Functional-Memory Architectures for Information-Processing Systems

Hans Jürgen Mattausch, Tetsushi Koide, M.Anwarul Abedin, and Koh Johguchi
Research Center for Nanodevices Systems, Hiroshima University, Higashi-Hiroshima, 739-8527, Japan

Abstract

The analysis of information processing systems reveals that the necessary data exchange between memory and processing parts represents a major limiting factor for their performance. Important methods for substantially improving this data exchange, and therefore also the performance of the processing system, are increased memory-access bandwidth by multi-porting of the memory as well as the unification of memory and processing parts. Efficient solutions for realizing both methods are proposed.

1. Introduction

Recently, real-time and intelligent processing for multimedia data is increasingly demanded even in the consumer market [1]. Consequently, integrated solutions for related capabilities like object detection, recognition and tracking in video data or the learning function for the realization of intelligent systems become more and more important.

The basis for fulfilling these objectives is a high performance integration technology and in particular a breakthrough for mitigating the memory-access bottleneck. Here the possibilities for significant steps in this direction are explored.

2. Information processing problems from the memory viewpoint and possible solutions

A simplified view of the basic conventional system structure for information processing, which applies a separation of storing and processing unit, is shown in Fig. 1. The connection between the storing unit, or the memory, and the processing unit is normally a data link of W bit width. Figure 2 shows the processing flow with such a conventional system, which consists of 3 steps. In the first step the next W-bit data word, due for processing, is read from the memory and transferred to the processing unit via the data link. The second step consists of the actual data processing. Finally, in the third step, the processing result is transferred back via the data link and is written into the memory. Consequently, each processing operation requires 2 memory accesses in such a conventional information processing system, which means that the memory access capability is a strong performance-limiting factor of the complete information processing system.

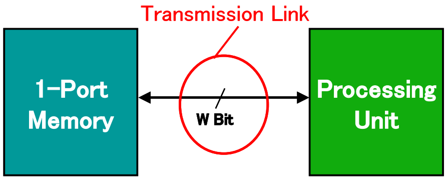

Fig. 1: Conventional structure of an information processing system with separation of storing and processing unit.

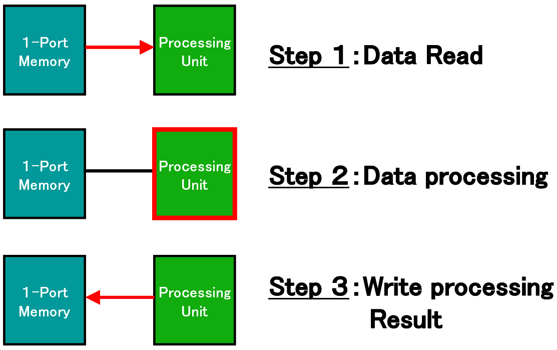

Fig. 2: Conventional data processing in 3 steps.

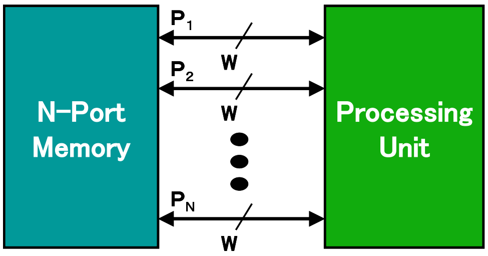

Fig. 3: Straight forward solution for increased data bandwidth between memory and processing unit.

The straight forward solution to this problem is to increase the memory-access bandwidth. This can be done by increasing the wordlength W of the link, by a higher access clock frequency (shortening the access time) or by a larger number of links and memory ports, as shown in Fig. 3. In particular, the increase of the number of memory ports is attractive, because it allows the simultaneous and random access to different data words in the memory for realizing a parallel processing approach. Achievable results with this approach will be discussed in section 3.

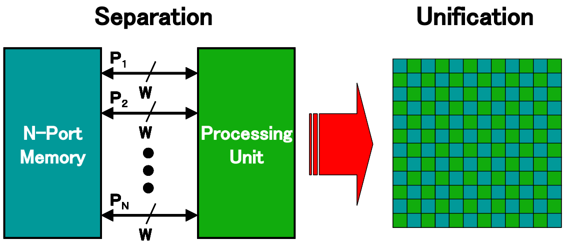

Fig. 4: Schematic view of the unification of memory and processing unit as a solution to the memory-access bandwidth bottleneck.

Another solution is to try to eliminate the data links between memory and processing unit by a unification of memory and processing unit as schematically shown in Fig. 4. In this way a very high system performance, which does not suffer from the memory bottleneck, can be expected. The application of this unification approach will be discussed in section 4.

3. Access-bandwidth improvement by increasing the number of memory ports

The straight forward approach of increasing the number of ports in each memory cell is found to be quite inefficient in terms of area consumption, which increases with the square of the port number [2]. Important negative consequences of the increased area are longer access time, higher power consumption and the difficulty to realize multi-port memories with large storage capacity.

3.1. Efficient multi-port architecture of the memory

To overcome the problems of the straight forward multi-port-cell approach, the usage of 1-port memory banks with a switching network for individually and flexibly connecting the N external ports to the 1-port banks turns out to be a useful solution. For example, the area of a 16 port SRAM with 16Kbit storage capacity can be reduced with the multi-bank approach by an order of magnitude. The required switching network for the multi-bank approach can be realized with 2 different concepts. The conventional crossbar architecture (CBA) uses a centralized crossbar switch for the interconnection purpose. The hierarchical memory architecture (HMA), which we propose, uses a distributed concept where the crossbar function is distributed to the individual banks. Figure 5 compares the transistor numbers required for HMA and CBA concepts, and verifies that for all practically relevant combinations of port and bank number the distributed HMA has a smaller transistor number than the centralized CBA [3].

3.2. Application fields and design examples

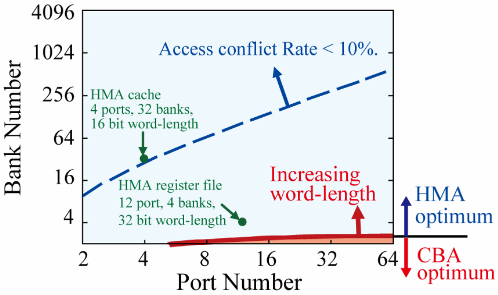

Fig. 5: Comparison of the required transistor numbers for the distributed (HMA) and the centralized (CBA) crossbar architecture. HMA has lower transistor numbers for nearly all combinations of bank and port numbers.

The conventional application fields for memories with high access bandwidth are computers and information networks. In the following we present design examples for a general purpose 16-port SRAM, for a 12-port register file and a 4-port cache memory.

3.2.1. Design example of a 16-port SRAM

Next technology-generation SRAMs will require sufficiently large static noise margin and high access bandwidth. In the reported 90nm CMOS design a high noise margin is achieved by applying a 2-port SRAM cell with separated read and write ports. For high access bandwidth 16 ports (8 read and 8 write ports) are implemented with the decentralized HMA architecture, whose hierarchically structure is further exploited to realize a high clock frequency by pipelining the access path.

Figure 6 shows the fabricated 16-port SRAM with 128Kbit storage capacity and 32bit wordlength. It is realized on an area of 0.97mm^2, has a simulated operating frequency of more than 1GHz, a power dissipation of 41mW at 1 GHz and an access bandwidth of 512 Gbps.

3.2.2. Register file and cache memory for parallel processors

The HMA multi-port memory architecture with banks consisting of 1 or 2-port memory cells has been shown to enable register file designs, which substantially reduce area consumption, power dissipation and access time in

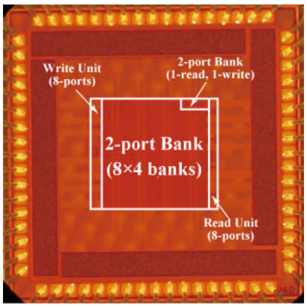

Fig. 6: Fabricated 16-port SRAM with 128Kbit storage capacity in 90nm CMOS technology.

Table I : Specification of the HMA register file test chip and comparison to a high-performance conventional register file [5].

	HMA-RF (Test chip)	HMA-RF (Estimated)	Conventional RF (ISSCC2002)
Technology	0.18 µm	0.11 µm	
Port number	12 (8r, 4w)	16 (10r, 6w)	
Register number	128	34	
Bank number	4	4	-
Word-length	32 bit	64 bit	
Storage Capacity	4096 bit	2176 bit	
Core Area	0.37 mm^2	**0.24 mm^2**	**0.5 mm^2**
Clock Frequency	580 MHz	**1140 MHz**	**545 MHz**
Power Dissipation @500MHz	220 mW	**105 mW**	**220 mW**

comparison to conventional multi-port cell register files. By avoiding register-access conflicts with techniques like register renaming, access combining, forwarding of execution unit results and out-of-order access techniques, nearly equal cycle-based execution performance in comparison to a superscalar processor with conventional multi-port-cell register file (<5% degradation) has been verified with SPECint2000 benchmark simulation [4]. Table 1 shows the comparison of a 12-port HMA register file designed in 180nm CMOS technology with a high-performance 16-port-cell register file presented at ISSCC'2002 in 110nm CMOS technology [5]. Even when designed in a technology with substantially lower basic performance, the HMA register file allows already higher clock frequency and smaller area consumption. To get a comparison based on equal technology, port number, wordlength and storage capacity, we have estimated the HMA register-file performance data in the 110nm CMOS by scaling down from the 180nm design results. As can be seen from the second column of table 1, the HMA register file is expected to increase the clock frequency by a factor 2 while reducing area consumption and power dissipation to one half of the conventional architecture.

In the case of the cache memory, the HMA architecture allows a unification of data and instruction caches, which is advantageous because dynamic allocation of the storage space for both data and instructions, and therefore optimized storage-space utilization, becomes possible [6, 7]. Longer wordlength for an instruction port is advantageous to facilitate fetching of multiple instructions in parallel processors and can also be realized easily with the HMA architecture. Figure 7 shows a 4-port cache example in 180nm CMOS technology with 128Kbit storage capacity. Included are 1 instruction port and 1 service port with 64 bit wordlength each, as well as 2 data ports with 16 bit wordlength. The area overhead for the 4 ports is only 25% and the access cycle time is 3.4nsec.

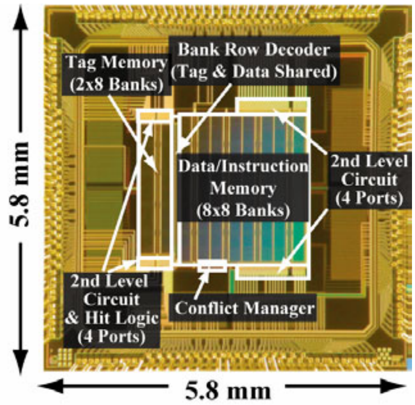
Tag Memory (2x8 Banks)
Bank Row Decoder (Tag & Data Shared)
Data/Instruction Memory (8x8 Banks)
2nd Level Circuit (4 Ports)
2nd Level Circuit & Hit Logic (4 Ports)
Conflict Manager
5.8 mm
5.8 mm

Fig. 7: HMA cache with 4-ports and 238Kbit storage capacity in 180nm CMOS technology.

4. Unification of processing and memory part

The unification of processing and memory part is a promising development direction, which can combine high performance of an integrated system with high integration density and low power consumption. This can be seen in the following example of the basic operation for realizing the recognition function, which is the determination of the most resembling among stored reference patterns in comparison to an input pattern. This so-called pattern-matching operation is difficult to implement with a fast parallel-comparison capability on the basis of small-size low-power hardware.

We have developed an integrated-circuit solution for this problem on the basis of mixed digital-analog circuitry (see Fig. 8), which unifies the memory and the processing part in the form of an associative memory [8, 9]. The comparators for the reference patterns are integrated in digital form into the memory. The distance-measure results are digitally determined and transformed into analog signals for each reference pattern in parallel. The minimum-distance or winner search is carried out again in parallel with a fast analog search circuit. Circuits for the distance calculation differ according to the distance measure and circuit versions for the Hamming-,

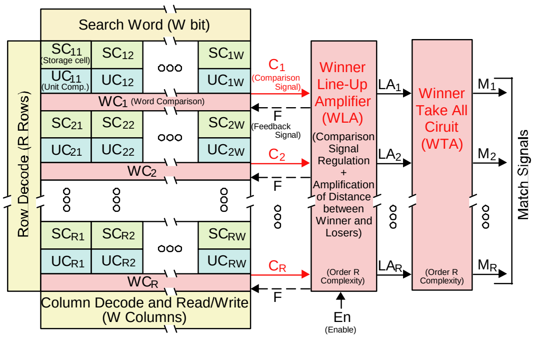

Fig. 8: Architecture diagram of the mixed digital-analog associative memory.

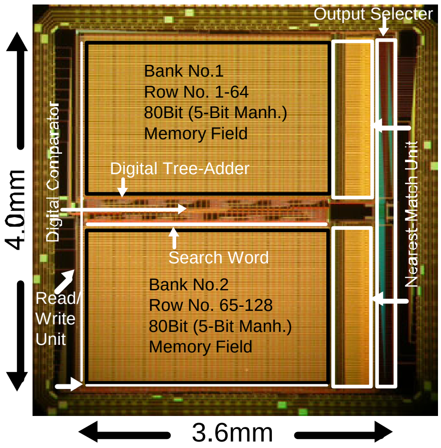

Fig. 9: Manhattan-distance design example in 350nm CMOS technology with 2 banks of 64 patterns each.

Manhattan- and Euclidean distance [10, 11] have been developed. The performance of CMOS test-chip designs verified extremely fast winner-search times below 200ns at a low power dissipation of less than 180mW. Figure 9 shows the example of a 2-bank design in 350nm CMOS technology for the Manhattan-distance measure with 128 patterns, each consisting of a 16 dimensional vector having 5bit components. In Fig. 10 a test chip with 64 reference patterns is depicted, which realizes the worldwide first fully-parallel Euclidean-distance-search hardware.

5. Conclusion

In this report the structure of information processing systems has been analyzed from the memory point of view. It has been shown that the increase of the memory-access bandwidth is a prerequisite to high performance systems and the realization of intelligent processing function. The two main possibilities for increasing the memory-access bandwidth sufficiently are the multi-port memory approach and the unification approach of memory and processing unit. The multi-port approach requires avoiding the usage of multi-port storage cells and leads to a multi-bank architecture, which is efficient to improve in particular multi-purpose processor applications like register file and cache. The unification approach can lead to high performance, high density solutions and is demonstrated by the realization of fully-parallel associative-memory circuits.

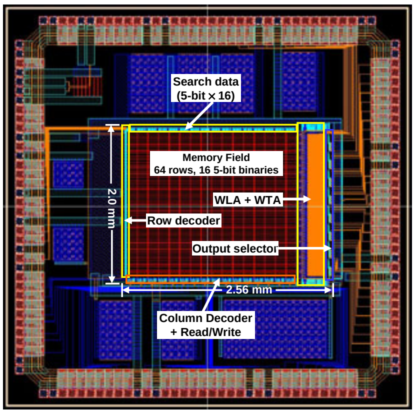

Fig. 10: Euclidean-distance design example in 350nm CMOS technology with 64 patterns each.

References

[1] *L. C. Jain, U. Halici, I. Hayashi, S.B. Lee, and S. Tsutsui, "Intelligent Biometric Techniques in Fingerprint and Face Recognition," CRC Press, 1990.*

[2] *Y. Tatsumi and H. J. Mattausch, "Fast quadratic increase of multiport-storage-cell area with port number," IEE Electronics Letters, Vol. 35, No. 25, pp. 2185-2187, 1999.*

[3] *S. Fukae, T. Inoue, H. J. Mattausch, T. Koide and T. Hironaka, "Distributed against centralised crossbar function for realizing bank-based multiport memories," IEE Electronics Letters, Vol. 40, No. 2, pp. 101–102, 2004.*

[4] *T. Saito, M. Maeda, T. Hironaka, K. Tanigawa, T. Sueyoshi, K. Aoyama, T. Koide, H. J. Mattausch, "Design of superscalar processor with multi-bank register file," Proc. of ISCAS2005, pp. 3507-3510, 2005.*

[5] *Z. Zhu, K. Johguchi, T. Hirakawa, T. Koide, T. Hironaka and H. J. Mattausch, "A novel hierarchical multi-port cache," Proc. of ESSCIRC2003, pp. 405–408, 2003.*

[6] *K. Johguchi, Z. Zhu, T. Hirakawa, T. Koide, T. Hironaka and H. J. Mattausch, "Distributed-crossbar architecture for area-efficient combined data/instruction caches with multiple ports," IEE Electronics Letters, Vol. 40, No. 3, pp. 160–162, 2004.*

[7] *N. Tzartzanis, W. W. Walker, H. Ngyuyen, A. Inoue, "A 34Word x 64b 10R6W write-through self-timed dual-supply-voltage register file," Dig. Of Tech. Papers, ISSCC 2002, pp. 416-417, 2002.*

[8] *H.J. Mattausch, T. Gyohten, Y. Soda and T. Koide, "An Architecture for Compact Associative Memories with Deca-ns Nearest-Match Capability up to Large Distances", IEEE International Solid-State Circuits Conference Digest of Tech. Papers (ISSCC'2001), pp. 170-171, 2001.*

[9] *H.J. Mattausch, T. Gyohten, Y. Soda and T. Koide, "Compact Associative-Memory Architecture with Fully-Parallel Search Capability for the Minimum Hamming Distance", IEEE Journal of Solid-State Circuits, 37, pp. 218-227, 2002.*

[10] *Y. Yano, T. Koide, and H.J. Mattausch, "Associative Memory with Fully Parallel Nearest-Manhattan-Distance Search for Low-Power Real-Time Single-Chip Applications", Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC'2004), pp. 543-544, 2004.*

[11] *M. A. Abedin, K. Kamimura, A. Ahmadi, T. Koide, H. J. Mattausch, "Minimum Euclidean Distance Associative Memory Architecture with Fully-Parallel Search Capability", Workshop on Synthesis And System Integration of Mixed Information Technologies (SASIMI'2006), pp. 350-354, 2006.*