

# Towards Logic Functions as the Device

Prasad Shabadi, Alexander Khitun, Pritish Narayanan,  
Mingqiang Bao, Israel Koren, Kang L. Wang and C. Andras Moritz

**Abstract** - This paper argues for alternate state variables and new types of sophisticated devices that implement more functionality in one computational step than typical devices based on simple switches. Elementary excitations in solids enabling wave interactions are possible initial candidates to create such new devices. The paper focuses on magnon-based spin-wave-logic functions (SPWF) and presents high fan-in majority, weighted high fan-in majority, and frequency-multiplexed weighted high fan-in majority devices as initial SPWFs. Experiments proving feasibility are also shown. Benefits vs. scaled CMOS are quantified. Results show that for 128 or larger inputs even a 2.5 $\mu$ m SPWF carry-look-ahead adder implementation is faster than the 45nm CMOS version. The 45nm SPWF adder is expected to be significantly faster across the whole range of input widths. In particular, the 45nm SPWF CLA adder is estimated to be at least 77X faster than CMOS version for input widths equal to or greater than 1024. A second example of a counter circuit is presented to illustrate the considerable reduction in complexity possible vs. CMOS.

**Keywords:** state variables, nanoscale fabrics, spin wave functions, high functionality devices.

## I. INTRODUCTION

CMOS technology scaling, driven by the goal of higher performance with minimum area and power consumption, is reaching fundamental limits forcing researchers to look beyond MOSFETs, the top-down CMOS manufacturing mindset, and the associated hierarchical multi-level logic organization, for new ideas. New devices based on spin-FETs [1], molecular-level physical phenomena, and FET devices based on emerging nanoscale materials, e.g., built with nanowires [2], graphene ribbons [3], and carbon nanotubes [4] are actively investigated, in addition to new types of computational circuits and fabrics [2][5-7].

The main focus in the device community, however, has been on improving the intrinsic delay, minimizing switching power and leakage in a single device, often assuming that the rest of the paradigm to design chips could remain almost unchanged from CMOS. For example, even with computation paradigms based on new types of physical phenomena (i.e. alternate state variables), what forms the basic device is often envisioned as a simple controlled/gated switch. The computational paradigm relies then on fairly conventional mindset: building a first gate out of these switches with relatively low fan-in and fan-out, cascading these into multiple levels with larger fan-in and more complex logic functions and then forming blocks of conceptually interesting components like an ALU, etc. What is notable is that by the time these blocks are composed, due to the many levels of logic and wiring requirements, the original goal of having a small switching delay, while still important, becomes less critical. Clearly, system-level performance does not always scale in proportion to the individual device performance. Furthermore,

it is increasingly accepted that charge-based electronics is very competitive and there is no alternate state variable based switch on the horizon that would be much faster [8].

Can there be then a better, game-changing way to improve system-level performance? While there are many possible pathways to attack the nanoscale fabric problem, what we propose in this paper is to shift the focus towards new types of devices that can be made more functional than simple switches. For example, imagine if it would be possible to devise devices that are able to implement arbitrary logic operations or even logic functions with high fan-in and fan-out without proportionally increasing delay, power and area.

One physical fabric approach that has that potential is based on non-equilibrium physical phenomena and wave interactions, e.g., spin waves. In the proposed new device structures, logic operations are based on wave superposition implementing spin wave functions. We show control mechanisms and structures to allow increased functionality for wave superposition to implement arbitrary logic with very high fan-in and fan-out but without increasing proportionally area and delay. Our thinking is that these novel computing devices and associated systems could ensure considerably more benefits than speedup optimizations of more basic device switches (coupled with a conventional logic organization).

The main contributions of this paper include: i) a vision for future computation based on sophisticated logic functions as new devices, ii) new high fan-in/fan-out electron spin SPWFs, iii) initial experimental validation, and iv) initial exploration of benefits over state-of-the-art scaled CMOS with two example arithmetic circuits.

The rest of the paper is organized as follows. Section II presents our vision for enhanced logic functionality incorporated into a single device. Section III presents Spin Wave logic Functions (SPWFs). Feasibility of the fabric is also discussed and initial experimental validation is shown. Sections IV and V present performance-gain and complexity-reduction benefits of this approach based on our initial explorations of two arithmetic circuits. Section VI concludes the paper.

## II. DEVICES THAT ARE LOGIC FUNCTIONS

Several proposals have been made towards implementing a highly efficient computational system at nanoscale. However, complex circuits have been typically realized based on devices that are simple switching elements. Most research towards emerging devices focuses on improving the performance of such simple devices. In contrast, our vision is to use sophisticated devices able to implement logic functions in one physical step as building blocks for more complex systems.

Fig. 1 shows the basic idea of the device in conventional computational systems and our envisioned approach.

---

This work was supported in part by the Focus Center Research Program (FCRP) – Center on Functionally Engineering Nano Architectonics (FENA), the Center for Hierarchical Manufacturing (CHM) at UMass Amherst, and NSF awards CCR:0105516, NER:0508382, and CCR:051066.

Prasad Shabadi, Pritish Narayanan, Israel Koren and C. Andras Moritz are with University of Massachusetts at Amherst, Amherst, MA 01003 USA.

Alexander Khitun, Mingqiang Bao and Kang L. Wang are with University of California Los Angeles, Los Angeles, CA 90024 USA.

Corresponding authors: [shabadi@ecs.umass.edu](mailto:shabadi@ecs.umass.edu), [andras@ecs.umass.edu](mailto:andras@ecs.umass.edu)

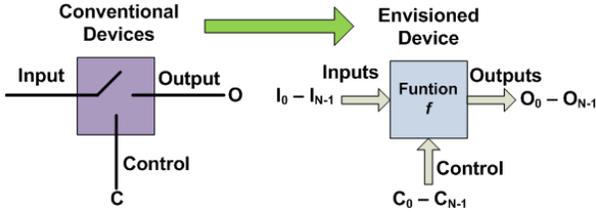


Figure 1. Devices for nano-fabrics: (left) conventional switch; (right) envisioned device with alternate state variables.

In our approach, a single device simultaneously processes a large number of inputs and accomplishes sophisticated logic functions based on external control. The key requirements for such a device would be i) alternate ways to encode information utilizing novel physical phenomena (alternate state variables) and ii) new types of interactions between inputs/control achieving a desired logic functionality.

Non-equilibrium physical phenomena with wave-based interactions could be leveraged to meet the above requirements. In a non-equilibrium fabric, switching times are lower than the thermal relaxation time, leading to fast processing. Information may be encoded in the amplitude and/or phase of a wave. Interactions between waves, such as interference and superposition, may then be utilized for achieving specific logic functions. Elementary excitations in solids may be used for information encoding and processing. Examples include spin waves, plasmonics, photon and phonon excitations and many quasi particles.

In this paper we present new functional devices based on these types of wave interactions. The key aspect here is how to disruptively scale differently the control inputs from the regular inputs. For example, we create several different high fan-in devices accomplishing logic in a single step of computation – without imposing added complexity on the control proportionally. We may then use such devices in circuits implementing complex logic. We show one possible approach using spin-waves but the general approach is applicable to other wave-based phenomena as well.

### III. SPIN WAVE FUNCTIONS (SPWFs)

A spin wave is a collective oscillation of spins in an ordered spin lattice around the direction of magnetization in ferromagnetic materials [9]. Information may be encoded into the phase of spin waves propagating in these materials. Superposition of spin waves of different phases helps achieve useful logic functionality. More conventional logic circuits (e.g. AND, NAND, OR, etc) based on spin waves have been proposed in [10]. Various topologies can be envisioned for a given functionality.

Fig. 2 shows key physical components of spin-wave-based fabrics. Magnetic waveguides, phase shifters and amplifiers [10] form the necessary infrastructure for realizing these systems. Spin waves propagate through waveguides. Phase shifters are used to manipulate the phases of the waves and invert logic states. Magneto-electric (ME) cells manipulate spin wave amplitudes [10]. Input/output functions are achieved using electro-magnetic coupling between the ferromagnetic material and input rails.

Superposition interactions between spin waves naturally lend themselves to majority function implementation. For example, consider interference of three spin waves with equal amplitudes. If two of the waves are in phase ‘0’ and the third wave is in phase ‘1’, the resultant wave will be of phase ‘0’. Majority logic is an efficient way of implementing digital logic. Instead of using Boolean logic operators (e.g. AND, OR, NAND), majority logic represents and manipulates digital inputs on the basis of majority decision.

In fact, the majority function described is an example of Spin Wave logic Functions (SPWFs). Additional functionality can be obtained by adjusting various other physical parameters: i) Amplitude of input signals and control can be manipulated using ME cells; ii) Frequency multiplexing can be used to simultaneously transmit several spin waves over a waveguide with different functionalities realized for different frequencies; iii) Control inputs (inputs that alter, for example, the majority decision) can be modified to achieve arbitrary functionality. Moreover, a small number of control signals can be used with a large number of inputs by adjusting the amplitude of the

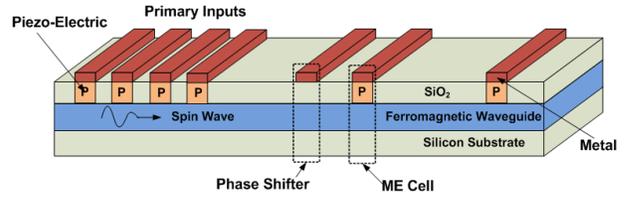


Figure 2. Key physical components of a spin-wave based computing fabric including ferromagnetic waveguides, phase shifters for phase manipulation, ME cells for spin amplifications and input rails.

control; and iv) Topology of the circuits can be adjusted to modify spin wave interactions. These knobs provide much flexibility to achieve sophisticated logic functions in a single step. More on these will be presented next.

We describe devices implementing SPWFs including 1) High Fan-in Majority Function (HFM), 2) Weighted High Fan-in Majority Function (WHFM), and 3) WHFM with frequency multiplexing.

#### A. High Fan-in Majority Function (HFM) SPWF

Fig. 3 shows the SPWF-based schematic of a HFM. Here, all inputs ( $I_0 \dots I_{n-1}$ ) are of equal amplitude but have different phases. A simple superposition of the waves would yield a majority function at the output node. Furthermore, by using a control signal  $C(A)$  whose amplitude is modulated by an ME cell, different Boolean logic operations may be obtained. A circular arrangement of inputs may be used in this case for signal attenuation purposes: spin-waves travel an equal distance before interacting and are therefore attenuated by the same amount, leading to correct superposition. It must be noted that in this HFM, the delay would depend purely on the propagation distance and not on the number of inputs.

Fig. 4 shows how additional functionality may be obtained from HFM devices by modulating a control input. An example of high fan-in Boolean function is shown. Phase shifters are used to manipulate the phase of the control and output spin waves.

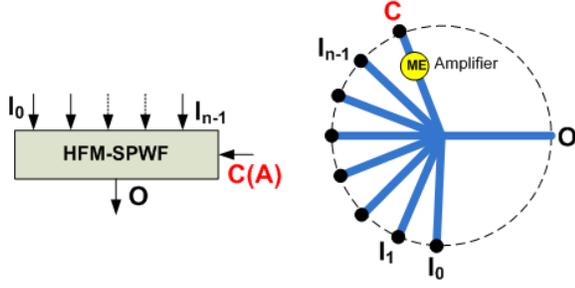


Figure 3. High Fan-in Majority Function Device (HFM) (left) Block diagram, (right) Schematic representation with waveguides and ME amplifier for control.

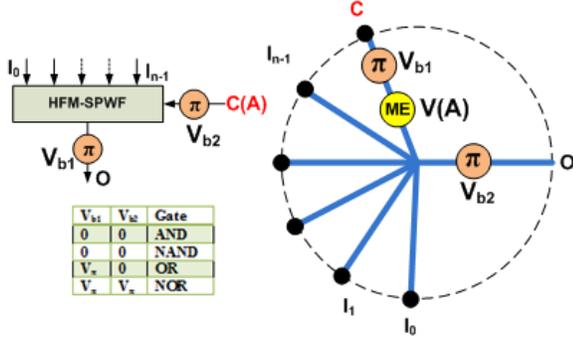


Figure 4. Implementation of majority and boolean functions using a HFM. Bias voltages on phase shifters and ME cells may be adjusted for dynamic reconfiguration of HFM.

By changing the bias voltage on the phase shifters ( $V_{b1}$ ,  $V_{b2}$ ) as well as the control ME cell ( $V_A$ ), dynamic reconfiguration of the HFM is possible. As the number of inputs scale, we would still need only one control input for configuration, albeit with higher amplification.

While high fan-in gates based on capacitive threshold logic have been proposed [11], such logic style has significant drawbacks due to difficulty in implementing accurate capacitors and sensitivity to soft errors. Some emerging technologies have proposed majority function logic gates. These include quantum dot cellular automata (QCA) [12], magnetic QCA [13], etc. All these have a very limited fan-in due to limited range of interactions and cannot be frequency multiplexed.

### B. Weighted High Fan-in Majority (WHFM) SPWFs

In the HFM SPWFs, all the interfering spin waves have equal amplitude; by varying the amplitude of the control input arbitrary functions could be implemented. Additional functionality may be obtained by modulating the amplitudes of the input spin waves for realization of a weighted majority function.

Fig. 5 shows the schematic of such a WHFM: in addition to high fan-in, these functions also have weighted inputs that alter the final result. For example, the schematic arrangement shows a linear topology for WHFM that implements the same functionality as in Fig. 3. In this case, inputs furthest from the output are expected to attenuate more towards the point of

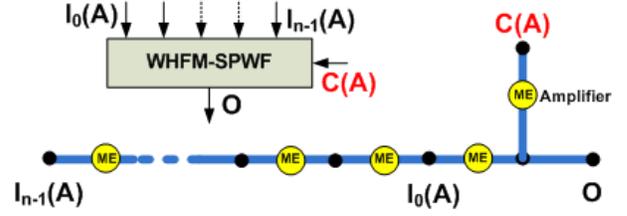


Figure 5. Weighted High Fan-in Majority Function Device (WHFM). (top left) Block diagram (bottom) Schematic representation with linear waveguide and ME cells for input amplification.

interaction, and are therefore amplified more. Inputs closer to the output node are amplified less. Thus a different topology for the device may be employed in conjunction with amplification of inputs to achieve high fan-in majority and Boolean logic functions.

### C. Frequency Multiplexed WHFM SPWFs

Fig. 6 shows the block diagram of a Frequency Multiplexed WHFM-SPWF. The general idea here is that multiple inputs of different frequencies can be simultaneously transmitted and evaluated over the ferromagnetic waveguide. The spin wave interference is frequency dependent and thus the information on individual spin waves is preserved. Similarly by multiplexing the control input, we can realize a large number of sophisticated functions simultaneously.

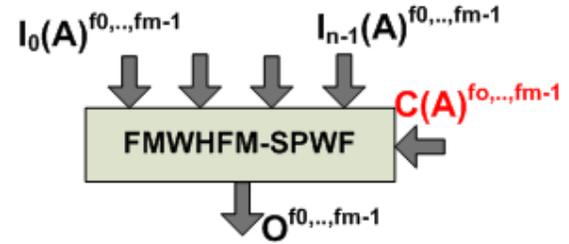


Figure 6. Block diagram of Frequency Modulated Weighted High Fan-in Majority Function Device (FMWHFM). Multiple inputs and control signals at different frequencies can be multiplexed over the same waveguide.

### D. Physical Implementation: Proof of Concept

This section details an experimental demonstration of a majority SPWF. The practical realization of the multi-input majority function devices requires manipulation of the amplitude and phases of a number of spin waves reaching the point of interference. Fig. 7 shows a five-terminal spin wave test structure used in the experimental study of the prototype majority gate, for which experimental data illustrating spin wave interference has been obtained. The material structure from the bottom to the top consists of a silicon substrate, a 300nm thick silicon oxide layer, a 20nm thick ferromagnetic layer made of  $Ni_{81}Fe_{19}$ , a 300nm thick layer of silicon oxide and a set of five conducting wires on top. The distance between the wires is  $2\mu m$ . Each of the five wires can be used as an input or an output port. In order to demonstrate a three-input one-output majority gate, three of the five wires were used as input ports, and two other wires were connected in a

loop to detect the inductive voltage produced by the spin wave interference.

The plot in Fig. 8 shows the output inductive voltage detected for different combinations of spin wave phases. An electric current passing through each wire generates a magnetic field, which, in turn, excites spin waves in the ferromagnetic layer. The direction of the current flow (the polarity of the applied voltage) defines the initial spin wave phase. The curves of different color in Fig. 8 depict the inductive voltage as a function of time for different combinations of the spin wave phases (e.g. 000, 010, 011 and 111). The results show that, phase of the output inductive voltage corresponds to the majority of phases of the interfering spin waves. The data are taken for 3GHz excitation frequency and at bias magnetic field of 950e (perpendicular to the spin wave propagation). All measurements were accomplished at room temperature.

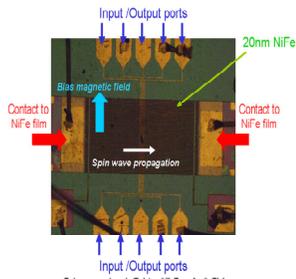


Figure 7. Image of the prototype spin wave device for majority functions.

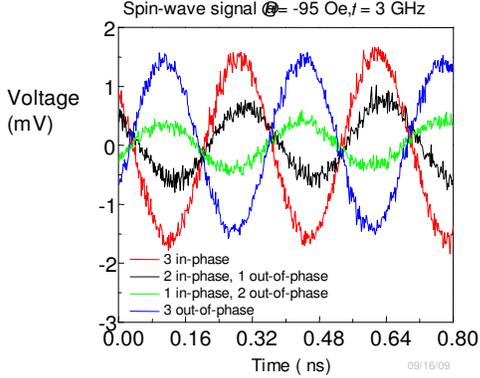


Figure 8. Experimental data illustrating device operation. The frequency of operation is 3GHz. All data are measured at room temperature.

#### IV. PROJECTED BENEFITS OVER STATE-OF-THE-ART SCALED CMOS: INTIAL EXPLORATION

The SPWFs enable the practical implementation of computational units accomplishing complex logic functions such as high fan-in majority function. These computational units may then be leveraged to build larger components (for e.g. arithmetic units). Using SPWFs represents a fundamental shift in mindset: high fan-in arithmetic logic may be implemented with fewer logic levels and lower delays than conventional designs. New algorithms can be devised to take advantage of the various SPWFs.

In this section we present the projected benefits of this new computational paradigm over state-of-the-art scaled CMOS; we analyze a carry-look-ahead (CLA) adder and a counter circuit. Many other arithmetic circuits and cryptographic algorithms would also similarly gain from this paradigm.

The main idea behind a carry-look-ahead addition is an attempt to generate all intermediate carries in parallel. The operation of a CLA adder, is dependent on the carry generate (G) and propagate (P) signals.

The main limitation for realizing high bitwidth adders in conventional CMOS circuits is the fan-in of conventional logic gates (e.g., AND/OR gates). For example, fan-in is being limited to 3 or 4 in 45nm technology [14] and expected to be similarly limited in ITRS-projected scaled CMOS; therefore, the CLA circuitry would need to utilize several levels of carry-look-ahead generators, leading to large delays.

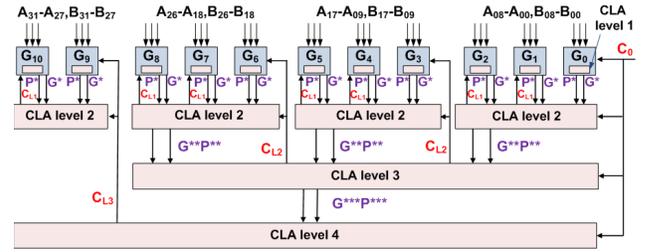


Figure 9. A 32-bit CLA adder in CMOS using 4 levels of CLA generation.

Fig. 9 shows the implementation of a 32-bit adder using a hierarchy of look-ahead circuits and exchange of carry, G and P signals between levels. The G and P signals are passed to the lower CLA levels. The carries are then passed back to the higher CLA levels. The number of such exchanges will increase as the number of bits to be added increases, contributing to higher delays as the bitwidth scales. In general,  $m = \log_k N$  is the number of CLA levels that would be required for an N bit adder where 'k' is the blocking factor [16]. The blocking factor 'k' depends on the maximum fan-in of the available gates for a particular implementation. e.g., if  $k = 3$ , then a 64-bit CLA adder would have four levels of CLA; 128-bit CLA adder would need five levels and so on.

TABLE I. CLA ADDER DELAY IN TERMS OF UNIT LOGIC GATE DELAY

Steps	Delay
Individual bitwise G&P	$\Delta g$
"Group Generate" to lower CLA levels	$(\log_k N - 1) * 2 * \Delta g$
Carries from CLA units	$\log_k N * 2 * \Delta g$
Final Sum	$2 * \Delta g$

Table I summarizes the delays of CMOS-based implementation as a function of the unit logic gate delay  $\Delta g$  (corresponding to delay of CMOS gate with fan-in of 'k'). The overall delay is the sum of all the components described in Table I. A logarithmic increase in delay is expected for CMOS-based implementations with increase in number of levels.

For the HFMG-SPWF adder design utilizing high fan-in gates, the  $G$  and  $P$  components will require one  $\Delta g$ . The generation of  $C$  and the calculation of final sum will each require  $2\Delta g$ . Since propagate can be overlapped with generate, a total delay of  $5\Delta g$  for the CLA adder is obtained.

Fig. 10 shows the factor of increase in number of logic gate levels for CMOS-based implementation of CLA adders. As expected, the graph shows that larger gains (with respect to number of logic gate levels) are obtained with increasing bitwidth. For example, 5.8X more logic gate levels would be required for a CMOS 1024-bit adder and 5X more logic gate levels would be required for a CMOS 256-bit adder vs. equivalent SPWF-based versions. In these experiments, the 45nm CMOS-based implementations have individual gates with a maximum fan-in of 3, thus a blocking factor of 3 is used ( $k=3$ ).

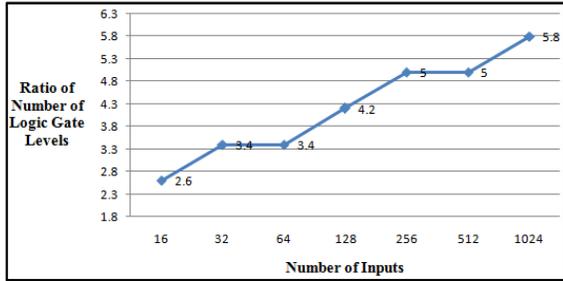


Figure 10. Ratio of required number of logic levels for CMOS ( $k=3$ ) vs. SPWF based implementations of CLA adders.

The other interesting aspect is to compare actual delay numbers for CMOS and SPWF-based implementations of the CLA adder. Fig. 11 shows delay in picoseconds for CMOS and SPWF implementations. Fig. 12 shows the performance improvements over CMOS for various input sizes of adders and for different feature sizes of the SPWF. The delay of SPWFs is independent of the number of the inputs, but it depends on the distance traversed by the spin waves before interference. This would be limited by the particular manufacturing process that is used which determines the feature size metric for the SPWFs. The comparison is based on state-of-the-art CMOS. As the manufacturing requirements are similar (probably more restrictive for CMOS), one can assume at least similar scaling for SPWFs as for CMOS in the future.

A delay of 60ps for a 3-input gate (based on HSPICE simulations) is used to compute the overall delay in the 45nm CMOS version. The delay for a SPWF is purely dependent on the group velocity of the spin waves and the distance traversed. Experiments have demonstrated a 500ps delay for a 5 $\mu$ m propagation distance [10]. Based on these delay values, the overall delay of various CLA adders is shown in Fig. 11. The graph indicates that the SPWF delay is bitwidth independent.

The corresponding performance benefits are reflected in the Fig. 12 and show that even a 2.5 $\mu$ m HFM-based implementation is faster for 128 or more inputs than the 45nm CMOS version. The 45nm SPWF version is estimated to be about 77X faster for input widths equal to or greater than 1024 as indicated by Fig. 12.

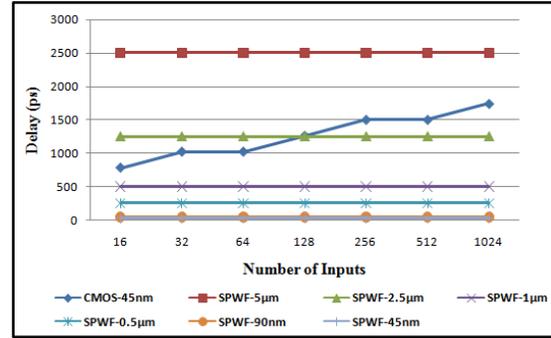


Figure 11. CLA adder delay in picoseconds as a function of input bitwidths for CMOS and SPWF-based implementations.

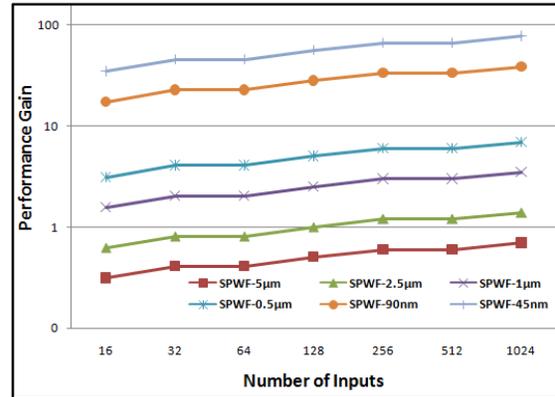


Figure 12. Performance gain over 45nm CMOS for different feature sizes of SPWFs.

## V. DISCUSSION – LOGIC COMPLEXITY

High fan-in structures can provide significant performance benefits for applications with a high degree of parallelism (e.g., single instruction multiple data (SIMD) applications). However, high levels of parallelism may not be available for certain applications (e.g., applications with bit-chain dependencies). In those cases performance benefits of SPWFs may not scale significantly with bitwidth. However, SPWFs may enable significant reductions in the number of logic gates used as well as simplify interconnect requirements.

For example, consider a  $(n, \log_2(n+1))$  parallel counter that counts the number of logic '1's in an  $n$ -bit input and yields  $\log_2(n+1)$  outputs. Parallel counters are commonly used in fast multipliers' implementation. Optimized CMOS implementations of this counter have been shown in [15] for different values of  $n$  and the design complexity has been quantified in terms of the number of logic gates needed. Designing these gates with WHFM provides a 2X performance gain per unit gate delay. More significantly, as shown in this section, it considerably reduces logic complexity.

Fig. 13 shows the implementation of a  $(7, 3)$  counter using WHFM. This design consists of three WHFM devices implementing majority functions (no control bias). Phase shifters are used for inversion, and ME cells for weighing

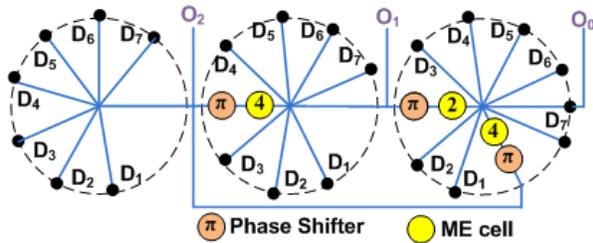


Figure 13. Implementation of (7,3) parallel counter using three WHFM devices. ME cells provide amplification, with the amplification factor shown. Phase shifters are used to invert outputs.

outputs (numbers in ME cells represent the amplification factor). The WHFM implementation requires only 3 logic devices, as opposed to 12 logic gates needed for an optimized CMOS (7, 3) parallel counter [15].

Table II shows how logic complexity scales for WHFM and CMOS versions of the parallel counter. In the WHFM implementations, the number of logic gates is always equal to the number of outputs (i.e., it scales as the log of the number of inputs). As input width scales, significant reductions in design complexity are obtained. (e.g., 8X fewer logic gates for (15, 4) and 15X fewer logic gates for (31, 5) counter). Performance improvements in conjunction with reductions in the logic complexity could enable new types of algorithms and efficient computational models in the future.

TABLE II. PARALLEL COUNTER DESIGN COMPLEXITY IN TERMS OF NUMBER OF GATES

Counter	Number of Gates CMOS	Number of Gates WHFM
(3,2)	3	2
(7,3)	12	3
(15,4)	33	4
(31,5)	78	5

## VI. CONCLUSION

New devices that simultaneously process a large number of inputs and implement more functionality in a single computational step compared to simple switches were discussed. These devices use the wave-based physical phenomena, e.g., the phase of spin-waves for information encoding, and interference and superposition for simultaneous processing of a large number of signals. More sophisticated SPWFs utilizing amplitude modulation, frequency multiplexing and modifying control schemes and topology were discussed. An experimental demonstration of a 5-input spin wave majority device was shown. Performance benefits for SPWF-based logic implementations over equivalent CMOS designs for carry-

look-ahead (CLA) adders were quantified. A 45nm SPWF-based CLA adder is estimated to be at least 77X faster than an equivalent CMOS design for a bitwidth of 1024. Complexity reduction is highlighted in a counter circuit showing that even when performance gains are limited by bit-level dependence chains complexity is significantly reduced. The new direction outlined in this paper has the potential of significantly impacting and altering assumptions in important areas of digital computation including cryptography.

## REFERENCES

- [1] S. Datta and B. Das, "Electronic analog of the electro-optic modulator," *Appl. Phys. Lett.*, vol. 56, pp. 655-667, 1990.
- [2] C. Moritz, T. Wang, P. Narayanan, M. Leuchtenburg, Y. Guo, C. Dezan, and M. Bennaser, "Fault-Tolerant Nanoscale Processors on Semiconductor Nanowire Grids," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 54, 2007, pp. 2422-2437.
- [3] Z.F. Wang, H. Zheng, Q.W. Shi, J. Chen, "Emerging nanocircuit paradigm: Graphene-based electronics for nanoscale computing," *IEEE/ACM Symposium on Nanoscale Architectures 2007*, 2007.
- [4] P. L. McEuen, et al., "Single-Walled Carbon Nanotube Electronics," *IEEE Trans. Nanotechnology*, vol. 1, no. 1, pp.78-85, 2002.
- [5] T. Wang, P. Narayanan, and C. A. Moritz, "Heterogeneous 2-level Logic and its Density and Fault Tolerance Implications in Nanoscale Fabrics," *IEEE Trans. on Nanotechnology*, vol. 8, no. 1, pp. 22-30, January 2009.
- [6] D. B. Strukov and K. K. Likharev, "Reconfigurable Hybrid CMOS Devices for Image Processing", *IEEE Transactions on Nanotechnology*, vol 6, pp. 696-710, November.
- [7] G. S. Snider and R. S. Williams, "Nano/CMOS architectures using a field-programmable nanowire interconnect", *Nanotechnology*, vol. 18, pp. 1-11, 2007.
- [8] K. Galatsis et al., "Alternate State Variables for Emerging Nanoelectronic Devices," *IEEE transactions on Nanotechnology*, vol. 8, pp. 66-75, 2009.
- [9] T. Schneider, A. A. Serga, B. Leven, B. Hillebrands, R. L. Stamps, and M. P. Kostylev, "Realization of spin-wave logic gates," *Appl. Phys. Lett.*, vol. 92, pp. 022505-3, 2008.
- [10] A. Khitun, M. Bao, and K. L. Wang, "Spin Wave Magnetic NanoFabric: A New Approach to Spin-based Logic Circuitry," *IEEE Transactions on Magnetics*, vol. 44, pp. 2141-53, 2008.
- [11] H. Ozdemir, A. Kepkep, Y. Leblebici, U.Cilingiroglu, "A Capacitive Threshold-Logic Gate," *IEEE Journal of Solid-state Circuits*, vol. 31, no. 8, August 1996.
- [12] P. Tougam and C. Lent, "Logical devices implemented using quantum cellular automata," *J. Appl. Phys.*, vol. 75, pp. 1818-1825, 1994.
- [13] A. Imre, G. Csaba, L. Ji, A. Orlov, G. Bernstein, and W. Porod, "Majority logic gate for magnetic quantum-dot cellular automata," *Science*, vol. 311, no. 5758, pp. 205-208, 2006.
- [14] International Technology Roadmap for Semiconductors, 2009 new edition. Available online at <http://public.itrs.net/>.
- [15] S. Veeramachaneni, L. Avinash, M. K. Krishna, M.B. Srinivas, "Novel architectures for efficient (m, n) parallel counters," *Proceedings of the 17th ACM Great Lakes symposium on VLSI*, 2007.
- [16] I. Koren, *Computer Arithmetic Algorithms*, 2nd Edition, A. K. Peters, Natick, MA, 2002.