24.4 Sandwich-RAM: An Energy-Efficient In-Memory BWN Architecture with Pulse-Width Modulation

Jun Yang[1], Yuyao Kong[1], Zhen Wang[2], Yan Liu[1], Bo Wang[1], Shouyi Yin[3], Longxin Shi[1]

[1]Southeast University, Nanjing, China
[2]Boxing Electronics, Nanjing, China
[3]Tsinghua University, Beijing, China

Convolutional neural networks (CNN) achieve state-of-the-art results in the field of visual perception, drastically changing the traditional computer-vision framework. However, the movement of massive amounts of data prevents CNN's from being integrated into low-power IoT devices. The recently proposed binary-weight network (BWN) reduces the complexity of computation and amount of memory access. A conventional digital implementation, which is composed of separate feature/weight memories and a multiply-and-accumulate (MAC) unit, requires large amounts of data to be moved [3]. To reduce power the weight memory and the computations are integrated together, into an in-memory computation architecture [1,2,5]. However, feature data is still stored externally, so data movement has only been partially addressed, especially for BWN. This paper blends feature and partial-weight memory with a computing circuit together, like a sandwich, that achieves significantly less data access (Fig. 24.4.1). It also uses a reconfigurable analog-computation engine, based on pulse-width modulation, that is small and flexible enough to be inserted into the memory.

Figure 24.4.2 shows the overall architecture of the sandwich–shaped computing SRAM (Sandwich-RAM): consisting of a control block for configuration, a weight pre-processing unit to reduce weight complexity, and a post-processing unit including an accumulation memory. The system supports two modes: a conventional SRAM mode, and a computing SRAM mode for convolutional operations. Sandwich-RAM contains 28×112 processing blocks, a configurable pulse generator and a pulse-quantizer by replica cell (PQRC). Each block comprises of 6×8b 8T SRAM cells to store partial feature data, several shift registers for partial weight storage, and an analog-compute unit sandwiched between the memory arrays. A pulse computation unit is shared between 6 features, and one feature is selected for computation at a time. After one calculation, the weights can be shifted for the next calculation, repeating as such until all convolutions with the stored feature map are computed. The traditional in-memory computing architecture is inflexible as it only operates on a row-by-row or column-by-column mode. This results in low utilization when the filter kernel size varies. Sandwich-RAM can be reconfigured, based on the network, into three modes: a large-, medium- or small-window mode for large, medium or small kernels. It can perform convolution operations for different kernel sizes to increase Sandwich-RAM utilization. Depending on the configuration mode from the controller, an input 3D-feature map and weight kernel can be broken up into several 2D planes and dispersed into Sandwich-RAM. A fixed-width pulse is generated on the left side of Sandwich-RAM, modulated by each pulse-width modulation unit (PWMU) and quantized on the right. The PWMU performs multiplication and accumulation in the time domain, by selecting different control voltages, depending on the local feature, to modulate the pulse width: positive values make pulses wider, while negative values make them narrower. A delay-sensitive control-voltage generator (DSCVG) adjusts the control voltages in real time to minimize the impact of supply voltage fluctuations, and a PQRC is used to measure the pulse width, which makes the analog computations more accurate and stable.

Each of the four blocks in a row can be seen as a stacked 3D-convolution map, and are bypassed when these blocks do not need to be calculated (Fig. 24.4.3). The PWMU has three paths: an add path, a subtraction path and a bypass path. They are controlled by a 2b shift-register: a 1b-weight and a 1b-bypass flag. The add and subtraction paths make the input pulse width wider or narrower. The RWL selects computation feature data F[7:0] in the partial SRAM array, which are divided into two parts: F[7:4] modulates the MSB chain and F[3:0] modulates the LSB chain. Each 2b feature selects the control voltage for a delay unit via 2:4 decoder. The control voltage V_1 is supplied from V_{DD}, while voltages V_2, V_3 and V_4 are generated by the DSCVG to guarantee the delay is 2, 3 or 4× of the delay controlled by V_1. That is for F[1:0] = 0, the control voltage is V_{DD} and pulse width is modulated by 1 unit of change. Setting F[1:0] to 1, 2, or 3 modules the pulse width by a 2, 3 or 4× of the unit change. The PWMU for F[3:2] is 4 times larger than for F[1:0], which makes the pulse change linearly.

Figure 24.4.4 shows a PQRC, which consists of dynamic C^{2}MOS sample registers, compressed adder tree, accumulation circuits and a voltage controlled oscillator(VCO) for the sample CLK. The delay cell in the VCO is as same as that in PWMU. A replica delay cell, controlled by V_{DD}, is used to mimic the unit addition/subtraction in the PWMU. The clock frequency changes with the pulse modulation in the PWMU when global PT varies. The VCO can be configured to 4-quantization accuracies, depending on the CNN network type; the generated clock measures the pulse width of the accumulation result. The PWMU output varies by 63.0% under process and temperature variation, but the quantization fluctuation is only 16.7% after PQRC. In addition, pulse width modulation is very sensitive to the control voltage, especially under PVT variation. We propose using a DSCVG for control voltage generation. A set of PMOS and NMOS are controlled by the width of the ENABLE signals. Meanwhile, V_2, V_3 and V_4 are dynamically tuned by the number of conducting PMOSs, and determine the width of ENABLE1. Ultimately, the width of the ENABLE1 pulse will be equal to the width the ENABLE2 pulse, and the control voltage reaches a balanced value that guarantees the delay of voltage-controlled buffer in chain 1 is K× that of chain 2. The DSCVG does not need to provide supplement current for V_2, V_3 or V_4, so power consumption is negligible. Compared with a fixed and proportional control voltage scheme, the DSCVG linearity-tuned PWMU performs better.

Figure 24.4.5 presents two effective methods to increase energy efficiency. (1) Filtered weights computation, where the original kernel is decomposed into a base kernel and a filtered kernel. The base kernel consists of all -1 or 1 values, while the filtered kernel is the difference between the original one and the base kernel: filtered ("/") means that the result doesn't need to be calculated. The weight pre-processing unit counts the number of -1 and 1 values in the original weight kernel, and then selects a suitable base kernel to filter it in order to reduce its complexity by ~2.3×. In a conventional base/filtered kernel scheme, the computing logic for the filtered part cannot be gated completely. Using the pulse width modulation scheme allows for the filtered parts in the filtered kernel to be easily bypassed. Nearly 50% of computing energy is saved. (2) Feature-aware computation, where the pulse is transmitted through different paths based on the feature. When the feature data is relatively large (F[7:4] = 010X, 0110, etc.), the LSB chain will be bypassed, which achieves computation-energy saving at the expense of a little bit of accuracy loss. Conversely, when the feature data is relatively small (F[7:4] = 0000), the MSB chain is bypassed, which also saves energy, but without accuracy loss. Comparing to a conventional scheme, the BWN is feature-aware and more flexible. After using the stochastic gradient Descent (SGD) algorithm to train with the ImageNet dataset, a 30%-45% reduction in energy is seen for different configurations, accuracy is reduced by less than 1%.

Sandwich-RAM is fabricated in a 28nm CMOS technology; the die photograph is shown in Fig. 24.4.7. Figure 24.4.6 shows the computing differential nonlinearity (DNL), which is the accumulation result for one row under different voltage and temperature corners. The DNL is less than 0.5·LSB, because DSCVG dynamically adjusts the control voltage and PQRC tracks global PT variation. The energy efficiency is improved by 4.16× based on filtered-weight computation and feature-aware computation. A 119.7TOPS/W peak-energy efficiency is achieved at 0.6V. When running AlexNet convolutional layers, Sandwich-RAM achieves an energy efficiency of 46.6TOPS/W.

Acknowledgements:
This work was supported by the National Natural Science Foundation of China (NO. 61474022) and the National Science and Technology Project under Grant 2018ZX01031101-005.
We would like to thank the TSMC University Shuttle Program for chip fabrication. We also thank Taomei Zhou, Mengyang Sun and Lizheng Ren for their help.

References:
[1] 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, 2018.
[2] D. Miyashita, et al., "A Neuromorphic Chip Optimized for Deep Learning and CMOS Technology with Time-Domain Analog and Digital Mixed-Signal Processing," *JSSC*, vol. 52, no. 10, pp. 2679-2689, Oct. 2017.
[3] R. Andri, et al., "YodaNN: An Architecture for Ultralow Power Binary-Weight CNN Acceleration," *IEEE TCAD*, vol. 37, no. 1, pp. 48-60, Jan. 2018.
[4] S. Yin, et al., "An ultra-high energy-efficient reconfigurable processor for deep neural networks with binary/ternary weights in 28nm CMOS," *VLSI*, pp. 37-38, 2018.
[5] W.-H. Chen, et al., "A 65nm 1MB Nonvolatile Computing-in-memory ReRAM Macro with Sub-16ns Multiply-and-Accumulate for Binary DNN AI Edge Processors," *ISSCC*, pp. 494-496, 2018.

 978-1-5386-8531-0/19/$31.00

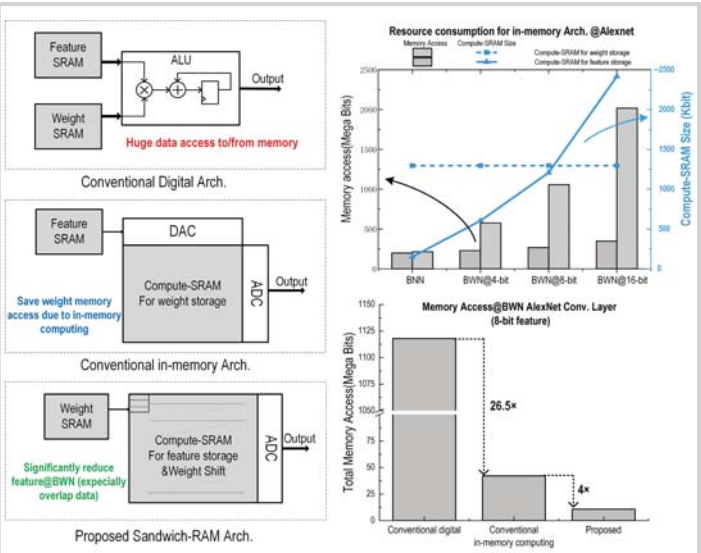
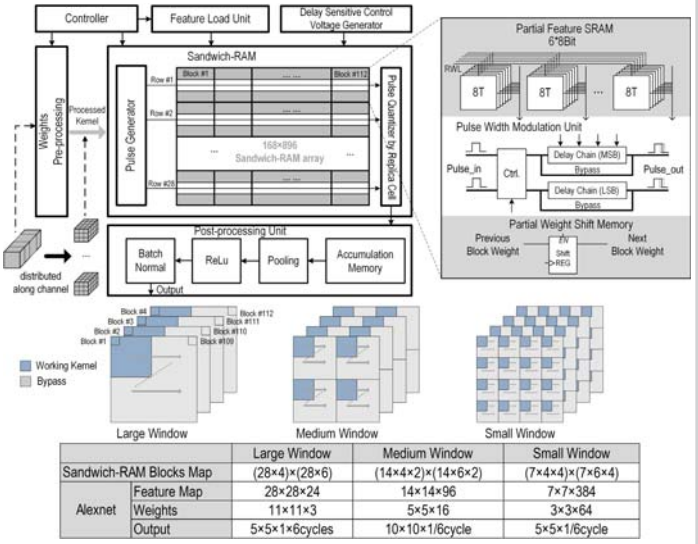


Figure 24.4.1: Proposed Sandwich-RAM architecture.



		Large Window	Medium Window	Small Window
Sandwich-RAM Blocks Map		(28×4)×(28×6)	(14×4×2)×(14×6×2)	(7×4×4)×(7×6×4)
Alexnet	Feature Map	28×28×24	14×14×96	7×7×384
	Weights	11×11×3	5×5×16	3×3×64
	Output	5×5×1×6cycles	10×10×1/6cycle	5×5×1/6cycle

Figure 24.4.2: Architecture, feature memory and CNN computation mode.

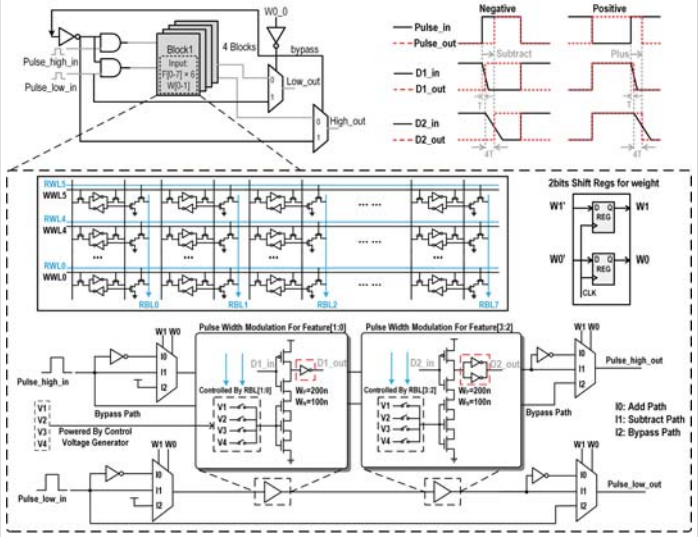


Figure 24.4.3: Processing block schematics including memory and PWMU.

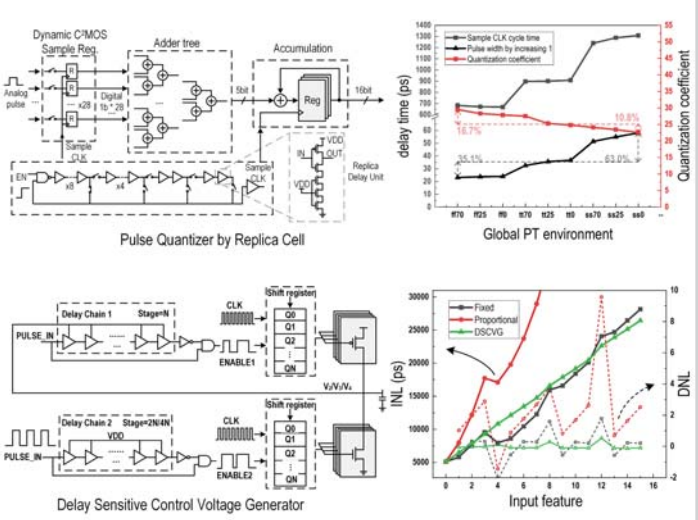
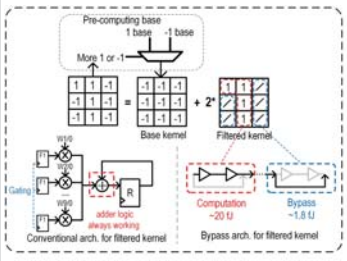
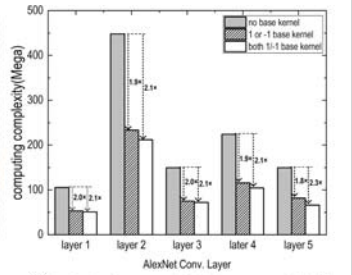
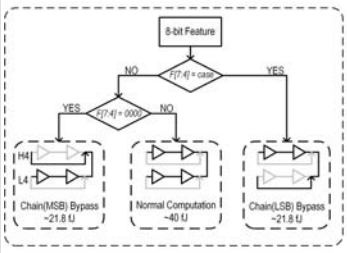


Figure 24.4.4: MAC accuracy improved by PQRC and DSCVG.







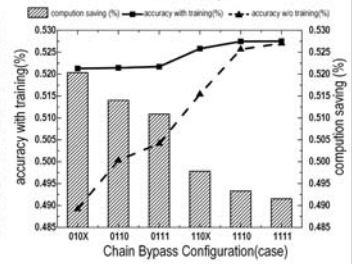
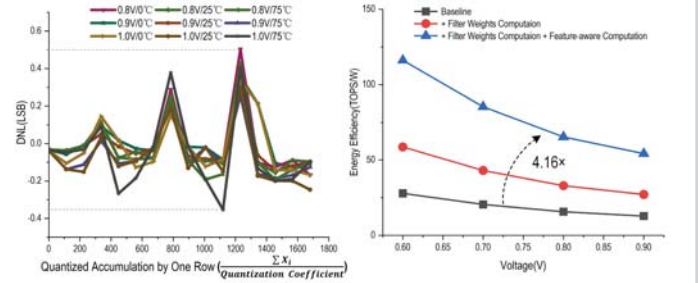


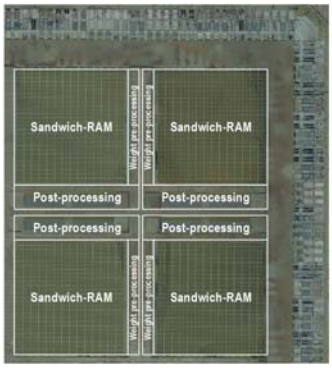
Figure 24.4.5: Complexity reduction by filtered-weight and feature-aware.



	Tech(nm)	Benchmark	Function	Computaion	Architecture	Energy Efficiency (TOPS/W)	
						Peak	Alexnet
Conv-RAM [1]	65	LeNet-5	BWN@6b	Analog(Voltage)	F ⊞	28.1	—
Miyashita [2]	65	LeNet-5	BNN	Analog(Time)	F ⊞	77	—
Yoda [3]	65	AlexNet Conv Layers	BWN@12b	Digital	F W +	—	2.2
YIN [4]	28	AlexNet Binary Weight	BWN@1~16b	Digital	F W +	95.8	19.9
This work	28	AlexNet Conv Layers	BWN@8b	Analog(Time)	⊞	119.7	46.6

*□means SRAM, +means MAC, ⊞means combine SRAM and MAC together.

Figure 24.4.6: Measurement and prior-art comparison table.



Technology	28nm
Supply voltage	0.6V ~ 0.9V
Frequncey	400MHz(max.)
Core size	0.424 × 0.51 mm^2
*Sandwich-RAM*4*	Feature:150Kb(28*28*4*6*8Bits) 8T Weight:6Kb(28*28*4*2)
Accumulation SRAM	2Kb
energy effiency (TOPS/W)	12.8~119.7

Figure 24.4.7: Die photograph and chip summary table.

978-1-5386-8531-0/19/$31.00 ©2019 IEEE