Extended Abstracts of the 2020 International Conference on Solid State Devices and Materials, VIRTUAL conference, 2020, pp91-92

Design of an Energy-Efficient Binarized Convolutional Neural Network Accelerator Using a Nonvolatile FPGA with Only-Once-Write Shifting

Daisuke Suzuki[1], Takahiro Oka[2], and Takahiro Hanyu[2]

[1] The University of Aizu, 90 Kami-iawase, Tsuruga, Ikki-machi, Aizu-Wakamatsu, Fukushima 965-8580, Japan
Phone: +81-0242-37-2667 E-mail: daisuke@u-aizu.ac.jp
[2] Tohoku University, 2-1-1, Katahira, Aoba-ku, Sendai, Miyagi 980-8577, Japan

Abstract

This paper presents an energy-efficient hardware accelerator for binarized convolutional neural networks (BCNNs). A magnetic-tunnel-junction (MTJ)-based nonvolatile field-programmable gate array, where the number of stored-data updating is minimized in configurable logic block, acts as an important component to save energy consumption in BCNN with maintaining high-speed shifting. It is demonstrated under 55nm CMOS/MTJ process technologies that the power consumption of the proposed hardware is 8.7x lower than that of a BCNN hardware without energy-efficient data shifting.

1. Introduction

The internet of things (IoT) has become the post-cloud era and edge computing, where data is collected and processed locally at each edge device, is an emerging technology. In such edge computing, a hardware accelerator for artificial intelligence (AI) plays a significant role [1]. A field-programmable gate array (FPGA) is a promising hardware platform owing to its reconfigurable and fully parallel architecture [2] and the use of a binarized convolutional neural network (BCNN) where network parameters are expressed in binary format is an effective approach to implementing an AI accelerator on the FPGA [3, 4]. However, since in a conventional SRAM-based FPGA, storage elements are volatile, the power supply must be continuously applied during the operation to keep the stored information, which causes a large amount of standby power consumption. As a result, to minimize the number of idle components is required for the SRAM-based BCNN accelerator design.

In order to essentially solve the above standby-power problem, it is important to design a nonvolatile FPGA (NV-FPGA) [5] based low-power BCNN accelerator with energy-efficient data-transfer scheme. Since all the data are stored into nonvolatile devices, a power-gating technique can be fully utilized and standby power consumption of idle processing elements (PEs) are eliminated. This standby power reduction mechanism is quite suitable for massively parallel architecture where a large number of PEs must be prepared. Moreover, the use of only-once-write shifting in lookup table (LUT) circuit [6, 7] makes it possible to perform energy efficient data transferring in the BCNN accelerator. As a typical design example, a PE is designed in 55nm CMOS/MTJ process technologies.

2. NV-FPGA-Based BCNN Accelerator

A typical BCNN model contains convolutional, pooling, and fully-connected layers as shown Fig. 1 (a). The first few layers usually capture regional information such as edges and curves, and the last few layers interpret these low-level features into high-level abstractions with the posterior probability assigned for classification. Figure 1 (b) shows the pseudo code of a convolutional layer whose operation is composed by multiplications and additions. In the BCNN, since all the data are expressed by 0 or 1, the convolutional operation is replaced by XNOR and bit-count operations [3]. Figure 2 shows the overall architecture of the NV-FPGA-based BCNN accelerator with N PEs. The convolutional operation is performed by several PEs and the number of PEs used in the convolutional function depends on which layer is calculated; thus, some PEs are idle during at each convolutional operation. The use of NV-FPGA makes it possible to reduce the standby power of such idle PEs.

Figure 3 shows a block diagram of the proposed PE which is composed of multiply-accumulate (MAC) units, input buffers (IBUFs), and weight buffers (WBUFs). Input feature maps and weights are serially sent to MAC units via IBUFs and WBUFs. Since these data are binary format, the MAC unit is simply implemented by an XNOR gate and an accumulator. Since the input feature map and the weight are long data stream, data-shift function performs an important role in the BCNN operation. Thus, it is very important to consider how to implement shift register in the BCNN accelerator.

Figure 4 (a) shows a typical shift register function in 2-input LUT circuit where each storage element (SE) is connected to its neighbors and the shift operation is directly performed by propagating data to the next SE. In this case, all the four SE are always active during the operation, which results in high write power consumption. Figure 4 (b) shows the only-once-write shifting in the 2-input LUT circuit where read/write address is updated at each cycle [6, 7]. The shift operation is performed by serially reading and updating the content of corresponding SE at each cycle which makes it possible to minimize the number of write access per cycle to one. This shifting method can be compactly implemented in the LUT circuit [6, 7] and WBUF and IBUF are implement by using the proposed LUT circuits as shown in Fig. 5 and Fig. 6 respectively.

3. Evaluations

Table 1 shows the comparison of the power consumption during computing the 1st layer of the BCNN for MNIST image recognition task. Since 6-input LUT circuits are used, each LUT circuits update 64 SEs at each 1-bit shift operation. In contrast, by using only-once-write shifting, the number of write-access for BCNN operation is reduced to 1/64 compared to that of conventional method which results in 8.7x lower power consumption.

4. Conclusions

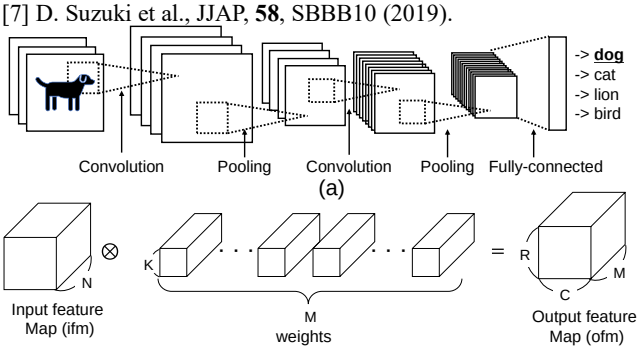
An energy-efficient hardware accelerator for BCNN has been presented using only-once-write shifting and its potential for low power operation is demonstrated. For the next step, it is very important to explore detailed design space and to evaluate quantitively in overall BCNN accelerator.

Acknowledgements

This research is supported by JST-CREST(JPMJCR19K3), JST-OPERA, JSPS KAKENHI Grant No.JP16H06300, CIES cons. program, and VDEC.

References

[1] W. Sun et al., IEEE Network, **33**, 68 (2019).
[2] C. Hao et al., Proc. DAC, 2019, p. 1.
[3] M. Courbariaux et al., arXiv:1602.02830, 2016.
[4] Y. Li et al., AC JETC, **14**, 18:1 (2018).
[5] D. Suzuki et al., VLSI Circuits Dig. Tech. Pap., 2015, p. 172.
[6] D. Suzuki et al., Microelectronics Journal, **83**, 39 (2019).
[7] D. Suzuki et al., JJAP, **58**, SBBB10 (2019).




```
for (to=0; to<M; to++)
  for (row=0; row<R; row++)
    for (col=0; col<C; col++)
      for (ti=0; ti<N; ti++)
        for (i=0; j<K; i++)
          for (j=0; j<K; j++)
            ofm[to][row][col] += weight[to][ti][row+i][col+j] * ifm[ti][row+i][col+j];
```

(b)

Fig. 1. Overview of BCNN: (a) network structure, (b) convolutional layer and its pseudo code.

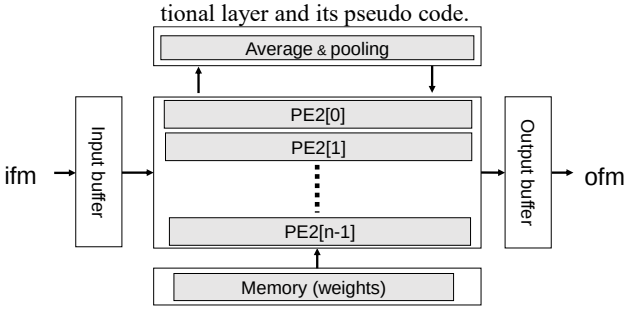


Fig. 2. Overall architecture of the NV-FPGA-based BCNN accelerator. Since all the data are nonvolatile, power supply of the idle PEs is turned off and standby consumption is reduced.

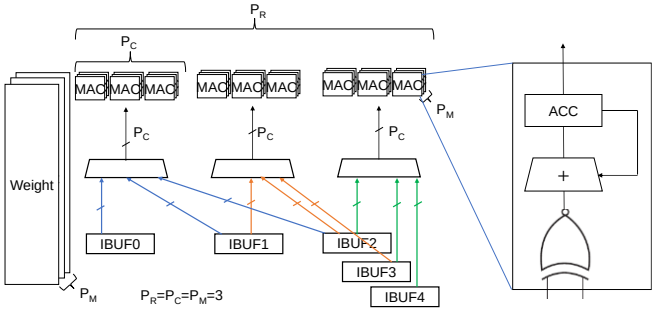


Fig. 3. Block diagram of the proposed PE.

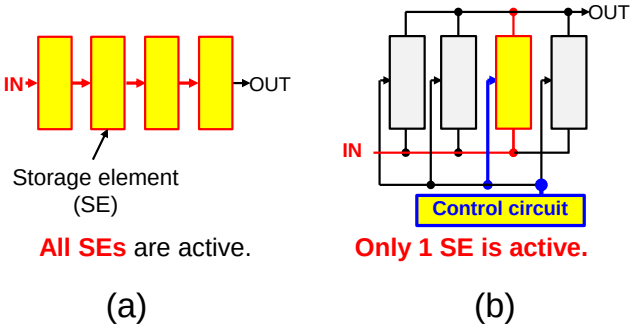


Fig. 4. Shift operation in 2-input LUT circuit: (a) conventional, and (b) proposed.

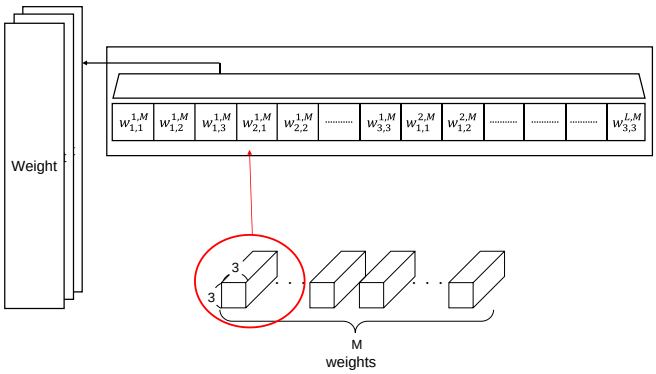


Fig. 5. Block diagram of weight buffer (WBUF).

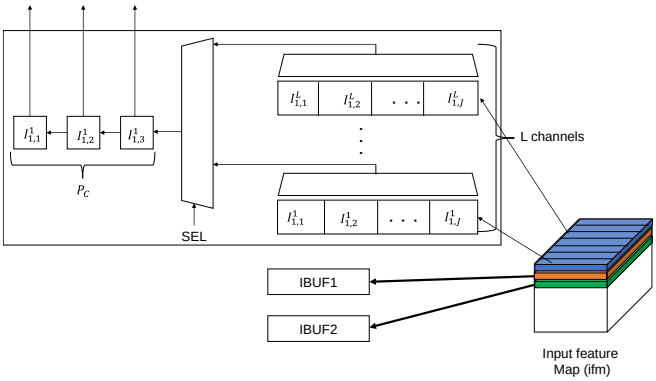


Fig. 6. Block diagram of input buffer (IBUF).

Table. 1. Comparison of power consumption.

		NV-FPGA (Conventional*2)		NV-FPGA (proposed)
# of shift operations (*1)	IBUF	51840	1/64	810
	WBUF	107,520	1/64	1680
Power consumption in shift operation (*3)		2.00	1/8.7	0.23

(1) The 1st layer of the BCNN for MNIST image recognition task
-> R=C=24, M=10, N=1, K=3 (see Fig. 1(b))
-> $P_R = P_C = 8$, $P_M = 5$ (see Fig. 3)

(2) 64 SEs are updated at each 1-bit shift operation.

(3) 55nm CMOS/MTJ technologies with 250MHz frequency