-14], resonant tunneling diodes (RTD) [15-17], and devices using electron spin [18-26].

For any of these emerging technologies be successful, it must present some characteristics that represent improvements over traditional CMOS transistors [27]. These can be related to higher frequency operation, smaller power consumption, smaller area, and so on. Additionally, it is welcome that the knowledge used in designing and fabricating CMOS circuits can be somehow adapted to the new technology to make the conversion process easier [27].

Recently, the magnetoresistive spin-diode has been proposed [20,21], and the possibility to implement a logic family using only this kind of device has been demonstrated [18]. It is the first diode based logic family that does not require any transistors. This new device is expected to improve the circuit frequency of operation while reducing area and power consumption.

This paper proposes an efficient synthesis algorithm for combinational circuits using the spin-diode family. Even though many works discuss synthesis algorithms for CMOS circuits, the spin-diode logic family presents some particularities that compromise the quality of results when considering algorithms developed for CMOS technology. The proposed algorithm is able to exploit the unique aspects of the spin diode logic family to obtain circuits with smaller numbers of devices. Since the standard cell methodology is widely adopted in CMOS design, the proposed algorithm guarantees that any synthesized circuit can be used in a similar flow.

The rest of the paper is organized as follows. Section II presents a background on the operation of the spin diode and on the utilization of this device to create different logic functions. Section III details the proposed algorithm, whereas Section IV compares the result obtained to the state-of-the-art synthesis algorithms for CMOS technology adapted to the spin diode logic family. Finally, Section V outlines the conclusions.

II. SPIN DIODE LOGIC

The recent invention of the magnetoresistive spin-diode allows for the creation of a complete logic family composed solely of spin-diodes [18,19]. The diodes are used as switches by manipulating the magnetoresistance with control currents that generate magnetic fields through the junction. With this device structure, basis logic elements and complex circuits consist of as few as 10% of the devices required in their conventional CMOS counterparts. Spin-diode logic circuits use only spin-diodes for computing, while no other devices are required. As MOS scaling reaches its inherent limits, such spin-diode logic family is an intriguing potential replacement for CMOS technology due to its material characteristics and compact circuits.

A. Device Characteristics

The spin-diode is a magnetoresistive p-n junction, i.e., a diode with a resistance that is affected by magnetic field. Spin-diodes of the type shown in Fig. 1 have been fabricated by doping a III-V semiconductor heterojunction with an element such as Mn that has a strong interaction with a magnetic field [20,21]. The spin-diode acts as a conventional diode in the presence of zero or low magnetic fields, with a high ratio of forward current to reverse current. However, when a magnetic field is applied across the junction, spin-dependent conduction results in decreased charge flow across the junction [22].

978-1-4799-1132-5/13/$31.00 ©2013 IEEE

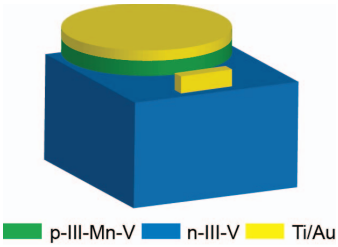


Fig. 1. III-V/III-Mn-V heterojunction diode [18].

Recent experiments have demonstrated strong magnetoresistive effects in InMnAs diodes [20,21]. As the measurements in Fig. 2 make clear, the current (I) decreases significantly as the magnetic field is increased. It is, therefore, possible to define two distinct states: a resistive state with current I_{off} in the presence of a magnetic field, and a conductive state with current I_{on} in its absence. When the voltage is increased, the I_{on}/I_{off} ratio increases in these devices. A high ratio is useful as it supports differentiation between two digital states. It is expected that increasing the Mn content of the heterojunction will lead to increased sensitivity to magnetic fields [23]. Additionally, it should be noted that while many magnetic effects can only be observed at low temperatures, this experimental data was measured at room temperature, making it a potential practical candidate to replace Si MOSFETs.

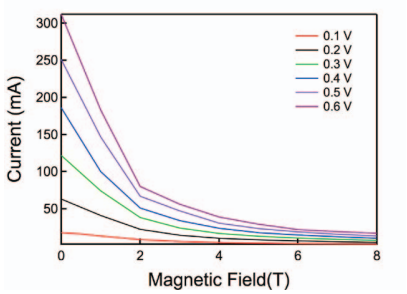


Fig. 2. Spin-diode current as a function of magnetic field at various voltage biases [21].

B. Logic Family Structure

The spin-diode logic family [18] is constructed by connecting the n- and p-regions of spin-diodes directly to ground, and the circuit high voltage, V_{DD}, respectively. Each spin-diode is therefore forward biased. The wires making these connections are routed next to other diodes to function as control wires. These control wires are used as inputs that produce magnetic fields, switching the diodes between their conductive and resistive states. The output current through the spin-diode junctions can therefore be switched between I_{on} and I_{off}. As the control wires affect the spin-diode states without making an electrical connection, the output current of a spin-diode can be used as the input of a large number of other spin-diodes.

Unlike CMOS, in which the state variable is voltage, the state variable in spin-diode logic is current. A binary '1' is defined as the "high" I_{on} current produced by the spin-diode conductive state, and a binary '0' is defined as the "low" I_{off} current produced by the resistive state. As mentioned previously, the wires that form these connections are routed to control the magnetoresistance of other spin-diodes. A spin-diode propagating a '1' so creates a magnetic field in another spin-diode, whereas a spin-diode propagating a '0' does not. As the spin-diodes are connected directly between V_{DD} and ground, there is a constant flow of current that is minimally affected by frequency.

C. Basis Logic Gates

Every spin-diode logic [18] basis gate is composed of at most a single diode: inverter, NOR gate, exclusive-NOR (XNOR) gate, and OR gate. In each of these configurations, the positive terminal of the diode is connected to V_{DD}, while the negative terminal is routed through the circuit before eventually being grounded. Therefore, in the absence of current through the various control wires, each of these diodes propagates a '1', and the configuration of the control wires dictates the logic functionof each gate. These basis logic gates are sufficient to perform any complex logic and are significantly more compact than conventional CMOS.

1) Inverter Gate

An inverter, the simplest gate, is shown in Fig. 3. The positive terminal of the spin-diode is connected to V_{DD} and the negative terminal is connected to ground through the output loop. The input current I_A is routed alongside the diode, and induces a field proportional to its current. If I_A is a '1', it creates a large magnetic field, thereby reducing the current through the diode, and causing the output current I_O to propagate a '0'. If I_A is a '0', it does not create a sufficient magnetic field through the junction, and a '1' is propagated.

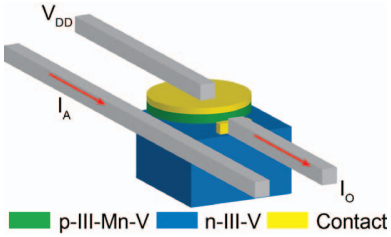


Fig. 3. Spin-diode logic inverter gate [18].

2) NOR Gate

The addition of a second input current I_B results in a NOR gate, shown in Fig. 4(a). In this arrangement, the presence of a current in either of the two inputs I_A or I_B results in a magnetic field through the diode oriented in the same direction. This magnetic field activates the spin-diode magnetoresistance, forcing the diode into the resistive state and attenuating the current. In the case of high currents on both inputs, the output current is doubly suppressed. Therefore, if at least one of the two inputs is a '1', the output propagates a '0'. Otherwise, there is no magnetic field through the spin-diode, and the output is a '1'.

3) Exclusive-NOR Gate

The exclusive-NOR (XNOR) gate is identical to the NOR gate in terms of device structure, but its relative placement within a circuit produces a different function. As shown in Fig. 4(b), the two input currents flow in the same direction, causing each current to create a magnetic field through the diode in the

opposite direction. Therefore, if both inputs are '1', there is no net magnetic field through the diode and a large output current flows, propagating a '1'.

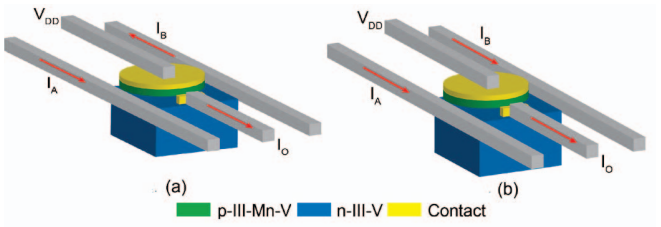


Fig. 4. Spin-diode logic NOR (a) / XNOR (b) gate [18].

4) OR Gate

As the magnitude of the currents defines the digital states, an OR gate is constructed simply by merging two wires (therefore, no symbol is needed). As shown in Fig. 5, the output current I_O is equal to the sum of the two input currents. By placing an OR gate as an input to another gate, a variety of logic functions with more than two inputs can be implemented.

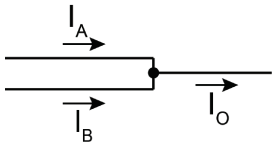


Fig. 5. Current summation wired-OR gate for use in spin-diode logic [18].

III. Synthesizing Spin Diode Networks

As presented in Section II, the basic logic gates using magnetoresistive spin diodes, hereafter referred simply as diodes, are the XNOR/NOR/INV gates. The OR is a special case, where two wires are connected, avoiding the use of diodes. Unfortunately, the incorrect use of the OR "gate" in the circuit can lead to logic errors. This happens because the input signals are lost and cannot be used after the OR operation. To the best knowledge of the authors, the only approach that manipulate these kind of gates in order to synthesize a Boolean function is presented in [28], but in this approach are used other basic gates as NAND/AND/XOR, not trivial in the spin diode technology, and does not guarantee optimality. It is, therefore, important to develop an algorithm that is able to synthesize Boolean functions using the basic gates of the spin diode technology, and use these synthesized functions as cells in a standard cell design flow. A new algorithm to synthesize any Boolean function to a fanout free network composed by spin diodes is introduced in the following. In order to synthesize these functions, a data structure that represents diode networks is needed and the electrical characteristics of the OR "gate" needs to be taken into account in order to synthesize a diode network.

A. Diode Representation

Factoring Boolean functions is one of the basic operations in algorithmic logic synthesis. Factoring is usually restricted to the use of INV (!), AND (*), OR (+) operators, even if it is known that the use of XOR ($\oplus$) operator can introduce more compact representations in terms of literals. In order to represent an expression using spin diode gates, this paper uses the notation to represent the NOR/XNOR operations as follows: the NOR is represented as ($-$) operator, and XNOR is represented as ($\ominus$) operator.

B. Logic Fanout

In a circuit, the same signal can be used by several different gates. In a traditional CMOS implementation, all receiving gates can be placed in "parallel" as shown in Fig 6(a). However, as the number of receiving gates increases the load capacitance of the driver gate also increases. This leads to a slower signal transition that can prejudice the circuit timing performance. In a similar manner, the number of transistors, as well as their size, controlled by the signal in a single receiving gate also impacts the output capacitance of the driver gate. CMOS gates should be synthesized considering these factors.

If the spin diode logic is targeted the signal distribution is different. Since the spin diode logic is current based, the same signal can drive as many gates as required without the concern about increased load capacitance [19]. However, the receiving gates cannot be placed in parallel because the input current for each gate is reduced which leads to a logical failure [19]. In the spin diode logic family, the receiving gates should be connected as shown in Fig. 6(b).

This different behavior between traditional CMOS and the spin diode logic must be considered by the synthesis algorithm. In short, it is not important to consider how many gates use the signal but care must be taken on how these gates are placed to avoid current splitting.

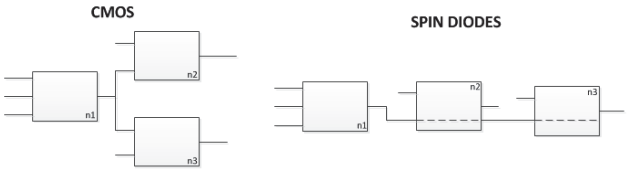


Fig. 6. Fanout in CMOS (a) and in spin diode technology (b).

C. Diode Electrical Feasibility

As discussed earlier, the OR "gate" misuse could harm the circuit logic integrity. Therefore, special precautions are needed in order to avoid logic errors. An analysis of the electrical feasibility of three possible implementations of a 2-input AND gate is shown in Fig. 7.

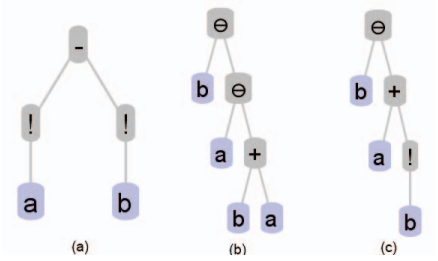


Fig. 7. Three possible implementations of a 2-input AND gate using spin diodes: (a) naïve approach; (b) two diodes version with short-circuit between 'a' and 'b'; and (c) two diodes version with a possible short-circuit in 'a'.

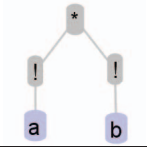
The implementation in Fig 7(a) is the naïve one. Nevertheless, this implementation is one of the most robust,

since there are no possible short circuits. Analyzing the implementation in Fig 7(b) is achieved using Boolean algebra manipulation techniques. This implementation is interesting, since it costs two diodes, but is less robust. The 'a' and 'b' input currents are combined by the (a+b) node, preventing the use of the ‘a’ and ‘b’ inputs in other gates. The implementation in Fig 7(c) is achieved in a similar way of the implementation (b), and also needs two spin diodes. But this circuit has a limitation of fanin 1 for the input 'a', since this input has a direct OR connection, avoiding the posterior use of ‘a’ information. This issue can be prevented if the OR gate is forbidden to appear in the input nodes of logic circuits.

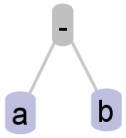
D. Using Functional Composition Principles

The Boolean factoring algorithm using functional composition (FC) approach is able to synthesize Boolean functions as expressions, being also able to consider multi objective goals [30]. As the FC approach is flexible, modifications can be performed to adapt the algorithm to synthesize a spin diode network. One of the most important changes is the bonded-pair representation. In this algorithm, the bonded-pair representation is composed by the tuple {function, spin diode logic tree}. The function can be represented as an integer or the root node of a binary decision diagram (BDD). Storing the truth table data as a computer word or an array of words, basic Boolean operations can be done in constant time by parallel operation over their truth tables. The spin diode network can be represented by a subject graph or an expression representing a spin diode logic tree. Fig. 8 shows the 2-input NOR function represented as a bonded-pair representation used in [29], the new bonded-pair representation used in this paper and the truth table of the NOR function. The integers in Fig. 8(a) and in Fig. 8(b) are the output *f* depicted in Fig. 8(c), where the most significant bit is in the leftmost digit.

(a)

Function	Expression Logic Tree
0001_2	

(b)

Function	Spin Diode Logic Tree
0001_2	

(c)

a	b	*f*
0	0	1
0	1	0
1	0	0
1	1	0

Fig. 8. Different bonded-pair representations using functions represented as integers: (a) bonded-pair representation {function, expression logic tree} used in [29]; (b) bonded-pair representation {function, spin diode logic tree} used in this paper; and (c) truth table of 2-input NOR function.

The second most important modification is related to the partial order. In the traditional CMOS approach, the combination of two functions also increases the cost of the resulting function, regardless of how the functions are combined. When using the spin diode logic, this observation no longer holds because the OR operator has cost zero. The pseudo-code used to generate a set containing **N** diodes is shown in Fig. 9. The algorithm has as input the set **L** with all functions created with maximum cost of N-1, classified by the number of diodes needed in order to implement such functions. The algorithm output is the set of functions **S**, implemented with **N** diodes. The **CREATE_INVERTED_FUNCTIONS** method negates all functions having N-1 diodes. Negating a function implies adding exactly one diode to the network, which means that the diode cost increases by one. The method **COMBINE_SETS** generates new functions using NOR, XOR or OR associations between two functions in the set L. The NOR, XOR and OR associations are performed by the methods **NOR_ASSOCIATION, XNOR_ASSOCIATION** and **OR_ASSOCIATION,** respectively**.** Indexes **I** and **J** are chosen so that the cost after association is **N**. For instance, a new set **S** containing functions with N=4 diodes, the indexes [I,J] used in COMBINE_SETS method for the association types for NOR_ASSOCIATION and XNOR_ASSOCIATION are {[0,3];[1,2]} and for association type OR_ASSOCIATION the indexes are {[1,3];[2,2]}.

```
    ASSOCIATE_FUNCTIONS(L, N)
 1  create a new empty set of functions S
 2  INV_SET :=CREATE_INVERTED_FUNCTIONS(N-1);
 3  add INV_SET in S
 4  for Index I:=0 to (N/2)-1 do
 5     J := N-I-1;
 6     NOR_SET := COMBINE_SETS(L, I, J, NOR_ASSOCIATION)
 7     add NOR_SET in S
 8     XNOR_SET := COMBINE_SETS(L, I, J, XNOR_ASSOCIATION)
 9      add XNOR_SET in S
10  end for
11  for Index I:=1 to (N/2) do
12     J := N-I;
13     OR_SET := COMBINE_SETS(L, I, J, OR_ASSOCIATION)
14     add OR_SET in S
15  end for
16  return S
```

Fig. 9. Pseudo code for the ASSOCIATE_FUNCTIONS method.

E. Library Generation

Optimal fanout free spin diode network implementations containing all functions with up to four variables can be easily generated with a simple adaptation of the approach presented in [29]. This approach is interesting for a mapping point-of-view, since only one execution is necessary to generate a library, and the results are saved in a file for posterior use, avoiding the matching step. The stop criterion of this algorithm is when all functions with up to the number of variables desired are generated.

The first library generation step is to generate the constants, stored in a separate set. The set used to store bonded-pairs is called a *bucket*. The direct variables are stored in bucket 0, since they do not have implementation costs. The remaining functions are generated combining existing functions and allowing the cost of implementation to increase unitary. When all implementations with a cost up to *n* are generated, the algorithm can generate functions with cost equal to*n+1*. Once

all desirable functions are implemented, the algorithm stops. As the algorithm is a bottom-up approach, it uses dynamic programming and initializes with all minimal cost functions, all implementations are guaranteed to have minimal costs (*i.e.,* diode count). It has been proved in [30].

In order to illustrate the diode network generation in a simple way, the generation of all 2-input functions is shown in Fig. 10. After the allocation of constants, all variables are allocated in the bucket 0. After the first combination, four functions are generated. These functions are allocated in the bucket 1, since they need only one diode. Continuing the combinations, five implementations with two diodes and three implementations with three diodes are generated. It achieves the total number of functions with up to two inputs (2+2+4+5+3 = 16).

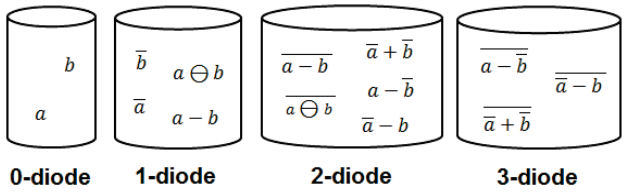


Fig. 10. : Buckets containing all 2-variable functions.

IV. Experimental Results

In this section, it is evaluated the quality of the spin logic generation algorithm, hereby called FC-SPIN, compared to the state-of-the-art ABC [31] and a commercial tool. The platform used was an Intel Core i5 processor with 2GB main memory.

The first step was the generation of an optimal fanout free spin diode networks for all functions with up to four inputs, stored in a look-up table (LUT). In order to analyze the impact of considering the OR gate, two LUTs were generated. The first (FC-SPIN) considers the complete set of operators {INV, NOR, XNOR, OR}, taking into account the OR gate electrical characteristics in the network, as discussed in Section III, allowing OR gates only in internal nodes. The second (FC-SPIN w/o OR) is synthesized using a partial set of operators {INV, NOR, XNOR}.

The execution time to generate each LUT was about one minute. The file with the LUT occupies less than 3 MB, and the LUT loaded in memory occupies less than 50 MB, so making it feasible in a logic synthesis tool. It is worth to mention that all 4-input functions were synthesized using at most nine spin diodes when considering the OR gate, and ten gates when excluding the OR gate.

In order to analyze the synthesis of spin diode networks using two logic synthesis tools, ABC and a commercial tool were applied. These tools were used to perform the technology mapping in a set of 3982 representative 4-input functions, grouped into permutation equivalent classes (4-P set). The gates present in the library were: INV (1), AND (3), NAND2 (2), OR2 (2), NOR2 (1), XOR2 (2), and XOR2 (1). The gate cost is defined as the required number of spin diodes necessary to implement such gate, shown within the parenthesis.. Since these tools cannot evaluate when it is safe to use the OR operator, this gate is provided only in the version NOR+INV to ensure a correct implementation.

The scripts used in the ABC tool in order to perform the technology mapping used primarily AIGs with choice approach or a supergate library approach, having area reduction as main objective.

The results of the technology mapping performed in ABC and the commercial tool for each function is compared with the FC-SPIN algorithm, and a histogram is shown in Fig. 11. Positive (negative) numbers indicate a worse (better) implementation based on spin diode count, compared to FC-SPIN. It is noted that the FC-SPIN only guarantees a minimal fanout free network. In this sense, there are a few cases that these tools can exploit logical sharing to generate a network with less spin diodes..

Another important consideration is the impact of the OR gate in the diode count, generating an implementation overhead in some functions with up to three diodes. On the other hand, the ABC and the commercial tools generate a considerable number of functions with more than five diodes compared to the optimal implementation, representing 1250 (31.39%) and 2022 (50.78%) functions, respectively.

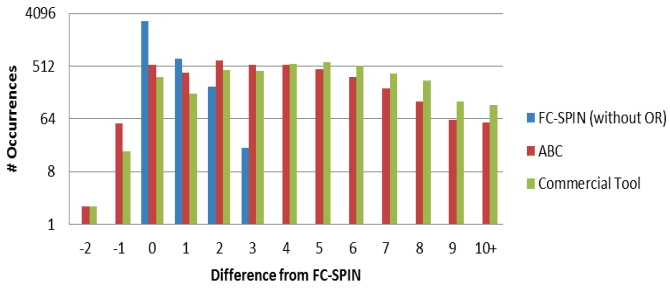


Fig. 11. Histogram of the differences from each algorithm compared to FC-SPIN.

An analysis of the overall results of the algorithms is shown in Fig. 12, which shows the total diode count for the 4-P set. The FC-SPIN w/o OR has an overhead of 4.63% against the FC-SPIN. In order to have a fair comparison, the ABC and the commercial tools are compared to the FC-SPIN (without OR), having an increase of 44.54% and 61.51% in diode count, respectively. These results demonstrate that logic synthesis tools are very far from optimality, mainly when considering technologies that do not have AND/OR/INV as basic gates. This considerable reduction allows generating very compact gates that will reduce the final area of circuits.

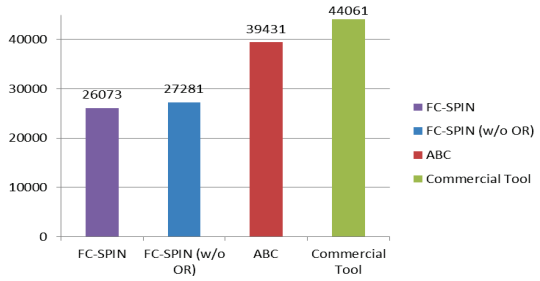


Fig. 12. Overall spin diode count for the 4-P set.

V. CONCLUSIONS

In this paper, a novel algorithm to synthesize spin diode networks using the functional composition paradigm is presented. Spin diode logic allows for a full range of logic functions to be built, making it a potential replacement for the CMOS technology. The proposed algorithm is able to generate an optimal lookup table with minimal fanout free implementations of all functions with up to 4 inputs. Experiments have shown that the proposed algorithm is able to reduce the spin diode count in 33.8% and 40.8%, when compared to ABC and commercial logic synthesis tools, respectively.

ACKNOWLEDGMENTS

Research partially supported by Brazilian funding agencies CAPES, CNPq and FAPERGS, under grant 11/2053-9 (Pronem).

REFERENCES

[1] D. J. Frank, R. H. Dennard, E. Nowak, P. M. Solomon, Y. Taur; and H. S. P. Wong, "Device scaling limits of Si MOSFETs and their application dependencies," *Proc. IEEE* , vol.89, no.3, pp.259-288, Mar 2001

[2] Q. Zhang, T. Fang, H. Xing, A. Seabaugh, and J. Debdeep, "Graphene Nanoribbon Tunnel Transistors," *Electron Device Letters, IEEE* , vol.29, no.12, pp.1344-1346, Dec. 2008

[3] P. L. McEuen, M. S. Fuhrer, and H. Park, "Single-walled carbon nanotube electronics," *IEEE Trans. Nanotechnol.*, vol.1, no.1, pp.78-85, Mar 2002

[4] P. Avouris, J. Appenzeller, R. Martel, S.J. Wind, "Carbon nanotube electronics," *Proc. IEEE* , vol.91, no.11, pp.1772,1784, Nov 2003

[5] M. Schroter, M. Claus, P. Sakalas, M. Haferlach, and D. Wang, "Carbon Nanotube FET Technology for Radio-Frequency Electronics: State-of-the-Art Overview," *IEEE J. Electron Devices Soc.,* vol.1, no.1, pp.9-20, Jan. 2013

[6] K. K Likharev, "Single-electron devices and their applications," *Proc. IEEE* , vol.87, no.4, pp.606,632, Apr 1999

[7] P. J. Koppinen, M. D. Stewart, and N. M. Zimmerman, "Fabrication and Electrical Characterization of Fully CMOS-Compatible Si Single-Electron Devices," *IEEE Trans. Electron Devices,* vol.60, no.1, pp.78-83, Jan. 2013

[8] W. Wei, J. Han, and F. Lombardi, "Design and Evaluation of a Hybrid Memory Cell by Single-Electron Transfer," *IEEE Trans. Nanotechnol.* , vol.12, no.1, pp.57,70, Jan. 2013

[9] W. Lu, P. Xie, and C. M. Lieber, "Nanowire Transistor Performance Limits and Applications," *IEEE. Trans. Electron Devices,* vol.55, no.11, pp.2859,2876, Nov. 2008

[10] K. Jansson, E. Lind, L. E. Wernersson, "Performance Evaluation of III–V Nanowire Transistors," *IEEE Trans. Electron Devices*, vol.59, no.9, pp.2375,2382, Sept. 2012

[11] C. S. Lent, amd P. D. Tougaw, "A device architecture for computing with quantum dots," *Proc. IEEE* , vol.85, no.4, pp.541,557, Apr 1997

[12] K. Walus, T. J. Dysart, G.A. Jullien, and R.A Budiman, "QCADesigner: a rapid design and Simulation tool for quantum-dot cellular automata," *IEEE Trans. Nanotechnol.* , vol.3, no.1, pp.26,31, March 2004

[13] D. Tougaw, and M. Khatun, "A Scalable Signal Distribution Network for Quantum-Dot Cellular Automata," *IEEE Trans. Nanotechnol.* , vol.12, no.2, pp.215,224, March 2013

[14] K. Kong; Y. Shang, and R. Lu, "An Optimized Majority Logic Synthesis Methodology for Quantum-Dot Cellular Automata," *IEEE Trans. Nanotechnol.* , vol.9, no.2, pp.170,183, March 2010

[15] R. H. Mathews, P. Sage, T. G Sollner, S.D Calawa, C. L. Chen, J. L. Mahoney, A.P. Maki, and K.M. Molvar, "A new RTD-FET logic family," *Proc. IEEE* , vol.87, no.4, pp.596,605, Apr 1999

[16] P. Mazumder, S. Kulkarni, M. Bhattacharya, J.P. Sun, and G.I. Haddad, "Digital circuit applications of resonant tunneling devices," *Proc. IEEE* , vol.86, no.4, pp.664,686, Apr 1998

[17] J. Lee, and K. Yang, "A Low-Power 40-Gb/s 1:2 Demultiplexer IC Based on a Resonant Tunneling Diode," *IEEE Trans. Nanotechnol.*, vol.11, no.3, pp.431,434, May 2012

[18] J. S. Friedman, N. Rangaraju, Y. I. Ismail, and B. W. Wessels, "A Spin-Diode Logic Family," *IEEE Trans. Nanotechnol.*, vol. 11, pp. 1026-1032, Sep. 2012.

[19] J. S. Friedman, N. Rangaraju, Y. I. Ismail, and B. W. Wessels, "InMnAs Magnetoresistive Spin-Diode Logic", *Great Lakes Symposium on VLSI (GLSVLSI) 2012*, May 2012.

[20] S. J. May and B. W. Wessels, "High-field magnetoresistance in p-(In,Mn)As/n-InAs heterojunctions," *Appl. Phys. Lett.*, vol. 88, pp. 072105-1-3, Feb. 2006.

[21] N. Rangaraju, P. C. Li, and B. W. Wessels, "Giant magnetoresistance of magnetic semiconductor heterojunctions," *Phys. Rev. B*, vol. 79, pp. 205209-1-5, May 2009.

[22] J. A. Peters, N. Rangaraju, C. Feeser, and B. W. Wessels, "Spin-dependent magnetotransport in a p-InMnSb/n-InSb magnetic semiconductor heterojunction," *Appl. Phys. Lett.*, vol. 98, pp. 193506-1-3, May 2011.

[23] B. W. Wessels, "Ferromagnetic semiconductors and the role of disorder," *New J. Phys.*, vol. 10, pp. 055008-1-17, May 2008.

[24] S. Sugahara, and N. Junsaku, "Spin-Transistor Electronics: An Overview and Outlook," *Proc. IEEE* , vol.98, no.12, pp.2124,2154, Dec. 2010

[25] A. S. Wolf, J. Lu, M. R Stan, E. Chen, and D.M. Treger, "The Promise of Nanomagnetics and Spintronics for Future Logic and Universal Memory," *Proc. IEEE*, vol.98, no.12, pp.2155,2168, Dec. 2010

[26] D. A. Allwood, G. Xiong, C. C. Faulkner, D. Atkinson, D. Petit, and R.P. Cowburn, "Magnetic domain-wall logic," *Science*, vol. 309, pp. 1688–1692, Sep. 2005.

[27] J. Welser, and K. Bernstein, "Challenges for post-CMOS devices & architectures," *Device Research Conference (DRC), 2011 69th Annual* , vol., no., pp.183,186, 20-22 June 2011.

[28] N. Song, and M. Perkowski"A new approach to AND/OR/EXOR factorization for regular arrays," *Euromicro Conference, 1998. Proceedings. 24th* , vol.1, no., pp.269,276 vol.1, 25-27 Aug 1998.

[29] M. G. A Martins, L. Rosa, A. B. Rasmussen, R.P. Ribas, A .I. Reis, "Boolean factoring with multi-objective goals," *Computer Design (ICCD), 2010 IEEE International Conference on* , vol., no., pp.229,234, 3-6 Oct. 2010

[30] M. G. A Martins, R. P. Ribas, A. I. Reis, "Functional composition: A new paradigm for performing logic synthesis," *Quality Electronic Design (ISQED), 2012 13th International Symposium on* , vol., no., pp.236,242, 19-21 March 2012

[31] Berkeley Logic Synthesis and Verification Group, ABC: A System for Sequential Synthesis and Verification, Release 130329. http://www.eecs.berkeley.edu/~alanmi/abc/