# Unified Data/Instruction Cache with Bank-Based Multi-Port Architecture

Koh Johguchi[1], Zhaomin Zhu[1], Hans Jürgen Mattausch[1], Tetsushi Koide[1] and Tetsuo Hironaka[2]

[1]Hiroshima Univ., Research Center for Nanodevices and Systems,1-4-2 Kagamiyama, Higashi-Hiroshima, 739-8527, Japan
[2]Hiroshima City Univ., Faculty of Computer Sciences, 3-4-1 OzukaHigashi, AsaMinami-Ku, Hiroshima, 731-3194, Japan
Phone: +81-82-424-6265, FAX : +81-82-424-3499, E-mail: {jouguchi, zzm, koide, hjm}@sxsys.hiroshima-u.ac.jp

## 1. Introduction

Modern processors simultaneously fetch, decode and execute many instructions. This results in the demand for a large access bandwidth of the processor's memory components and has already led to register files with many ports. However, for the cache memory the conventional solution of 1-port-data and instruction caches is still in use. Since the demand for parallelism tends to increase at a high rate, the cache system will become the bottleneck of processor performance and has to be innovated.

We propose to improve the access bandwidth of the cache with a 1-port-bank-based Hierarchical Multi-port memory Architecture (HMA), which can simultaneously realize small area, high bandwidth and low power dissipation [1, 2]. Moreover, by combining instruction and data caches into a single multi-port cache, we are able to dynamically schedule the memory amount used for data and instructions, resulting in a more efficient usage of the caches storage capacity.

## 2. Multi-port Cache with HMA Structure

HMA is a 1-port-bank based multi-port memory architecture which further improves area consumption and parformance of the conventional crossbar architecture. The crossbar's switch network is distributed into the bank structure, which decreases global wiring and transistor number. A two dimentional bank decoder reduces the overhead for bank selection and allows easy matrix arrangement of the banks [1].

Fig. 1 shows the structure-example of a direct mapped cache which uses HMA [2]. The cache index, consisting of line number (LN) and line offset (LO), is divided into two portions, a bank internal address (BI) and a bank number (BN). BI is used for selecting a cache word or tag within memory banks, and BN is used for selecting the respective banks within data/instruction or tag memory. BN uses the lower rank bits in order to make sure that consecutive lines and words within lines are located in different banks, so that they can be accessed in parallel without access conflict.

## 3. Combination of Instruction and Data Cache

### *3.1. Advantages of a Unified Cache*

Using the proposed HMA cache, instruction and data cache can be unified without loss in access bandwidth, but with the advantage of a lower miss rate at the same storage capacity. On the other hand, using a bank-based multi-port cache, access to one bank is restricted to 1 port, and access-conflict rate may increase.

The miss-rate advantage of the unified cache and the required number of banks for sufficiently small access-conflict rate are examined for the example of a 4-way superscalar processor. The simulation is carried out with a modified version of SimpleScalar [3]. Dhrystone and SPEC95 (gcc, ijpeg, etc.) are used as benchmarks.

The results for splitted and unified direct-mapped cache are shown in Figs. 2 and 3. The storage capacity values in the figures show the total capacity of instruction and data cache, being the same for splitted and unified cache. According to Fig. 2, in the case that sufficient capacity is not prepared, the unified cache has higher miss rate than the splitted cache because data rewriting takes place frequently. However, if the total miss rate becomes lower than 10%, as required in real processors, the miss rate of the unified cache is clearly lower than that of the splitted cache. Moreover, it turns out that the miss rate of the unified cache is approximately equal to that of a splitted cache at 25% reduced storage capacity. We conclude from the result of Fig. 3, that the access-conflict rate becomes sufficiently low when more than 16 banks are provided.

### *3.2. Optimum Combination of Instruction and Data Cache with HMA Structure*

Nomally the accesses to the instruction cache are consecutive. For a 4-way superscalar processor, it is therefore expected, that one instruction port with 4-time larger word length will deliver sufficient instruction-fetch performance. The optimum number of data-access ports is estimated to be 2 or 3.

Above considerations suggest that an optimized unified data/instruction cache should have different word length for data and instruction ports. Fig. 4 shows our HMA proposal of a unified write-through cache with 2 data ports and 1 instruction port, with 4 times larger word length, for 4-way superscalar processors. Although, it uses internally only a 1-to-3 port convertor with a relatively small area-overhead, the externally available access bandwidth correspounds to 6 ports, due to the 4 times increased word length of the instruction port.

## 4. Test-Chip Design for an HMA Cache

For the test chip of an HMA cache memory, a configuration with 4 ports was chosen and the design was carried out in a 5 metal, 0.18μm CMOS technology. The chip-layout shown in Fig. 5 contains all needed new functional units. The design data are summarized in Table I. Small area and short delay are achieved with a dynamic CMOS circuit technology and effective floor planning. The area-overhead of the 1:4-port convertor for the 1Kbyte bank of Fig. 6 is less than 25%. We also applied a new access method which overlaps bank-conflict management and bank decoding with the precharging phase of the banks. As a result, bank-access time, complete cache-access time and power dissipation are 1.9ns, 3.8ns and 247mW at 250MHz, respectively, as determined with layout-based simulation.

## 5. Conclusions

In this paper, a bank-based unified data/instruction cache with multiple ports has been proposed and the advantages have been verified by simulation. Especially important is our method of providing a different word length for data and instruction ports, which takes advantage of the internal bank structure. To minimize bank conflicts, we use an addressing method, which insures that the words in one cache-line and also consecutive cache-lines are located in different banks. A test-chip design of a 4-port bank-based cache in 0.18μm CMOS technology showed, that the area-overhead for the 4 ports is about 25%. A minimum clock cycle time of 3.8 ns could be achieved with a dynamic CMOS circuit technology and by overlapping the external bank access with the bank-internal precharge.

The proposed bank-based multi-port cache is also very attractive for low power dissipation, because the number of activated banks, determining power dissipation, correponds to the port number and is independent of the total number of banks in the cache.

## Acknowledgements

This research is sponsored by Semiconductor Technology Academic Research Center (STARC).

The VLSI chip in this study has been fabricated in the chip fabrication program of VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Hitachi Ltd., Dai Nippon Printing Corporation and Cadence Design Systems, Inc.

This work was supported in part by the 21st Century COE program and a JSPS research fellowship, No.1605246, Japanese Government.

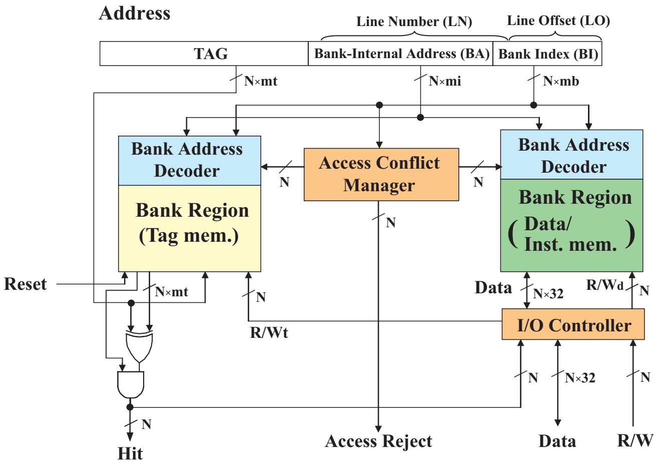


Fig. 1 Block diagram of direct mapped data cache with N ports in bank-based HMA structure.

Table I Datasheet of the design.

| | |
|---|---|
| Technology | 180nm logic CMOS, 5 Al layers |
| Si-Area | 6.2 mm$^2$ |
| Total Storage Capacity | 20.5 KByte |
| Port Number | 4 ports |
| Minimum Cycle Time | 3.8 nsec |
| Power Dissipation | 247mW at 250MHz |
| Instruction & Service Port | |
| Port Number | 2 |
| Wordlength | 64 bit |
| Data Ports | |
| Port Number | 2 |
| Wordlength | 16 bit |
| Tag Memory | |
| Storage Capacity | 4.5 KByte |
| Bank Number | 16 |
| Bank Capacity | 2304 bit |
| Data/Instruction Memory | |
| Storage Capacity | 16 KByte |
| Bank Number | 64 |
| Bank Capacity | 2 Kbit |

## References

[1] S. Fukae, N. Omori, H. J. Mattausch, T. Koide, T. Inoue and T. Hironaka, *Optimized Bank-Based Multi-Port Memories through a Hierarchical Multi-Bank Structure, Proc. of SASIMI2003*, pp.323-330, 2003.

[2] H. J. Mattausch, K. Kishi and T. Gyoten, *Area-effcient multi-port SRAMs for on-chip data-storage with high random access bandwidth and large storage capacity, IEICE Trans.Electron., Vol.E84-C, No.37*, pp.410-417, 2001.

[3] D. Burger, T. Austin, *The SimpleScalar tool set Version 2.0, Univ. of Wisconsin-Madison Computer Sciences Department Technical Report #1342*, 1997.

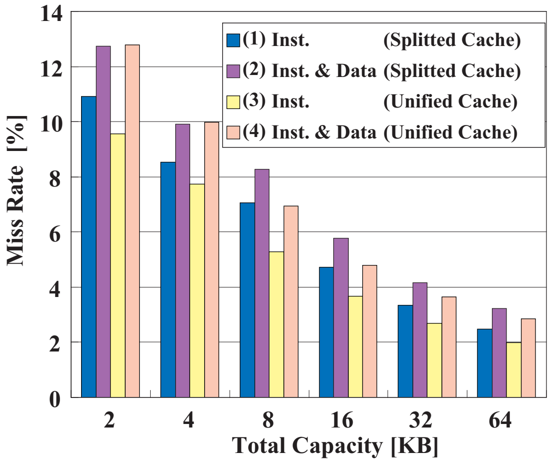


Fig. 2 Miss rate of splitted and unified direct-mapped cache.

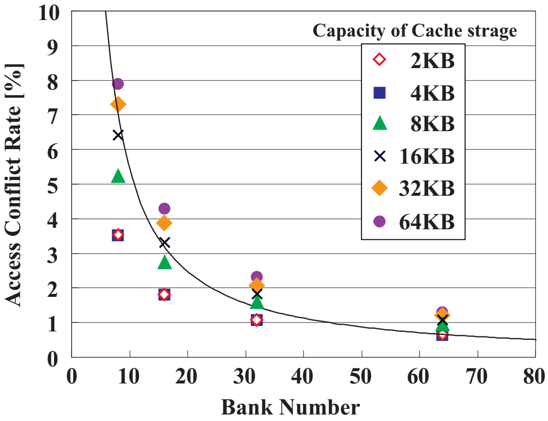


Fig. 3 Access conflict rate of unified cache.

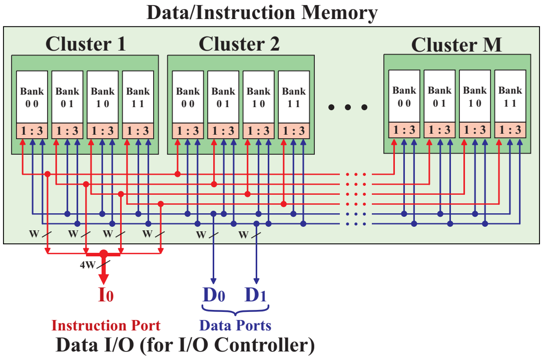


Fig. 4 Proposal of a unified data/instruction cache for a 4-way superscalar processor. The instruction port has 4 times the word length of the 2 data ports.

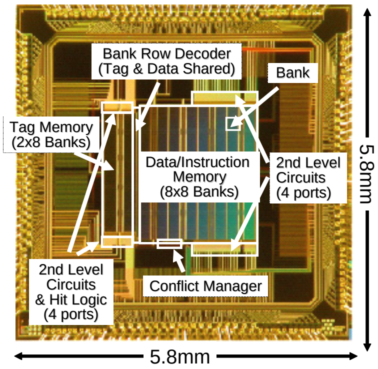

Bank Row Decoder (Tag & Data Shared)
Bank
Tag Memory (2x8 Banks)
Data/Instruction Memory (8x8 Banks)
2nd Level Circuits (4 ports)
2nd Level Circuits & Hit Logic (4 ports)
Conflict Manager
5.8mm
5.8mm

Fig. 5 Microphotograph of the test chip for 4-port cache.

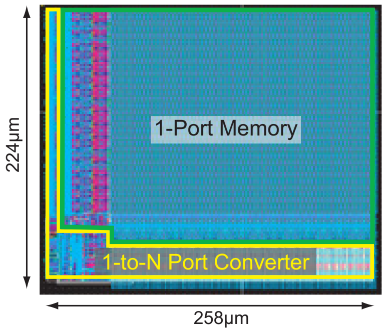


Fig. 6 Layout of a bank with 1:4-port convertor.

NTIP

Hiroshima University
Nanoelectronics for Tera-Bit
Information Processing

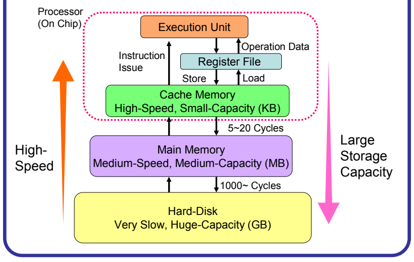
# Unified Data/Instruction Cache with Bank-Based Multi-Port Architecture

○Koh Johguchi[1], Zhaomin Zhu[1], Hans Jürgen Mattausch[1], Tetsushi Koide[1], Tetsuo Hironaka[2]

[1] Research Center for Nanodevices and Systems (RCNS), Hiroshima University
[2] Department of Computer Engineering, Hiroshima City University

## Background & Motivation

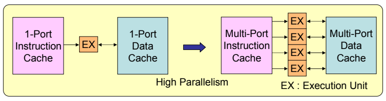
### Memory Hierarchy of Processors

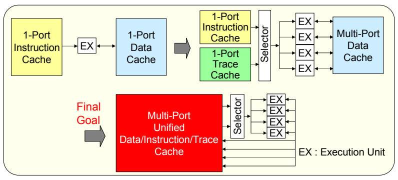


### Multi-Port Cache Applications

Multiple-Issue Processor ⇨ Instruction Cache : Multiple instruction fetches per cycle
Data Cache : Multiple load/store operations per cycle

High performance multi-port cache is needed.





**Motivation**
Designing multi-port cache memories
with small-area and high-performance

Combination of Instruction and Data Caches ⇨ Flexible cache-line usage
Reduced miss rate and storage capacity

## Special Issues of Conventional Multi-Port Cache & Proposed Solution

### Area Efficiency Problem

**Multi-Port SRAM Cell**

- Multi-port switches are connected directly to SRAM cells.
- Area increase is proportional to square of port number.
- Enough capacity for low miss rate cannot be achieved.

**Multi-Bank Type Multi-Port Memory**

- Multi-port switches are connected to banks with 1-port SRAM cells.
- High area efficiency, because of 1-port SRAM cells.
- High access conflict rate, because of arbitration.

High performance multi-port cache request :
Enough bank number is necessary
for bank structure with high area efficiency

### Solution for Multi-Port Caches

(1) Low area efficiency

- Application of 1-port SRAM bank based multi-port memory (HMA).
- Dynamic CMOS technology can be used for port-converters and conflict manager.

(2) Increase of delay time for multi-port memory

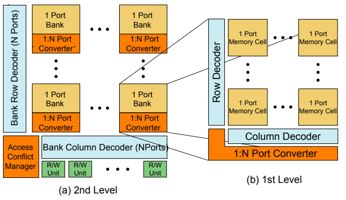
- Synchronous 2-stage access mode for hiding precharge phases.

(3) Difference of the access pattern between data and instruction caches (in order to unify of two caches)

- Development of a new multi-port cache with the port-dependent word-length and interleaved cache-line words

## Multi-Port Cache with Hierarchical Multi-Port Architecture (HMA)

### Hierarchical Multi-port memory Architecture (HMA)

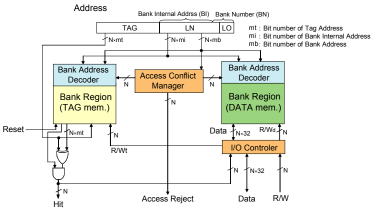


(a) 2nd Level (b) 1st Level

- 1:N port converter is integrated into the banks to achieve modularity.
- Bank Decoders are two dimensional.

⇨ Applicable to for large bank numbers

### HMA Application of Multi-Port Caches

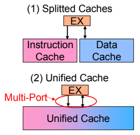


- Tag memory and data memory both use HMA.
- Bank number is determined by the lower bits of the index, so that conflicts will not take place when addresses are consecutive.

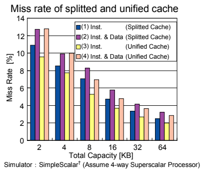
## Unified Data/Instruction Cache with HMA

### Simulation

Motivation : By using a multi-port cache, a instruction and data cache can be unified.



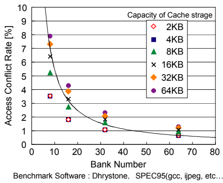
Miss rate of splitted and unified cache



Simulator : SimpleScalar† (Assume 4-way Superscalar Processor)
Benchmark Software : Dhrystone, SPEC97(gcc, etc…)

- In the case that the total miss rate becomes lower than 10%, the miss rate of the unified cache is clearly lower than that of the splitted cache.
- It turns out that the miss rate of the unified cache is approximately equal to that of a splitted cache at 25% reduced storage capacity.

†SimpleScalar © Todd Austin ( URL : http://www.simplescalar.com )



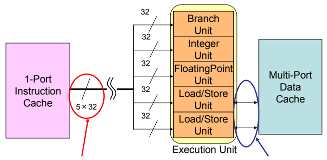
Benchmark Software : Dhrystone, SPEC95(gcc, ijpeg, etc…)

When access conflict rate and miss rate are the same, performance degradation due to bank structure becomes negligible, because access-conflict penalty is only 1 clock.

⇨ Access conflict rate becomes sufficiently low when more than 16 banks are provided.

### Optimized Structure

#### Wordlength Selection for Data and Instruction Port



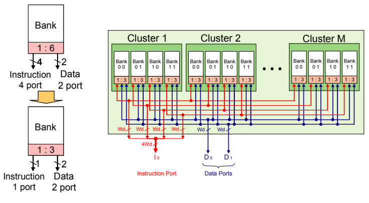
Instructions are ordered mainly sequentially.
⇩
1-port with increased wordlength is sufficient.

Data is ordered mainly not sequentially.
⇩
Wordlength cannot be increased and each port needs random access capability.

Multiple issue processor require different number and wordlength for instruction and data ports.

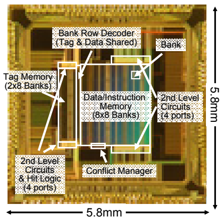
#### Optimized Unified Cache for 4-way Superscalar Processor

Average instruction number issued in parallel : 2.5-2.8 ⇨ 2 data ports and 1 instruction port with increased wordlength are sufficient



## Test Chip Design

### 4-Port Unified Data/Instruction Cache Test Chip



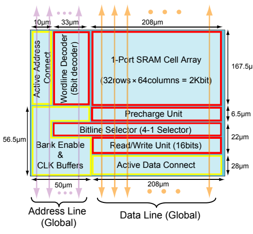
Test Chip Specification

| | |
|---|---|
| Technology | 180nm logic CMOS, 5 Al layers |
| Si-Area | 6.2 mm$^2$ |
| Total Storage Capacity | 20.5 KByte |
| Port Number | 4 ports |
| Minimum Cycle Time | 3.8 nsec |
| Power Dissipation | 247mW at 250MHz |
| Instruction & Service Port | |
| Port Number | 2 |
| Wordlength | 64 bit |
| Data Ports | |
| Port Number | 2 |
| Wordlength | 16 bit |
| Tag Memory | |
| Storage Capacity | 4.5 KByte |
| Bank Number | 16 |
| Bank Capacity | 2304 bit |
| Data/Instruction Memory | |
| Storage Capacity | 16 KByte |
| Bank Number | 64 |
| Bank Capacity | 2 Kbit |

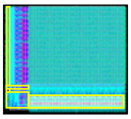
Using HMA and dynamic CMOS technology,
we have achieved high speed and area-efficiency, simultaneously.

### Layout Design of Bank (1st Level)

Floor Plan

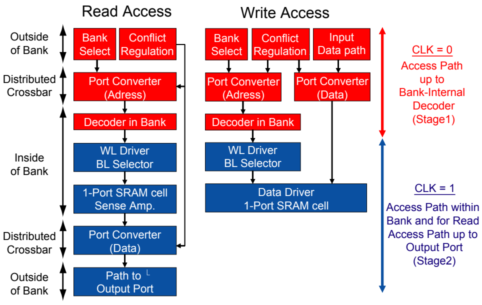


Layout Result



Height = 224 μm
Width = 258 μm
The area-overhead for multi-port capability is about 16%!!
(Considering only 1st level)

## New Access Method for Multi-Port Memory



CLK = 0
Access Path up to Bank-Internal Decoder (Stage1)

CLK = 1
Access Path within Bank and for Read Access Path up to Output Port (Stage2)

## Conclusions

- Test-chip design of a unified data/instruction HMA cache with 4 ports, 16KByte and 64 data-banks.
  - Cycle time is 3.9ns and area overhead for multi-port capability is less then 25%.
- Unified data/instruction cache advantages.
  - Reduced miss rate
  - High area efficiency
- Concept proposal for additional unification of trace cache with data/instruction cache.