A 45-nm 37.3 GOPS/W Heterogeneous Multi-Core SOC with 16/32 Bit Instruction-Set General-Purpose Core

Osamu NISHII[†a)], *Member*, Yoichi YUYAMA[††], Masayuki ITO[††], Yoshikazu KIYOSHIGE[††], Yusuke NITTA[††], *Nonmembers*, Makoto ISHIKAWA[†††], Tetsuya YAMADA[†††], *Members*, Junichi MIYAKOSHI[†††], Yasutaka WADA[††††], *Nonmembers*, Keiji KIMURA[††††], Hironori KASAHARA[††††], *Members, and* Hideo MAEJIMA[†††††], *Fellow*

SUMMARY   We built a 12.4 mm × 12.4 mm, 45-nm CMOS, chip that integrates eight 648-MHz general purpose cores, two matrix processor (MX-2) cores, four flexible engine (FE) cores and media IP (VPU5) to establish heterogeneous multi-core chip architecture. The general purpose core had its IPC (instructions per cycle) performance enhanced by adding 32-bit instructions to the existing 16-bit fixed-length instruction set and executing up to two 32-bit instructions per cycle. Considering these five-to-seven years of embedded LSI and increasing trend of access-master within LSI, we predict that the memory usage of single core will not exceed 32-bit physical area (i.e. 4 GB), but chip-total memory usage will exceed 4 GB. Based on this prediction, the physical address was expanded from 32-bit to 40-bit. The fabricated chip was tested and a parallel operation of eight general purpose cores and four FE cores and eight data transfer units (DTU) is obtained on AAC (Advanced Audio Coding) encode processing.
key words: heterogeneous, instruction set, MMU

1. Introduction

The recent trend toward integration (within a single chip) trend for an embedded SOC (system on chip) results in common characteristics for multiple chips. First, an SOC chip can hold multiple CPU cores, and this is a basic performance improvement means. Within this trend, homogeneous multi-core designs have been reported [1], [2]. Second, accelerators have evolved and they are placed as an application-specific core with current integration [3]. When we take a look at processing overall, we see that not all processing is general-core processing, in many target cases. In some embedded chips, most of the portion of chip's processing of a whole chip is media processing (audio data, still-image, video data). A feature for such processing is that the width of each data is less than 32 bits — i.e., it is 8 bits or 16 bits. In these cases, the amount of data is large and the processing is uniform for each element.

In this embedded processing trend, the integration of
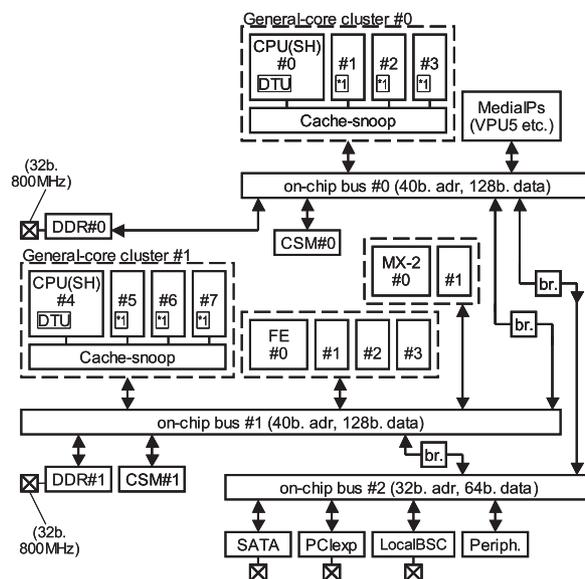
Fig. 1   Chip block diagram.

general-purpose cores and special-purpose cores is a focus in the development of future SOC. We have developed a heterogeneous multi-core chip that aims to be the base platform for present and future embedded systems and the parallel software. Here the word "parallel" is meant in two ways: (i) parallel within homogeneous cores, and (ii) parallel between heterogeneous cores.

The chip block diagram, the chip micrograph and the specifications are shown on Figs. 1 and 2 and Table 1, respectively. The total performance of eight general cores reaches 13 GIPS (giga instructions per second), and it is equal to about 2.5 times that of the chip presented in 2007 [4], [5]. This performance gain is divided as 2.0 (number of general core), 1.08 (frequency), and 1.16 (IPC: instructions per cycle).

The total performance of special-purpose cores (MX-2 and FE) is 78 GOPS (here operation count is normalized to 32-bit operation).

In this paper, Sect. 2 provides a summary of previously developed multi-core chips and their technical points, those created the basis of the multi-core LSI. This section

Fig. 2 Chip micrograph.

Table 1 Chip specifications.

Geneal purpose cores	Eight SH-4As, 648MHz
Special purpose cores	Foue FEs (flexible engine) 324MHz, Two MX-2s (matrix processors) 324MHz Media IPs (VPU5)
Process technology	45nm LP CMOS, 8 metal layers, triple Vth
Chip size	12.4 mm x 12.4 mm (effective area is 11.0 mm x 11.0 mm including PAD)
Supply voltage	1.0 – 1.2V (internal) 1.2V – 3.3V (I/O)
Power consumption	3.07W (general-purpose core 648MHz, 1.15V)
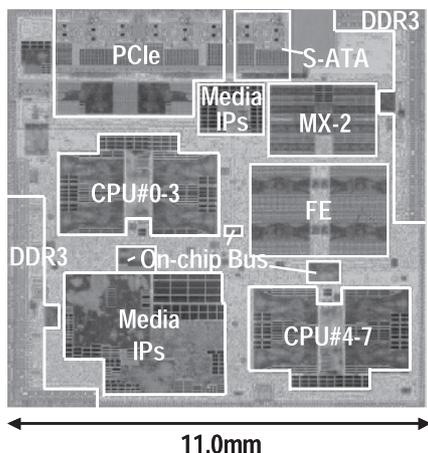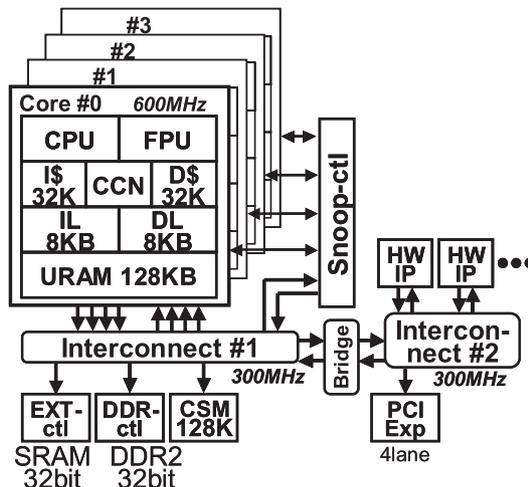
Fig. 3 Block diagram of the first LSI [4], [5], [7].

Fig. 4 Block diagram of the second LSI [8].

is placed so that this paper summarizes chips supported by NEDO (New Energy and Industrial Technology Development Organization, of Japan). Section 3 describes the top-level structure of the newly developed heterogeneous multi-core chip. Section 4 describes the performance enhancements added to the general-purpose core. Section 5 describes the specification and features of special-purpose cores integrated within this chip. Section 6 describes the implementation of the chip and evaluation results.

Although this paper mainly focuses LSI hardware, the software execution environment is also important to use these heterogeneous processing cores efficiently. To cover efficient parallel processing for heterogeneous chip architectures, an API (application program interface) for a parallelizing compiler has been extended to support the heterogeneous cores [6].

2. Previous Work

We developed two multi-core chips. At the first LSI (Fig. 3) [4], [5], [7], we enhanced a cache snoop mechanism to enable AMP/SMP (asymmetric multi-processing/symmetric

multi processing) both type multi-core operation. With this enhancement, four cores could operate at different frequencies with cache coherency maintained.

The second LSI (Fig. 4) [8] has three main three features. Firstly, the number of cores is enhanced from four to eight, and since the cache coherency structure is maintained in a "cluster" (four cores), eight cores are placed as two clusters. The second enhancement is a power gating mechanism for each individual core and individual RAM (128 KB each). The power domain is divided into 17 power domains, and unnecessary power supply can be cut off by an integrated switch that minimizes the current leakage, lowering the total processing load. The third feature is a low-cycle inter-CPU barrier synchronization.

3. Chip Top-Level Structure

According to the policy described at Sect. 1, our goal was to integrate general-purpose cores and special-purpose cores. The general cores included this chip are eight general cores. In a cluster, four cores are connected with cache coherency maintained, with the same structure as previous 90-nm chip [8]. As the special purpose core, two MX-2 (matrix type accelerator) cores which features four-bit processing ele-

ment, and four FEs (flexible engine) cores which features 16-bit processing element, and one media processing hardware VPU5 (video processing unit 5) are integrated on this chip.

Both, general-purpose and special-purpose cores have the following features:

(i) Chip addressing: To enable large memory capacity operation, a 40-bit on-chip address is used. IPs: general-purpose cores, DTU (data transfer unit that is capable of address translation), and DMAC (direct memory access controller) support this 40-bit physical address. Using a dual DDR-3 interface, each can control up to 2 GB of DDR memory, this chip has a total of 4 GB of main memory (DDR: double data rate).

(ii) The control between special-purpose cores and general-purpose cores: The special purpose and the general-purpose cores shares data in DDR (DDR#0, DDR#1) and CSM (central shared memory) (CSM#0, CSM#1). The special-purpose cores can be activated by any general-purpose core. The maximum frequency is

- 648 MHz for general-purpose cores,
- 324 MHz for special-purpose cores (MX-2, FE, VPU) and on-chip bus #0, #1
- 162 MHz for on-chip bus #2.

The whole-logic scheme above has a synchronous timing design.

The maximum DDR3 IO is 800 bps (800 MHz), and this IO logic is designed for a different synchronous timing domain.

The SATA (serial ATA) interface is integrated mainly to interface storage. The PCI express interface is integrated to interface other logic LSI.

This chip does not support the multi-power domain structure [6]: because the current leakage of used 45-nm LP CMOS process was lower than that of 90-nm CMOS used for previous two chips, the necessity to suppress current leakage became low.

4. Enhancements of a General-Purpose Core

Considering the functionality of an efficient general-core with a current integration constraint, we enhanced the general-core. The major enhancements were (i) instruction set enhancement for a larger IPC (instructions per cycle) index, and (ii) physical address enhancements into 40 bits.

4.1 IPC Performance

The recent multi-core trend achieves the performance gains by increasing the number of processors, but the single-core performance also contributes to the total performance.

The existing core (SH-4A) instruction set [9] features 16-bit fixed-length instructions that were designed to enable a higher code density compared to many RISC (reduced instruction set computer) 32-bit fixed-length instructions. This 16-bit instruction set thus has a high code density, but it has a side effect that a total operand combination (or a total instruction combination) is narrower than that of a 32-bit in-
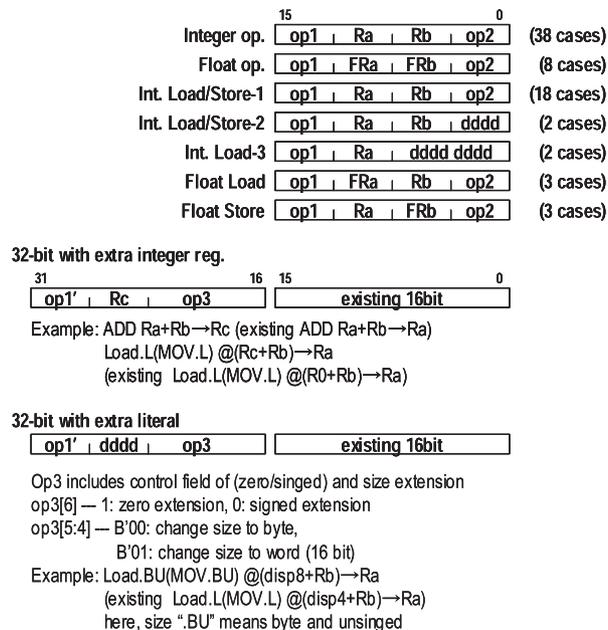
Fig. 5 Enhanced instructions of general-purpose core.

struction set. This instruction set has two register operands (register number field), whereas most 32-bit instruction sets have three register operands. Another point is that the length of displacement (or immediate) value is limited compared to that of a 32-bit length instruction.

Here, the new instruction length is a mixture of 16-bit and 32-bit, to enhance the single instruction space. All of the existing SH-4A 16-bit instruction set (= I.S.) remains unchanged. The first 16-bit codes of 32-bit instructions are defined to vacant space as a 16-bit code. This backward compatibility for existing instructions makes a mode-bit to distinguish the existing 16-bit I.S. mode from the new I.S. mode unnecessary. It is desirable to exclude a mode-bit because the mode-bit can be a problem when branching between existing codes and new codes, requiring mode-state management.

Figure 5 shows the instruction format of basic 16-bit instructions and enhanced 32-bit instructions. The encoding of the 32-bit instructions includes a prefix control policy. In this 32-bit instruction, the second 16-bit portion is encoded the same as the existing 16-bit instruction format. The first 16-bit portion acts as a 'modifier' to the second 16-bit portion. This encoding policy makes the extra combinational logic small (that is, the existing 16-bit decoded output is utilized as decoded information by the 32-bit instruction).

Figure 6 shows the instruction decoder logic. The instruction decoder of general core can decode four sets of 16-bit code (not two sets), and the core can issue a maximum of two 32-bit instructions.

The first case of the 32-bit instruction format is "32 bit, with extra integer reg." (reg. = register), to enable three register operand integer operations, and three register memory load/store operations. The second case of a 32-bit instruction is "32 bit, with extra literal"; this format extends
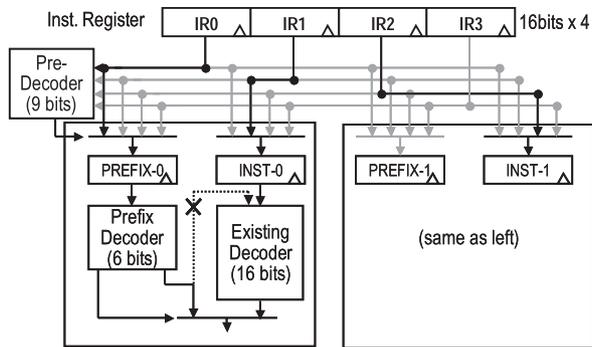
Fig. 6 Implementation of instruction decoder.
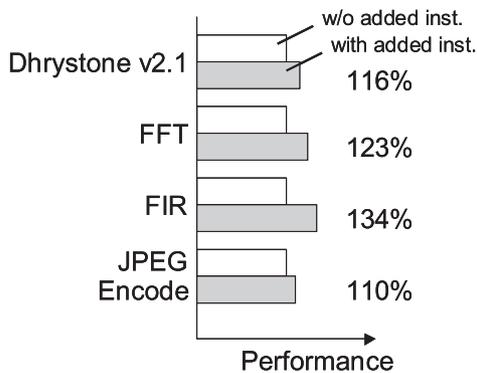
Fig. 7 Performance improvement using enhanced instructions.

Fig. 8 Virtual and physical address map of general-purpose core.

the length of displacement (of existing 16-bit instruction) by four bits. The displacement is used as follows:

Rn[31:0] + zero_extension (displacement)

Example: Rn + disp8[7:0]

The four-bits extension effects to {0, 4, 8}-bit displacement field into {4, 8, 12}-bit, respectively. The number of 32-bit instructions (as a combination of 16 bits and 16 bits) is 130.

Figure 7 shows the speed-up ratio with this 16-bit/32 bit instruction enhancements for four benchmark programs. With Dhrystone 2.1 code, a performance of 2.65 MIPS/MHz (MIPS: million instructions per second) is obtained.

In the whole static code of the Dhrystone 2.1 benchmark, the new code occurrence ratio over total is:

(i) First case: 32-bit instruction with extra integer reg. — 3.7%,

(ii) Second case: 32-bit instruction with extra literal — 1.8%.

The major part of the performance enhancement comes from the register extension.

4.2 Address Translation into 40-bit Physical Address

As we mentioned in abstract, the chip-total memory usage will exceed 4 GB even in the embedded chip. An 8-GB DDR module (DIMM) came onto the market in 2009.

To meet those memory trends, we expanded the function of physical space addressing of the general-purpose core to 40 bits. Figure 8 shows the extension of the physical address. The address space of a logical address (virtual address) remains 32 bits long, so the existing CPU-core register-programming model (pointer register length, branch address calculation) needs no change. The MMU (memory management unit) maps the logical address into a 40-bit physical address (1 TB, TB = terabyte). The virtual 32-bit space has a TLB (translation lookaside buffer) translation area and a PMB (privileged space memory buffer) translation area. The PMB translation area is a large page basis translation mechanism, it is separate to TLB. The MMU has two sub-unit, a TLB translator and a PMB translator. The whole physical space can hold 256 32-bit spaces. The P4 area in the virtual space is assigned as a control register area that is mapped on the bottom of physical space so that the linear physical space is not divided by the control register space.

The cache memory (instruction/data) is based on a physical cache, and snoop control processing is executed on this 40-bit basis.

5. Special-Purpose Cores

This chips integrates two MX-2 cores [10], four FE ("flexible engine") cores [11] and one VPU5. Here, we provide an overview of the processing of each type of core.

The MX-2 core tightly couples processing element (PE) array and SRAM (static random access memory). A large number (1,024) of PEs are placed in an MX-2 core, and the bit width of each PE is small: each can processes 4-bit addition and multiplication. This architecture can effectively process SIMD–type (single instruction multiple data) operations with large-number data. For the case with 4-bits data processing, each PE is utilized with bit width matched (no area loss). This PE core has enhanced the basic processing bits from two to four, and this change consequently enhances multiplication performance compared with addition performance (because the cycle count is proportional to square to bit-number in multiplication, and proportional

to bit-number in addition). Also, data larger than 4 bits (i.e., 8 bits, 12 bits, 16 bits, ...) can be processed by PE with multiple cycles (sequencing of PE). The MX-2 matches image data filtering and matching.

The FE targets a smaller number of SIMD operations smaller compared with the MX core. It integrates 24 16-bit ALUs and eight 16-bit multipliers. It matches 16-bit audio processing and image processing with a smaller number of elements.

VPU5 is a video processing unit that can support multiple video standard formats (H.264, MPEG2, etc.). For the moving picture processing, the final input/output stream can be compressed using this VPU5 and that minimizes the IO bandwidth.

Using that special-purpose core integration, we targeted decoding intelligent TV (or other moving picture processing systems). We define the intelligent TV as a television system that can also process searching and analysis of moving picture image. For example, we aimed to perform parallel processing that would (i) decode moving picture stream and (ii) extract the representative picture, and (iii) to search the moving picture.

Examples of target processing for the MX-2 and FE are as follows.
(i) Filtering for moving pictures:

For input picture data, in_data[x, y], the spatial filtering algorithm is

$$out\_data[x, y] = \sum_{i,j} w[i,j]\ in\_data[x+i, y+i]$$

and the approximate processing amount equals 2× (pixel number) × (range of [i, j])

For the HD image (60M pixels per second) and range of 25 (5 × 5), the processing requirement is 3 GOPS.
(ii) Extracting the representative picture,
To recognizing the object in a moving picture, the following matching algorithm is used,

$$match(x1, y1, x2, y2)$$
$$= \sum_{i,j} |in\_data[x1+i, y1+j] - ref\_data[x2+i, y2+j]|$$
$$match\_search[x,y] = \min_{i2,j2} (match(x, y, x+i2, y+i2)).$$

where "ref_data" is the reference (previous) picture, the similar to the algorithm used for moving picture compression. For the sampling points in every 1 of 16 for both x, y in an HD image (60 M/256 = 0.234 M pixels per second) and at a range of {i, j} 100 (10 × 10), and a range of {i2, j2} 100 (10×10) the processing requirement is 7 GOPS, where inner processing counts as three operations (addition, subtraction and absolute operation).

6. Chip Implementation and Evaluation

As shown on Table 1, a 45-nm LP CMOS process with eight metal layers is used to implement this chip. The transistor count is 545 M, and the clock signal is 13, which includes a variety between synchronous clocks, for example 648–324 MHz. The physical chip size is 12.4 mm × 12.4 mm and the effective area including PAD is 11.0 mm × 11.0 mm. This 12.4 mm was chosen by a dicing condition. There are no circuits, no PAD outside the 11.0 mm × 11.0 mm.

On the fabricated chip, each general-purpose core operates with 648 MHz at 1.15 V. The MX-2, FE, and VPU5 operates 324 MHz. The measured power of each IP is calculated with subtraction of operating power and no-operation (no-clock) power. In total, we obtained 37.3 GOPS/W performance, where the operation count was normalized with 32-bit operation [12].

Eight general-purpose cores and four FEs, eight DTUs operated functionally, with AAC encoding parallel processing [12].

7. Conclusion

Using a 45-nm CMOS, we built a 12.4 mm × 12.4 mm, (effective area 11.0 mm × 11.0 mm) chip, with heterogeneous core integration of eight 648 MHz general-purpose CPU cores, two matrix-type MX-2 cores, four FE cores and one VPU5. The on-chip bus supports a 40-bit address. This general-purpose core has a 16-bits/32-bits mixed instruction set that enhances the performance of a general-purpose core by 16, 23, 34 and 10 percent for four benchmark programs, where the frequency is normalized. The fabricated chip exhibited 37.3 GOPS/W performance power efficiency.

References

[1] J. Howard, S. Dighe, Y. Hoskote, S. Vangal, D. Finan, G. Ruhl, D. Jenkins, H. Wilson, N. Borkar, G. Schrom, F. Pailet, S. Jain, T. Jacob, S. Yada, S. Marella, P. Salihundam, V. Erraguntla, M. Konow, M. Riepen, G. Droege, J. Lindemann, M. Gries, T. Apel, K. Henriss, T. Lund-Larsen, S. Steibl, S. Borkar, V. De, R. Van Der Wijngaart, and T. Mattson, "A 48-core IA-32 message-passing processor with DVFS in 45 nm CMOS," ISSCC Dig. Tech. Papers, pp.108–109, Feb. 2010.

[2] K. Hagiwara, Y. Kiyoshige, T. Kataoka, T. Hiraoka, H. Sato, H. Yamada, Y. Sugure, S. Iwata, and T. Hattori, "High performance and low power SH2A-DUAL core for embedded microcontrollers," Cool Chips XI, session 11-2, April 2008.

[3] K. Mizumoto, T. Tanizaki, S. Kobayashi, M. Nakajima, T. Gyohten, H. Yamasaki, H. Noda, M. Higashida, Y. Okuno, and K. Arimoto, "A multi matrix-processor core architecture for real-time image processing SoC," Proc. 2007 A-SSCC, pp.180–183, Nov. 2007.

[4] Y. Yoshida, T. Kamei, K. Hayase, S. Shibahara, O. Nishii, T. Hattori, A. Hasegawa, M. Takada, N. Irie, K. Uchiyama, T. Odaka, K. Takada, K. Kimura, and H. Kasahara, "A 4320 MIPS four-processor core SMP/AMP with individually managed clock frequency for low power consumption," ISSCC Dig. Tech. Papers, pp.100–101, Feb. 2007.

[5] O. Nishii, I. Nonomura, Y. Yoshida, K. Hayase, S. Shibahara, Y.

Tsujimoto, M. Takada, and T. Hattori, "Design of a 90 nm 4-CPU 4320 MIPS SoC with individually managed frequency and 2.4 GB/s multi-master on-chip interconnect," Proc. 2007 A-SSCC, pp.18–21, Nov. 2007.

[6] K. Kimura, "Low power many-core compiler and extensions of OS-CAR API for many-cores and heterogeneousness," Symposium for Low Power Many-core Processor System Technology, Tokyo, Feb. 2010.

[7] M. Takada, S. Shibahara, K. Hayase, T. Kamei, Y. Yoshida, K. Takada, N. Irie, O. Nishii, and T. Hattori, "Performance and power evaluation of SH-X3 multi-core system," Proc. 2007 A-SSCC, pp.43–46, Nov. 2007.

[8] M. Ito, T. Hattori, Y. Yoshida, K. Hayase, T. Hayashi, O. Nishii, Y. Yasu, A. Hasegawa, M. Ito, H. Mizuno, K. Uchiyama, T. Odaka, J. Shirako, M. Mase, K. Kimura, and H. Kasahara, "An 8640 MIPS SoC with independent power-off control of 8 CPUs and 8 RAMs by automatic parallelizing compiler," ISSCC Dig. Tech. Papers, pp.90–91, Feb. 2008.

[9] Renesas Electronics, "SH-4A extended functions software manual," Rev 1.0. March 2007.

[10] T. Kurafuji, M. Haraguchi, M. Nakajima, T. Gyoten, T. Nishijima, H. Yamasaki, Y. Imai, M, Ishizaki, T. Kumaki, Y. Okuno, T. Koide, H.J. Mattausch, and K. Arimoto, "A scalable massively parallel processor for real-time image processing," ISSCC Dig. Tech. Papers, pp.334–335, Feb. 2010.

[11] M. Ito, T. Todaka, T. Tsunoda, H. Tanaka, T. Kodama, H. Shikano, M. Onouchi, K. Uchiyama, T. Odaka, T. Kamei, E. Nagahama, M. Kusaoke, Y. Nitta, Y. Wada, K. Kimura, and H. Kasahara, "Heterogeneous multiprocessor on a chip which enables 54× AAC-LC stereo encoding," IEEE 2007 Symp. VLSI, pp.18–19, June 2007.

[12] Y. Yuyama, M. Ito, Y. Kiyoshige, Y. Nitta, S. Matsui, O. Nishii, A. Hasegawa, M. Ishikawa, T. Yamada, J. Miyakoshi, K. Terada, T. Nojiri, M. Satoh, H. Mizuno, K. Uchiyama, Y. Wada, K. Kimura, H. Kasahara, and H. Maejima, "A 45 nm 37.3GOPS/W heterogeneous multi-core SoC," ISSCC Dig. Tech. Papers, pp.100–101, Feb. 2010.

Osamu Nishii received the B.S. and M.S. degrees in mathematical engineering and instrumentation physics from the University of Tokyo, Japan, in 1985 and 1987, respectively. In 1987, he joined Hitachi, Ltd., where he has been engaged in the research and development of processor logic architecture and design. He moved to Renesas Technology Corporation in 2004. Since 2010, he has been with Renesas Electronics Corporation. He is currently engaging development of microcontrollers.

Yoichi Yuyama received M.S. and Ph.D. degrees in Informatics from Kyoto University, Japan in 2003 and 2006, respectively. He joined Renesas Technology Corporation in 2006, and transferred to Renesas Electronics Corporation in 2010. Currently, he is a member of CPU design and verification group.

Masayuki Ito was born in Kyoto, Japan in 1970. He received the B.S. and M.S. degrees in computer engineering from Kyoto University in Kyoto, Japan, in 1993 and 1995, respectively. He is currently working with Renesas Electronics in Kanagawa, developing CPU cores with particular interest in low-power microprocessor design. He is the section manager of CPU development department at Renesas Electronics.

Yoshikazu Kiyoshige is a senior engineer of 1st CPU Core Development Department, System Core Development Divison at Renesas Electronics Corporation. He joined Hitachi Ltd. in 1987–2002. He is designing micro processing unit. He is a member of the IPSJ.

Yusuke Nitta received the B.E. and M.E. degree in applied physics from Osaka University, Japan in 1989 and 1991, respectively. He joined the Central Research Laboratory of Hitachi, Ltd., in 1991. He is currently involved with physical design of micro-processors in Renesas Electronics Corp., Japan.

Makoto Ishikawa received the B.E., M.E. and Ph.D. degrees from Tokyo Institute of Technology in 1997, 1999 and 2008, respectively. In 1999, he joined the Central Research Laboratory of Hitachi, Ltd., Japan, where he was engaged in the research and development of embedded processor architecture. He was seconded to the Automotive Products Laboratory of Hitachi America, Ltd., in Michigan from 2005 to 2007, where his research focused on model-based system development methods using a micro-controller model. He is currently back at the Central Research Laboratory, Hitachi, Ltd. and is working on multi-core processor technologies performance design technologies for embedded systems. He is a member of IEEE and SAE.

Tetsuya Yamada received the B.E. degree in information engineering, the M.E. degree in information systems, and D.E. degree in information engineering from Kyushu University, Japan, in 1992, 1994, 2009 respectively. He joined the Central Research Laboratory of Hitachi Ltd., in 1994. He engaged in low-power embedded microprocessor design. His research interest includes the PC server architecture.

Junichi Miyakoshi received the B.E. degree in electrical and information engineering and the M.E. degree in electronics and computer science from Kanazawa University, Japan, in 2002 and 2004, respectively. He received the Ph.D. degree in electrical engineering from Kobe University, Japan, in 2007. He joined the Central Research Laboratory of Hitachi, Ltd., in 2007. He has been engaged in the medical image processing.

Yasutaka Wada received B.S. and M.S. degrees in electrical engineering and Ph.D. degree in computer science and engineering from Waseda University, in 2002, 2004 and 2009 respectively. He was a research associate of the Department of Computer Science and Engineering in 2006, and a junior researcher of the Advanced Multicore Processor Research Institute in 2009 at Waseda University. Since 2010, he has been an assistant professor of the Graduate School of Fundamental Science and Engineering at Waseda University. He is a member of IPSJ and IEEE.

Keiji Kimura received the B.S., M.S. and Ph.D. degrees in electrical engineering from Waseda University, in 1996, 1998, 2001 respectively. He was an assistant professor in 2004, and has been associate professor of Department of Computer Science since 2005 at Waseda University. His research interest includes microprocessor architecture, multiprocessor architecture, multicore processor architecture, and there compiler. He is a member of IPSJ, ACM and IEEE.

Hironori Kasahara received Ph.D. degree in electrical engineering from Waseda University, Tokyo, in 1985, was an Assistant Professor in 1986, has been a Professor of Department of Computer Science since 1997 and currently also a Director of Advanced Chip Multiprocessor Research Institute at Waseda University. He was a Visiting Scholar at University of California at Berkeley in 1985 and Center for Supercomputing R&D, University of Illinois at Urbana-Champaign in 1989–1990. He received the Young Author Prize of IFAC World Congress, IPSJ Sakai Memorial Special Research Award, STARC Industry-Academia Cooperative Research Award, LSI of the year 2nd Grand Prize and IEEE Computer Society Golden Core Member Award. He has served as a member of IEEE Computer Society Board of Governors, a chair of IEEE Computer Society Japan Chapter, a Board member of IEEE Tokyo Section, a member of IEEE Japan Council Long Term Strategy Committee, a Chair or a member of Program Committees of a lot of conferences including IEEE/ACM SC, ICS, ASPLOS, PPoPP, ICPADS, ICPP and so on. He led several Japanese national projects, such as METI/NEDO "Advanced Parallelizing Compiler" and "Multicores for Real-time Consumer Electronics." He served as a member of MEXT "the Earth Simulator Advisory Committee," "Next Generation Supercomputer Evaluation Committee," "HPCI," Cabinet Office R&D Strategy Committees on Supercomputing and Software and a chair of NEDO "Computer Strategy WG."

Hideo Maejima is a professor at department of information processing, graduate school, Tokyo Institute of Technology. His current interests include multi-/many-core architecture, reconfigurable computing and integrated development environments for heterogeneous multicore systems. Maejima has a Ph.D. in department of computer science from Tokyo Institute of Technology. He is a member of IEEE Computer Society and the Information Processing Society of Japan.