Vol. 46 No. 6 IPSJ Journal June 2005

Regular Paper

A Novel Fingerprint SoC with Bit Serial FPGA Engine

Yiwen Wang,[†] Dongju Li,[†] Tsuyoshi Isshiki[†]
and Hiroaki Kunieda[†]

The paper presents firstly a novel system-on-chip (SoC) architecure consisting of a 32-bit RISC processor, on-chip memory, state-of-the art IPs and embedded full-custom bit serial FPGA (BSFPGA) I/O interface. The system inherits the compatibility of AMBA architecture, the flexibility of BSFPGA, so that it can be used for various types of applications without any additional I/O pins. Example application is to realize fingerprint authentication system in a chip. The paper presents secondly design flow for SoC, including a full-custom block. With RTL model for BSFPGA, it enables a timing simulation of the total system in RTL level and a timing verification in transistor level.

1. Introduction

For many designs, non-recurrent engineering expense (NRE) and time to market are key points. Therefore, many engineers want to make the system should be flexible. In other words, they hope to program and optimize both system software and hardware even after SoC is fabricated. In application field of security, this kind of requirement becomes stronger and stronger. Currently fingerprint authentication modules usually have only one sensor interface and limited communication interfaces. However, sensor techniques are improved so fast that new sensors appear in every years. This fact requires that fingerprint authentication module must have an ability to utilize the upgraded sensors without developing a new module. Moreover, the security applications often need to insert some special circuit such as hardware accelerator or encryption circuit into the system.

Embedding FPGA into SoC is an efficient way to enable the system with reconfigurable ability, especially for those I/O interfaces. Reconfigurable system is a hot topic and many progresses have been made. Some systems treated embedded FPGA as a co-processor [1]. It enables the users to add customized instruction sets. Some systems focus on configurability of system bus [2], etc. Above systems are valid to develop a complicate reconfigurable system. However, in most cases, the customers don't need to change their architectures and designs greatly. Usually they just need to add some customized modules or general-purpose modules into current-used system. Mostly, they just need to update old driver protocol or I/O connection assignment.

Our proposed re-configurable SoC just fits these kinds of requirements. Its advantage is easy to use and provides reasonable performance. To keep compatibility, we selected popular AMBA [3] as our backbone bus structure. This makes users easier to add customized design into the proposed system. BSFPGA plays a key role in the system. Comparing with parallel FPGA, BSFPGA [4] has better routability, so that it can reduce area and cost greatly. As BSFPGA have shorter path for application design, it can run in higher speed.

To improve system performance, following strategies are used in the system. A specific controller including DMA mode is designed to control the configuration and normal operation of BSFPGA; Input FIFO and output FIFO is used between APB bus and BSFPGA to speed and synchronize parallel and serial data exchange. 1 k × 16 bits dual port SRAM between AHB bus and BSFPGA is used to solve bottleneck for high-speed data exchange. All above ensure the system have high performance.

On the other hand, bit-serial computation and bit serial circuit has become more and more popular. The proposed SoC address issues of mapping both parallel type circuit and serial type circuit inside BSFPGA. During mapping and synthesis, parallel-serial converter and serial-parallel converter can be generated automatically and implemented into BSFPGA combined with user's circuits.

[†] Tokyo Institute of Technology

The real author is the Editorial Board of the *Trans. IPSJ.*

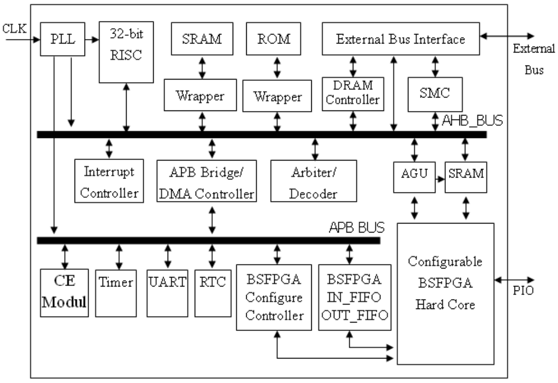

Fig. 1 System architecture.

These features make BSFPGA become a good solution for reconfiguration SoC. Our typical application case, a fingerprint authentication algorithm, which was named as minutia-ridge shape algorithm[5] is stored in on-chip ROM. The algorithm is easy to be invoked by other application programs. On-chip ROM can also enhance the system security level. The algorithm is robust against fingerprint rotation and displacement. The template data size is only 64 bytes for each fingerprint.

Various sensor interface and communication protocol can be mapped into BSFPGA easily. In a typical application, a fingerprint-matching accelerator is mapped into BSFPGA. It speeds up matching speed with one hundred and twenty times as fast as DSP modules. The details of system architecture and design flow are introduced in the following.

2. System Architecture

The system architecture is shown in **Fig. 1**. The system chip includes a 64 KB ROM in which fingerprint image processing and minutia extraction algorithm are embedded. This feature increases the security level for whole system comparing with storing fingerprint authentication programs in external memory. The user application programs and some tuning parameters of sensor and algorithm are stored in external Flash memory. Speed Sensitive functions are mapped into BSFPGA as Fingerprint hardware accelerator to improve the performance of whole system.

A 32-bit RISC processor was used in the system. The processor works in 200 MHz frequency with 8 KB data cache, 8 KB instruction cache and memory protection unit. Bus interface unit was designed to support clock frequency of RISC clock or 1/N times RISC core clock, where N is any positive integer. RISC clock and bus interface clock must be synchronized.

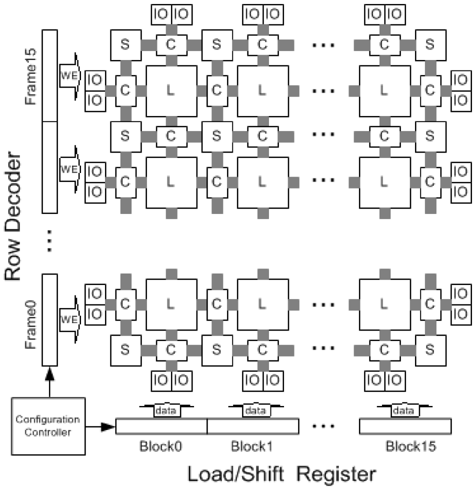

Fig. 2 Structure of BSFPGA.

BSFPGA is a key part in the system to bring configurability to users. Specific interfaces are used to connect BSFPGA with processor and other modules seamlessly.

Figure 2 shows the detail architecture of BSFPGA. The BSFPGA consists of 256 basic cells (16 × 16) with up to 30 K ASIC equivalent programmable gates. Each tile includes one L block, two C block and one S block. One of the main features of BSFPGA is the large logic block consisting of 4-input LUTs and 6 FFs with dedicated carry-save logic for bits-serial operations. Another feature is that the two level routing architecture aimed to make the inter-block routing simple with short delays while providing another layer of routing inside the logic block to provide pin flexibility in order to increase routeability.

BSFPGA application development flow is shown in **Fig. 3**. Verilog design entries is translated into BSFPGA netlist after synthesis and technology mapping. Algorithm written in C++ is compiled and also translated into BSFPGA netlist by using BSPipeline Synthesizer. After placement and routing, BSFPGA bit stream Generator produces configuration data for application mapping. To ease the simulation and verification process, BSFPGA bit stream Generator also can generate simulation model and test vector automatically.

The application development tools can handle either parallel data path design or serial

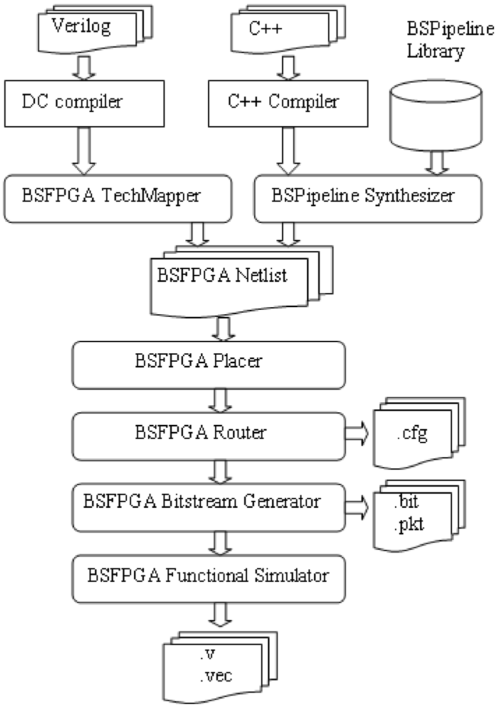

Fig. 3 Application development flow.

data path design. Instead of designing special data converting circuit outside BSFPGA, parallel serial converter and serial parallel converter are inserted dynamically into application design. It is compatible with AMBA parallel architecture. This feature is very flexible for customized design

BSFPGA interface between AMBA [5] is designed by using FIFO to synchronize the clock and data transfer between APB bus and BSFPGA. For many applications, local memory is necessary. Also the data transfer speed becomes a bottleneck when we use APB bus-the low speed bus. Therefore, in our proposed system, we insert a fast local dual port memory between BSFPGA and fast speed bus-AHB to speed the data access between processor and BSFPGA. The local memory is also used as workspace by fingerprint matching engine mapped into BSFPGA.

The configuration file is stored in external flash memory in parallel package format. During reset period, The Configuration control program inside ROM will check the status of BSFPGA and load the configuration data into BSFPGA configuration controller automatically. Configuration data are transferred in DMA mode and input BSFPGA with parallel mode, therefore configuration cycle is less than 0.6 ms. Once BSFPGA controller complete the initialize and configuration, the status registers will be set and wait the operating instruction come from processor.

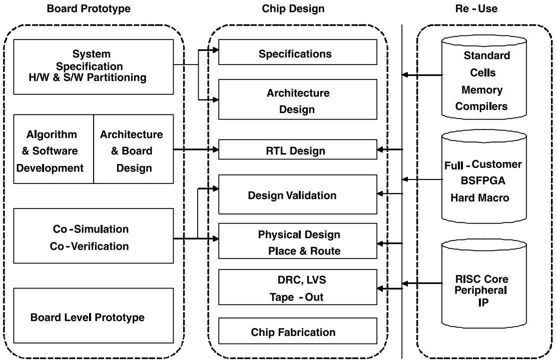
Board Prototype
Chip Design
Re - Use
System Specification H/W & S/W Partitioning
Algorithm & Software Development
Architecture & Board Design
Co - Simulation Co - Verification
Board Level Prototype
Specifications
Architecture Design
RTL Design
Design Validation
Physical Design Place & Route
DRC, LVS Tape - Out
Chip Fabrication
Standard Cells Memory Compilers
Full - Customer BSFPGA Hard Macro
RISC Core Peripheral IP

Fig. 4 Design flow.

Besides BSFPGA, some general-purpose peripheral modules such as UART, RTC, Timer, Customized Module, etc. are embedded into the system. All the modules are managed under power control unit to decrease the power consumptions.

3. Design Flow

The design flow is shown in **Fig. 4**. The design methodology of proposed SoC is based on Spiral SoC design principle rather than traditional waterfall design flow.

The system specification and parameters were carefully decided at the start phase to minimize the iteration loops. Considering the complexity and timing requirement, we address all aspects of hardware and software concurrently, including logic design, timing, verification and physical design. The proposed system design flow can be viewed as a mixture of top-down design and bottom-up design process. System is designed by using top-down method but specific module (BSFPGA) was designed by using bottom-up method.

In order to simulate and verify the system, we need to provide a complete logic simulation model for the BSFPGA. The difficulty lies on the fact that BSFPGA is implemented in full-customer design, and the logical description of the full-customer design is done at transistor-level netlist. Writing a functionally equivalent simulation model on logic-level is very time consuming, Moreover, verifying the equivalency of the simulation model and the transistor-level netlist is required. In order to avoid the verification problems between the simulation model and the transistor-level netlist, we establish the simulation model directly from the transistor-level netlist. The problem with this approach is that the sequential circuits such as SRAM cell, flip-flops, and LUT elements composed of dynamic shift registers cannot be simulated at the logic-level. Therefore, we have to provide behavioral Verilog model for these sequential circuits and replace them during simulation. Equivalency of each sequential circuit between the simulation model and the transistor-level netlist is verified by SPICE simulator. By this scheme, we are able to establish a Verilog functional model based on transistor-level netlist, and any modification on the transistor-level netlist is directly reflected on the simulation model.

At the start phase, in order to run the whole system synthesis and static timing analysis (STA) before BSFPGA full-custom design is finished, A quick timing model (QTM) of BSFPGA is created. And then a real timing model will be extracted and used after finish BSFPGA full-custom design.

For fingerprint application, how to partition hardware design and software design is a key point. Also the fingerprint algorithm have very close relations with application program. Therefore, a board level prototype fingerprint authentication module (SoB) is developed to evaluate and verify specific hardware and software. Based on the board prototype, the architecture and specification of SoC is refined. The program size and processing time are analyzed and estimated under RISC compiler to find critical function and then assigned the function into BSFPGA to improve the performance of the system. The final optimized program is stored in on-chip ROM after verification on board level. The final optimized program size of fingerprint authentication algorithm is only 30 KB.

Verification always be a challenge for SoC design. In our proposed SoC, Verification environment is setup by using VERA and DesignWare. Both directed test and random test are performed to increase the fault coverage. For BSFPGA, we import self-test method to meet design for test (DFT) challenge.

In physical design, RISC, BSFPGA, SRAM, ROM, are treated as hard macro. BSFPGA configuration controller, BSFPGA FIFO interface and peripheral models of systems were designed by using standard cell. Due to the effect of IR drop and noise in deep-sub-micron design, the physical design is performed based on some basic principles, especially for routing power ring and clock-tree.

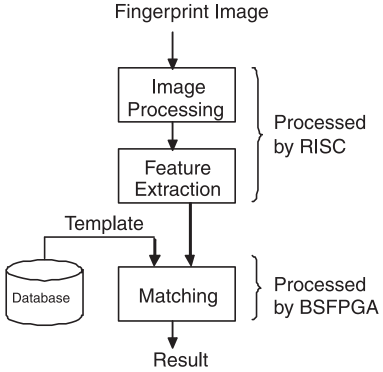

Fig. 5 Fingerprint identification flow.

4. BSFPGA Based Fingerprint Engine

As a typical application, a minutia based fingerprint identification 1:N system is designed by using proposed SoC. In the system, RISC and BSFPGA share the work of fingerprint processing efficiently and get better performance.

4.1 Fingerprint Identification Flow

Fingerprint identification flow consists of image processing, feature extraction and matching as shown in **Fig. 5**. In prior to fingerprint identification, the registration process is performed in similar way to extract genuine features from fingerprint image and then save the features as a template into database. Fingerprint identification works as follows:

(1) Image processing includes prefiltering, 1st binarization, direction detection, direction filter, 2nd binarization, thinning, etc.

(2) Fingerprint features extraction is to extract minutia and their associated information.

(3) Matching is a comparison of extracted features between an input and N templates. It is repeated to calculate the similarity scores between an input fingerprint and each template in the database and to find out a template with the best score. In prior to get a similarity score, the displacement and rotation of templates against the input fingerprint should be compensated.

In our system, One-time operations such as image processing and fingerprint features extraction are handled by RISC. On the other hand, the matching process needs huge repeated computations. Therefore, a hardware matching engine is our solution.

For matching engine, the hierarchical approach based on a small memory matching method[6)] is adopted, which consists of global rough search, best direction decision and local precise search as shown in **Fig. 6**.

4.1.1 Global Rough Search

Global rough search is performed to find the rough displacement value by using horizontal and vertical 1D memory array instead of 2D memory array, which consists of shape matching, score matching and histogram as shown in **Fig. 7**. This search is fast with less memory requirement.

Shape matching module calculates the shape similarity of minutia pairs between input data and template data by using the independent parameters on the displacement and rotation. Score matching module translates the shape similarity into score data. Both shape and score matching are performed in pipeline function.

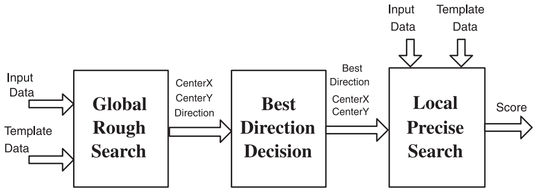

Fig. 6 Block diagram of fingerprint matching engine.

Histogram module calculates the displacement of each minutia pair between input data and template data and treats the displacement as address. The associated matching score of the minutia pair is stored into score memory locate in the above address. By repeating this operation for all possible pairs, we get the accumulated score, which are distributed in memory array. Among them, the displacement with largest matching score is selected as displacement of all minutia pairs. In order to find the rotation transformation between input fingerprint and template, the input minutia is rotated 32 directions step by step in 360 degree. In each step, we use the histogram module to search the maximum score. This histogram can be designed either in parallel mode or serial mode. To decrease matching engine size, we adopt serial scheme in proposed system.

4.1.2 Best Direction Decision

Best direction decision module receives the max score for 32 directions. The direction with biggest score is selected as the rotation transformation between input fingerprint and template fingerprint.

4.1.3 Local Precise Search

After above processing, the displacement and rotation of input fingerprint are found. Therefore, we use local precise search to calculate and search the maximum score in a windows area of memory array. Finally it gives the identification result by comparing the maximum score

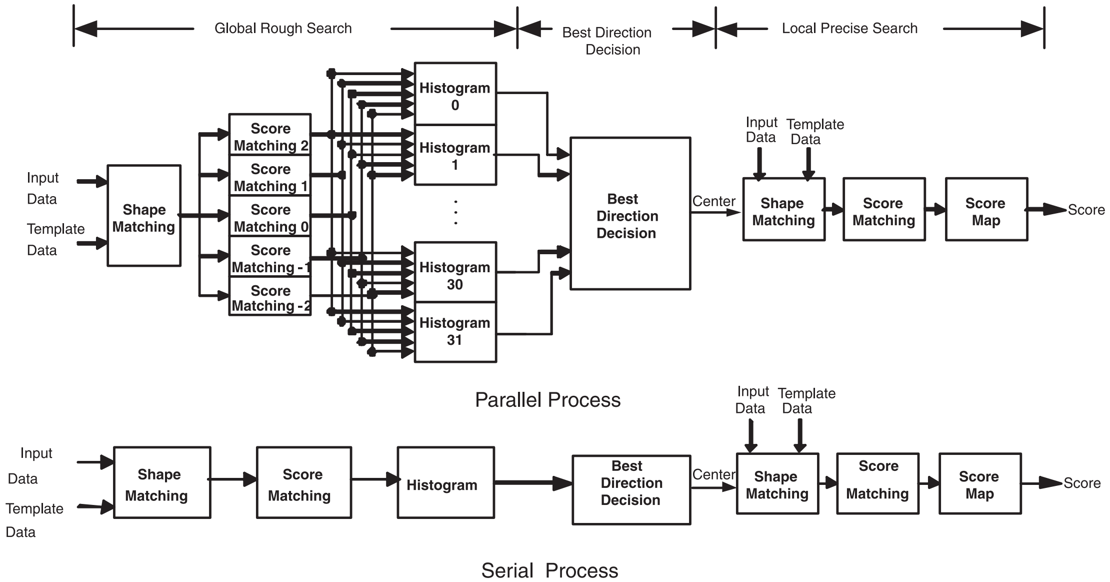

Fig. 7 Structure of fingerprint matching engine.

with defined threshold.

4.2 Implementation on BSFPGA

In our proposed system, the fingerprint matching engine is designed in BSFPGA. It shares fingerprint processing work with host processor seamlessly. Comparing with the single CPU system by which all task is processed, the proposed system improves system performance tremendously and make the system reconfigurable as well. Comparing with the fingerprint system built on common FPGA architecture[7], the proposed system has less size. Hence, the system is easy to be integrated into chip level instead of board level. The BSFPGA works in pipeline mode to avoid the tradeoff for parallel-serial data exchanging.

Shape matching module, score module and best direction decision module use about 20% of BSFPGA resource totally. However, Histogram module will use a lot of register to store score data. To overcome this bottleneck, we employ an on-chip dual port SRAM to address this issue. Therefore, we can realize the fingerprint matching engine using less than 65% of BSFPGA. The engine runs in 50 MHz to synchronize with APB clock.

Table 1 Performance comparison.

	matching speed	speed comparison
TMS320VC5409 DSP (Compiler: CCS IDE1.22) (Clock: 100 MHz)	7.2×10^{-2}	1
Paralle Process (Clock: 50 MHz)	3.6×10^{-5}	1,984 times
Serial Process (Clock: 50 MHz)	6.0×10^{-4}	119 times

Note: The unit of matching speed is second / fingerprint. The speed of DSP Module is normalized as one.

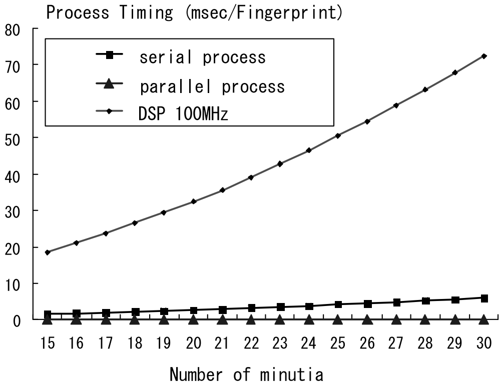

Fig. 8 Speed compare.

The matching speed of proposed system is given in **Table 1**. Comparing with a DSP based fingerprint module[8], the proposed system improves the performance greatly. Both systems give the same accuracy such as FAR less than 0.001% and FRR less than 0.1%. For our proposed system, 1,000 fingerprint of database can be processed within one second. **Figure 8** shows the relationship between matching process time and minutia number extracted from one fingerprint. For DSP module, matching time is worsen greatly while minutia number increase. On the contrary, the matching process time on proposed system has little change even minutia number increase. This indicates that the proposed system is stable and essential on matching process.

5. Implementation and Result

The proposed SoC is about to be manufactured in a standard CMOS 0.18 μm technology featuring 6 metals and 1 poly. Die size is 5 mm × 5 mm (pad limited). Configuration data of BSFGPA is about 60 KB and it takes about 600 μS for configuration. The specification, characteristics is showed in **Table 2** and **Table 3**. **Figure 9** shows the layout of proposed SoC and one tile of BSFPGA.

In a typical application, a fingerprint matching engines is designed and mapped into BSFPGA. Furthermore, various sensor interfaces and general purposed-used modules can also be

Table 2 SoC characteristics.

Process	0.18 μm CMOS/6 metal
Package	360 BGA
Power Supply	3.3 V (I/O), 1.8 V (Core)
RISC speed	200 MHz
Diffusion ROM	64 KB
SRAM	32 KB
SoC Die Size	5 mm × 5 mm
SoC Gate Count	1,200 K

Table 3 BSFPGA characteristics.

Array Size	16 × 16 tiles (256 logic blocks)
Area of Per Tile	107.84 μm × 87.78 μm (logic block + S-Block + 2 C-block)
User-gate count	18,000–30,000 gates (depends on applications)
Operating Clock frequency	50 MHz (APB clock)
Max Clock frequency	300 MHz (assume 4 manhattan distance for critical net routing)
BSFPGA Area	1.9 mm × 1.5 mm
Transistor Count	1,000,000 transistors

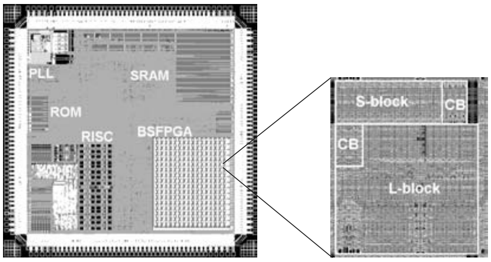

Fig. 9 VLSI layout.

mapped into BSFPGA easily.

6. Conclusion

In this paper, we presented a novel SoC architecture with embedded BSFPGA, which can be used in multi purpose application field to bring the system with more flexibility and extensibility. As typical application example, a fingerprint authentication system is designed by using the proposed system. Comparing with the developed fingerprint authentication SOB, proposed SOC has much higher performance and low cost.

References

1) Borgatti, M., Lertora, F., Foret, B. and Cali, L.: A reconfigurable system featuring dynamically extensible embedded microprocessor, FPGA, and customizable I/O, *Solid-State Circuits, IEEE Journal*, Vol.38, No.3, pp.521–529 (Mar. 2003).

2) Winegarden, S.: Bus Architecture of a system on a chip with User-Configurable System Logic, *Solid-State Circuits, IEEE Journal*, Vol.35, No.3, pp.425–433 (Mar. 2000).

3) Advanced Microcontroller Bus Architecture (AMBA) Spec Document Number: ARMIHI0011A. issued 13th May 1999. http://www.arm.com/

4) Ohta, A., Isshiki, T. and Kunieda, H.: New FPGA Architecture for Bit-Serial Pipeline Data path, *IEICE Trans.*, pp.1663–1672 (Aug. 2000).

5) Mostafa, M., Li, D.J. and Kunieda, H.: Minutia-Ridge Shape Algorithm for Fast On-Line Fingerprint Identification System, *IEEE ISPACS 2000*, pp.593–598 (Nov. 2000).

6) Rikin, A.S., Li, D.J., Isshiki, T. and Kunieda, H.: Small Memory Minutia Matching Method, *IEEE ISPACS 2003*, pp.887–892 (Dec. 2003).

7) Schaumont, P. and Verbauwhede, I.: Thumbpod puts security under your thumb, *Xilinx Xcell Journal 2003*, pp.887–892 (Oct. 2003).

8) Rikin, A.S., Wang, Y.W., Nakada, T., Li, D.J., Isshiki, T. and Kunieda, H.: Realization of Fingerprint Identification Module on DSP Board, *IEEE APCCAS 2002*, pp.509–512 (Dec. 2002).

(Received December 1, 2004)
(Accepted April 1, 2005)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.1, pp.226–233.)

Yiwen Wang received the B.S. degree from Wuhan University, China, in 1988. He worked as an IC design and test engineer in VLSI Design Laboratory of Northeast Micro-electronics Institute, Electronic Industry Bureau, China, from 1988–2001. He is currently a Ph.D. candidate at Department of Communications and Integrated Systems in Tokyo Institute of Technology. His current research focuses on SoC design and DFT research.

Dongju Li received the B.S. degree from LiaoNing University and M.E. degree from Harbin Institute of Technology, China, in 1984 and 1987, respectively. She worked as an IC design engineer in VLSI Design Laboratory of Northeast Micro-electronics Institute, Electronic Industry Bureau, China, from 1987–1993. She is currently a Research Associate at Department of Communications and Integrated Systems in Tokyo Institute of Technology. Her current research interests include multiprocessor architecture and VLSI design.

Tsuyoshi Isshiki received the B.E., M.E. degrees in electrical and electronics engineering from Tokyo Institute of Technology in 1990 and 1992, respectively. He received his Ph.D. degree in computer engineering from University of California at Santa Cruz in 1996. He is currently an Assistant Professor at Department of Communications and Integrated Systems in Tokyo Institute of Technology. His research interests include high-level design methodology for configurable systems, bit-serial synthesis, FPGA architecture, image processing, computer graphics, and speech synthesis.

Hiroaki Kunieda received the B.E., M.E. and Dr. Eng. degrees from Tokyo Institute of Technology in 1973, 1975 and 1978, respectively. He was Research Associate in 1978 and Associate Professor in 1985, at Tokyo Institute of Technology. He is currently Professor at Department of Communications and Integrated Systems in Tokyo Institute of Technology. He has been engaged in researches on Distributed Circuits, Switched Capacitor Circuits, IC Circuit Simulation, VLSI CAD, VLSI Signal Processing and VLSI Design. His current research focuses on VLSI Multimedia Processing including Video Code, Design for System On Chip, VLSI Signal Processing, VLSI Architecture including Reconfigurable Architecture, and VLSI CAD. He is a member of IEEE CAS and SP society and IPSJ.