PAPER • OPEN ACCESS

Area efficient in-plane nanomagnetic multiplier and convolution architecture design

To cite this article: Santhosh Sivasubramani *et al* 2021 *Nano Ex.* **2** 020008

View the article online for updates and enhancements.

NANO EXPRESS

PAPER

Area efficient in-plane nanomagnetic multiplier and convolution architecture design

Santhosh Sivasubramani [ID] , Sanghamitra Debroy and Amit Acharyya [ID]

Advanced Embedded Systems and IC Design Laboratory, Department of Electrical Engineering, Indian Institute of Technology (IIT) Hyderabad, 502285, India

E-mail: ee15m16p100001@iith.ac.in and amit_acharyya@iith.ac.in

Abstract

In this study, we propose a nanomagnetic logic (NML) based 2 bit multiplier architecture design for the first time to the best of author's knowledge. This complex combinational logic (nanomagnetic multiplier) design proposed is built by exploiting shape, positional hybrid anisotropy and the ferromagnetically coupled fixed input majority gate. Subsequently, we extend this proposed multiplier architecture along with the NML adder architecture in introducing NML based convolution architecture design which is efficient in terms of number of nanomagnets, majority gates and clock-cycles. The proposed NML design yields ~21%–72%, ~26%–42%, ~36%–63%, and ~20%–68%, reduction in the required number of nanomagnets, majority gate, clock cycles and energy compared to the state-of-the-art designs.
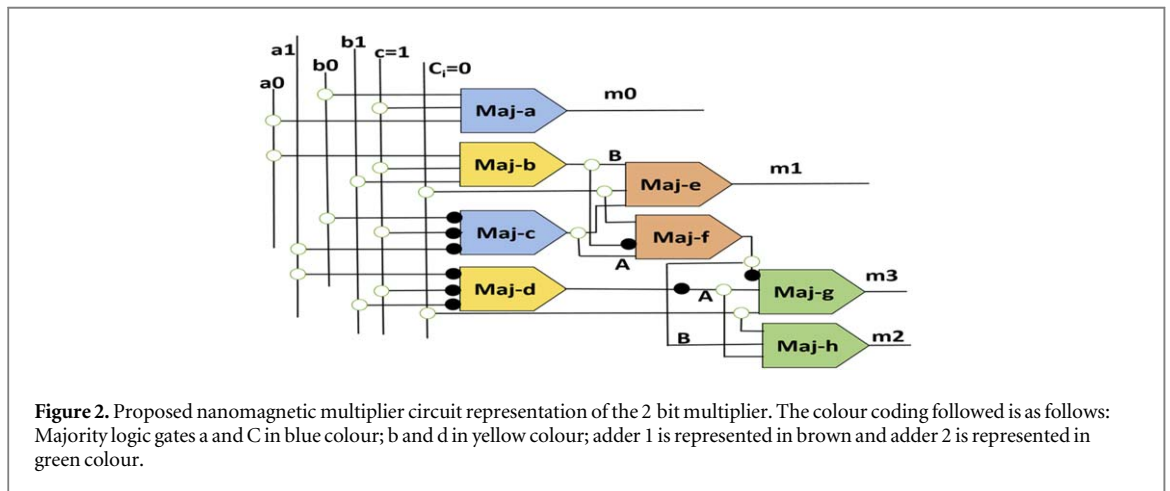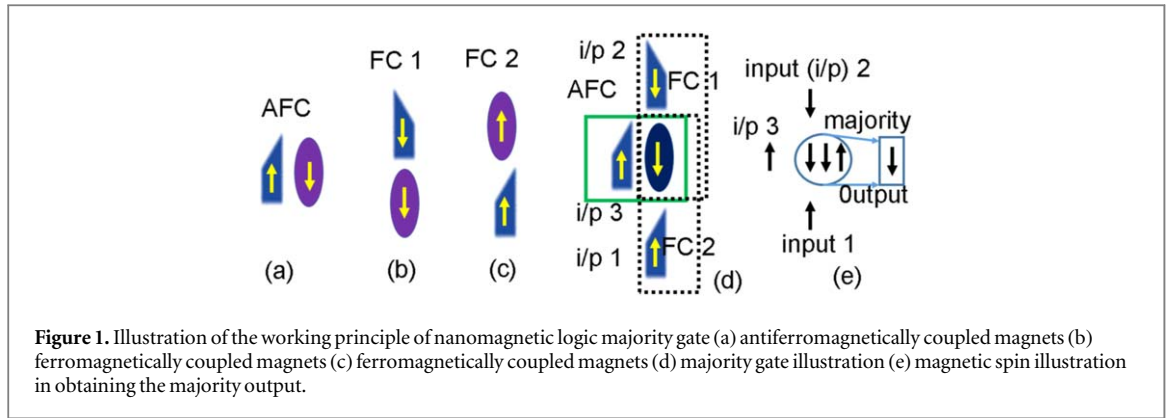
1. Introduction

Quantum-dot Cellular Automata (QCA) are nanoscale devices which are physically implemented combining the discrete property of quantum mechanics and cellular automata. Fabrication of QCA by exploiting the interaction among magnetic nanoparticles has gained momentum owing to its room temperature implementation. Since its first revelation by Cowburn *et al.*, [1], Magnetic Quantum-dot Cellular Automata (MQCA) based in-plane Nanomagnetic Logic (NML) has been emerging as a rebooting computing platform [2] with the prospect to complement CMOS devices in the domain of spintronics [3–8]. Followed by its revelation, Csaba *et al.*, have demonstrated that nanomagnets can be used for information propagation [9]. Subsequently the first universal majority logic gate (MLG) implementation has been shown by Imre *et al.*, [10] using oval shaped input and elongated drivers which are clocked by applied field from magnetic force microscopy (MFM) tips [11]. In consequence, the first time demonstration of NML full adder has been shown by Varga *et al.*, [12] using fanout and interconnects.

Subsequently, researchers started exploiting shape (S) anisotropy [13, 14] of the nanomagnets leading to optimization of the arithmetic circuit implementation [15]. Furthermore, positional (P) anisotropy of the nanomagnets [16] have been exploited aiding towards mis-alignment free design. In continuation, Li *et al.*, [16] have shown the 1 bit full adder implementation using 45 degree aligned input nanomagnets.

In parallel, researchers demonstrated approximate arithmetic computing using NML [17]. Consequently, in our recently reported works, we have utilized SP hybrid anisotropy [18] of nanomagnets to design an 1bit full adder. In subsequence, the authors proposed an optimized NML adder implementation by exploiting physical analogy of ferromagnetically coupled fixed input majority gate (FMG) [19] and further extended it to perform runtime reconfigurable approximate arithmetic computation [20, 21].

In view of the above, both from the material and architecture point of view the adder architecture design has been the focus till date. The main objective of this study is to explore how these basic building blocks can be extended further to implement complex arithmetic logic using NML.

Figure 1. Illustration of the working principle of nanomagnetic logic majority gate (a) antiferromagnetically coupled magnets (b) ferromagnetically coupled magnets (c) ferromagnetically coupled magnets (d) majority gate illustration (e) magnetic spin illustration in obtaining the majority output.

Figure 2. Proposed nanomagnetic multiplier circuit representation of the 2 bit multiplier. The colour coding followed is as follows: Majority logic gates a and C in blue colour; b and d in yellow colour; adder 1 is represented in brown and adder 2 is represented in green colour.
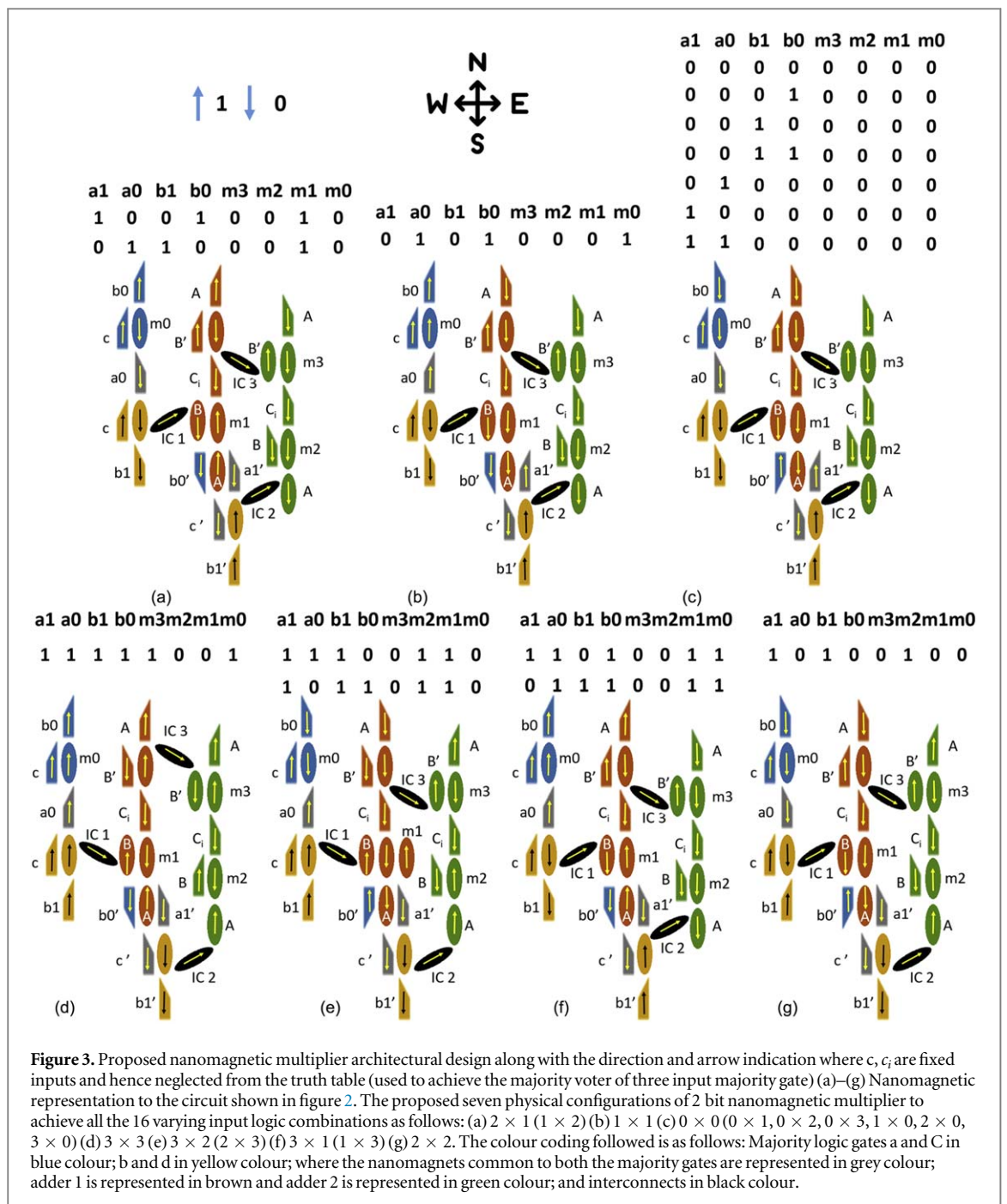
Table 1. Truth table entries with all possible input logic combinations for 2 bit multiplier with inputs a1, a0, b1, b0 and outputs m3, m2, m1, m0.

Input				Output			
a1	a0	b1	b0	m3	m2	m1	m0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1
0	1	1	0	0	0	1	0
0	1	1	1	0	0	1	1
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	1	0	0
1	0	1	1	0	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	1

With the emerging demand of the resource constrained implementation [8] of the computationally intensive convolutional neural networks (CNN) on the edge, requires area and energy efficient architecture design for performing convolution operation. The authors believe that the MQCA based logic computing devices would play a major role in such design and motivated by this, in a first attempt, we propose an NML based efficient nanomagnetic multiplier architecture design along with the convolution.

The rest of the study is organized as follows: section 2 briefs the proposed design and anaysis, followed by its discussion and conclusion in section 3.

Figure 3. Proposed nanomagnetic multiplier architectural design along with the direction and arrow indication where c, c_i are fixed inputs and hence neglected from the truth table (used to achieve the majority voter of three input majority gate) (a)–(g) Nanomagnetic representation to the circuit shown in figure 2. The proposed seven physical configurations of 2 bit nanomagnetic multiplier to achieve all the 16 varying input logic combinations as follows: (a) 2×1 (1×2) (b) 1×1 (c) 0×0 ($0 \times 1, 0 \times 2, 0 \times 3, 1 \times 0, 2 \times 0, 3 \times 0$) (d) 3×3 (e) 3×2 (2×3) (f) 3×1 (1×3) (g) 2×2. The colour coding followed is as follows: Majority logic gates a and C in blue colour; b and d in yellow colour; where the nanomagnets common to both the majority gates are represented in grey colour; adder 1 is represented in brown and adder 2 is represented in green colour; and interconnects in black colour.
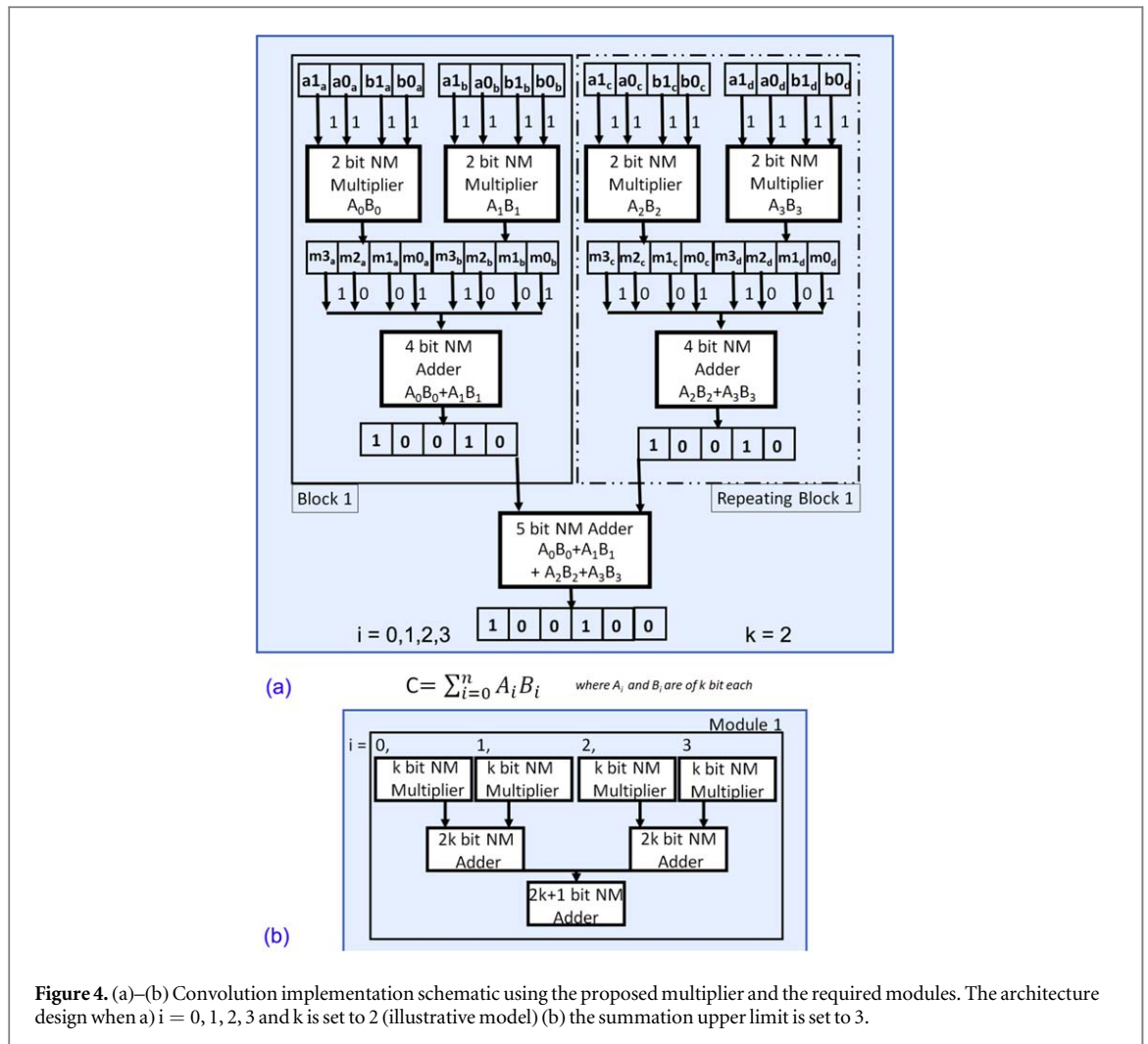
2. Proposed design and analysis

The working principle of nanomagnetic majority logic is depicted in figures 1(a)–(e). When two nanomagnets are placed side by side it exhibits antiferromagnetic data propagation (inverter) and when placed one above another it enables ferromagnetic data propagation. When this both antiferromagnetic and ferromagnetic data propagation are coupled in a single system that exhibits a majority logic gate where the state of the compute middle nanomagnet is determined by the combined influence of inputs 1,2 & 3 (cf 1(d), (e)).

Proposed MQCA based NML Architecture Design for Convolution:- The objective of this section is to introduce the MQCA based architecture to accomplish the computationally intensive convolution in area and energy efficient way. In general convolution is expressed as follows

$$C = \sum_{i=0}^{n} A_i B_i \tag{1}$$

where A_i, B_i are of k bits and C is the convoluted output.

Figure 4. (a)–(b) Convolution implementation schematic using the proposed multiplier and the required modules. The architecture design when a) i = 0, 1, 2, 3 and k is set to 2 (illustrative model) (b) the summation upper limit is set to 3.

To begin with, we propose the design of first of its kind nanomagnetic multiplier. The inputs of the 2 bit multiplier comprises of a1, a0 (bits of A), b1, bo (bits of B) and their corresponding truth table entries are tabulated in table 1.

The 2 bit multiplier [22] outputs are defined as m3, m2, m1 and m0.

As shown in figures 2 and 3, here we propose to use the majority logic blocks Maj-a, Maj-b, Maj-c, Maj-d, Maj-e, Maj-f, Maj-g and Maj-h to build the 2 bit nanomagnetic multiplier. The majority logic blocks and their functionalities are defined as follows

- (a) Maj-a and Maj-b form the two three-inputs AND implementation blocks

- (b) Maj-c and Maj-d form the two three inputs NOR implementation blocks and

- (c) Maj-e, Maj-f and Maj-g, Maj-h form the two individual 1 bit nanomagnetic adder blocks respectively.

The majority gates Maj-a and Maj-b are fed with the inputs a0, b0, c and a0, b1, c respectively where c is set to 1 and a0 acting as a common input for both Maj-a and Maj-b. The Maj-a corresponds to the output m0. Maj-c and Maj-d are fed with the inverted inputs of a1, b0, c and a1, b1, c respectively where $\overline{a1}$, $\overline{c}$ acts as the common inputs for both Maj-c and Maj-d.

We have adopted the DeMorgan's theorem as stated in equation (2) for the optimization as defined below. By DeMorgan's first theorem we know that,

$$\overline{A \ + \ B} = \overline{A}\,\overline{B} \tag{2}$$

Thus the inputs of Maj-c and Maj-d can be represented as,

$$\overline{\overline{a1} \ + \ \overline{b0}} = \overline{\overline{a1}}\,\overline{\overline{b0}} = a1bo \tag{3}$$

and similarly,

$$\overline{\overline{a1} + \overline{b1}} = \overline{\overline{a1}\,\overline{b1}} = a1b1 \tag{4}$$

Hence, we propose to use the NOR implementation using Maj-c and Maj-d for the reduction of nanomagnets. By using this proposed optimization and placement of nanomagnets together yields 37.5% reduction in the number of nanomagnets to implement the gate functionalities compared to the state-of-the-art approaches which will be detailed in the upcoming discussions.

As it is evident from the aforementioned, the proposed multiplier comprises of adder blocks, therefore its efficient implementation requires efficient adder architecture. Hence to enhance the optimization further, an efficient adder architecture recently reported by the authors' [19] have been used to replace the existing adder block (for brevity the details of this architecture is omitted [19]). The 2bit multiplier architecture representation shown in figure 2 is substituted with their corresponding nanomagnetic representation as depicted in figure 3.

Maj-a, Maj-b, Maj-c, Maj-d constitutes the inputs of the 2 bit multiplier and the Maj-e, Maj-f and Maj-g, Maj-h constitutes the two adder blocks for computation respectively. The outputs of Maj-b, Maj-c and carry-in (set to 0) are fed as inputs to the Maj-e which computes the output m1 and this translates for the sum compute of the adder block-1. Similarly, carry-in (set to 0), the output of Maj-c, and the inverted output of Maj-b are fed as inputs of Maj-f which does the intermediate carry computation of the adder block-1.

The inverted outputs of Maj-f, Maj-d (representing NOR implementation) and carry-in (set to 0) constitutes the Maj-g for carry computation of the adder block-2 representing the output m3. Maj-h computes the sum output of the adder block-2 which comprises of the inputs as:- the output of Maj-f, inverted output of Maj-d and carry-in (set to 0) representing the output m2 of the 2 bit nanomagnetic multiplier. The direction indication, figure legends such as ↑ , ↓ representing 1, 0 respectively are depicted in the figure along with the color coding of individual majority blocks.

The architecture represented in figure 3 constitutes for the implementation of the possible logic variation for 2 bit multiplier as tabulated in table 1. Select configurations 1–7 have been used to pick the particular architecture corresponding to the input logic variations as depicted in the figure. By following this procedure, the number of design configurations required to achieve all the input logic variation is reduced. As an example 16 design configurations (cf table 1) required for 2 bit multiplication implementation is reduced to 7 design configurations (cf figure 3).
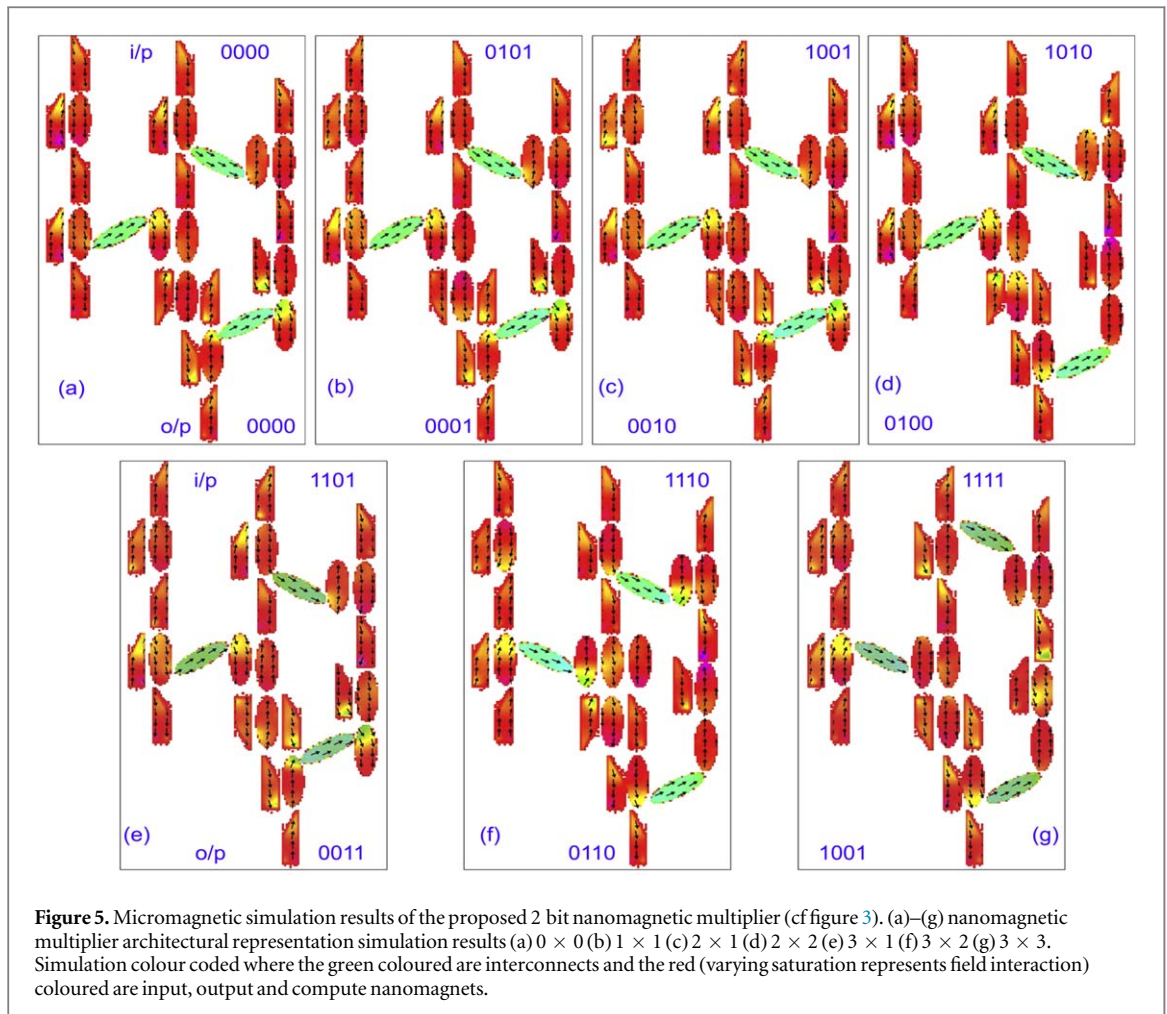
This proposed procedure hence leads to 56.25% reduction in the area footprint required for the optimized multiplier implementation which will be detailed further. This area efficient design plays a significant role in the convolution architecture design. Thus, the combined adder block and the proposed multiplier block are used to implement the convolution architecture as detailed below.

$$C = A_0 B_0 + A_1 B_1 + A_2 B_2 + A_3 B_3 \tag{5}$$

Convolution Illustration:- The modules to implement equation (5) is depicted in figure 4(a). The first module block, representing $A_0 B_0$ consists of 30×7 totaling 210 nanomagnets (cf figure 3), similarly the module blocks representing $A_1 B_1$, $A_2 B_2$ and $A_3 B_3$ individually corresponds to 210 nanomagnets each. The design of k-bit nanomagnetic adder design using 1 bit adder is briefly discussed in one of our recently reported work [19]. Subsequently, the module blocks representing $A_0 B_0 + A_1 B_1$, $A_2 B_2 + A_3 B_3$ and $A_0 B_0 + A_1 B_1 + A_2 B_2 + A_3 B_3$ comprises a total of 32, 32 and 40 nanomagnets respectively. The convolution design illustrated in figure 4(a) constitutes a total of $210 \times 4 + 32 \times 2 + 40 = 944$ along with $\sim$40 interconnects nanomagnets leading to $\sim$984 nanomagnets; $56 \times 4 + 8 \times 2 + 10 = 250$ majority gate operations and $6 \times 4 + 4 \times 2 + 5 = 37$ clock-cycles to compute C as defined in equation (5). These calculations have been performed considering both Block 1 and Repeating Block 1 as depicted in figure 4(a). However, it can be seen that the repeating block 1 is found to be redundant and hence the resources of block 1 can be shared for computing both $A_0 B_0 + A_1 B_1$ and $A_2 B_2 + A_3 B_3$ individually which will also result in drastic reduction in the total of number of nanomagnets, majority gate operations and clock cycles.
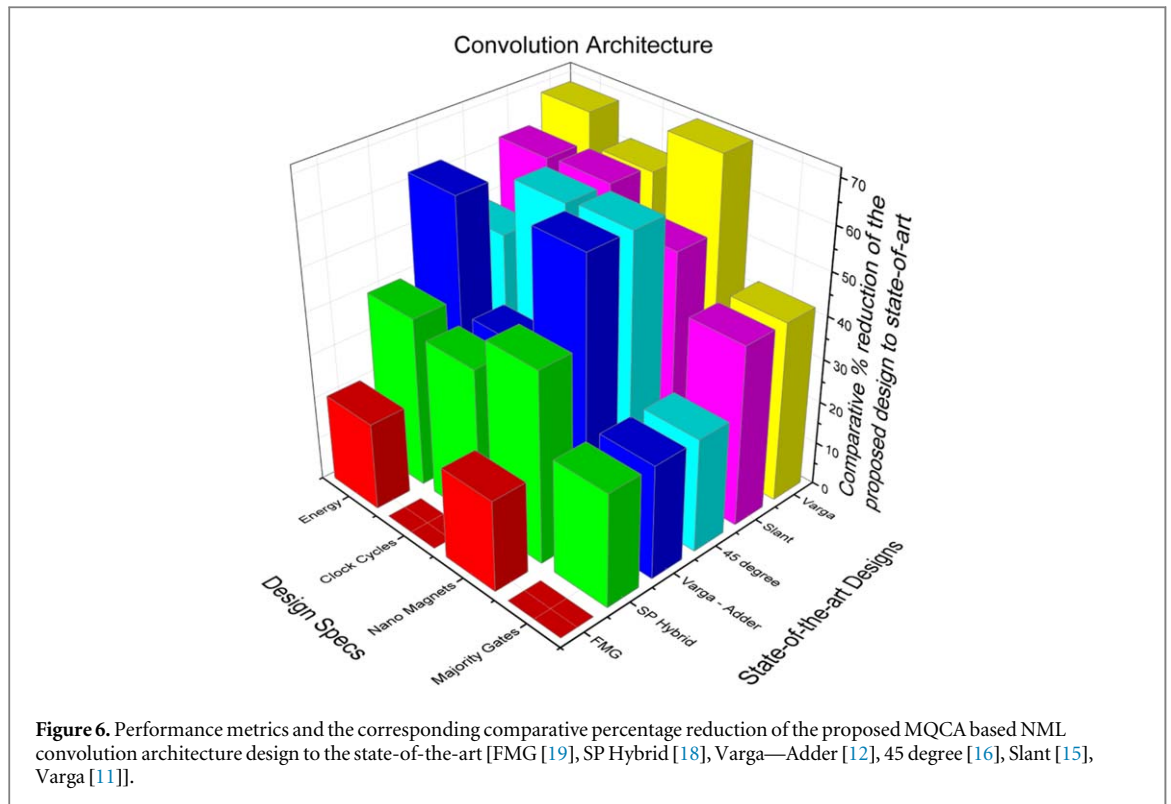
Subsequently, reduction in the number of nanomagnets, majority gates accounts for the area, energy efficiency and similarly the reduction in the number of clock-cycles leads to high speed. Here, we have generalized our design approach to implement K bit convolution operation using k bit nanomagnetic multiplier and 2k bit nanomagnetic adder as depicted in figure 4(b). This block is represented as module 1 and this module is repeated multiple times to achieve computationally intensive convolution , where the summation upper limit is set to 15 (generic model using module 1). In the upcoming discussions we will describe the inter-module communication required for implementation of this proposed convolution architecture design.

Each module contains set of logic functionalities for execution as detailed above. Output data propagation from one module to another module can be achieved by incorporating the following:- The foremost way of inter-module communication is to have the nanomagnetic wire architecture translating in the form of interconnect nanomagnets exploiting SP hybrid anisotropy. Data propagation is also achieved by using buffer

Figure 5. Micromagnetic simulation results of the proposed 2 bit nanomagnetic multiplier (cf figure 3). (a)–(g) nanomagnetic multiplier architectural representation simulation results (a) 0×0 (b) 1×1 (c) 2×1 (d) 2×2 (e) 3×1 (f) 3×2 (g) 3×3. Simulation colour coded where the green coloured are interconnects and the red (varying saturation represents field interaction) coloured are input, output and compute nanomagnets.

and inverters which comprises of odd and even number of nanomagnets respectively. Similarly, (a) the input and output of the first and last modules can be interfaced to the external CMOS modules using nanoscale spin valves; (b) input can be field-coupled and the output using spin-valve; (c) electro magnetic interface can also be achieved by domain walls (DWs) and (d) magnetic tunnel junction based I/O interface by exploiting the free layer and giant magnetoresistance. However, the focus of this study is to propose an efficient convolution architecture design, for brevity a short summary on inter-module communication is included.

Discussion:- The micromagnetic simulation results of the proposed multiplier design (cf figure 3) is depicted in figure 5. The application of an external magnetic field powers the input slanted edge standalone input nanomagnets. To achieve all the input logic variations of the 2 bit nanomagnetic multiplier, different positions of the slanted edges are required comprising 16 different physical configurations (constituting 480 nanomagnets) as proven effective by earlier experimental realizations [10–12, 15]. Though proven effective in literatures, in the process of scaling up to higher bit multiplier designs there arises a significant need for optimization. To mitigate this, with our proposed design we have clustered it to only 7 different physical configurations (constituting 210 nanomagnets) resulting in 56.25% reduction in the area footprint and the reduction of number of nanomagnets. As shown in figure 3 by choosing select configuration 1 to 7 the implementation of varying input logic combinations are achieved (cf table 1). Not only from the number of physical configurations perspective, our proposed design and optimization of a) 2 NOR majority blocks and placement of nanomagnets together yields 37.5% reduction in the number of nanomagnets; b) optimization of 2 AND majority blocks yields 12.5% reduction in the number of nanomagnets compared to the traditional implementation of AND and NOR majority blocks. Figure 5(a) corresponds to the multiplication ouput 0; (b) corresponds to the simulation outputs of 1×1 leading to output 1; (c) corresponds to 2×1 vis-a-vis 1×2 leading to output 2; (d) corresponds for 2×2 and its output 4 (e) 3×1 vis-a-vis 1×3 for output 3 (f) 3×2 vis-a-vis 2×3 corresponding to output 6; and (g) corresponding simulation output for 3×3. As detailed, the resource requirements for the implementation of the nanomagnetic convolution (for illustration cf figure 4) is 984 nanomagnets, 250 majority gates and 37 clock cycles using the proposed efficient multiplier and the state-of-the art authors' proposed adder architecture.

Figure 6. Performance metrics and the corresponding comparative percentage reduction of the proposed MQCA based NML convolution architecture design to the state-of-the-art [FMG [19], SP Hybrid [18], Varga—Adder [12], 45 degree [16], Slant [15], Varga [11]].

The performance metrics of the proposed MQCA based architecture design in comparison to the state-of-the art is depicted in figure 6. The analysis presented are for one physical configuration and without interconnects. From which it is evident that the proposed architecture design is efficient in terms of required number of nanomagnets, majority gates and clock cycles.

Simulation Framework:- Object Oriented MicroMagnetic Framework (OOMMF) [23], a micromagnetic open source simulation tool developed by NIST at ITL, which solves the Landau–Lifshitz Gilbert's Ordinary Differential equation using the 4th order Runge Kutta solver is used throughout this study. We opted for OOMMF, as the researchers vividly use it for designing, validating and reporting, owing to its reproducible and reliable system development that aids in the real-time realization of the developed theoretical models [10, 12–15, 18]. In OOMMF, 3D spins on 2D mesh cells are relaxed using Landau–Lifshitz PDE solver. Spin orbit coupling interaction gives rise to magnetic anisotropic energy. Stoner Wohlfarth model is applied for magnetization rotation to place 45° aligned single-domain nanomagnets. Slanted edge of the nanomagnets aids in achieving standalone inputs and all its magnetic moments are aligned in the easy (long) axis. The direction of the slant edge determines the final state (My) (relaxed state) of the nanomagnet upon removal of the applied field (X component). We have used permalloy (Py) (78.5% nickel, 21.5% iron composition) magnetic dots which is a pronounced polycrystalline soft magnetic material with uniaxial anisotropy constant value set as zero (ie., low coercive field) and larger exchange energy comparatively to the magnetocrystalline anisotropy energy (MAE) [24]. Thus requiring high axial symmetry in maintaining the magnetic anisotropy which is dominated by the exchange interaction as specified by the single-domain bistable nanomagnetic exchange Hamiltonian. In lieu of time, the maximum torque $| m \times h |$ set to 10^{-5} A/m, with damping coefficient of 0.25, saturation magnetization of 800×10^3 A/m and the exchange stiffness constant of 13×10^{-12} J/m [18, 19]. A hierarchical layout editor CleWin [25] developed by WieWeb software has been used for modeling. Design specifications for the proposed nanomagnetic architecture are as: dimensions of slanted edge nanomagnets: 15nm×30 nm area; 10 nm thickness; oval shaped nanomagnets: 10 nm×30 nm area; 10 nm thickness and antiferromagnetic and ferromagnetic coupling wall separation of 10 nm | 15 nm. Finer 3D spins in 2D mesh has been taken into account and the underestimation of gap by ±5% is less likely to have its effect on the reliability of the circuit. The architecture proposed here is designed with sub-50 nm design node which requires special attention (in-spite of its earlier experimental realization of slanted edges and 45 degree aligned nanomagnets), which could be better realizable with the state of the fabrication techniques. The proposed design is also evaluated against thermal fluctuations and premature bit-flip as defined by theoretical models and found satisfactory which is also briefly reported in one of our recently reported work [19]. The sub 50nm sized nanomagnets used here exhibits the possible scaling down limits of nanomagnets using in-plane dipolar coupling adhering to the theoretical standards without entering the superparamagnetic regime. However, it is to be noted that the proposed design

methodologies are independent of the design nodes and could be realized using existing design nodes of sub 180 nm and 250 nm [21].

3. Conclusion

We have proposed the area and energy efficient MQCA based 2 bit nanomagnetic multiplier architecture and convolution design approach as a simulation based proof-of-concept demonstration. The proposed design yields ~21%–72%, ~26%–42%, ~36%–63%, and ~20%–68%, reduction in the required number of nanomagnets, majority gate, clock cycles and energy compared to the state-of-the-art designs. The proposed energy efficient architecture design is envisaged to have its applications in edge computing and also have its potential impact on graphene [26–28] based on-chip clocking.

Acknowledgments

Author Contributions

S. Sivasubramani, and A. Acharyya, conceived the idea. S. Sivasubramani designed and modeled the theoretical setup, NML architecture, micro-magnetic simulations and analysis. S. Sivasubramani performed B-field optimization for data propagation. S. Sivasubramani wrote the paper. S. Debroy revised the draft. A. Acharyya and S.G. Acharyya supervised the study and revised the paper. All the authors discussed the results and commented on the final manuscript.

ORCID iDs

Santhosh Sivasubramani ⬤ https://orcid.org/0000-0003-1607-0989
Amit Acharyya ⬤ https://orcid.org/0000-0002-5636-0676

References

[1] Cowburn R P and Welland M E 2000 Room temperature magnetic quantum cellular automata *Science* **287** 1466–8
[2] Sivasubramani Santhosh 2020 Rebooting computing: Nanomagnetic logic based computing architecture design methodology *PhD dissertation, Indian Institute of Technology Hyderabad* **1** 1-250
[3] Cowburn R P 2002 Probing antiferromagnetic coupling between nanomagnets *Phys. Rev.* B **65** 092409
[4] Debroy S, Sivasubramani S, Ghosh Acharyya S and Acharyya A 2019 Nanomagnetic computing for next generation interconnects and logic design. *VLSI and Post-CMOS Electronics: Devices, circuits and interconnects* **2** 151
[5] Sivasubramani S, Debroy S and Acharyya A 2020 Power and area-efficient architectural design methodology for nanomagnetic computation *In Nanoscale VLSI* (Singapore: Springer) 241–70
[6] Fashami M S *et al* 2011 Magnetization dynamics, bennett clocking and associated energy dissipation in multiferroic logic *Nanotechnology* **22** 155201
[7] Wolf S A, Lu J, Stan M R, Chen E and Treger D M 2010 The promise of nanomagnetics and spintronics for future logic and universal memory *Proc. IEEE* **98** 2155–68
[8] Porod W and Niemier M 2015 Better computing with magnets-the simple bar magnet, shrunk down to the nanoscale, could be a powerful logic device *IEEE Spectr.* **52** 44–60
[9] Csaba G, Imre A, Bernstein G H, Porod W and Metlushko V 2002 Nanocomputing by field-coupled nanomagnets *IEEE Trans. Nanotechnol.* **1** 209–13
[10] Imre A, Csaba G, Ji L, Orlov A, Bernstein G H and Porod W 2006 Majority logic gate for magnetic quantum-dot cellular automata *Science* **311** 205–8
[11] Edit V 2013 *Chapter-10, Experimental Study of Novel Nanomagnet Logic Devices, Doctoral Dissertation, University of Notre Dame, United States of America* University of Notre Dame
[12] Varga E, Csaba G, Bernstein G H and Porod W 2011 Implementation of a nanomagnetic full adder circuit *2011 XI IEEE International Conference on Nanotechnology* 1244–7 IEEE
[13] Niemier M T, Varga E, Bernstein G H, Porod W, Alam M T, Dingler A, Orlov A and Hu X S 2010 Shape engineering for controlled switching with nanomagnet logic *IEEE Trans. Nanotechnol.* **11** 220–30

[14] Zhang B, Yang X, Wang Z and Zhang M 2014 Innovative orderly programmable in-plane majority gates using trapezoid shape nanomagnet logic devices *Micro & Nano Letters* **9** 359–62

[15] Varga E, Niemier M T, Csaba G, Bernstein G H and Porod W 2013 Experimental realization of a nanomagnet full adder using slanted-edge magnets *IEEE Trans. Magn.* **49** 4452–5

[16] Li Z and K M Krishnan A 2017 3-input all magnetic full adder with misalignment-free clocking mechanism *J. Appl. Phys.* **121** 023908

[17] Labrado C, Thapliyal H and Lombardi F 2017 Design of majority logic based approximate arithmetic circuits *2017 IEEE International Symposium on Circuits and Systems (ISCAS)* 1–4 IEEE

[18] Sivasubramani S, Mattela V, Pal C, Islam M S and Acharyya A 2018 Shape and positional anisotropy based area efficient magnetic quantum-dot cellular automata design methodology for full adder implementation *IEEE Trans. Nanotechnol.* **17** 1303–7

[19] Sivasubramani S, Mattela V, Pal C and Acharyya A 2019 Nanomagnetic logic design approach for area and speed efficient adder using ferromagnetically coupled fixed input majority gate *Nanotechnology* **30** 37LT02

[20] Sivasubramani S, Mattela V, Pal C and Acharyya A 2019 Dipole coupled magnetic quantum-dot cellular automata-based efficient approximate nanomagnetic subtractor and adder design approach *Nanotechnology* **31** 025202

[21] Sivasubramani S, Mattela V, Rangesh P, Pal C and Acharyya A 2020 Nanomagnetic logic based runtime Reconfigurable area efficient and high speed adder design methodology *Nanotechnology* **31** 18LT02

[22] Cho H and Swartzlander E E 2009 Adder and multiplier design in quantum-dot cellular automata *IEEE Trans. Comput.* **58** 721–7

[23] Donahue M J and Porter D G 1999 OOMMF User's Guide, Version 1.0 (National Institute of Standards and Technology, Gaithersburg, MD, 1999). NIST *Technical Report No.* NISTIR 6376

[24] Arnold H D and Elmen G W 1923 Permalloy, a new magnetic material of very high permeability *Bell Syst. Tech. J.* **2** 101–11

[25] WieWib Software, CleWin—a hierarchical layout editor, MESA+ Research Institute at the University of Twente and Deltamask.

[26] Sivasubramani, S., Acharyya, A. Investigation on electronic transport and magnetic properties of graphene for its applications in nanomagnetic computing. (Masters thesis, Indian Institute of Technology Hyderabad). http://raiith.iith.ac.in/id/eprint/4203 (2018)

[27] Sivasubramani S, Debroy S, Acharyya S G and Acharyya A 2018 Tunable intrinsic magnetic phase transition in pristine single-layer graphene nanoribbons *Nanotechnology* **29** 455701

[28] Debroy S, Sivasubramani S, Vaidya G, Acharyya S G and Acharyya A 2020 Temperature and Size Effect on the Electrical Properties of Monolayer Graphene based Interconnects for Next Generation MQCA based Nanoelectronics *Sci. Rep.* **10** 1–11