AN ENERGY-EFFICIENT MEMORY-BASED HIGH-THROUGHPUT VLSI ARCHITECTURE FOR CONVOLUTIONAL NETWORKS

Mingu Kang, Sujan K. Gonugondla, Min-Sun Keel, and Naresh R. Shanbhag

ABSTRACT

In this paper, an energy efficient, memory-intensive, and high throughput VLSI architecture is proposed for convolutional networks (C-Net) by employing compute memory (CM) [1], where computation is deeply embedded into the memory (SRAM). Behavioral models incorporating CM's circuit non-idealities and energy models in $45\,\mathrm{nm}$ SOI CMOS are presented. System-level simulations using these models demonstrate that the probability of handwritten digit recognition $P_r > 0.99$ can be achieved using the MNIST database [2], along with a $24.5\times$ reduced energy delay product, a $5.0\times$ reduced energy, and a $4.9\times$ higher throughput as compared to the conventional system.

Index Terms— Compute memory, Convolutional networks, Machine learning, Pattern recognition

1. INTRODUCTION

Emerging applications such as in health care, social networks and smart infrastructure leverage the ubiquitous presence of sensing and surveillance/monitoring based on statistical inference techniques. These emerging applications require a real time processing of massive data volumes in limited form factors. Energy efficiency is also important if such analysis is to be performed on battery powered systems such as biomedical and consumer electronics.

Convolutional networks (C-Net) is one of the most widely used pattern recognition algorithms due to its state-of-the-art performance in computer vision applications such as handwriting recognition and face detection [3, 4]. However, the C-Net requires complex interconnect, massive inner product computations, and access to a large data volume.

GPU [3] and FPGA-based [4] implementations were proposed recently in order to speed up C-Net computation over a purely software implementation. It is well-known that the energy and throughput of general purpose computing platforms such as GPU and FPGA are at least one to two orders-of-magnitude worse than dedicated VLSI implementations [5].

This work was supported by Systems on Nanoscale Information fabriCs (SONIC), one of the six SRC STARnet Centers, sponsored by SRC and DARPA.

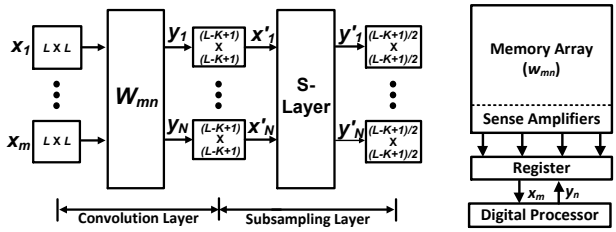

Fig. 1: Convolutional network (C-Net): (a) data flow, and (b) architecture.

This paper presents a dedicated VLSI architecture based on our previously proposed CM, where computation is embedded inside the memory array. Our CM-based C-Net implementation is shown to provide a $24.5\times$ reduced energy delay product as compared to the conventional system.

2. BACKGROUND

2.1. Convolutional Networks (C-Net)

A C-Net is a multi-layer network (see Fig. 1(a)) consisting of interleaved convolutional layers (C-layers) and subsampling layers (S-layers). The C-layer is computationally intensive and is described as follows:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \phi\left\{ \begin{bmatrix} w_{11} & \cdots & w_{1M} \\ \vdots & \cdots & \vdots \\ w_{N1} & \cdots & w_{NM} \end{bmatrix} * \begin{bmatrix} x_1 \\ \vdots \\ x_M \end{bmatrix} + \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix} \right\} \quad (1)$$

where $\boldsymbol{x_m}$ $(m = 1, ..., M)$ and $\boldsymbol{y_n}$ $(n = 1, ..., N)$ are the $L \times L$ input and $(L - K + 1) \times (L - K + 1)$ output feature maps, respectively, $\boldsymbol{w_{mn}}$ is a $K \times K$ kernel function, $*$ is a convolutional operator, and b_n is a bias term. Here, ϕ is a non-linear, typically sigmoid, activation function. The subsampling layer (S-layer) simply reduces the dimensions of input feature map $\boldsymbol{x'_n}$. As indicated in (1), large data volumes need to be processed by the C-Net. Hence, a memory-based architecture, as proposed in this paper, can be highly effective in implementing C-Nets.

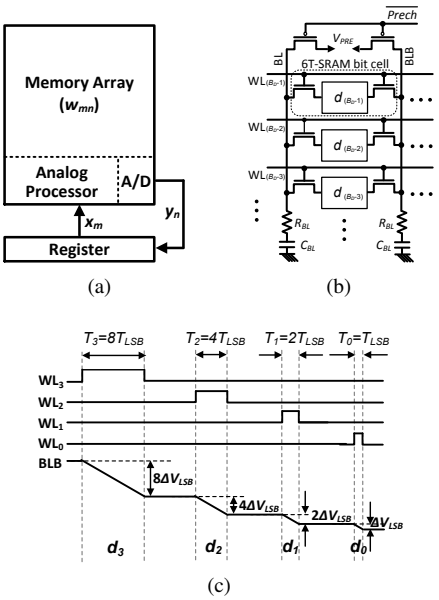

Fig. 2: Compute memory [1]: (a) architecture, (b) column structure of bit cell array (c) and the MR-READ waveform for 4-bit word read-out, D = 1111b' [1].

Figure 1(b) shows the block diagram of a conventional C-Net system [4], where a conventional SRAM stores the weights $\boldsymbol{w_{mn}}$, and the input feature map $\boldsymbol{x_m}$ is stored in a register bank. The register contents are updated with the output feature map $\boldsymbol{y_n}$ at the completion of one layer.

The energy consumption to process a single feature map in a conventional system can be expressed as

$$E_{conv} = K^2 E_{read} + E_{leak} + (L-K+1)^2 K^2 E_{MAC} + E_{reg} \quad (2)$$

where E_{read} and $E_{leak} = P_{leak} T_{conv}$ represent the single word SRAM read energy and the SRAM leakage energy per feature map computation, respectively. Here, P_{leak} is the leakage power consumption and T_{conv} is the time needed to generate a feature map. It is assumed that a deep-sleep mode is enabled during standby using techniques such as power gating or lowering the supply voltage for bit cell array [6]. E_{MAC} and E_{reg} are the multiplier and accumulator (MAC) and register bank energies, respectively.

2.2. Compute Memory (CM)

Recently, the compute memory (CM) [5] was proposed for memory-based implementations of inference kernels. The CM employs multi-row READ (MR-READ) access, which is a form of digital-to-analog conversion, followed by bit line analog signal processing (BL-ASP) stage to process data as shown in Fig. 2(a). Thus, the CM eliminates the memory-processor interface present in conventional systems thereby achieving high throughput. Both the MR-READ and BL-ASP are low-swing/low-SNR operations which, along with its high throughput, contributes to the CM's intrinsic energy efficiency. The low-SNR nature of CM operations makes it eminently suited for inference applications where output quality is measured in terms of statistical metrics, e.g., probability of detection. The CM preserves the read/write functionality and the storage density of a standard SRAM.

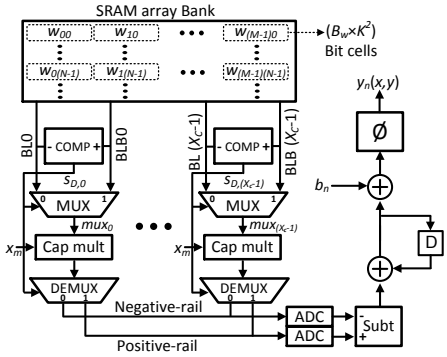

Fig. 3: Compute memory based architecture for C-Net.

In the CM, the MR-READ process is the key to its functionality. Let data D stored in a memory be represented by a B_D-bit vector $\mathbf{d} = \{d_0, d_1, ..., d_{B_D-1}\}$ as shown in Fig. 2(b). Then, $D = \Sigma_{k=0}^{B_D-1} 2^k d_k$ is the decimal value of D, where k is the bit position. In the MR-READ operation [1]: (a) $\mathbf{d}$ is stored in one column, (b) B_D word lines (WLs) are activated per precharge, and (c) the WL access pulse width $T_k \propto 2^k$ ($k \in [0, B_D - 1]$), i.e., the access pulses are pulse-width modulated. By ensuring that $T_k \ll RC$, where RC is the time constant of the bit lines BL and BLB, the BLB voltage drop at the end of the MR-READ process is given by [1]:

$$\Delta V_{BLB}(D) = \frac{V_{PRE}}{R_{BL} C_{BL}} T_{LSB} \sum_{k=0}^{B_D-1} 2^k d_k \quad (3)$$

where T_{LSB} is the LSB pulse width. Figure 2(c) shows how a 4-bit word, 1111b' is read out on BLB via binary-weighted WL pulse widths T_0 to T_3. The 8-bit word is implemented by sub-ranging into two 4-bit words separately and merging via weighted charge sharing. The MR-READ process reduces the number of precharge operations which contributes to its energy efficiency.

3. PROPOSED CM-BASED C-NET SYSTEM

3.1. The CM-based C-Net Architecture

Figure 3 shows CM-based C-Net architecture, where X_c is the number of columns in the SRAM array. The K^2 coefficients of $\boldsymbol{w_{mn}}$ are stored in a block of $B_w \times K^2$ bit cells, where B_w is the bit precision of $\boldsymbol{w_{mn}}$, and each B_w-bit word is stored in one column. In addition, $\boldsymbol{w_{mn}}$ with the same

value of n are horizontally aligned in a single row occupying NK^2 columns. If $NK^2 > X_c$, the $\boldsymbol{w_{mn}}$ are stored in $\lceil NK^2/X_c \rceil$ rows. The $\boldsymbol{w_{mn}}$s required to compute a single pixel of $y_n(x,y)$ are MR-READ and multiplied with $\boldsymbol{x_m}$ provided in the digital domain from feature map registers.

In the following, we employ D and P to represent specific values of $\boldsymbol{w_{mn}}$ and $\boldsymbol{x_m}$, respectively, in order to simplify the exposition. In the CM, negative values are difficult to represent in analog domain. An 1's complement representation is employed for D and an unsigned representation for P. Thus, $|D| \times P$ and $S_D = sign(D)$ is computed separately. Here, $|D|$ is computed as follows:

$$|D| = \begin{cases} \Sigma_{k=0}^{B_D-1} 2^k d_k \propto \Delta V_{BLB}(D), & if\ D \geq 0 \\ \Sigma_{k=0}^{B_D-1} 2^k \overline{d}_k \propto \Delta V_{BL}(D), & if\ D < 0 \end{cases} \quad (4)$$

The S_D is obtained by using a differential amplifier with $\Delta V_{BL}(D)$ and $\Delta V_{BLB}(D)$ as its inputs, which is then used as a select signal of the multiplexer to select the greater of $V_{BL}(D)$ and $V_{BLB}(D)$ thereby generating $|D|$ as the output V_{mux} as shown in Fig. 3.

Next, the outputs of multipliers are transferred to a positive or negative rail via demultiplexers based on the S_D. The rails are shared with multiple columns so that the absolute values of positive and negative products in (1) are added separately via charge-sharing on each rail. Finally, each value on the rail is converted into a digital number through two analog-to-digital converters (ADCs), whose outputs are subtracted to generate the a convolution sum. These steps are repeated $\lceil NK^2/X_c \rceil$ times if $NK^2 > X_c$ to fetch all the required $\boldsymbol{w_{mn}}$s. Then, the sequentially generated outputs of the subtractor and the b_n are accumulated. The activation function is implemented with three additions and two shifts in the digital domain as introduced in [4].

3.2. Inner Products via BL-ASP

The product of $\Delta V_{BLB}(D)$ (or $\Delta V_{BL}(D)$) from the multiplexer and a B_P-bit digital value P is obtained via the capacitive multiplier shown in Fig. 4(a). The multiplier output ΔV_m from V_{PRE} is (see Fig. 4(b)):

$$\Delta V_m = (0.5)^{B_P} P \Delta V_{BLB}(D) = \alpha P D \quad (5)$$

where α is a constant depending on T_{LSB}, C_{BL}, and R_{BL}.

The voltage level $V_{PRE} - \alpha \sum_{j=0}^{K^2-1} D_j P_j$ corresponding to the inner product between $\overrightarrow{D} = D_0, D_1, ..., D_{K^2-1}$ and $\overrightarrow{P} = P_0, P_1, ..., P_{K^2-1}$ can be achieved by charge-sharing the multipliers' outputs in K^2 columns.

3.3. Energy and Behavioral Models with Circuit Non-idealities

The analog-intensive CM operation is subject to a number of circuit-level non-idealities. Dominant among these are: (a) non-linearity of the multi-row READ process. This is caused by voltage-dependent discharge path resistance, R.

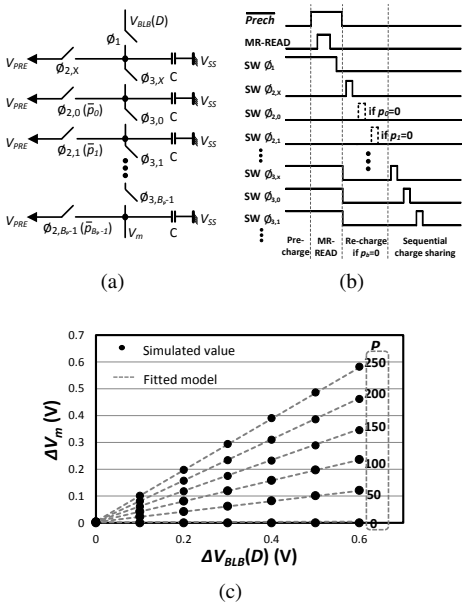

Fig. 4: Capacitive multiplier: (a) structure, (b) timing diagram of control signals, and (c) validation of (5) with circuit simulation in 45 nm ($C = 10fF$).

(b) local transistor threshold voltage V_t-mismatch across bit cells caused by random dopant fluctuations. (c) non-ideality of analog multiplication.

The non-linearity of MR-READ was previously modeled in [1] by a polynomial fit as $\Delta V'_{BLB}(D) = \sum_{k=0}^{4} c_k D^k$, where $\Delta V'_{BLB}(D)$ is a distorted version of $\Delta V_{BLB}(D)$, and c_ks are the fitting parameters. In this paper, we model the V_t-mismatch and non-ideality of the analog multiplier.

The impact of V_t-mismatch is modeled as a Gaussian distributed random variable as shown below:

$$\Delta \widehat{V}_{BLB}(D) \sim N(\Delta V'_{BLB}(D), \sigma_D^2) \quad (6)$$

where σ_D^2 is the variance of ΔV_{BLB} due to V_t-mismatch across bit cells corresponding to the stored value D.

The behavior of multiplier can be captured by a polynomial model with fitting parameters $f_{0,1,2,3}$ as follows:

$$\Delta V_m = f_0 \Delta V_{BLB}(D) P + f_1 \Delta V_{BLB}(D) + f_2 P + f_3 \quad (7)$$

These models are employed in the Section 3 to study the impact of circuit non-idealities on application level behavior.

The energy consumption of CM to process single feature map is given by

$$\begin{aligned} E_{CM} \quad = \quad & (L-K+1)^2 K^2 E_{MR\_read} + E_{leak\_CM} \\ & + (L-K+1)^2 K^2 E_{MAC\_add} + E_{reg} \end{aligned} \quad (8)$$

where E_{MR_read} is the energy consumed to read single word by the MR-READ. The scaling factor of the first term in (8) is larger than that of (2). This is because the CM reads the

Parameter	Values	Parameter	Values
V_{DD}	$1.1\,V$	f	$1\,GHz$
input L	32	K	5
B_w	8	B_x	6
N	$C1:6\,, C3:16, F5:120\,, F6:10$		
f_0,..., f_4	$1, 1.11\times10^{-2}, -5.4684\times10^{-4}, 4.0506\times10^{-6}$		
c_0,..., c_4	$-9.5\times10^{-3}, 3.2\times10^{-2}, 3.5\times10^{-4}, -1.7\times10^{-5}, 1.3\times10^{-7}$		

Table 1: Design and model parameters of compute memory.

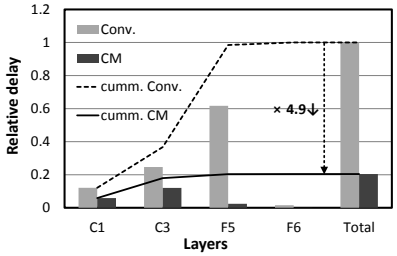

Fig. 5: Estimated relative delays for C-Net.

$\boldsymbol{w_{mn}}$s from SRAM again whenever the processing window slides as the analog level from MR-READ cannot be sustained. The leakage energy $E_{leak\_CM} = P_{leak}T_{CM}$, where T_{CM} is the processing time of single feature map by the CM, and smaller than T_{conv}. Thus, E_{leak_CM} is also smaller than E_{leak}. The E_{MAC_add} is the energy consumed for the analog multiplier and charge sharing based adder. This is also smaller than the E_{MAC} due to the operation with small voltage swing.

4. SIMULATION RESULTS

In this section, a hand written digit recognition with MNIST database [2] is chosen as an application to measure the performance of CM system. All the design and model parameters are summarized in Table I. The variant of LeNet5 [3] is employed including total six layers.

Horizontally aligned four banks of SRAM array with a size of 512×256 bit cells are employed for the CM to store trained kernel $\boldsymbol{w_{mn}}$s. Thus, roughly 40 $\boldsymbol{w_{mn}}$s can be aligned in one row and processed at a time ($X_c(=256\times4)/K^2 \approx 40$). The embedded SRAM's IO is 32 bits in the conventional system. The number of multipliers is $K^2 = 25$ in the conventional system to achieve area comparable to the CM.

4.1. Model Validation

HSPICE simulations are performed in 45 nm SOI process technology to obtain the behavior models. The normalized standard deviation (σ_D/μ_D) of ΔV_{BLB} in the model (6) is measured by Monte-Carlo HSPICE simulations. The minimum value 7% is achieved with $D = 15$, and a maximum value 12.5% is obtained with $D = 1$. The simulated and modeled behavior of capacitive multiplier with $C = 10fF$ is described in Fig. 4 with fitting parameters $f_{0,1,2,3}$ in Table I

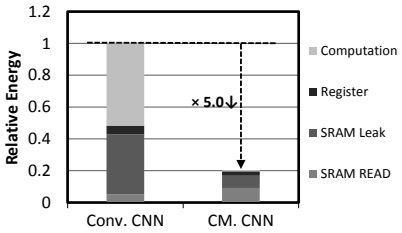

Fig. 6: Estimated relative energy consumptions for C-Net.

4.2. Recognition Accuracy

The $\boldsymbol{w_{mn}}$ are obtained with 60000 training data set of MNIST [2] with back propagation algorithm through roughly 80 iterations. The error rates are measured on MNIST test data set by the system simulations with the behavior models in following configurations: 1) conventional system with a floating point numbers, 2) with fixed point (B_w and B_x), 3) CM system with fixed point and 4) with fixed point numbers and $\boldsymbol{w_{mn}}$ trained reflecting non-linearity of MR-READ. Error rates 0.8% and 0.85% are achieved in the first and second configurations, respectively. In the third configuration, the error rate is degraded to 1.36% due to circuit non-idealities. However, the error rates is 0.87% in the fourth configuration. It indicates that the non-idealities from CM is effectively compensated by the inherent error resiliency of C-Net.

4.3. Energy and Delay savings

The SRAM access and multiplication of conventional system require two cycles and one cycle of clock, respectively, and those can be pipelined. On the other hand, the MR-READ and ME-ASP of CM require total 20 cycles, where the CM processes $X_c = 1024$ words reading and multiplications in parallel. Based on the previous specifications, each delay from the convolutional and fully connected layers and cumulative delays are estimated in Fig. 5. The CM achieves roughly 4.9× reduced total delay especially achieving higher throughput in the memory intensive F5 layer due to the parallel read and computations.

Based on the energy consumption to process single feature map modeled in (2) and (8), the energy consumptions to process all the feature maps of layers are estimated in Fig. 6. About 5.0× energy saving in overall system is achieved mostly by the low power inner product computation and the reduced leakage energy due to high throughput.

In conclusion, 24.5× smaller energy delay product is achieved by the CM as compared to the conventional system with 0.02% larger error rate.

5. REFERENCES

[1] M. Kang, M.-S. Keel, N. R. Shanbhag, S. Eilert, and K. Curewitz, "An energy-efficient VLSI architecture for pattern recognition via deep embedding of computation in sram," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 8326–8330.

[2] Y. LeCun and C. Cortes, "MNIST handwritten digit database," *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2010.

[3] D. Strigl, K. Kofler, and S. Podlipnig, "Performance and scalability of GPU-based convolutional neural networks," in *IEEE Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, February 2010, pp. 317–324.

[4] C. Farabet, C. Poulet, J. Y. Han, and Y. LeCun, "CNP: An FPGA-based processor for convolutional networks," in *IEEE International Conference on Field Programmable Logic and Applications (FPL)*, August 2009, pp. 32–37.

[5] M. Horowitz, "Computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. (ISSCC)*, February 2014, pp. 10–14.

[6] M. Yamaoka, Y. Shinozaki, N. Maeda, Y. Shimazaki, K. Kato, S. Shimada, K. Yanagisawa, and K. Osada, "A 300-mhz 25-μa/mb-leakage on-chip SRAM module featuring process-variation immunity and low-leakage-active mode for mobile-phone application processor," *IEEE J. Solid-State Circuits*, vol. 40, no. 1, pp. 186–194, 2005.