A FAST INNER PRODUCT PROCESSOR IMPLEMENTATION FOR MULTI-VALUED EXPONENTIAL BIDIRECIONAL ASSOCIATIVE MEMORIES*

Chua-Chin Wang†, *Yih-Long Tseng*, *Ying-Pei Chen*‡

Department of Electrical Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan 80424
Tel : 886-7-525-2000 ext. 4144
Fax : 886-7-5254199
E-Mail: ccwang@ee.nsysu.edu.tw

Chenn-Jung Huang

Department of Computer Science
and Information Education
National Taitung Teachers College
Taitung, Taiwan 95004
E-Mail: cjh@cc.ntttc.edu.tw

ABSTRACT

Inner product calculations are often required in digital neural computing. The critical path of the inner product of two vectors is the carry propagation delay generated from individual product terms. In this work, a novel and high-speed realization of inner product processor for the MV-eBAM [1], [2] is presented in order to reduce the carry propagation delay, wherein the treatment of inner product of two vectors is given. The architecture we propose here might offer a sub-optimal solution for the digital hardware realization of the inner product computation.

1. INTRODUCTION

The systolic-like architecture of partial product reduction tree for the parallel multiplier introduced by Wallace [3] motivated the implementation of several parallel schemes for the inner production calculation. However, Oklobdzija et al., [4], [5] pointed out that it is the interconnection of the compressors rather than the structure of the compressors that leads to the fastest realization of partial product reduction in multiplication operation. The novel inner product processor dedicated to the MV-eBAM presented in this paper will include a systolic-like architecture of compressor unit wherein the arrangement of the 3-2 compressors in order such that the carry propagation delay in the critical path is reduced. In addition to the compressor unit, an inner product term generator is also proposed to produce the individual inner product terms as the inputs to the compressor unit.

2. HIGH-SPEED INNER PRODUCT PROCESSOR FOR OF MV-EBAM*

In order to reduce the carry propagation delay produced in the implementation of the MV-eBAM, it is demanding to develop a special-purpose processor for the inner product of two multi-valued operands. The entire design of multi-valued inner product processor is divided into two parts, which are an individual inner product term generator, and a compressor unit. Fig. 1 shows the dataflow of a multi-valued inner product calculation.

*This research was partially supported by Nation Science Council under grant NSC 88-2219-E-110-001

† Contact author

‡ Ms. Chen is currently an ASIC design engineer of VIA Inc., Taiwan.

2.1. Theory of MV-eBAM

Suppose we are given M pattern pairs, which are $\{(X_1, Y_1), (X_2, Y_2), \cdots, (X_M, Y_M),\}$ where $X_i = (x_{i1}, x_{i2}, \cdots, x_{in})$, where $Y_i = (y_{i1}, y_{i2}, \cdots, y_{ip})$, where n is assumed to be smaller than or equal to p without any loss of generality. Hence, the evolution equations of the MV-eBAM are shown as

$$\begin{aligned} y_k &= H\left(\frac{\sum_{i=1}^{M} y_{ik} b^{-||X-X_i||^2}}{\sum_{i=1}^{M} b^{-||X-X_i||^2}}\right), \\ x_k &= H\left(\frac{\sum_{i=1}^{M} x_{ik} b^{-||Y-Y_i||^2}}{\sum_{i=1}^{M} b^{-||Y-Y_i||^2}}\right) \end{aligned} \quad (1)$$

where X and Y are input key patterns, b is a positive number, called the radix, $b > 1$, x_k and x_{ik} are kth digits of X and X_i with y_k and y_{ik} for Y and Y_i, respectively, and $H(\cdot)$ is a staircase function shown as the following equation,

$$H(x) = \begin{cases} l, & (l-0.5)\cdot\frac{D}{L} \le x < (l+0.5)\cdot\frac{D}{L} \\ 1, & x < 1.5\cdot\frac{D}{L} \\ L, & x \ge D \end{cases} \quad (2)$$

where $l = 1, 2, \cdots, L$ is the number of finite levels, and D is the finite interval of the staircase function. Note that if $D \to \infty$, and $L \to \infty$, then $H(x) \approx x, for\ x > 0$. The reason why the staircase function is used is the argument in $H(\cdot)$ in Eqn. (2) is not necessarily a positive integer. We, hence, have to assign this argument to a nearest integer.

2.2. Inner product term generator

Considering the compatibility with the binary digital system, the number of finite levels, L, in Eqn. (2) is set to $2^w - 1$ in the implementation of the inner product processor for the MV-eBAM. Each product term in Eqn. (1), *product*, can be evaluated by

$$\begin{aligned} product &= \vec{A}\cdot\vec{B} = \left(\sum_{i=0}^{w-1} A_i\cdot 2^i\right)\cdot\left(\sum_{i=0}^{w-1} B_i\cdot 2^i\right) \\ &= A_{w-1}B_{w-1}\cdot 2^{2w-2} \\ &+ (A_{w-2}B_{w-1} + A_{w-1}B_{w-2})\cdot 2^{2w-3} + A \\ &+ (A_0B_2 + A_1B_1 + A_2B_0)\cdot 2^2 \\ &+ (A_0B_1 + A_1B_0)\cdot 2^1 + A_0B_0\cdot 2^0 \end{aligned} \quad (3)$$

where A_i, B_i is 0 or 1. Notably, the design of the inner production term generation becomes simple because only w^2 AND gates are

required to produce the w^2 partial products in Eqn. (3). Fig. 2 shows the configuration of the inner product term generation unit. Note that the dimension of the stored patterns is set to the count of the inputs to a $(2^m-1)-to-m$ compressor, which will be introduced in the next section. Therefore, length of the inputs to the inner production term generator is $(2^m-1)\cdot 2w$, and the length of the outputs is $(2^m-1)\cdot w^2$ according to Eqn. (3). In case that the dimension of the stored patterns is less than 2^m-1, all the unused inputs to the inner production term generator are padded with $0's$.

2.3. Framework of the compressor unit

2.3.1. Systolic-like $(2^q-1)-to-q$ compressor building block

A 3-2 compressor is basically a full adder. The equations of a full adder are shown as follows:

$$\begin{aligned} S &= (\alpha\oplus\gamma)\cdot\beta'+(\alpha\oplus\gamma)'\cdot\beta &&= F\cdot\beta'+F'\cdot\beta \\ C &= (\alpha\oplus\gamma)\cdot\beta+(\alpha\oplus\gamma)'\cdot\beta &&= F\cdot\beta+F'\cdot\gamma \end{aligned} \tag{4}$$

where F denotes $(\alpha\oplus\gamma)$. As shown in Fig. 3, the logic structure of a typical 3-2 compressor can be split into two logic layers. One of the three inputs, $\beta(\beta')$, is not required in the first logic layer. A $(2^q-1)-to-q$ compressor building block can be constructed by cascading 3-2 compressors. This architecture, inspired by the design methodology of systolic arrays, consists parallelized 3-2 compressor building blocks only at every processing stage.

To compute the total count of 3-2 compressors used in a $(2^q-1)-to-q$ compressor, we consider an alternative architecture of the $(2^q-1)-to-q$ compressor, which is composed of two $(2^{q-1}-1)-to-(q-1)$ compressors and $(q-1)$ 3-2 compressors, as shown in Fig. 5. We can derive the count of the 3-2 compressors used in this architecture as follows:

$$N_2=1, N_3=4,\ldots,N_q=2\cdot N_{q-1}+q-1, q>2 \tag{5}$$

where N_q denotes the number of the 3-2 compressors used in a $(2^q-1)-to-q$ compressor. By solving the above recurrence relation, we obtain

$$N_q=2^q-q-1 \tag{6}$$

The number of 3-2 compressors used in these two architectures we present above is identical because no unused inputs to the 3-2 compressors appear in both $(2^q-1)-to-q$ compressor structures. Thus, we can conclude that the count of 3-2 compressors used in the systolic-like architecture of the $(2^q-1)-to-q$ compressor is also 2^q-q-1.

2.3.2. Framework of digital compressor design

According to Eqn. (3), the summation of the partial product terms is not computed in the inner product term generator. This implies that the outputs of the w^2 AND gates are fed into the compressor unit at the required bit positions. Besides, 2^m-1 individual inner product terms need to be accumulated at each bit position. Thus, there will be 2^m-1 partial product terms at LSB, $2\cdot(2^m-1)$ partial product terms at the second bit position, $w\cdot(2^m-1)$ partial product terms at the wth bit position, and 2^m-1 partial product terms at the $(2w-1)$th bit position (MSB), and so forth, as shown in Fig. 2. Since many accumulation operations must be performed to obtain the final result, the improvement of the carry propagation delay of the critical paths is the major consideration for the architecture of the compressor unit. The entire architecture we propose to achieve this goal is shown in Fig. 6. Since this compressor unit is composed of one or several$(2^m-1)-to-m$ compressors at each bit position, we tend to set the dimension of the stored patterns to 2^m-1 to reduce the number of unused inputs to the basic 3-2 compressor building blocks.

Although it is difficult to derive a general form of the critical delay for the compressor unit due to its irregular structure, the estimated delay can be derived by attaching fictitious 3-2 compressor of $(2w-2)$ stages on the top of the compressor unit to form a single $(2^q-1)-to-q$ compressor tree. Since there are $(2^w-1)\cdot w^2$ inputs to the compressor unit, the length of the inputs to the made-up $(2^q-1)-to-q$ compressor tree now becomes $(2^m-1)\cdot w^2\cdot(\frac{3}{2})^{2w-2}$ after tracing back to the top level of the $(2^q-1)-to-q$ compressor. We assume $D_{m,w}$ denotes the count of 3-2 compressors in the critical path of the compressor unit, then the delay can be derived as follows:

$$\begin{aligned} D_{m,w} &\approx \left\lceil \frac{\log\frac{(2^m-1)\cdot w^2\cdot(\frac{3}{2})^{2w-2}}{2}}{\log\frac{3}{2}} \right\rceil-(2w-2) \\ &< \left\lceil (m-1)\cdot\frac{\log 2}{\log\frac{3}{2}}+\frac{2\log w}{\log\frac{3}{2}}+(2w-2)\right\rceil-(2w-2) \\ &< 2m+11.36\log w-1.71 \end{aligned} \tag{7}$$

The number of 3-2 compressors used in the compressor unit can be estimated based on Eqn. (6). Let N_q denotes the number of the 3-2 compressors used in the made-up $(2^q-1)-to-q$ compressor tree. Firstly, we get

$$2^q-1 = (2^m-1)\cdot w^2\cdot(\frac{3}{2})^{2w-2}, \tag{8}$$

$$q \approx m+2\log_2 w+1.17w-1.17, \tag{9}$$

$$\begin{aligned} \zeta &\approx (2^m-1)\cdot w^2\cdot(\frac{3}{2})^{2w-2}\cdot(1-(\frac{2}{3})^{2w-2}) \\ &= (2^m-1)\cdot w^2\cdot((\frac{3}{2})^{2w-2}-1), \end{aligned} \tag{10}$$

where ζ denotes the count of the 3-2 compressors at the fictitious $(2w-2)$ levels. Then N_q can be computed as follows:

$$\begin{aligned} N_q &\approx 2^q-q-1-\zeta \\ &= (2^m-1)\cdot w^2-m-2\log_2 w-1.17w+1.17 \end{aligned} \tag{11}$$

3. SIMULATION AND ANALYSIS

3.1. Performance analysis

A conventional multi-valued inner product processor is presented in Fig. 4 to facilitate the overhead analysis of our design. As Fig. 4 shows, the 2^m-1 components of the two input vectors are fed into the individual inner product term generator serially, where the number of finite levels. Eqn. (2) is set to 2^w-1. During each cycle, the w^2 outputs of the individual inner product term generator is streamed into a Wallace-tree multiplication array to obtain the $2w$−bit product. Notably, a fast adder such as carry-lookahead adder (CLA) is required at the final stage of the multiplication. Then the product is fed into another CLA to get the accumulated partial sum of the inner product. Since the maximum value of the inner product, P_{max}, can be derived by

$$\begin{aligned} P_{max} &= (2^m-1)\cdot(2^w-1)\cdot(2^w-1) \\ &= 2^{m+2w}-2^{2w}-2^{m+w+1}+2^{w+1}+2^m-1 \\ &> 2^{m+2w-1}, \end{aligned} \tag{12}$$

for $m>1$ and $w>2$, the output bit length of the CLA is required to be at least $m+2w$, which is also the output bit length of the compressor unit as given in Fig. 1.

Similar to the approach taken for the derivation of Eqns. (7) and (11), the propagation delay of the Wallace-tree multiplication array counted by the number of 3-2 compressors can be estimated as follows:

$$D_{Wallace-tree} \approx \frac{\log \frac{w^2 \cdot (\frac{3}{2})^{2w-2}}{2}}{\log(\frac{3}{2})} - (2w-2) < 11.36 \log w - 1.71, \quad (13)$$

and the approximate number of 3-2 compressors used in the multiplication array becomes:

$$N_{Wallace-tree} \approx w^2 - 2\log_2 w - 1.17w + 1.17. \quad (14)$$

Next we need to evaluate the critical delay and the hardware complexity of the two CLA's. Based on the tree-like architecture of the CLA proposed by Dozza et al. [6], it can be shown that the delay of an r-bit CLA counted by the number of 2-input logic gates is

$$D_{r-bit\ CLA} = \log_2 r + 3. \quad (15)$$

Meanwhile, the number of 2-input logic gates used in this tree-like CLA can be shown as

$$N_{r-bit\ CLA} = 3r \cdot \log_2 r - 3. \quad (16)$$

In summary, the conventional scheme requires an extra $2w$-bit CLA, an $(m+2w)-$bit CLA, and an $(m+2w)-$bit register. However, the compressor unit as shown in Fig. 6 is replaced by the simpler Wallace tree, and only one set of individual inner product term generator is needed in this scheme. The extra hardware cost for our proposed scheme can be estimated as follows:

1). $w^2 \cdot (2^m - 1)$ AND gates for the inner product term generator.

2). $(2^m - 2) \cdot w^2 - m$ 3-2 compressors used in the compressor unit.

Although the hardware complexity of the above-mentioned conventional inner product processor is simpler than our scheme, the total delay of the inner product calculation caused by this simple yet slow architecture turns out to be

$$Delay_{Conventional\ scheme} = (2^m - 1) \cdot (D_{AND} + D_{Wallace-tree}, + D_{2w-bit\ CLA} + D_{(m+2w)-bit\ CLA}), \quad (17)$$

where D_{AND} denotes the delay of the AND gate, $D_{Wallace-tree}$ represents the delay of the Wallace tree multiplication array, while $D_{2w-bit\ CLA}$ and $D_{(m+2w)-bit\ CLA}$ stand for the delay of the two CLA's. As for the total delay of our proposed scheme, it can be expressed as follows:

$$Delay_{Our\ scheme} = D_{AND} + D_{Compressor-unit}, \quad (18)$$

where $D_{Compressor-unit}$ stands for the delay of the compressor unit. From Eqns. (7), (13), (15), (17) and (18), it can be shown that the total delay of multi-valued inner product calculation counted by the number of 2-input logic gates is reduced by

$$\Delta_{delay} \approx (2^m - 2) \cdot (22.72 \log w - 2.42) + (2^m - 1) \cdot (\log_2(m+2w) + \log_2 w + 7) - 4m. \quad (19)$$

As seen from Eqn. (19), the delay of inner product is improved significantly in our proposed scheme.

3.2. Verilog simulations and Chip implementation

In order to verify the correctness and the performance of the implementation of the inner product processor for the MV-eBAM, Verilog HDL is used to conduct a series of simulations with over 20,000 random testing vectors to explore the critical delays of the proposed architecture. The dimension of the input vectors to the inner product processor is 31, and each digit is two bits wide. The simulation results indicate a delay of about 6.4 ns for the critical paths. We use the TSMC 0.6 μm 1P3M technology to design the chip ,and we use Cadence Silicon Ensemble automatic place and route tools to generate the abstract view and the layout of the chip. At last the DRACULA and TimeMill are utilized to execute the full-chip-scale post-layout simulation. The circuit layout of the inner product processor is given in Fig. 7.

4. CONCLUSION

In this paper we have proposed a novel architecture of the inner product processor which can be employed in the implementation of multi-valued exponential bidirectional associative memory. It is deemed as a key component for SOC solutions for neural networks implementation. The systolic-like architecture of the compressor units can significantly reduce the carry propagation delay in the critical path of the inner product, which is clearly the bottleneck of the whole computation.

5. REFERENCES

[1] L. Breveglieri, and L. Dadda, "A VLSI inner product macrocell," *IEEE Trans. VLSI Systems*, vol. 6, no. 2, pp. 292-298, June 1998.

[2] C.-C. Wang, C.-J. Huang, and P.-M, Lee, "A comparison of two alternative architectures of digital ratioed compressor design for inner product processing," *Proc. IEEE Inter. Symp. on Circuits and Systems*, vol. I, pp. 161-164, June 1999.

[3] C.S. Wallace, "A suggestion for a fast multiplier," *IEEE Trans. Computers*, vol. 13, no. 2, pp.14-17, Feb. 1964.

[4] V. G. Oklobdzija, D. Villeger, and S. S. Liu, "A method for speed optimized partial product reduction and generation of fast parallel multipliers using an algorithmic approach," *IEEE Trans. Computers*, vol. 45, no. 3, pp. 294-305, Mar. 1996.

[5] D. Zhang, and M. I. Elmasry, "VLSI compressor design with applications to digital neural networks," *IEEE Trans. VLSI Systems*, vol. 5, no. 2, pp. 230-233, June 1997.

[6] D. Dozza, M. Gaddoni, and G. Baccarani, "A 3.5 ns, 64 bit, carry-lookahead adder," *Proc. IEEE Inter. Symp. on Circuits and Systems*, vol. II, pp. 297-300, June 1996.

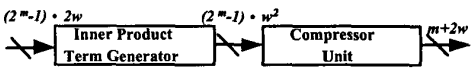

Fig. 1: The data flow of an inner prodect calculation for the MV-eBAM.

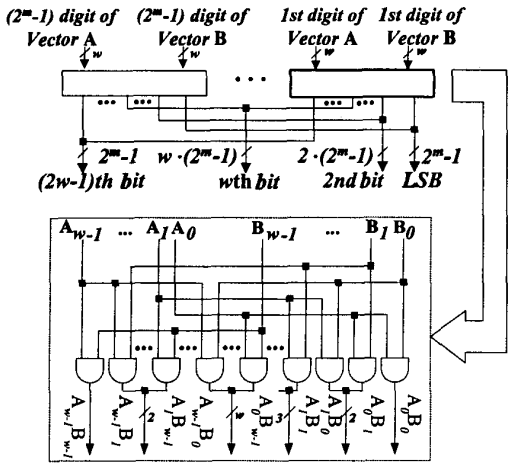

Fig. 2: The inner product term generator.

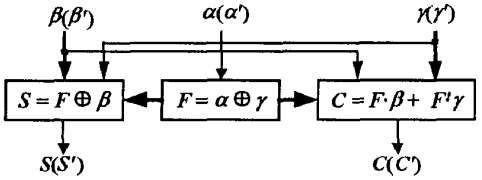

Fig. 3: A 3-2 compressor building block.

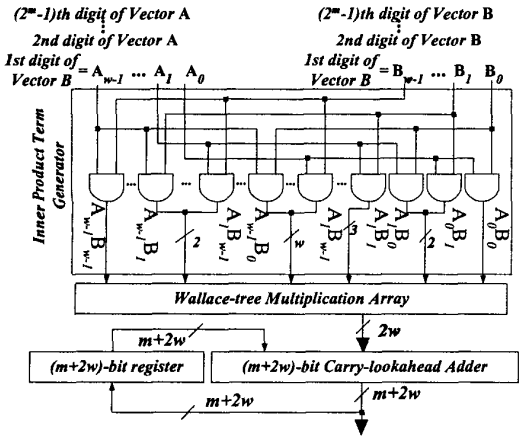

Fig. 4: A convential multi-valued inner product processor.

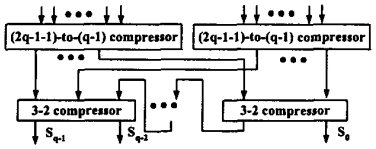

Fig. 5: An alternative architecture of $(2^q-1)-to-q$ compressor.

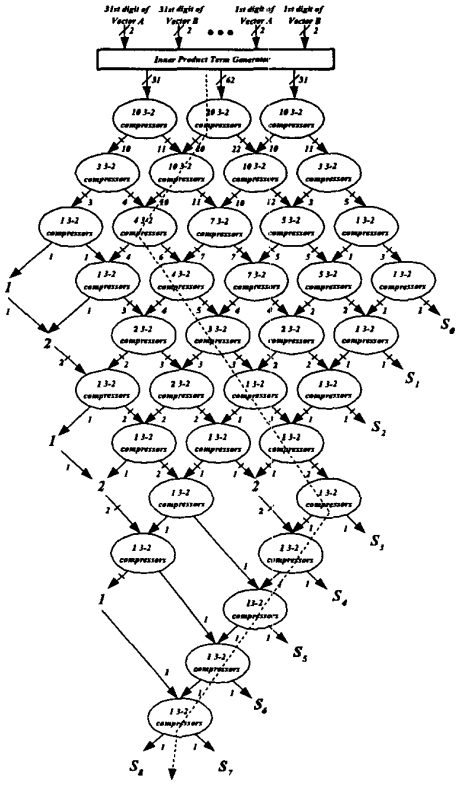

Fig. 6: Systolic-like architecture of a MV-eBAM compressor for $m = 5$ and $w = 2$.

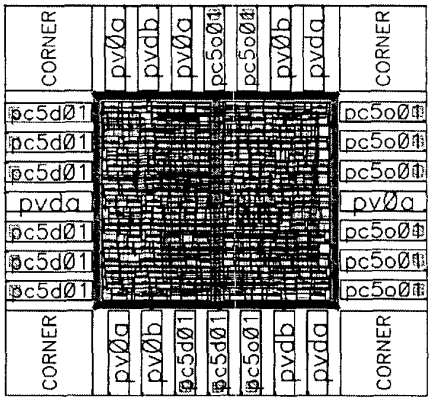

Fig. 7: Circuit layout of the MV-eBAM inner product processor for m=5 and w=2