15.3 A 351TOPS/W and 372.4GOPS Compute-in-Memory SRAM Macro in 7nm FinFET CMOS for Machine-Learning Applications

Qing Dong[1], Mahmut E. Sinangil[1], Burak Erbagci[1], Dar Sun[2],
Win-San Khwa[2], Hung-Jen Liao[2], Yih Wang[2], Jonathan Chang[2]

[1]TSMC, San Jose, CA
[2]TSMC, Hsinchu, Taiwan

Compute-in-memory (CIM) parallelizes multiply-and-average (MAV) computations and reduces off-chip weight access to reduce energy consumption and latency, specifically for AI edge devices. Prior CIM approaches demonstrated tradeoffs for area, noise margin, process variation and weight precision. 6T SRAM [1-3] provides the smallest cell area for CIM, but cell stability limits the number of activated cells, resulting in low parallelization. 10T and twin-8T [4-5] isolate the read/write paths for noise margin improvement, however both require special design of the bit cell using logic layout rules, resulting in over a 2× area overhead compared to foundry yield-optimized 6T SRAMs. Furthermore, single-bit precision of weights, in prior work [1-4], cannot meet the requirement for high-precision operations and scalability for large neural networks.

This work presents a CIM macro built around a standard two-port compiler macro using foundry 8T bit-cell. 8T SRAM provides sufficient noise margin compared to 6T SRAM and ensures stable multi-word activation for CIM operation at the expense of ~30% bit-cell area overhead. The proposed design supports MAV of 64 4b inputs with 16 4b weights in one computation cycle. Number of RWL pulses is used to represent the 4b input. RWL pulses are precisely controlled by row-wise 4b digital counters, which is more variation-tolerant and compact compared to row-wise DAC or analog delay line [1-2]. 4b weight is realized by charge sharing among binary-weighted computation caps. Each unit of computation cap is formed by the inherent cap of sense amplifier (SA) inside the 4b Flash ADC to save area and minimize kick-back effect. 64×64 8T macro is fabricated in 7nm FinFET technology and achieves 5.5ns access time with 0.8V power supply at room temperature. Energy efficiency is 351 TOPS/W and throughput is 372.4 GOPS for 1024 (64×16) 4×4b MAV operations.

Figure 15.3.1 shows the detailed block diagram of CIM macro with 64×64 8T SRAM cells. Besides peripheral circuits of standard two-port compiler SRAM macro, row-wise RWL counters and column-wise Flash ADCs with compensation caps are added to perform MAV computations. With single-ended read bit-line (RBL) and decoupled read/write path, 8T push-rule SRAM cell is used to balance cell stability and area overhead, occupying an area of only 0.053μm^2 in 7nm FinFET technology. Each row has a 4b digital counter which converters the 4b input value into number of RWL pulses. Four SRAM cells in the same row are combined together as 4b weight. Each four RBLs share one 4b Flash ADC. Compensation caps are added on each RBL so that each RBL contains equal lumped capacitance of 9*Cu.

Figure 15.3.2 illustrates the multi-bit weight realization using charge sharing among computation caps inside Flash ADC. From LSB to MSB, the corresponding RBL connects to computation caps with 1:2:4:8 capacitance ratios. Before RWL activation, all the caps on the RBL are pre-charged to VDD first. In RBL pre-charging and sampling phases, both compensation caps and computation caps are connected to the RBL and each RBL has same amount of capacitance. RBL sampling will start once RWLs fire according to the 4b inputs. The voltage on each RBL will be lowered by the discharge currents (if corresponding bit-cell is storing a '1'), and the final voltage is determined by the multiplication of 4b inputs and the single-bit weights stored in the bit-cell. After RBL sampling, the computation caps are isolated from RBLs. Each binary-weighted computation cap holds the voltage same as its corresponding RBL. Charge sharing happens among the computation caps, averaging out the voltage. The voltage represents the MAV result of 64 4×4b, which will be converted by the Flash ADC into 4b digital output.

Column-wise Flash ADC uses area-efficient SA instead of analog comparator to save area and reduce energy consumption. Each 4b Flash ADC consists of 15 SAs. In this design, we use inherent cap inside each SA as unit cap (Cu). As shown in Fig. 15.3.3, each Cu consists of gate caps of MPL/MNL/MPR/MNR in the SA. One drawback of MOS cap is its voltage dependency. To deal with this issue, the RBL[3:0] is distributed among SA0 to SA14 in a balanced fashion, such that the voltage dependency can be further averaged out and capacitance ratio (1:2:4:8) is well maintained. Moreover, as the gate caps of MNL/MPR are complementary to V_{RBL}-V_{REF} and that of MPL/MNR has proportional correlation, the voltage dependency of MOS caps can be further mitigated. Lastly, layout of ADC is implemented with appropriate wire lengths to make sure wiring cap contribution is also binary-weighted. Unlike analog comparators that require pre-amp or offset caps to minimize kick-back effect, the proposed SA is immune to this effect because of the self-sampling on the SA internal caps. Also, the SH switches use transmission gate and turn off before SAE fires for SA evaluation, further reducing kick-back effect. The area of each 4b Flash ADC in 7nm FinFET technology is 13.62μm by 4 SRAM cell pitches. Figure 15.3.3 shows the simulated integral nonlinearity (INL) of the proposed 4b SA-based Flash ADC.

Since the RBL has finite capacitance, larger discharge current (I_{DS}) can quickly drain RBL to ground, which limits the dynamic range; while smaller I_{DS} is more susceptible to process variation. To limit the I_{DS}, we applied tunable pulse width and voltage for the RWL control. We also placed extra metal cap on each RBL to make sure total capacitance of RBL can be sufficient enough to cover dynamic range, relaxing I_{DS} requirement. Furthermore, the final voltage drop on RBL is decided by not only the MAV of inputs and weights but also I_{DS} of the 8T SRAM cell (Fig. 15.3.4). Above 300mV on the RBL, the saturation region I_{DS} is mostly insensitive to V_{DS}; while below 300mV, the I_{DS} starts to go into linear region where it is significantly more sensitive to V_{DS}. This transistor characteristic is fundamental and common to all bit-cell types. As a result, the activation function is not ideal and linear in the full dynamic range (from power supply to ground), which is the same challenge for all CIM works [1-5] using charge accumulation for computation. We address the variation and non-linearity by incorporating the non-linear I_{DS} correlation and the distribution of I_{DS} into training. As shown in Fig. 15.3.4 (right), the normal distribution of MAV results is shaped to reflect the non-linearity of I_{DS} and fit the linear part of dynamic range. With incorporating the non-linearity in the training, the accuracy of MNIST dataset is improved by 3.1% for an MLP with single hidden layer. Figure 15.3.4 also shows the accuracy saturates at 4b precision for MNIST dataset.

The proposed CIM macro is fabricated in 7nm FinFET technology (Fig. 15.3.7). The 64×64 8T SRAM CIM macro (including all test and re-configurability features) occupies an area of 0.0032mm^2. Figure 15.3.5 (top) shows the measured energy consumption across MAV results and power supply. The energy consumption of each pattern is determined by the number of 1s on both inputs and weights. The maximum energy consumption at 0.8V is 7.8pJ for the whole macro computing maximum MAV results at all 16 neurons. Figure 15.3.5 (bottom) shows the measured shmoo plots of CIM function and SRAM function of the 8T macro,. The cycle time of CIM for parallel 1024 4×4b MAV computation at 0.8V is 5.5ns. Compared with the listed references (Fig. 15.3.6), the proposed CIM macro achieves best energy efficiency and throughput with high precision. The proposed additional CIM peripherals can be compatible and easily applied to all other 8T compiler macros with different size.

Acknowledgement:
The authors wish to thank Peter Hsu, Bing Wang, Yunle Sun, Vincent Tsai, Matt Lin and Sharon Wang for their great help.

References:
[1] J. Zhang et al., "In-Memory Computation of a Machine-Learning Classifier in a Standard 6T SRAM Array," *JSSC*, vol. 52, no. 4, pp. 915-924, 2017.
[2] S. K. Gonugondla et al., "A 42pJ/Decision 3.12TOPS/W Robust In-Memory Machine Learning Classifier with On-Chip Training," *ISSCC*, pp. 490-491, Feb. 2018.
[3] W.-S. Khwa et al., "A 65nm 4Kb Algorithm-Dependent Computing-in-Memory SRAM Unit-Macro with 2.3ns and 55.8TOPS/W Fully Parallel Product-Sum Operation for Binary DNN Edge Processors," *ISSCC*, pp. 496-497, Feb. 2018.
[4] A. Biswas et al., "Conv-RAM: An Energy-Efficient SRAM with Embedded Convolution Computation for Low-Power CNN-Based Machine Learning Applications," *ISSCC*, pp. 488-489, Feb. 2018.
[5] X. Si et al., "A Twin-8T SRAM Computation-In-Memory Macro for Multiple-Bit CNN-Based Machine Learning," *ISSCC*, pp. 396-397, Feb. 2019.

978-1-7281-3205-1/20/$31.00 ©2020 IEEE

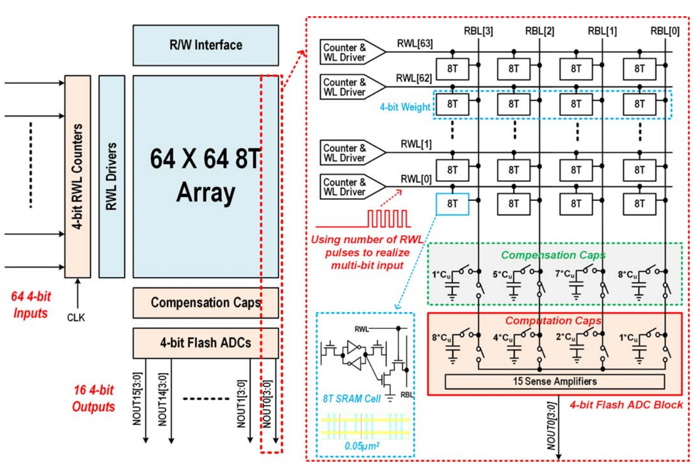


Figure 15.3.1: CIM SRAM architecture designed around a standard two-port macro using a foundry 8T SRAM in a 7nm FinFET technology. Proposed design can do 64×16 4b input/weight multiplication in one computation cycle. Multi-bit input is realized by a repeated number of RWL pulses.

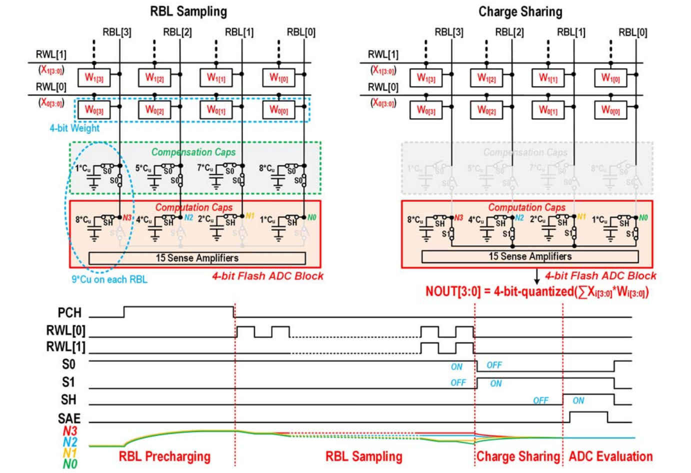


Figure 15.3.2: A 4b weight is realized using charge sharing with binary-weighted capacitors. Compensation capacitors are connected to make sure all RBLs have an equal capacitance during RBL sampling and are disconnected during binary-weighted charge sharing.

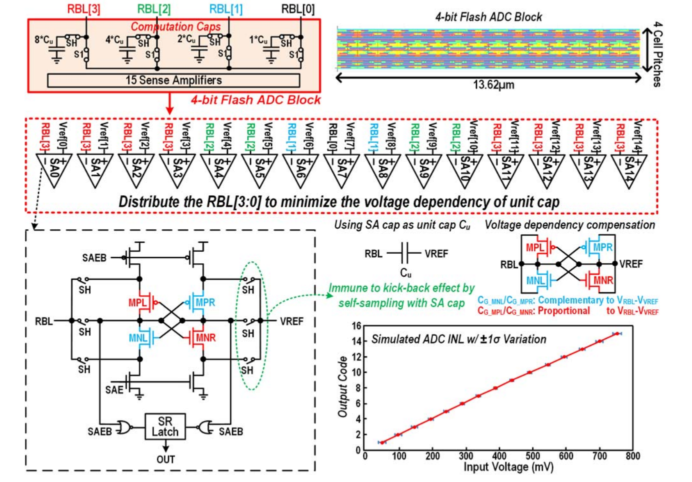

RBL[3]
RBL[2]
RBL[1]
RBL[0]
Computation Caps
15 Sense Amplifiers
4-bit Flash ADC Block
4-bit Flash ADC Block
13.62µm
4 Cell Pitches
SA0
SA1
SA2
SA3
SA4
SA5
SA6
SA7
SA8
SA9
SA10
SA11
SA12
SA13
SA14
Distribute the RBL[3:0] to minimize the voltage dependency of unit cap
Using SA cap as unit cap Cu
Voltage dependency compensation
SAEB
SH
MPL
MPR
MNL
MNR
RBL
VREF
SAE
SR Latch
OUT
Immune to kick-back effect by self-sampling with SA cap
CG_MNL/CG_MPR: Complementary to VRBL-VREF
CG_MPL/CG_MNR: Proportional to VRBL-VREF
Simulated ADC INL w/ ±1σ Variation
Output Code
Input Voltage (mV)

Figure 15.3.3: Kick-back and area is reduced by using an SA capacitor as a unit capacitor. The voltage dependency of the MOS capacitor is minimized by distributing RBL connections across 15 SAs in a balanced way and by using cross-coupled compensation.

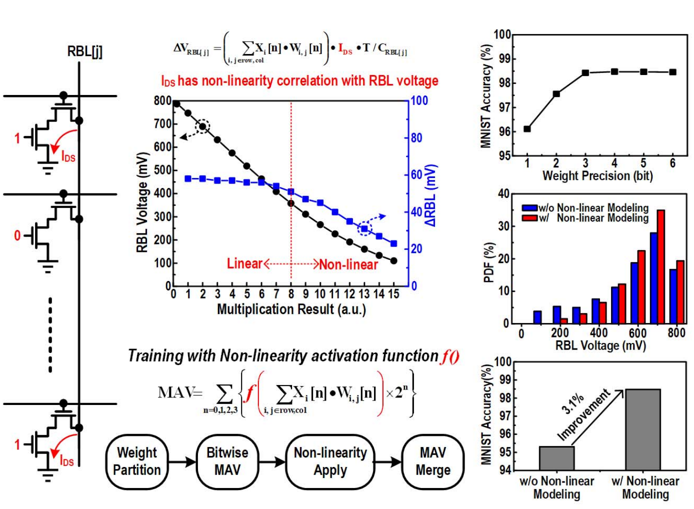


Figure 15.3.4: Considering silicon non-linearities while training improves the modeling accuracy.

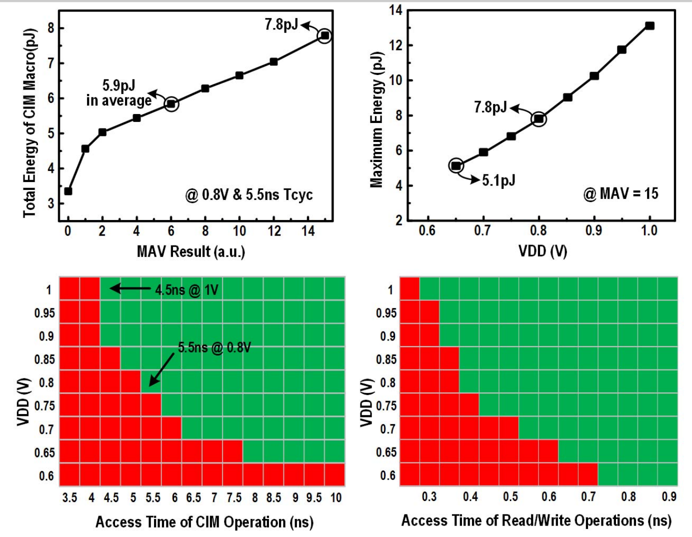


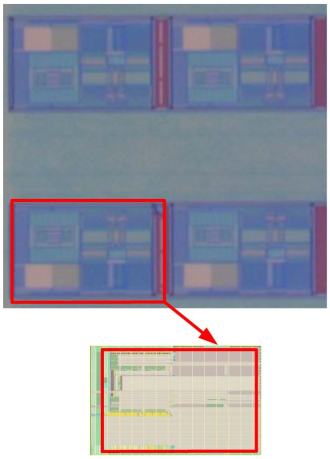
Figure 15.3.5: Summary of measurement results: the CIM macro achieves a 5.5ns access time and 7.8pJ maximum energy at 0.8V at room temperature.

	JSSC'17 [1]	ISSCC'18 [2]	ISSCC'18 [3]	ISSCC'18 [4]	ISSCC'19 [5]	This Work	
Technology	130nm	65nm	65nm	65nm	55nm	7nm	
Array Size	16kb	128kb	4kb	16kb	3.8k	4kb	
Cell Type	6T	6T	S6T	10T	T8T	8T	
Push Rule	No	No	Yes	No	Yes	Yes	
Bitcell Area (µm²)	4.334	NA	0.525	NA	0.865	0.053	
Input Bits	4	8	1	7	4	4	
Weight Bits	1	8	1	1	5	4	
Output Bits	1	4	1	7	7	4	
Power Supply (V)	1.2 & 0.4	1.0	1 & 0.8	1.2 & 0.9	1.0	0.8	1.0
Cycle Time (ns)	20	NA	2.3	150	10.2	5.5	4.5
Throughput (GOPS)	NA	4	1780 1)	10.67	17.6	372.4 2)	455.1
Energy Efficiency (TOPS/W)	NA	3.125	55.8	28.1	18.4	262.3 ~ 610.5 351 in average	189.3 ~ 435.5 321 in average

1) Each operation is only 1b X 1b
2) Each 4b X 4b is considered as 2 operations

Figure 15.3.6: Feature summary and comparison to prior work.

15



Technology	7nm
Array Size	4kb
Macro Area (mm²) *	0.0032
Input/Weight/Output Precision	4 / 4 / 4
Voltage Range (V)	0.65 ~ 1
Cycle Time @ 0.8V (ns)	5.5
Max Power @ 0.8V (mW)	1.42
Max Energy @ 0.8V (pJ)	7.8
Throughput (GOPS)	372.4
Energy Efficiency (TOPS/W)	262.3 ~ 610.5 351 in average

* Including testing & reconfigurable blocks

Figure 15.3.7: Die photo and summary table.

978-1-7281-3205-1/20/$31.00 ©2020 IEEE