Circuit Technologies for VLSI Associative Processors

(VLSI 連想プロセッサのための回路技術の研究)

Makoto OGAWA

Abstract

Associative processing is the widely used computational scheme in intelligent information processing. In associative processing, a large number of past experiences or knowledge are stored in the vast memory as template vectors representing their features. Then, when unknown objects or events are input into the system, they are converted to vectors and the most similar one is searched for from the template vectors in order to recognize what the input is. The associative processing is powerful and general, however, it is computationally very expensive and time consuming. Thus VLSI implementation is essential. In this thesis, variety of VLSI associative processors based on vector-quantization (VQ) or dynamic-programming (DP) matching have been developed for varieties of intelligent information processing. Especially, VQ processing is quite general and utilized in varieties of applications, thus VQ processors have been developed based on both analog and digital circuit technologies, enabling us to choose the optimum performance or functionality for each application.

For applications requiring computational efficiency, an analog VQ processor featuring non-volatile analog-memory-merged matching cell has been developed. The memory-merged matching cell computes similarity using the template vector data stored in itself. As a result, high-density circuit implementation has been achieved without memory-access bottle-necks. The prototype processor was designed and fabricated using 0.7-μm double-poli 1-metal CMOS process with EEPROM technology. Writing analog data into the memory-merged cell is experimentally verified using the prototype chip. The write-and-verify scheme using hot channel electron injection provides less-than 5mV resolution and the range of the memorized voltage from 3V to 4.2V for analog data writing.

An analog VQ processor technology has been developed based on a bell-shape element matching circuit aiming at high-density integration. The bell-shape current characteristics of the matching cell are produced by only four NMOS transistors with two complementary analog signals. The

matching cell developed in this work is compatible to the non-volatile analog memory, and the layout area is reduced to a quarter of the matching cell developed in the previous work. In addition, compact digital-to-analog converter (DAC) circuit with cyclic architecture has been developed for on-chip highly parallel conversion. A single CMOS-inverter featuring a double resetting scheme was employed as its gain stage. It significantly improves unity-gain characteristics of the CMOS inverter buffer in spite of its finite gain. The circuit ideas have been verified by measurements on experimental chips fabricated in a 0.6-μm double-poly CMOS process.

For applications requiring processing flexibilities, a general-purpose VQ processor featuring high-speed and versatile winner search capabilities has been developed. In order to achieve a high-speed operation, a two-dimensional bit-propagating scheme has been introduced to the winner-take-all (WTA) circuitry. As a result, the winner search is accomplished in a single clock cycle as compared to the conventional bit-sliced WTA approaches where clock cycles equal to the bit length of distance value are required. The high-speed WTA circuit with the variable-binary-block addressing scheme developed in this work allows various winner-search options like local winner search, winner sorting and so forth. The novel addressing scheme has been implemented by adding only a single auxiliary bit to the ordinary address code. In order to enhance flexibility in similarity evaluation, a multiplier function is also included in the SIMD distance computation unit with a minimal area penalty. As a result, weight multiplication to vector elements as well as the choice of either Manhattan distance or Euclidian distance as the dissimilarity measure has been made possible. A prototype VLSI chip was designed and fabricated using a 0.6-μm standard CMOS technology and the new concepts have been experimentally demonstrated. In addition, the experimental processor using 0.18-mm 5-metal process has been designed, whose performance is estimated at about 150 GOPS with 1.8W.

For the sequence-based matching, computationally expensive dynamic-programming matching of data sequences has been directly implemented in a fully-parallel-architecture VLSI chip. The circuit operates as digital logic in

the signal domain, while analog processing is carried out in the time domain based on the delay-encoding-logic scheme. As a result, high-speed low-power best-match-sequence search has been established with a small chip area. The typical matching time of 80ns with the power dissipation of 2mW has been demonstrated with fabricated prototype chips.

VLSI associative processors and their elemental circuits developed in this work would contribute to enhance performance or efficiency in the intelligent information processing. Introducing the low-cost VLSI associative processing, its applied field would be widened to resource-limited area such as mobile or ubiquitous computing, and it would be frequently or aggressively utilized in general IT systems or services.

Acknowledgement

With the utmost gratitude I would like to thank my advisor Prof. Tadashi Shibata for his loyal support and enduring guidance over many years.

Special thanks to Prof. Kunihiro Asada, Prof. Takayasu Sakurai, Prof. Hitoshi Aida, Prof. Shuichi Sakai, Prof. Kiyoharu Aizawa, and Prof. Minoru Fujishima for their review of the thesis, and their valuable comments, discussions to my research.

I would like to thank Prof. Yoshio Mita for his motivating engineering lessons. His insightful teaching has greatly expanded my interest in and enriched my knowledge of MEMS. I express my appreciation to Mr. Toru Murai for his advice; Ms. Tomi Takei, Ms. Michiyo Abe, Ms. Miei Komiyama, Ms. Kimiko Mori, and Ms. Motoko Inagaki for their supports.

I am very thankful to my colleagues and friends; Dr. Huaiyu Xu, Dr. Gu Qian Rong, Dr. Masakazu Yagi, and Dr. Toshihiko Yamasaki for their advices; Daisuke Kobayashi, Hiroe Kimura, Shuou Nomura, Kiyoto Ito, Teruyasu Taguchi, Hideo Yamasaki, Yasufumi Suzuki, and Masayuki Umejima for being dependable and supportive; and Yusuke Nakashita, Tomoyuki Nakayama, Hitoshi Hayakawa for their input and enthusiasm; Portions of this dissertation have been made possible with their assistance and backing.

I am immensely grateful to my parents and grandparents for their very prudent unconditional support. Finally, my deepest gratitude I must reserve for my wife Rie, whose patience and understanding I am very thankful for.

The VLSI chips in this study has been fabricated in the chip fabrication program of VDEC, the University of Tokyo with the collaboration by Rohm Corporation and Toppan Printing Corporation and by Hitachi Ltd. and Dai Nippon Printing Corporation, respectively. The analog EEPROM in this study has been fabricated with the collaboration by Oki Electric Industry Co., Ltd.. The work is partially supported by the Ministry of Education, Culture, Sports, Science and Technology under Grant-in-Aid for Scientific Research (No. 11305024, and No. 14205043) and by Japan Science Research for Evolutional Science and Technology (CREST).

Makoto Ogawa

The University of Tokyo

List of Publications

Journal Papers

[1] Kiyoto Ito, **Makoto Ogawa** and Tadashi Shibata, "A High-Performance Ramp-Voltage-Scan Winner-Take-All Circuit in an Open Loop Architecture," Japanese Journal of Applied Physics, Vol. 41, Part 1, No. 4B, pp. 2301-2305, April (2002).

[2] **Makoto Ogawa** and Tadashi Shibata, "A Delay-Encoding-Logic Array Processor for Dynamic Programming Matching of Data Sequences," submitted to IEEE Journal of Solid-State Circuits.

Refereed Papers at International Conferences

[1] **Makoto Ogawa** and Tadashi Shibata, "NMOS-based Gaussian-Element-Matching Analog Associative Memory," Proceedings of the 27th European Solid-State Circuits Conference (ESSCIRC 2001), Ed. by F. Dielacher and H. Grunbacher, pp. 272-275 (Frontier Group), Villach, Austria, September 18-20, 2001.

[2] Kiyoto Ito, **Makoto Ogawa**, and Tadashi Shibata, "A High-Performance Time-Domain Winner-Take-All Circuit Employing OR-Tree Architecture, Extended Abstracts, the 2001 International Conference on Solid State Devices and Materials (SSDM 2001), pp. 94-95, Tokyo, September 26-28, 2001.

[3] **Makoto Ogawa**, Kiyoto Ito, and Tadashi Shibata, "A General-Purpopse Vector-Quantization Processor Employing Two-Dimensional Bit-Propagating Winner-Take-All," in the Digest of Technical Papers of 2002 Symposium on VLSI Circuits, pp. 244-247, Honolulu, June 13-15, 2002.

[4] Kiyoto Ito, **Makoto Ogawa**, Tadashi Shibata, "A Variable-Kernel Flash-Convolution Image Filtering Processor," in Digest of Technical Papers, 2003 IEEE International Solid-State-Circuit Conference (ISSCC), Paper No. 26.7, pp. 470-471, San Francisco, February, 2003.

[5] Teruyasu Taguchi, **Makoto Ogawa**, and Tadashi Shibata, "An Analog Image Processing LSI Employing Scanning Line Parallel Processing," in

the Proceedings of the 29th European Solid-Sate Circuits Conference (ESSCIRC 2003), pp.65-68, Estoril, Portugal, September 16-18, 2003.

[6] **Makoto Ogawa**, and Tadashi Shibata, "A Delay Encoding-Logic Array Processor for Dynamic-Programming Matching," 30th European Solid-State Circuits Conference (ESSCIRC), pp. 311-314, Leuven, Belgium, September 21-24 2004.

Papers at Domestic Conferences (in Japanese)

[1] **小川誠**、伊藤潔人、柴田直、「2 次元ビットプロパゲーション WTA を用いた汎用 VQ プロセサ」、電子通信学会技術研究報告、（集積回路研究専門委員会(ICD)）、論文番号 ICD2002-170、pp.37-42、 2002 年 12 月.

[2] 伊藤潔人、**小川誠**、柴田直、「フラッシュコンボルーション型画像フィルタ演算プロセサ」、電子通信学会技術研究報告、（集積回路研究専門委員会(ICD)）、2003 年 5 月.

Awards

[1] Finalist Commendation, The Takeda Techno-Entrepreneurship Award 2001 to Tadashi Shitaba, Masakazu Yagi, Rong Gu, Toshihiko Yamasaki, **Makoto Ogawa**, and Daisuke Kobayashi, "Development of Human-Intelligence Systems Using Right-Brain Computing VLSI"

[2] 第 6 回 LSI IP デザイン・アワード IP 賞、伊藤潔人、**小川誠**、柴田直、「カーネルサイズ可変フラッシュコンボルーション型画像フィルタ演算プロセッサ」

Patents

[1] **小川誠**, 伊藤潔人, 柴田直,「画像処理装置及び画像処理方法」特願 2003-031569

[2] **小川誠**, 柴田直,「画像処理装置及び画像処理方法」特願 2003-036754

[3] **小川誠**、柴田直,「半導体回路」特願 2003-039740

[4] **小川誠**, 柴田直,「情報処理装置」特願 2003-039741

Contents

CHAPTER 1.
Introduction

I. Introduction

In the era of the information technology, computers are demanded to be more close to the human or to process more intelligently. Especially in such applications; agent system, dialog user-interface system, search engine, automatic monitoring system, etc., the human-like intelligent information processing is essential. However, flexible activities like the human, such as robust recognition under noisy environments or flexible decision-making from slight clues, are very difficult tasks for state-of-the-art computers, because the computers are designed to process the information using arithmetic or logical operation along with strictly defined sequential programs.

On the other hand, it is considered that the human processes the information flexibly by associating it with his/her vast memory or knowledge in parallel way. This associative processing is very powerful and widely used

in intelligent information processing. However, its computational cost on the conventional microprocessor is quite large. Thus, the purpose of this work is to develop dedicated VLSI processors for associative processing in order to enhance performance and efficiency.

In section II, another approach to the intelligent information processing is briefly described. The section III explains associative processing scheme and introduction to our systems with its applications. The section IV shows the scope of the thesis, and finally the organization of the thesis is described in the section V.

II. Approaches to Human Intelligence

The major approach to intelligent computing is to develop the algorithm or software technology based on microprocessors. Since the original idea of the microprocessor architecture was introduced by Shima, et al. in 1971, the performance of microprocessors has been explosively enhanced by process technology developments according to the Moore's Law [1] for a quarter of a century, with retaining the concept of the original architecture. By scaling of the device size, the state of the art microprocessors are growing to process several billion operations every second. In this regards, developing the program carrying out the human intelligence for the microprocessor, the computer is expected to become more intelligent using its extremely powerful computational power.

One of such approach is building a model of neurons, which is the processing element in human's brain, and emulating its behavior on the microprocessor. An artificial model of a single neuron was firstly introduced by McCulloch and Pitts in 1943 [2] . Their neuron has weighted multiple inputs and a single output, which fires when the linear summation of inputs exceeds a threshold. The system has plural of neurons, and they are

connected each other. And then, the behavior of the system is determined by connections and their weights between neurons. A lot of connection models, such as perceptron, Hopfield network, and so on, followed the neuron model, and actual systems were implemented. In such neural network systems, optimum network carrying out flexible computation is automatically constructed through the learning process.

In growth of the neural networks, to enhance the performance, VLSI implementations of the neuron were also introduced, such as neuron MOS by Shibata, et al. in 1991 [3] . It consists of a single MOSFET with a floating gate capacitance-coupled with multiple input terminals. The transistor turns on when the linear summation of inputs exceeds its threshold voltage as well as McCulloch-Pitts' neuron model. The software implementation of the neural network is quite computationally expensive, and the VLSI direct implementation offers the opportunity of efficient processing. However, connections between neurons are very complicated and varying transiently, thus a direct VLSI implementation is quite difficult in large networks.

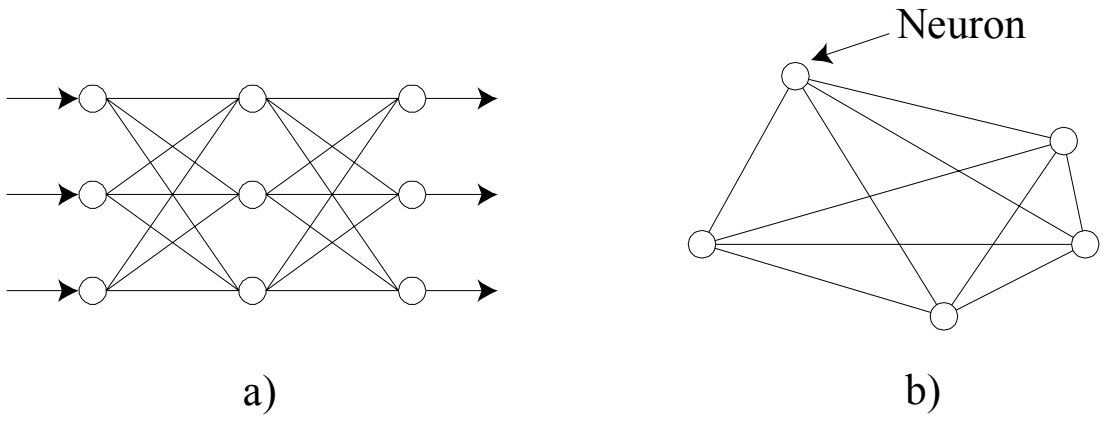


Figure 1.1 Connection models of neurons. a) Perceptron. b) Hop-field network.

III. Our Approach

Each neuron is quite simple, however, building the large-scale neural network comparable with the human brain requires extremely expensive computational cost. On the other hand, building the intelligent system from higher-level function in the human's brain is easier than that from the single neuron level. In this regards, we employs the higher-level function of the human brain as a primitive operation in the intelligent system, i.e. associative processing [4]. Figure 1.2 shows the concept of associative processing. In the associative processing, a large number of past experiences are stored in the vast memory, and when unknown objects or events are input into the system, the most similar one is searched for from the memory in order to recognize what the input is. The algorithm of associative processing is very simple, however, the computational cost is extremely expensive by the software implementation on the conventional microprocessor. To carry out this process in practical performance, the dedicated hardware accelerator, i.e. associative processor, is essential. The feature of our system is developing the VLSI-friendly algorithm for associative processing, and implementing it with VLSI technology, in order to enhance the performance or processing efficiency. Implementation of low-cost associative processing will enable us to build more intelligent system with the combination of simple associative operations.

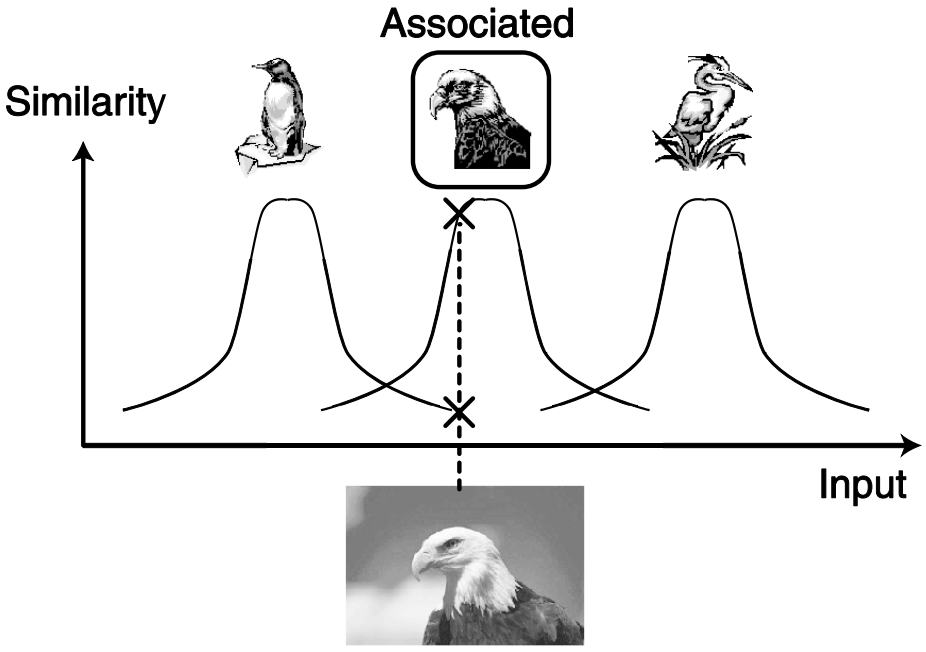


Figure 1.2. Concept of associative function.

Figure 1.3 shows a block diagram of image recognition system as an example of our associative processing system. The system has a VLSI associative processor as the core of the system. Input data given by sensors or user-interface program are firstly vectorized. This process is called as feature vector extraction, and optimum information for discriminating input object from others is extracted in a vector format. In image recognition system, the feature vector composed of 64-element scalar value is generated from spatial edge distributions of the 64×64 pixels image [5-9], where noisy and redundant information in the image are removed and compressed enough to represent features of the original image well. In the actual system implementation, the image feature extraction is also implemented using dedicated VLSI processor, or is embedded into the image sensor VLSI. The algorithm of feature vector extraction is quite different on applications. For instance, in the speech recognition system, the vector is generated from the

difference between the expected waveform from the learned data and real waveform [10]. A feature extraction process dominates performance or precision of associative system, thus its algorithm should be determined carefully.

After the feature vector extraction, association is carried out, where the most similar vector to the input is found out from the template vectors. The template vectors are generated using the same algorithm as the feature vector extraction and represents the past memory or knowledge. A large number of template vectors are stored in the huge database memory. Searching for the most similar template vector is carried out by calculating vector distance between the input vector and each template vector, and identifying the template vector having the minimum-distance. The associative processing is algorithmically equivalent to well-known and widely-used vector-quantization (VQ) algorithm, which carries out an approximation or mapping from multi-dimensional space to finite vector set.

Associative computing or VQ are currently applied to a lot of intelligent applications and their potentials are verified by software simulations on microprocessors or demonstrations on experimental chips. In the database search, an estate search engine is demonstrated on experimental FPGA association processor. It carries out flexible search taking customer's preference into accounts [11, 12]. For instance, it suggests the best matching estate to the customer whether matched item is found or not, like a real sales man. The other example of the database system is found at online IC-chip catalogue searching for the specific performance IC products [14-15]. In the image recognition system, cephalometric landmarks of the X-ray image can be identified, which is difficult task for professionals experienced for more than ten years [6-7]. In the hand-written character recognition, overlapped patterns are successfully discriminated, which is very difficult problem in image recognition tasks [5].

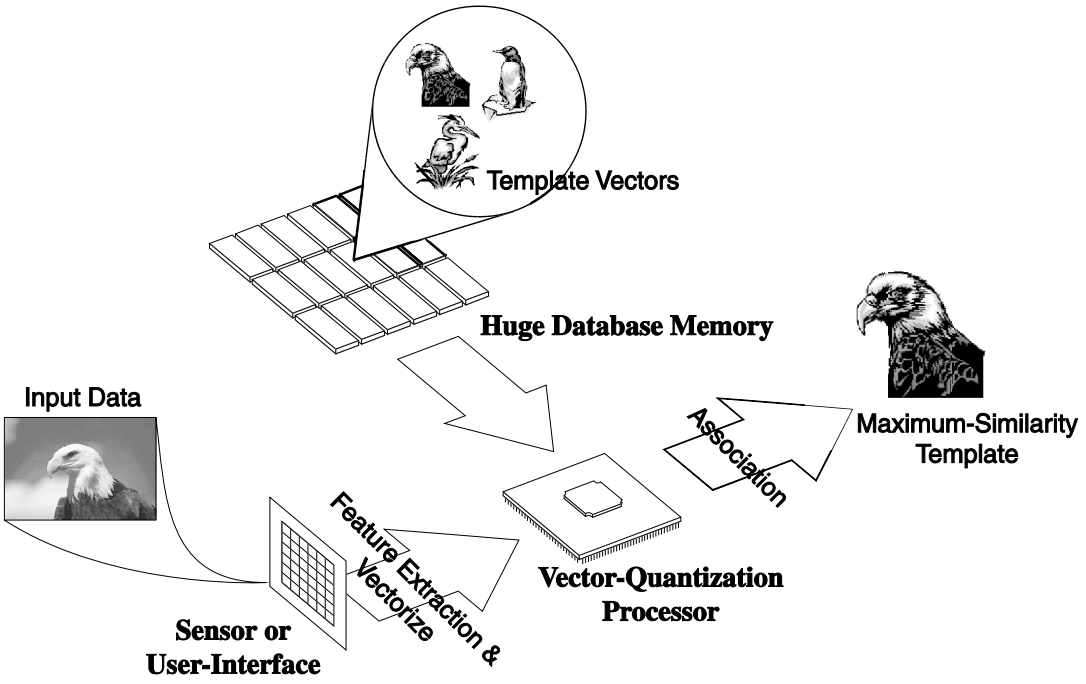


Figure 1.3. Block diagram of associative computing system.

IV. Scope of The Thesis

A scope of the thesis is the study of the VLSI associative processor. In order to enhance performance or efficiency of the computationally very expensive associative processing, the VLSI associative processor has been developed.

Associative processing or VQ is a generic function for the human-like intelligent computing, and can be utilized in a lot of applications, such as image recognition [5-9], speech recognition [10], database search engine [11-13], etc. In addition, the associative computing is required to be utilized in variety of environments from mobile or ubiquitous system with resource-limited devices to public services or system working running on high-end servers or workstations. In order to utilize associative processing in variety of applications with variety of requirements, the associative processors and circuit technologies developed in this work contains both analog and digital

technologies, enabling us to choose optimum performance or functionality for each application. Furthermore, the sequence-based matching processor is included for applications here the conventional vector-distance-based matching is not suitable, such as speech recognition or DNS sequence search.

V. Organization of The Thesis

In CHAPTER 2, a quick overview of vector quantization (VQ) and reviews on previous works of VLSI VQ processors are described. From CHAPTER 3 to CHAPTER 6, the primary contributions of this thesis are discussed. In CHAPTER 3, non-volatile analog-memory-merged matching circuit and its basic operation are described. In CHAPTER 4, an analog VQ processor is described, including novel bell-shape matching circuits and elemental circuit for the highly-parallel DA converter. In CHAPTER 5, general-purpose digital VQ processor is described, which features a high-speed winner-take-all circuit with a block-addressing scheme. In CHAPTER 6, dynamic-programming matching processor featuring delay-encoding-logic architecture is described. CHAPTER 7 summarizes major accomplishments of this study.

CHAPTER 2.
Review: Matching Processors

I. Introduction

In associative processing, pattern-matching techniques are extensively used to carry out robust classification and flexible decision-making under noisy environment. The vector-quantization (VQ) is a powerful tool for the pattern matching, which is very general and used in a lot of applications [16]. In addition, computation of VQ is quite simple, and is easy to implement on the VLSI circuits. Thus, VQ is the most potent candidates for associative VLSI processors.

In this chapter, the VQ algorithm and its VLSI implementations are reviewed. Firstly, the section II shows the algorithm and the issues when implemented in hardware. In section III, the several analog VLSI architectures are reviewed, and sections IV and V provide the conventional analog

implementation of the key circuits. In section VI, the digital VLSI architectures are reviewed, and sections VII and VIII provide the conventional digital implementation of the key circuits. Finally, in section IX, summarize the VLSI implementation of the VQ algorithm.

II. Vector Quantization Algorithm

Basically, vector quantizer Q of dimension k and size N is a mapping from a vector in k-dimensional Euclidean space into finite set containing N output or reproduction points, which are called template vectors or a codebook. Namely, the VQ process is a mapping from multi-dimensional vectors to the scalar values.

$$Q: \quad R^k \rightarrow C, \{ C = (y_1, y_2, ..., y_N) \text{ and } y_i \in R^k \} \text{ (Eq. 2.1)}$$

In usual VQ applications, such as data compression or pattern matching, the nearest neighbor template vector is chosen as representative vector as shown in Figure 2.1, and this type of VQ is called as a nearest-neighbor vector quantization. In the nearest-neighbor VQ, input vector is approximated to the one having with the maximum-similarity or the minimum-dissimilarity out of template vectors, and encodes into its index value. The obtained index value is utilized, for instance, as the short bit-length substitute value in the compression applications, or as the symbol of the recognized object in the recognition applications.

In the VQ computation, similarity in vector space is measured as the distance function between input vector and each template vector, and the vector having the smallest distance value is searched for from all the template vectors. For similarity evaluation, Euclidean distance is usually employed in software implementations as following.

$$d_i = \sum_{j=0}^{k} \left(x_j - t_{i,j}\right)^2 \qquad \text{Eq. (2.2)}$$

Here, index i represents the index of the template, and index j is the index of the element. x_i and $t_{i,j}$ is input vector element and i-th template vector element, respectively. Algorithmically Euclidean distance is favorable, however, Manhattan distance is more advantageous for VLSI implementations due to its simplicity.

$$d_i' = \sum_{j=0}^{k} \left|x_j - t_{i,j}\right| \qquad \text{Eq. (2.3)}$$

Additionally, in some applications such as database search, significance of the vector elements is altered.

$$d_i'' = \sum_{j=0}^{k} w_j^2 \left(x_j - t_{i,j}\right)^2 \qquad \text{Eq. (2.4)}$$

$$d_i''' = \sum_{j=0}^{k} w_j \left|x_i - t_{i,j}\right| \qquad \text{Eq. (2.5)}$$

In typical VQ processing, the memory access and similarity computation become bottleneck, because similarity evaluation for all template vectors must be carried out to find out the nearest template vector. In any case of the similarity function, the order of computational volume to measure distance is determined by the product of dimension size k and template vector number N as $O(k \times N)$. In usual applications, k becomes more than several tens, and N becomes more than several hundreds, thus the computational volume is quite large. Therefore, efficient similarity evaluation and memory access are key issues in hardware implementation.

After the similarity evaluation, the most similar template is searched for, comparing all the similarities. This operation is called as winner-take-all (WTA). In software implementations, WTA requires $O(N)$ steps for a loop

scanning all the N inputs. In real-time processing, however, it becomes quite large delay time, thus parallel search by the dedicated circuit is favorable.

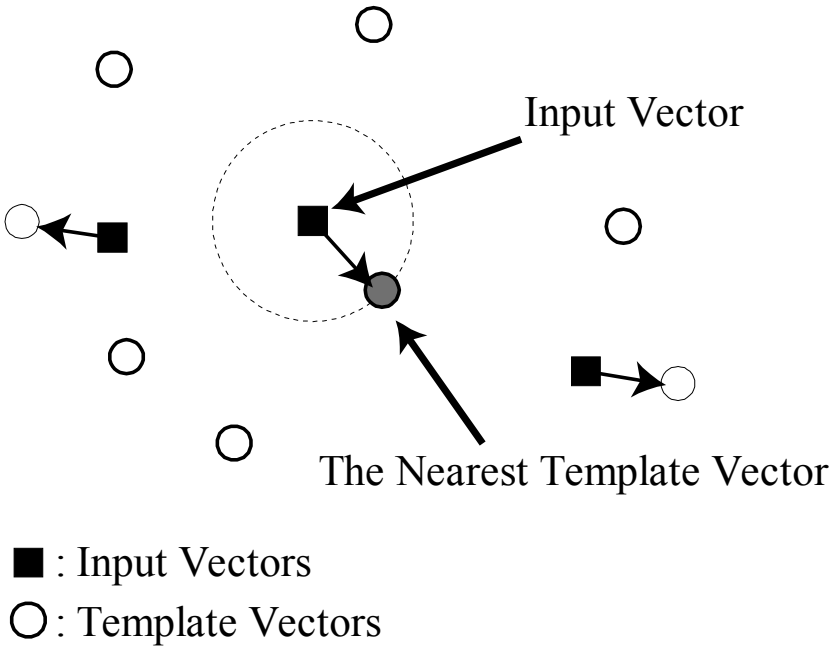


Figure 2.1. Concept of nearest neighbor vector quatization.

III. Analog VQ Processor Architecture

Analog circuits usually require less transistors than digital one to compute a single function, and the analog processors have been developed aiming at highly efficient implementation in terms of power dissipation and layout area. Typical analog VQ processors [17-21] employ matrix architecture as shown in Figure 2.2. The input vector is input in element parallel to the two-dimensional matching cell array. In the cell array, computations for all vector elements of all template vectors are carried out in parallel. Each cell calculates similarity between elements and stores the template data in itself. Cells of processors in Refs. [17, 19-21] store the analog data dynamically

using capacitors, thus they require refreshing every several milliseconds. Cycle time for refreshing is far longer than matching time, but each refreshing operation requires a large volume of digital-to-analog conversions and memory access, which limit processing parallelism. On the other hand, the processor in Ref. [18] employs non-volatile analog-memory for template-vector storage, and is free from refreshing. As the result, it achieves higher parallelism than others.

Calculated similarities between elements are summed up for each template vector. When the matching cell outputs the similarity as current [17] or charge [18, 21], the summation is easily carried out on a single bus-line. On the other hand, when it outputs as voltage, the summation is carried out using capacitance coupling [19]. The output vector similarities are fed to the winner-take-all circuit in parallel, and the most similar template is identified.

.

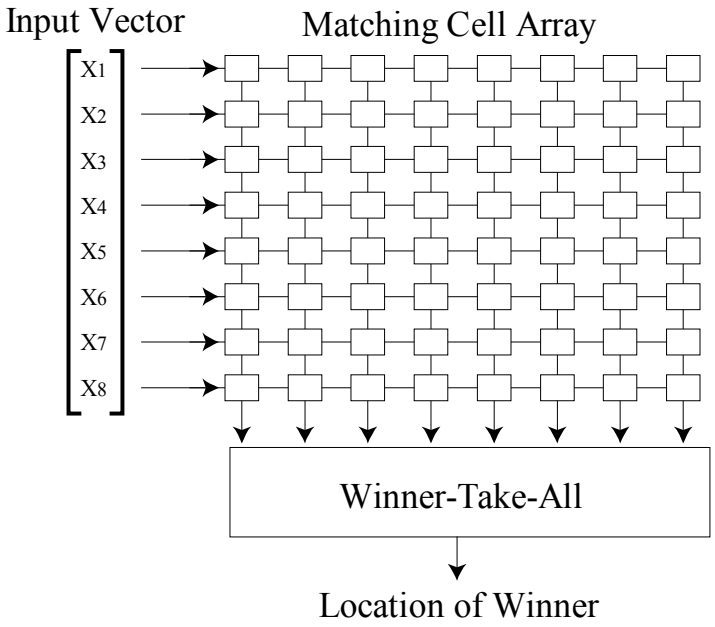


Figure 2.2. Matrix architecture for typical analog VQ processor.

IV. Analog Similarity Measurement Circuit

Analog circuits provide opportunities for efficient computation, and a lot of analog matching circuits have been developed [17-20, 22-28]. These circuits calculate similarity between two input voltage based on Gaussian [23, 25, 26], square function [17, 26], absolute difference [18, 24], and so on. Some circuits provide reconfigurabilities for similarity function with simple circuit configurations [26-28]. All the analog circuits achieve desired function with only several transistors, while digital implementations are far more complicated. In this section, several circuits with memory storage function in itself are shown, which are free from memory access bottleneck and provide large impact to associative processors.

A. Current-mode Square-Low Circuit

Current-mode analog VQ processor by Tuttle has massively parallel analog matching cell array, which carries out Euclidean distance calculation [17]. Matching cell array store template vector data in itself, and also calculates the correlation between input and stored data. The feature of matching cell is self-calibrated storage function, which depresses the device deviations. The problem of the circuit is the large power dissipation due to the DC current flow during the comparison.

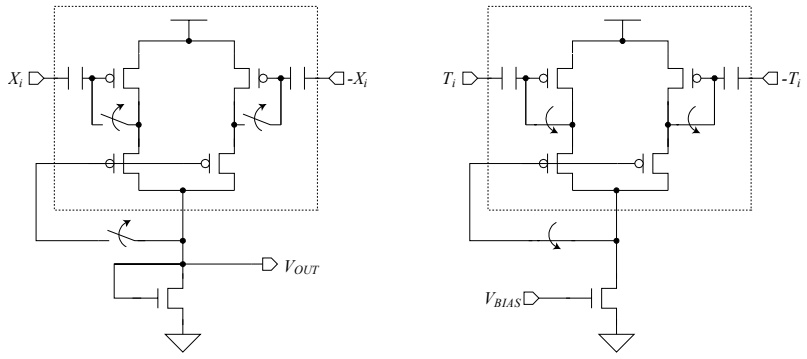


Figure 2.3. Circuit configuration of current mode square-low circuit.

B. Charge-based Absolute-difference Circuit

Charge-based VQ processor was proposed by Kramer, which employs analog flash memory technology [18]. The similarity measurement is carried out on floating gate and channel charge of the memory cell. The feature of the processor is very low-power dissipation by the charge-mode computation without no DC current, and highly integrated matching cell by the flash memory. For these reasons, very efficient computation is achieved.

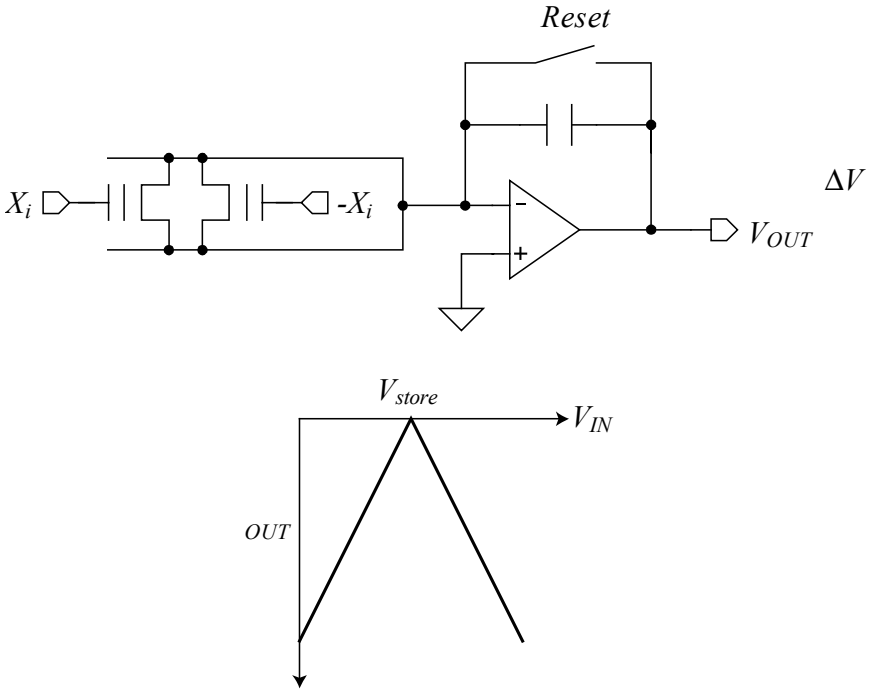


Figure 2.4. Circuit configuration of charge-based matching circuit.

V. Analog Winner-Take-All Circuit

The first analog winner-take-all (WTA) circuit was introduced by Lazzaro, et. al. in 1989 [29]. It identifies the maximum current input with simple circuit configuration. Then, varieties of analog WTA circuits have been

developed based on voltage-mode [30, 32], current-mode [31], time-domain [34, 36], and mixed-signal [33, 35] implementations. In this section, several high-performance WTA circuits are shown.

A. Comparator Tree Winner-Take-All

The comparator tree for finding out the maximum-similarity template is composed of two-input comparator in a binary-tree shape [17]. The comparison is carried out with two inputs precisely, and the winner is passed to the next comparator of the tournament tree through the switch. In this tree configuration, the delay for the comparison becomes logarithm of the number of input data, namely the number of stages. Thus, the delay time is quite small.

The problem of the configuration is the long distance analog signal transfer in the comparator tree. Too large distance of analog signal transfer is easily affected by cross talk noise, and requires a large delay time. The large power-dissipation for the analog buffer is also disadvantage.

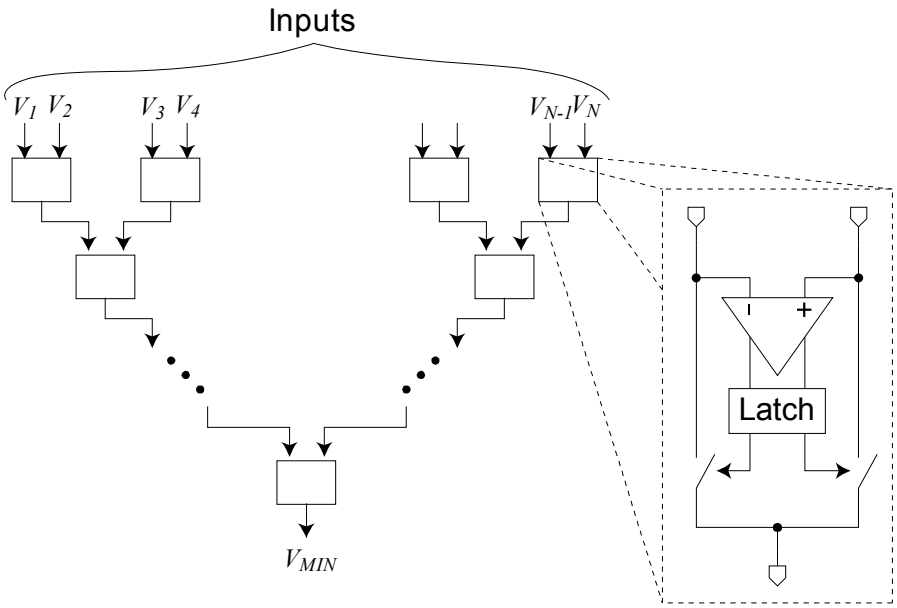


Figure 2.5. Comparator-tree winner-take-all circuit.

B. Mixed-signal Winner-Take-All

The mixed signal winner-take-all circuit carries out maximum/minimum AD conversion [33]. The maximum/minimum value is searched by the binary search technique in n steps for the n-b precision analog inputs. An auto-zero resetting comparator attached to each input carries out highly precise comparison with common reference voltage generated by DA converter. The identified maximum/minimum value is output in digital format. A disadvantage of this configuration is relatively low-speed operation compared to time-domain scheme described in the next section. It's due to large delay time of DA conversion and of the large-OR-gate to feedback comparison results to the DA converter.

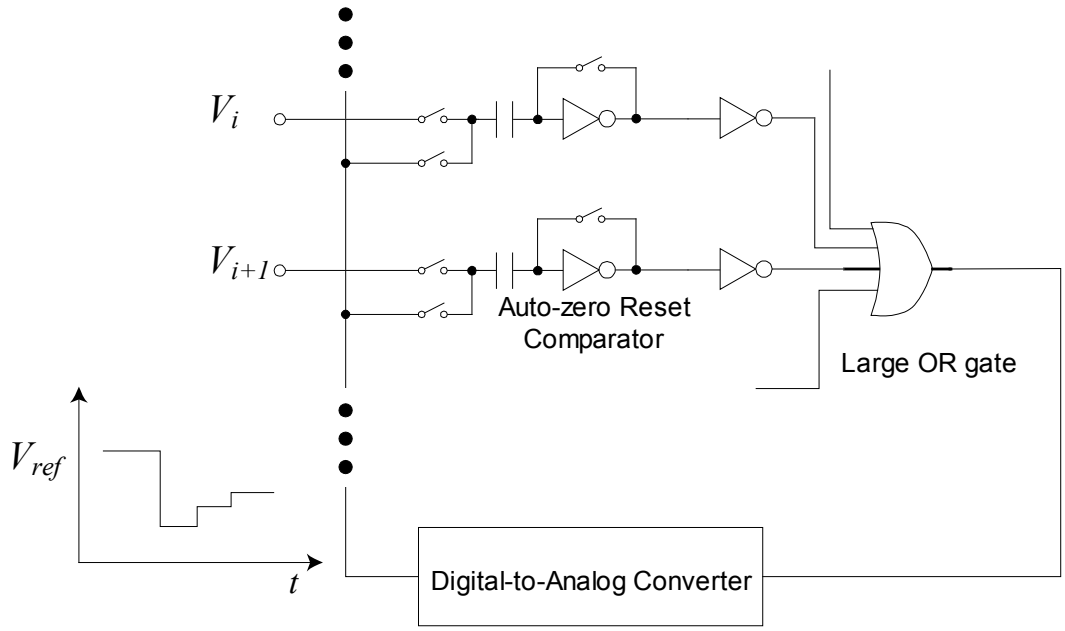


Figure 2.6. Binary-search-type winner-take-all circuit.

C. Time-domain Winner-Take-All

The analog signal transfer to far away destination is quite difficult in terms of the noise or delay time. Time domain technique is employed to transfer the analog signal with digital bit signal [36, 65]. The input voltage is converted to the difference of delay time by the voltage to time converter with ramp-up reference voltage [36]. The difference of time is compared by the latch, which signal comes first, and the first arriving signal is passed to the next stage in tournament tree. The advantage of this configuration is pure digital configuration in latch tree, and analog signal transfer is limited in local area.

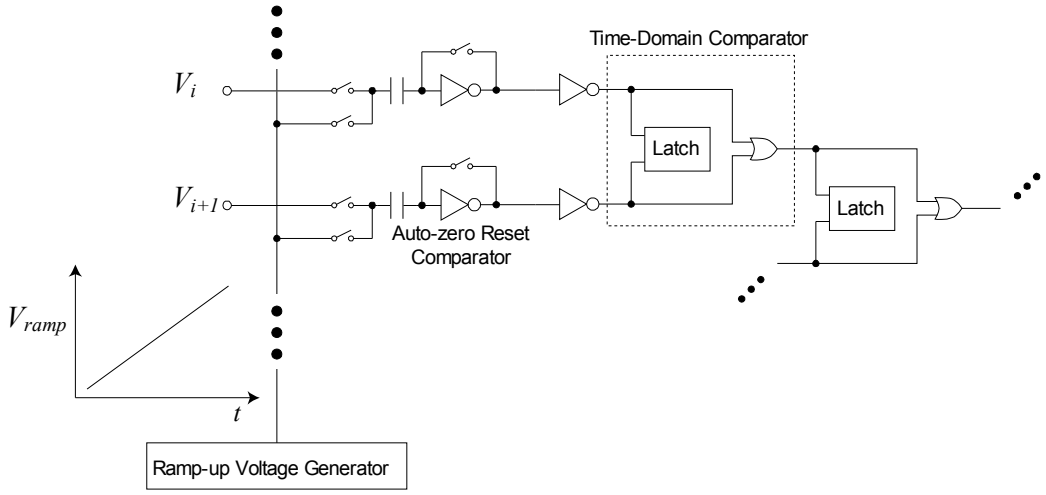


Figure 2.7. Time-domain winner-take-all circuit.

VI. Digital VQ Processor Architecture

Conventional digital VQ processors are divided into two types of processors; full-search type processors [37, 38, 40-43, 45, 47-52] and pruned-search type processors [39, 44, 46]. In full-search type processors, all the template vectors are calculated for similarities, while pruned-search type processors algorithmically omit computations for template vectors having

low probability to become the winner. The full-search type processor requires far more computational power than pruned-search type. However, the straightforward full-search scheme is quite general and sometimes achieves higher throughput due to the highly parallel computation [49]. In this section, two types of processor architectures for full-search type processors are described.

A. Digital Memory-based Architecture

Typical digital VQ processors employ memory-based architecture in order to solve the memory-access bottleneck [47-53]. The similarity evaluation elements are attached near the memory block, and highly parallel memory access and computation are achieved. Processing is carried out in vector parallel and element serial scheme with single-instruction-stream multiple-data-stream (SIMD) architecture. WTA is also carried out in vector parallel scheme, so the latency of each VQ operation is small. The dimension size is easily reconfigured by controlling memory access, but the vector number reconfiguration is difficult due to the fixed number of vector-parallel computation unit and the number of winner-take-all inputs.

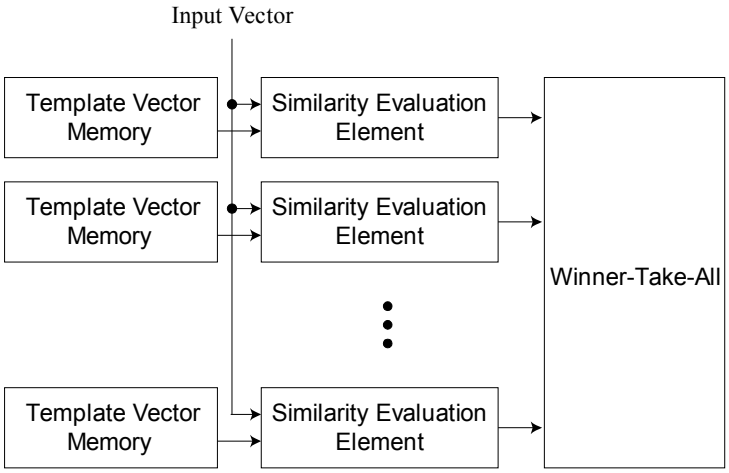


Figure 2.8. Memory-based architecture for digital VQ processor.

B. Digital Systolic-Array Architecture

Systolic-array architecture is employed for the flexible vector-number reconfiguration, which is achieved by the vector-serial element-parallel scheme [45, 52]. It also achieves the good compatibility to the serial maximum-similarity search, which can be implemented more easily than parallel WTA circuit. However, in typical VQ applications, the dimension size is smaller than the vector number, and parallelism of element-parallel systolic-array is lower than that of vector-parallel memory-based scheme. Therefore, the latency for each VQ is relatively larger than memory-based architecture.

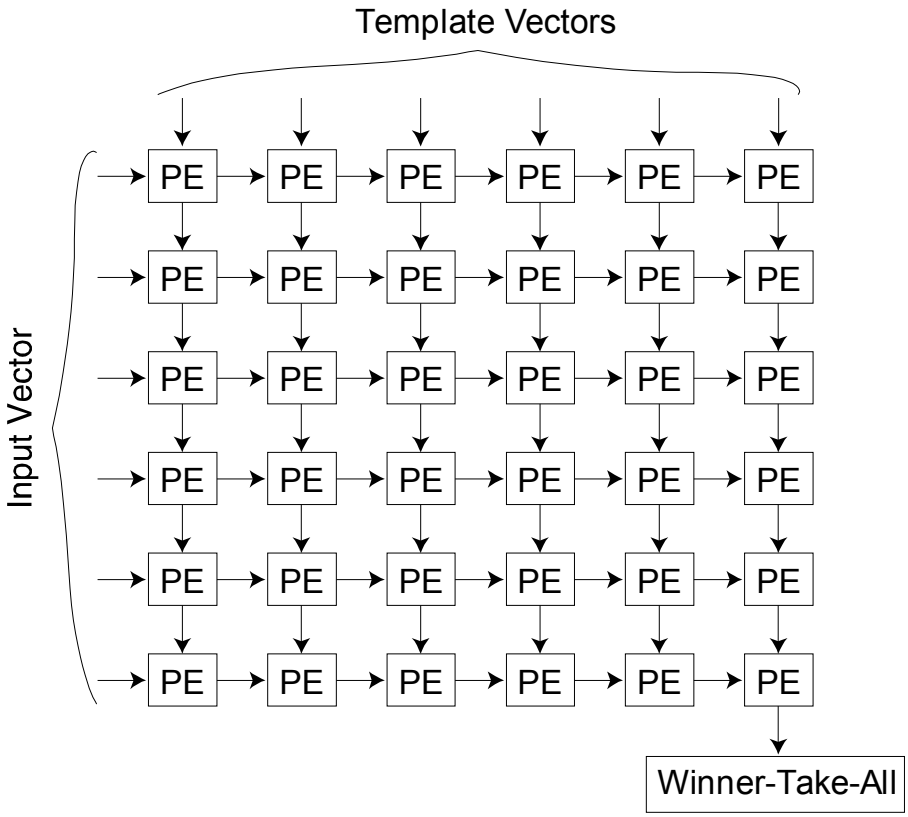


Figure 2.9. Systolic array architecture.

VII. Digital Similarity Measurement Circuit

A. Manhattan Distance Datapath

In typical digital VQ processor, Manhattan distance is used due to its simplicity. The Manhattan distance computation is composed of operations of absolute-differences and their accumulations. Usually, more than two adders are required for each absolute difference operation, and adder has large delay time and not favorable in high-speed configuration. However, absolute difference operation is embedded in Manhattan distance calculation, the adder for the negation can be eliminated by the accumulator as illustrated in Figure 2.10 [49]. This scheme contributes to small layout area and high-speed operation. This element serial scheme is compatible to the reconfiguration of the dimension size, which is achieved by controlling accumulation times.

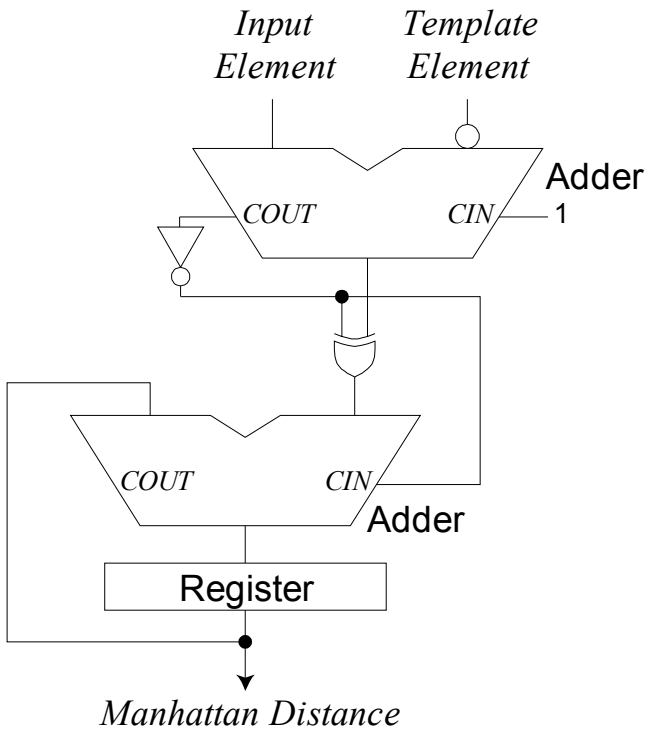


Figure 2.10. Digital Manhattan distance computation circuit.

B. Redundant Manhattan Distance Datapath

The performance for Manhattan distance computation is enhanced by the fully parallel computation with element parallel bit serial scheme, and redundant bit expression [50, 51]. As a result, the similarity measurement is carried out in clock cycles identical to the bit-length of vector element. In this configuration, however the number of element is strictly restricted to the processor design, and general-purpose usage is quite difficult.

VIII. Digital Winner-Take-All Circuit

A. Bit-sliced Winner-Take-All

Bit-sliced WTA scheme is the highly parallel digital WTA [47-51, 53, 54]. The input data are processed in bit-serial input-parallel manner and carries out WTA action in clock cycles identical to the bit-length. The advantage of this scheme is that processing clock cycles is independent from the number of inputs, and determined by the bit-length. It is very compatible to the highly parallel configuration. The disadvantage is the large latency for each 1-bit WTA action due to a large AND gate, which is distributed in the entire chip.

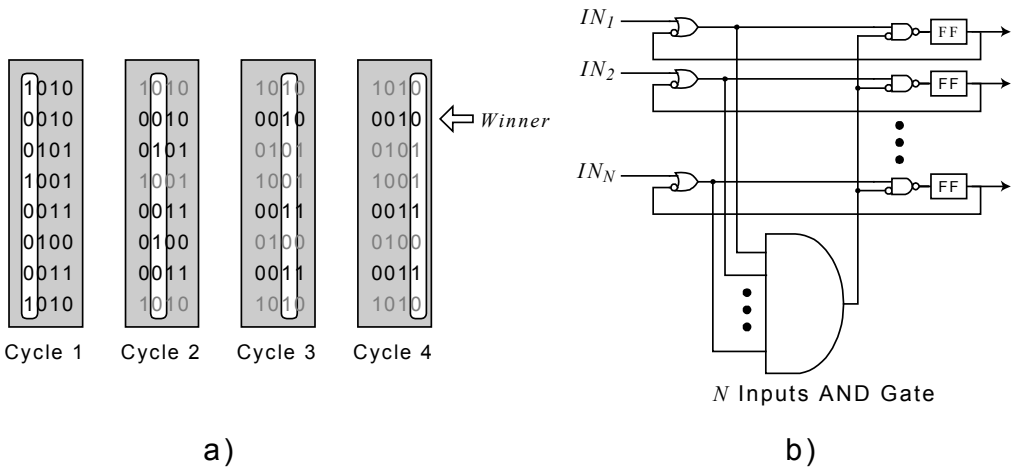


Figure 2.11. Bit-sliced winner-take-all circuit.

IX. Summary

Applied fields of the VQ algorithm are quite wide, and the requirements for the VQ processor is quite various. In this regards, the circuit technology for the VQ should be chosen carefully.

In terms of applied applications, analog circuits are suitable with the applications dealing information containing a lot of noises, due to the limitation of the processing precision. For example, the pattern classification or pattern recognition at early visual processing is the one of the most suitable application, since the natural scene containing a lot of unexpected noise. On the other hand, applications like personal identification using biometrics such as face or fingerprint features are to be implemented in digital circuits. In such applications, input vectors are usually well-defined feature-extracted vectors generated using application specific algorithms. Pattern matching in such area has a priority to enhance processing precision rather than processing efficiency. Compressions of image or sound data are also suitable for digital implementation. In these applications, in order to minimize compression errors, the most precise and algorithmic processing is essential.

In terms of performance and power-dissipation, analog processor has a opportunity for extremely highly parallel and low-power processing. An analog non-volatile memory-based matching cell provides more efficient computation for similarity measurement than the other conventional analog implementation or digital implementation. It has still competitive performance against state of the art scaled digital circuits. However, other analog implementations are suffered from high power-supply voltage, and the merit of low power computation is decreasing.

CHAPTER 3.
Non-volatile Analog-memory-merged Matching Cell

I. Introduction

The vector quantization (VQ) algorithm requires a large number of template vectors, and the VQ processor has large on-chip memory and highly parallel processors. On mobile or ubiquitous devices, and so on, requirements for power-dissipation or memory size are strictly limited, and the efficient processing is a key issue. Fortunately, the VQ algorithm itself is very robust, and computational error caused by analog processing affects a little to the result, thus analog processors has a good opportunity in certain class of applications.

In this respect, analog matching processors have been developed in the

past [17-21]. The conventional processors employ simple circuit for the distance computation or winner-take-all processing in order to reduce layout area and computational efficiency. On the other hand, the template vectors are stored as digital values in the external memory [17, 19-21], and they are transferred to the analog processing elements via digital-to-analog converters. As a result, total merits of the analog processing become smaller than expected.

In this work, the non-volatile analog-memory-merged matching cell has been developed in order to reduce overhead of transfer and conversion for the template vector data in terms of power-dissipation and layout area. The matching cell developed in this work merges two functions: storage for the template memory and computation for distance, employing functional-memory logic configuration. The analog data is directly stored in the matching circuit using current refereed scheme enabling high-precision analog-data writing [55]. Then, the computation is carried out on their floating gates. As a result, the template data stored in the chip can be directly processed at the memory circuit without data transfer and conversion overhead.

In the section II, the memory-merged matching circuit is presented. In the section III, the prototype processor is shown, and then the disturbance against analog memory writing is discussed. Finally, conclusions are given in the section V.

II. Circuit Configurations

A. Conventional Matching Circuit

In order to clarify our analog-memory-merged matching cell, let us first explain the conventional matching circuit not employing analog-memory technology [24]. Figure 3.1 shows the circuit configuration of the conventional matching circuit. The circuit carried out the absolute difference

function, which composes of the distance measurement function as follows,

$$D = \sum_{i=0}^{n} |x_i - t_i| \qquad \text{(Eq. 3.1)}$$

The circuit consists of a pair of NMOS transistors whose gate is connected to a capacitor and a switch. The pair of NMOS transistor calculates the subtraction of two analog values V1 and V2, which are inputted one after the other using switches SW1, SW2, SW3, and SW4. And then, the maximum one of subtraction results, i. e. the absolute difference, is outputted using source-follower circuit.

Figure 3.2 shows the operation of absolute-difference circuit. Firstly, the source nodes of NMOS transistors are set to VSS, and the analog voltages V_1 and V_2 are applied to gates by turning on the switches SW1 and SW4 and turning off the switches SW2 and SW3, as shown in the Figure 3.1A. And then, the switches connected to NMOS's gates are turned off by applying Φ_R="0". At this moment, Q_1 and Q_2 are charged on the gates, which are given as follows,

$$Q_1 = -CV_1 \qquad \text{(Eq. 3.2)}$$

$$Q_2 = -CV_2 \qquad \text{(Eq. 3.3)}$$

Here, C represents a capacitance value of the capacitor. After resetting, the inputs are exchanged each other, then the voltages on the floating gates are given as follows,

$$V_{F1} = \frac{C}{C_{TOT}} V_2 + \frac{1}{C_{TOT}} Q_1 = \gamma(V_2 - V_1) \qquad \text{(Eq. 3.4)}$$

$$V_{F2} = \frac{C}{C_{TOT}} V_1 + \frac{1}{C_{TOT}} Q_2 = \gamma(V_1 - V_2) \qquad \text{(Eq. 3.5)}$$

Here, C_{TOT} represents the total capacitance of the floating gate, and γ is

C/C_{TOT}. NMOS transistors are connected as the source follower in parallel. When the two NMOS transistors operate as source-follower circuit, the output voltage appeared at V_{OUT} is following to the maximum of their gate voltages. In this manner, the absolute of subtraction of V_1 and V_1 is appeared at the output terminal.

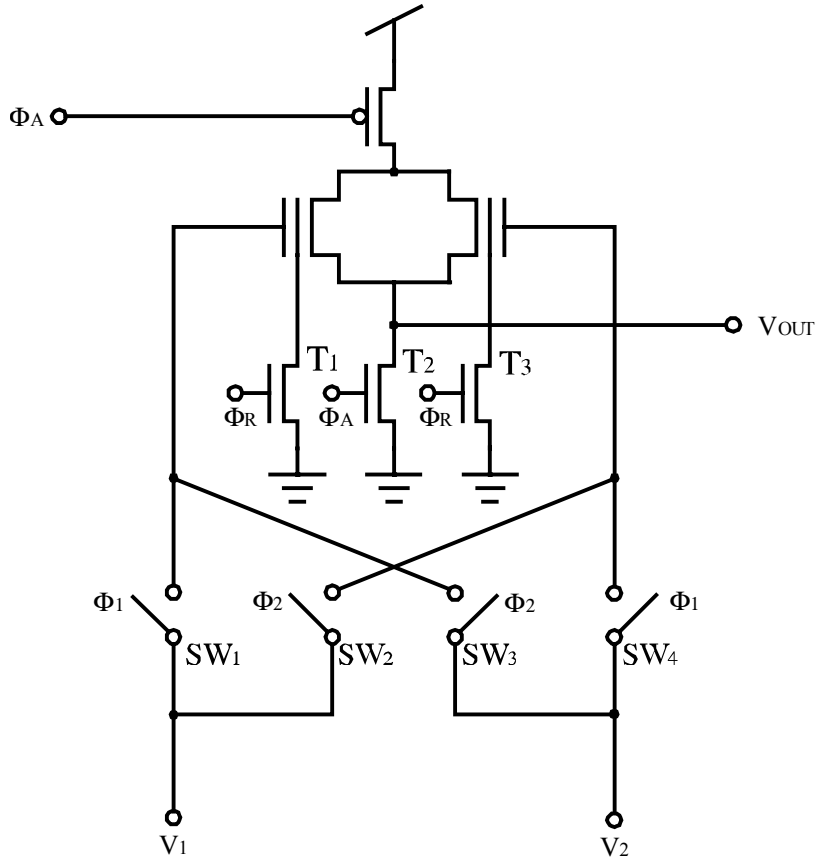


Figure 3.1. Conventional absolute-difference circuit.

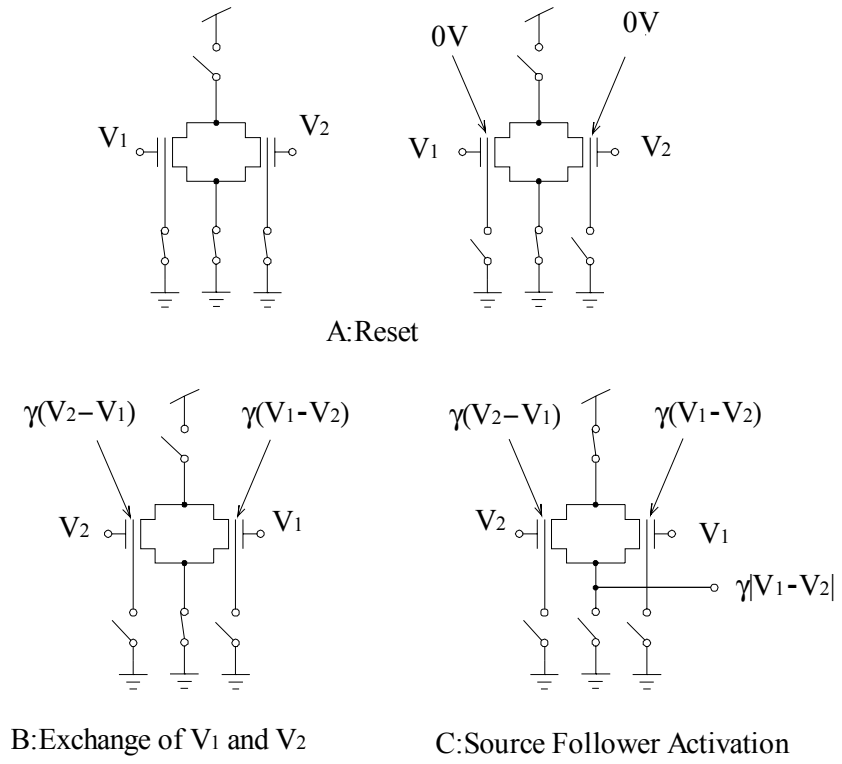


Figure 3.2. Operations of conventional absolute-difference circuit.

B. Analog-Memory-Merged Matching Circuit

Figure 3.3 shows the non-volatile analog-memory-merged matching circuit. The circuit is composed of two NMOS memory transistors M1 and M2, and a reset transistor T1. To store the analog voltage V_M, the memory transistor has floating gate, which is fully isolated. After memorizing analog voltage to the memory, a charge on the floating gate becomes

$$Q = C_{TOT}V_{REF} - CV_M \qquad \text{(Eq. 3.6)}$$

Here, C_{TOT} represents the total capacitance of the floating gate including parasitic capacitance such as gate-drain, gate-source and source. Details of writing scheme of the memory is described in the section C. The matching operation is carried out as follows. The matching circuit has the two memory

transistors, and the voltages $V_B + T_i$ and $V_B - T_i$ are memorized. V_B is the common bias voltage utilized to adjust floating gate voltage, which does not affect the subtraction results. In this manner, the two memory transistors memorize the complementary voltage. After the writing analog voltage V_M, the input value is applied to the control gate as analog voltage V_X. At this moment, the floating gate voltage V_F is given as

$$V_F = \frac{C}{C_{TOT}} V_X + \frac{1}{C_{TOT}} Q + V_{REF} \qquad \text{(Eq. 3.7)}$$

$$= \gamma(V_X - V_M) + V_{REF} \qquad \text{(Eq. 3.8)}$$

From the Eq. 2, it is understandable that the subtraction between input voltage V_X and memorized voltage V_M is appeared at the floating gate. After the subtraction, the absolute value of them is output using source-follower circuit same as the conventional matching circuit. The result becomes

$$V_{OUT} = \max\{\gamma(X_i - T_i) + V_{REF} - V_T, \quad \gamma(T_i - X_i) + V_{REF} - V_T\} \qquad \text{(Eq. 3.9)}$$

$$= \gamma|X_i - T_i| + V_{REF} - V_T \qquad \text{(Eq. 3.10)}$$

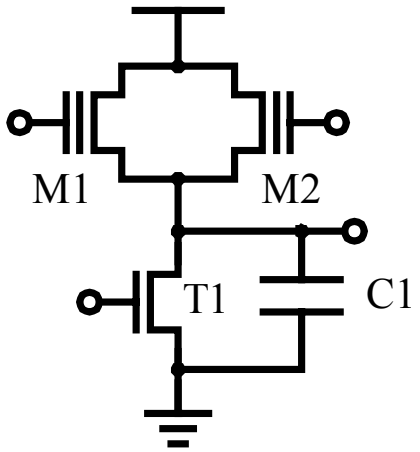


Figure 3.3. Non-volatile analog-memory-merged absolute-difference circuit.

Figure 3.4 shows the matrix configuration of the matching cell. The absolute difference at each memory circuit is coupled with other cells of the same template vectors, and the average voltage of the absolute differences is output from the array.

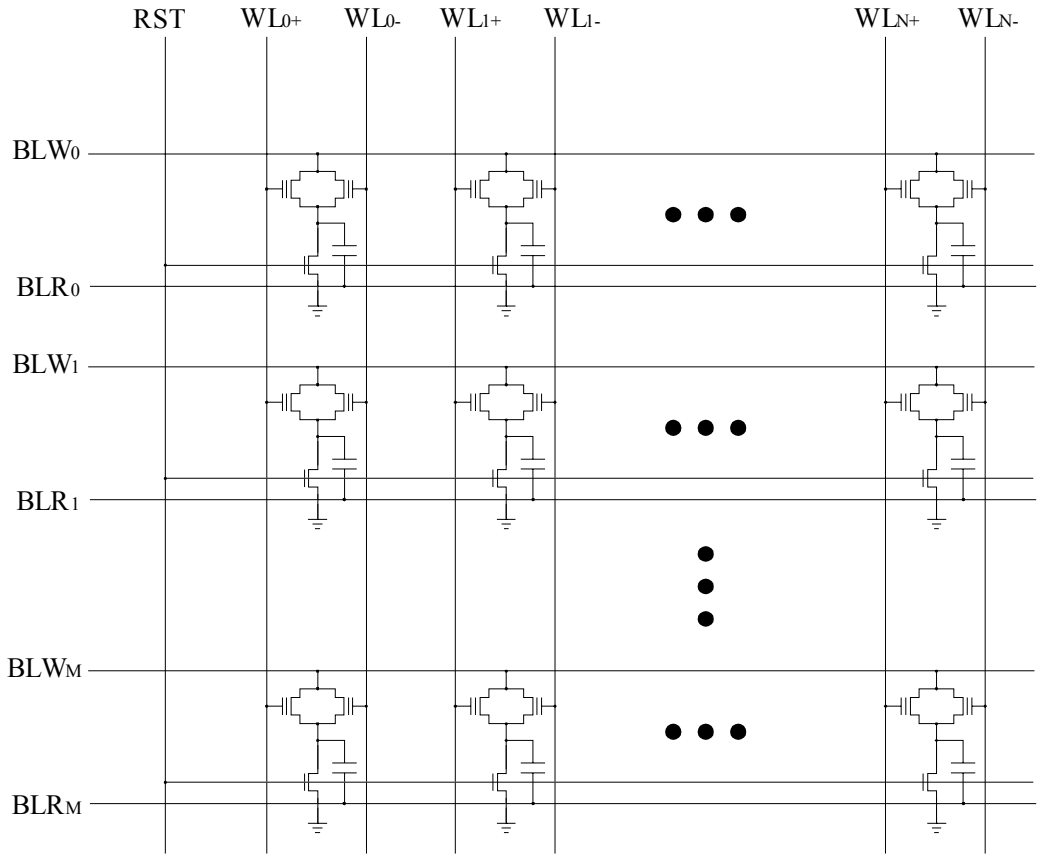


Figure 3.4. Circuit configuration of memory array.

C. Memory Write and Verify Circuit

The writing analog value is carried out as follows. Firstly, the high voltages are applied to the control gate and the drain-source of the memory, which cause the channel hot electron injection (CHEI). The electrons on the channel are pass through the isolator, and the charges are stored on the floating gate. And then, verifying the floating gate voltage is carried out by

measuring the drain-source current I_{REF} with applying the voltage to store V_M to the control gate. The charge injection and verification of the floating gate voltage are continued, until the floating gate voltage becomes the constant voltage V_{REF}.

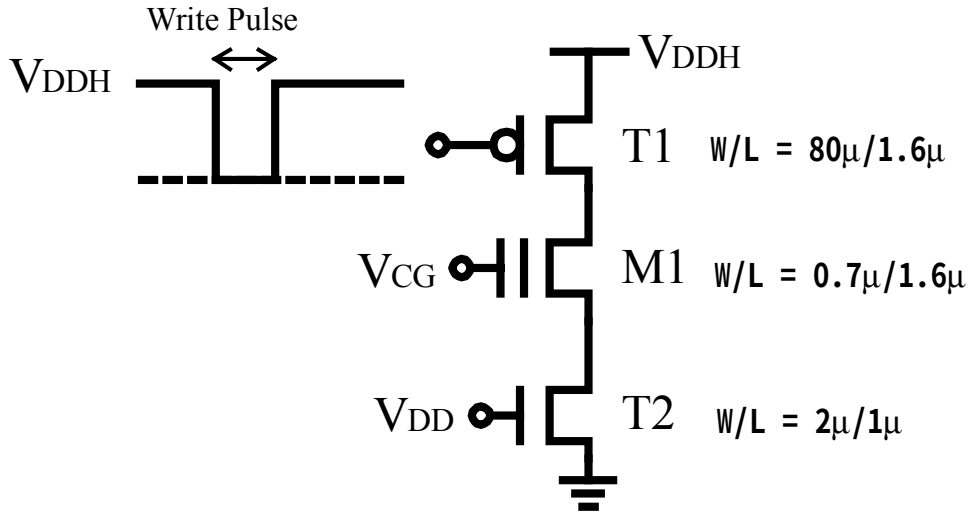


Figure 3.5. Memory writing circuit.

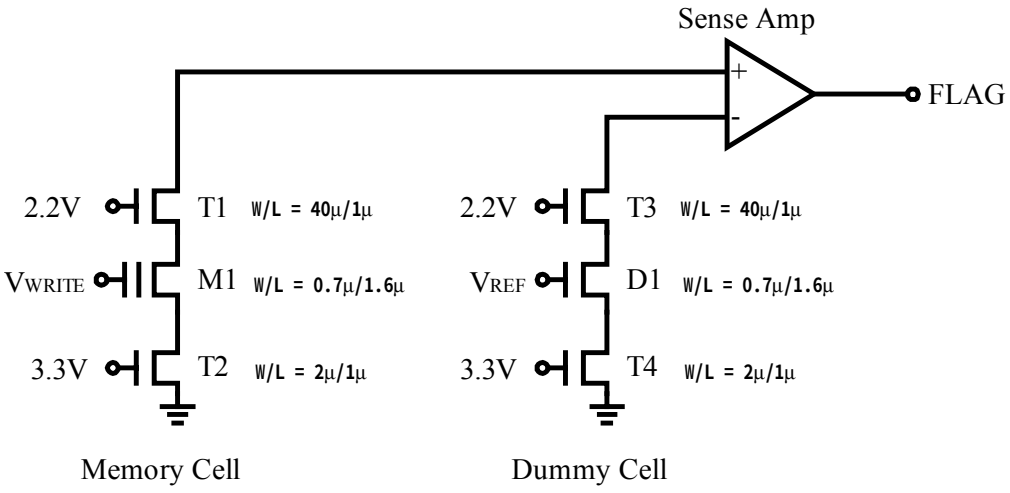


Figure 3.6. Verifying circuit.

III. Results and Discussions

A. Simulation Result of Matching Circuit

Figure 3.7 shows the HSPICE simulation result of the matching circuit. The constant voltages (1.0V, 0.6V, 1.25V, 1.9V, and 2.5V) are memorized into the matching cell, and then the input voltage *Xi* is swept from 0V to 2.5V. Output voltages of the source-follower V_{out} have the desired characteristics of absolute-difference as shown in the figure.

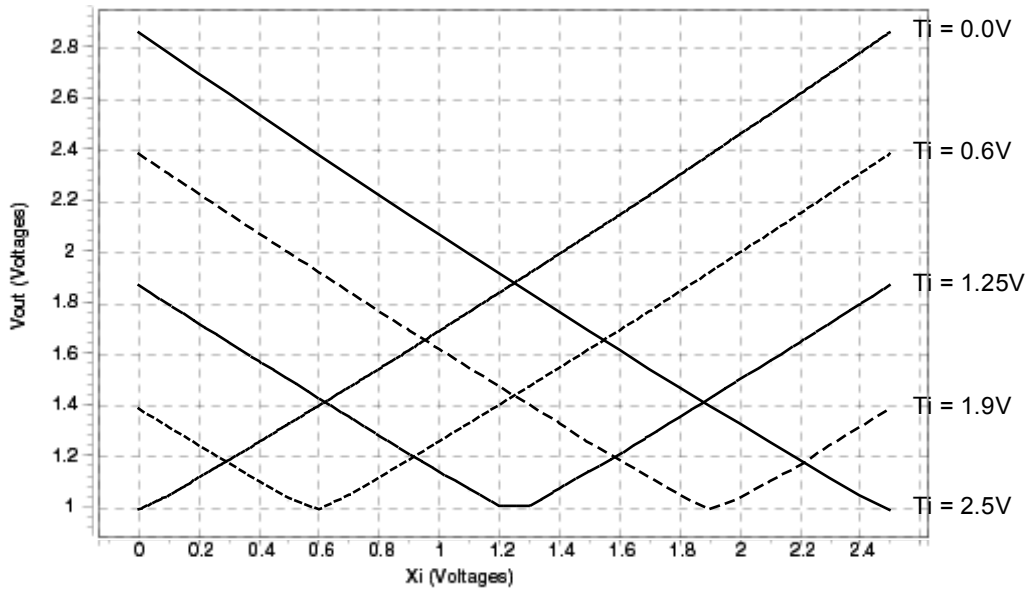


Figure 3.7. HSPICE Simulation result of matching circuit.

B. Prototype Chip

Figure 3.8 shows the chip photo of the prototype processor. The processor has been designed and fabricated with 0.7μm double-poly 1-metal CMOS process. The processor occupies the area of 7.35mm × 4.43mm. The processor is configured for 64-element × 256 template-vectors. The processor has sample-and-hold circuits for the analog input voltages, and a binary-search-type winner-take-all circuit, and the sense amplifier for memory writing.

Table 3.1 summarizes specifications of the processor.

Figure 3.9 shows the layout of the memory array. It includes 32 memory cells. Lines applied to the control gates are layed out in vertical direction, and the results of the absolute differences are taken out to left or right sides. The coupling capacitors for the averaging the absolute differences are layed out in the periphery of the memory array.

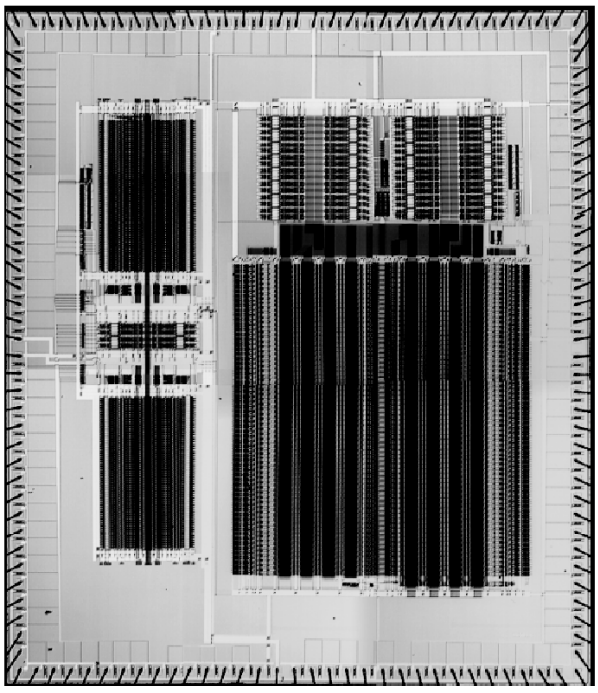


Figure 3.8. Chip photomicrograph of prototype processor.

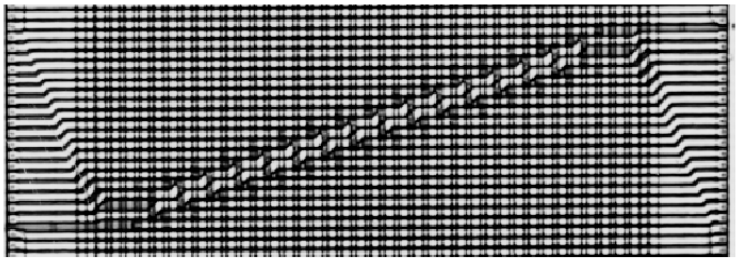


Figure 3.9. Layout of memory cell array.

Technology	0.7μm Double-Poly 1 Metal
Size of Processor Core	7.35mm × 4.43mm
Power Supply	3.3V
The number of template vectors	256
The number of vector element	64

Table 3.1. Specifications of VQ processor.

C. Analog Memory Characteristics

In the matching processor, the memory transistors are utilized as a functional logic, and its circuit configuration is not normal usage as a memory. In this respect, the characteristics of the memory writing should be carefully verified. Figure 3.10 shows the characteristics of the memory writing. The transition of the memorized voltage for each writing pulse is measured as a function of the memorized voltage. The writing pulse has width of 1ms, and the voltage of the pulse is set to 7V. During the memory writing, control gate voltage is set to 10V. The reference voltage for the reference current is set to 1.75V. The memorized voltage is read out by

searching for the control gate voltage when the output of the sense amplifier transits, namely the drain-source current of the memory transistor exceeds the reference current I_{REF}. The results show that writing efficiency is varied by the read out voltage, namely the floating gate voltage. It is because that efficiency of hot electron charge injection depends on the channel current, i.e. the floating gate voltage. When the read out voltage is low, the efficiency becomes quite small, and it results that a long time is required for writing. However, by adjusting the control gate voltage at the writing to optimum voltage, the efficiency of the writing can be improved. Writing precision is also determined by the voltage transition for a single writing pulse. The maximum transition voltage was 5mV, when the read out voltage is about 5V. If higher precision than 5mV is required, it can be achieved by using shorter length of writing pulse signal than 1ms or adjusting the control gates voltage to the one where the writing efficiency is smaller.

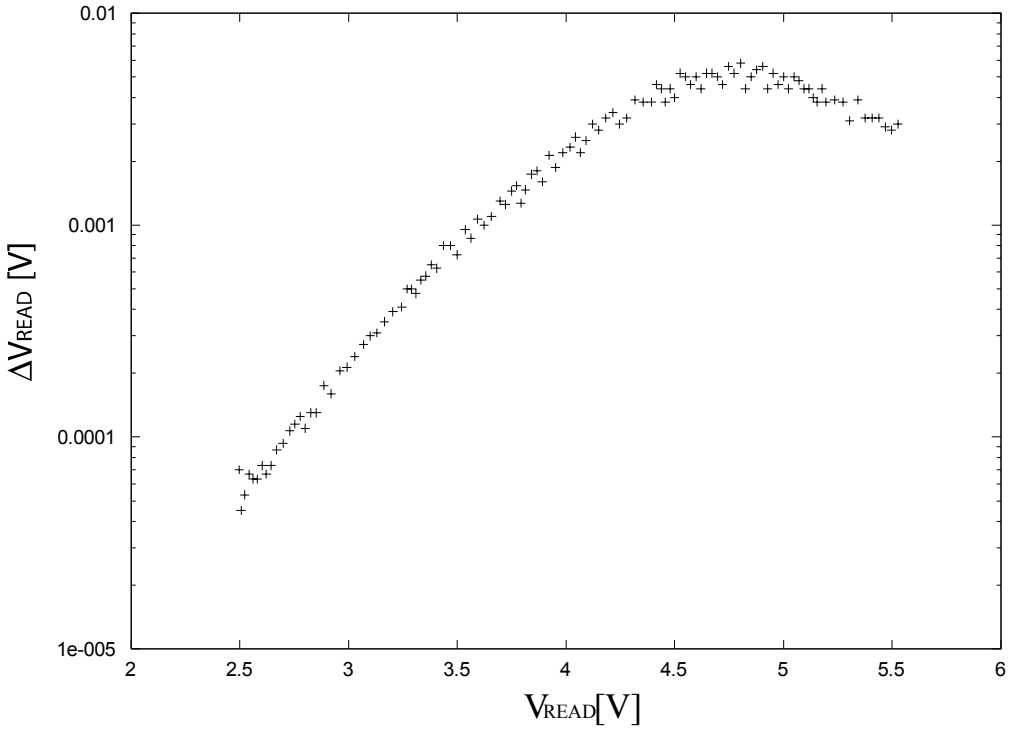


Figure 3.10. Measured memory writing efficiency.

In the array configuration of the memory, unselected memory data is disturbed during writing. The range of memorized voltage should be determined not to occur the disturbance. Figure 3.11 shows the principles of disturbance caused at writing in the memory array. The memory cell to write is selected by applying the high voltage to its control gate and its drain terminals. For the unselected word-line, 0V is applied to the control gate. For the unselected bit-lines, the drain and the source terminals of the cell are pull down to the 0V to prevent current flow. However, when the voltage difference between the floating gate and the drain becomes large, Fouler-Nordheim (FN) tunneling phenomenon occurs. Thus, unselected cells belonging to the same bit-line or the same word-line are suffered from disturbance in certain conditions.

At the memory cell belonging to the same word-line, the floating gate voltage becomes high, especially when the memorized voltage V_M is low. On the other hand, the drain or the source voltage or unselected cell is 0, thus the voltage difference between the floating gate and the drain or source should be low enough not to occur FN tunneling. The gate source voltage V_{GS} is given as follows,

$$V_{GS} = \gamma(V_{CG} - V_M) + V_{REF} \qquad \text{(Eq. 3.11)}$$

Here, V_{CG} represents the control gate voltage, which is applied to the selected word-line. And γ = 0.65 is utilized, which is determined by the memory cell structure. As the speed of FN tunneling increases exponentially against the difference between the voltage and the FN-tunneling starting voltage, the floating gate voltage caused tunneling converges to the starting voltage after a long time. In the experiments, for 1 minute, a high voltage is applied to the control gate. From the experiments, when the memorized voltage V_M is 2.5V, the disturbance occurred from the control voltage V_{CG} = 12V, thus the condition starting FN tunneling can be conducted from Eq. 3.11

as V_{GS} = 8V. When the control gate voltage is fixed to 10V, the minimum memorized voltage V_M is determined from Eq. 3.11 as 0.4V, which is smaller than 2.5V of the read out voltage from fully erased memory cell.

At the memory cell belonging to the same bit-line, the floating gate voltage of unselected cell becomes very low, while the drain voltage is pull up to high voltage, thus the FN tunneling occurs near the gate-drain isolator. The voltage V_{GD} between floating gate and drain is given as follows,

$$V_{GD} = -\gamma V_M + V_{REF} - V_{DDH} \qquad \text{(Eq. 3.12)}$$

The disturbance occurs when the memorized voltage is large, thus, a large voltage was memorized before the experiments. When V_{DDH} = 6V was applied, the memorized data converges to 5.8V. From Eq. 3.12, the starting voltage of FN tunneling is conducted as V_{GD} = -8V. In the normal write operation, the V_{DDH} is set to 7V, thus the maximum range of the memorized voltage becomes 4.2V from Eq. 3.12.

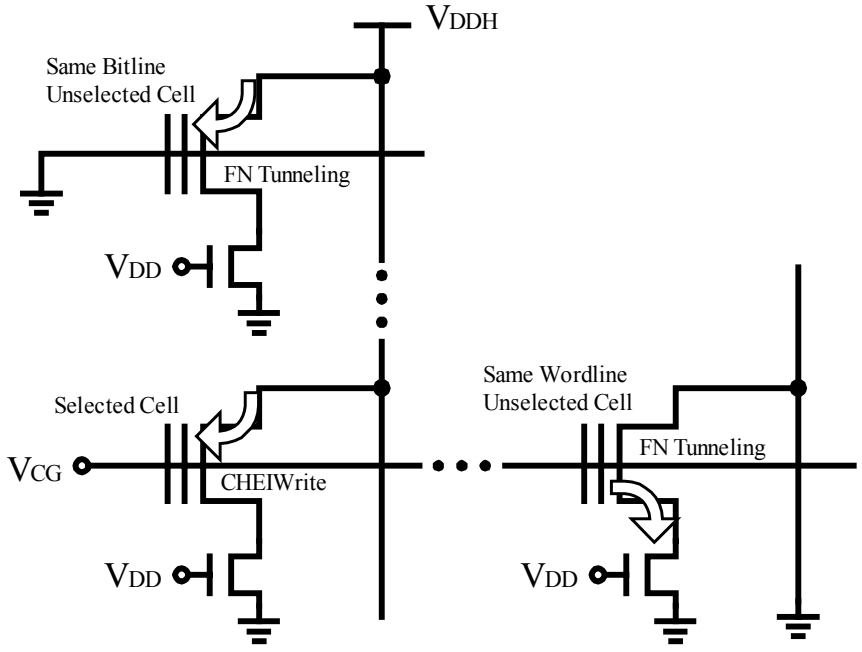


Figure 3.11. Unselected memory data disturbance caused by writing.

IV. Conclusions

An analog matching processor featuring non-volatile analog-memory-merged matching circuit has been developed. The memory-merged matching circuit stores the template vector data and computes the distance in its circuitry itself. As a result, the high-density and highly parallel circuit implementation has been achieved. The processor has 256 64-element template vectors, and carries out fully parallel analog processing.

Writing analog data into the highly integrated functional memory is verified. The write-and-verify scheme using hot channel electron injection enables us to write analog data more precisely than 5mV, and the range of the memorized voltage from 3V to 4.2V has been conducted from experiments.

The functional non-volatile analog-memory-merged matching circuit developed in this work would contribute to intelligent data processing applications requiring low-cost and high-power-efficiency implementation.

CHAPTER 4.
NMOS-based Bell-shape Matching Cell

I. Introduction

Similarity evaluation between an input vector and a large number of template vectors stored in the database is computationally expensive, and its efficient execution is a key issue in VQ processor. Since digital implementation requires a large hardware volume, analog implementations [17-21] are preferred in certain classes of applications, for instance in mobile applications where both memory and computational resources are very limited.

In this work, an analog similarity-evaluation circuit compatible to high-density integration has been developed. Each matching cell consists of four NMOS' for computing bell-shape element-similarity, two capacitors for template data storage, and two NMOS switches. This pure NMOS

configuration is superior to CMOS cells [17, 27, 28] in terms of the chip real estate, because well regions can be excluded form high-density cell arrays. In addition, our correlation circuit can modulate the weight factor dynamically for each element of a vector, which provides flexible preference-based search. Moreover, a compact cyclic digital-to-analog converter (DAC) has been developed for providing easy interfacing to digital systems. It also allows us to generate various analog control signals for bell-shape similarity-evaluation all in simple digital circuits. The on-chip DAC is composed of two unity-gain buffers employing a single inverter and switched capacitors.

Test circuits were fabricated in a 0.6-μm double-polysilicon triple-metal CMOS process, and the circuit ideas have been experimentally demonstrated.

II. System Architecture

Figure 4.1 shows the architecture of an analog VQ processor. It consists of the DAC array for digital data inputs, the two-dimensional array of matching cells each evaluating a bell-shape similarity between the input vector element and the template vector element, and the comparator tree for coding the location of the maximum-similarity template vector. Each element row is equipped with two DAC's for complementary analog data generation and supplying them to the matching cell array. The matching result in each template vector is obtained by taking the wired sum of current outputs from all elements of the vector. In this manner the chip has a digital-in and digital-out specification, and internal computation is carried out in an efficient analog-domain.

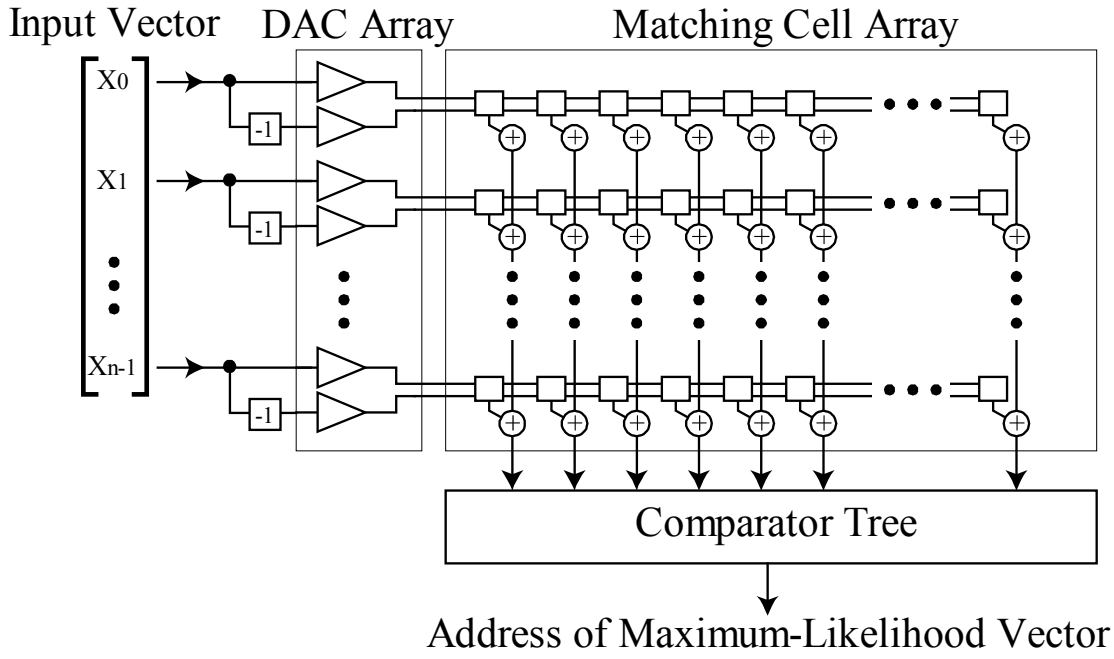


Figure 4.1. System architecture of analog matching processor.

III. Circuit Configurations

A. Matching Cell

Figure 4.2 shows the bell-shape matching cell developed in this study. It is basically a simplification and modification of the Delbruck's bump circuit [22]. The current source and the current mirrors in the bump circuit were eliminated and our cell is composed of only NMOS transistors. This allows us to reduce the layout area. Instead of using the differential amplifier configuration, the difference signal is produced by charge sharing using a switched capacitor technique. Namely, the difference between the template and input data is produced at the floating gate. This allows us to replace the NMOS with an analog flash memory transistor, realizing a very compact memory merged cell. A non-volatile analog-memory technology is being developed in our lab. for this application in mind [55]. The bell-shape

characteristics is produced by two complementary analog signals: V_m and $V_{bias} - V_m$. The four NMOS' are connected by cross coupling to remove the asymmetry in the bell-shape characteristics. The operation of the matching cell is illustrated in Figure 4.3.

For storing the template data in the matching cell, V_m and $V_{bias} - V_m$ are given to respective input terminals, while resetting the floating gate potential to V_{RST}. For calculating the similarity between V_m and the input V_{in}, V_{in} and $V_{bias} - V_{in}$ are similarly given to the gates after disconnecting the floating gates by turning off the switch NMOS' T5 and T6.

The element matching result is given as an output current V_{out} and the result of multi-dimensional vector matching is given by taking the wired sum of currents from all elements of the vector. The array configuration of matching cell enables the massively parallel computation.

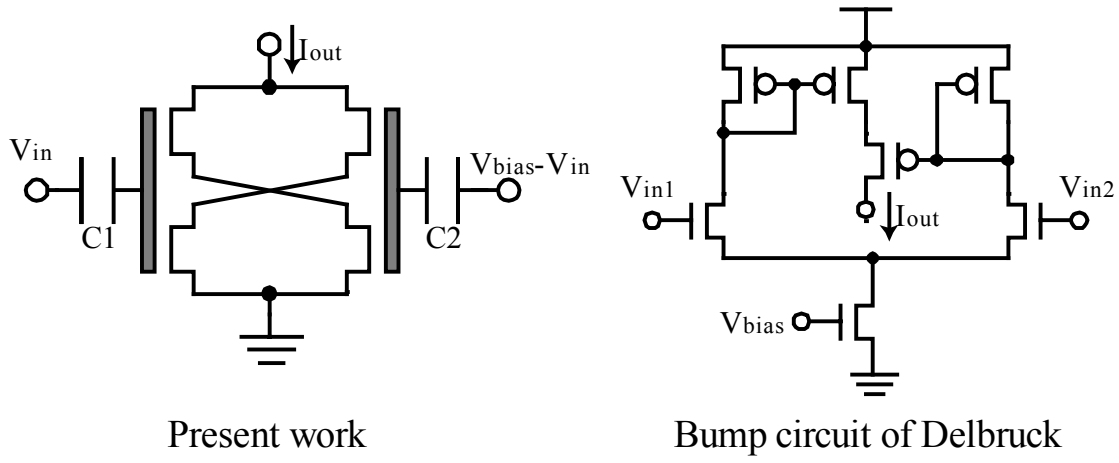


Figure 4.2. Schematic of bell-shape element matching cell.

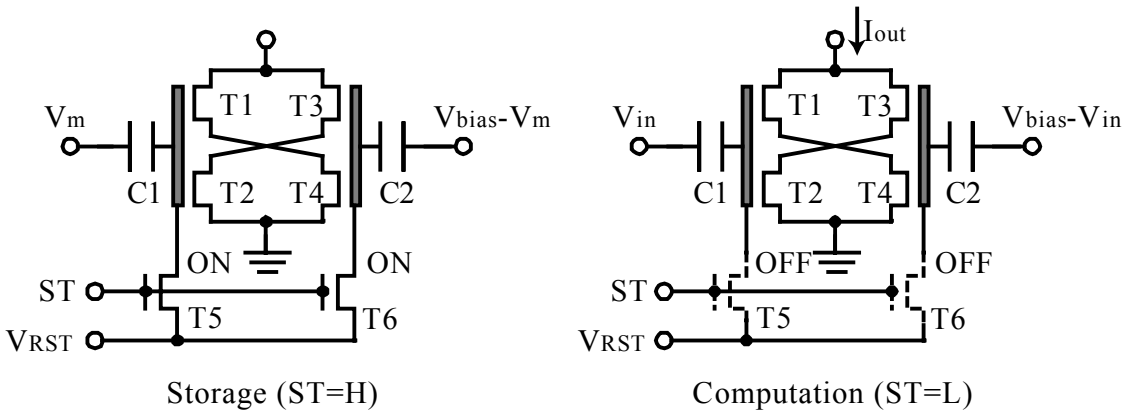


Figure 4.3. Operation of matching cell.

B. Cyclic DA Converter

On-chip DAC converters provide such advantages as easy communication with external digital processors, flexible manipulation of input data like weighting or masking etc. The complementary analog signal inputs to the matching cell are also very easily produced by using the on-chip DAC with bit inversion data.

Since an associative memory requires a number of DAC's for element-parallel processing, the on-chip DAC needs to meet the requirement of small layout area and low power dissipation. The cyclic DAC is one of the efficient schemes recycling the circuit, and it has been employed in applications, which require a lot of DAC in the processor [56, 57]. The cyclic DAC algorithm [56] is based on the recurrence formula given below, and the conversion is carried out by applying the formula cyclically to a bit-serial digital data input.

$$A_i = \frac{1}{2}\left(D_i + A_{i-1}\right) \qquad \text{(Eq. 4.1)}$$

The cyclic DAC requires a high-precision unity-gain buffer, which is usually implemented as a voltage follower of a high DC-gain op-amp, but it is not favorable in terms of area and power dissipation due to a lot of current

sources and gain-boosting stages. For these reason, a unity-gain buffer with CMOS inverter, the simplest amplifier, as its gain stage is employed in this study. Since a gain of a CMOS inverter is finite, and characteristics of unity-gain buffer is not good as shown in Figure 4.4. The conventional unity-gain inverter buffer employs the multi-stage inverter to obtain large DC-gain as shown in Figure 4.5. This configuration consumes large power-dissipation due to the short-circuit current of inverters, and requires sensitive design-parameter determination to depress a oscillation because of feedback loop.

Figure 4.6 shows the architecture of the simple double-reset unity-gain inverter buffer that has been developed in this work. It consists of a CMOS inverter and two switched capacitors. Its operation is illustrated in Figure 4.7. At first, pre-reset is done by shorting the inverter output and the inverter input N1 while applying an input voltage V_{in} to the capacitor terminal N2. Then, main reset is carried out in a similar manner by shorting the inverter output and the capacitor terminal N2 while applying V_{in} to the capacitor terminal N3. For reading, the inverter output is fedback to the capacitor terminal N3.

This double-reset operation has significantly improved the buffer characteristics of a CMOS inverter, which is inferior to that of an op-amp voltage-follower due to the finite gain of the inverter. Schematic of the cyclic DAC employing the double-reset unity-gain inverter buffer circuitry is shown in Figure 4.8.

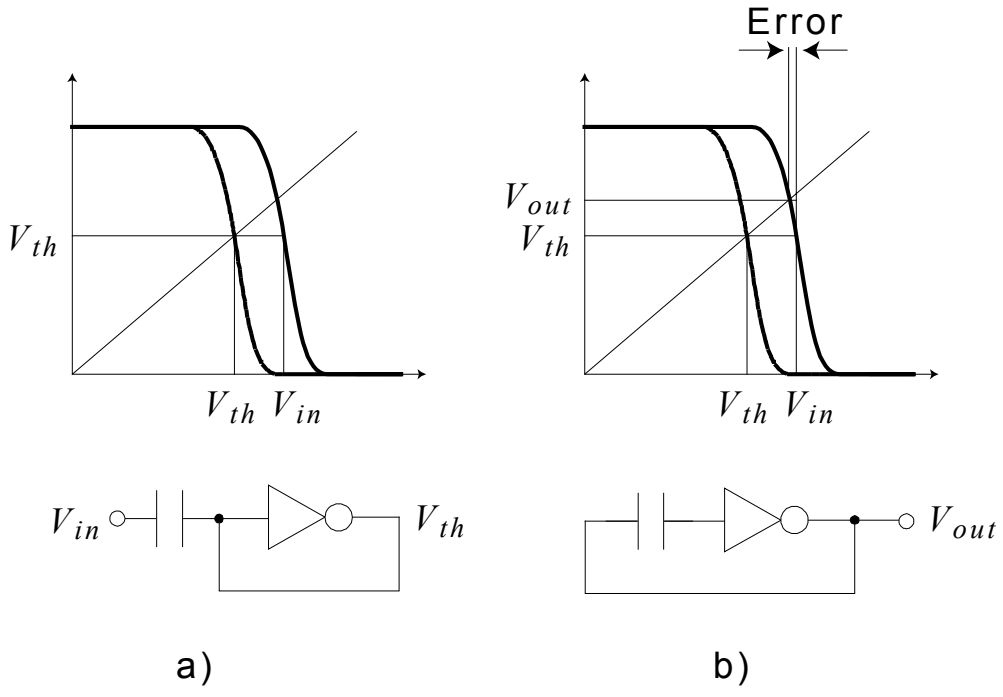


Figure 4.4. Error caused by finite DC-gain of amplifier.

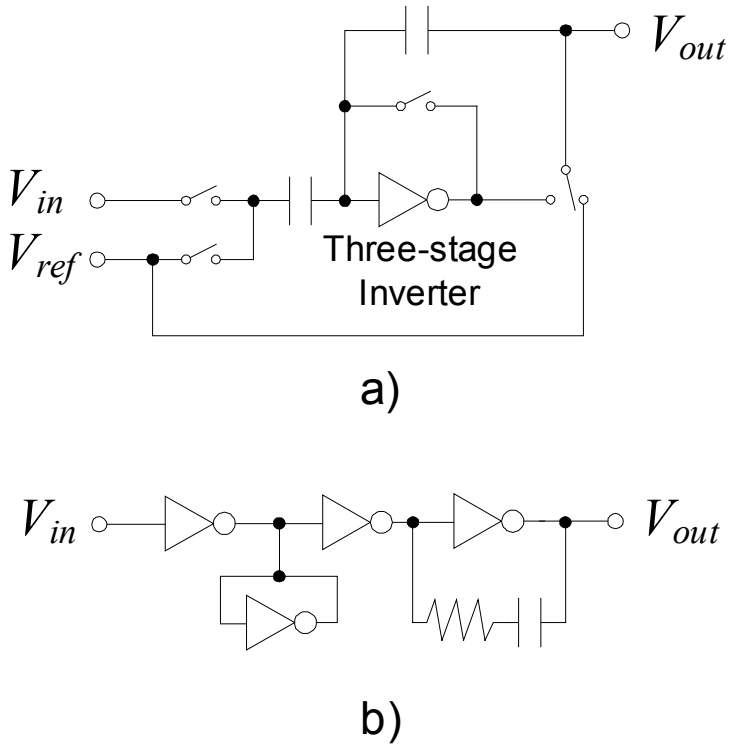


Figure 4.5. Conventional inverter-based sample-and-hold circuit.

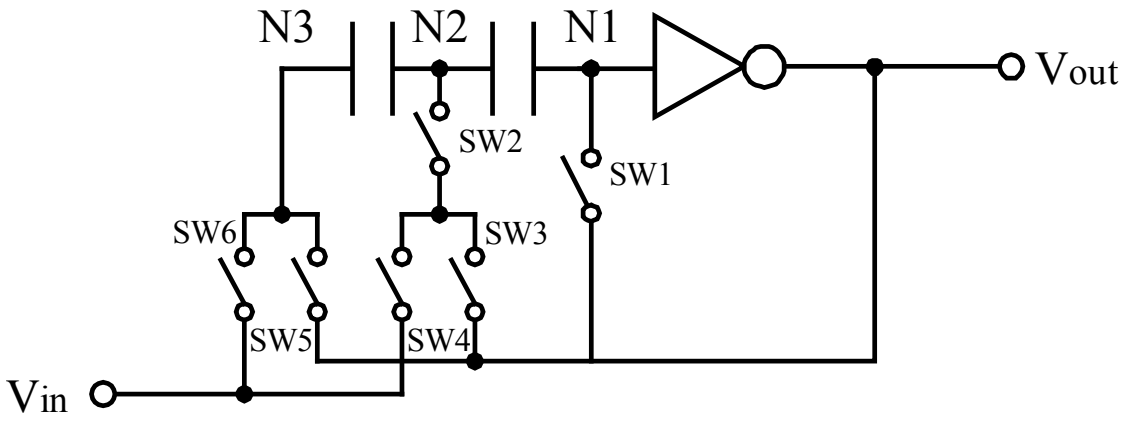


Figure 4.6. Circuit configuration of unity-gain CMOS inverter buffer.

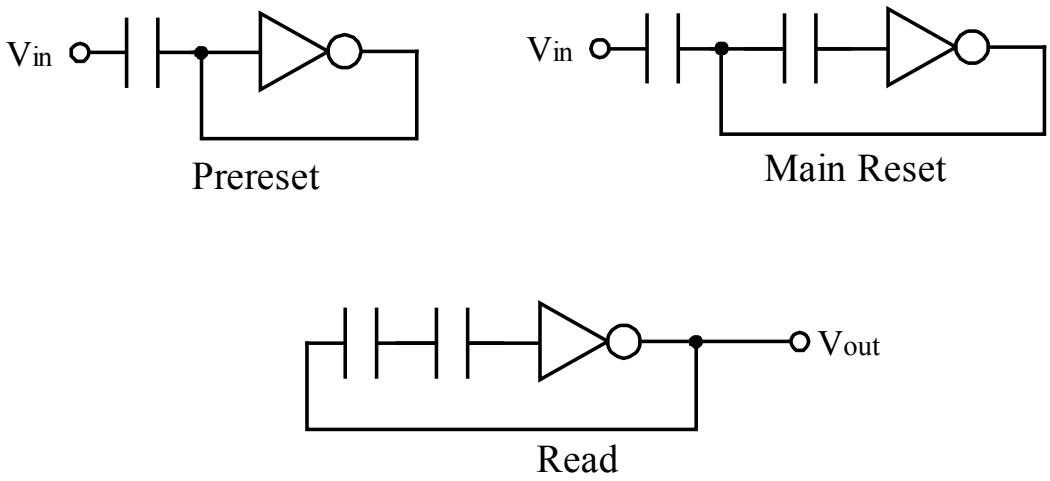


Figure 4.7. Operations of unity-gain CMOS inverter buffer.

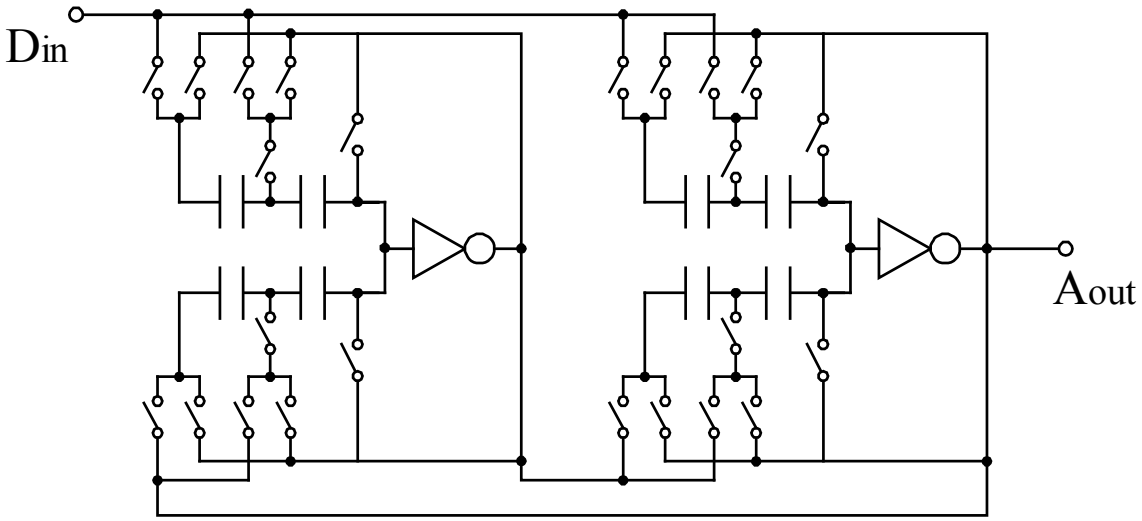


Figure 4.8. Circuit configuration of cyclic DA converter employing unity-gain CMOS inverter buffer.

IV. Experimental Results

Test chips were designed and fabricated in a 0.6-μm double-polysilicon triple-metal CMOS process. Figure 4.9 shows a photomicrograph of the matching cell and DAC array. The size of matching cell is 26mm × 43mm and that of a DAC 530μm × 22μm. The CMOS type bell-shape matching cell presented in Ref. [27, 28] is larger than the present cell due to the n-well regions for PMOS' in the matching cell array. This is the advantage of pure NMOS configuration. However, the present matching cell size is still large due to the large area required for capacitor layout. This is due to the relatively small capacitance value available in the foundry process.

Since our cell is compatible to direct replacement of NMOS' by analog flash memory transistors described in Ref. [55] as shown in Figure 4.10, the layout of a flash-memory-merged cell was designed assuming the same 0.6-μm layout rules. The results are shown in Figure 4.11(a). The impact of merging analog flash technology [55] to matching cell circuitry is significant.

Compared with the cell array developed in the previous work (described in Chapter 3), the NMOS-based bell-shape cell has 4 times of cells in the identical layout area.

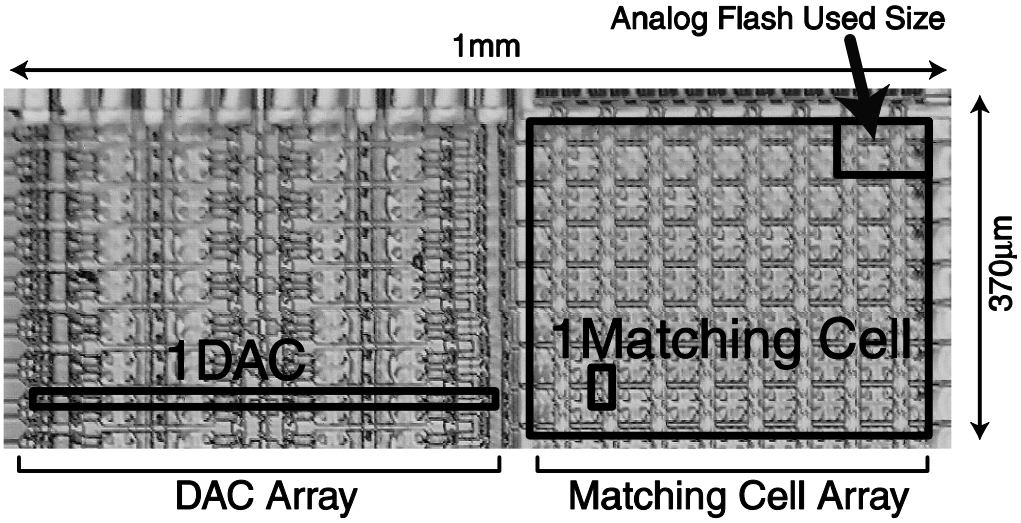


Figure 4.9. Photomicrograph of DAC array and matching cell array.

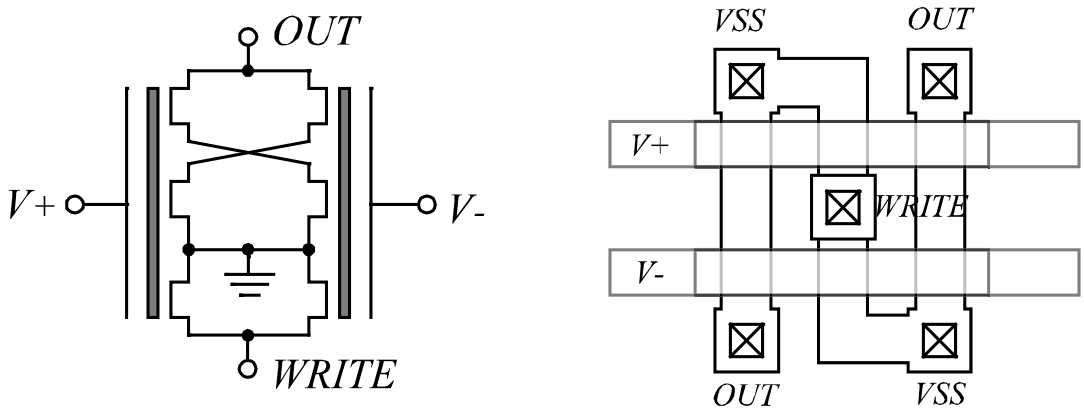
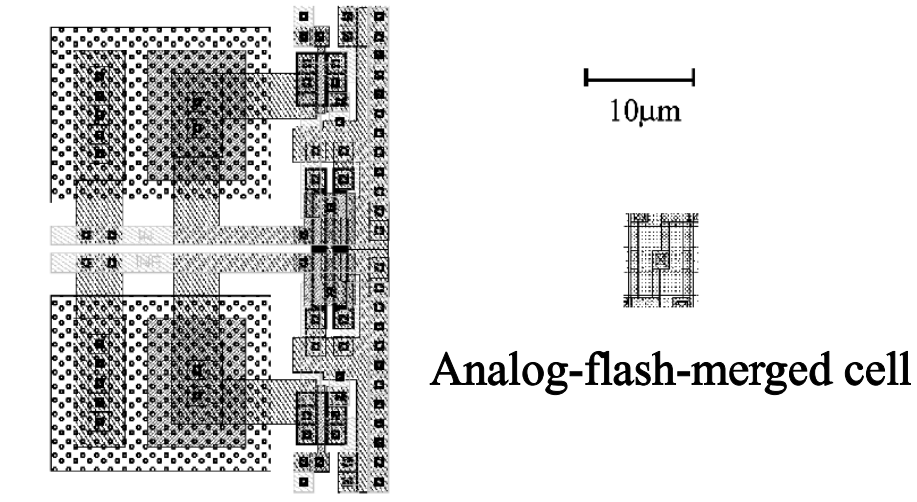


Figure 4.10. Analog-memory-merged matching cell.



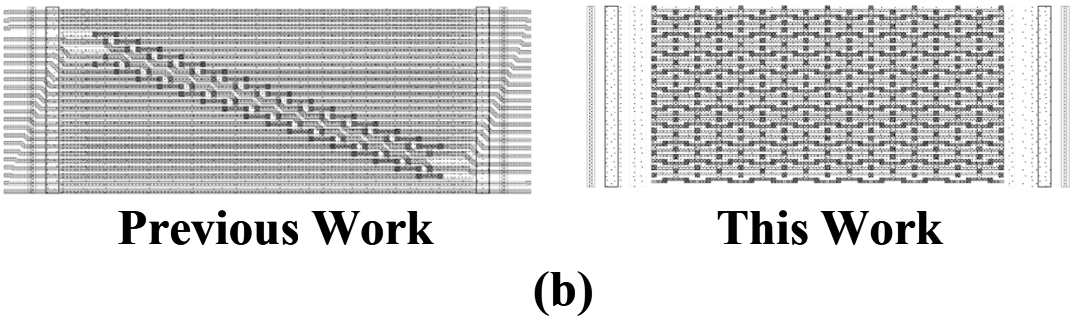


Figure 4.11. (a) Layout comparison of NMOS-based bell-shape cell. (b) Layout comparison between analog-flash-merged cells. In the cell array of the previous work (described in Chapter 3) has 32 cells, and in the cell array of this work has 128 cells in the identical layout size.

Figure 4.12 shows the measured characteristics of a matching cell. Template value was varied as 0.25V, 0.5V, and 0.75V. The difference between the stored data and the measured peak position is within 25mV at a 1V input range. In Figure 4.13, the height variation by biasing control are demonstrated.

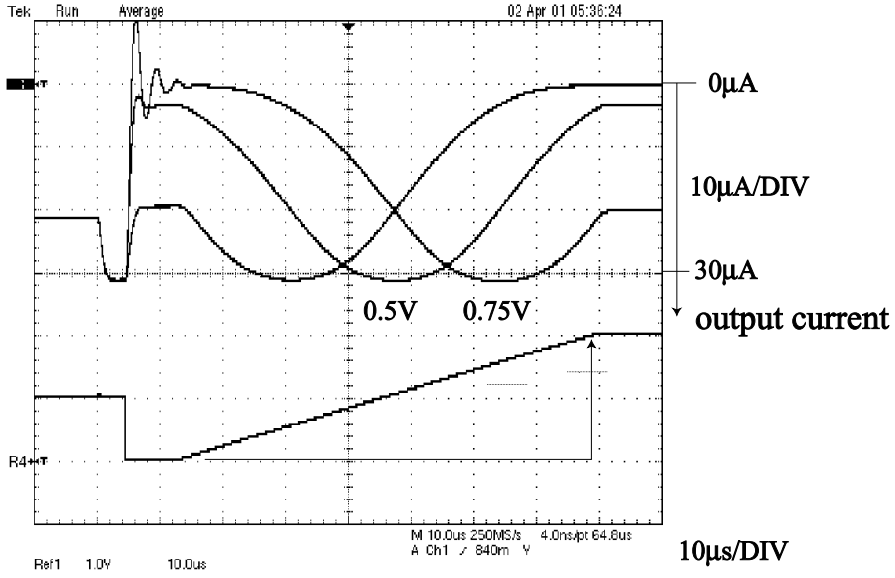


Figure 4.12. Measured matching cell characteristics (peak position variation).

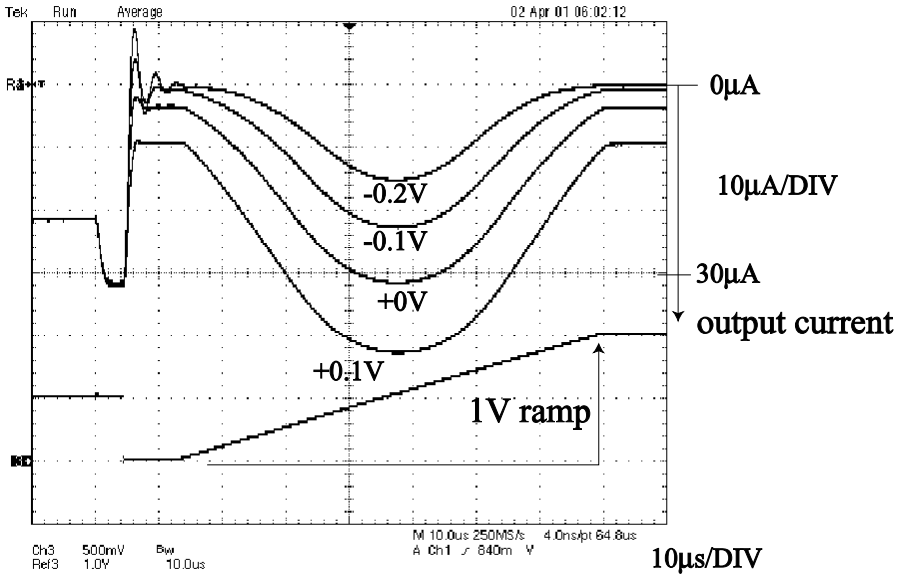


Figure 4.13. Measured matching cell characteristics (peak height variation).

Figure 4.14 shows the simulated results of unity-gain CMOS inverter buffer characteristics. Error due to the finite gain of CMOS inverter is reduced by the double-reset scheme compared with the conventional single-reset scheme.

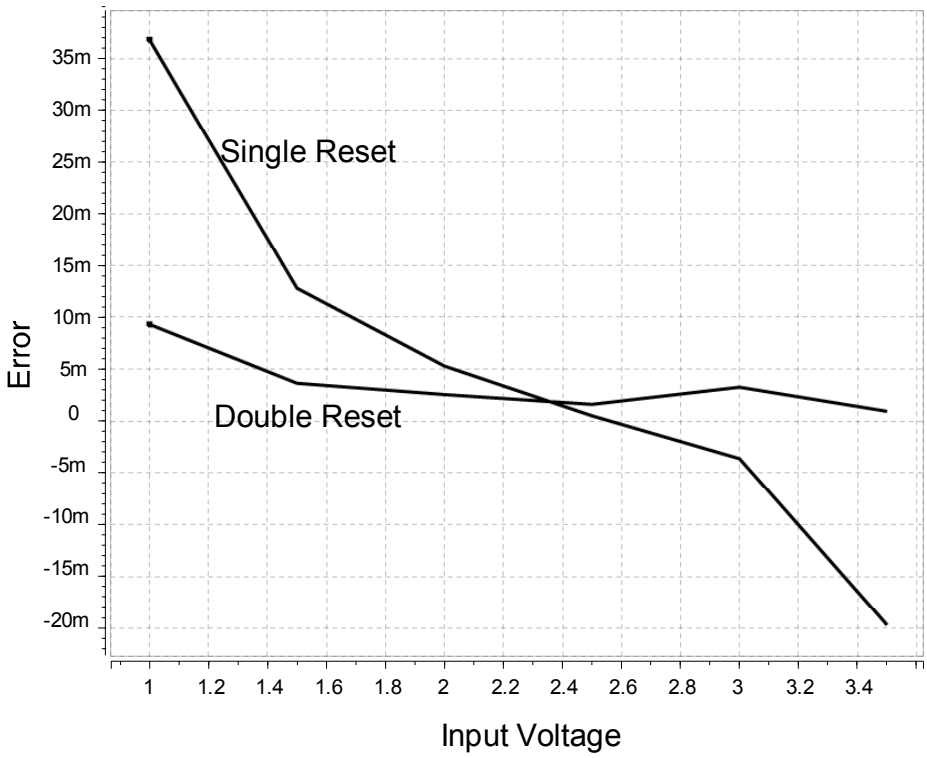


Figure 4.14. Simulated results of unity-gain buffer.

Figure 4.15 shows the measured waveform of the DAC output, which employs the double-reset scheme. In Figure 4.16, the DAC output voltage read from the oscilloscope display was plotted on post-layout HSPICE simulation results. Some nonlinearity is observed in both simulation and measured data. This is due to the capacitance unbalance caused by parasitic capacitances. However, this would not lead to a serious error in the matching because both the template and input are produced using the same DAC unit. However, more careful artwork design is essential to achieve a high-precision processing.

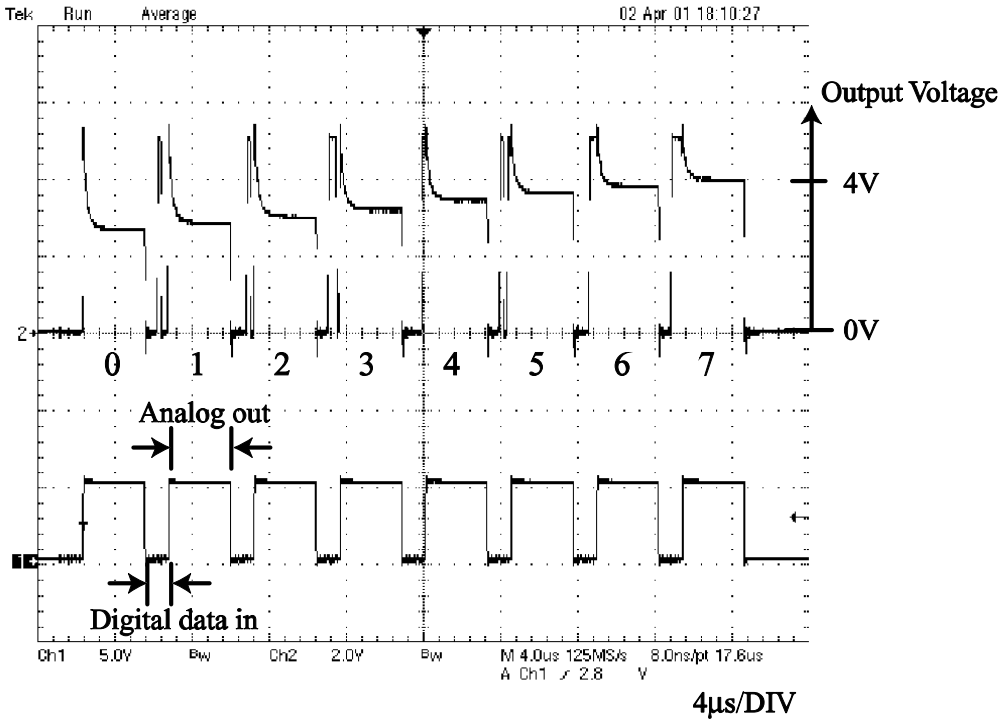


Figure 4.15. Measured waveforms of DA converter.

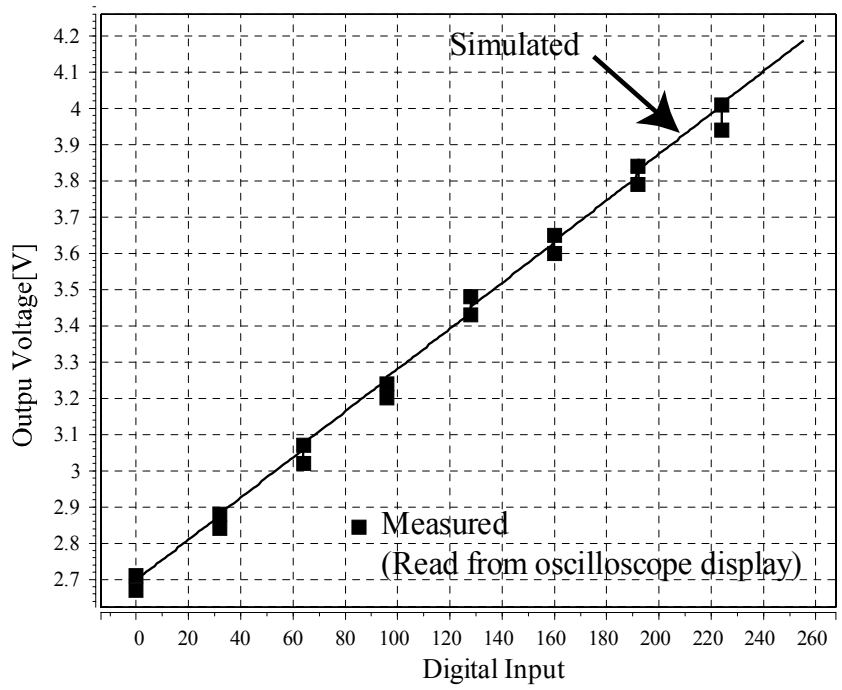


Figure 4.16. DAC output characteristics.

Demonstration of a simple pattern matching with four-dimensional vector is shown in Figure 4.17. The maximum-similarity template vector was identified by the two-stage comparator-tree. The latch enable signal and the output of comparator's decision flag at each stage are shown in the Figure 4.17.

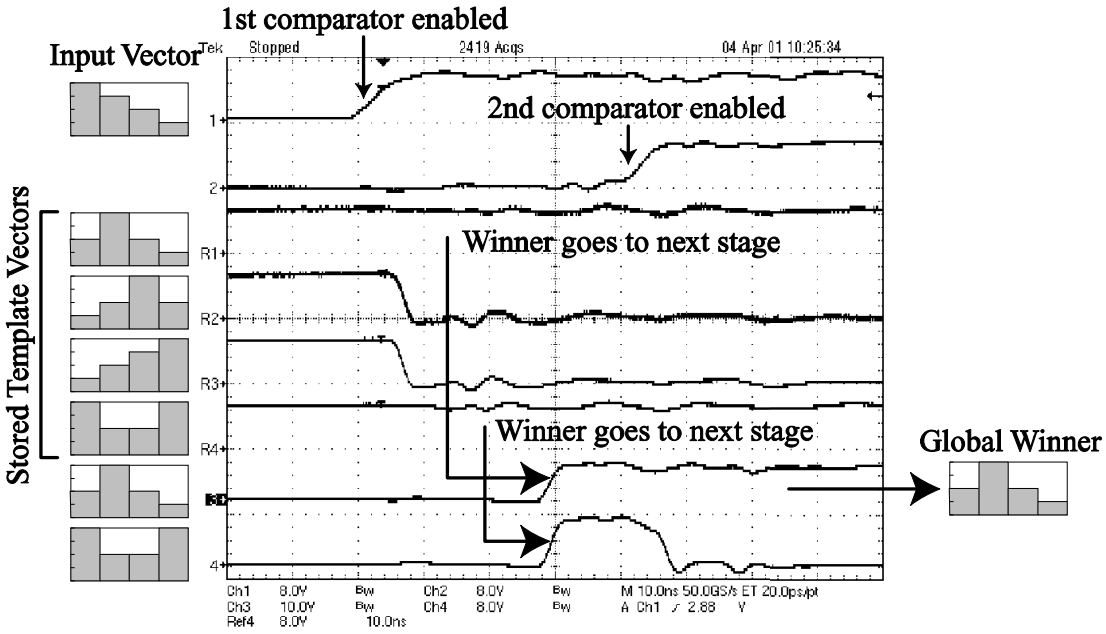


Figure 4.17. Result of template matching demonstration.

V. Conclusions

An analog VQ processor technology has been developed based on a bell-shape-type element matching circuit aiming at high-density integration of matching cells. The matching cell is composed of six NMOS transistors and two capacitors. The pure NMOS configuration is superior to CMOS-based cell in terms of cell area because well regions can be eliminated

from the high-density cell array. The current cell size is 26mm × 43mm in 0.6-μm layout rules. However, the size can be reduced by a factor of about 1/25 if the NMOS' are replaced by analog flash memory transistors. Our cell structure has been designed to be compatible to such replacement. In addition, an analog flash technology is being developed in our lab. [55] aiming at such applications.

Furthermore, a novel unity-gain buffer employing CMOS inverter is developed for compact cyclic DAC aiming at highly parallel conversion. To improve the buffer characteristics, the double-resetting scheme is employed, which eliminates the error due to the finite DC-gain of CMOS inverter.

The analog VQ processor developed in this work would contribute to low-cost intelligent data processing applications.

CHAPTER 5.
General-Purpose Digital VQ Processor

I. Introduction

Vector quantization (VQ) is known as a powerful tool for such purposes and has been utilized in a variety of applications in data compression, pattern classification and intelligent data processing [16]. However, due to its high computational cost, the development of hardware accelerators for the processing is a vital area of VLSI research.

To this end, various digital VQ processors have been developed [37-57]. In the processors in Ref. [47, 48], flexible operation in the distance computation unit was realized by functional-memory-type parallel processor, while at the expense of a reduced speed performance due to the increased clock cycles for a single VQ operation of 165 (WTA takes 53 clock cycles). The digital VQ processors developed for image compression [49-51] achieved a high

throughput by optimizing the pipeline configuration of the bit-sliced WTA. As a result, the number of clock cycles necessary for a single VQ operation was reduced to 19. However, the data flow in the pipeline is fixed by hardwiring specifically tuned for the image compression application, and it is not possible to alter the function flexibly for other applications. Namely, the vector dimension is fixed to 16, it is not possible to change the form of distance function (only Manhattan distance available), and other winner search options like local winner search, second and third winners search etc. cannot be carried out on the chip.

The purpose of this work is to develop a general-purpose VQ processor, aiming at enhancing the processing flexibility without degrading the speed performance. In this work, a two-dimensional bit-propagating (2DBP) WTA has been developed to minimize the delay or clock latency in the winner search operation. In the 2DBP WTA circuit, the signal propagation paths in the circuit have been optimized in order to complete a WTA operation within a single clock cycle. As a result, a much higher speed performance of the WTA has been achieved as compared to the bit-sliced WTA circuits in [49-51]. In addition, the variable-binary-block addressing scheme has been developed to allow arbitrary masking on the distance inputs to the WTA circuitry with a minimal hardware overhead. The combination of high-speed WTA and arbitrary WTA-inputs masking enables a variety of winner search options, like local winner search or winner sorting etc. Furthermore, the multiplier function is embedded in each SIMD distance-computation unit with a minimal area penalty of only 18% to enable weight multiplication to vector elements. The weight multiplication enables more complicated distance function than the simple Manhattan distance, such as Euclidean distance. It is also particularly important in performing preference-based search as described in Refs. [11, 12].

The chip was fully designed by hand-layout using Cadence tool (icfb) to minimize the chip real estate except for the controller, which was designed

with verilogHDL. The chip was fabricated in a 0.6μm 3-metal CMOS process, and the concept was experimentally verified. In addition, the highly parallel configuration using 0.18-μm 5-metal CMOS process was experimentally designed.

The system organization of the general-purpose VQ processor is presented in Section II, and key circuits are described in Section III. Measurement results from the fabricated prototype chip and more discussions are shown in section IV, and finally conclusions are given in Section V.

II. System Organization

Figure 5.1(a) shows the system organization of the VQ processor. The processor consists of a controller, a SRAM memory, SIMD distance-computation units, a variable masking unit and an winner-take-all (WTA) circuit. The controller decodes instructions fed to the processor and controls other blocks.

The SRAM memory consists of four banks of 256-columns 256-rows SRAM, and utilized for template vector storage. It is directly connected to the SIMD distance-computation units with 256-bit (8bits 32-vector parallel) data bus in order to download template vector data with high bandwidth.

The processor has 32 parallel SIMD distance-computation units. Figure 5.1 (b) shows a data-path logic from the SRAM memory to the WTA circuit corresponding to a single distance-computation unit. Its processing element (PE) has arithmetic units, such as an absolute-difference (AbsDif), a shifter (Shift) and an accumulator (Acc). The PE supports multiplication in distance computation by using the shifter and the accumulator as described in Section III-C. The PE has also four 24-bits distance registers (DR's), which store calculated distances. When more than 32 template vectors are required for matching, the PE computes multiple distances in serial manner, and distance registers store them. Thus computations for up-to 4 template

vectors at each PE or up-to 128 in total (32 PE's) can be carried out.

Calculated distances are fed to the WTA circuit via the variable masking unit. It has 128 masking elements (ME's), and they are placed between the distance register and the WTA circuit as in Figure 5.1(b). The ME is attached to each distance register, and independently masks distance data according to a value in a masking register. When not masked is specified, it transfers a similarity data (bit-inverted distance data) to an input of the WTA circuit. When masked, it transfers all '0' bit data, which will be ignored at the WTA circuit identifying the maximum values. Values in the masking registers are efficiently updated using a variable-binary block addressing scheme described in the Section III-B.

The WTA circuit identifies the maximum similarity value (equivalent to identify the minimum distance value) from all the inputs fed from the masking unit, and outputs its location code with the value. The WTA circuit processes 6 bits out of a similarity input data in a single clock cycle. For more than 6-bit inputs, an WTA operation is repeated during multiple clock cycles, for instance, 4 cycles required for 24-bit similarity data.

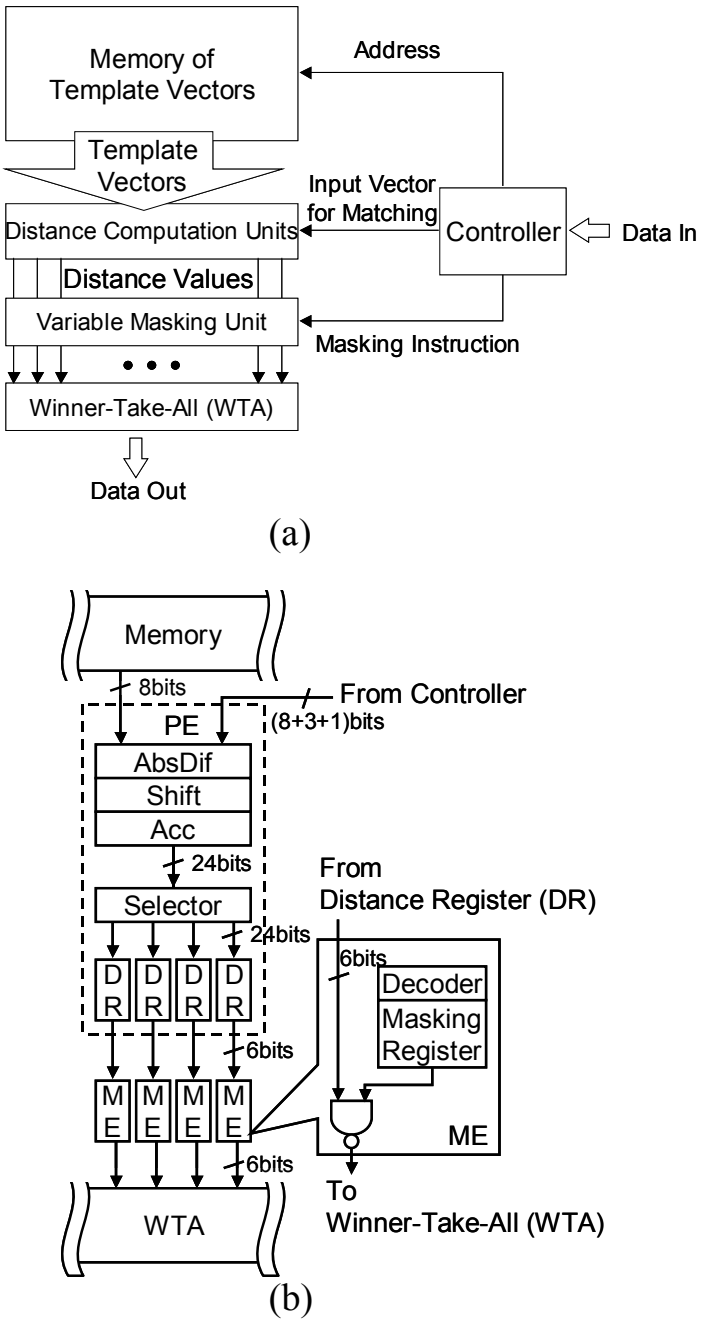


Figure 5.1. (a) System organization of VQ processor. (b) Block diagram of data-path corresponding to single distance computation unit.

III. Circuit Configurations

A. Two-Dimensional Bit-Propagating Winner-Take-All

The winner-take-all (WTA) circuit identifies the maximum value from a large number of inputs, and outputs its location code with the value. The two-dimensionally bit-propagating (2DBP) scheme developed in this work is illustrated in Figure 5.2(a), in comparison to the word-comparator (WC) based scheme in Figure 5.2(b). In order to clarify the advantage of our 2DBP scheme over the conventional WC-based approach, let us first explain briefly the WC-based approach. The WC-based WTA is composed of word comparators connected in the shape of tournament tree. At each tournament stage, the WC compares two input words in all the bits parallel, using carry look-ahead circuit in order to minimize the delay time for each tournament stage. After the word comparison, the maximum out of two input is transferred to the WC of the next tournament stage. Comparing words two by two at each tournament stage, the maximum of all the inputs is finally obtained at the final tournament stage. On the other hand, in 2DBP WTA, the circuit has the same tournament tree configuration except for a comparator circuit at each tournament stage. The comparator circuit in 2DBP scheme is composed of bit comparators connected in bit-serial manner instead of carry look-ahead parallel configuration. In the 2DBP scheme, the comparison is initiated from the most significant bit (MSB). After the bit comparison at the MSB finished, the next significant bit's comparator begins to compare, and the comparison result at the MSB comparator (in this case the MSB value of the maximum one) is simultaneously transferred to the MSB comparator of the next tournament stage. Namely the result data transfer is initiated as soon as each bit-comparison finished. In this manner, results of bit-comparators are propagating in two directions: a bit serial direction and a tournament ascending direction. As a result, the 2DBP scheme provides higher speed performance than the conventional WC-based

one. Details of the delay time of these schemes are discussed in the Section IV-A.

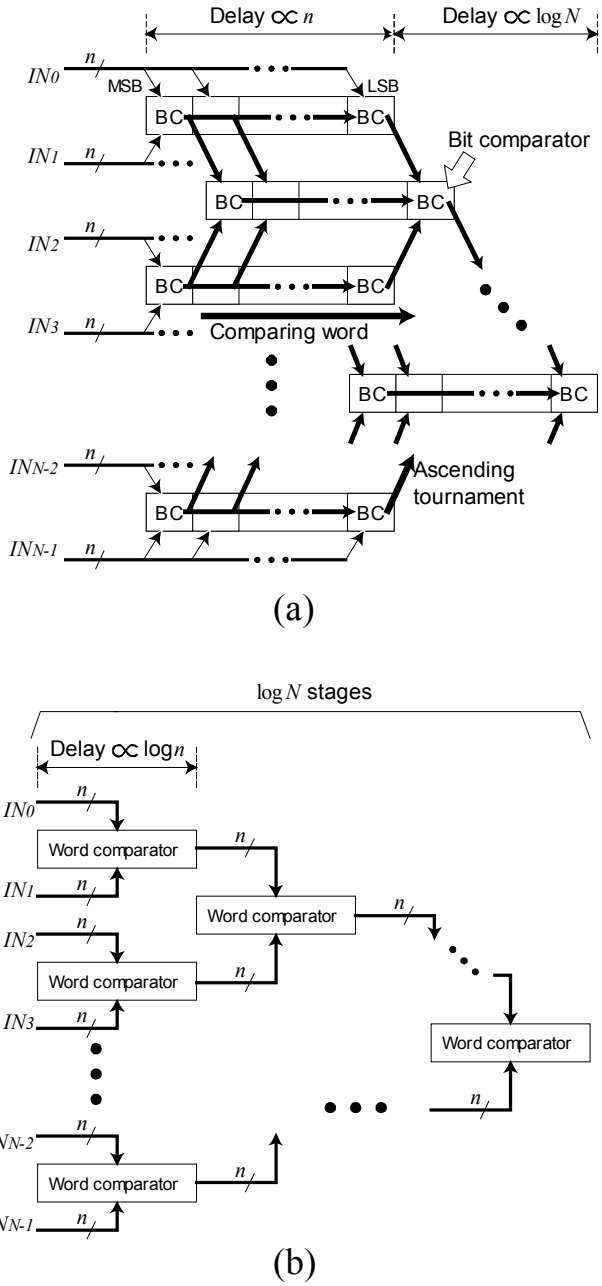
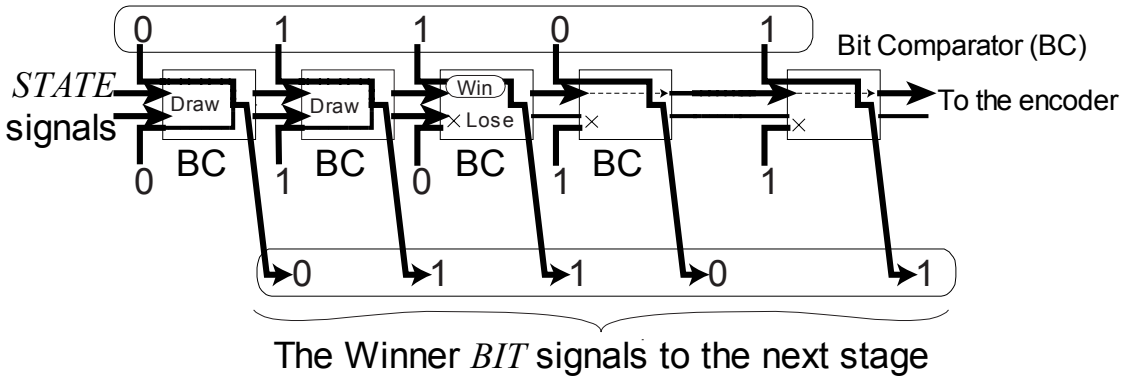


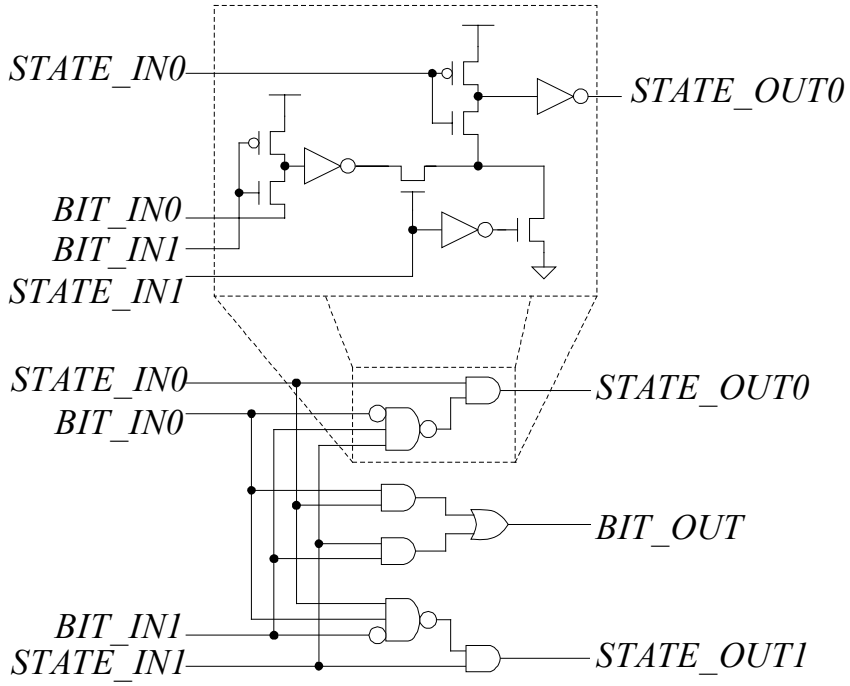
Figure 5.2. Organization of winner-take-all circuits. (a) Two-dimensional bit-propagating scheme employed in this work. (b) Conventional word-comparator-based scheme.

Figure 5.3 shows the comparing operation at each tournament stage (Figure 5.3(a)) and the circuit configuration of a bit comparator circuit (Figure 5.3(b)). The signal *BIT* represents the bit data to compare and propagates in the direction ascending the tournament tree. On the other hand, the signal *STATE* represents which of the two inputs is the winner (the one having the larger value), and propagates in the direction of the bit-serial word comparison. At the bit comparator, when both of *STATE_IN* signals are '1', the bit data *BIT_IN0* and *BIT_IN1* is compared. And then, the *STATE_OUT* signal corresponding to the loser falls to '0', and the bit datum of the winner is output as the output signal *BIT_OUT*, which is input as *BIT_IN* signal at the next tournament stage. When the one of *STATE_IN* signal is '0', comparison is not carried out, and the *BIT_IN* signal corresponding to the winner is output as the signal *BIT_OUT*. Initial *STATE_IN* signals fed to the MSB comparator are set to '1', and the *STATE_OUT* signals at each bit comparator are inputted as *STATE_IN* signals at the following bit comparator. The *STATE_OUT* signals from the LSB comparator represent the word-comparison result at each tournament stage. To minimize the delay time, the *STATE* signal propagation path employs a pass-transistor configuration. This configuration is particularly advantageous in a large bit-length organization because the *STATE* signal propagation delay is minimized, which is proportional to the bit length.

An encoder circuit for the winner location of the maximum input is configured as a network reflective to the WTA tournament tree. Encoding is carried out by tracking the winner in each tournament stage from the final tournament stage, after the operation at the LSB comparator of the final tournament stage finished. Each encoder stage receives a pair of *STATE* signals from the LSB bit comparator of the corresponding WTA tournament stage, and select a larger input. In the encoder, priority encoding is employed for the case of more than one input concurrently having the maximum value.



(a)



(b)

Figure 5.3. Bit comparators in each tournament stage of two-dimensional bit-propagating scheme. (a) Operation of bit-comparators. (b) Circuit configuration of the bit comparator.

B. Variable-Binary-Block Address Decoder

In order to provide a variety of WTA functions, like local winner search, winner sorting etc., the input signals to the WTA unit can be arbitrarily selected from a large number of parallel inputs. Conventional masking scheme such as changing masking flags one by one, or applying bit-mask-pattern for selection are requiring a large number of clock cycles or not scalable for the large number of WTA inputs, respectively. In this work, the variable-binary-block addressing scheme employed in the variable masking unit (See Figure 5.1(a)) has been developed, which allows us to select multiple input lines using a single address code with a constant delay time despite increase of the number of inputs. Whether an input is masked or not is determined by a value in the masking register provided for each input (See Figure 5.1(b)). Multiple masking registers can be simultaneously specified and updated by a single block-address code.

Figure 5.4(a) shows an example of address assignments for the variable-binary block address. In the figure, the 16 targets to address and the corresponding block address codes are shown. For example, in the case of an address code “20” specified, the targets having the number from “8” to “11” are activated simultaneously.

The block address code is assigned in certain rule. Figure 5.4(b) illustrates the block-address decoding algorithm. A block address code has two parts: a block size part and a starting position part. The block size is decoded by searching for the first "1" bit from the least significant bit (LSB) of the address code. After retrieving the block size from the address code, the starting position is given as the remaining bits shifted 1-bit left. In this manner, the block size and its starting position are restricted to 2's power and multiples of the block size, respectively. As a result, the address code is only one bit longer than the normal addressing.

Figure 5.4(c) shows the circuit configuration of the block address decoder. The main decoder is composed of regular decoders (multiple input AND's) to

yield masking flags into masking registers. In order to specify the plural of flag signals simultaneously, predecoder and "don't care" generator circuits are provided. The "don't care" generator circuit computes signals $D_1 \ldots D_N$ to neglect some bits from the LSB in the address code, according to the block size. Which block to select is determined by the predecoder, which generates upper bits signals in the address code. An overhead to the conventional address decoder is only the "don't care" generator circuit, and it has only negligible increase in the circuit delay time and area for the total of the decoder circuit. In the processor developed in this work, the "don't care" generator circuit has only less than 5% increase of them. It is quite advantageous in applications requiring the plural of targets simultaneously to select, such as WTA circuit.

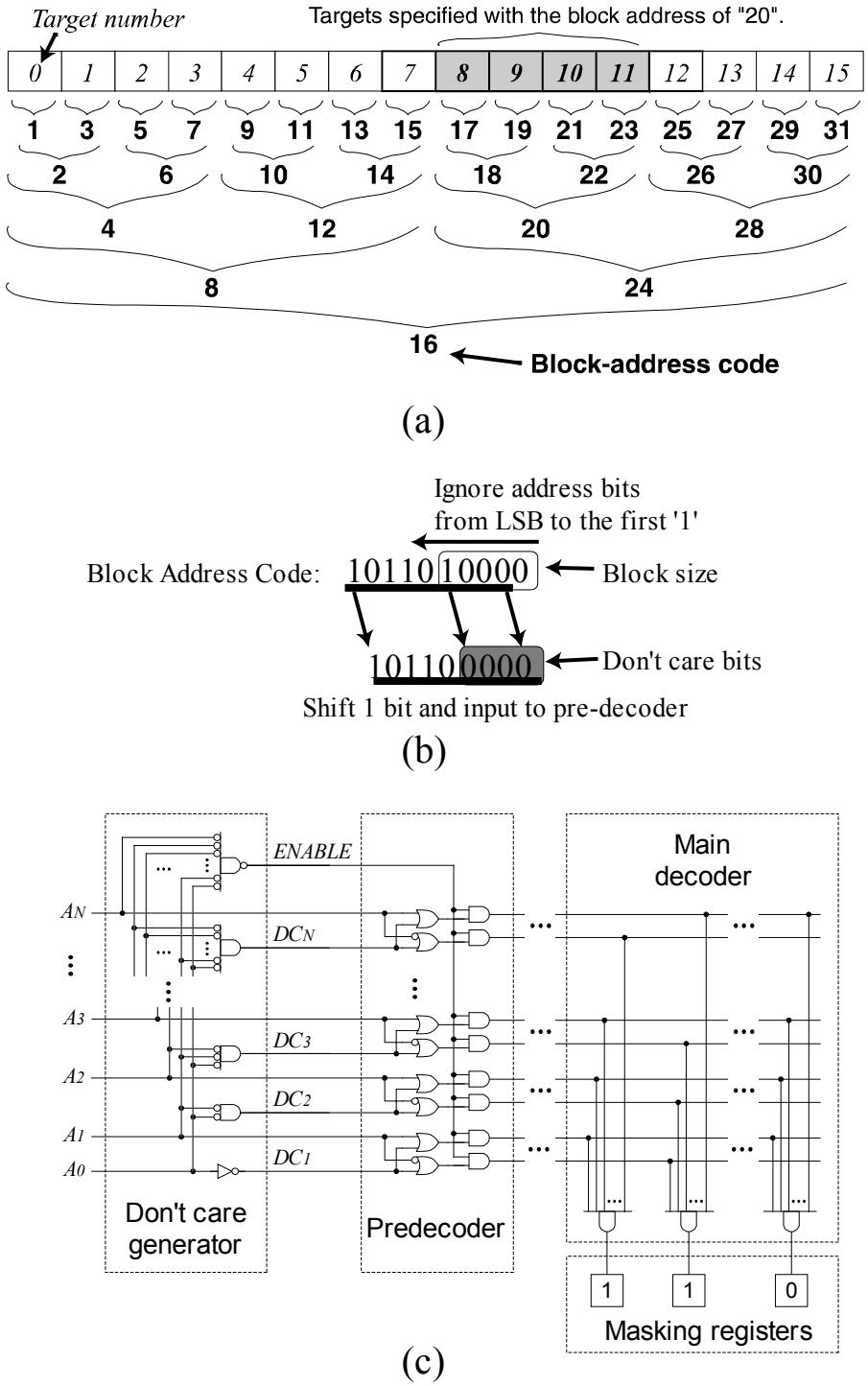


Figure 5.4. Variable-binary-block addressing scheme. (a) Example of address assignments. Targets specified with the block address of “20” are highlighted as example. (b) Circuit configuration of variable-binary-block address decoder.

C. Datapath for Distance Computation

Figure 5.5 shows the datapath logic for a distance computation. It has an 8bit adder for absolute-subtraction, a 16bit left shifter, a 24 bit adder for accumulation, several registers, and some logic gates. The distance computation unit receives the input vector data from the controller and the template vector data *TMP* from the memory, respectively. (See Figure 5.1(a)) The signals from the controller contain an 8b input-element value *IN*, a 3b shift amount *SHIFT* and a sign for accumulation *SIGN*. The logic carries out the Manhattan distance calculation and weight multiplication as follows,

$$Acc \mathrel{+}= (-1)^{SIGN} \times 2^{SHIFT} \times |IN - TMP| \qquad \text{(Eq. 5.1)}$$

Here, an *Acc* represents internal value for the accumulation. In the case of calculating simple Manhattan distance, *SIGN* and *SHIFT* inputs are set to the constant value of 0. Multiplication is realized by iterations of bit-shift and accumulation to minimize the layout penalty. Negative sign of accumulation is also supported, thus reducing multiplication steps to half with the Booth's algorithm. If a result of subtraction at the first adder is negative, absolute function is carried out based on two's complement by bit-inversion after subtraction and adding "1" to the result. The adding "1" is applied to the carry-in input for the LSB bit of the following accumulator. When shifting is applied, however, the value "1" to add is also required to shift and becomes 2^{SHIFT}. Therefore, inputs for shift-in of the shifter are also employed for addition in such a way that the sum of the carry-in of the accumulator and shift-in of the shifter becomes 2^{SHIFT}. As a result, the area increase in the present distance-computation element as compared to the simple Manhattan distance element [49] is only 18%.

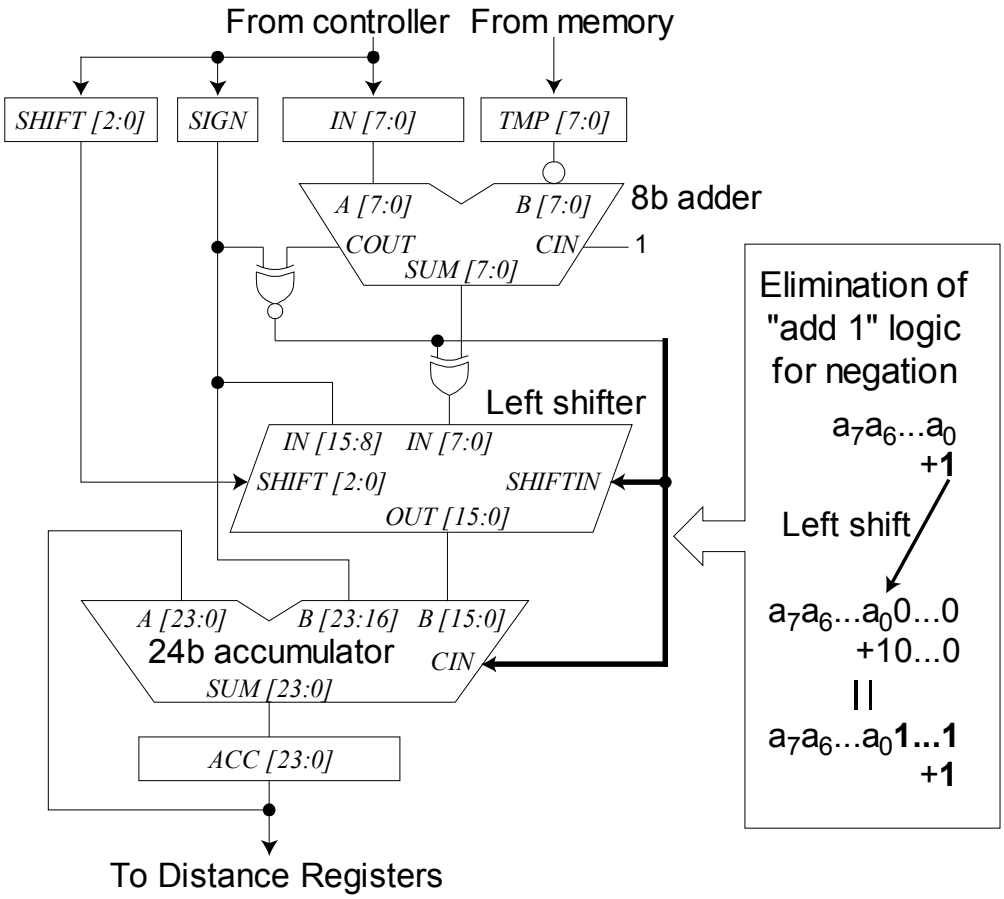


Figure 5.5. Data-path logic of distance-computation unit.

IV. Results and Discussions

A. Discussion on Delay Time of WTA Schemes

Figure 5.6 shows the simulated delay times of the three WTA schemes: the two-dimensional bit-propagating scheme (2DBP) developed in this work (See Figure 5.2(a)), the word-comparator (WC) based scheme (See Figure 5.2(b)) and the bit-sliced scheme [49-51] (See Figure 2.11(b)). It shows that the delay times of the 2DBP scheme are increasing slowly against the number of inputs. An advantage of 2DBP scheme in delay time is explained as follows.

In any scheme, the total delay time depends on the scale of the circuit,

namely the number of input N and the bit length n of each input word. In typical WTA circuits, n is from 8 to 32, and N is larger than several hundreds. In both of 2DBP scheme and WC-based scheme, the circuits are configured as the tournament tree network, and the number of tournament stages is $\log N$. In conventional WC-based scheme, the delay time at each tournament stage is proportional to $\log n$ due to the carry look-ahead circuit, thus the total delay time of the tournament tree becomes $O(\log n \times \log N)$.

On the other hand, the delay time of the bit-serial comparator employed in the 2DBP scheme is $O(n)$. It is slower than the WC-based scheme, however, the bit comparison operation proceeds not only in the direction of a word from MSB to LSB, but also in the direction ascending the tournament tree. As a result, the total delay time is determined, not by $\log N$ times comparator delay, but $\log N$ plus comparator delay n, i.e., by $O(n + \log N)$.

In the bit-sliced scheme, a comparison is carried out in bit-serial manner. The circuit is divided into bit-processing blocks, which processes only one bit for all the inputs in parallel. The bit-processing blocks are connected in bit serial from the MSB to the LSB, thus the total delay time is proportional to the bit length n. At each bit-processing block, one bit comparison is carried out by searching for the maximum bit value (in the case of the maximum WTA circuit), using a large OR gate, which receives bit data from all the inputs. The delay time of the OR gate is not negligible and becomes proportional to $\log(N)$, since the number of inputs N is quite large. Therefore, the total delay time of the bit-sliced WTA becomes $O(n \times \log N)$.

As described above, The 2DBP scheme has the delay time of $O(n + \log N)$, the WC-based scheme has $O(\log n \times \log N)$, and the bit-sliced scheme has $O(n \times \log N)$. When the number of WTA inputs becomes large, the 2DBP scheme is the fastest as shown in Figure 5.6.

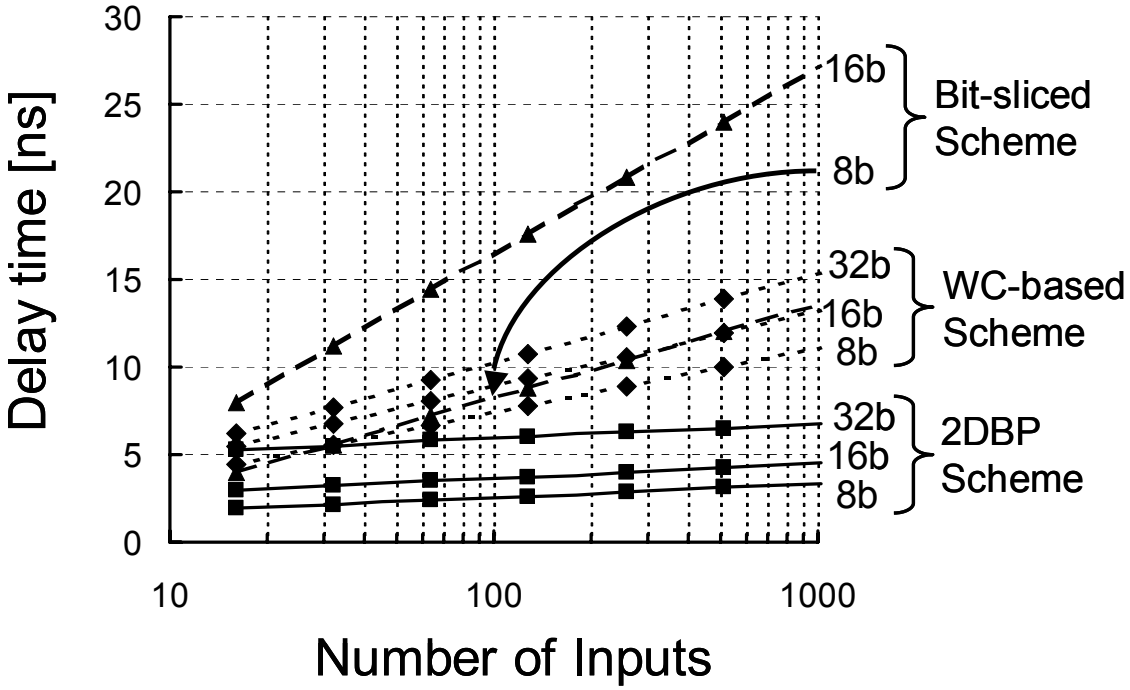


Figure 5.6. HSPICE simulation results for delay time of three winner-take-all schemes: two-dimensional bit-propagating (2DBP) scheme, word- comparator (WC) based scheme and bit-sliced scheme. Transistor models for 0.18mm process were used.

B. Prototype Chip

A prototype chip was fully designed by hand-layout using Cadence tool (icfb) to minimize the chip real-estate except for the controller which was designed by verilogHDL. The chip was fabricated in a 0.6-μm 3-metal CMOS process. Figure 5.7 shows the chip photomicrograph. The prototype chip has a 32KB SRAM for template vector storage, 32 parallel distance-calculation elements, and a 6-bit 128-input WTA circuit. It operates at 33 MHz with a power dissipation of 800 mW under a power-supply voltage of 4.0 V.

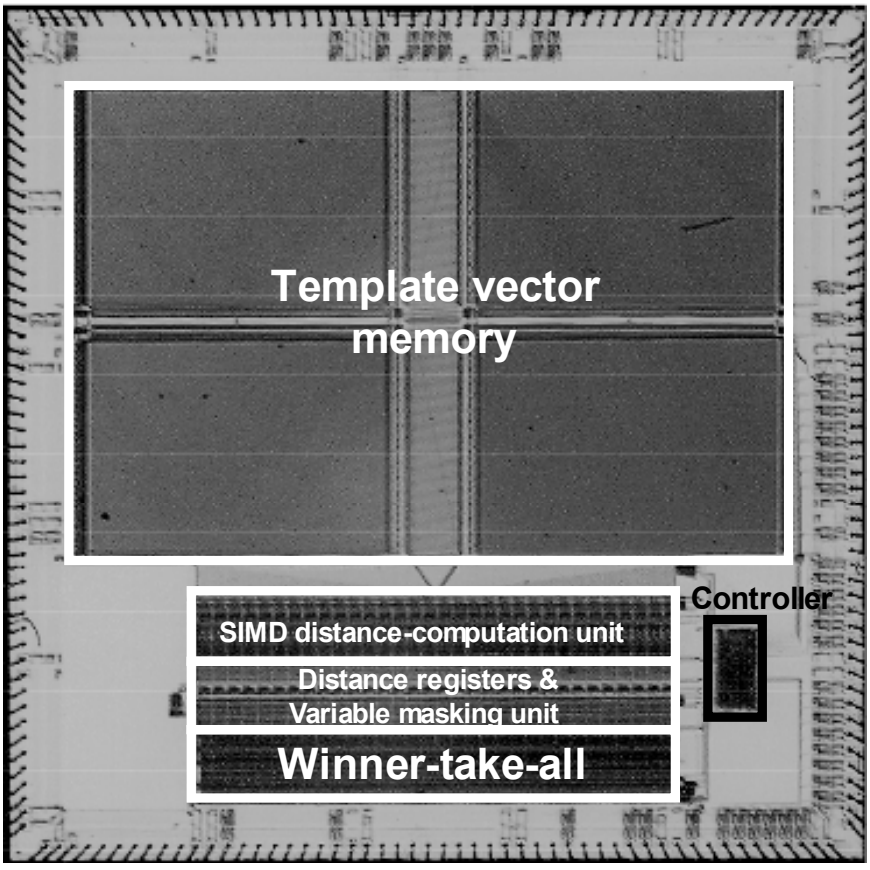


Figure 5.7. Photomicrograph of prototype chip.

Figure 5.8 shows measured waveforms of the two-dimensional bit-propagating WTA circuit on the chip. Three signals, the clock, the first tournament-stage output, and the final stage output, were monitored. At the positive edge of the clock signal *CLK*, the WTA action is initiated. The delay of 6ns represented as τ1 from the clock edge to the first stage output is the delay for the bit-serial word comparison, which is proportional to the bit-length n of the input data. On the other hand, the delay of 7ns represented as τ2 from the first stage output to the final stage output is the delay for the *BIT* signal to ascend the tournament tree, which is proportional to the logarithm of the the number of inputs log$N$, i.e., the number of stages in the tournament tree. The total delay of the WTA becomes the sum of these two delays of 13 ns.

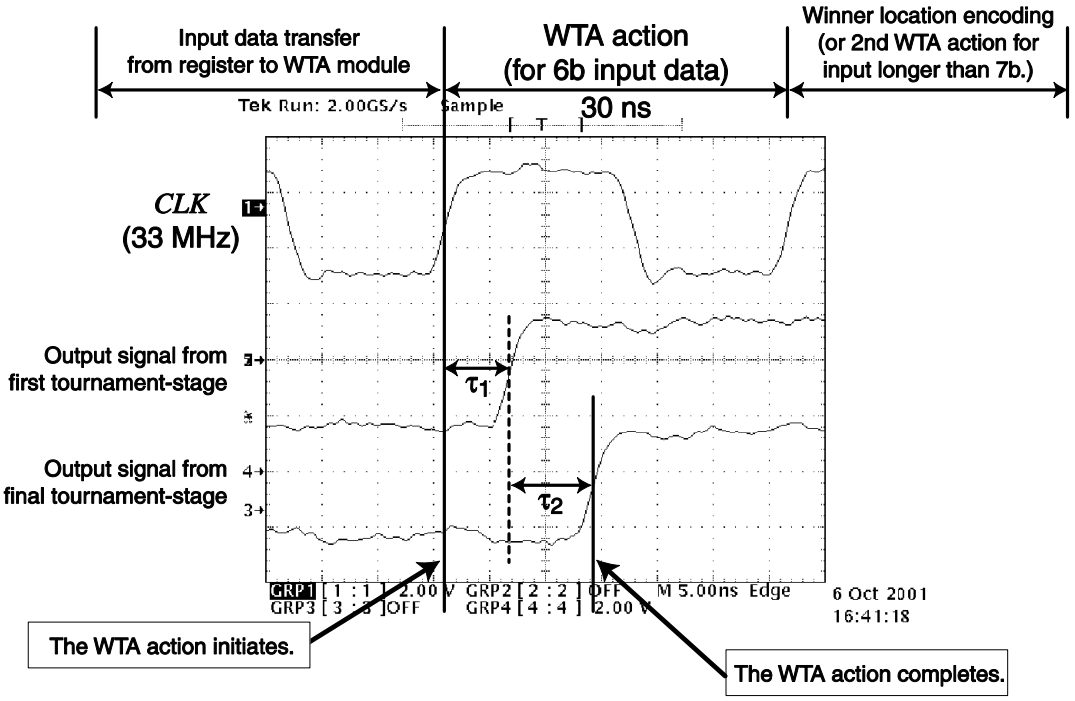
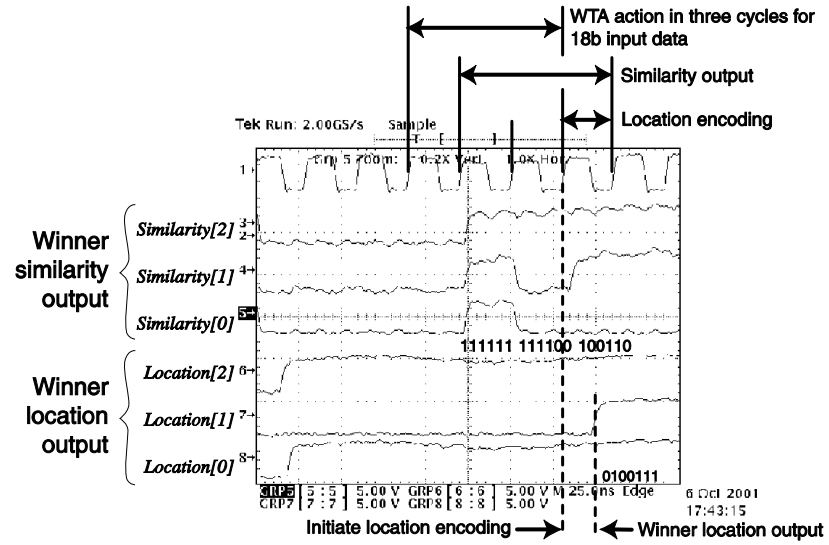
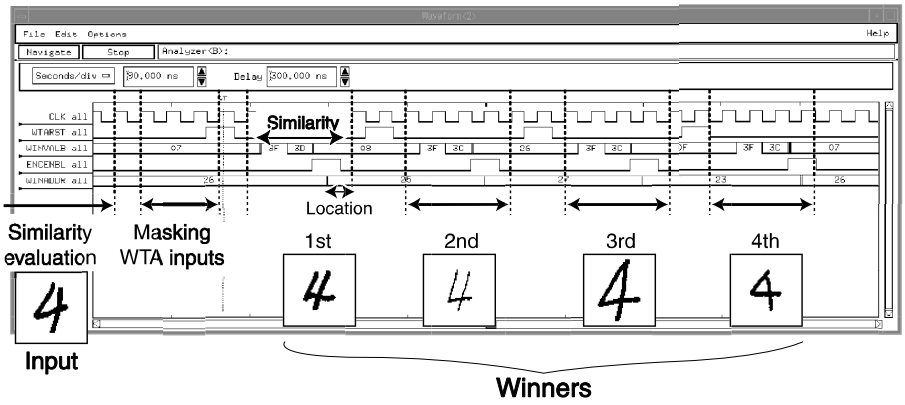


Figure 5.8. Measured waveforms of two-dimensional bit-propagating WTA circuit.

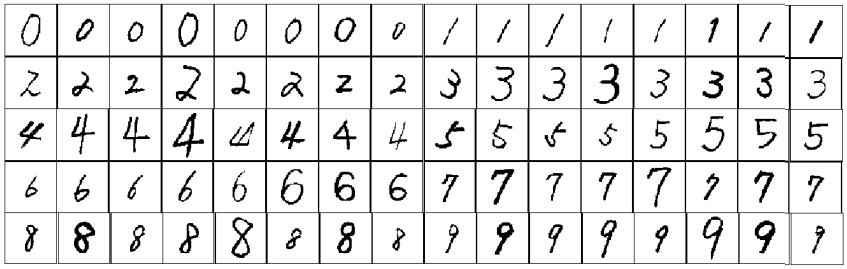
Figure 5.9 shows the measured waveforms from an oscilloscope (Figure 5.9 (a)) and a logic analyzer (Figure 5.9(b)), demonstrating the handwritten character recognition [5]. 80 samples of handwritten digits as shown in Figure 5.9(c) were taken from the Electro Technical Lab. database (ETL-1) and each 64×64 pixels image was transformed into a 64-dimensional feature vector using the projected principal edge distribution (PPED) vectorization algorithm [5-9]. Since the human perception of similarity in images is very well represented by PPED vectors, handwritten digit recognition can be easily carried out by a simple vector matching operation using the VQ processor. In the matching experiments, 80 out of 128 WTA inputs are activated using the variable-binary-block addressing scheme, and top four candidates for the most similar pattern to the input were identified using the high speed WTA circuit developed in this work. The successful search is evident from the result.



(a)



(b)



(c)

Figure 5.9. Handwritten character recognition experiment using the VQ processor. (a) Measured waveforms from a oscilloscope. (b) Logic analyzer output. (c) Template patterns used for matching experiments.

C. Experimental Design in Advanced Technology

Figure 5.10 shows a layout of experimental design for a highly parallel VQ processor. The processor has been designed in 0.18-μm 5-metal CMOS process. In order to localize the clock distribution, the processor is divided into 32 local blocks, and each local block independently distributes a local clock signal. The local block has 1Kbytes SRAM memory (Region "A" in the figure), 8 parallel SIMD distance computation units (Region "B" in the figure). The 1Kbytes SRAM memory can store eight 128-elmenet vectors. The WTA circuit is configured as 512 word inputs in the whole chip and forms a large combinational logic. The region "C" in the figure represents the WTA circuit corresponding to two local blocks. The WTA circuit operates in asynchronous from local clock signals, and outputs the winner distance and its location code at once (in a single global clock). Each WTA input has 33b length for 24b of a distance data and for 9b of a location code, and the address encoding circuit is replaced to the lower significant bits of the WTA circuit. The processor core occupies an area of 4.5mm × 4.5mm. The area ratio of the processor is 36% for SRAM's, 38% for distance computation units, and 26% for the WTA circuit.

The processor operates at 200MHz of local clock signals under 1.8V power-supply voltage with 1.8W power dissipation. The performance of the processor corresponds to 154 GOPS, assuming absolute-difference and accumulation require 3 operations in conventional microprocessors. Compared with the state-of-the-art processors, the efficiency of power dissipation is large, as well as the performance.

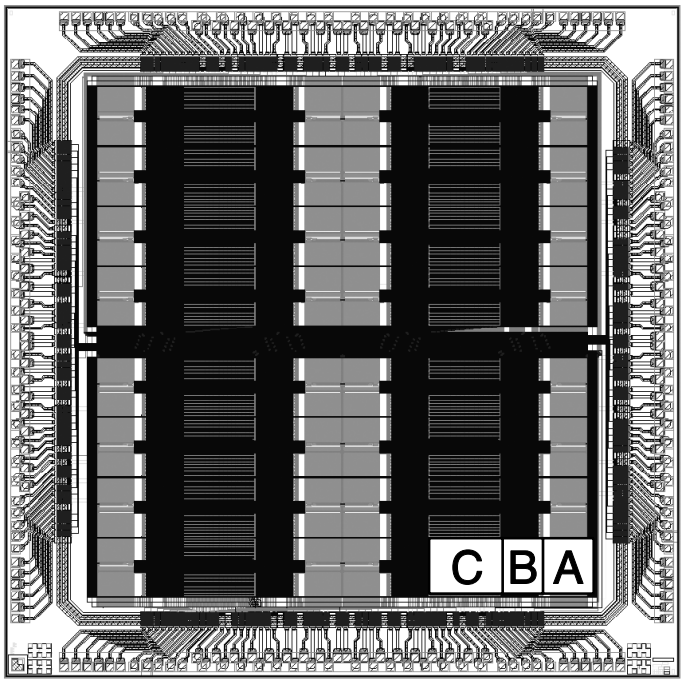


Figure 5.10. Layout of experimental chip design using 0.18 mm technology. Region "A" has 32 input 33b WTA circuit, Region "B" has 8 parallel SIMD distance computation units, and Region "C" has 1Kbytes SRAM memory.

V. Conclusions

A general-purpose VQ processor featuring high-speed and versatile winner search functions has been developed. The two-dimensionally bit-propagating scheme minimizes the delay in the winner search operation, and 6b 128-input comparison is carried out in a single clock cycle, which is five times faster than the conventional bit-sliced scheme. In addition, it is scalable to the large number of WTA inputs. Its delay was measured as 13ns with the prototype chip. Additionally, the variable-binary-block addressing scheme has been developed to enable variety of winner search options. By the block-addressing scheme, multiple WTA inputs are specified at once with

only a single extra bit. Furthermore, the multiplier function is embedded in distance-computation units with the minimal area penalty of 18% increase.

The prototype chip was designed and fabricated in the standard CMOS process. It operates at 33 MHz with a power dissipation of 800 mW under a power-supply voltage of 4.0 V. The functioning of the VQ processor was successfully demonstrated by applying to handwritten character recognition as an example.

CHAPTER 6.
Dynamic Programming Matching Processor

I. Introduction

In human centric computing, computers are expected to have more intelligent functions in order to achieve higher flexibility in interfacing to humans. To this end, pattern matching plays an essential role. In speech recognition, image recognition, intelligent database search, for instance, pattern-matching techniques are extensively used to carry out robust classification and flexible decision making under noisy environment. Since the matching operation for exhaustive search is computationally very expensive, the low-cost implementation is a key issue. To minimize computational complexity, conventional matching algorithms usually employ

quite simple similarity-evaluation functions, such as Hamming distance, Manhattan distance, Euclidean distance and so on. These algorithms where element-to-element matching is employed, however, have a problem. Let's consider the matching between the two words, "MOTHER" and "OTHERS", for instance. In conventional matching algorithms, "complete mismatch" will result despite the common letters of "OTHER" in both words. It is because the comparison is carried out letter by letter at the same location in the sequence from the beginning of the words. On the other hand, in sequence-based matching algorithms known as dynamic programming (DP) matching, we can shift the letters in order to obtain the best matching result. The sequence matching is quite robust and is essential in certain class of problems, like DNA matching, speech recognition, etc. However, it requires far more expensive computational powers, because exhaustive search must be conducted for all possible combinations of shifts in elements for missing and/or extra elements. Therefore, hardware implementation is essential to achieve real-time response performances.

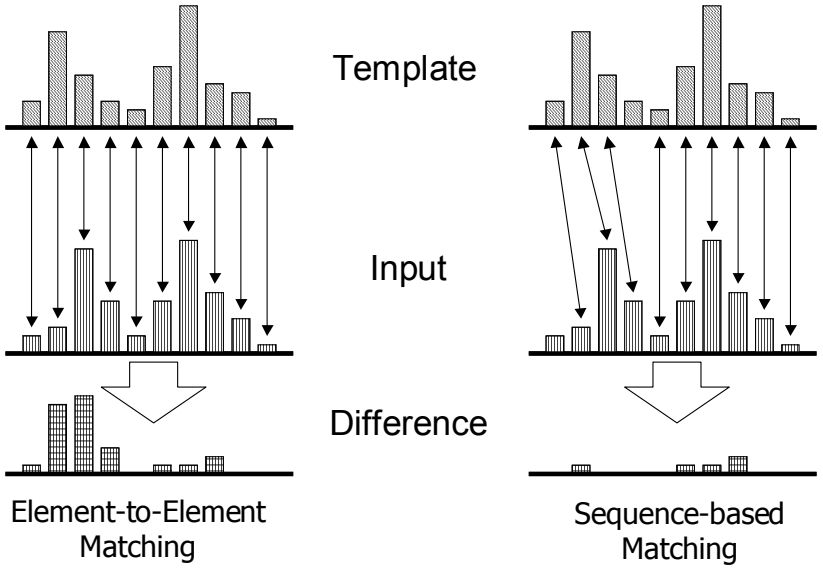


Figure 6.1. Element-to-element matching and sequence-based matching.

For this reason, VLSI systems for DP matching have been developed [58-63]. Conventional VLSI systems are classified into three types of parallel systems, microprocessor-based multi-chip systems [58, 61, 62], systolic array processors [60], or dedicated processors [59, 63]. In microprocessor-based multi-chip systems, general-purpose microprocessors or processors having limited instruction-sets are employed as processing elements (PE's), and plural of such chips are connected. Each PE chip has usually an instruction decoder, a general-purpose ALU unit, and data/instruction memories on the chip, and their advantage is a large flexibility in accommodating themselves to various matching algorithms. The PE in Ref. [61] is optimized for efficient DP matching while retaining flexibility, and the cost and performance are improved. In Ref. [62], the flexibility and efficiency in a multi-chip configuration is enhanced by the ring array architecture. However, the PE configuration and the inter-PE-connection interface are quite complicated, and the chip real estate and power dissipation are too large. The systolic array processor [60] has a large number of PE's dedicated to DP matching algorithms. Each PE has only limited functions, but high throughput has been achieved by a highly parallel processing architecture. The application-specific processor for dictionary search [63] enhances parallelism to execute simultaneous matching of inputs with a number of templates. However, the maximum dissimilarity range is limited to only several levels.

In this paper, a DP matching processor has been developed featuring an array architecture using delay-encoding-logic. In this architecture, signals in the circuits are all digital in voltage domain as shown in Figure 6.2, while analog processing is carried out in the time domain like in Refs. [64-65]. As a result, high speed and low-power processing, and small-chip-area implementation have been achieved. A prototype chip was designed and fabricated in a 0.18-μm CMOS technology, and the typical matching time of 80ns with the power dissipation of 2mW under the power supply of 1.3V has been demonstrated.

In section II, details of the dynamic programming matching algorithm is described. The processor architecture for DP matching and circuit configurations are shown in section III and section IV, respectively. Measurement results from the fabricated prototype chips are presented in section V, and conclusions are given in section VI.

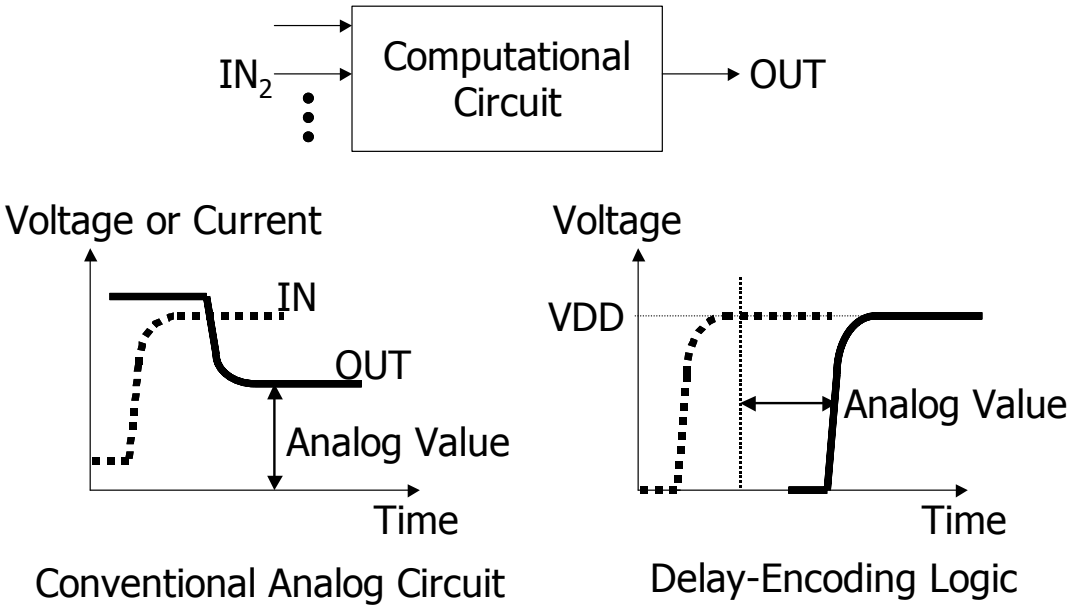


Figure 6.2. Voltage-domain digital and time-domain analog scheme.

II. Dynamic-Programming Matching Algorithm

Figure 6.3 explains the DP matching algorithm using an array network. Two sequences of letters, "MOTHER" and "OTHERS" are compared as an example. The two sequences of letters are placed at the top and the left sides of the network, and each branch of the network has a certain penalty according to the mismatch between the two letters at corresponding locations in the two letter sequences. Paths in the network from the start node to the goal node represent all possible sequence combinations allowing element

skipping (or shifting). Here the best-matching result is indicated by the thick line, representing the path having the minimum penalty among all possible combinations. The task of DP matching is to find out the best-matching path in the network along with its penalty.

Firstly, each branch of the network is assigned a certain penalty representing the partial matching result. The penalty for the diagonal branch is proportional to the degree of mismatch between corresponding two letters. Namely, a large (or small) penalty is assigned when two letters are different (or identical). For vertical or horizontal branches, a constant penalty representing skipping or shifting is assigned. In this example, only penalty values of 0 and 1 are given in the figure for the sake of simplicity of explanation. (In the real chip implementation, however, the diagonal and vertical/horizontal penalties are represented by 5-b and 4-b scalar values, respectively.) Then, to find out the best-matching path, $D(i, j)$ at each cross point of the network is calculated. $D(i, j)$ stands for the minimum penalty from the starting point (0, 0) to the point (i, j), which is given by the following recurrence formula:

$$D(i,j) = \min \begin{cases} D(i,j-1) + \text{Penalty of Vertical Path} \\ D(i-1,j-1) + \text{Penalty of Diagonal Path} \\ D(i-1,j) + \text{Penalty of Horizontal Path} \end{cases} \qquad \text{(Eq. 6.1)}$$

$D(i, j)$ must be calculated for all cross points, and the penalty for the best-matching path is finally obtained as $D(7, 7)$.

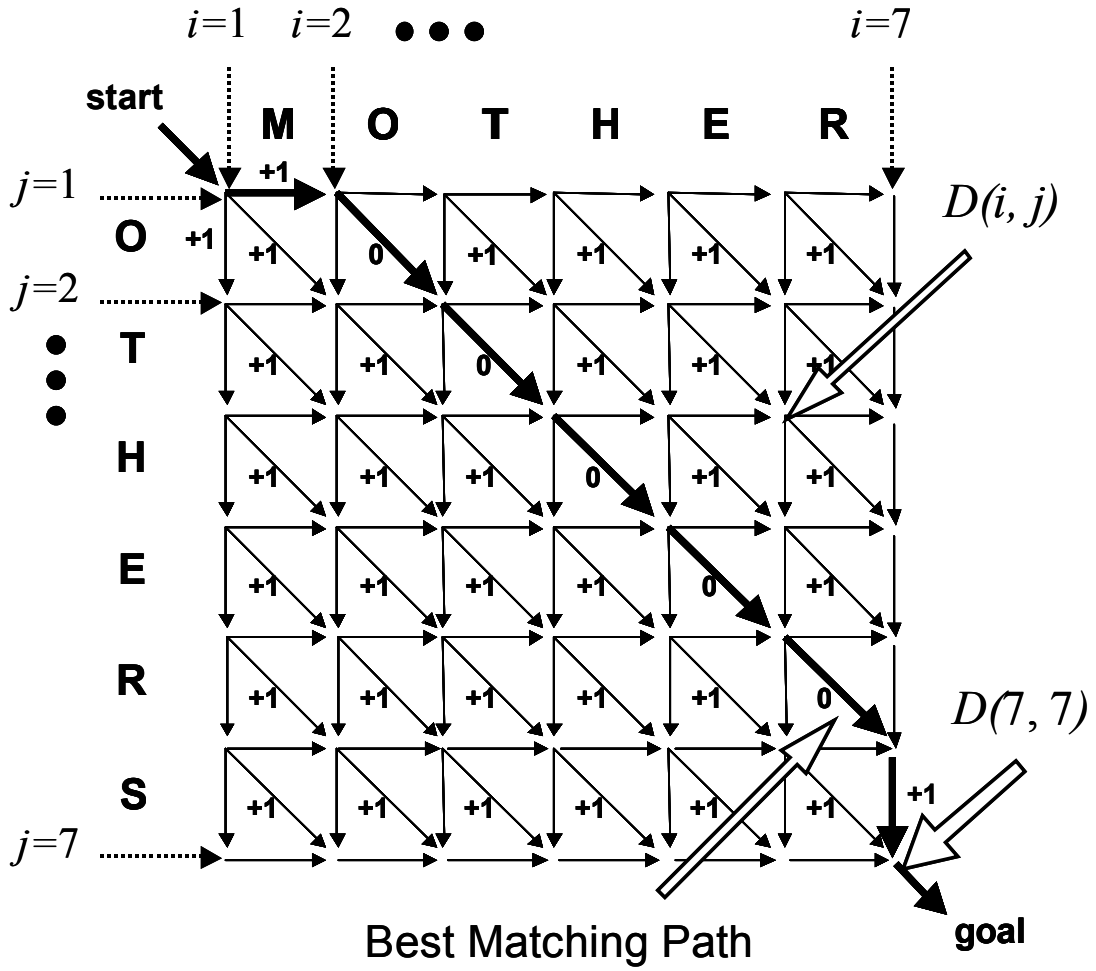


Figure 6.3. Dynamic-programming matching algorithm.

III. Processor Architecture

A. System Organization

Figure 6.4 shows the system organization of the DP matching processor. The processor consists of an array network of delay-lines, element-to-pulse converters, and a time-to-digital converter. The core of the system is the array network, physically representing all possible sequence combinations between the input vector **X** and the template vector **T**. In the array network, programmable delay lines are placed in diagonal as well as in horizontal and

vertical directions. Each delay line has a delay-time representing the penalty, i.e., the degree of mismatch. Namely, the delay time of the diagonal line represents the mismatch between the corresponding elements in **X** and **T**. On the other hand, a constant delay time is assigned to horizontal and vertical delay lines, which specifies the penalty associated with a single-element skip (or shift) in the matching. In order to obtain the DP matching result, a step signal is given to the top left corner of the network, and then the matching result is obtained as a delay time observed at the bottom right corner. In this respect, the circuit architecture is called "delay encoding logic".

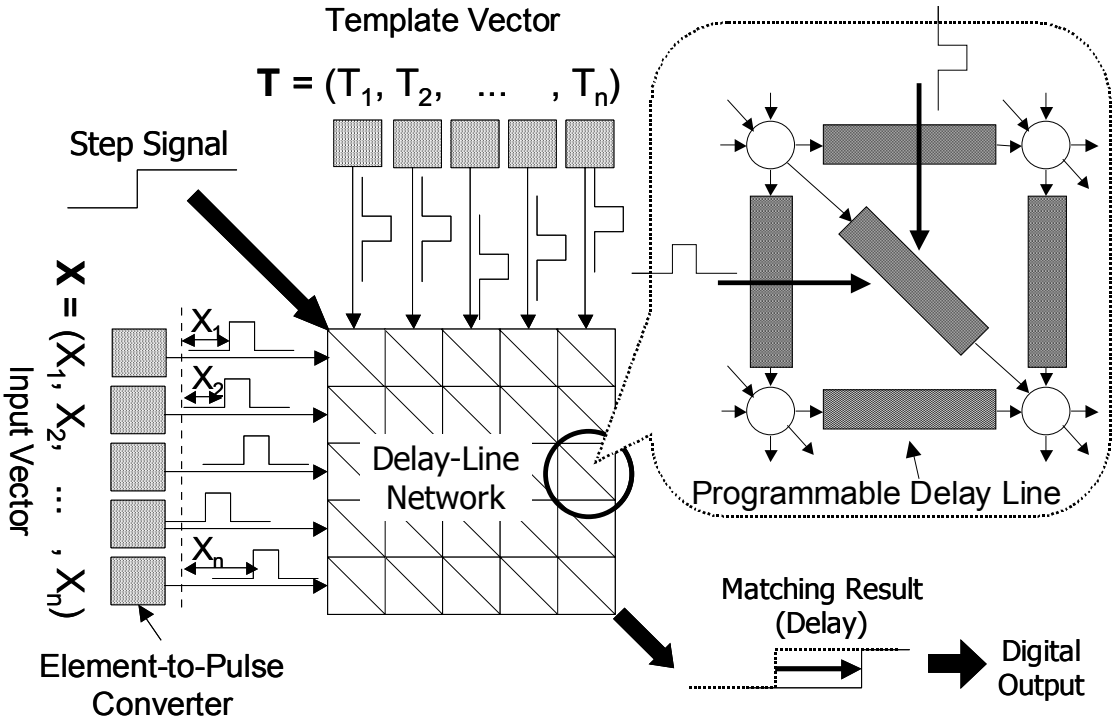


Figure 6.4. Delay-encoding-logic architecture of DP matching processor.

B. Operation in the Delay Setting Phase

The processor operates in two phases: the delay-setting phase for delay line programming and the DP matching phase for step signal propagation.

In the delay-setting phase, the diagonal delay line is programmed to have the delay time proportional to the mismatch between the corresponding elements in **X** and **T**. All the diagonal lines are programmed in parallel using time-domain-analog pulse signals. The pulse signals are generated from vector-element values by the element-to-pulse converter circuits. Each pulse signal has an identical shape, while it has variations in its position according to the element value. As shown in Figure 6.5(a), at each diagonal delay-line, two pulses from input vector **X** and template vector **T** are fed to an "AND" gate. The AND gate computes the overlapping of the two pulses and outputs a pulse signal which represents the degree of mismatch by the pulse width. The output pulse from the AND gate is utilized as "WRITE PULSE" signal in Figure 6.5(b) to program the delay time in the delay line. The details for operation of the programmable delay line are described in Section IV. In this manner, the delay lines are easily programmed using simple logic gates in a fully parallel operation.

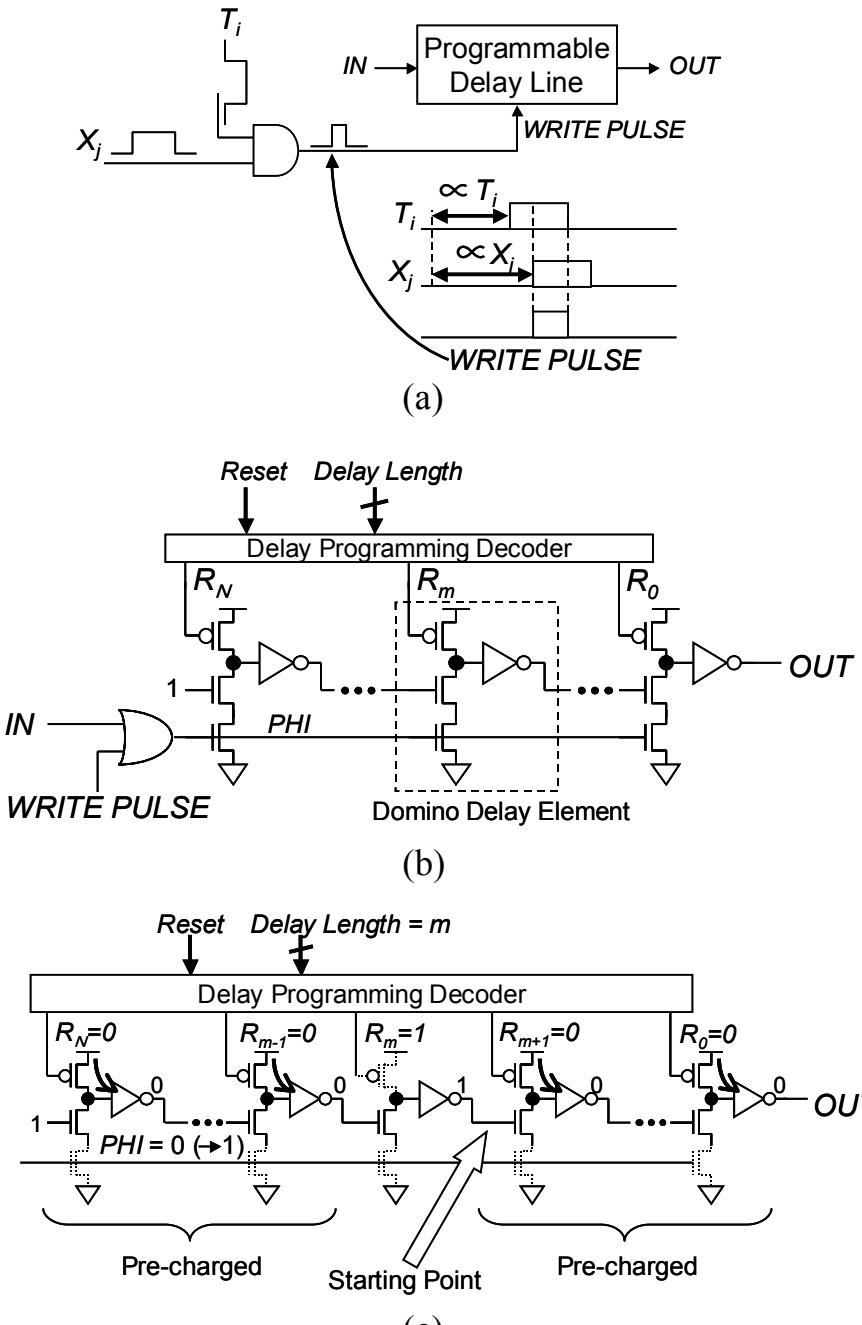


Figure 6.5. (a) Programming scheme of the diagonal programmable delay line using two pulse signals representing vector elements Ti and Xj. (b) Programmable delay line composed of domino delay elements. (c) Programming a constant delay time in the delay line. (N is 32 in the diagonal delay line and 16 in the horizontal and vertical delay line.)

C. Operation in the DP Matching Phase

After the delay-setting phase, a step signal is given to the top left corner of the delay-line network. The step signal is propagating through the network to all directions, and a delay time representing the penalty at each network branch is accumulated to the propagating step signal. At each network node, is provided with a simple "OR" gate which receives three inputs from vertical, horizontal and diagonal delay lines. The OR gate selects the first-arrival signal, and passes it down to the following delay lines. Thus the minimum penalty path is automatically selected at each node. The first arriving signal at the right bottom corner is the one that has propagated through the best-matching path. In this manner, the DP matching result is easily observed as a delay time in the signal propagation through the network. The result is converted to a digital format using a time-to-digital converter circuit.

IV. Circuit Configuration

A. Programmable Delay Line

Figure 6.5(b) shows the diagonal delay line composed of domino delay elements. The circuit has 32 stages of domino delay elements, and a decoder circuit for delay time programming. The circuit has two input terminals: the WRITE PULSE terminal for delay time programming, and the IN terminal for receiving a propagating step signal.

In the delay setting phase, after resetting all domino elements, a WRITE PULSE signal having the pulse width proportional to the degree of matching between the elements Xj and Ti (See Figure 6.5(a)) is applied to the signal line PHI. The domino elements are activated by the PHI signal. Then the domino signal is allowed to propagate through some stages in the delay line

according to the pulse width and to stop at a certain intermediate stage. The number of undisturbed domino stages remaining in the delay line, then represents the dissimilarity between Xj and Ti. In this manner, the dissimilarity is programmed as a delay time in the delay line. When the WRITE PULSE signal is narrow (because Xj and Ti have a large difference), the programmed delay time becomes large. Or if the WRITE PULSE signal is wide (because Xj and Ti are close to each other), it has a short delay time. This is the operation in the delay-setting phase. In the DP matching phase, when the step signal arrives at the IN terminal, the domino circuit is reactivated and the step signal starts to propagate from the intermediate stage through the remaining domino stages and is outputted from the OUT terminal. Thus, the delay time observed at the OUT terminal is proportional to the dissimilarity between Xj and Ti.

The horizontal and vertical delay lines have the same circuit configuration, but with 16 domino elements. The delay time is programmed by setting the domino starting point at some intermediate stage. This is carried out by the delay programming decoder during the reset of domino elements. For setting a delay time *m*, for instance, the decoder circuit makes Rm=1, while making others all "0" in order to pre-charge domino elements as shown in Figure 6.5(c). Then, in the DP matching phase, the domino starts from the intermediate domino element next to "m" upon the arrival of the step signal at the PHI signal line. In this manner, a constant delay time is added to the propagating step signal.

B. Element-to-Pulse Converter

Figure 6.6 shows the configuration of an element-to-pulse converter (EPC), enabling arbitrary shaping of the programming pulse. The circuit consists of two programmable delay lines with a vector-element value register, and a logic gate. The circuit given in Figure 6.5(b) is also utilized in these programmable delay lines. The delay line composed of 64-stage domino

elements is used for setting the pulse position, and the one with 32-stage domino elements is used for setting the pulse width. A vector element value is stored in the element value register in a 6-b digital format, which is fed to the upper delay line for pulse position setting using the technique explained with Figure 6.5(c). A 5-b pulse width value is fed to the lower delay line for pulse width setting. This value is common to all EPC's in the processor. In the pulse generation, the timing reference signal REF is given to all EPC's in the processor. (The REF is given to the PHI signal line in Figure 6.5(c).) Upon the arrival of REF signal, the first programmable delay line for pulse position generates a step signal whose rise position is delayed according to the element value. Then its output signal activates the second delay line for pulse width setting, generating a step signal with a delay specified as the pulse width. Then a simple AND gate generates a desired pulse.

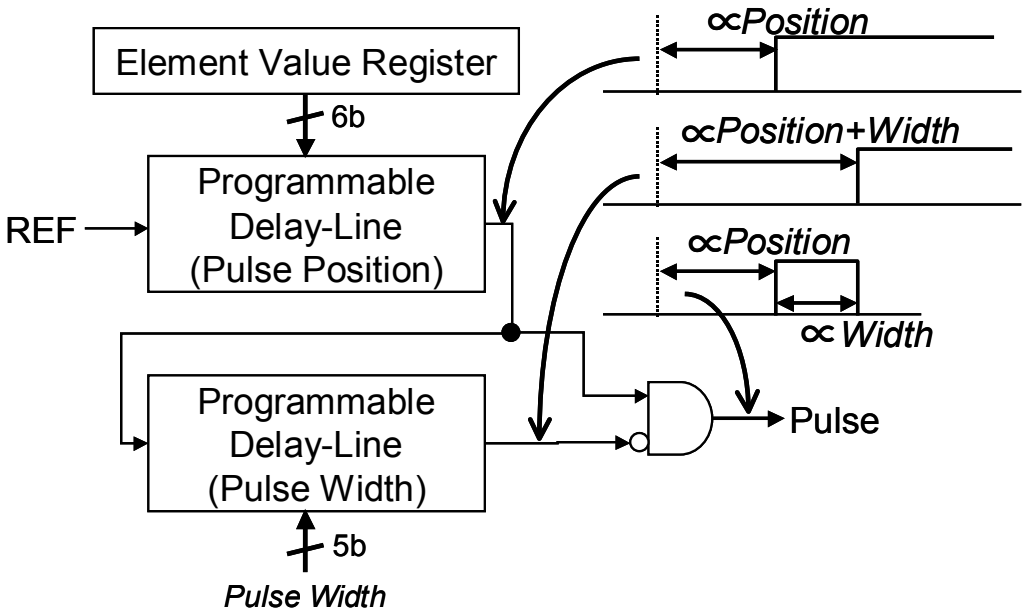


Figure 6.6. Element-to-pulse converter circuit enabling arbitrary shaping of the programming pulse.

V. Experimental Results

Figure 6.7 shows a photomicrograph of the prototype chip fabricated in a 0.18-μm 5-metal CMOS technology, where the DP matching core occupies the area of 1.0mm × 1.4mm. The processor is configured for the matching between two 16-element vectors, and circuits for the input and template vectors are layed out at both sides of the network symmetrically, in which each vector-element having 6-b precision is stored. The time-to-digital converter having a 256-level resolution is equipped to yield the output in a digital format.

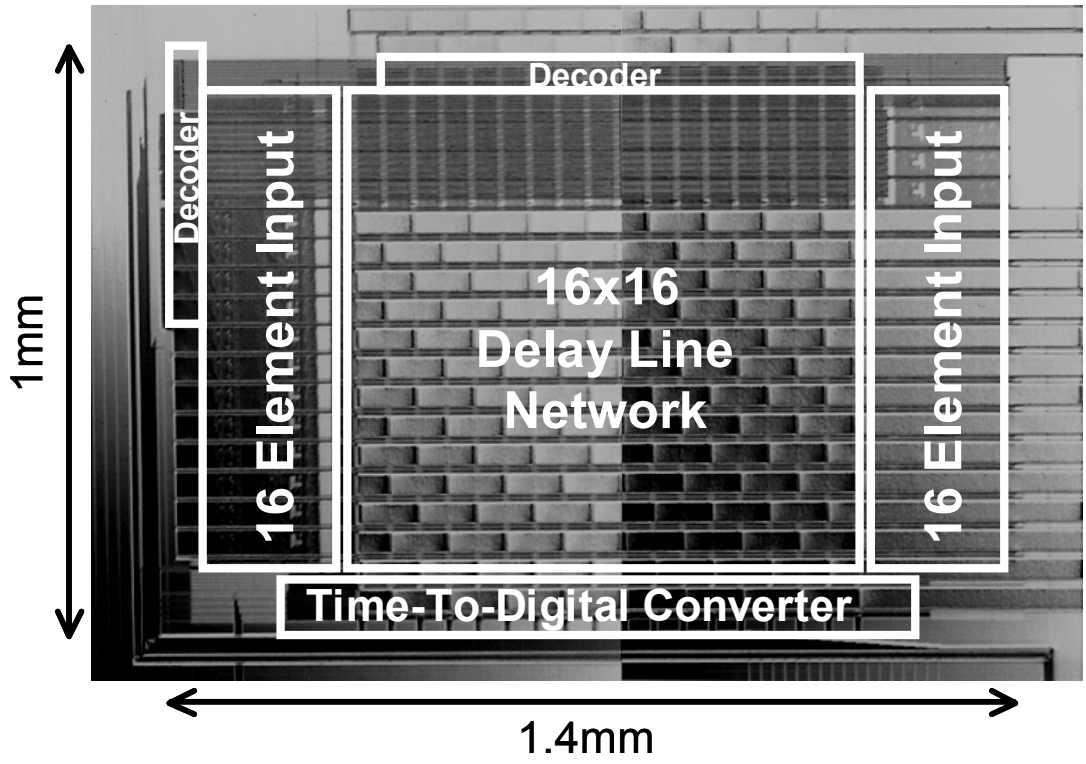


Figure 6.7. Chip photomicrograph.

Figure 6.8 shows the waveforms observed by an oscilloscope during matching operation. When ENABLE is high, the processor is under the delay

setting phase. In the DP matching phase with ENABLE low, the best-matching-path search is carried out by inputting the step signal into the delay network. Then the step signal is observed at the output after some delay. This delay time represents the dissimilarity of matching result. The digitally encoded result of the matching score is being outputted concurrently, and digital output encoding stops when the step signal is outputted.

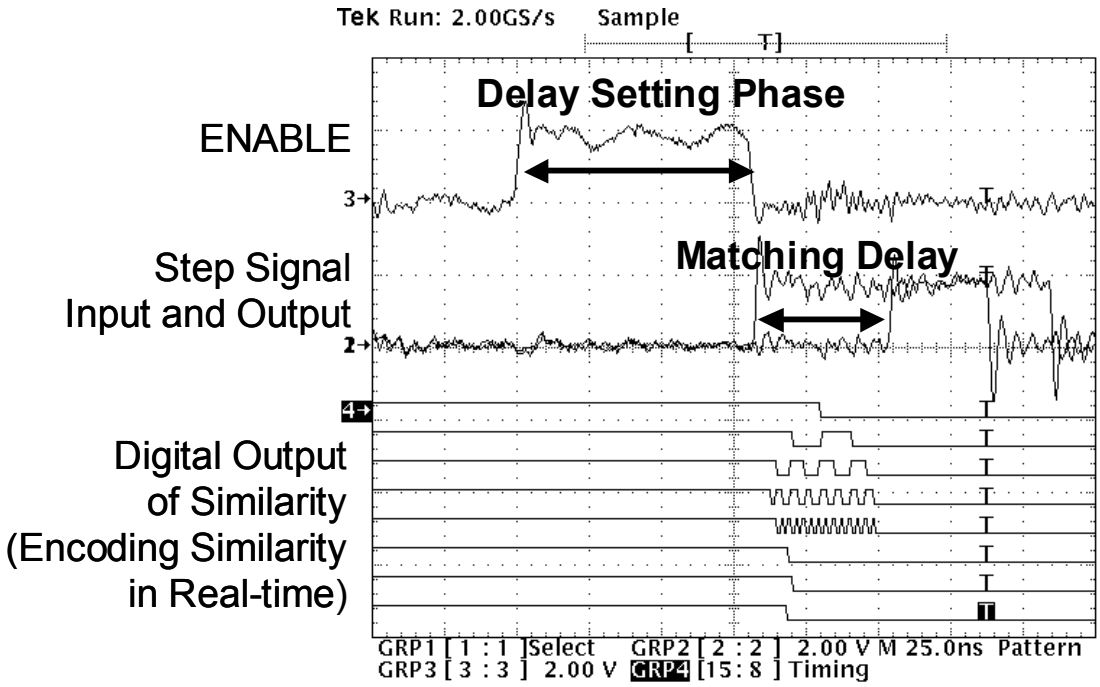


Figure 6.8. Measured waveforms of matching operation.

Figure 6.9 shows the delay time of the single delay line programmed using the programming decoder. The measured delay time is proportional to the programmed value. The small non-linearity is observed for every 8 programming values. It is caused by the layout grouped every 8 delay elements.

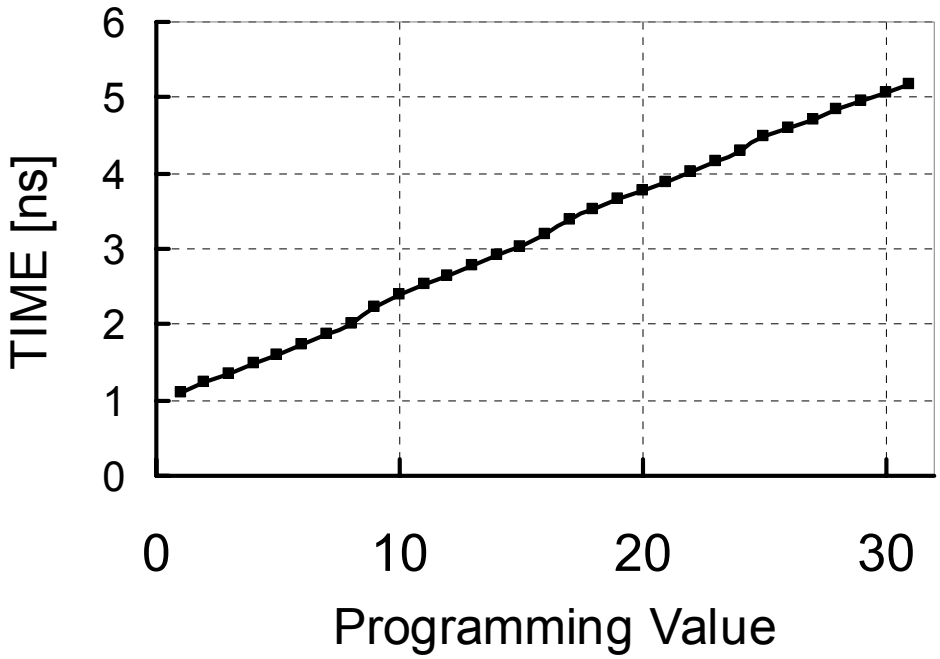


Figure 6.9. Measured delay times of the programmable delay line circuit.

Firstly, the intrinsic delay time along the diagonal path containing 512 (32 × 16) domino elements from the start node to the goal node was measured. For this purpose, the delay time in the horizontal and vertical lines was set at the maximum, thus excluding the signal propagation in these directions. It was measured as 78 ns. Figure 6.10 demonstrates the matching result of two vectors **X** and **T**. Both **X** and **T** have only one non-zero element Xs and Ts at the same element location. The element for Ts is set to a constant value of 16, 32 or 48, and matching was carried out for varying values of Xs. For Ts=16, for instance, the delay time representing the pulse position in Fig. 4 was approximately 2.1ns and the pulse width was 2.1ns (the digital value was set to 16) in this experiment. Figure 6.10 shows the difference between the measured total delay time along the diagonal path and the intrinsic delay time (78ns) as a function of Xs for three difference values of Ts. The minimum peak in the delay time occurs approximately at Xs = Ts. The

uncertainty in the minimum delay time of about 150ps is observed, roughly corresponding to the single domino-stage delay, the resolution of the time domain operation.

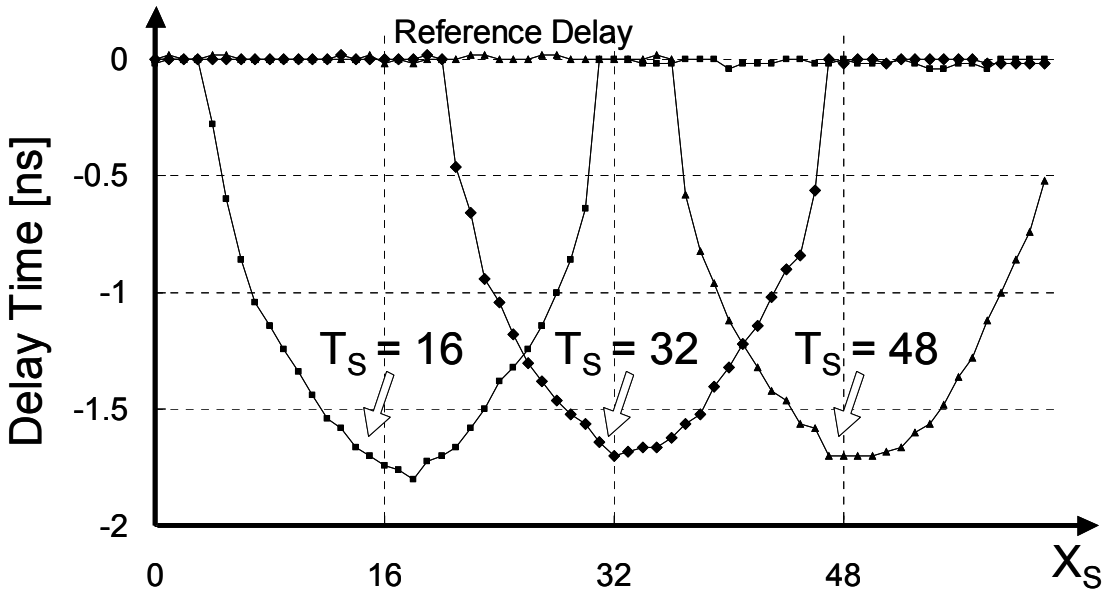


Figure 6.10. Measured delay times along the diagonal path in the delay line network representing the dissimilarity between two vectors.

Figure 6.11 demonstrates the results of both DP matching and conventional element-to-element matching between the two simple vectors shown in the figure. (The conventional element-to-element matching is emulated on the DP matching processor by assigning the maximum penalty to horizontal/vertical delay-lines.) Out of 16 elements in a vector, only two elements have non-zero values. The two vectors have an identical shape, but the relative non-zero element-position was varied. The element value took either 63 or 0, i.e., the maximum or minimum of a 6-b number. The DP matching yields the smallest delay time when the two sequences have identical position, while the delay time increases according to the relative shift in the position between the two sequences. On the other hand, in the

element-to-element matching, undesirable drop in the dissimilarity occurs when one of the elements coincides. This simple example illustrates the robust nature of the DP matching algorithm.

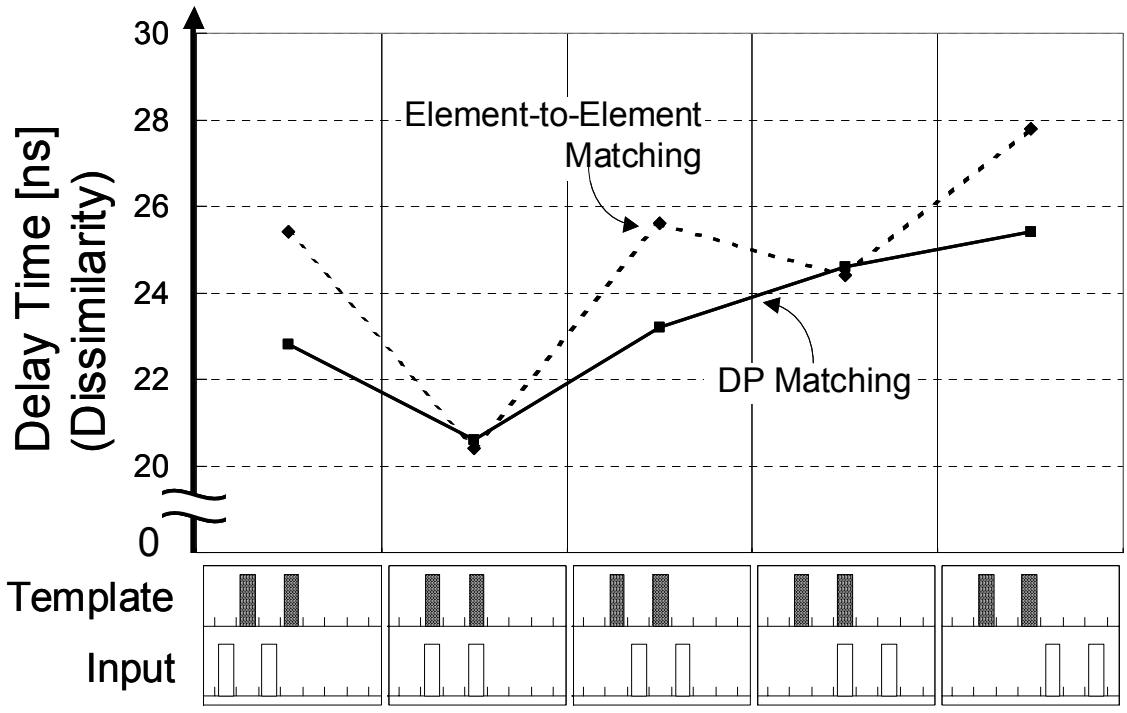


Figure 6.11. Measured results of simple sequence matching. The delay time represents the dissimilarity between the input sequence and the template sequence.

The results of DP matching as applied to more general vectors are presented in Figure 6.12. Test vectors were generated from face and non-face sample images[1] based on the edge information according to the algorithm given in Refs. [5-9]. Only 16 essential elements were selected and used in the present experiment. The ordinate represents the measured delay time and the digital signal output representing the degree of similarity between the

[1] The facial data in this experiment are used by permission of Softopia Japan, Research and Development Division, HOIP Laboratory.

input face image and a template image. Higher scores are obtained for face samples, thus showing a possible application to face detection in natural scenes [8, 9].

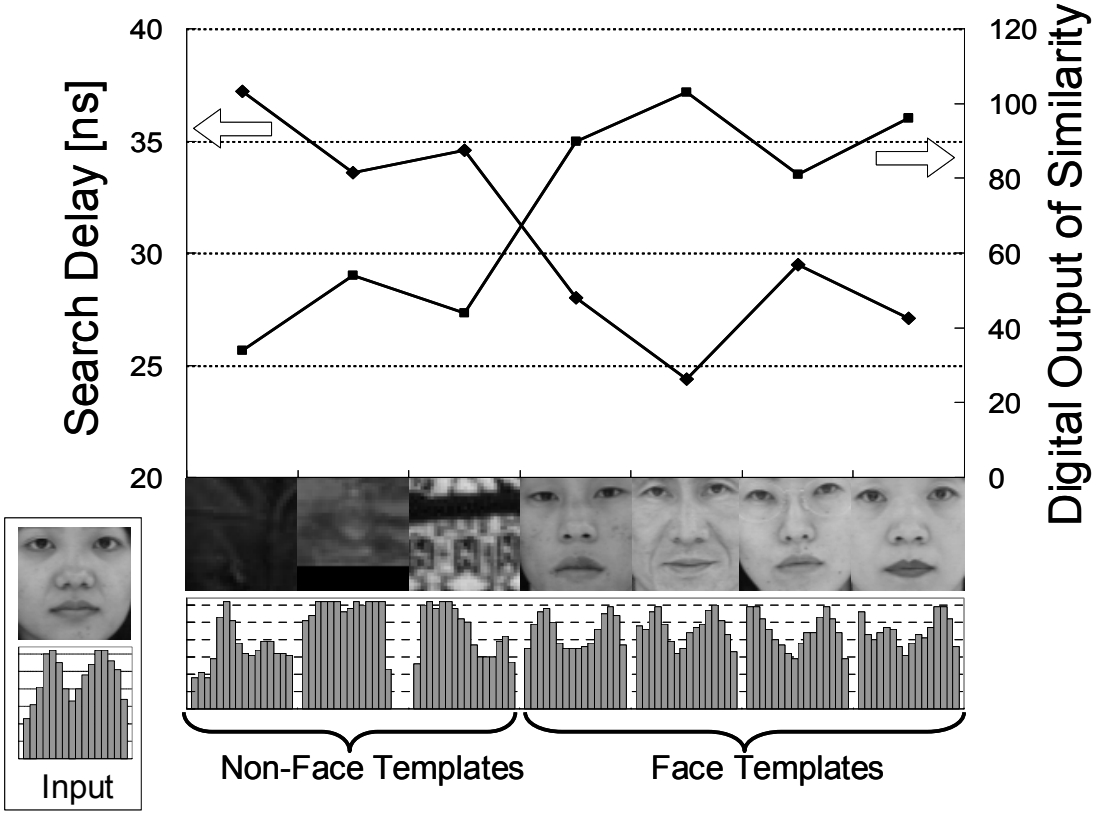


Figure 6.12. Matching demonstration using gray scale patterns.

Figure 6.13 shows the DP matching time and power dissipation as a function of power supply voltage for three different matching conditions. "A" represents the case where two vectors are identical, "C" the case where the Manhattan distance takes the maximum value, and "B" the case where the Manhattan distance is about a half of the maximum. The power-dissipation is normalized under matching rate of 1MHz. It decreases remarkably by reducing the power supply voltage. The circuit employs analog operation only in the time domain, and the circuit operates in a pure digital mode in

the signal domain. Therefore the circuit can be operated at low supply voltages to reduce the power dissipation like conventional pure digital processors. Under the power supply voltage of 1.3V, DP matching time is about 80ns with a power dissipation of about 2mW under the typical condition of "B". The matching delay of the condition of "B" is not the theoretical worst case but the practical upper limitation in usual applications, since the result having a large matching delay time is insignificant and negligible in the matching process. In the face detection as shown in the Figure 6.12, for instance, the matching delay time of 40ns is enough long to classify into a face or a non-face. Simulated delay time of the straightforward digital implementation is also indicated as the condition "D" in the figure. In the simulation, the circuits have only combinational logic for the DP matching computation except peripheral circuits such as a controller, registers or I/O buffers. It contains four adders (one for an absolute difference and three for penalty accumulations) and two comparators are utilized for the calculation of Eq. 6.1. In the configuration for 16-element vectors, the processor has more than 250 nodes in the network, and about 1000 adders and 500 comparators are required. The volume of the straightforward implementation becomes quite large and practical. Other digital implementations require much more processing time and power dissipation than the straightforward one. The computation of the DP matching between 16-element vectors corresponds to about 1800 operations, assuming two operations for absolute difference, three for penalty accumulations, and two for comparison, respectively, at each network node. If the DP matching is carried out every 100ns (it is enough long time for the developed processor under 1.3V power-supply), a performance requirement becomes 18 GOPS. However, state of the art processors with performance of several 10 GOPS consume more than several watts, which is far larger than the processor developed in this work.

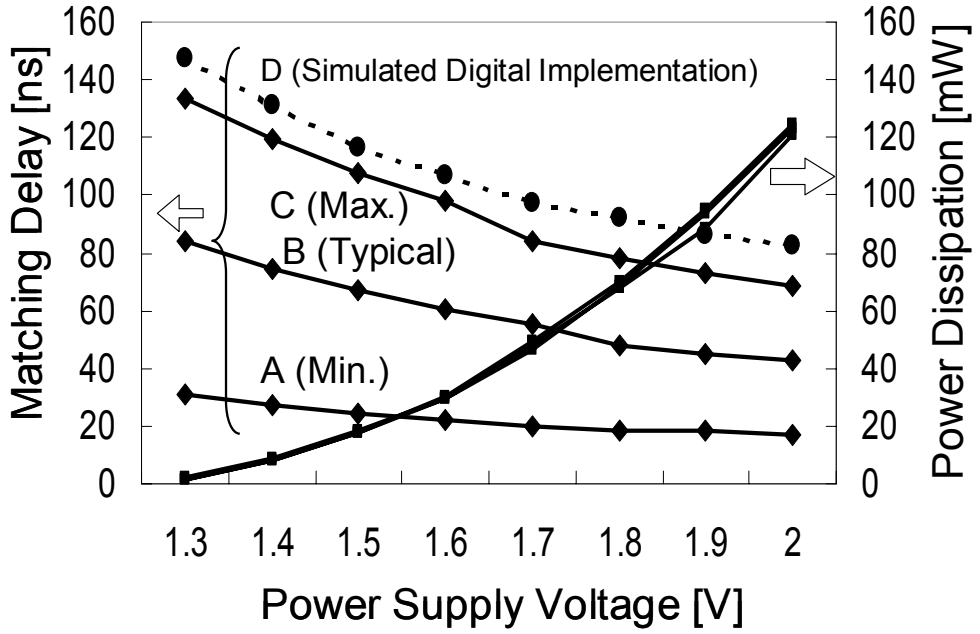


Figure 6.13. DP matching time and power dissipation as a function of power supply voltage.

VI. Conclusions

Computationally very expensive dynamic-programming (DP) matching has been implemented using delay-encoding-logic architecture in the entire processing, including element matching, best-matching-path search, and arbitrary-shaped pulse generation. In this architecture, signals in circuits are digital in the voltage domain, while analog processing is carried out in the time domain. As a result, low-power, high-speed, and compact implementation have been achieved. The voltage-scaling-compatible analog processing enables us to reduce supply-voltage and power dissipation as in conventional pure-digital processors. The typical matching time of 80ns with the power dissipation of 2mW under the 1.3V power supply has been demonstrated with the fabricated prototype chip.

CHAPTER 7.
Conclusions

I. Summary of This Thesis

In this work, aiming at variety of intelligent applications, VLSI associative processors based on vector-quantization (VQ) and dynamic-programming (DP) matching have been developed. Especially, VQ processing is general and utilized in variety of applications, thus VQ processors have been developed based on both analog and digital circuit technologies, enabling us to choose the optimum performance or functionality for each application.

An analog VQ processor featuring non-volatile analog-memory-merged matching cell has been developed. The memory-merged matching cell stores the template vector data and computes the similarity in itself. As a result, the high-density and highly parallel circuit implementation has been achieved without a bottleneck caused by a large number of memory access. The processor has 256 template vectors composed of 64 elements, and carries out fully parallel analog processing. Writing analog data into the functional

memory circuit is experimentally verified by the prototype chip. The write-and-verify scheme using hot channel electron injection provides analog-data-writing more precisely than 5mV and the range of the memorized voltage from 3V to 4.2V for analog data writing.

An analog VQ processor technology has been developed based on a bell-shape-type element matching circuit aiming at high-density integration. The bell-shape current characteristics of the matching cell are produced by only four NMOS transistors with two complementary analog signals. The matching cell developed in this work is compatible to the non-volatile analog memory, and the layout area is reduced to a quarter of the matching cell developed in the previous work. A compact digital-to-analog converter (DAC) circuit with cyclic architecture has been developed for on-chip highly parallel conversion. A single CMOS-inverter featuring a double resetting scheme was employed as its gain stage. It significantly improved the unity-gain characteristics of the CMOS inverter buffer in spite of its finite gain. The circuit ideas have been verified by measurements on experimental chips fabricated in a 0.6-μm double-poly CMOS process.

A general-purpose VQ processor featuring high-speed and versatile winner search capabilities has been developed. In order to achieve a high-speed operation, a two-dimensional bit-propagating scheme has been introduced to the winner-take-all (WTA) circuitry. As a result, the winner search is accomplished in a single clock cycle as compared to the conventional bit-sliced WTA approaches where clock cycles equal to the bit length of distance value are required. The variable-binary-block addressing scheme developed in this work allows various winner-search options, like local winner search, winner sorting and so forth. Such an addressing scheme has been implemented by adding only a single auxiliary bit to the ordinary address code. A multiplier function is also included in the SIMD distance computation unit with a minimal area penalty. As a result, weight multiplication to vector elements as well as the choice of either Manhattan

distance or Euclidian distance as the dissimilarity measure has been made possible. A prototype VLSI chip was designed and fabricated using a 0.- μm standard CMOS technology and the new concepts have been experimentally demonstrated.

Computationally very expensive dynamic-programming matching of data sequences has been directly implemented in a fully-parallel-architecture VLSI chip. The circuit operates as digital logic in the signal domain, while analog processing is carried out in the time domain based on the delay-encoding-logic scheme. As a result, high-speed low-power best-match-sequence search has been established with a small chip area. The typical matching time of 80ns with the power dissipation of 2mW has been demonstrated with fabricated prototype chips.

VLSI associative processors and their elemental circuits developed in this work would contribute to enhance performance or efficiency in the intelligent information applications. Introducing the low-cost VLSI associative processing, its applied field will be widened to resource-limited area such as mobile or ubiquitous computing. The general-purpose associative processor featuring flexibility can be frequently and aggressively utilized in general IT systems or services.

II. Future Perspectives

In this work, mainly three types of associative processor have been developed; analog VQ processors featuring highly-parallel and efficient computation, digital VQ processors featuring high flexibilities, and dynamic-programming (DP) matching processor for sequence-based matching.

In analog VQ processor, the high-density matching cell has been integrated using the analog EEPROM technology. However, the analog

EEPROM has the problem that high supply voltage is required for writing analog data. On the other hand, the ferroelectric memory may be the major candidates for this problem, and the analog associative processor would operate in lower power dissipation.

In digital associative processors, enhancement of flexibility is one of the primary concerns. The digital processors developed in this work employ SIMD architecture, however, for more flexible processing, MIMD or MSIMD architecture is promising. Even if these architectures are employed, the building blocks developed in this work, such as high-speed winner-take-all circuit, variable-binary block-addressing scheme, and the similarity evaluating datapath logic, would provides them with high flexibilities as well as in SIMD architecture.

The DP matching processor provides more robust associative processing than the VQ processor. In image recognition system, for instance, images even scaled or shifting in its position would be appropriately matched, while the conventional VQ processing would result incorrectly.

In terms of whole associative processing system, a lot of applications would benefit from the processors developed in this work, which are suitable for more wide-ranging requirements than that of the past. Furthermore, introducing software/hardware co-design associative system, such as programming languages synthesizing the associative processors automatically selected the optimum circuit technology according to requirements, would maximize the advantage of the associative processors developed in this work.

References

[1] G. E. Moore, "Progress in digital integrated electronics," in IEDM Tech. Dig., 1975, pp. 11-13.

[2] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," Bull. Math. Biophys., 5, pp. 115-133, 1943.

[3] T. Shibata and T. Ohmi, "An intelligent MOS transistor featuring gate-level weighted sum and threshold operations," in IEDM Tech. Dig., Dec. 1991, pp. 919-922.

[4] T. Shibata, "Intelligent Signal Processing Based on a Psychologically-Inspired VLSI Brain Model," IEICE Trans. Fundamentals, Vol. E85-A, No. 3, pp. 600-609, 2002.

[5] M. Adachi, and T. Shibata, "Image representation algorithm featuring human perception of similarity for hardware recognition systems," in Proc. International Conference on Artificial Intelligence (IC-AI), Jun. 2001.

[6] M. Yagi, M. Adachi, and T. Shibata, "A hardware-friendly soft-computing algorithm for image recognition," in Proc. EUSIPCO, Tampere, Sept. 2000, pp. 729-732.

[7] M. Yagi, T. Shibata, "An image representation algorithm compatible to neural-associative-processor-based hardware recognition systems," IEEE Trans. Neural Networks, Vol. 14, No. 5, pp. 1144-1161, Sept. 2003.

[8] Y. Suzuki and T. Shibata, "Multiple-clue face detection algorithm using edge-based feature vectors," in Proc. of 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. V-737 - V-740, Montreal, May 17-21, 2004.

[9] Y. Suzuki and T. Shibata, “An edge-based face detection algorithm robust against illumination, focus, and scale variations,” in Proc. of the 12th

European Signal Processing Conference, pp. 2279-2282, Vienna, Austria, September 6-10, 2004.

[10] Q.-R. Gu and T. Shibata, "A low-cost vector quantization system for voice compression based on analog and neuron MOS technology," in the Proc. of 2000 IEEE International Symposium on Intelligent Signal Processing and Systems (ISPACS 2000), pp. 222-227, Honolulu, Hawaii, U. S. A. , November 5-8, 2000.

[11] H. Xu, Y. Mita, and T. Shibata, "Intelligent Internet search applications based on VLSI associative processors," in Proc. The 2002 International Symposium on Applications and the Internet (SAINT), 2002.

[12] H. Xu, Y. Mita and T. Shibata, "Optimizing vector-quantization processor architecture for intelligent query search applications," Japanese Journal of Applied Physics, Vol. 41, Part 1, No. 4B, pp. 2295-2300, 2002.

[13] I. Vollrath, W. Wilke, and R. Bergmann, "Intelligent electronic catalogs for sales support," Advances in Soft Computing - Engineering Design and Manufacturing, 1999.

[14] A. Aamodt, and E. Plaza, "Case-based reasoning: foundational issues, methodological variations, and system approaches," in AICOM, vol. 7, no. 1, pp. 39-59.

[15] D. Leake, "CBR in context: the present and future," in Case-Based Reasoning Experiences, Lessons, & Future Directions. ed. D. Leake, Menlo Park, Calif., MIT Press, 1996.

[16] A. Gersho and R. M. Gray, "Vector quantization and signal compression," Boston: Kluwer Academic Publisher, 1992.

[17] G. T. Tuttle, S. Fallahi, and A. A. Abidi, “An 8b CMOS vector A/D converter”, ISSCC, pp. 38-39, 1993.

[18] A. Kramer, R. Canegallo, M. Chinosi, D. Doise, G. Gozzini, L. Navoni, P. L. Rolandi, and M. Sabatini, “55GCPS CAM using 5b analog flash”, ISSCC, pp. 44-45, 1997.

[19] A. A. Biyabani, L. R. Carley, T. Kanade, “An analog CMOS IC for template matching”, ISSCC, pp. 82-83, 1999.

[20] A. Nakada, M. Konda, T. Morimoto, T. Yonezawa, T. Shibata, and T. Ohmi, "Fully-parallel VLSI implementation of vector quantization processor using neuron-MOS technology," IEICE Trans. Electron, Vol. E82-C, No. 9, pp. 1730-1738, Sept. 1999.

[21] R. T. Edwards, and G. Cauwenberghs, "Mixed-mode correlator for micropower acoustic transient classification," IEEE J. Solid-State Circuits, vol. 34, no. 10, pp. 1367-1372, Oct. 1999.

[22] T. Delbruck, “Bump Circuits for computing similarity and dissimilarity of analog voltages”, IJCNN, pp. I-472-I-479, 1991.

[23] J. Choi, B. J. Sheu, and J. C.-F. Chang, “A Gaussian synapse circuit for analog VLSI neural networks”, IEEE Trans. VLSI Syst., vol. 2, pp. 129-133, 1994.

[24] M. Konda, T. Shibata, and T. Ohmi, “Neuron-MOS correlator based-on Manhattan distance computation for event recognition hardware”, IEEE International Symposium on Circuit and Systems, pp. 217-220, 1996.

[25] L. Theogarajan, and L.A. Akers, “A scalable low voltage analog Gaussian radial basis circuit”, IEEE Trans. Circuits Syst., vol. 44, pp. 977-979, 1997.

[26] S.-Y. Lin, R.-J. Huang, and T.-D. Chiueh, "A tunable Caussian/square function computation circuit for analog neural networks," IEEE Trans. Circuits Syst. II Analog and Digital Signal Processing, vol. 45, no. 3. pp. 441-446, Mar. 1998.

[27] T. Yamasaki and T. Shibata, “An analog similarity evaluation circuit featuring variable functional forms,” in Proc. 2001 IEEE International Symposium on Circuits and Systems (ISCAS 2001), Sydney, Australia, May. 6-9, 2001.

[28] T. Yamasaki and T. Shibata, "Analog soft-matching classifier using floating-gate MOS technology," IEEE Trans. Neural Networks, Vol. 14, No.

5, pp. 1257-1265, Sept. 2003.

[29] J. Lazzaro, S. Ryckebusch, M. A. Mahowald, and C. A. Mead, “Winner-take-all networks of O(N) complexity,” in Advances in neural information processing systems 1, pp. 703-711, Morgan Kaufmann Publishers Inc., 1989.

[30] T. Yamashita, T. Shibata, and T. Ohmi, “Neuron MOS winner-take-all circuit and its application to associative memory”, ISSCC, pp. 236-237, 1993.

[31] T. S.-Gotarredona and B. L-Barranco, "A high-precision current-mode WTA-MAX circuit with multichip capability," IEEE J. Solid-State Circuits, vol. 33, no. 2, pp. 280-286, Feb. 1998.

[32] B. Sekerkiran, and U. Cilingiroğlu, "A CMOS k-winner-take-all circuit with O(N) complexity," IEEE Trans. Circuits Syst. II Analog and Digital Signal Processing, vol. 46, no. 1. pp. 1-5, Jan. 1999.

[33] A. Okada and T. Shibata, "A neuron-MOS parallel associator for high-speed CDMA matched filter," in Proc. The 1999 IEEE International Symposium on Circuits and Systems (ISCAS '99), Vol. 2, Orlando, Florida, May. 30 - June 2, 1999, pp. II-392-395.

[34] J. Barnden, K. Srinivas, and D. Dharmavaratha, “Winner-take-all networks: time-based versus activation-based mechanisms for various selection goals,” in Proc. 2000 IEEE International Symposium on Circuits and Systems (ISCAS 2000), vol. 1, pp. 215-218, Geneva, Switzerland, May. 28-31, 2000.

[35] C. K. Kwon and K. Lee, "Highly parallel and energy-efficient exhaustive minimum distance search engine using hybrid digital/analog circuit technique," IEEE Trans. on VLSI Syst., vol. 9, no. 5, pp. 726-729, Oct. 2001.

[36] K. Ito, M. Ogawa and T. Shibata, "A high-performance ramp-voltage-scan winner-take-all circuit in an open loop architecture," Japanese

Journal of Applied Physics, Vol. 41, Part 1, No. 4B, pp. 2301-2305, Apr. 2002.

[37] S. Panchanathan, and M. Goldberg, "A content-addressable memory architecture for image coding using vector quantization," IEEE Trans. Signal Processing, vol. 39, no. 9, pp. 2066-2078, Sept. 1991.

[38] K. Dezhgosha, M. M. Jamali, and S. C. Kwatra, "A VLSI architecture for real-time image coding using a vector quantization based algorithm," IEEE Trans. Signal Processing, vol. 40, no. 1, pp. 181-189, Jan. 1992.

[39] R. Jain, A. Madisetti, and R. L. Baker, "An integrated circuit design for pruned tree-search vector quantization encoding with an off-chip controller," IEEE Trans. Circuits Syst. Video Technol., vol. 2, no. 2, pp. 147-157, Feb. 1992.

[40] K. E. Grosspietsch and R. Reetz, "The associative processor system CAPRA: architecture and applications," IEEE Micro, vol. 12, no. 6, pp. 58-67, Dec. 1992.

[41] K. Tsang and B. W. Y. Wei, "A VLSI architecture for a real-time code book generator and encoder of a vector quantizer," IEEE Trans. VLSI Syst., vol. 2, pp. 360-364, Sept. 1994.

[42] A. P. Chandrakasan, A. Burstein, and R. W. Brodersen, "A low-power chipset for a portable multimedia I/O terminal," IEEE J. Solid-State Circuits, vol. 29, no. 12, pp. 1415-1428, Dec. 1994.

[43] J. E. Fowler, Jr., K. C. Adkins, S. B. Bibyk, and S. C. Ahalt, "Real-time video compression using differential vector quantization," IEEE Trans. Circuits Syst., vol. 5, pp. 14-24, Feb. 1995.

[44] C.-Y. Lee, S.-C. Juan, and Y.-J. Chao, "Finite state vector quantization with multipath tree search strategy for image/video coding," IEEE Trans. Circuits Syst. Video Technol., vol. 6, no. 3, pp. 287-294, Jun. 1996.

[45] C. L. Wang and K. M. Chen, "A new VLSI architecture for full-search vector quantization," IEEE Trans. Circuits Syst. Video Technol., vol. 6, pp. 389-398, Aug. 1996.

[46] E. K. Tsern and T. H. Meng, "A low power video-rate pyramid VQ decoder," IEEE J. Solid-State Circuits, vol. 31, no. 11, pp. 1780-11794, Nov. 1996.

[47] K. Kobayashi, M. Kinoshita, M. Takeuchi, H. Onodera, and K. Tamaru, "A memory-based parallel processor for vector quantization," in Proc. 22nd ESSCIRC, Sept. 1996, pp. 184--187.

[48] K. Kobayashi, M. Kinoshita, M. Takeuchi, H. Onodera, and K. Tamaru, "A memory-based parallel processor for vector quantization: FMPP-VQ," IEICE Trans. Electron., vol. E80-C, pp. 970-975, Jul. 1997.

[49] A. Nakada, T. Shibata, M. Konda, T. Morimoto, and T. Ohmi, "A fully-parallel vector-quantization processor for real-time motion-picture compression," IEEE J. Solid-State Circuits, vol. 34, no. 6, pp. 822-830, Jun. 1999.

[50] M. Imai, T. Nozawa, M. Fujibayashi, K. Kotani, and T. Ohmi, "Fast computational architecture to decrease redundant calculations – eliminating redundant digit calculation and excluding useless data," IEICE Trans., vol. E82-C, no. 9, pp. 1707-1714, Sept. 1999.

[51] T. Nozawa, M. Konda, M. Fujibayashi, M. Imai, K. Kotani, S. Sugawa, and T. Ohmi, "A parallel vector-quantization processor eliminating redundant calculation for real-time motion picture compression," IEEE J. Solid-State Circuits, vol. 35, no. 11, pp. 1744-1751, Nov. 2000.

[52] P. L. Tai, C. T. Liu, and J. S. Wang, "A unified systolic array design for kernel functions of video compression," IEEE Trans. Circuits Syst. II Analog and Digital Signal Processing, vol. 48, no. 5. pp. 523-531, May. 2001.

[53] C.-Y. Lee, S.-C. Juan, and W.-W. Yang, "A parallel bit-level maximum/minimum selector for digital and video signal processing," IEEE Trans. Circuits Syst. II Analog and Digital Signal Processing, vol. 41, no. 10. pp. 693-695, Oct. 1994.

[54] C.-S. Lin, S.-H. Ou, and B.-D. Liu, "Design of k-WTA/sorting network using maskable WTA/MAX circuit," in Proc. 2001 International Symposium on VLSI Technology, Systems, and Applications, pp. 69-72, Hsinchu, Taiwan, Apr. 2001.

[55] T. Yamasaki, A. Suzuki, D. Kobayashi, and T. Shibata, “A fast self-convergent flash-memory programming scheme for MV and analog data storage,” in Proc. 2001 IEEE International Symposium on Circuits and Systems (ISCAS 2001), Sydney, Australia, May. 6-9, 2001.

[56] R. H. McCharles, V. A. Saletore, and D. A. Hodges, “An algorithmic analog-to-digital converter”, ISSCC, 1977.

[57] D. A. Martine, H.-S. Lee, and I. Masaki, “A mixed-signal array processor with early vision applications”, IEEE J. Solid-State Circuits, vol. 33, NO. 3, pp. 497-502, 1998.

[58] N. H. E. Weste, D. J. Burr, and B. D. Ackland, "A systolic processing element for speech recognition," in ISSCC Dig. Tech. Papers, Feb. 1982, pp. 274-275.

[59] R. A. Kavaler, M. Lowy, H. Murviet, and R. W. Brodersen, "A dynamic-time-warp integrated circuit for a 1000-word speech recognition system," IEEE J. Solid-State Circuits, vol. 22, no. 1, pp. 3-14, Feb. 1987.

[60] M. J. Irwin, "A digit pipelined dynamic time warp processor," IEEE Trans. Acoustics, Speech, and Signal processing, vol. 36, no. 9, pp. 1412-1422, Feb. 1988.

[61] C. Quenot, J. Gauvain, J. Gangolf, and J. J. Mariani, "A dynamic programming processor for speech recognition," IEEE J. Solid-State Circuits, vol. 24, no. 2, pp. 349-357, Apr. 1989.

[62] J. Takahashi, S. Hamaguchi, K. Tansho, and T. Kimura, "A modularized processor LSI with a highly parallel structure for continuous speech recognition," IEEE J. Solid-State Circuits, vol. 26, no. 6, pp. 833-843, Jun. 1991.

[63] M. Motomura, H. Yamada, and T. Enomoto, "A 2k-word dictionary search processor (DISP) LSI with an approximate word search capability," IEEE J. Solid-State Circuits, vol. 27, No. 6, pp. 883-891, 1992.

[64] A. Iwata and M. Nagata, "A concept of analog-digital merged circuit architecture for future VLSI's," IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences, vol. E79-A, no.2, pp. 145-157, Feb. 1996.

[65] M. Ikeda and K. Asada, "Time-domain minimum-distance detector and its application to low power coding scheme on chip interface," in Proc. 24th European Solid-State Circuit Conf., Sep. 1998, pp. 464-467.