Crosstalk Logic Circuits with Built-in Memory

Authors
USA
E-mail:

Abstract—**Memory and Logic computing beyond Von Neumann architecture is of paramount importance in Artificial Intelligence data-intensive applications. To this end, we propose a novel concept of Logic with Built-in Memory. We intend to combine Crosstalk logic, a novel computing method that harnesses deterministic interference between interconnect metal lines for logic calculations, with data storage elements to generate neoteric Crosstalk circuits capable of storing the state of the computed data. In this paper, we present how Crosstalk Computing can be leveraged to implement basic and complex circuits with built-in memory at 16 nm technology node. We show a range of circuits and discuss both logic and memory characteristics through circuit simulations. In addition, we discuss comparison results with equivalent CMOS circuits. Our results indicate a 32% reduction in transistor count for a full adder (FA) circuit with respect to CMOS at 16nm. The average power for a FA is 3μW and the maximum propagation delay is 58 ns.**

Keywords—**Crosstalk, Built-in Memory, Flip-Flop (FF), Crosstalk Logic (CL), PMOS, NMOS, Complementary MOSFET (CMOS)**

I. INTRODUCTION

One of the essential components of computing is the storage element or memory functions. The conventional Von-Neuman Architecture incurs high computational power in fetching data between the computational unit and memory in Machine Learning and Deep Learning applications. As technology shrinks below 5 nanometers (nm), circuits having integrated computing and memory functions form pillars of parallel computing. An Integrated Circuit (IC) chip consists of combinational logic along with sequential logic like flip-flops, latches, and registers to store the state of computed logic for one clock cycle. Flip-flops (FF) consume almost 50% of the area and power on an SoC (System-on-Chip). With extensive pipeline techniques used for parallel computing, there is a considerable increase in the number of flip-flops/registers. The traditional transmission-gate based D-FF consumes around 22 transistors and with clock connected to 8 transistors makes it power-hungry [1]. Many circuits have been proposed to reduce area, power, and improve on performance like the Semi-dynamic FF, Pulse-Triggered FF, Sense Amplifier Based FF (SAFF), Topologically Compressed FF (TCFF), Logic structure Reduction Flip-Flop (LRFF), Dual dynamic node FF (DDFF), etc. compared to the traditional Transmission Gate FF (TGFF) [1-8]. Logic embedding feature has been demonstrated by few as discussed in the literature survey but embedding the complex combinational circuits in FFs is ambitious as it faces issues like latency, charge sharing, and area-overhead of FFs. We propound to combine the combinational logic with area-efficient memory elements to get combinational circuits with a built-in flip-flop/register. This reduces the need for additional flip-flops and ultimately the load on the clock.

*This work was done as a part of PhD thesis

In this paper, we propose to use the datum of a new computing paradigm known as Crosstalk Computing with the proposed Built-in Memory to implement basic and complex logic circuits. In our previous papers [9-16], we have validated that the crosstalk can be leveraged to implement basic circuits- NAND, NOR, OR, AND, XOR, complex circuits- AO21, OA21, Full-Adder, MUX, and truly polymorphic circuits like Multiplier-Adder-Sorter. The lateral coupling capacitance between the closely spaced nets, acting as inputs and the charge induced on the victim node, conditioned as output, determines the logic implemented. The reduction in transistor count offered by Crosstalk Polymorphic Circuits ranges from 25% to 83% than other approaches [10]. This paper introduces Crosstalk Built-in Memory Logic Circuits (CBML), which can retain the states of the computed logic irrespective of the change or absence of inputs. The states change only when a discharge cycle appears. The logic circuits are implemented using positive and negative Crosstalk gates [14]. The paper is organized as follows. Section II discusses various FF architectures through a literature survey. Section III enumerates the basic Crosstalk Circuits and the proposed Built-in Memory block. A pull-up and pull-down feedback network of NMOS and PMOS forms the memory block in the Crosstalk (CL) circuits. Section IV elaborates the 2-input CBML with simulations and presents the Cascaded CBML where a Full-Adder is implemented with Built-in Memory and a minimum number of transistors. Section V compares the CBML with other D-FFs in terms of transistor count and Section VI concludes the paper.

II. DIFFERENT FLIP-FLOP ARCHITECTURES WITH EMBEDDED LOGIC

Over the past decade, a multitude of architectures have been designed for power-efficiency, fast performance, and low area with respect to the traditional Transmission-Gate static D-FF (TGFF). TGFF uses 22 transistors, and its high capacitive loading causes huge dynamic power consumption [1]. The Semi-dynamic Single-Phase Pulsed FF (SDFF) has a small clock load, reduced latency, and logic embedding capability but with a delay penalty over the conventional pulsed-FF [2]. The pulsed flip-flops use narrow pulses derived from the clock locally. It is truly dynamic in nature and imposes all disadvantages of dynamic FF. The SDFF adds back-to-back inverters, buffers the output, and adds a conditional shut-off circuit to the pulsed FF to improve metastability and noise sensitivity. Delicate Pulse width control and its distribution is the drawback of SDFF. Implementing complex embedded logic in it increases the NMOS stack and results in increased latency. The Implicit-Pulse Semi-dynamic (ip-DCO) FF [8] has reduced transistor count faster than the SDFF. But it has a high capacitive load to the internal node and degrades the overall performance. Implicit pulse-triggered FF with a pulse-control scheme in [3], reduces the stacked NMOS transistors in the previous pulsed-FF designs and proposes a new in-built pulse generation scheme. It suffers from a longer hold time.

The Sense-Amplifier-based FF (SAFF) [4] has transistor overhead compared to SDFF to meet high-frequency

performance, near-to-zero setup time, and improved hold time. It does not have the built-in logic and requires a separate sense-amplifier stage along with the latch stage. The TCFF [5] uses a single phased clock, fewer transistors than TGFF, and is low-powered FF but with increased setup time due to its weak pull-up network. If logic is to be implemented with TCFF, it will drastically increase the transistor count. The LRFF discussed in [6], is an enhancement to TCFF with logic re-structured, circuit optimized and eliminating floating node. The drawback of LRFF is the lesser hold time which limits the maximum propagation delay of the combinational block in the pipeline structure. The DDFF-ELM [7] implements the Flip-Flop with embedded logic, but as the circuit gets complex, the charge sharing becomes uncontrollable with an increase in the NMOS stack. Hence, circuits like Full-Adders will incur additional transistors to cope with charge sharing.

III. PROPOSED BUILT-IN-MEMORY CONCEPT

A. Basic Crosstalk Logic

The Crosstalk Logic (CL) forms the premise for the proposed CMBL circuits. Fig. 1 shows the basic CL gate. The aggressor nets (inputs to the gates), upon transition, induce a voltage on the victim node (V_i) based on the strength of the mutual capacitances (CC) between them. The net voltage on V_i net depends on its engineered coupling network and the inverter depending upon its logic threshold voltage (V_{TH}) decides the logic level. If $V_i < V_{TH}$, V_i is at Logic 0, and if $V_i > V_{TH}$, V_i is at Logic 1. CL gates operate in two phases, Discharge State (DS) and logic Evaluation State (ES). During DS state, the V_i and inputs are initialized to 0 for Positive CL (PCL) circuits (Fig. 1.a) and 1 for Negative CL (NCL) circuits (Fig. 1.a). The outputs will be impacted accordingly during the Discharge phase. Here, V_i is initialized by discharge transistor controlled by Discharge (*Dis*) signals as shown in Fig 1.a&b and inputs are initialized by their previous stage gates or

B. Enabling Memory in Crosstalk Logic gates

Using duals, PCL and NCL circuits, in parallel are called Dual CL (DCL) gates. The DCL circuits connected with Memory Enabler (ME) circuit in a special circuit topology can give inherent memory feature to crosstalk circuits, thus implements Crosstalk Built-in Memory Logic (CBML) circuits. The ME in a PCL and NCL consists of PMOS and NMOS ladder as shown in Fig. 2.a&b. Its functionality is to statically pull-up (PU) or pull-down (PD) the logic at the victim node during the evaluation phase, depending upon the logic intended on the victim node V_i. The positive CL (Fig. 1.a) and negative CL (Fig. 1.b) are connected in parallel, each consisting of ME. The feedback (Fp) to the $P2$ PMOS is from the inverter output of the positive CL and the feedback (Fn) to the $N2$ NMOS is from the inverter output of the negative CL in both circuits. At a particular time, either PU or PD logic is activated. CL gates have weak logic levels (non-full swing voltage levels) on Vi net due to its nature of computation. If V_i is at weak Logic 1, PU logic is activated through $P2$ PMOS, and if it is at weak Logic 0, PD logic is activated through $N2$ NMOS. Thus, a memory-loop/latch with all statically driven nodes is established in the circuit giving rise to the data storage feature. This data latched state can be released during the discharge phase by deactivating the pull-up and pull-down branches of the ME circuit. PCL and NCL circuits complement each other in latching the evaluated logic equal to 0 or 1. Thus, the sequence of operations performed by the CBML circuit in the evaluation phase are, activation of ME branches, evaluation of crosstalk logic through signal interference (both for PCL and NCL), latch the evaluated logic output through ME circuits in PCL and NCL. The latched data is retained in this static loop state for the entire evaluation phase regardless of the change or absence in inputs as the V_i is no longer floating. The new inputs for the next round of computation are considered after the discharge phase.
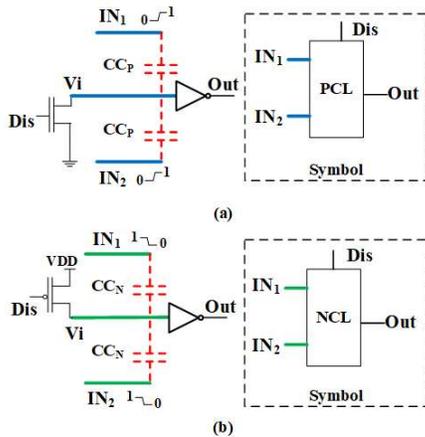

Fig 1. Basic Crosstalk Circuits and their symbol (a) PCL (b) NCL

special initializer circuits consisting of transmission gate and a pull-up PMOS [14]. The major difference is that PCL circuits operate on 0-to-1 transition whereas NCL circuits operate on 1-to-0 transition. During the evaluation phase, the discharge signal is deactivated, and the logic executes. Note that during the discharge phase V_i is floating for accurate crosstalk computation. The inverter is used for signal conditioning and for setting the V_{TH}.
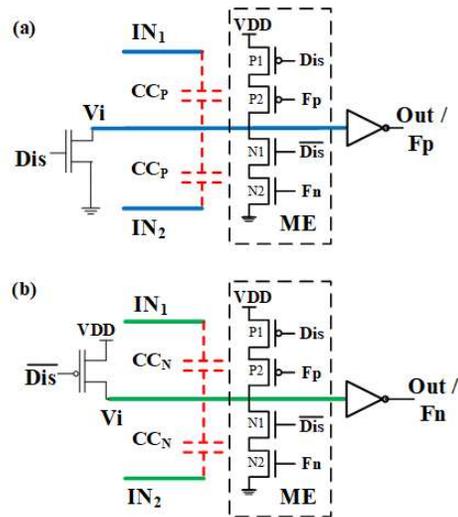

Fig. 2. Memory Enabler circuit integrated with (a) PCL (b) NCL

It should be noted that memory feature in CBML circuits offer a novel kind of edge-sensitive memory without requiring redundant Master-Slave latches as in case of traditional Flip-Flops with an added benefit of computation in memory. Few implementations of CBML logics are presented in the next section.

IV. Building Crosstalk Built-in-Memory Logic Circuits

A. Basic 2-Input CBML Circuits

Fig. 3 shows the 2-input CBML circuit for an AND and an OR gate. NCL and PCL are connected in parallel. A second inverter is added only at the output of PCL to function it as AND/OR gate. Each The output of the first inverter of the PCL goes as feedback to both PCL and NCL as explained in the previous section and can also be used as $\overline{out}$. The second inverter output is the intended logic (out). When ($Dis = 1$), PCL is initialized to Logic 0 and since, $\overline{Dis} = 0$, NCL is initialized to Logic 1. This is the discharge phase. During the evaluation phase ($Dis = 0$), the logic executes. The inputs (acting as aggressor nets) induce a voltage on the V_i, after transitioning from low to high in PCL and from high to low in NCL. The ratio of the size of PMOS to NMOS ($P:N$) of the first inverter and the coupling capacitances determines whether it is an AND or an OR logic. During the evaluation phase, the PU and PD branches in ME ensures that there are no weak logic levels on the V_i. The feedback from PCL pulls strong/static 1 and NCL pulls strong/static 0 on V_i for both PCL and NCL gates; those they aid each other to staticize their V_i nets and bring in the memory feature. Also, all PMOS and NMOS transistors in the circuit need to be sized to achieve the intended logic and memory functions. The capacitance, transistor-sizing, and PU-PN logic values are displayed in Table I for AND and OR gate, respectively.

TABLE I.   CROSSTALK COUPLING, TRANSISTOR SIZING, AND PU-PD VALUES FOR AND, OR GATE

Crosstalk Logic	Gate	CC (F)	Pull-up/Pull-Down Logic				Width Ratio (P:N)
			P1	P2	N1	N2	
Positive	AND	600a	3	3	1	1	1:1
	OR	700a	2	2	1	2	1:3
Negative	AND	700a	1	2	2	2	3:1
	OR	600a	3	3	1	1	1:2

The simulation results are shown in Fig. 4 and Fig. 5. When ($Dis = 0$), the victim nets $V_{i\_}p$ and $V_{i\_}n$, hence corresponding outputs, are at Logic 0 and 1, respectively. When the inputs transition, A (0 to 1) and B (0 to 1), voltage is induced on the $V_{i\_}p$ whereas the $V_{i\_}n$ remains at Logic 1 and the output obtained is 1. For OR gate if any of the input transitions from 0 to 1, the output is 1 whereas for AND gate, only when both transition to level 1, the output is 1. The inputs are changed in every evaluation cycle to depict the memory feature. In Fig. 4 it can be seen after 16 ns in absence of inputs, the value is retained perpetually till the discharge cycle. The logic execution and memory retention happen simultaneously. Similarly, in Fig. 5, the OR logic can hold the output indefinitely. This validates the logic circuits embedded with memory. The same circuit implements the NAND and NOR logic with $\overline{out}$ as the output. This will further reduce the transistors of an inverter.
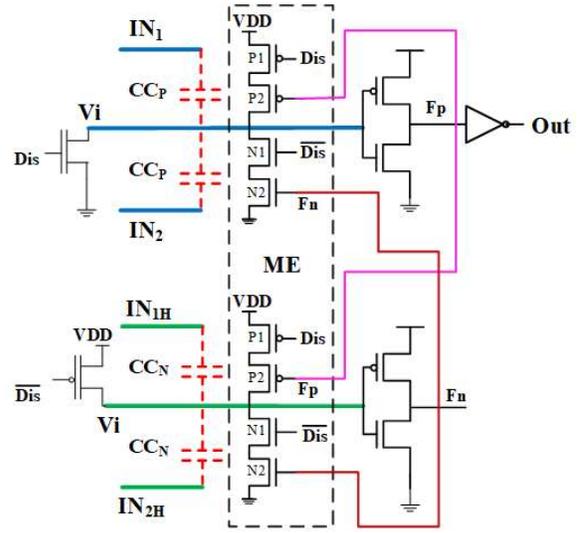
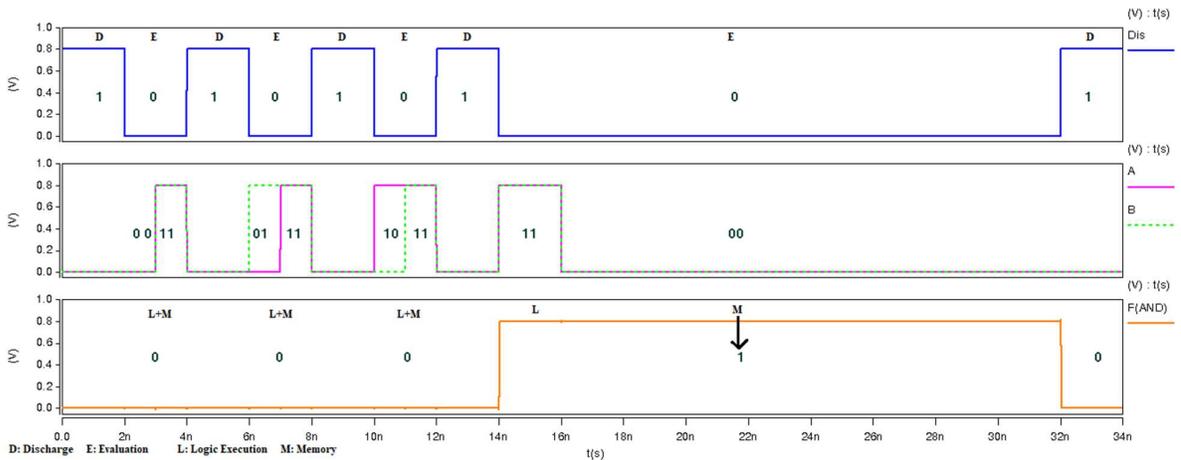

Fig. 3. CBML circuit for 2-input AND and OR gate

Fig. 4. Simulation of Crosstalk Built-in Memory AND Logic. D: Discharge, E: Evaluation, L: Logic Execution and M: Memory
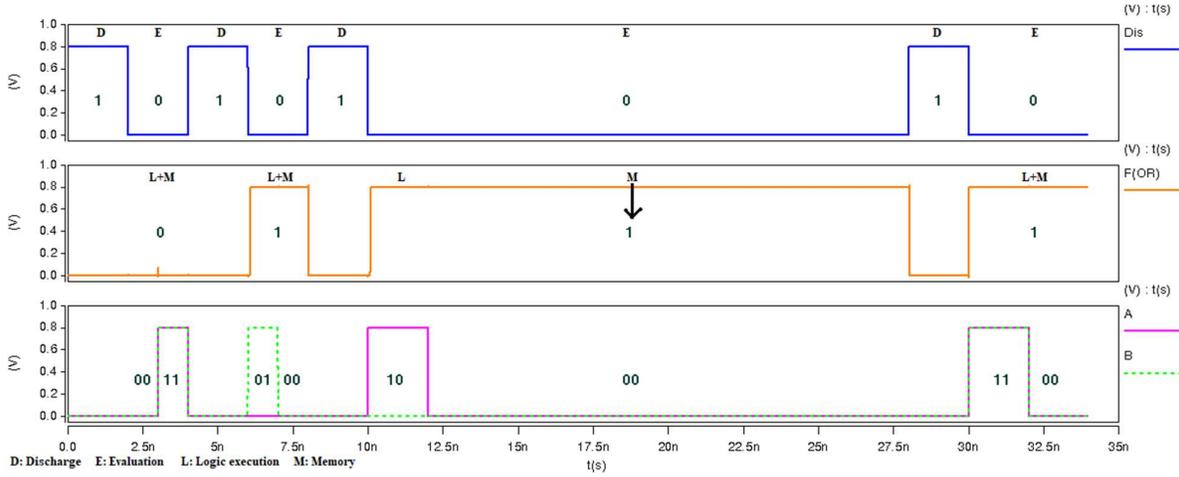
Fig. 5. Simulation of Crosstalk Built-in Memory OR Logic. D: Discharge, E: Evaluation, L: Logic Execution and M: Memory

B. Cascaded CBML Circuits

Crosstalk circuits have limitations in implementing non-unate functions like $EXOR$ and $EXNOR$ logic within a single stage CL gate. An additional controlling net, a circuit tweak, or a cascaded dual transition logic is essential [14]. In the CBML circuits both PCL and NCL are available and they can be proficiently utilized to implement the $EXOR$ function required by the Full-Adder. A Full-Adder is a basic building block of the Arithmetic and Logic Unit and implementation with CBML results in a significant decrease in transistor count. The $CARRY$ logic ($AB + BC_{in} + AC_{in}$) is implemented using a 3-input CBML circuit. An aggressor net is added for the third input and a mutual capacitance with multiplying factor of K i.e. (KC_{CC}). is connected to V_i. The rest of the circuit is similar to Fig. 3. Table II shows the coupling capacitances and transistor sizes for the Full-Adder. The SUM ($A \oplus B \oplus C_{in}$) operation is implemented by cascading PCL (Fp) of the $CARRY$ CBML with the NCL (V_i) of SUM CBML as shown in Fig. 6. The $\overline{CARRY}(Fp)$ from first inverter output is the control net for the SUM logic with coupling capacitance twice of the CC of the SUM logic.
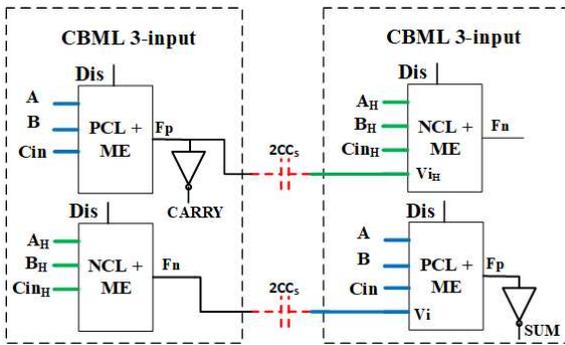

Fig. 6. Cascaded CBM: Full-Adder Block Diagram

The Simulation results are shown in Fig. 7. It can be verified that during $Dis = 0$, the $CARRY$ logic output shown in panel 3 is 1 only when two or more inputs are one. The SUM output is shown in panel 2. The circuit's inherent memory feature is verified by changing inputs during the evaluation phase as seen in panel 4. E.g. at 30 ns, the inputs are 111 which gives both SUM and $CARRY$ output = 1. When

inputs are switched OFF i.e., 000, the outputs are retained at Logic 1.

TABLE II.   CROSSTALK COUPLING AND TRANSISTOR SIZING FOR FULL- ADDER

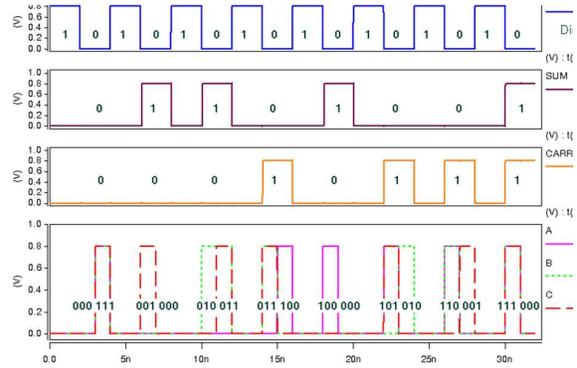| CL	Gate	CC (F)	K	Pull-up/Pull-Down Logic				Width Ratio (P:N)
				P1	P2	N1	N2	
PCL	CARRY3	600a	0.67	1	1	3	3	1:2
	SUM	500a	1	1	1	3	3	1:2
NCL	CARRY3	600a	0.67	1	2	2	2	2:1
	SUM	500a	1	1	1	3	3	2:1

Fig. 7. Simulation of CBM Full-Adder

V. COMPARISON AND ANALYSIS

The Crosstalk Built-in Memory Logic Circuits are compared with the CMOS logic. The CMOS Logic gates are implemented along with the Static Transmission gate D-Flip-flop to add the memory feature to the Logic Gates for a fair comparison. Fig. 8 shows that, as the complexity of circuits increases, i.e., for complex logic, the transistor count required by the proposed CMBL circuits drastically reduces than the CMOS circuits. The reduction in the number of transistors for Full-Adder is 32.8%. This brings down the area and power for implementing combination logic along with the flip-flop. A significant count reduction is observed when compared with the number of transistors for Full-Adder required by the Pulse-Triggered FF, SAFF, TCFF, LRFF, DDFF-ELM along with the respective logic. Here, the transistor count for the logic is
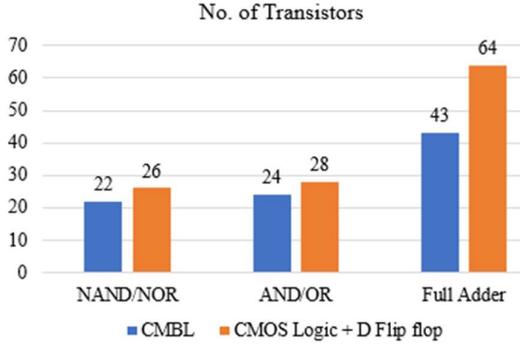
Fig. 8. Transistor Count of Logic Gates and Memory

added with the number of transistors each architecture takes for the implementation. DDFF has an embedded logic feature with transistor count comparable to CBML but faces a charge sharing issue and does not talk about the complex circuit implementation.

The *Dis* signal can be synchronous (with a relaxed period) or asynchronous to the clock depending upon the application and hence, these circuits can reduce the load on the clock tree
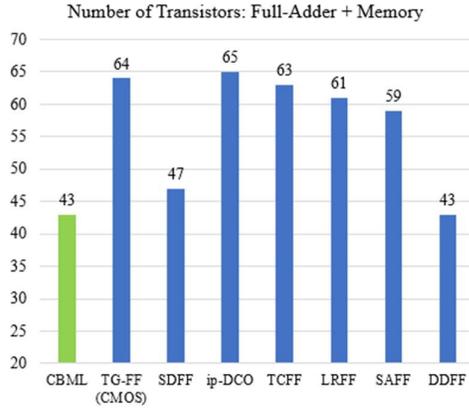

Fig. 9. Comparison of different Flip-Flop architectures

ultimately saving on. power. The power consumption of CBML circuits reduces as the transistor count decreases for the complex circuits. The results for power and performance for the CBML using 16 nm FINFETs are presented in Table III. The memory enabler circuit also helps in reducing the leakage power of the circuits as it staticizes the CBLM. The propagation delay (T_{PD}) of the CBML is two inverter delays.

TABLE III.   POWER AND DELAY FOR CBML

Gate	Avg. Power (W)		Leakage Power (W)	Propagation Delay (Tpd) (ps)	
	CT	CBML	CBML	Min	Max
NAND	2.02u	949n	267n	7.35	
NOR	4.29u	662n	355n	4.3	70
AND	2.06u	983.95n	288n	10.655	
OR	4.36u	749.23n	383n	6.29	79.2
FA	11.9u	3.03u	1.02u	10	58.12

It also depends on the mutual capacitances between the inputs. T_{PD} is maximum when only one input transitions as it takes more time to induce a voltage on V_i and is minimum when all

the inputs transition. Hence, OR logic will see more T_{PD} than the AND logic. Power and Delay for CBML

VI. CONCLUSION

The proposed Crosstalk Built-in Memory Logic circuits employ crosstalk computing phenomena to embed the memory in the logic circuits. These novel circuits offer transistor count reduction for complex gates up to 32% compared to other flip-flop architectures. The average power of FA is 3µW which is less than the normal CT circuits. The maximum propagation delay of FA is 58 ns. Our initial investigations also reveal that the CBML circuits, owing to their novel merged logic and memory feature, could open new circuit design opportunities for special high-speed macros alleviating memory latency issues. Finally, Crosstalk-Computing specific engineered physical structures on the chip could further usher the optimal and efficient implementation of the CBML circuits.

REFERENCES

[1] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2003.

[2] Klass F., et al., "New Family of Semidynamic and Dynamic Flip-Flops with Embedded Logic for High-Performance Processors," IEEE Journal of Solid-State Circuits, vol. 34, no. 5, May 1999

[3] Hwang Y.T., et al., "Low-Power Pulse-Triggered Flip-Flop Design With Conditional Pulse-Enhancement Scheme," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no. 2, pp. 361-366, Feb. 2012

[4] Strollo A., et al., "A Novel High-Speed Sense-Amplifier-Based Flip-Flop," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 13, no. 11, pp. 1266-1274, Nov. 2005

[5] Kawai. N, et al., A Fully Static Topologically-Compressed 21-Transistor Flip-Flop with 75% Power Saving ,"IEEE Asian Solid-State Circuits Conference (A-SSCC), Nov. pp. 117-120, 2013

[6] Lin J.-F., et al., "Low-Power 19-Transistor True Single-Phase Clocking Flip-Flop Design Based on Logic Structure Reduction Schemes," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 7, no. 11, pp. 3033 - 3044, Nov. 2017.

[7] Absel K., et al., "Low-Power Dual Dynamic Node Pulsed Hybrid Flip-Flop Featuring Efficient Embedded Logic," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 21, no. 9, pp. 1693-1704, Sept. 2013

[8] J. Tschanz, et al., "Comparative delay and energy of single edge-triggered and dual edge-triggered pulsed flip-flops for high-performance microprocessors," in *Proc. ISPLED*, pp. 207–212, 2001

[9] Macha N.K., et al., "A New Concept for Computing Using Interconnect Crosstalks," IEEE International Conference on Rebooting Computing (ICRC), 2017, pp. 1-2.

[10] Macha N.K., et al.,"Crosstalk based Fine-Grained Reconfiguration Techniques for Polymorphic Circuits,"IEEE/ACM NANOARCH,2018

[11] Macha N.K., et al., "A New Paradigm for Fault-Tolerant Computing with Interconnect Crosstalks" IEEE Conference on Rebooting Computing, 2018

[12] Iqbal M. A., et.al., "Designing Crosstalk Circuits at 7nm," IEEE International Conference on Rebooting Computing (ICRC), 2019

[13] Iqbal M. A., et.al., "From 180nm to 7nm: Crosstalk Computing Scalability Study," IEEE S3S Conference, 2019

[14] Macha N.K. "Crosstalk Computing: Circuit Techniques, Implementation and Potential Applications," Ph.D. Dissertation, ECE, University of Missouri -Kansas City, MO, USA, 2020. Available: https://mospace.umsystem.edu/xmlui/handle/10355/80941

[15] Macha N.K., et al., "A New Computing Paradigm Leveraging Interconnect Noise for Digital Electronics Under Extreme Environments," IEEE Aerospace Conference, 2019

[16] Iqbal M. A., et.al., "A Logic Simplification Approach for Very Large Scale Crosstalk Circuit Designs," IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH), 2019