社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2007－ARC－173 (12)
2007／6／1

Design Techniques of Wave Pipelines

Masa-aki FUKASE† and Tomoaki SATO‡

† Dept. of Electronic and Information Systems Engineering, Hirosaki University
Bunkyo-cho 3 , Hirosaki, 036-8561 Japan
‡ Computer and Network Systems Center, Hirosaki University Bunkyo-cho 3 , Hirosaki, 036-8561 Japan
E-mail: † slfuka@eit.hirosaki-u.ac.jp, ‡ tsato@cc.hirosaki-u.ac.jp

Abstract In order to improve rather complicated design and testing methods of wave-pipelines, our policy is to cover rough tuning as well as fine tuning. This is practically useful for both universal chips and FPGAs. The reconfigurable feature of FPGAs is crucial for quickly corresponding to the drastic change of ubiquitous network environment. In this study, we have focused our attention to three key techniques. The first topic is the development of a CAD tool that dynamically shows a wave map. The second topic is the application of wave-pipelining to sequential circuits. The final topic is a double clocking scheme.
Keyword Wave-pipeline, sequential logic, design for testability, processor, chip, CMOS

1. Introduction

Pipeline is crucial for the parallelism of processor control. Although the pipeline control is shared by wave-pipelines and regular pipelines, regular pipelines have fundamental issues in the deep submicron era after Moore's law that are accompanied with pipeline registers. They have been the bottle neck of enhancing processors performance and power dissipation. This is because they are composed of many latches that occupy larger area than combinational gates [1]. In this respect, wave-pipelines are superior to regular pipelines. Wave-pipelines are set up without pipeline registers. Yet, wave-pipelines are synchronized with clock. Wave-pipelines are processor techniques practically promising for ever growing ubiquitous net-work that demands power conscious high speed multimedia processing [2].

In spite of wave-pipelines feasibility, they have not always been benefited by today's processor techniques and CAD tools. Since these are dedicated to the development of regular pipe-lines, their applications to wave-pipelines is really complicated. Thus, we have explored design and testing methods specific for wave-pipelines. For the global standardization of wave-pipelines, our policy is to cover rough tuning as well as fine tuning. This is practically useful for both universal chips and FPGAs. The reconfigurable feature of FPGAs is crucial for quickly corresponding to the drastic change suddenly happened in ubiquitous network environment [3].

This paper focuses on three techniques we have developed for wave-pipelines. The first topic is a graphical approach that interfaces a timing report and GUI (graphical user interface), and draws a wave map [4]. Since the wave map exactly shows instantaneous wave's shape or sphere on an actual circuit design space, its graphic power is definitely superior to regular methods like timing chart and cone model [5].

The second topic is DFT called a double clocking scheme [6]. This is a practical testing method of wave-pipelines. Although delay tuning is the essence to achieve wave-pipelines' high speed parallelism, circuit delays strongly depend on operating conditions like voltage, temperature, etc. Wave-pipelines have been hard to measure by standard methods and general measurement equipments. Thus, DFT is crucial for the exact verification of wave-pipelines.

The third topic is the wave-pipelining of multifunctional units and sequential logic circuits. Although wave-pipelining has been applied to unifunctional combinational logic circuits (CLB), multifunctional CLB and sequential logic circuits play important roles in any processor in the market. They are vital for the practical development of wave-pipeline circuits, and are useful to further enhance processor specification. Since multifunctional wave-pipelines are released from awkward instruction scheduling without the slowdown of clock speed, the arrangement makes it possible to realize wide-range dynamic ILP at a rate higher than regular superscalar architecture [7]. Sequential logic circuits are frequently used in the various scene of wireless LAN environment. In addition, they have similar structure. Thus, the effect of wave-pipelining them is very large [8].

2. Wave-pipeline's Background

The target of processors development has been indicated by processor road maps, taking into account of the progress of related techniques, market trend, and the analysis of user demands for highs speed, low power, and high performance. Although those indications have been so far growing steadily, they have encountered the slowing of increase in the deep submicron era due to several issues in bellow.

a. The end of the mictrostructural refinement technologies of VLSI process: Since it is physically reasonable, Moore's law is reconsidered. Power constraint is another factor that interrupts minituarization. Decreasing chip size consumes more static power, because small transistors use electricity all the time even in switch off condition [9]. This suppresses the increasing of transistor density [10]. Even the new integration approach of a three dimensional LSI/package technique departs from the spectrum of integration so far continued, because the three dimensional technique does not affect regular two dimensional layout.

b. The saturation of clock speed: It is to due to several factors of processor design trend based on the regular pipeline. An architectural factor in causing the slowdown is pipeline registers. A gate level factor is the dynamic power of CMOS proportional to clock frequency. A layout level factor is wire delay. A critical path closely related to clock speed is determined not gate delay but wire delay in the deep submicron era. Thus, the decreasing of wire delay saturates as the decrease of layout size.

c. The saturation of processor performance: This is concerned with clock speed and parallel degree. Since clock is at the end of speed, the new parallelism of multicore has emerged [11]. The multicore scheme supplementing clock speed saturation covers the parallelism among loops and functions, and the parallelism within a loop. Since not only clock but also power constraints are distributed among cores, the overall performance improves. A remaining subject of multicore is a technique to abstract the parallelism of application software [12].

d. The increase of power dissipation: This is closely related to integration and clock speed as is described above.

The bottle neck of processor road maps is clock speed and power dissipation. They have many factors to be analyzed and these factors are closely related each other. One of most important factor is the pipeline register. In this respect, the wave-pipeline without the pipeline register is an innovative processor technology that overcomes regular trend. Although wave steering without the pipeline register is also attractive, it is applicable only for the logic synthesis by using BDD (binary decision diagram) [13].

3. Design Principle of Wave-pipelines

Our attention is focused on how wave-pipelines may con-tribute to processor design [14], [15]. The basis of a processor is digital or logic circuits that are modeled by Huffman model shown in Fig. 1. Fig. 2 summarizes the common concept for the delay tuning of the wave- vs. regular pipelines.

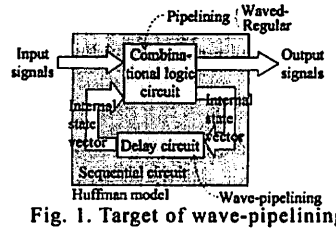

Fig. 1. Target of wave-pipelining

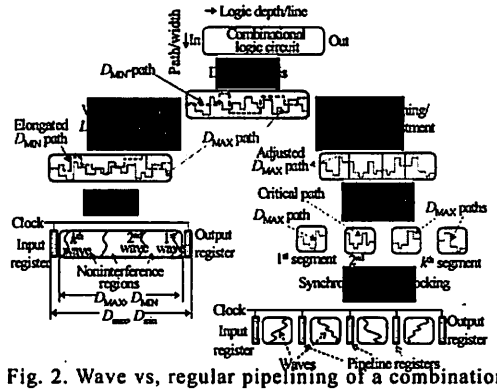

Fig. 2. Wave vs, regular pipelining of a combinational logic circuit.

A wave implies a set of signals that propagate like a wave packet. Then, a wave-pipeline is a sort of fault tolerant CLB that propagates plural waves without collision. Thus, it behaves as a regular pipeline. The number of stages of regular pipelines corresponds to wave degree or the number of waves. Since the wave-pipeline does not use pipeline registers, the collision of waves is the cause of error. The collision is the worst case of the interference of waves, which is concerned with expansion of waves or variation among signal delays. Thus, the principle of wave-pipelining is to minimize the width of waves.

Different ways are taken for the delay tuning of wave- and regular pipelines. The essence of delay tuning to

reduce the width of waves is not decreasing the maximum delay, D_{MAX}, but increasing the minimum delay, D_{MIN}. Decreasing D_{MAX} is subsidiary and almost effort is paid to increasing D_{MIN}. This is repeated, because a new D_{MIN} appears when the current D_{MIN} is increased. A regular pipeline shown in Fig. 2 is for reference.

The wave-pipelining of CLB can be applied to sequential logic circuits, because the raison d'etre of the delay circuit in Fig. 1 is its delay characteristic. Fig. 3 shows the principle of wave pipelining sequential logic circuits [8], [16]. The delay circuit is actually a register and this is accompanied with no logic functions. Thus, it can be made by 1-input flip flops like D-FF, T-FF, etc. Flip flops with logical functions like SR-FF are useless. Then, the delay of 1-input FFs is substituted by buffers as is similar to the wave-pipelining of CLB. Rough tuning adjusts the number of buffers. Fine tuning is also taken when it is needed.

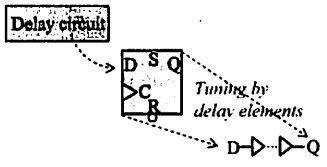

Fig. 3. Principle of wave pipelining sequential logic circuits.

4. Development Methods of Wave-pipelines

The wave-pipeline achieves both speed up and power saving. It is immune to that sort of shortcomings as shown in Table 1. However, the wave-pipeline has not widely used by circuit designers. Due to the minority party, the development methods of wave-pipelines are immature. Actually, the delay tuning of wave-pipelines almost depend on manual. Global standard tools for the design of regular pipelines are not always useful for wave-pipelines. Automatic design is one of most important issues of wave-pipelines. Thus, we have explored a CAD tool that dynamically shows a wave map and a double clocking scheme.

The wave is a physical phenomenon observed in any circuit, but any circuit can not be a wave-pipeline. The condition necessary for being a wave-pipeline is the avoidance of waves collision. To clear this critical condition, a wave map shown in Fig. 4 is very useful for designing. It is the direct expression of waves on a two-dimensional design space. The wave map is drawn in two ways. The one maps the position of waves on a logic synthesis plane filled with nets. The other maps the position of waves on a physical layout plane. Fig. 4 exemplifies a case failed in wave-pipelining.

Table 1. Characteristics and issues of wave pipelines.

Aspect	Wave	Regular
Clock	Fast	Moderate
Power	Low	Moderate
Area	Small	Moderate
Register	Needless	Need
CAD tool	Inopportune	Opportune
Designing effort	Cumbersome	Moderate

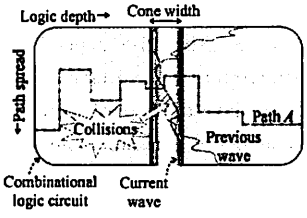

Fig. 4. Wave map.

We have developed a prototype of a graphic tool that inter-faces a timing report and GUI. Fig. 5 exemplifies a wave map in the case of an 8-bit ALU that is waved by using a teaching material of logic synthesis [4].

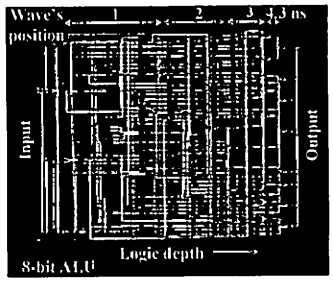

Fig. 5. An example of a wave map.

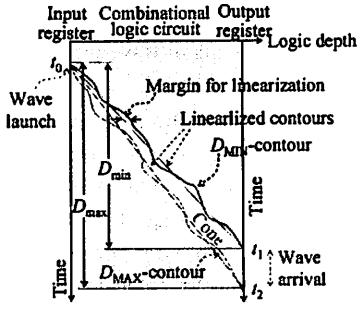

Fig. 6. Cone model.

A regular way to understand the behavior of the wave has been the cone model shown in Fig. 6 [5]. It represents well the time dependency of waves. But, the

1-dimensional representation by the cone model is not suited to explicitly show the form of waves in the deep submicron era. Especially, it is almost hard to precisely show a non interference region between continuous waves in a VLSI circuit. Table 2 summarizes the ability to represent circuit behavior by the wave map and regular methods.

Not only evaluation by tools but also monitoring real chips is necessary for the implementation of wave-pipelines. But, the measurement of wave-pipelines is hard due to its extremely high speed operation. In addition, delay times strongly depend on operating condition like temperature and voltage. Regular methods like a logic analyzer have taken hard works in controlling timing period by a timing generator and adjusting speed of I/O buffers and a chip core.

Table 2. Circuit behavior representation ability of wave map vs. regular methods.

Means	Definition	Coordinates	Design process	Wave representation: Time dependency: Progress of a wave	Wave representation: Time dependency: Location of parallel waves	Wave representation: Two-dimensional distribution	Resultant wave degree
Wave map	Layout mapping the position of a wave or waves	Physical dimensions on a real chip	Physical layout	Possible by presenting time-variant positions of a wave	Possible by presenting the positions of parallel waves at an instant	Suitable due to the explicit 2-dimensional space coordinates	Final
	Schematic circuit diagram mapping the position of a wave or waves	Logic depth vs. signal path spread	Logic synthesis				Accurate
Cone model	Linearization of D_{max} and D_{min} contours of a wave or waves on the coordinates plane	Logic depth vs. delay time		Suitable due to the explicit time coordinate		Unsuitable due to the 1-dimensional space coordinate	Tentative
Timing report, chart		Time at a fixed point					

DFT we have taken for wave-pipelines is the double clocking scheme shown in Fig. 7 [6]. This scheme surrounds a high speed waved-circuit by a regular circuit, which is driven by an external clock. Since the external clock is slower, the testing of wave-pipelines is very easy.

5. Extension of Wave-Pipelines Application

As shown in Table 3, wave-pipelines have been exclusively applied for unifunctional CLB. Although multifunctional CLB and sequential logic circuits play important roles in any processor and controller, they have not been so far wave-pipelined. For the practical development of these circuits, the essence of the application of wave-pipelining is discussed.

The wave-pipelining of multifunctional units realizes wide-range dynamic ILP at a rate higher than can be achieved by regular superscalar architecture. Although much emphasis has been put on thread-level parallelism in recent years [17], the performance enhancement of the dynamic extraction of ILP is still an important subject even for multithreaded processors [18]. Actually, regular scalar units need instruction scheduling since they are composed of independent unifunctional regular pipelines each with different latency. This shortcoming will be overcome by multifunctional pipelines that execute each function within the same latency. Thus, they do not need instruction scheduling. However, the fully merging of regular pipe-lines results in the drastic reduction of clock speed due to the scale up of circuit structures and thus the increase of critical path delays.

Table 3. Examples of wave-pipelining.

Category		Examples
Combinational logic circuit	Single function	Adder, multiplier, counter, DRAM (Research level), SRAM control of Ultra SPARC III (Commercial level)
	Multifunction	Our study
Sequential circuit		Our study

Multifunctional wave-pipelines are free from not only instruction scheduling but also the reduction of clock speeds because the clock speed of a wave-pipeline is determined by the difference between the critical path delay and the mini-mum path delay. Although both delays naturally increase with the scale up of circuits, the balancing of them is essential in wave-pipelining. We have so far applied wave-pipelines to multifunctional units like EX-stage and the access of register file.

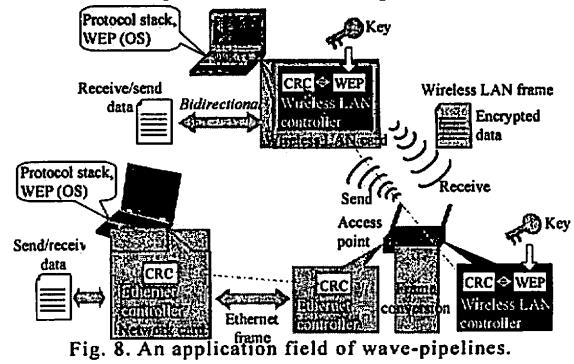

Fig. 8. An application field of wave-pipelines.

The wave-pipelining of sequential logic circuits actually targets CRC (Cyclic Redundancy Check) and WEP (Wired Equivalent Privacy). Although WEP is usually implemented in software and installed in OS, we have developed hardware WEP to achieve power conscious high speed processing. As shown in Fig. 8, CRC and WEP do not work alone. In the various scene of wireless LAN environment, they are frequently used as a

pair of sender and receiver. In addition, they have similar structure. Thus, the effect of wave-pipelining CRC and WEP is very large.

CRC unifies LFSR (Linear Feedback Shift Register) and the hardware implementation of a generator polynomial. WEP includes PRNG (Pseudo Random Number Generator). This is also LFSR accompanied with input and output terminals. In a word, LFSR, CRC, and PRNG have the same structure that is a serial connection of D-FFs feedback by using XOR as shown in Fig. 9. The common structure can be waved by using the method shown in Fig. 3.

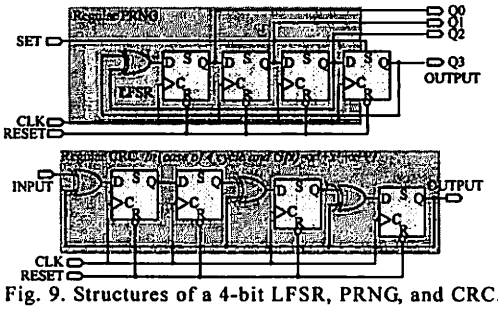

Fig. 9. Structures of a 4-bit LFSR, PRNG, and CRC.

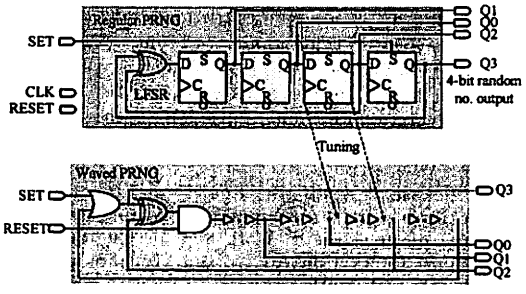

Fig. 10. Wave-pipelining of a 4-bit PRNG.

A regular way to understand the behavior of the wave has been the cone model shown in Fig. 6 [5]. It represents well the time dependency of waves. But, the 1-dimensional representation by the cone model is not suited to explicitly show the form of waves in the deep submicron era. Especially, it is almost hard to precisely show a non interference region between continuous waves in a VLSI circuit. Table 2 summarizes the ability to represent circuit behavior by the wave map and regular methods.

Fig. 10 exactly shows the wave-pipelining of a 4-bit PRNG. Changing the bit width, we have evaluated the clock speed and power dissipation of PRNG. As shown in Fig. 11, the effect of wave-pipelining is obvious. Wave-pipelining increases clock speed 1.4 times and decreases power dissipation 15% com-pared with regular pipelines. As for the dependency on the number of bits or D-FFs, power dissipation increases as the increase of bits, while clock speed is constant. The independency of clock speed on bit width is because there is no timing variation at the clock terminals of LFSR when the bit width is increased. LFSR and CRC have almost the same effect on wave-pipelining as PRNG, because they have a similar structure. They have also similar dependency on bit width.

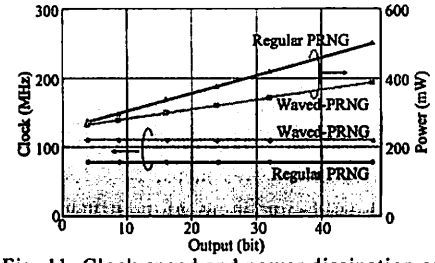

Fig. 11. Clock speed and power dissipation of a waved-PRNG vs. regular PRNG.

A regular way to understand the behavior of the wave has been the cone model shown in Fig. 6 [5]. It represents well the time dependency of waves. But, the 1-dimensional representation by the cone model is not suited to explicitly show the form of waves in the deep submicron era. Especially, it is almost hard to precisely show a non interference region between continuous waves in a VLSI circuit. Table 2 summarizes the ability to represent circuit behavior by the wave map and regular methods.

Table 4. Prototypes of wave-pipelines.

Chip		Process	Area		Number of gates	Wave degree	Clock speed	Voltage	Power	Functions
			Chip	Core						
ALU	8-bit	0.5μm	2.3mm square	0.9mm× 0.6mm	863	8	1GHz	3.3		+, -, AND, OR, NOT, EXOR
	16-bit	0.6μm	4.6mm square	3.3mm square	780	2	750MHz	3.3		16 functions like +, -, AND, OR, NOT, EXOR
	32-bit	0.35μm	4.93mm square	2.95mm square	15k	4	450MHz	3.3		+, -, ×, AND, OR, NOT, EXOR, shift
Wired/ wireless LAN controller	PRNG	FPGA	EPM7128LC84-15			48	110MHz	5.0	300mW	RNG
		1.2μm	2.33mm square			4	800MHz	3.3		RNG
	CRC	FPGA	EPM7128LC84-15			48	110MHz	5.0	300mW	FCS
CPU	gorilla035	0.35μm	4.93mm square	2.95mm square	26k	2	240MHz	3.3	500-700mW	EX
	HCgorilla018	0.18μm	2.8mm square	1.38mm square		2	400MHz	3.3	?100 mW	EX, register file access
Implementation & evaluation		0.35μm						3.3		

Table 4 summarizes the prototypes of wave-pipelines we have so far made. We have exploited the wave-pipelining of both standard cell and FPGA. Fig. 12 shows a 4-bit PRNG we have made by using a CMOS standard cell chip [8]. This implements both the waved- and regular PRNG to compare them.

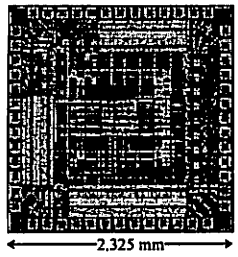

Fig. 12. Die photo of a 4-bit PRNG.

6. Conclusion

We have described in this paper our studies for further enhancement of processor specifications in the era after Moore's law. Three techniques we have developed for the global standardization of wave-pipelines are the graphical interface tool to draw wave maps, DFT for high speed behavior, and wave-pipelining of multifunctional units and sequential logic circuits.

The graphic power of the wave map, which exactly shows instantaneous wave's shape on an actual circuit design space, is definitely superior to regular methods. DFT we have explored is a double clocking scheme. This is a practical testing method of extremely high speed wave-pipelines. Multifunctional wave-pipelines are useful to achieve wide-range dynamic ILP at a rate higher than regular superscalar architecture. The wave-pipelining of sequential logic circuits has been studied considering practical demands for wireless LAN environment.

The next steps of our study are the unification of wave-pipelining and repeater-inserting [19], and the improvement of the graphic tool to be applicable for physical layout.

Acknowledgement

This work is supported by VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Synopsys, Inc. and Cadence Design Systems, Inc. We are financially supported in part by the dean of Faculty of Science and Technology, Hirosaki University.

References

[1] M. Fukase, T. Sato, R. Egawa, and T. Nakamura, "Scaling up of Wave-Pipelines," Proc. of the Fourteenth International Conference on VLSI Design, pp. 439-445, Jan. 2001.

[2] M. J. Flynn, P. Hung, and K. W. Rudd, "Deep-Submicron Microprocessor Design Issues," IEEE MICRO, pp. 11-22, Jul.-Aug. 1999.

[3] T. Sato, K. Kikuchi, and M. Fukase, "A PRNG Circuit on PLD with Feature of Low-Power, High-Speed, and Various Generation of Random Number Sequence," Proc. of TENCON 2006, Nov. 2006.

[4] M. Fukase and T. Sato, "A Graphic Approach for Wave-Pipelines Design," Proc. of ISCIT, pp. 688-693, Oct. 2004.

[5] W. P. Burleson, M. Ciesielski, F. Klass, and W. Liu, "Wave-pipelining: A Tutorial and Research Survey," IEEE Trans. on Very Large Scale Integration (VLSI) Systems, Vol. 6, No. 3, pp. 464-474, Sept., 1998.

[6] T. Sato, R. Egawa, M. Fukase, and T. Nakamura, "Exploitation of a Test Circuit for High Speed Power Conscious Waved Circuits," Social Information, Vol.13, No. 2, pp. 99-108, Mar. 2004.

[7] M. Fukase, T. Sato, R. Egawa, and T. Nakamura, "Break-through of Suparscalar Processors by Multifunctional Wave-Pipelines," Proc. of 9th NASA Symposium on VLSI Design, pp. 6.3.1-6.3.17, Nov. 2000.

[8] R. Sakuma, M. Fukase, and t. Sato, "Waved-LFSR," Ac-tivity Report of VLSI Design and Education Center, The Univ. of Tokyo, p. 150, Sept. 2006.

[9] T. Mudge, "Power: A First-Class Architectural Design Constraint," Computer Magazine, Vol. 34, No. 4, pp. 52-58, Apr., 2001.

[10] D. Geer, "Chip Makers Turn to Multicore Processors," Computer Magazine, Vol. 38, No. 5, pp. 11-13, May, 2005.

[11] H. Kasahara and K. Kimura, "Multicores Emerge as Next Generation Microprocessors," IPSJ Magazine, Vol. 47, No. 1, pp. 10-16, Jan. 2006.

[12] Y. Lin, C. Kozyrakis, Ali-Reza Adl-Tabatabai, "Multi-core Programming: From Threads to Transactional Memory," HOT Chips 18, Aug. 2006.

[13] A. Mukherjee and M. M.-Sadowska, "Wave Steering to Integrate Logic and Physical Syntheses," IEEE Trans. on VLSI Syst. Vol. 11, No. 1, pp. 105-120, Feb. 2003.

[14] M. Fukase, T. Sato, R. Egawa, and T. Nakamura, "Designing a Wave-Pipelined Vector Processor," Proc. of the Tenth Workshop on Synthesis and System Integration of Mixed Technologies, pp. 351-356, Oct. 2001.

[15] M. Fukase, T. Sato, R. Egawa, and T. Nakamura, "A Wave-Pipelined Biprocessor Achieving Remarkable Compati-bility between Low Power and High Speed," Proc. of 10th NASA Symposium on VLSI Design, pp. 8.3.1-8.3.8, Mar. 2002.

[16] T. Sato, R. Sakuma, D. Miyamori, and M. Fukase, "Waved-PRNG for a Wave-Pipelining Test Circuit," Proc. of 12th NASA Symposium on VLSI, pp. 3.3.1-3.3.4, Oct. 2005.

[17] L. Hammond, B. A. Hubbert, M. Siu, M. K. Prabhu, M. Chen, and K. Olukotun, "The Stanford Hydra CMP," IEEE MICRO, Vol. 20, No. 2, pp. 71-84, Mar./Apr. 2000.

[18] V. Krishnan and J. Torrellas, "A Chip-Multiprocessor Architecture with Speculative Multithreading," IEEE Trans. on COMPUTERS, Vol. 48, No. 9, pp. 866-880, Sept. 1999.

[19] D. Sylvester and K. Keutzer, "A Global Wiring Paradigm for Deep Submicron Design," IEEE Trans. on CAD of IC and Syst., Vol.19, No. 2, pp. 242-252, Feb. 2000.