15.5 A 28nm 64Kb 6T SRAM Computing-in-Memory Macro with 8b MAC Operation for AI Edge Chips

Xin Si[1,2], Yung-Ning Tu[1], Wei-Hsing Huang[1], Jian-Wei Su[1], Pei-Jung Lu[1], Jing-Hong Wang[1], Ta-Wei Liu[1], Ssu-Yen Wu[1], Ruhui Liu[1], Yen-Chi Chou[1], Zhixiao Zhang[1], Syuan-Hao Sie[1], Wei-Chen Wei[1], Yun-Chen Lo[1], Tai-Hsing Wen[1], Tzu-Hsiang Hsu[1], Yen-Kai Chen[1], William Shih[1], Chung-Chuan Lo[1], Ren-Shuo Liu[1], Chih-Cheng Hsieh[1], Kea-Tiong Tang[1], Nan-Chun Lien[3], Wei-Chiang Shih[3], Yajuan He[2], Qiang Li[2], Meng-Fan Chang[1]

[1]National Tsing Hua University, Hsinchu, Taiwan
[2]University of Electronic Science and Technology of China, Chengdu, China
[3]M31 Technology, Hsinchu, Taiwan

Advanced AI edge chips require multibit input (IN), weight (W), and output (OUT) for CNN multiply-and-accumulate (MAC) operations to achieve an inference accuracy that is sufficient for practical applications. Computing-in-memory (CIM) is an attractive approach to improve the energy efficiency (EF_{MAC}) of MAC operations under a memory-wall constraint. Previous SRAM-CIM macros demonstrated a binary MAC [4], an in-array 8b W-merging with near-memory computing (NMC) using 6T SRAM cells (limited output precision) [5], a 7bIN-1bW MAC using a 10T SRAM cell (large area) [3], an 4bIN-5bW MAC with a T8T SRAM cell [1], and 8bIN-1bW NMC with 8T SRAM (long MAC latency (T_{AC})) [2]. However, previous works have not achieved high IN/W/OUT precision with fast T_{AC}, compact-area, high EF_{MAC}, and robust readout against process variation, due to (1) small sensing margin in word-wise multiple-bit MAC operations, (2) a tradeoff between read accuracy vs. area overhead under process variation, (3) limited EF_{MAC} due to decoupling of software and hardware development.

This work presents an SRAM-CIM macro using 6T SRAM for multibit MAC operations with up to 8b-IN, 8b-W, and 20b output precision for CNN applications using (1) weight bitwise MAC (WbwMAC) operations to enlarge the sensing margin and enhance IN/W/OUT precision, (2) a 6T local-computing-cell (LCC) for compact area and robust read against process variations, (3) a weight bitwise low-MAC aware readout (Wbw-LMAR) scheme based on software-hardware co-design to improve EF_{MAC}. A fabricated 28nm 64Kb 6T SRAM-CIM macro demonstrated 8bIN-8bW MAC operations featuring the world's highest output precision among SRAM-CIM works, the fastest T_{AC} (4.1-8.4ns), and the best EF_{MAC} (11.5-68.4TOPS/W) for MAC operations with 4b or higher precision in inputs and weights.

Figure 15.5.2 presents the structure of the proposed SRAM-CIM, including 8 multibit MAC computing blocks (MMACCB), 8 hybrid MAC readout blocks (HMACRB), 8 row controllers (RCB) with WL drivers (WLDRV) and horizontal global-BL (HGBL) headers (HGBL-H), multibit-input drivers (MIN-D), and other peripheral circuits. Each MMACCB comprises 8 weight-bitwise MAC cell-arrays (Wbw-MACAs) for dual 4b-weight or single 8b-weight configurations. Each Wbw-MACA includes q (64 in this work) bitwise multiply-units (BMUs) for channel accumulation. Each BMU has s (16 in this work) 6T SRAM cells and 1 local computing cell (LCC) for the multiplication of 4b-input and 1b-weight. Each HMACRB comprises 16 analog-based Wbw-LMARs and a digital weight-configuration circuit (DWCC) to combine the bitwise MAC output from the 16 HMACRBs with their respective place values.

Each 8-b weight ($W_{p,q}[7{:}0]$) of an $n{\times}n$-size kernel is stored in 2's complement format within a MMACCB in the same row-column position as Wbw-MACA [7:0]. During a MAC operation, each Wbw-MACA computes the weight-bitwise partial-MAC value (WbwpMACV) and accumulates j multiply results from the activated BMUs. The WbwpMACV results are then sent to HMACRB for weight combination to obtain the final near-full precision MAC value (MACV) for multibit MAC operations $\Sigma(IN_i[m\text{-}1{:}0] \cdot W_{p,j}[k\text{-}1{:}0])$.

Figure 15.5.3 outlines the operations of Wbw-MACA and LCC for MAC operations (j channel accumulation) with 8b-IN ($IN_i[7{:}0]$) and 1b-W ($W_i[k]$). In SRAM mode, the SRAM cells are accessed through local BL pairs (LBL/LBLB), the two pass-gates (NM1/NM2 with HWL=1) in an LLC, and the vertical global BL (VGBL/VGBLB). In standby mode, HWL=1 to precharge LBL/LBLB to V_{HWL}-$V_{TH\text{-}N}$. In CIM mode, HWL=0 to turn off MN1 and MN2. When WL=1, the LBL reflects the weight data ($W_i[k]$) stored in an accessed SRAM cell and controls the MN3/MN4 of LLC. Due to short BL length, LBL/LBLB has a fast, large voltage swing to turn on or off MN3/MN4. The 8b-input (IN[7:0]) can then be processed through two continuous phases. In each phase, a 4b-input (i.e. IN[3:0]) is split into two group and applied to VGBL (IN[3:2]) and VGBLB (IN[1:0]). Each VGBL/VGBLB uses 4 voltage levels (VDD, VIN10, VIN01 and GND) to represent a 2b-input. The N3-N5 pair then multiplies IN[3:2] and W, and outputs the result ($P_M = IN[3{:}2] \cdot W$) to the horizontal global BL (HGBL). The N2-N6 pair outputs its multiplication results ($P_L = IN[1{:}0] \cdot W$) to HGBLB.

The accumulation results $\Sigma(IN_i[3{:}2] \cdot W_i[k])$ and $\Sigma(IN_i[1{:}0] \cdot W_i[k])$ of j (16 or 32) activated BMUs (VGBL/VGBLB) respectively appear at HGBL and HGBLB, depending on the voltage-dividing behavior using a HGBL-H. The voltage at HGBL/HGBLB (V_{HGBL}/V_{HGBLB}), which represents the bitwise MAC value (WbwMACV), is then passed to HMACRB for multibit readout. The limited number of WbwMACV levels on RBL in the analog domain increases the sensing margin of the WbwMAC structure beyond that of a word-wise MAC structure.

Figure 15.5.4 shows the operations of HMACRB and Wbw-LMAR. Similar to previous works, a switch-capacitor voltage sense amplifier (VSA) is used to perform sequential multibit MAC-readout operations for V_{HGBL}/V_{HGBLB}. Analysis using multiple datasets (i.e., CIFAR-10, CIFAR-100) revealed that WbwMAC operations increased the likelihood of reading the low WbwMACV at the VSA, before the DWCC combines all of the WbwMACVs to generate a MAC value (MACV). Using the CIFAR-100 dataset, WbwpMACV = 0-3 exceeded 60%. In conventional multibit MAC readout schemes, the reference voltage (V_{REF1}) in the 1st-judgement phase of VSA is selected as the mid value of the full WbwMACV range. In the Wbw-LMAR scheme, we added an additional 1st phase to detect whether V_{HGBL}/V_{HGBLB} is above or below the target low-WbwMACV threshold (LMT), which is determined by the probability of WbwMACV for each CNN layer during training. If WbwMACV<LMT, then NLM_DET = 1 is flagged and the VSA may use fewer SAEN toggling phases to readout WbwMACV. For example, if LMT=2 and WbwMACV = 0, then the VSA requires only one LMT-detection phase (VREF1 for WbwMACV = 1.5) and one additional phase (V_{REF2} for WbwMACV = 0.5) to generate an accurate VSA output (SAOUT). If the WbwMAC>LMT, then NLM_DET = 0 and the VSA continues normal sequential multibit sensing as in conventional schemes. To achieve high EF_{MAC} and larger sensing margin without significant degradation in inference accuracy, this work translates the 48-level full-precision WbwMACV to 32-level 5b near-full (NF) precision output with 6th-bit = 0. This quantization scheme is based on software analysis of WbwMACV distribution and has less than 1b loss in the analog-domain. Finally, the DWCC combines all 16 VSAOUTs (6b WbwMACV) to generate an NF-precision MACV. In the 4bIN-4bW/4bIN-8bW case, this CIM requires only one cycle to generate the NF-precision MACV. In the 8bIN-8bW case, the CIM repeats the 4b-IN-8bW MAC operation twice to generate the 20b MACV.

Figure 15.5.5 shows the performance of the proposed scheme. Using fewer MAC levels for each VSA readout, the sensing margin of the proposed WbwMAC operation is 85×, or more, higher than that of previous word-wise MAC operations [1]. Thanks to multiple SRAM cells sharing a single LCC with a larger transistor width to suppress process variation, and the product of area and signal-margin variation is 1.24-2.18× smaller than that of previous SRAM-CIM works [1]-[4]. The proposed Wbw-LMAR scheme was shown to improve EF_{MAC} by 1.06-1.38× across various cases of low WbwMAC distribution.

Figure 15.5.6 presents the results measured from a fabricated 28nm 64Kb SRAM-CIM macro supporting up to 8bIN-8bW-20bOUT MAC operations. Each BMU (16 SRAM cells + 1 LLC) achieve 1.34× area of 16 compact-rule 6T SRAM cells and smaller array size than 8T or T8T cell arrays. Under 4bIN-8bW MAC operations with 16 channel accumulation, this CIM macro achieved T_{AC} = 4.2ns and 23.26TOPS/W for 16b-OUT. In the inference of 10K CIFAR-100 images, the BW-LMAR scheme enabled a 1.37×, or more, increase in EF_{MAC} across various IN-W precisions. This work achieved a 6.8×, or more, improvement in FoM (EF_{MAC} × Throughput × Output-ratio) thanks to higher IN-W-OUT precision, better EF_{MAC}, and faster T_{AC}, compared to previous SRAM-CIM works. Figure 15.5.7 presents a die photo.

Acknowledgement:
The authors gratefully acknowledge support from CIC, TSRI, TSMC-JDP, MTK-JDP, MOST-Taiwan, and M31.

References:

[1] X. Si et al., "A Twin-8T SRAM Computation-in-Memory Macro for Multiple-bit CNN based Machine Learning," *ISSCC,* pp. 397-399, Feb. 2019.
[2] J. Yang et al., "Sandwich-RAM: An Energy-Efficient In-Memory BWN Architecture with Pulse-Width Modulation," *ISSCC*, pp. 394-396, Feb. 2019
[3] 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.
[4] V. Khwa et al., "A 65nm 4Kb Algorithm-Dependent Computing-In-Memory SRAM Unit-Macro with 2.3ns and 55.8 TOPS/W Fully Parallel Product-Sum Operation for Binary DNN Edge Processors," *ISSCC,* pp. 496-498, Feb. 2018.
[5] S. K. Gonugondla et al., "A 42pJ/decision 3.12 TOPS/W Robust In-Memory Machine Learning Classifier with On-Chip Training," *ISSCC,* pp. 490-492, Feb. 2018.

 978-1-7281-3205-1/20/$31.00

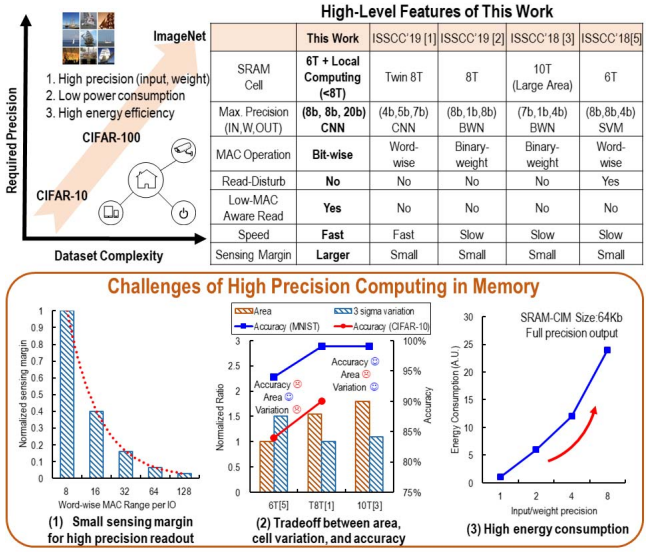


Figure 15.5.1: Motivation and high-level features of this work.

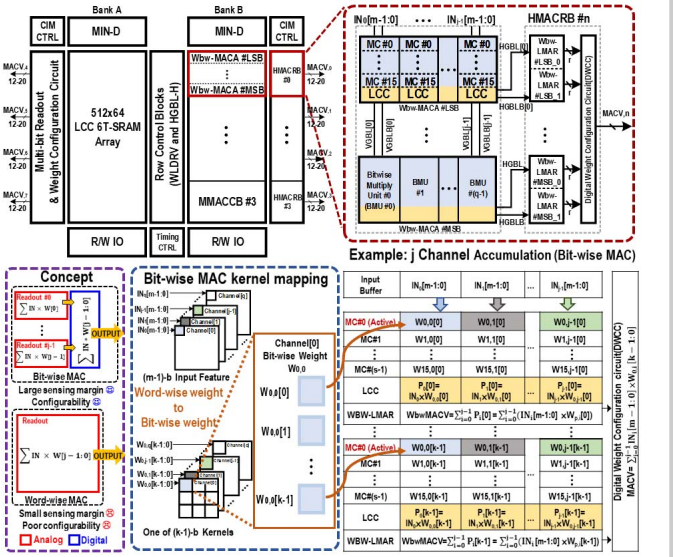


Figure 15.5.2: Macro structure and weight bitwise MAC (WbwMAC) operation.

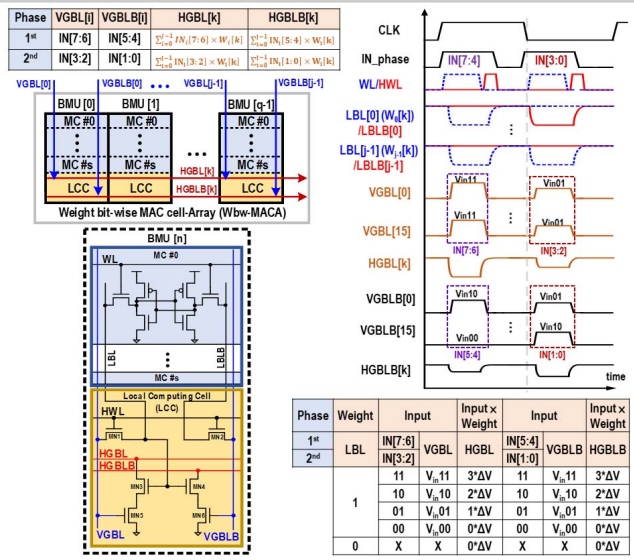


Figure 15.5.3: Wbw-MACA and LCC operation for multibit MACs.

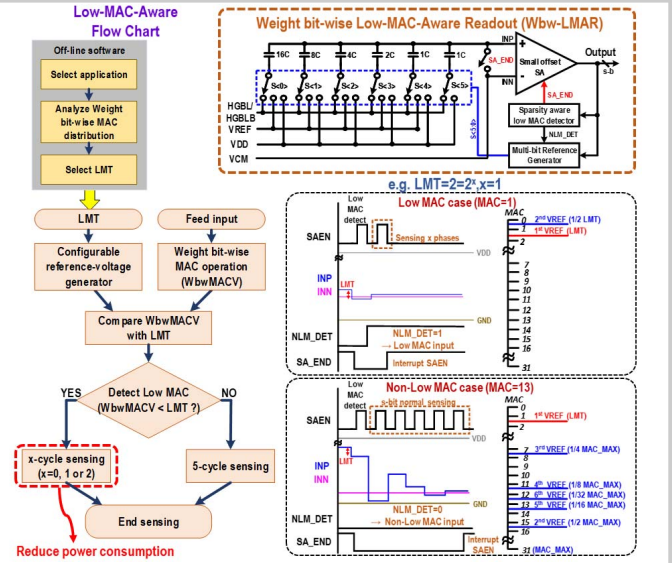


Figure 15.5.4: HMACRB and Wbw-LMAR operation.

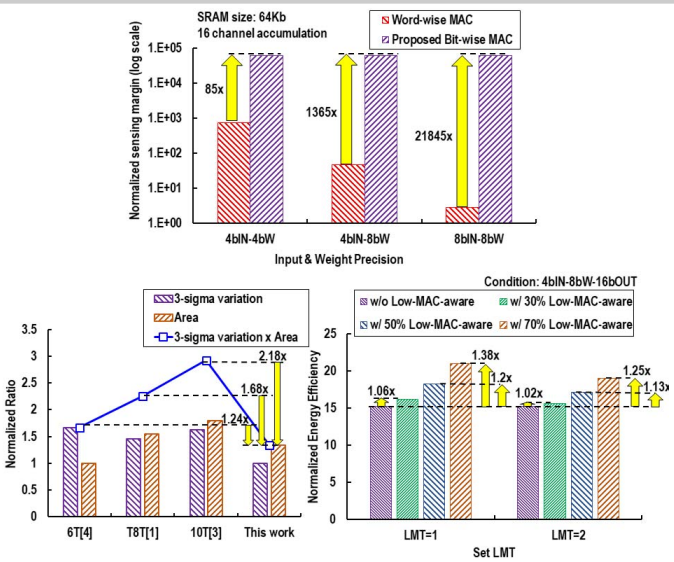


Figure 15.5.5: Simulated performance of this work.

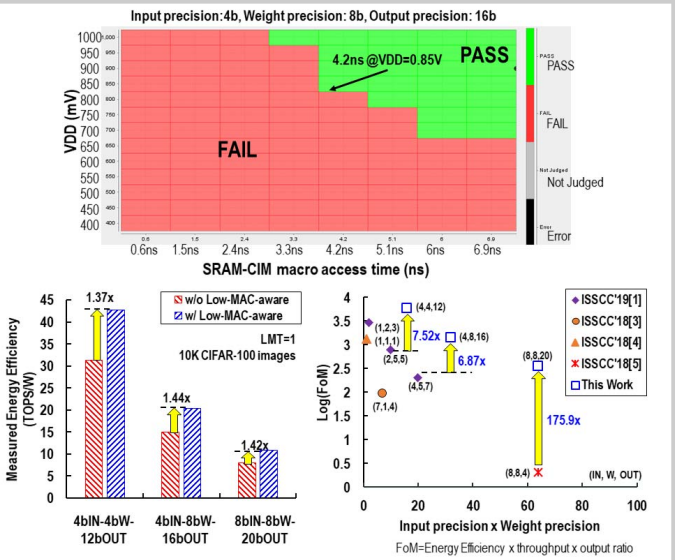
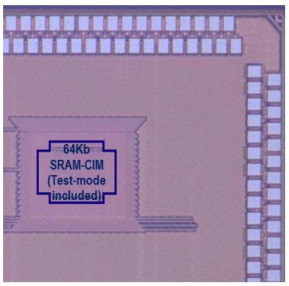


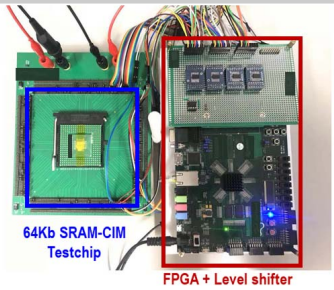
Figure 15.5.6: Measurement results.



CHIP SUMMARY			
Technology	28nm CMOS		
Macro size	64Kb		
Area of BMU (16 cells + 1 LCC)	5.795um×0.71um (Equivalent cell-size = 1.34x of compact 6T cell)		
Input precision(bit)	4	4	8
Weight precision(bit)	4	8	8
Output precision(bit)	12	16	20
Number of channels	16	16	16
Supply voltage(V)	0.7-0.9		
Cycle time(ns)	4.1	4.2	8.4
Energy efficiency (TOPS/W)	47.85-68.44	23.26-33.52	11.54-16.63
Measured Accuracy* (CIFAR10)	-	91.9%	92.02%
Measured Accuracy* (CIFAR100)	-	67.6%	67.89%

*Based on ResNet20 NN model

Figure 15.5.7: Die photo and chip summary.



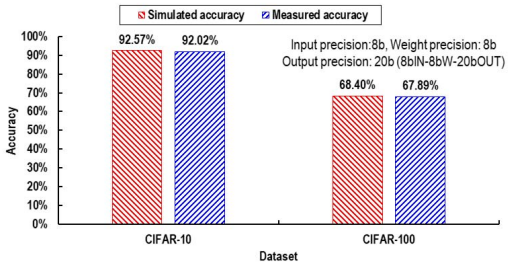


Figure 15.5.S1: Setup of the high-precision CNN demo system.

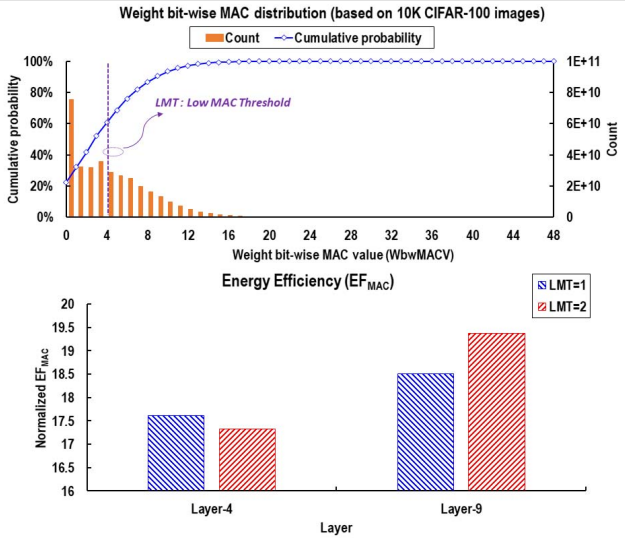


Figure 15.5.S2: Weight bitwise MAC distribution and simulated energy efficiency (EF_{MAC}).

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