Multifunctional In-Memory Computation Architecture Using Single-Ended Disturb-Free 6T SRAM

Chua-Chin Wang, Nanang Sulistiyanto, Tsung-Yi Tsai and Yu-Hsuan Chen

Abstract This paper presents an In-Memory Computation (IMC) architecture using Full Swing Gate Diffusion Input (FS-GDI) in a single-ended disturb-free 6T SRAM. Not only are basic boolean functions (AND, NAND, OR, NOR, XOR2, XOR3, XNOR2) fully realized, a Ripple-Carry Adder (RCA) is also realized such that IMC is feasible without ALU (Arithmetic Logic Unit) or CPU. FS-GDI reserves the benefits of the original GDI, and further resolves the reduced voltage swing issue, but it leads to speed degradation and large static power. Therefore, by using in-memory computing technique, the well-known von-Neumann bottleneck will be mitigated as well as energy efficiency is enhanced.

Keywords In-Memory Computing (IMC) · Single-ended 6T SRAM · Ripple-carry adder · Von-Neumann bottleneck · FS-GDI · Boolean functions

1 Introduction

The pursuit of speed in computing system development has never been changed. However, almost all computing architecture used for computation-intensive applications, such as Artificial Intelligence (AI), biological systems, and neural networks, are based on von-Neumann machines, which separates the storage units (memory) with Arithmetic Logic Units (ALU for computation). Thus, despite the advanced CMOS technology, it still runs into a well-known issue called von Neumann bottleneck [1]. Due to the large amount of data flow between memory and CPU and overhead limitations, many types of solutions have been developed, including IMC [2–5]. The aim of IMC is to bypass von Neumann bottleneck and realize computation in memory arrays directly and locally.

C.-C. Wang (✉) · N. Sulistiyanto · T.-Y. Tsai · Y.-H. Chen
National Sun Yat-Sen University, Kaohsiung 80424, Taiwan
e-mail: ccwang@ee.nsysu.edu.tw

© Springer Nature Singapore Pte Ltd. 2020

Z. Zakaria and R. Ahmad (eds.), *Advances in Electronics Engineering*,
Lecture Notes in Electrical Engineering 619,
https://doi.org/10.1007/978-981-15-1289-6_5

SRAMs, usually as the core of CPU cache, consume a great portion of power. With reference to [6], a 4T load-less SRAM has been proposed and implemented to reduce the power consumption. However, the disturbance of the bit line during read/write data has been pointed out to compromise Static Noise Margin (SNM) [7]. Therefore, a write-assist loop with multi-Vth transistors is presented to ensure the disturb-free feature [8]. Nevertheless, when read/write operations are kept in a long period, the leakage current will destroy the stored data, which needs to be resolved.

Gate Diffusion Input (GDI) technique [9] is a method to relieve two basic problems of Pass Transistor Logic (PTL) circuit. One is the performance degradation from Vth drop, and the other is high power dissipation from half-closed PMOS transistor. Moreover, several boolean functions can easily be expressed by only two transistors. For instance, FS-GDI was revealed to resolve voltage swing hazards [10]. According to the demand mentioned above, a single-ended disturb-free 6T SRAM with IMC architecture utilizing FS-GDI to carry out logic circuit may be a good solution for AI system realizations.

2 SRAM Design with IMC

The proposed single-ended disturb-free 6T SRAM cell with the associated control circuit is shown in Fig. 1. The 6T SRAM cell has been proved to attain the edge of low power and small area. The Control circuit is in charge of generating all the required control signals for the cell. Figure 2 shows an illustrative IMC architecture composed of a 4 × 4 SRAM array, four pre-charged circuits, four MUXs, four RCA unit, and forty-eight 2T switches. Notably, this work also demonstrates a 4-bit Ripple-Carry Adder (RCA) and all the xes in this work (including figures) stand for 0, 1, 2, or 3. Detailed sub-circuits and data flow will be explained below.

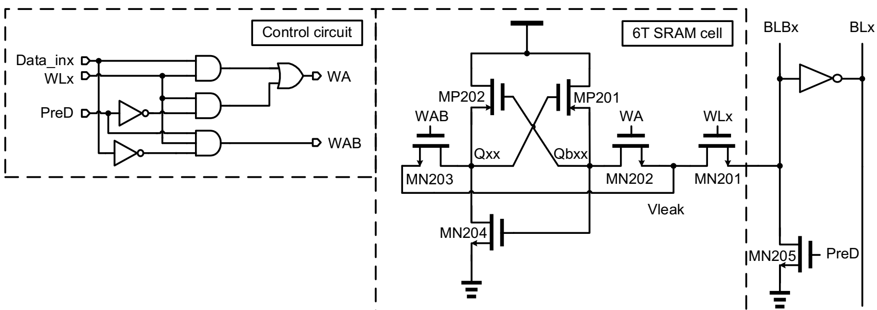

Fig. 1 A 6T SRAM cell with a control circuit. (x = 0, 1, 2, 3)

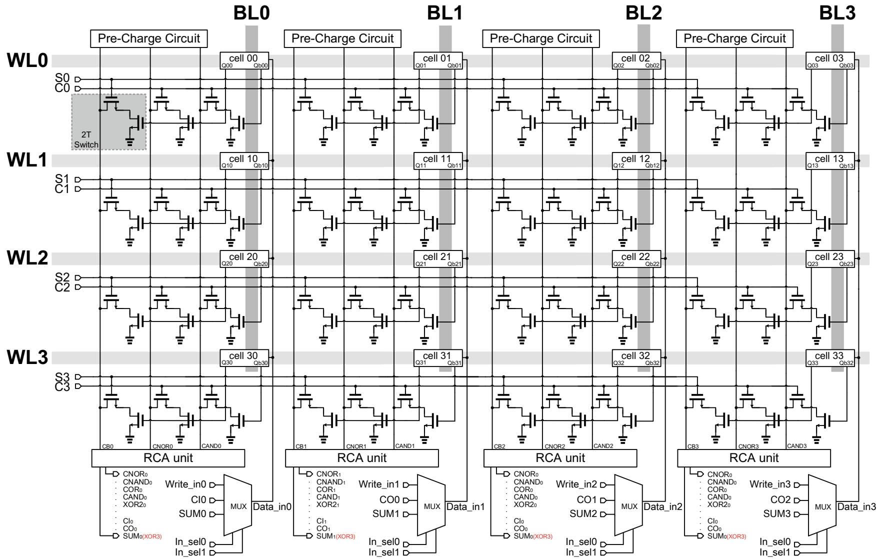

Fig. 2 A 4 × 4 IMC architecture for demonstration

2.1 6T SRAM Cell with Control Circuit

Data_inx in Fig. 1 is the input data to be stored in the cell. PreD is the pre-discharge signal to reset BLBx (BLx). WLx will select which word line to be accessed, and control MN201 to resist the potential disturbance from the bit lines. WA and WAB assist the write operation. If the SRAM cell is realized by the prior 5T SRAM in [8] and Qbxx is logic "1" in read operation, the leakage current will flow through Vleak to Qxx after WA and WLx are switched on. The accumulation on Qxx will soon destroy the data state. Therefore, adding MN204 as a foot switch will fortify the data state on Qxx.

2.2 RCA Unit

RCA unit in Fig. 3 is composed of combinational circuits as well as simplified FS-GDI circuits. Notably, NMOS and PMOS highlighted by grey scale are neglected when one of the inputs are kept coupled to VDD or GND, respectively. Table 1 tabulates detailed logic function in an RCA unit.

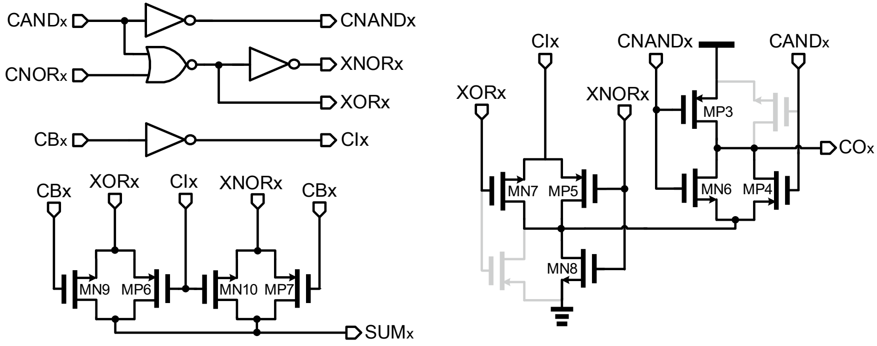

Fig. 3 Combinational logic with simplified FS-GDI. (x = 0, 1, 2, 3)

Table 1 Boolean expressions in the RCA unit

$CBx = \overline{CIx}$	$XORx = \overline{AB + \overline{(A + B)}} = \overline{(A + B)} \cdot (A + B) = A \oplus B$
$CANDx = AB$	$SUMx = (A \oplus B) \oplus CIx = CIx \cdot \overline{(A \oplus B)} + CBx \cdot (A \oplus B)$
$CNANDx = \overline{AB}$	$COx = (A \oplus B) \cdot CIx + A \cdot B$
$CNORx = \overline{A + B}$	

A, *B* Input bit; *CI* Carry in bit; *CO* Carry out bit

2.3 In-Memory Computing Operation

The IMC operation of the proposed design employs the logic operation strategy reported in [11]. Referring to Fig. 2, the pre-charge circuit will charge CBx, CNORx, and CANDx to high level in the first half of every write cycle. Then 2T switches [12], controlled by Sx and Cx signals (x = 0, 1, 2, 3), will store the digital state in Qxx or Qbxx to CBx, CNORx, and CANDx accordingly. Firstly, only one signal among S0 to S3 will be turned on to read Qxx. If Qxx is high, CBx is low. Secondly, two signals among C0 to C3 will be on to carry out NOR function. If one of the selected Qxx is high, CNORx will turn low. Thirdly, by the same procedure as the previous one, two Cx signals will be on to execute the function of AND gate of Qbxx. If both selected Qbxxes are low, CANDx is high. Overall logic function is tabulated in Table 2. Therefore, input signals, CBx, CNORx, and CANDx, will trigger RCA units to compute the summation and carry bit generation.

For the sake of clarity, we demonstrate X (0101) + Y (0110) = Sum (1011). Logic transition waveforms are shown in Fig. 4. Figure 5 shows the data flow of the 4-bit addition, which is a simplified version of Fig. 2. Notably, the stored data is labeled in red. The data transition of cell blocks is labeled in orange (cell 00, 10, 20, 30, and 21), and the detailed description of the addition is listed below.

Table 2 Logic function table of the RCA unit

S0/C0/C1	Q00	Q10	Qb00	Qb10	CBx	CNORx	CANDx
1/−/−	0	–	–	–	1	–	–
1/−/−	1	–	–	–	0	–	–
−/1/1	0	0	–	–	–	1	–
−/1/1	0	1	–	–	–	0	–
−/1/1	1	0	–	–	–	0	–
−/1/1	1	1	–	–	–	0	–
−/1/1	1	1	0	0	–	–	1
−/1/1	1	0	0	1	–	–	0
−/1/1	0	1	1	0	–	–	0
−/1/1	0	0	1	1	–	–	0

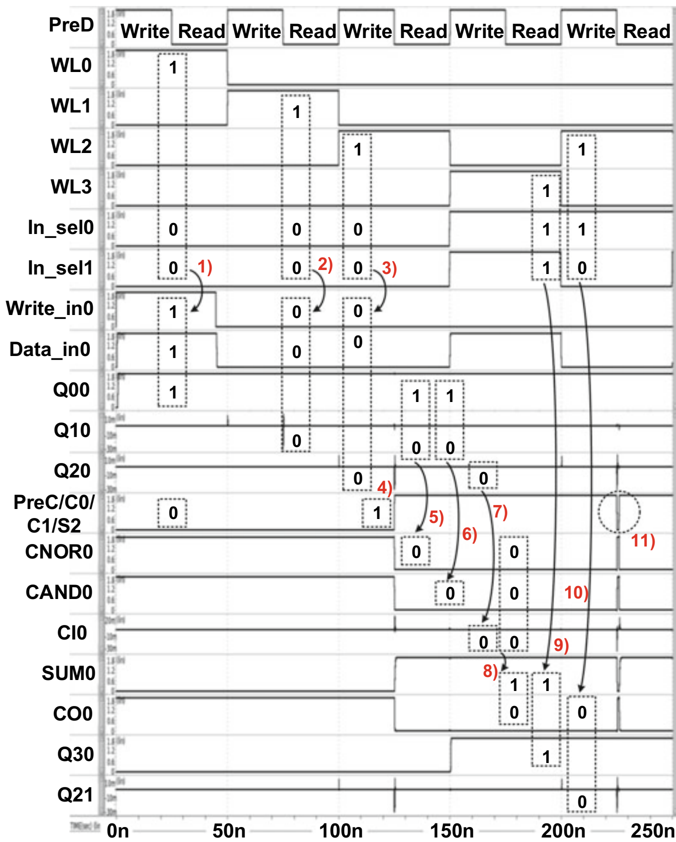

Fig. 4 Detailed logic transitions of an addition

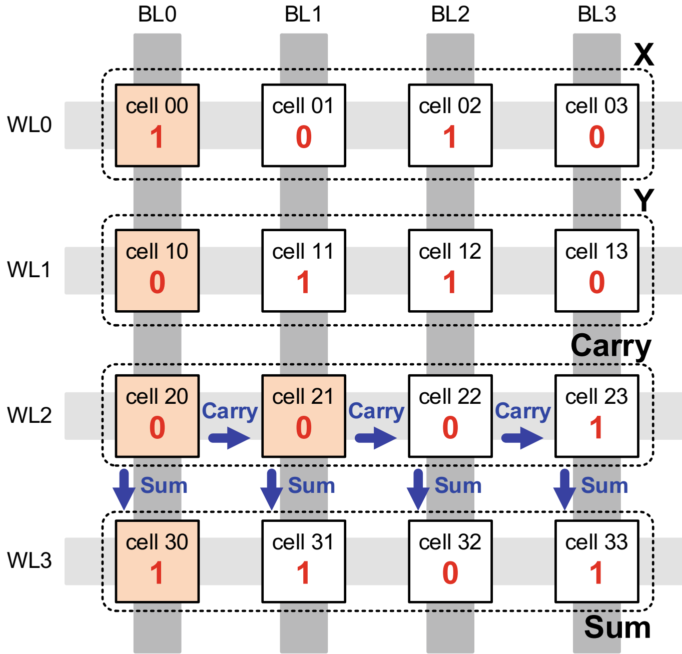

Fig. 5 Demonstration of the operation of the 4-bit ripple carry adder

(1) Enable signal PreC drives the pre-charge circuit to pull CB0, CNOR0, and CAND0 up high (CI0 is low). WL0 is selected to be loaded with data. [0, 0] of In sel[1:0] drives MUX to select Write in 0 as the input data. Q00 is then pulled up high.
(2) Same step as (1). WL1 is then selected to be loaded with data. Q10 is low.
(3) Same step as previous (1) and (2). However, additional carry bit 0 needs to be stored in Q20 to accomplish addition.
(4) PreC pulls high to disable charging, where C0, C1, and S2 are turned on simultaneously to start calculations.
(5) NOR function: Q00 (1), Q10 (0), CNOR0 (0)
(6) AND function: Q00 (1), Q10 (0), CAND0 (0).
7) NOT function: Q20 (0), CB0 (1), CI0 (0).
(8) Addition of bit 0 is complete. SUM0 and CO0 are 1 and 0, respectively.
(9) WL3 is then selected as well as [1] of In sel[1:0] drives MUX to store SUM0 into Q30.
(10) WL2 is finally selected, where [0, 1] of In sel[1:0] enables MUX to reach CO0 as the carry bit to be stored in Q21.
(11) PreC pulls CB1, CNOR1, and CAND1 up high in a short period of time to prepare for the calculation of the next bit.
(12) Repeat steps, (4) to (11) until the calculation is complete.

3 Simulation and Verification

The proposed work is carried out and simulated using UMC 0.18 μm CMOS process. Figure 6 shows the all-PVT-corner simulation (5 Process corners, 3 Voltage variation levels, 3 Temperature) of this 4-bit ripple carry adder. The final result shows that this IMC architecture successfully completes the addition for IMC demand. Comparison with prior IMC SRAMs is tabulated in Table 3. Although we use a legacy CMOS technology in our design, we still attain the least normalized energy on both write and read operations. Most important of all, we are the only ones to realize the addition in a single-ended 6T SRAM.

4 Conclusion

This work presents an IMC ripple carry adder architecture using FS-GDI in a novel single-ended disturb-free 6T SRAM. Not only accumulation problems in original 5T SRAM are resolved, but a simple strategy using FS-GDI to realize the RCA function is proved inside a memory unit.

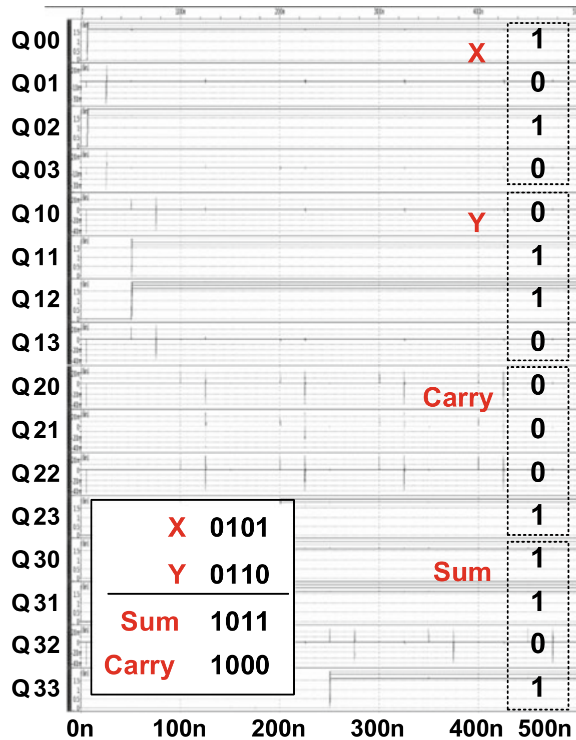

Fig. 6 All-PVT-corner simulation results

Table 3 In-memory computation SRAM performance comparison

	[11]		[12]			*This work*
Year	2018		2018			2019
Process	Fujitsu 55 nm DDC		N/A			UMC 180 nm
Cell type	4 + 2T		8T	8T	8 + T	6T (single-ended)
Operation	AND NOR XOR		NAND NOR XOR RCS	IMP XOR RCS	NAND NOR XOR RCS	NAND NOR XOR SUM
Array size	128 × 128 (16 kB)		N/A			4 × 4
Normalized write energy	28.91 (0.8 V)	88 (0.25 V)	N/A			14.87 (1.98 V, SF, 25 °C) (worst case)
Normalized read energy	25.94 (0.8 V)	78.4 (0.25 V)	N/A			6.99 (1.8 V, TT, 25 °C) (worst case)

Normalized write/read energy = $\frac{\text{fJ/bit}}{(\text{supply voltage})^2}$

Acknowledgements The investigation was partially supported by Ministry of Science and Technology (MOST), Taiwan, under grant MOST 107-2218-E-110-004- and MOST-107-2218-E-110-016-. The authors would like to express their deepest gratefulness to CIC (Chip Implementation Center) in NARL (Nation Applied Research Laboratories), Taiwan, for the assistance of thoughtful chip fabrication.

References

1. Backus J (1978) Can programming be liberated from the von Neumann style? A functional style and its algebra of programs. Commun ACM 21:613–641. https://doi.org/10.1145/359576.359579
2. Wang Y, Yu H, Ni L, Huang G, Yan M, Weng C, Yang W, Zhao J (2015) An energy-efficient nonvolatile in-memory computing architecture for extreme learning machine by domain-wall nanowire devices. IEEE Trans Nanotechnol 14(6):998–1012. https://doi.org/10.1109/TNANO.2015.2447531
3. Jain S, Ranjan A, Roy K, Raghunathan A (2018) Computing in memory with spin-transfer torque magnetic RAM. IEEE Trans Very Large Scale Integr VLSI Syst 26(3):470–483. https://doi.org/10.1109/tvlsi.2017.2776954
4. Jeloka S, Akesh NB, Sylvester D, Blaauw D (2016) A 28 nm configurable memory (TCAM/BCAM/SRAM) using push-rule 6T bit cell enabling logic-in-memory. IEEE J Solid-State Circ 51(4):1009–1021. https://doi.org/10.1109/JSSC.2016.2515510
5. Fan D, Angizi S (2017) Energy efficient in-memory binary deep neural network accelerator with dual-mode SOT-MRAM. In: IEEE international conference on computer design (ICCD). IEEE Press, Boston, pp 609–612. https://doi.org/10.1109/iccd.2017.107
6. Wang C-C, Tseng Y-L, Leo H-Y, Hu R (2004) A 4-Kb 500-MHz 4-T CMOS SRAM using low-V_{THN} bitline drivers and high-V_{THP} latches. IEEE Trans Very Large Scale Integr VLSI Syst 12(9):901–909. https://doi.org/10.1109/tvlsi.2004.833669

7. Wang C-C, Lee C-L, Lin W-J (2007) A 4-Kb low power SRAM design with negative word-line scheme. IEEE Trans Circ Syst I Regul Pap 54(5):1069–1076. https://doi.org/10.1109/tcsi.2006.888767
8. Wang C-C, Hsieh C-L (2016) Disturb-free 5T loadless SRAM cell design with multi-vth transistors using 28 nm CMOS process. In: IEEE international SoC design conference (ISOCC). IEEE Press, Jeju, pp 103–104. https://doi.org/10.1109/isocc.2016.7799754
9. Morgenshtein A, Fish A, Wagner IA (2002) Gate-diffusion input (GDI): a power-efficient method for digital combinatorial circuits. IEEE Trans Very Large Scale Integr VLSI Syst 10 (5):566–581. https://doi.org/10.1109/tvlsi.2002.801578
10. Ahmed MA, Abdelghany MA (2018) Low power 4-bit arithmetic logic unit using full-swing GDI technique. In: International conference on innovative trends in computer engineering (ITCE). IEEE Press, Aswan, pp 193–196. https://doi.org/10.1109/itce.2018.8316623
11. Dong Q, Jeloka S, Saligane M, Kim Y, Kawaminami M, Harada A, Miyoshi S, Yasuda M, Blaauw D, Sylvester D (2018) A 4 + 2T SRAM for searching and in-memory computing with 0.3-V V_{DDmin}. IEEE J Solid-State Circ 53(4):1006–1015. https://doi.org/10.1109/jssc.2017.2776309
12. Agrawal A, Jaiswal A, Lee C, Roy K (2018) X-SRAM: enabling in-memory boolean computations in CMOS static random access memories. IEEE Trans Circ Syst I Regul Pap 65 (12):1–14. https://doi.org/10.1109/tcsi.2018.2848999