# A 40-nm CMOS Multifunctional Computing-in-Memory (CIM) Using Single-Ended Disturb-Free 7T 1-Kb SRAM

Chua-Chin Wang, *Senior Member, IEEE*, Lean Karlo S. Tolentino, *Student Member, IEEE*, Chia-Yi Huang, and Chia-Hung Yeh, *Senior Member, IEEE*

***Abstract*— This investigation proposes a computing-in-memory (CIM) design to circumvent the von Neumann bottleneck which causes limited computation throughput for effective artificial intelligence (AI) applications. The proposed CIM performs multiple operations such as single-instruction basic Boolean operations, addition, and signed number multiplication, and multiple functions such as normal mode and retention mode for the built-in self-test (BIST). Its 2T-Switch requires only two transistors to be utilized for static random-access memory (SRAM) array; thus, the arithmetic unit can be chosen easily and the area overhead is minimized. Its ripple carry adder and multiplier (RCAM) unit based on single-ended disturb-free 7T 1-Kb SRAM was developed using the full swing-gate diffusion input (FS-GDI) technology that has full voltage swing resolution, low power consumption, and less chip area cost. Its Auto-Switching Write Back Circuit restores addition and multiplication operations automatically to assigned memory address. The CIM is implemented using the TSMC 40-nm CMOS process, where the core area is 432.81 × 510.265 $\mu$m$^2$. Among the related works, the proposed CIM performs the most number of operations and functions.**

***Index Terms*— Computing-in-memory (CIM), disturb-free, full swing-gate diffusion input (FS-GDI), static random-access memory (SRAM), von Neumann bottleneck.**

## I. Introduction

ARTIFICIAL intelligence (AI) and neural networks have contributed very much to the development of Industry 4.0. These applications utilized von Neumann architecture that consists of memory as storage device and arithmetic logic unit (ALU) as calculation device. However, von Neumann bottleneck [1] is still a serious problem wherein a large amount of data flow between the memory and the ALU, leading to timing overhead, throughput, and energy efficiency limitations. Several studies have been conducted to resolve these limitations by implementing computing-in-memory (CIM) architectures [2]–[4], which mainly implements calculations directly in the memory array. With this, it does not need to execute data transfer from memory arrays to processors [2]–[4]. Apparently, it performs operations by reading and writing them back to the memory. To achieve a structure that can perform CIM, a small additional circuit for operations must be allotted. This circuit must be able to switch the source and the destination of the calculation flexibly.

Static random-access memories (SRAMs) are commonly used as CIM devices over dynamic random-access memories (DRAMs) [3], [4], because they perform bit-wise logical operations and read out data faster and are more highly reliable which are needed for AI applications. However, they have higher power and area consumption. For this reason, a 4T-load less SRAM was developed [5], but there are disturbances in bitline during the reading and writing of data which causes static noise margin (SNM) to get worse [6]. As a solution, a disturb-free property for SRAMs was recommended where it was implemented using a write assist-loop based on high-$V_{\text{th}}$ transistors [7]. Meanwhile, a single-ended disturb-free 6T SRAM was proposed for a multifunctional CIM performing addition and logical operations [8]. However, when applied in AI and convolution operations, the positive and negative values for 0 and 1 are the needed values for the calculation process. Moreover, the convolution operation requires simultaneous addition and multiplication operations.

In this work, we propose a novel CIM architecture based on disturb-free 7T 1-Kb SRAM as shown in Figs. 1 and 2, which performs single-instruction Boolean operations, four-bit addition, and four-bit signed number multiplication and multiple functions such as normal mode and retention mode for the built-in self-test (BIST). The CIM architecture aims to replace the traditional SRAM in the system edge for the ALU to have less calculation work. Its ripple carry adder and multiplier (RCAM) unit was constructed using full swing-gate diffusion input (FS-GDI)

Manuscript received June 12, 2021; revised August 29, 2021; accepted September 16, 2021. Date of publication October 13, 2021; date of current version November 30, 2021. This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2218-E-110-002 and Grant MOST 109-2218-E-110-007. *(Corresponding author: Chua-Chin Wang.)*

Chua-Chin Wang is with the Department of Electrical Engineering and the Institute of Undersea Technology (IUT), National Sun Yat-sen University (NSYSU), Kaohsiung 80424, Taiwan (e-mail: ccwang@ee.nsysu.edu.tw).

Lean Karlo S. Tolentino and Chia-Yi Huang are with the Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan (e-mail: leankarlo.tolentino@g-mail.nsysu.edu.tw; mark584967@vlsi.ee.nsysu.edu.tw).

Chia-Hung Yeh is with the Department of Electrical Engineering, National Taiwan Normal University, Taipei 10610, Taiwan, and also with the Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan (e-mail: yeh@mail.ee.nsysu.edu.tw).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TVLSI.2021.3115970.

Digital Object Identifier 10.1109/TVLSI.2021.3115970

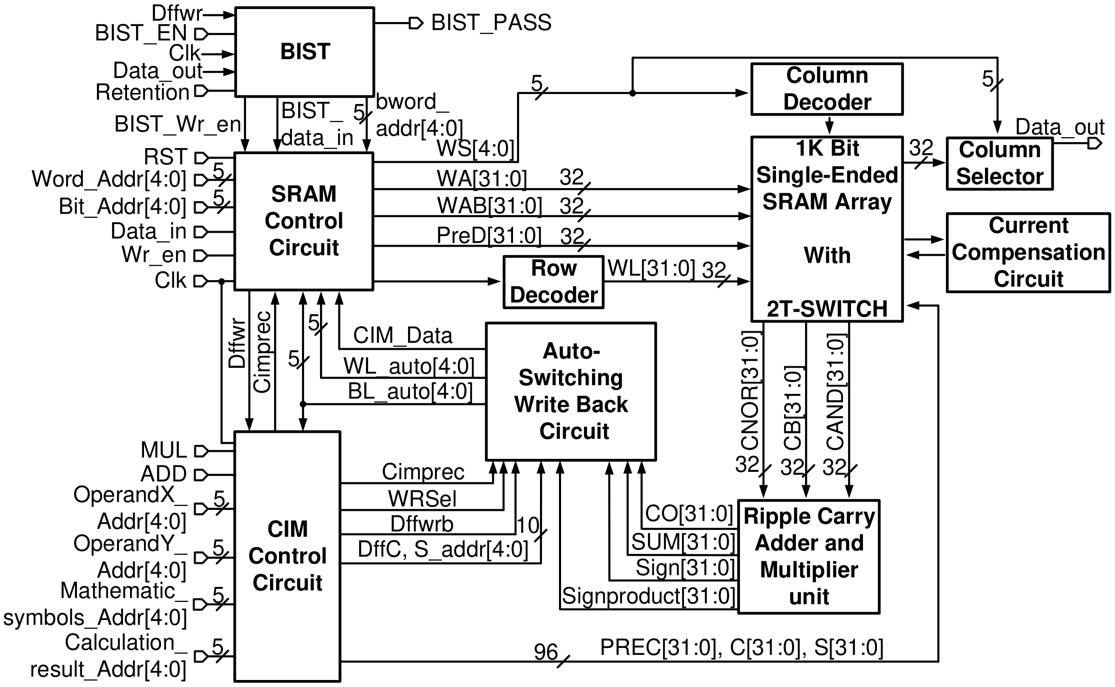

Fig. 1. Block diagram of the 1-Kb CIM architecture.

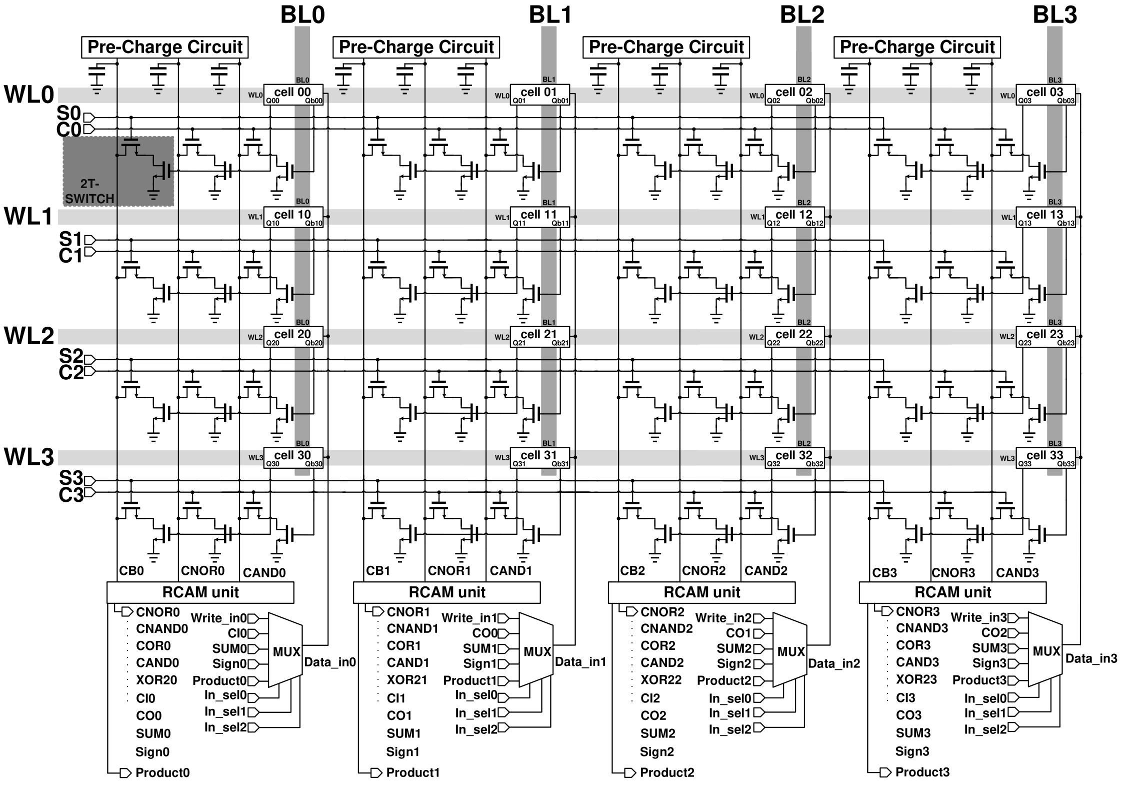

Fig. 2. CIM schematic (4 × 4 example).

technology [9], [10] which has the advantages of good voltage swing resolution, low power consumption, and small chip area consumption.

The organization of this study is as follows. Section I describes the background and motivation of this research. Section II presents the related prior works on SRAM-based CIM. Section III explains the theory and features of the proposed CIM. Section IV discusses the CIM's implementation and measurement results. Finally, Section V concludes this study.

## II. Related Works

SRAM-based CIMs that performed different functions and operations were proposed and developed for several applications. A 64-Kb, 28-nm SRAM operating at 18.2 MHz was developed for image recognition, which utilized one-write and two-read ports for active energy saving [11]. Operating at a much higher frequency of 28.6 MHz, it was further improved by implementing a selective sourceline drive concept modifying the SRAM footer for cutting off the read bitline discharge [12]. However, these two SRAMs used 8T scheme which are power and area hungry.

A 4-Mb, 40-nm SRAM was proposed for face detection and recognition which is realized using the 5T scheme [13]. Support vector machine, a popular AI algorithm due to its good performance as classifier [14]–[16], was used for recognizing the face. Although the said SRAM used 5T to save more power and area, it mostly performed read operations since its function is to detect and recognize face image. On the other hand, a 6T-SRAM-based, 65-nm deep neural network (DNN) chip was realized that performs best at 400-MHz clock and 0.6 W. Though it has low CIM energy efficiency, it has higher area efficiency [17]. Meanwhile, another 40-nm, 3-D SRAM-based, DNN CIM was proposed that achieved 7.49 TOPS in binary precision but 1.96 TOPS only for four-bit precision at 300 MHz [18]. Finally, a different 28-nm, 8T-SRAM-based CIM was developed that can execute unsigned and ternary multiplication and logical operations [19].

Unlike the prior works, our proposed CIM has normal and retention BIST modes. Normal mode determines whether the memory's read and write functions are functioning. Retention mode writes data 0 to the memory, reads for a long time, and tests whether the stored data will be distorted for a long time of reading. It ensures and notifies if no stored data will be destroyed.

## III. CIM System Architecture Based on Single-Ended Distrub-Free 7T SRAM

The block diagram of the 1-Kb CIM system architecture is displayed in Fig. 1. It is composed of several featured blocks which will be discussed in Sections III-A–III-G.

The following input and output signals of this CIM are introduced.

1) *Wr_en:* Read and write signals for reading and writing the memory.
2) *Word_Addr[4:0]:* Select the five-bit character line of the memory data storage address.
3) *Bit_Addr[4:0]:* Select the five-bit bitline of the memory data storage address.
4) *OperandX_Addr[4:0]:* Five-bit operand address line for input $X$.
5) *OperandY_Addr[4:0]:* Five-bit operand address line for input $Y$.
6) *Mathematic_Symbols_Addr[4:0]:* Five-bit arithmetic symbol number address line.
7) *Calculation_Result_Addr[4:0]:* Five-bit calculation result digital address line.
8) *ADD:* Addition operation start signal.

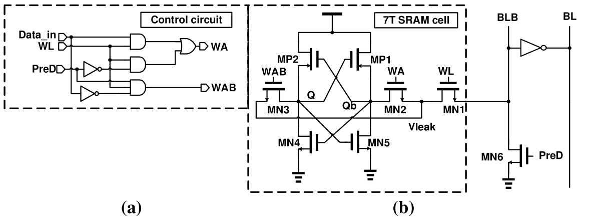

Fig. 3. (a) Control circuit of 7T SRAM. (b) 7T SRAM cell.

9) *MUL:* Multiplication operation start signal.
10) *Data_in:* Data input line.
11) *Clk:* Clock signal.
12) *RST:* Operation reset signal.
13) *BIST_EN:* Self-test control line.
14) *Retention:* Retention time mode activation signal.
15) *BIST_PASS:* Built-in self-test (BIST) results.
16) *Data_Out:* Data output line.

Meanwhile, the schematic of 4 × 4 CIM is shown in Fig. 2. The system uses 2T Switch that connects the data output points $Qxx$ and $Qbxx$ ($xx = 00, 01, 02, \ldots, 33$) of each SRAM cell (see Fig. 3) to detect the values stored in any two rows of SRAM on the address line (BL) and calculates the results of these values CB$x$, CAND$x$, and CNOR$x$ ($x = 0, 1, 2, 3$). Then, it uses the RCAM unit to perform addition and multiplication. Next, it uses a multiplexer (MUX) to write the calculation result back to the corresponding SRAM to achieve the addition and multiplication operations in the memory. Utilizing the above-mentioned in-memory operation architecture, the operation of the memory only requires general read and write operations, and the data switching between Precharge Circuit and MUX can be used to complete in-memory operations. If the data is already saved in the SRAM cell, only four clock cycles are used to do the calculation for one bit.

### A. 7T SRAM and Associated Control Circuit

As shown in Fig. 3, this 7T SRAM in Fig. 1 uses word line (WL), memory cell control lines (WA and WAB), and pre-discharge lines (PreD) to read and write data. WL can only select one row at a time. In addition, MN4 and MN5 provide a discharge path to keep the state of $Q$ and $Qb$ stably.

When writing 0, PreD is kept at high condition to pull the reverse bitline (BLB) to ground, and at the same time, the cell is selected to be written. When WL and WAB are high, WA is turned off and the value of $Q$ is pulled to ground. On the contrary, when writing 1, WA is turned on and $Qb$ is pulled to ground. In this way, VDD will pull $Q$ to high through MP2. When reading, WA and WL are turned on to transmit $Qb$ to BLB and generate the state of BL through an inverter. The SRAM control circuit controls WA and WAB based on the inputs, namely Data_in, WL, and PreD, where the function table is shown in Table I.

### B. 2T Switch

Fig. 4 shows the 2T Switch in Fig. 1. It increases the energy efficiency, thanks to its simplicity. Referring to an example

TABLE I
SRAM CONTROL CIRCUIT FUNCTION TABLE

| | $Data\_in_x$ | PreD | $WL_x$ | WA | WAB |
|---|---|---|---|---|---|
| Standby | X | X | 0 | 0 | 0 |
| Read | X | 0 | 1 | 1 | 0 |
| Write 0 | 0 | 1 | 1 | 0 | 1 |
| Write 1 | 1 | 1 | 1 | 1 | 0 |

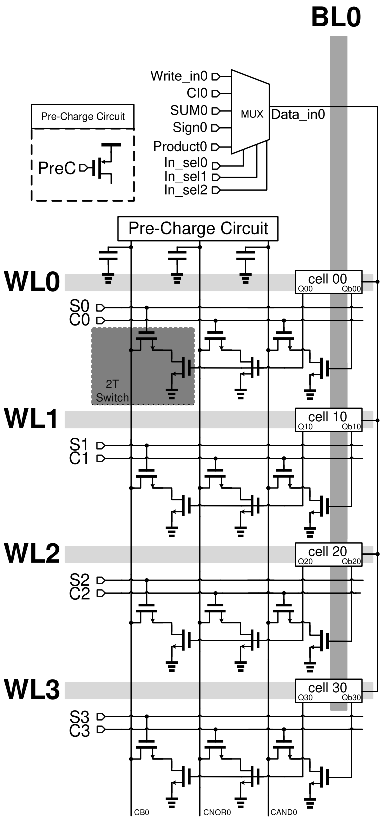

Fig. 4. 2T Switch and its control circuit.

in Fig. 5, the waveform in Fig. 6, Table II, and the RCAM unit circuit in Fig. 7, its function can be described step-by-step as follows: initialization, startup, computation phase I, and computation phase II.

During initialization (referring to phase 1 in Fig. 6), the PreC signal is low in each write cycle. After CB$x$, CNOR$x$, and CAND$x$ are precharged to high, PreC is pulled high to operate at startup mode (referring to phase 2 in Fig. 6). Notably, referring to Fig. 7 and Table III, CB is the inverting carry bit and CI is the carry bit. Two sets of control signals $Sx$ and $Cx$ control 2T Switch to couple $Qxx$ or $Qbxx$ to CB$x$, CNOR$x$, and CAND$x$ (where $x = 0, 1, 2, 3$).

During computation phase I (referring to phase 3 in Fig. 6 and Table II), only one of the signals ($S0$–$S3$) can be logic high at the same time to select the SRAM cell where the carry bit is placed for addition. When $Qxx$ is logic 1, CB$x$ is logic 0 as shown in the left side of Fig. 5.

At computation phase II (referring to Fig. 8 where addition and multiplication is performed by the RCAM unit), when one of $Qxx$ signals is logic 1, CNOR$x$ is at logic 0 through the

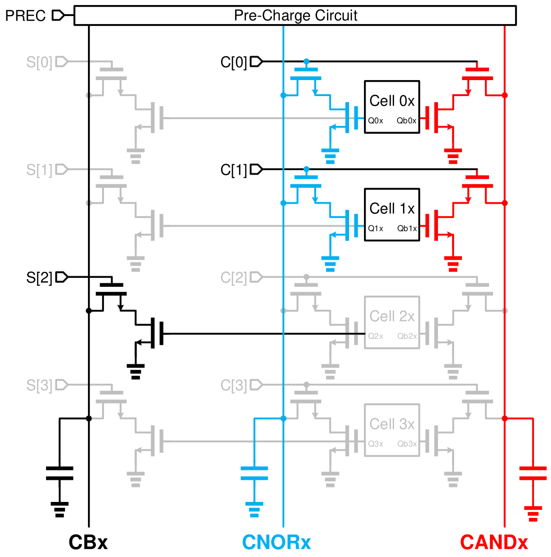

Fig. 5. 2T Switch operations (an example).

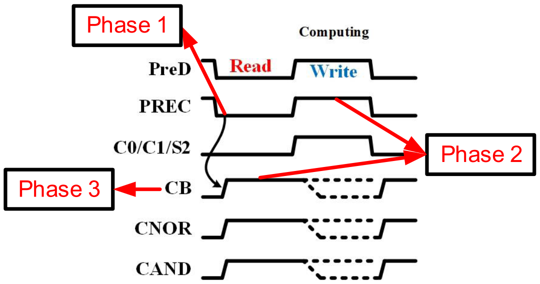

Fig. 6. Waveform showing the first three computation phases.

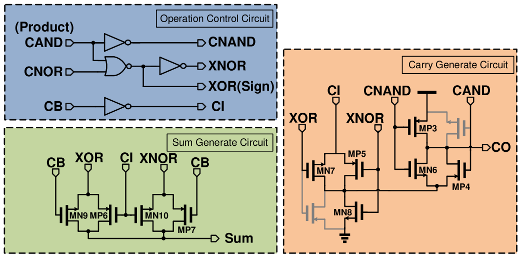

Fig. 7. RCAM unit.

2T Switch. $C0$ to $C3$ turn on any two signals at the same time for NOR operation. To perform the AND operation of $Qbxx$, two $Cx$ ($C0$, $C1$) signals are turned on. When the two $Qbxx$ signals are both logic 0, the CAND$x$ signal remains logic 1. The layout of the 7T SRAM together with the three 2T Switches is shown in Fig. 9.

## *C. Ripple Carry Adder and Multiplier (RCAM) Unit*

The RCAM unit in Fig. 1 is shown in Fig. 7. It is composed of combinational circuits and reduced FS-GDI. As shown in Fig. 7, the right side is a degenerated FS-GDI logic where all the transistors, as seen from the left side of the said figure, whose source connected to either VDD or GND are removed. With this, the number of transistors and the power

TABLE II
2T SWITCH AND RCAM FUNCTION TABLE

| S2/C0/C1 | Q0x | Q1x | Qb0x | Qb1x | CBx | CNORx | CANDx |
|---|---|---|---|---|---|---|---|
| 1/-/- | 0 | - | - | - | 1 | - | - |
| 1/-/- | 1 | - | - | - | 0 | - | - |
| -/1/1 | 0 | 0 | 1 | 1 | - | 1 | 0 |
| -/1/1 | 0 | 1 | 1 | 0 | - | 0 | 0 |
| -/1/1 | 1 | 0 | 0 | 1 | - | 0 | 0 |
| -/1/1 | 1 | 1 | 0 | 0 | - | 0 | 1 |

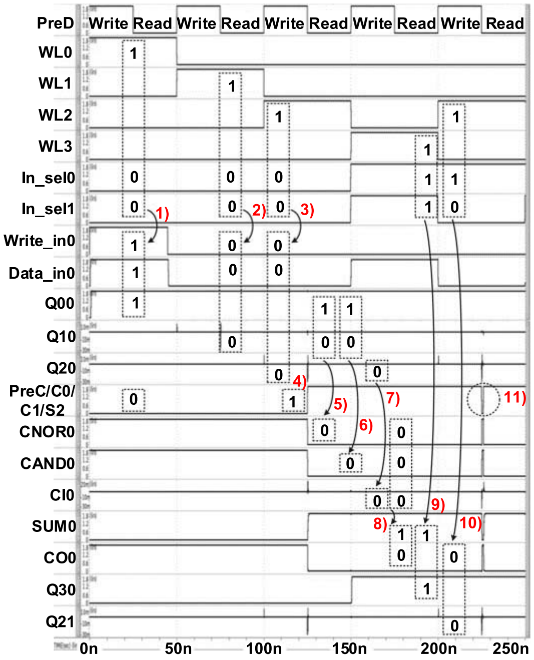

Fig. 8. Computation phase II.

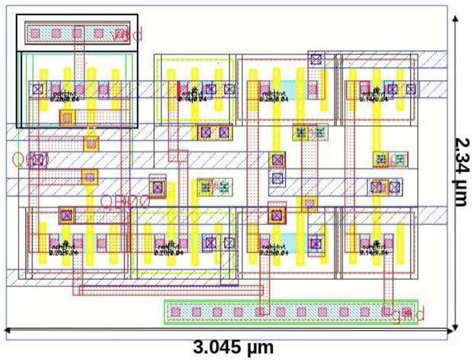

Fig. 9. Layout of the 7T SRAM and 2T Switch.

consumption is reduced and the computational capability is increased. Table III shows the logical functions implemented by the RCAM unit.

To further explain the addition operation defined by the computation phase II which is implemented by the 2T Switch and RCAM unit, Figs. 8 and 10 shows the logic transitions and data flow when four-bit addition is carried out, respectively. Moreover, important to note is that the stored data and the cell blocks' data transition (cell 00, cell 10, cell 20, cell 30, and cell 21) is highlighted in red and orange, respectively.

TABLE III
LOGICAL FUNCTIONS IN AN RCAM UNIT

| | |
|---|---|
| $CB = \overline{CIx}$ | $XOR = A \oplus B$ |
| $CAND = AB$ | $Sum = (A \oplus B) \oplus CIx$ |
| $CNAND = \overline{AB}$ | $CO = (A \oplus B) \cdot CIx + A \cdot B$ |
| $CNOR = \overline{A + B}$ | $Sign = XOR = C \oplus D$ |
| $Product = CAND = EF$ | |

$A, B$ = input addition bit; $CIx$ = carry in; $CO$= carry out; $C, D$ = input sign bit; $E, F$ = input multiplication bit

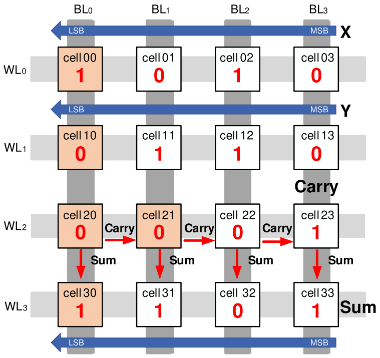

Fig. 10. RCAM's data flow for four-bit addition.

Referring to numbered labels in Fig. 8 as follows.

1) PreC instructs the precharge circuit to set CB0, CNOR0, and CAND0 to high, while CI0 is set to low. Data is loaded in WL0. In_sel[1:0] causes MUX to write 0 as its input. $Q00$ is raised to high.
2) Follow the same procedure as Step 1. Data is loaded in WL1. $Q10$ is set to low.
3) Follow the same procedure as Steps 1 and 2. To accomplish addition, however, an additional carry bit 0 must be stored in $Q20$.
4) For charging to be disabled, PreC is pulled high, and $C0$, $C1$, and $S2$ are all switched on at the same time to begin calculations.
5) NOR is carried out where $Q00$ is high, $Q10$ is high, and CNOR0 is low.
6) AND is carried out where $Q00$ is high, $Q10$ is low, and CAND0 is low.
7) NOT is carried out where $Q20$ is low, CB0 is high, and CI0 is low.
8) Bit 0 addition is finished. SUM0 is high and CO0 his low.
9) After that, MUX selects WL3, and [1] of In_sel[1:0] causes MUX for SUM0 to be stored in $Q30$.
10) Then, MUX selects WL2. In_sel[1:0]'s [0, 1] allows MUX to designate the carry bit in CO0 and store it in $Q21$.
11) PreC rapidly raises CB1, CNOR1, and CAND1 to high levels in order to prepare for the next bit's calculation.

Procedure as Steps 4–11 is repeated. It will be completed when the calculation is finished.

The multiplication process is similar but the difference in Step 3 where the sign-product bit is stored in cell $Q20$ and the product bit is stored in $Q30$. For Step 9, WL2 is instead

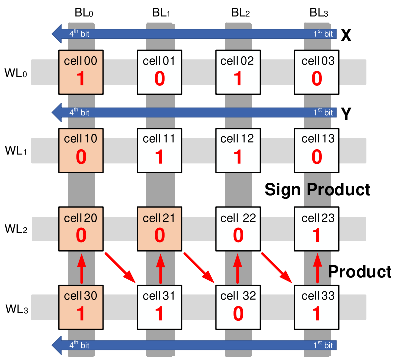

Fig. 11. RCAM's data flow for multiplication.

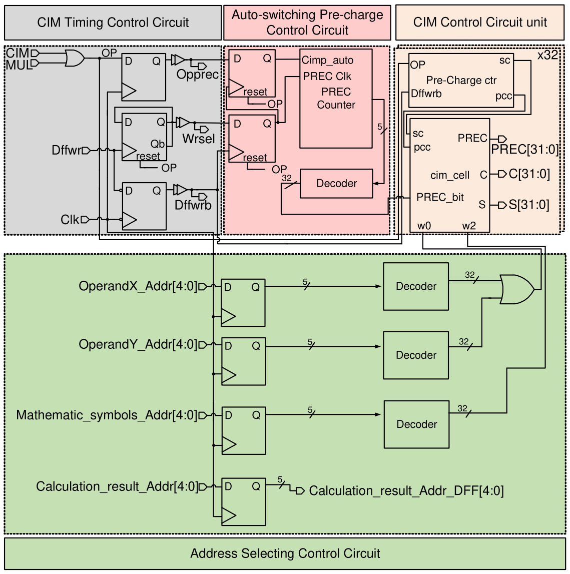

Fig. 12. CIM control circuit.

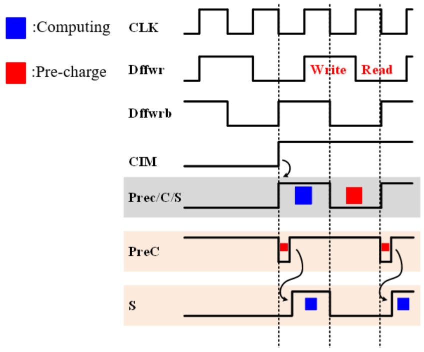

Fig. 13. Auto-switching write back operation.

selected and the product bit is stored into $Q30$. For Step 10, WL3 is instead selected and the sign-product bit is stored into $Q20$. This is shown in Fig. 11.

### D. CIM Control Circuit

A CIM control circuit in Fig. 1 is shown in Fig. 12. It controls the precharge of the specified row address, selects the data stored in the specified row unit, and controls the 2T Switch for calculation. It consists of CIM control circuit unit, CIM timing control circuit, address selecting control circuit, and auto-switching precharge control circuit.

The CIM timing control circuit determines whether to perform calculations and generate timing control signals. When one of the CIM or MUL signal is pulled high, OP is also pulled high and the CIM control circuit starts to generate the corresponding calculation control signals to initiate the needed calculations through its output signal Opprec. If the CIM signal is pulled high, the memory will perform addition on the specified operation address from $\mathrm{BL}_i$ $(i = 0, 1, 2, 3)$ in Fig. 2 one by one until CIM signal is turned off. On the other hand, when the MUL signal is pulled high, the multiplication is performed by the memory on the specified operation address from $\mathrm{BL}_i$ $(i = 0, 1, 2, 3)$ until the MUL signal turns off. At the same time, the CIM timing control circuit divides the read latch clock control signal (Dffwr) and triggers the inverted latch with Clk. It outputs the clock control signals (Wrsel, Dffwrb) to generate the two different operation clocks required by the operation.

The auto-switching precharge control circuit generates the specified row addresses in the memory for sequential precharging, with the precharging switching clock signal (PREC Clk) triggered by Wrsel through the D-type flip-flop. The precharge counter (PREC Counter) increments once every clock cycle. The 5-to-32 Decoder generates the precharge timing control signals PREC_bit[31:0].

The address selecting control circuit specifies the arithmetic address in the memory such as OperandX_Addr[4:0], OperandY_Addr[4:0], Mathematic_symbols_Addr[4:0], and Calculation_result_Addr[4:0]. These addresses are read through D-type flip-flop, three 5-to-32 multiplexers, and OR gate. The address of the arithmetic memory is the output to the CIM control circuit unit.

The CIM control circuit uses the precharge control circuit unit (Precharge ctr) to alternately generate the precharge unit control signals (sc, pcc), so that the operation is performed after the precharge is completed. The CIM calculation unit control circuit (cim_cell) outputs the precharge control signal (PREC) and the calculation unit control lines ($C$, $S$), and controls the designated address to control the 2T Switch.

Referring to Figs. 12 and 13, when the CIM signal is pulled high, the PreC signal is precharged, and the $S$ signal selected by the Carry_addr address starts to perform calculations. Dffwr is the waveform after wr_en in Fig. 1 is sampled by Clk and represents the read and write operations. Write is performed at high potential, and read at low potential.

### E. Auto-Switching Write Back Circuit

Fig. 14 shows the auto-switching write back circuit that includes the bitline automatic switching circuit (BL auto-switching circuit), data switching circuit, and WL automatic switching circuit (WL auto-switching circuit). As shown in Fig. 14, when OP is pulled to a high potential, the auto-switching write back circuit in Fig. 1 writes back the sum or product to an assigned address starting from LSB (BL0) to MSB (BL3).

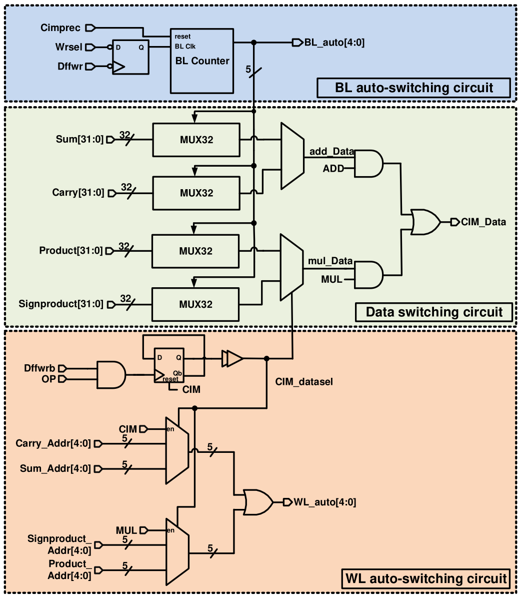

Fig. 14. Auto-switching write back circuit.

At the BL auto-switching circuit, when the pre-calculation signal (Cimprec) is at a high level, the inverted Wrsel is used to switch the clock signal (BL Clk) triggered by the D-type flip-flop as a clock cycle and the bitline counter (BL Counter). Each clock cycle will increment once, and the output BL_auto[4:0] will replace the original memory bitline address to achieve the automatic memory bitline switching function.

The data switching circuit supervises the data selection. It has a single-ended output, so it is necessary to generate two addition results or two multiplication results, interchangeably. This is done by four 32-to-1 multiplexers which is controlled by BL_auto. Through the two 2-to-1 multiplexers, the data selection signal (CIM_datasel) is used to alternately switch the addition unit operation addresses, sum and carry, and multiplication unit operation addresses, product and sign product, and finally ADD and MUL are used for data selection. Its output CIM_Data generates carry and sum for addition, and product and sign product for multiplication, respectively.

At the BL auto-switching circuit, when the operation start signal (OP) is high, the Dffwrb provided by the CIM timing control circuit in Fig. 12 will be latched by the D-type flip-flop to generate CIM_datasel and use this signal to control two 2-to-1 multiplexers. These multiplexers make it alternately switch the add carry address (Carry_Addr) and the sum address (Sum_Addr) with the multiplication product address (Product_Addr) and the sign product address (Signproduct_Addr). The output WL_auto[4:0] replaces the original memory address of the WL to automatically switch the WL.

### F. Built-in Self-Test Circuit

The BIST in Fig. 1 is presented in Fig. 15. When self-test control line (BIST_EN) is high, the BIST circuit is activated. As shown in Fig. 18, it has two modes, namely normal and retention modes, where they send testing data and then read or write signals through the BIST controller (see Fig. 16). Normal mode uses the pattern generator (PG) to generate a five-bit random number address and the BIST controller to generate alternate switching data 0 and 1. It compares the memory with the above data of the output response analyzer (ORA) at the same time and finally determine whether the memory's read and write function is correct. If the output result matches the data generated by the BIST controller, the self-test result (BIST_PASS) is high; otherwise, the memory's read function is not functioning. Retention mode uses the BIST controller to latch the self-test data (BIST_Data) and reset the self-test read–write signal (BIST_WR) to a low level (writing data 0 to the memory), so that the memory can be read for a long time, and to test whether the stored data will be either destroyed or distorted for a long time of reading. If no stored data is changed, BIST_PASS is pulled to high.

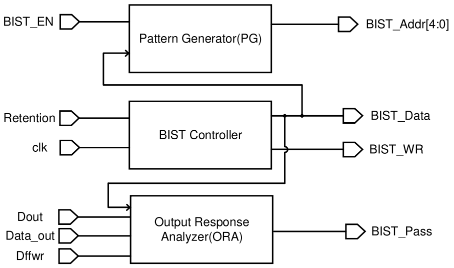

Fig. 15. BIST circuit.

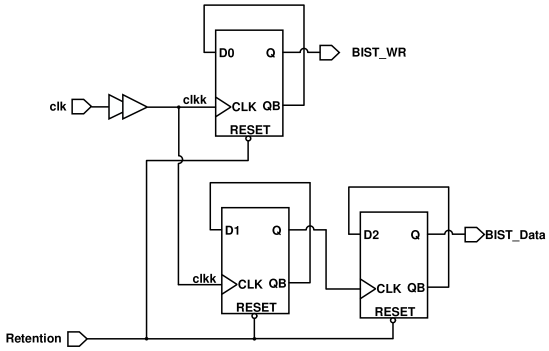

Fig. 16. BIST controller circuit.

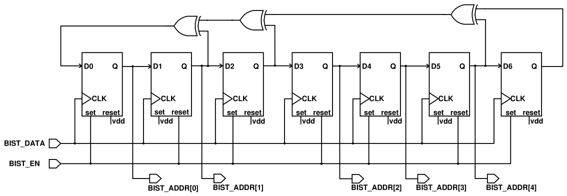

Fig. 17. PG circuit.

As shown in Fig. 17, PG generates specific and sequential self-test addresses. It sends to the 1-Kb SRAM array for reading and writing tests. When signal retention is high, the retention test mode is activated. This mode will latch BIST_Data and set BIST_WR to low to make the memory read for a long time. The rest of the actions can be referred to the Timing Diagram of Self-Test mode in Fig. 18.

Finally, the data (BIST_Data) and read/write signal (BIST_WR) are sent to the ORA circuit as shown in Fig. 19

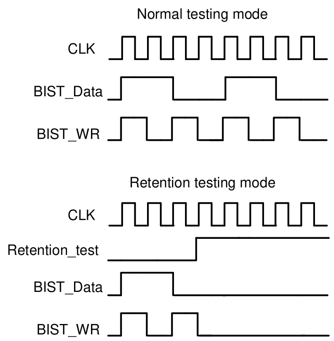

Fig. 18. BIST circuit timing diagram.

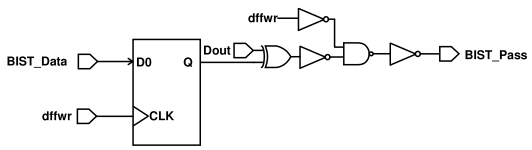

Fig. 19. Output response analysis (ORA) circuit.

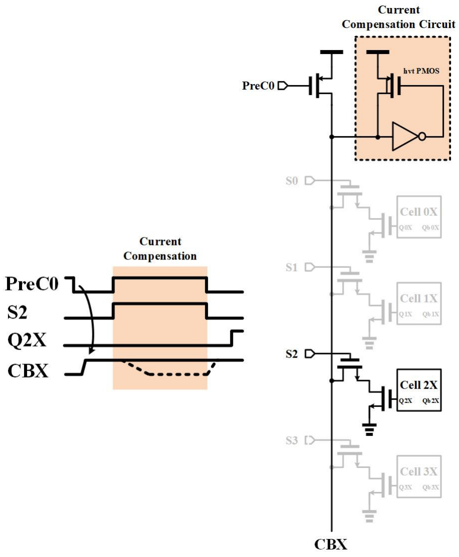

Fig. 20. 2T-Switch current compensation circuit.

and the data after read/write (Data_out_b) are compared. When the read/write mode is low (BIST_WR = 0), the data state is read in the memory cell to compare the BIST_Data and Data_out. If these two signals are the same, the resulting BIST_Pass at high state is generated.

### *G. 2T Switch Current Compensation Circuit*

As mentioned earlier, calculations are started when signals $S2$ and PreC are pulled high. However, since 2T Switch circuit is bothered by charge-sharing problems and CB$x$ needs to be maintained at high (i.e., $Q2x$ is at high), the loss of charge still causes the voltage at node CB$x$ to drop. Therefore, the Current Compensation Circuit in the dashed box in Fig. 20 needs to be added.

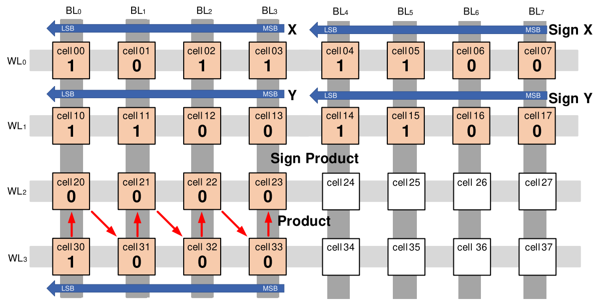

Fig. 21. Demonstrating multiplication in memory.

### *H. Analysis of Operations in Memory*

In this section, the operations executed in the memory is analyzed. For example, Fig. 21 shows the demonstration of multiplication operation implemented in the memory. Taking a 4 × 4 operation, the operation data is sequentially from the least significant bit (LSB) or first bit to the most significant bit (MSB) or the fourth bit and reads into four columns of words through the memory wordline addresses WL0 to WL3 and eight rows of bitline addresses BL0 to BL7. Operand $X$ (1101) is stored in WL0's WL address and BL0 to BL3's bitline address, while the sign of the operand, Sign $X$ (0011), is stored in the WL address of WL0 and the bitline address of BL4 to BL7. For the operand $Y$ (0011) and the sign of the operand, Sign $Y$ (0011), the WL and the corresponding bitline address are stored in WL1. When the multiplication operation start signal (MUL) is turned on, the memory starts to operate sequentially from the bitline address BL0, first generates the LSB product operation result as data 1, and then outputs the LSB product sign operation result, which is data 0. By analogy, the final output data is classified as (10, 00, 00, 00) by bitline.

The following items below explain the selection of data stored in the memory and the switching of the signals in the memory for calculation and analysis.

*1) Selection of Data Stored in Memory:* The data flow control signal in the memory of this architecture includes the automatic switching bitline signal BL_auto[4:0] and the automatic data switching signal CIM_datasel in the automatic switching write-back circuit in Fig. 14. As shown in Fig. 22, two data filtering processes are performed here in sequence after row selection and column selection, and a 16-bit memory array is taken as an example to illustrate.

Row selection: When the operation is completed, by automatically switching the bitline signal BL_auto[4:0] in two read and write cycles as the unit, the corresponding row from the LSB or first bit to the MSB or fourth bit is selected in data output sequence.

Column selection: When the row selection is completed, the automatic data switching signal CIM_datasel alternately switches the product sign operation result and the product operation result in each read and write cycle. Then, the single-ended output operation unit data (CIM_Data) is ready to be stored in the memory unit.

*2) Operation Control Signal in Memory:* The arithmetic control signals in the memory of this architecture include the precharge signal PREC[31:0] of the CIM control circuit

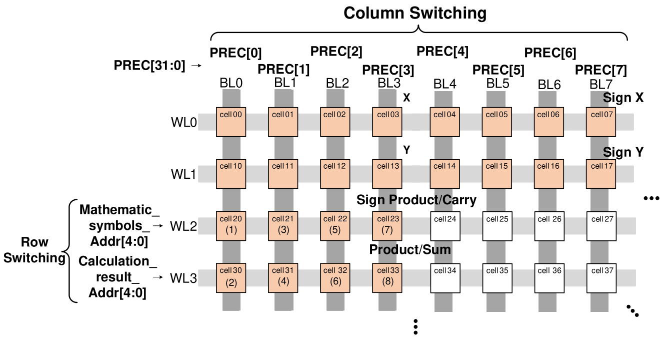

Fig. 22. Switching mode operation in memory.

in Fig. 12, the auto-switching WL signal WL_auto[4:0] and the auto-switching bitline signal BL_auto[4:0] in the automatic switching write-back circuit in Fig. 14. As shown in Fig. 22, two control modes, row switching, and column switching, in a 16-bit memory array are taken as an example.

Row switching: When the operation starts with the precharge signal PREC[31:0], each group of four bits is used as the unit. The architecture has a total of 32 rows divided into eight groups, and each group goes from the LSB to the MSB turning on precharge for BL0 to BL3 in sequence. The automatic switching bitline signal BL_auto[4:0] takes two read and write cycles as the unit, sequentially increases from the LSB to the MSB, and sequentially switches the data write-back address.

Column switching: When the memory data is stored in the corresponding address, the Mathematic_symbols_Addr[4:0] and Calculation_result_Addr[4:0] are alternately switched in units of each read and write cycle. The output automatically switches the WL signal WL_auto[4:0] control. Then, the data is written back to the address, and the odd read/write cycle is switched to Calculation_result_Addr[4:0]. On the contrary, the even read/write cycle is switched to Mathematic_symbols_Addr[4:0].

## IV. Implementation and Measurement

The TSMC 40-nm CMOS process was used to realize the proposed CIM. Fig. 23 shows the (a) layout and (b) die micrograph. The core area is 432.81 × 510.265 $\mu\text{m}^2$. The whole chip size is 840.9 × 867.31 $\mu\text{m}^2$.

### A. Simulation and Analysis

Five hundred sets of Monte Carlo simulations were performed for this architecture. Figs. 24–26 show the Monte Carlo simulation results of data transition, reading data 0, and reading data 1, respectively, at node $Q$ of the SRAM cell as shown in Fig. 3. During data transition, the drift range of the time axis is about 375.08 to 375.19 ns. Meanwhile, when the data 0 is read, voltage drift range is about 2.4–10.1 $\mu$V. Finally, when reading data 1, the voltage drift range is about 899.78–900 mV. The Monte Carlo simulations results prove that the proposed CIM can effectively prevent the resulting circuit characteristics' fluctuations, thereby improving the yield.

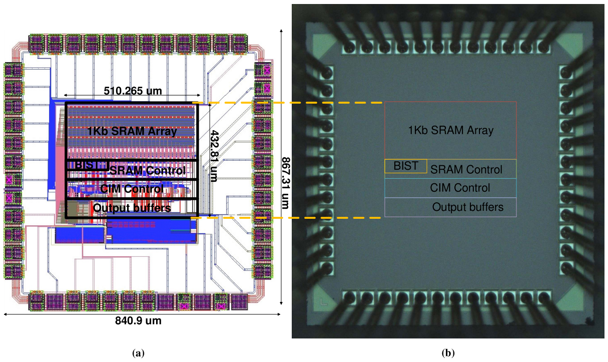

Fig. 23. (a) Layout; (b) die micrograph of the CIM.

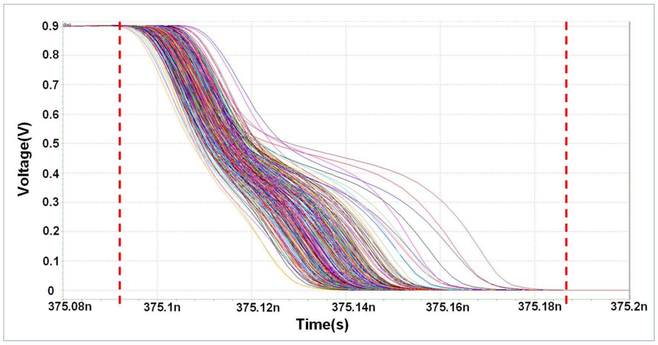

Fig. 24. Monte Carlo simulation results of data transition.

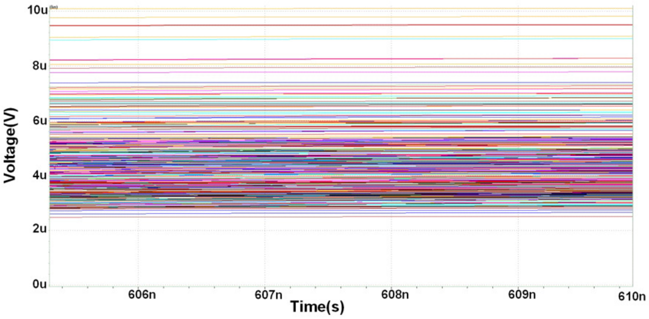

Fig. 25. Monte Carlo simulation results of reading data 0.

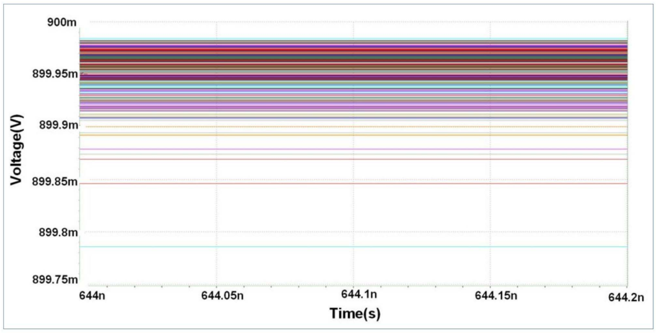

Fig. 26. Monte Carlo simulation results of reading data 1.

Fig. 27 shows the Monte Carlo histogram of reading data 1 (high potential). From the bell-shaped statistical graph, its average value ($\mu$) is 0.9 V, which is in line with the designed

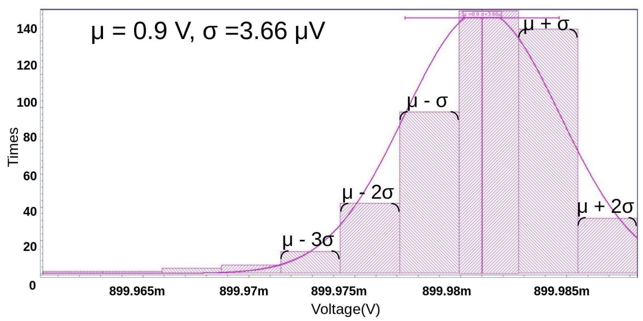

Fig. 27. Monte Carlo histogram of reading data 1.

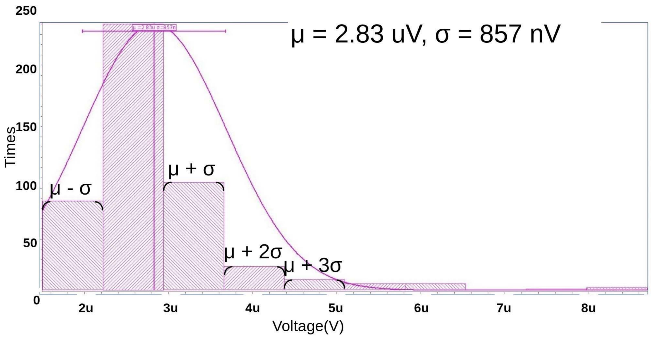

Fig. 28. Monte Carlo histogram of reading data 0.

operating voltage, and its standard deviation ($\sigma$) is 3.66 $\mu$V. Since the upper limit of the operating voltage is 0.9 V, there is no $\mu + 3\sigma$ data in this histogram. Based on Fig. 27, most of the data are within the range of positive two standard deviations and negative three standard deviations of the mean. As shown in the said figure, read data 1 corresponds to the normal distribution and conforms to the three-sigma rule of thumb.

Fig. 28 shows the Monte Carlo histogram of reading data 0 (low potential) versus times, where $\mu$ is 2.83 $\mu$V and $\sigma$ is 857 nV. Similarly, because the lower limit of data 0 is 0 V, there is no data for $\mu - 2\sigma$ and $\mu - 3\sigma$. However, based on the same figure, most of the sub-data are all within the range of positive three standard deviations and negative one standard deviation of the mean. This shows that read data 0 corresponds to the normal distribution and conforms to the three-sigma rule of thumb.

Figs. 29 and 30 show the SNM and dynamic noise margin (DNM) of the proposed CIM's SRAM cell, respectively. Apparently, the SNM plot does not look like the usual symmetric butterfly curve since single-ended SRAM cell is used in this study. A higher good value of 840 mV is selected as the SNM, since it is the highest noise voltage where the SRAM cell will continue to operate having no stored or output bit disturbances. For the DNM, any noise can be resisted under a pulsewidth of 90 ps at a pulse voltage of 0.3 V when supply voltage (VDD) is 0.9 V and clock frequency is 100 MHz.

Fig. 31 shows the energy efficiencies of the SRAM for different SRAM densities (nm$^{-2}$) using the 40-nm process.

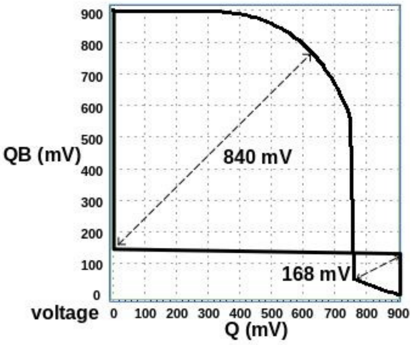

Fig. 29. Plot of the SRAM cell's SNM.

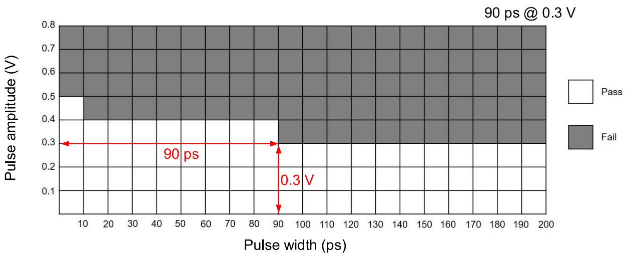

Fig. 30. Plot of the SRAM cell's DNM.

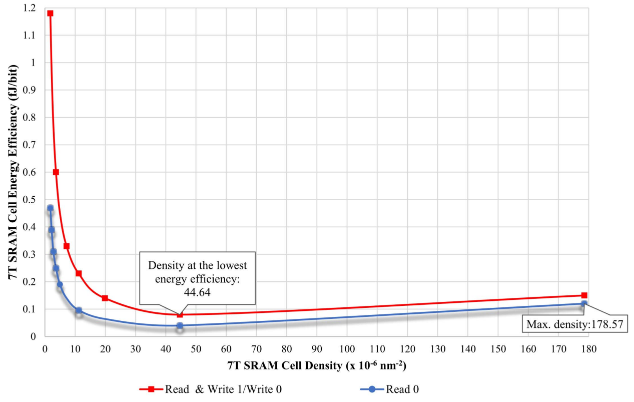

Fig. 31. 7T SRAM cell energy efficiency for different values of SRAM density.

The SRAM density is the reciprocal of the value of the SRAM transistor's gate area. As shown in the said figure, the SRAM energy efficiency decreases as the SRAM density increases. However, the energy efficiency is lower in SRAM density of $44.64 \times 10^{-6}$ nm$^{-2}$ than the energy efficiency exhibited by the SRAM at maximum density ($178.57 \times 10^{-6}$). Meanwhile, the energy efficiency of SRAM cell when reading both 0 and 1 and writing 1 is higher than the efficiency when reading 0, because BL is pulled up at those three modes as stated in Table I.

### *B. Measurement on Chips*

As shown in Fig. 32, Agilent E3631A and Agilent 81 250 were used as 0.9-V power supply and PG, respectively, while Keysight MXR254 was used to measure the CIM.

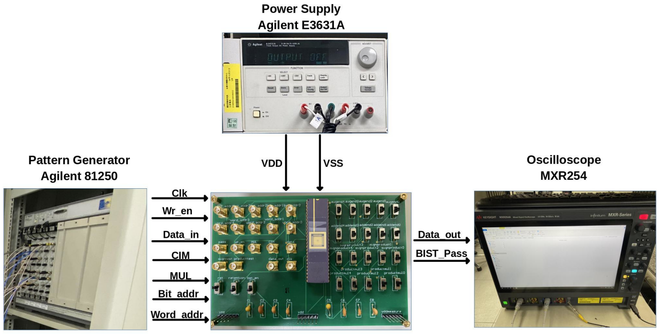

Fig. 32. Measurement setup for the proposed CIM.

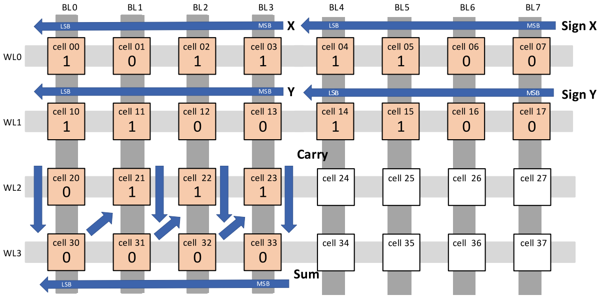

Fig. 33. Operation example of the four-bit addition.

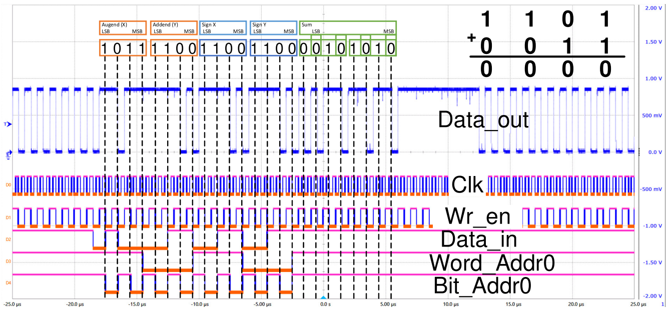

Fig. 34. Waveforms for the four-bit addition at logic frequency of 2 MHz.

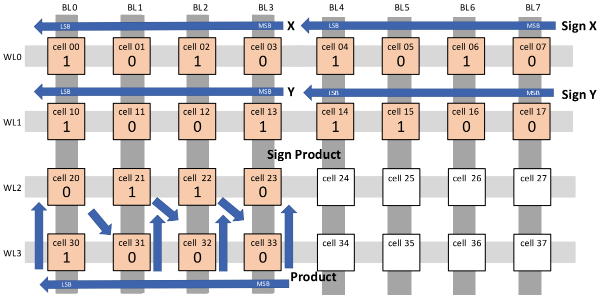

Fig. 35. Operation of the multiplication demonstrated on one word sample.

The SRAM's power consumption during read and write is 10.35 and 10.17 $\mu$W, respectively. The CIM's power consumption is 0.73 mW at VDD $=$ 0.9 V at system clock $=$ 100 MHz. Its instructional performance or throughput is $8/(1.43 \times 10^{-9}) = 5.594$ GOPS $= 0.005594$ TOPS, where $1.43 \times 10^{-9}$ is the precharge value of a normal processing element (PE) cell for the addition and multiplication operation; and since there are four input bits in one set and there are two operations in the RCAM, there is a total of eight sets in parallel calculation simultaneously. Its CIM energy efficiency (ratio of throughput and power consumption) is 7.66 TOPS/W. Its area efficiency (throughput/core area) is 0.027 TOPS/mm$^2$. The density of on-chip SRAM is 22.513%. The CIM has greater energy efficiency at 0.8 V (8.5 TOPS/W) while lesser energy efficiency at 1 V (7 TOPS/W). Moreover, it still performs best at the lowest possible voltage of 0.57 V when the clock frequency is 5 MHz. At a maximum clock frequency of 50 MHz, it can be operated at a supply voltage of 0.6 V.

The proposed CIM cannot perform multiplication and addition at the same time. However, it can perform NOR, AND, NAND, XOR, and XNOR basic logic functions at the same time. Its SRAM can read and write data at 100 MHz but the correct calculation can be achieved at 50 MHz. To clarify the addition operation of the CIM, we let augend $X$ (1101) $+$ addend $Y$ (0011) $=$ sum $Y$ (0000) as shown in Fig. 33. With reference to Fig. 33, cell 20 (Carry bit) is initially low. Next, the values of first three cells along the block line (BL0), namely 00, 10, and 20, are added; the sum is written in the fourth cell of the same block line (cell 30) and the carry is written in the third cell of the next block line (BL1). The process is repeated until the sum is expected. The output waveforms at logic frequency of 2 MHz for the CIM's adder are displayed in Fig. 34. The sequence of the resulting bits can be seen at the arrow flow from cell 20 to cell 33. Meanwhile, to explain the multiplication operation of the CIM, we implement the multiplication on a sample word for demonstration purpose to show our proposed chip can execute four-bit multiplication. The signed bits for positive and negative are represented as 0 and 1 (as reflected in Sign $X$ and Sign $Y$), respectively. We represent multiplicand bits $-1$, $+0$, $-1$, and $+0$, that is, $X = [(-1)\ (+0)\ (-1)\ (+0)]$ and multiplier bits $+1$, $+0$, $-0$, and $-1$, that is, $Y = [(-1)\ (-0)\ (+0)\ (+1)]$ which generate results, namely $+1$, $-0$, $-0$, and $+0$, that is, Product $= [(+1)\ (-0)\ (-0)\ (+0)]$ as shown in Fig. 35. With reference to Fig. 35, the product of the bits on WL0 and WL1 is reflected on WL3 while the sign of the Product is reflected on WL2. The sequence of the resulting bits can be seen at the arrow flow from cell 30 to cell 23. The output waveforms at logic frequency of 2 MHz for the CIM's multiplier are presented in Fig. 36. Meanwhile, Figs. 37 and 38 illustrate the waveforms for the CIM's addition and multiplication operations at a maximum logic frequency of 50 MHz, respectively. Finally, Figs. 39 and 40 show the waveforms for the BIST's normal and retention modes, respectively.

Table IV shows the performance comparison of the proposed CIM with several recent CIM architectures. The proposed CIM chip measurement results show that the normalized average energy is higher, since our design (40-nm process) attains the highest clock rate. If compared with [12] and [11] which also had measurement results, our figure of merit (FOM) is smaller than that of [11] and [12] (both used

TABLE IV
PERFORMANCE COMPARISON OF SRAM-BASED CIM ARCHITECTURES

| | | Electronics [19] | JSSC [18] | JSSC [17] | TVLSI [20] | JSSC [13] | TCAS-1 [12] | CICC [11] | This work |
|---|---|---|---|---|---|---|---|---|---|
| Year | | 2021 | 2019 | 2018 | 2017 | 2017 | 2019 | 2015 | 2021 |
| Process (nm) | | 28 | 40 | 65 | 65 | 40 | 28 | | TSMC 40 |
| Verification | | Simul. | Meas. | Meas. | Meas. | Meas. | Meas. | | Meas. |
| Supply Voltage (V) | | 0.9 | 1.1 | 0.6-1.1 | 1.2 | 0.6 | 0.7 | | 0.9 |
| Cell Type | | 8T | - | 6T | 6T | 5T | 8T | | 7T (single-ended) |
| Operation | | Unsigned Multiplication, Ternary Multiplication, Logic | SRAM for Deep Learning | SRAM for Deep Learning | SRAM | SRAM for Face Recognition | SRAM for Image Recognition | | NAND<br>NOR<br>XOR<br>SRAM for AI Applications<br>Addition,<br>Signed Multiplication,<br>Normal mode<br>Retention mode |
| Array Size | | 32×32 | N.A. | N.A. | 32×32 (1 Kb) | 4 Mb | 64 Kb | | 1 Kb |
| Write | Freq. (MHz) | N.A. | N.A. | N.A. | 100 | N.A. | 28.6 | 18.2 | 100 |
| | Norm. energy[1] (fJ/bit) | | | | 39.5 | | 21.1 | 23.7 | 63.6 |
| Read | Freq. (MHz) | N.A. | N.A. | N.A. | 166 | 100 | 28.6 | 18.2 | 100 |
| | Norm. energy[2] (fJ/bit) | | | | 4.1 | 64 | 31.1 | 51.8 | 64.7 |
| FOM[3] (fJ/bit/MHz) | | 0.9[4] | N.A. | N.A. | 0.42 | 0.64 | 0.92 | 2.07 | 0.64 |
| CIM energy efficiency (TOPS/W) | | N.A. | 0.877 | 2.3-6.0 | N.A. | N.A. | N.A. | N.A. | 7.66 |
| CIM area efficiency (GOPS/mm$^2$) | | N.A. | - | 89-365 | N.A. | N.A. | N.A. | N.A. | 27 |

[1] Norm. write energy $= \frac{Write\ energy}{Process^2} \times 10^3$

[2] Norm. read energy $= \frac{Read\ energy}{Process^2} \times 10^3$

[3] FOM $= \frac{Avg.\ energy}{Frequency\ (MHz) \times Process\ (nm)^2} \times 10^3$; Avg. energy $= \frac{Norm.\ read\ energy\ + Norm.\ write\ energy}{2}$

[4] The average of the energy-to-logic frequency ratio for ternary multiplication, unsigned multiplication, and logic operation was calculated.

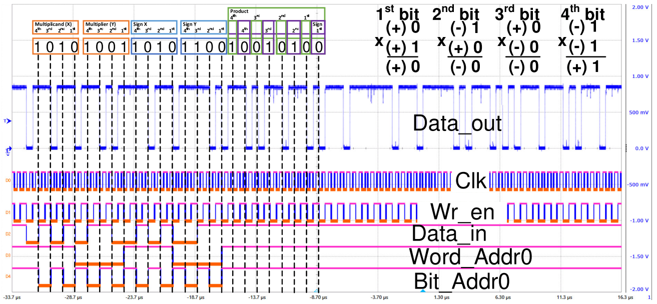

Fig. 36. Waveforms for the four-bit multiplication at logic frequency of 2 MHz.

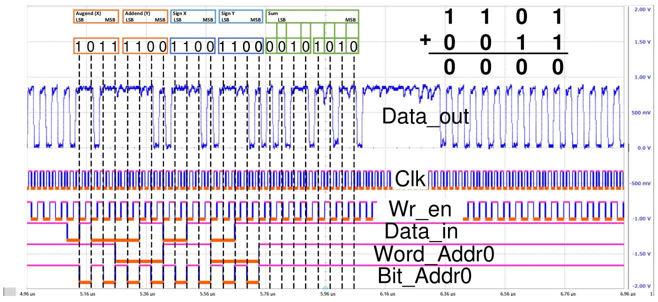

Fig. 37. Waveforms for the four-bit addition at maximum logic frequency of 50 MHz.

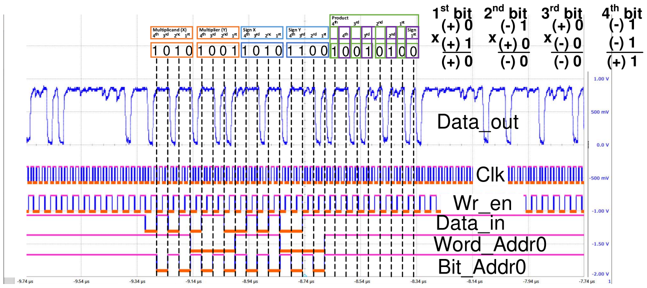

Fig. 38. Waveforms for the four-bit multiplication at a maximum logic frequency of 50 MHz.

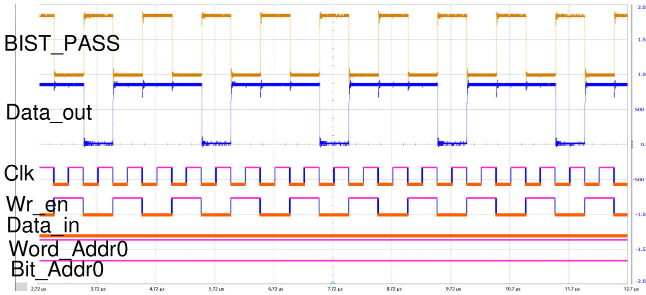

Fig. 39. Waveforms of BIST normal mode.

28-nm process) which is 0.92 and 2.07, respectively. The only work which has a better FOM than our work is [12]. However, this work is a pure SRAM without any computation capability. Though another prior work had a similarly FOM to our work [13], its write function was not proved. Moreover, our CIM's energy efficiencies are greater compared to CIMs

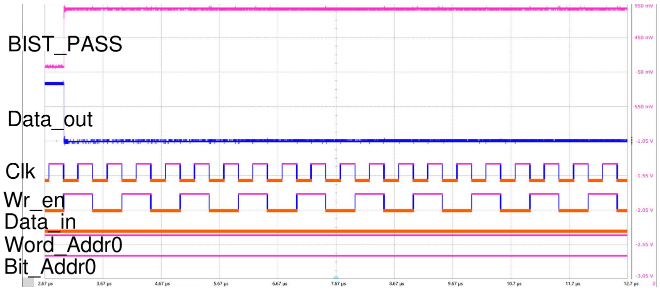

Fig. 40. Waveforms of BIST retention mode.

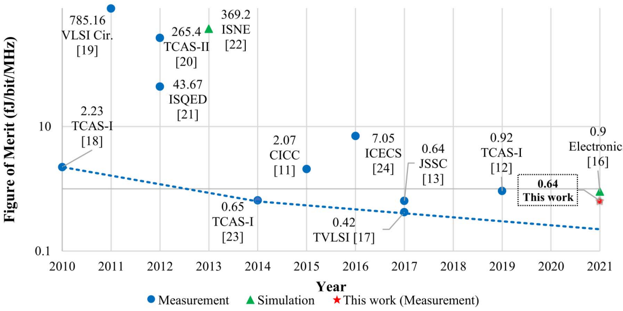

Fig. 41. Technology roadmap of CIMs using SRAM and DRAM.

in [17] and [18]. Besides, our CIM has the most number of performed operations (NAND, NOR, XOR, addition, and multiplication) and functions (normal and retention modes for BIST) among the prior CIMs. Finally, a technology roadmap of CIMs using SRAM and DRAM is presented in Fig. 41. As shown in this roadmap, this proposed CIM has a better performance in FOM than the prior works. It can be seen from the dotted line trend that the FOM of this work is close to the projected FOM for year 2021.

## V. Conclusion

This work presents a 40-nm CMOS-based multifunctional CIM architecture using single-ended disturb-free 7T 1-Kb SRAM. The proposed CIM resolves the problem of von Neumann bottleneck, the accumulation issues in the 5T SRAM, and the high power consumption and large chip area through FS-GDI circuitry. Finally, it performs the most number of operations and functions among the CIMs to date.

## Acknowledgment

The authors would like to thank Taiwan Semiconductor Research Institute (TSRI), Hsinchu, Taiwan, for the fabrication and measurements of the chip.

## References

[1] J. Backus, "Can programming be liberated from the von Neumann style? A functional style and its algebra of programs," *Commun. ACM*, vol. 21, no. 8, pp. 613–641, Aug. 1978.

[2] S. Jain, A. Ranjan, K. Roy, and A. Raghunathan, "Computing in memory with spin-transfer torque magnetic RAM," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 3, pp. 470–483, Mar. 2018.

[3] Q. Dong *et al.*, "A 4 + 2T SRAM for searching and in-memory computing with 0.3-V $V_{\mathrm{DDmin}}$," *IEEE J. Solid-State Circuits*, vol. 53, no. 4, pp. 1006–1015, Apr. 2018.

[4] A. Agrawal, A. Jaiswal, C. Lee, and K. Roy, "X-SRAM: Enabling in-memory Boolean computations in CMOS static random access memories," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 12, pp. 4219–4232, Jul. 2018.

[5] C.-C. Wang, Y.-L. Tseng, H.-Y. Leo, and R. Hu, "A 4-kB 500-MHz 4-T CMOS SRAM using low-$V_{\mathrm{THN}}$ bitline drivers and high-$V_{\mathrm{THP}}$ latches," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 9, pp. 901–909, Sep. 2004.

[6] C.-C. Wang, C.-L. Lee, and W.-J. Lin, "A 4-kB low-power SRAM design with negative word-line scheme," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 5, pp. 1069–1076, May 2007.

[7] C.-C. Wang and C.-L. Hsieh, "Disturb-free 5 T loadless SRAM cell design with multi-Vth transistors using 28 nm CMOS process," in *Proc. Int. SoC Design Conf. (ISOCC)*, Oct. 2016, pp. 103–104.

[8] C.-C. Wang, N. Sulistiyanto, T.-Y. Tsai, and Y.-H. Chen, "Multifunctional in-memory computation architecture using single-ended disturb-free 6T SRAM," in *Advances in Electronics Engineering* (Lecture Notes in Electrical Engineering), vol. 619. Singapore: Springer, 2020, pp. 49–57.

[9] A. Morgenshtein, A. Fish, and I. A. Wagner, "Gate-diffusion input (GDI): A power-efficient method for digital combinatorial circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 10, no. 5, pp. 566–581, Oct. 2002.

[10] M. A. Ahmed and M. A. Abdelghany, "Low power 4-bit arithmetic logic unit using full-swing GDI technique," in *Proc. Int. Conf. Innov. Trends Comput. Eng. (ITCE)*, Feb. 2018, pp. 193–196.

[11] H. Mori *et al.*, "A 298-fJ/writecycle 650-fJ/readcycle 8 T three-port SRAM in 28-nm FD-SOI process technology for image processor," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Sep. 2015, pp. 1–4.

[12] H. Mori *et al.*, "A 28-nm FD-SOI 8T dual-port SRAM for low-energy image processor with selective sourceline drive scheme," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 4, pp. 1442–1453, Apr. 2019.

[13] D. Jeon *et al.*, "A 23-mW face recognition processor with mostly-read 5 T memory in 40-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 52, no. 6, pp. 1628–1642, Jun. 2017.

[14] R. O. S. Juan and J. Kim, "Photovoltaic cell defect detection model based-on extracted electroluminescence images using SVM classifier," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 578–582.

[15] T. M. Amado *et al.*, "Development of predictive models using machine learning algorithms for food adulterants bacteria detection," in *Proc. IEEE HNICEM*, Nov. 2019, pp. 1–6.

[16] J. S. Velasco *et al.*, "Alphanumeric test paper checker through intelligent character recognition using openCV and support vector machine," in *Proc. World Congr. Eng. Technol. Innov. Sustainability*, Nov. 2018, pp. 119–128.

[17] K. Ando *et al.*, "BRein memory: A single-chip binary/ternary reconfigurable in-memory deep neural network accelerator achieving 1.4 Tops at 0.6 W," *IEEE J. Solid-State Circuits*, vol. 53, no. 4, pp. 983–994, Apr. 2018.

[18] K. Ueyoshi *et al.*, "QUEST: Multi-purpose log-quantized DNN inference engine stacked on 96-MB 3-D SRAM using inductive coupling technology in 40-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 186–196, Jan. 2019.

[19] J. Zhang *et al.*, "An 8 T SRAM array with configurable word lines for in-memory computing operation," *Electronics*, vol. 10, no. 3, pp. 1–13, Jan. 2021.

[20] J. Lee, D. Shin, Y. Kim, and H.-J. Yoo, "A 17.5-fJ/bit energy-efficient analog SRAM for mixed-signal processing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 10, pp. 2714–2723, Oct. 2017.

[21] M.-H. Tu, J.-Y. Lin, M.-C. Tsai, S. J. Jou, and C.-T. Chuang, "Single-ended subthreshold SRAM with asymmetrical write/read-assist," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 12, pp. 3039–3047, Dec. 2010.

[22] S. Yoshimoto *et al.*, "A 40-nm 0.5-V 20.1-$\mu$W/MHz 8 T SRAM with low-energy disturb mitigation scheme," in *Symp. VLSI Circuits-Dig. Tech. Papers*, Jun. 2011, pp. 72–73.

[23] M.-H. Chang, Y.-T. Chiu, and W. Hwang, "Design and iso-area $V_{\mathrm{min}}$ analysis of 9 T subthreshold SRAM with bit-interleaving scheme in 65-nm CMOS," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 7, pp. 429–433, Jul. 2012.

[24] M. Terada *et al.*, "A 40-nm 256-kB 0.6-V operation half-select resilient 8 T SRAM with sequential writing technique enabling 367-mV VDDmin reduction," in *Proc. 13th Int. Symp. Qual. Electron. Design (ISQED)*, Mar. 2012, pp. 489–492.

[25] L.-Y. Chiou, C.-R. Huang, C.-C. Cheng, and Y.-L. Tsai, "A 300 mV sub-1 pJ differential 6 T sub-threshold SRAM with low energy and variability resilient local assist circuit," in *Proc. Int. Symp. Next-Gener. Electron.*, Feb. 2013, pp. 337–340.

[26] N.-C. Lien *et al.*, "A 40 nm 512 kB cross-point 8 T pipeline SRAM with binary word-line boosting control, ripple bit-line and adaptive data-aware write-assist," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 12, pp. 3416–3425, Dec. 2014.

[27] H. Mori *et al.*, "An low-energy 8 T dual-port SRAM for image processor with selective sourceline drive scheme in 28-nm FD-SOI process technology," in *Proc. IEEE Int. Conf. Electron., Circuits Syst. (ICECS)*, Dec. 2016, pp. 532–535.

**Chua-Chin Wang** (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the State University of New York (SUNY) at Stony Brook, Stony Brook, NY, USA, in 1992.

Since 1998, he has been a Full Professor with the Department of Electrical Engineering, National Sun Yat-sen University (NSYSU), Kaohsiung, Taiwan. From 2012 to 2014, he was the CEO of the Operation Center of Industry—University Cooperation, NSYSU. From 2014 to 2015, he was designated as the NSYSU's Vice President of the Office of Industrial Collaboration and Continuing Education Affairs. From 2014 to 2017, he was its Dean of the College of Engineering. Since 2018, he has also been the Director of the Underwater Vehicle Research and Development Center, NSYSU. He is currently the Vice President of the Office of Research and Development, NSYSU. His research interests include memory and logic circuit design, communication circuit design, biomedical circuits, and particularly interfacing I/O circuits.

Dr. Wang was a fellow of the Institution of Engineering and Technology (IET) in 2012. In the same year, he was awarded a Distinguished Engineering Professor by the Chinese Institute of Engineers and the Outstanding Research Award by NSYSU. In 2013, he was recognized as the ASE Chair Professor for his achievement. In 2018, he won the Outstanding Technical Achievement Award of the IEEE Tainan Section. He was the General Chair of the 2012 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS). From 2010 to 2013, he was an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS. From 2010 to 2011, he was an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS. He has also been a Distinguished Lecturer of the IEEE Circuits and Systems Society (CASS) since 2019.

**Lean Karlo S. Tolentino** (Student Member, IEEE) received the B.S. degree in electronics and communications engineering from the Technological University of the Philippines (TUP), Manila, Philippines, in 2010, and the M.S. degree in electronics engineering major in microelectronics from Mapúa University, Manila, in 2015. He is currently working toward the Ph.D. degree in electrical engineering at the National Sun Yat-sen University, Kaohsiung, Taiwan.

From 2010 to 2013, he worked as an IC Layout Engineer with the Mask Design Division, ROHM LSI Design Philippines, Inc., Pasig, Philippines. Since 2015, he has been a Faculty Member with TUP. From 2017 to 2019, he was the Head (Chair) of the Department of Electronics Engineering, TUP. He was designated as the Director of the TUP's Office, University Extension Services (UES), Manila, from 2019 to 2020. His research interests include artificial intelligence, IC design, and power electronics.

Mr. Tolentino has been serving as one of the members of the Technical Committee on Audio, Video, and Multimedia Equipment (TC 59) and the Technical Committee on Electromagnetic Compatibility (TC 74) of the Bureau of Philippine Standards under the Department of Trade and Industry of the Philippine Government since 2017.

**Chia-Yi Huang** was born in Taiwan in 1997. He received the B.S. and M.S. degrees from the National Sun Yat-sen University (NSYSU), Kaohsiung, Taiwan, in 2019 and 2021, respectively.

His current research interests include CIM and analog design.

**Chia-Hung Yeh** (Senior Member, IEEE) received the B.S. and Ph.D. degrees from the Department of Electrical Engineering, National Chung Cheng University, Chiayi, Taiwan, in 1997 and 2002, respectively.

He has served as an Assistant Professor from 2007 to 2010, an Associate Professor from 2010 to 2013, and a Professor from 2013 to 2017 at the Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan. He is currently a Distinguished Professor with Taiwan's National Taiwan Normal University (NTNU), Taipei, Taiwan. He has coauthored over 250 technical international conference papers and journal articles. He is the holder of 47 patents in the U.S., Taiwan, and China. His research interests include deep learning, video coding, 3-D reconstruction, and image/video processing.

Dr. Yeh was elected as a fellow of the Institution of Engineering and Technology (IET) in 2017. He received the IEEE Multimedia Signal Processing (MMSP) Top 10% Paper Award in 2013, the IEEE Global Conference on Consumer Electronics (GCCE) Outstanding Poster Award in 2014, the Asia–Pacific Signal and Information Processing Association (APSIPA) Distinguished Lecturer in 2015, the NTNU Distinguished Professor Award in 2017, and the Outstanding Technical Achievement Award from IEEE Tainan Section. He was the 2017 IEEE Signal Processing Society (SPS) Tainan Section Chair. He has served as an Associate Editor for the *Journal of Visual Communication and Image Representation*, the *EURASIP Journal of Advances in Signal Processing*, and the *International Journal of Pattern Recognition and Artificial Intelligence*.