

E-3-2

Eliminating Needless Calculations on Circuit Level: Most-Significant-Digit-First Digit-Serial Processing

Toshiyuki Nozawa, Makoto Imai, Kenji Mochizuki and Tadahiro Ohmi¹

Department of Electronic Engineering, Graduate School of Engineering, Tohoku University

¹New Industry Creation Hatchery Center, Tohoku University

Aza-Aoba 05, Aramaki, Aoba-ku, Sendai 980-8579, JAPAN

Phone: +81-22-217-7124, Fax: +81-22-263-9395, e-mail: nozawa@sse.ecei.tohoku.ac.jp

1. Introduction

In various kinds of information processing, a lot of compare operations are executed on results of arithmetical operations. And this type of operation, combination of arithmetical operations and comparisons, often limits the system performance. In such an operation, the outcome is very simple: "true" or "false". So precise arithmetical operations are not necessary if correct judgement is guaranteed. From this point of view, a conventional hardware carries out a large amount of needless calculations, because lower-digit-calculations which have no impact on outcome of comparison are inevitably executed. Motion estimation of video coding, for example, most of the distances are not necessary to be calculated precisely, because those distances are much larger than the minimum one and distance calculations on a few higher digits are enough to complete correct comparison. In other words, most of the calculations on lower digits are needless. However, precise and all-digit calculations are carried out in conventional hardware, deteriorating system performance.

In this paper, we will propose a novel technique to remove this wastefulness by calculating and comparing simultaneously from the uppermost digit: most-significant-digit-first (MSD-first) digit-serial processing. To evaluate effectiveness of this technique, we will also present a vector quantization (VQ) processor employing this technique.

2. Most-Significant-Digit-First Digit-Serial Processing

The MSD-first digit-serial processing is a technique to eliminate redundant lower-digit-calculations. This technique improves system performance by eliminating redundant calculation steps.

Figure 1 shows the concept of this technique. In the MSD-first digit-serial processing, digit-by-digit arithmetic operations and comparisons are simultaneously carried out from the MSD to the least significant digit (LSD) in successive digit-serial manner. If the compare unit made the final judgement (i.e. "true" or "false") after a few digit calculations and comparisons, the operations on following lower digits which do not have any impacts on the result are immediately stopped. Thus redundant calculations on lower

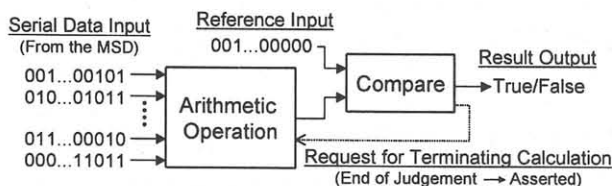


Fig. 1. Concept of the MSD-first digit-serial processing.

digits are automatically eliminated.

To realize the MSD-first digit-serial processing, it is necessary to start from the uppermost digit for calculation and comparison. However, it is impossible to start calculation from the MSD in binary number system, because carry signal might propagate boundlessly. A carry signal generated at a lower digit prevents higher digits from settling. Introduction of redundant number system can solve this problem [1]. We have developed an MSD-first digit-serial adder utilizing binary signed-digit number system (digit set: $\{-1, 0, 1\}$) in which carry propagation is limited to two digits [2]. Figure 2(a) shows the block diagram of the developed MSD-first digit-serial adder. Truth tables of each block are shown in Fig. 2(b). In Fig. 2(a), X_N , Y_N , and Z_{N+2} are binary signed-digit numbers, which are expressed by two bits of binary signal.

It is also possible for redundant number system to start comparison from the MSD. We have already developed an MSD-first digit-serial comparator using binary signed-digit number system.

Components mentioned above enables to realize the MSD-first digit-serial processing technique.

3. Vector Quantization Processor Employing MSD-First Digit-Serial Processing

We have developed a VQ processor employing the MSD-first digit-serial processing technique to evaluate its effectiveness.

VQ is a well-known technique for data compression, especially for image data [3]. The concept of VQ image compression is described below [4]. An original image is divided into small blocks of 4×4 pixels which form 16-dimensional input vectors. And typical vectors are stored as template vectors in a codebook. In VQ encoding, distances between an input vector and every template vector in the

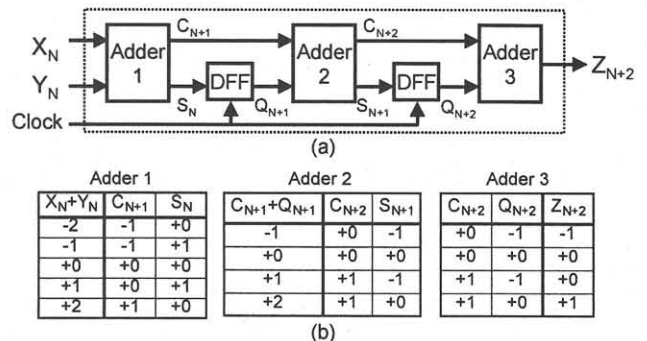


Fig. 2. MSD-first digit-serial adder employing binary signed-digit number system. (a) Block diagram. (b) Truth tables of each block.

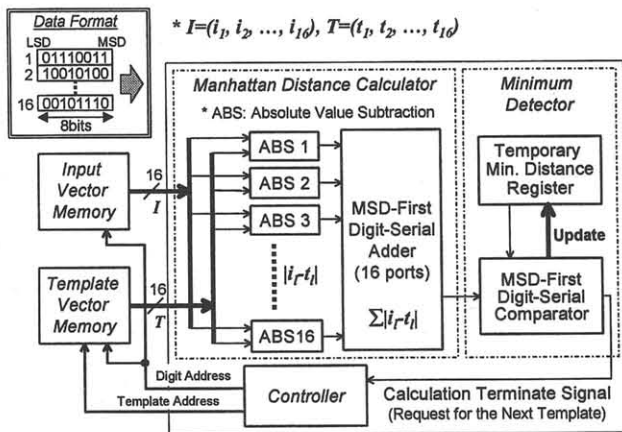


Fig. 3. Block diagram of the VQ processor employing MSD-first digit-serial processing.

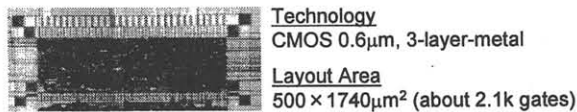


Fig. 4. Die micrograph of the VQ processor.

TABLE I FEATURES OF VQ PROCESSOR

Distance Measure	Manhattan
Input/Template Vector	16 elements, 8bits/element
Codebook Size	2048 template vectors
Max. Operating Frequency	37MHz (at 5.0V supply)
Power Dissipation	2.1mW/MHz (at 5.0V supply)

codebook are computed, and the nearest template vector to the input vector is selected out. Then, only the index of the selected template vector is outputted and transmitted. VQ is, so to speak, a variation of typical best-match operation.

The block diagram of the developed VQ processor employing the MSD-first digit-serial processing is shown in Fig. 3. VQ is carried out as follows. A vector datum is inputted to the Manhattan distance (sum of absolute values of element difference) calculation unit in digit-serial and element-parallel manner. The distance is calculated from the MSD by a single digit, and then the calculated distance datum is immediately transferred to the MSD-first comparator in MSD-first digit-serial format. Progress of distance calculation and comparison is simultaneous. After some digit calculation and comparison, the comparator outputs the signal to terminate calculation if the calculated distance is found larger than the temporary minimum distance. On the other hand, if the calculated distance is found smaller than the temporary one is, the temporary datum is updated. In this case, the distance is fully calculated to the lowermost digit. Thus the nearest template vector to the input vector is searched out.

This VQ processor was fabricated using a 0.6μm CMOS process. A die micrograph is shown in Fig. 4. Features of the chip are summarized in Table I. Figure 5 shows measured waveforms of the fabricated chip. Note that the interval for transition of the template address varies. This demonstrates that elimination of redundant lower-digit-calculations is successfully preformed. Figure 6 shows benefit brought out by elimination of lower-digit calculations on VQ image coding. Horizontal axis gives the average cycles consumed

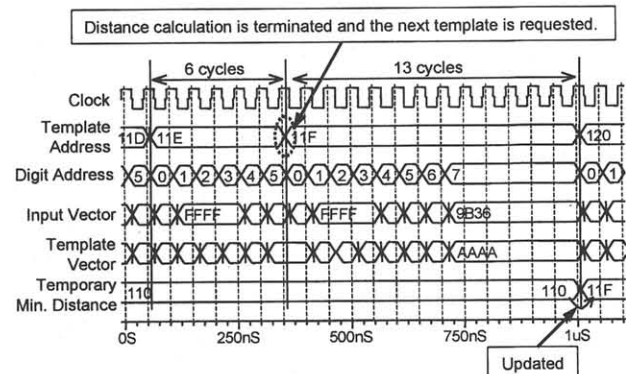


Fig. 5. Measured waveforms of the VQ processor.

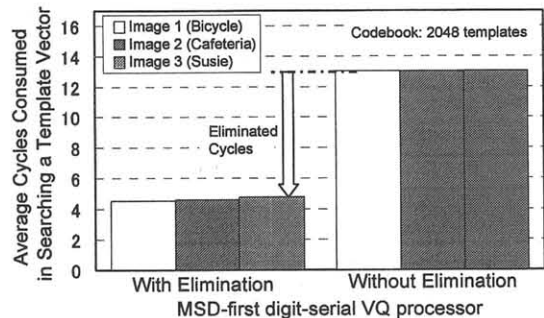


Fig. 6. Effectiveness of eliminating redundant calculations.

in searching a template vector. When elimination of calculation is carried out, 63% redundant lower-digit-calculations are eliminated on the average. In other words, processing speed improves by 2.7 times. Figure 6 clearly demonstrates the advantage of the MSD-first digit-serial processing technique.

In this section, We demonstrated the advantage of the proposed technique through a VQ processor. But application of this technique is not limited to VQ.

4. Conclusions

The MSD-first digit-serial processing technique to remove redundant lower-digit-calculations is proposed. MSD-first serial adder and comparator are developed to realize this technique. A VQ processor employing this technique has been developed to evaluate its effectiveness. Processing speed of the VQ processor improved by 2.7 times when elimination of calculations is carried out. This result demonstrates the clear advantage of the MSD-first digit-serial processing.

Acknowledgments

The chip has been fabricated in the chip fabrication program of VLSI design and education center (VDEC), the University of Tokyo with the collaboration by Rohm Co. and Toppan Printing Co

References

- [1] M.J. Irwin, IEEE Trans. Acoust., Speech, Signal Processing, **36**, pp.1412-1422 (1988)
- [2] A. Avizienis, IRE Trans. on Elect. Computers, pp.389-400 (1961)
- [3] P.C. Cosman, R.M. Gray and M. Vetterli, IEEE Trans. Image Processing, **5**, pp.202-225 (1996)
- [4] A. Nakada, T. Shibata, M. Konda, T. Morimoto and T. Ohmi, IEEE J. Solid-State Circuits, **34**, pp.822-829 (1999)