Logic-In-Control-Architecture-Based Reconfigurable VLSI Using Multiple-Valued Differential-Pair Circuits

Nobuaki OKADA^{†a)}, Nonmember and Michitaka KAMEYAMA[†], Fellow

A fine-grain bit-serial multiple-valued reconfigurable SUMMARY VLSI based on logic-in-control architecture is proposed for effective use of the hardware resources. In logic-in-control architecture, the control circuits can be merged with the arithmetic/logic circuits, where the control and arithmetic/logic circuits are constructed by using one or multiple logic blocks. To implement the control circuit, only one state in a state transition diagram is allocated to one logic block, which leads to reduction of the complexity of interconnections between logic blocks. The fine-grain logic block is implemented based on multiple-valued current-mode circuit technology. In the fine-grain logic block, an arbitrary 3-variable binary function can be programmed by using one multiplexer and two universal literal circuits. Three-variable binary functions are used to implement the control circuit. Moreover, the hardware resources can be utilized to construct a bit-serial adder, because full-adder sum and carry can be realized by programming in the universal literal circuit. Therefore, the logic block can be effectively reconfigured for arithmetic/logic and control circuits. It is made clear that the hardware complexity of the control circuit in the proposed reconfigurable VLSI can be reduced in comparison with that of the control circuit based on a typically sequential circuit in the conventional FPGA and the fine-grain field-programmable VLSI reported until now. key words: multiple-valued current-mode logic, fine-grain reconfigurable VLSI, direct allocation, control circuit, sequential logic circuit

1. Introduction

Reconfigurable VLSIs such as field-programmable gate arrays (FPGAs) are widely used to implement special purpose processors. The advantages are low-cost for small volume products and short time-to-market. However, there are problems of large delay time and large area in comparison with application specific integrated circuits due to low utilization of a logic block and complexity of an interconnection network [1]–[3].

On the other hand, fine-grain bit-serial reconfigurable VLSI has potential advantages in that fine-grain pipelining and high utilization of a logic block make performance and parallelism high. Localized data transfer architecture based on direct allocation of a control/data flow graph (CDFG) is also introduced to reduce the complexity of switch blocks and interconnections [4]–[6].

Ordinary processor architecture requires some control and some data. Control circuits drive control signals of the datapath component in each clock cycle. Therefore, not only data-path architecture but control architecture gives significant impact on performance of the whole system [7]. In the direct allocation, each node in a CDFG corresponds to processing in each arithmetic/logic circuit. Therefore, dynamic data-path control is not required for sharing hardware resources. However, the utilization ratio of arithmetic/logic circuits sometimes becomes low, because the processing of each node in a CDFG is not always required to be executed by the corresponding arithmetic/logic circuit.

To solve the problem, hybrid programming scheme based on wired programming and dynamic data-path control programming is effective. In wired programming, interconnection programming is done according to direct allocation of a CDFG. On the other hand, multiplexer control signals are applied to arithmetic/logic circuits step by step in dynamic data-path control. In dynamic data-path control, logic-in-control architecture is proposed to design high-speed dynamic control. Distributed control is introduced to make interconnections between arithmetic/logic circuits and control circuits very short. The control circuits are constructed by using the same logic blocks used in the arithmetic/logic circuits. To implement the control circuit, sequential logic circuit design based on a state machine model is introduced. Also, direct allocation such that only a single state in a state transition diagram of the sequential circuit is allocated to one logic block is introduced to reduce complexity of interconnection between multiple logic blocks [8], [9]. Moreover, the utilization ratio of the fine-grain logic block can be increased.

The proposed fine-grain logic block is designed based on multiple-valued current-mode logic (MVCML) circuit technology. The MVCML circuit technology can be effectively employed for implementing high performance circuit modules and reducing the circuit area in comparison with CMOS implementation [10]–[12].

In the logic block, an arbitrary 3-variable binary function and a bit-serial addition can be realized by using the two universal literal (UL) circuits, a multiplexer and a D-flipflop (D-FF). An arbitrary 3-variable binary function is used to implement both of the logic and control circuits. The bitserial addition can be effectively employed for reduction of the hardware complexity of the arithmetic circuit such as an adder, a subtractor and a multiplier. The UL circuit can be shared as a common hardware resource for an arbitrary 2variable binary function and a full-adder. Therefore, the UL circuits can be used to implement the arithmetic/logic and control circuits, which leads to high utilization ratio of the logic block.

It is made clear that the hardware complexity of the

Manuscript received November 12, 2009.

Manuscript revised March 17, 2010.

[†]The authors are with the Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980–8579 Japan.

a) E-mail: nokada@kameyama.ecei.tohoku.ac.jp

DOI: 10.1587/transinf.E93.D.2126

control circuit can be reduced in comparison with that of the control circuit based on a typically sequential circuit in a conventional FPGA, a fine-grain field-programmable VLSI (FPVLSI) reported until now and a reconfigurable VLSI using encoder-less logic blocks.

2. Review of the Fine-Grain Bit-Serial Field-Programmable VLSI and the Multiple-Valued Current-Mode Logic Circuit Technology

In Field-Programmable VLSIs (FPVLSIs) proposed in [4] and [6], bit-serial architecture and localized data transfer architecture based on direct allocation of a CDFG is effectively employed for reducing the complexity of interconnections. In the direct allocation of a CDFG, each arithmetic/logic circuit executes only a single node as shown in Fig. 1. The arithmetic/logic circuit is realized by using one or multiple cells, where the cell consists of a logic block and switch block. Each edge of a CDFG corresponds to the data transfer between the arithmetic logic circuits. The complexity of logical connections between arithmetic/logic circuits becomes almost same as that of a CDFG. Moreover, the connection can be implemented by a single interconnection in bit-serial architecture. Therefore, the logical data transfer can be usually done by using the localized data transfer network. The performance evaluation was done for the FPVLSIs. In the application of 16-point FFTs, the total throughput of the FPVLSI proposed in [6] is evaluated to be 9 times higher than that of the conventional FPGA under the condition of the same chip area. Moreover, in comparison with the FPVLSI, the fine-grain FPVLSI of [4] can achieve higher performance in a 6-input addition and an absolute differences and addition computation.

Fine-grain multiple-valued reconfigurable VLSIs based on the MVCML circuit technology have been proposed to achieve a high-performance reconfigurable computing [12], [13]. In the MVCML circuit, the logical value can be represented by a current signal. Therefore, linear summation can be realized simply just by wiring, which leads to reduction of the number of active devices as well as wiring complexities. Moreover, the MVCML circuit can be effectively employed for reduction of the number of interconnection switches in the reconfigurable VLSI. Differentialpair circuits (DPCs) are used to realize a high-performance threshold detector, because the DPCs make a signal voltage swing small yet current-driving capability large.

Figure 2 shows a basic structure of the MVCML circuit. A linear summation result of a multiple-valued signal enters to the comparator. The output of the DPC used in the comparator is represented by a complementary voltage signal (V_G , $\overline{V_G}$) is applied to the output generator. Then, a binary current signal (0, I_0) is generated, where (0, I_0) is corresponding to the logical value (0, 1). The voltage swing of the DPC output is an important parameter for reliability and performance. If parameters of pMOS transistors used in the DPC are set to make the voltage swing too small, the output current signal is not equal to 0 or I_0 . On the other



Fig. 1 Direct allocation of a control/data flow graph.



Fig. 2 Basic structure of the multiple-valued current-mode logic circuit.

hand, if the voltage swing is large, the circuit reliability can be improved, but delay time becomes large. In our reconfigurable VLSI, the voltage swings are designed to be more than 0.3 V which is large enough to make the desired current output.

The multiple-valued reconfigurable VLSI proposed in [12] are constructed by using UL-based cells which are effectively employed for implementing arithmetic/logic circuits. The architecture is same as that of the FPVLSIs. In the UL-based cells, an arbitrary 2-variable binary operation, an addition and a subtraction can be programmed. Also, an nxn-bit multiplication can be programmed by using 4n cells. As a result, the area-time product of the UL-based cell can be improved in comparison with CMOS implementation. In the following sessions, design of a multiple-valued cell suitable for both of arithmetic/logic and control circuits is proposed.

3. Architecture of the Reconfigurable VLSI Using Multiple-Valued Differential-Pair Circuits

As shown in Fig. 3, the proposed reconfigurable VLSI consists of many identical cells, where each cell can be connected to its eight adjacent cells through 1-bit programmable switches. The cell consists of a logic block and a switch block. Bit-serial architecture is introduced to reduce the complexity of interconnections. Therefore, highly parallel cell array can be constructed. Moreover, a high bit-level utilization ratio of the cell can be achieved, because the bitlevel utilization ratio does not depend on word length.

A behavioral description given by a CDFG specifies the sequences of operations to be performed by the reconfigurable VLSI. Direct allocation of a CDFG can make the interconnection between arithmetic/logic circuits very simple. Wired programming based on direct allocation of a CDFG is effectively employed for reduction of the complexity of interconnections. However, the utilization ratio of arithmetic/logic circuits sometimes becomes low, because the processing of each node in a CDFG is not always required to be executed by the corresponding arithmetic/logic circuit. On the other hand, dynamic data-path control makes it possible to improve the utilization ratio of arithmetic/logic circuits as shown in Fig. 4, although additional hardware resources such as multiplexers and switch blocks are necessary to keep the performance. Hybrid programming scheme based on wired programming and dynamic data-path control programming is introduced to increase the utilization ratio of arithmetic/logic circuits with small overhead of additional hardware resources.

As shown in Fig. 5, the given CDFG is divided into some subgraphs which are defined as macro nodes. We apply dynamic data-path control for nodes in a macro node to increase the utilization ratio. Each processing element (PE) composed of an arithmetic/logic circuit and a control circuit executes only a single macro node. Between the macro nodes, direct allocation is introduced. Distributed control is introduced to make interconnections between con-



Fig. 3 Block diagram of the proposed reconfigurable VLSI.



Fig. 4 Allocation based on resource sharing.



Fig. 5 Hybrid programming based on wired programming and dynamic data-path control programming.

trol circuits and arithmetic/logic circuits short, which leads to smaller propagation delay of control signals in comparison with centralized control.

4. Design of the Proposed Logic Block

Figure 6 shows the proposed logic block. In the logic block operations, there are a control mode and an arithmetic/logic mode. One of the operation modes is selected by programming configuration data. The control mode is selected when a control circuit is implemented, and the arithmetic/logic mode is selected when an arithmetic/logic circuit is implemented.

4.1 Control Mode

Let us consider design of a sequential logic circuit based on a Moore-type state transition diagram to design the control circuit. If we introduce the Moore-type sequential logic circuit, its output function will be very simple. To implement the sequential logic circuit, only one state is allocated to a delay element in one logic block. Only one delay element corresponding to the present state is set to "1", and all the others are set to "0". Then, the number of logic blocks becomes the same as that of states. However, complexity of interconnections between logic blocks will be same as that of transition paths between states in the state transition diagram.

Figure 7 (a) shows a design example of the sequential logic circuit, where state transition is done by only one transition path. Assume that the present state is S_0 and the input x = 1 arrives. Then, the logical value "1" stored at the delay element for S_0 is sent to the delay element for S_1 through an AND gate. Figure 7 (b) shows another design example of the sequential logic circuit, where there are two or more transition paths which originate from all the states S_0 , S_1 and S_2 . The output of the state transition function is generated by State Transition Circuit with four inputs. If there are two or more transition paths entering to one state, the delay elements allocated to the original states and x are connected to State Transition Circuit.

The logic block consists of an encoder, a programmable logic module (PLM), and a D-FF. Now, let



Fig. 6 Logic block.



(b) State transition from three states

Fig. 7 Fundamental operation of a sequential logic circuit.

Table 1Truth table of the encoder.

x_1	x_2	<i>x</i> ₃	<i>x</i> ₄	<i>y</i> 0	<i>y</i> 1	<i>y</i> 2
0	0	0	0	0	0	1
1	0	0	0	1	0	0
0	1	0	0	0	1	0
0	0	1	0	1	1	0
0	0	0	1	0	0	0

us consider reduction of the number of inputs of the programmable logic module to make the size of the logic block smaller in comparison with that of State Transition Circuit. There are two cases in the four inputs from the adjacent logic blocks. One is the case where only one of them is "1". The other is the case where all of them are "0". The number of inputs of the programmable logic module can be reduced by using the simple encoder. Table 1 shows a function of the encoder. The encoder outputs y_0 , y_1 and the cell input x_0 are applied to the programmable logic module in which any 3-variable binary functions can be programmed.

The output of a sequential logic circuit is generated by an OR gate connected with the delay elements corresponding to the states whose outputs are "1" in a Moore-type state transition diagram. The output circuit becomes very simple in comparison with that of a Mealy-type sequential logic circuit.

Figure 8 (b) shows a design example of a sequential logic circuit for the state transition diagram of Fig. 8 (a). The number of states is the same as that of logic blocks. Two outputs of the logic blocks corresponding to the states S_1 and S_2 are connected to an OR gate to generate the final output of the sequential logic circuit *y*.

As shown in Fig. 9, a programmable interconnection network is used to transfer the sequential logic input which enters into all the cells in the sequential logic circuit. Final output of the sequential logic circuit with 16 states is generated by the control signal output circuit.

4.2 Arithmetic/logic Mode

As shown in Fig. 10, the programmable logic module consists of two universal literal (UL) circuits, two multiplexers and a D-FF. Inputs y_0 and y_1 are represented by binary current signals. The two inputs are linearly summed by wiring so that the summation result of a multiple-valued signal *S* is applied to the UL circuits. In the programmable logic mod-



Fig. 8 Example of a sequential logic circuit by using the logic blocks.





Fig. 10 Programmable logic module.

ule, any 3-variable binary functions and a bit-serial adder can be programmed.

An arbitrary 2-variable binary function can be programmed in the UL circuit. Any 2-variable binary logic functions are expressed by a 4-valued universal literal shown in Table 2, because two binary inputs $x = \{0, 1\}$ and $y = \{0, 2\}$ can be converted to a 4-valued input *S*. Also, fulladder sum and carry can be programmed in the UL circuit, because three binary inputs $x = \{0, 1\}$, $y = \{0, 1\}$ and $C = \{0, 1\}$ can be converted to one 4-valued input *S* without losing input information. Therefore, the UL circuit can be shared as a common hardware resource for an arbitrary 2-variable binary function, full-adder sum and carry, which leads to high utilization ratio of the logic block. Also, this circuit can be used to make 1-clock delay by using the D-FF provided for the bit-serial adder.

A low-power and compact UL circuit can be implemented by using a series-gating differential-pair circuit [12]. Figure 11 (a) shows the circuit diagram of the UL circuit, where the inputs are represented by (S, \overline{S}) , and where the complement \overline{S} is defined as 3-*S*. A 4-valued universal literal can be realized by programming two line exchangers and two threshold values T_1 and T_2 , where two switching patterns shown in Fig. 11 (b) can be programmed in the line exchanger 1 and the line exchanger 2, and where the threshold values T_1 is selected from 0.5 and 1.5 and T_2 is selected from 1.5 and 2.5.

Table 24-valued universal literal.

S	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
3	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1



5. Evaluation

A test chip based on a 90 nm CMOS design rule is implemented to confirm the operation of the universal literal circuit. The supply voltage is 1.0 V. As shown in Fig. 12, a desired input/output waveform of a bit-serial adder using two universal literal circuits can be observed, where the frequency is 10 kHz.

The prototype of the reconfigurable VLSI is designed based on a 65 nm CMOS design rule as shown in Fig. 13, where the supply voltage is 1.2 V. These chips are implemented based on full-custom design. The 16 × 32 cellular array is constructed in the prototype chip. The cell size is $19 \,\mu\text{m} \times 36 \,\mu\text{m}$. The inputs and outputs of the proposed cells and the circuit modules used in the cell are represented by complementary dual-rail voltage signals, which leads to highly immune to common mode noise.

The performance of the cell is evaluated based on HSPICE simulation using a 65 nm CMOS design rule. Table 3 shows the comparison result of propagation delay between two adjacent cells, power consumption and the number of transistors. Power consumption of the proposed cell



Fig. 12 Waveform of the bit-serial adder using two UL circuits in a chip based on a 90 nm CMOS design rule.



Fig. 13 Chip layout based on a 65 nm CMOS design rule.

OKADA and KAMEYAMA: LOGIC-IN-CONTROL-ARCHITECTURE-BASED RECONFIGURABLE VLSI USING MULTIPLE-VALUED DIFFERENTIAL-PAIR CIRCUITS 2131

Tuble 5	5 Terformanee comparison.				
	Encoder-less cell	Proposed cell			
Propagation delay [nsec]	0.42	0.41			
Power consumption [µW]	175	210			
The number of transistors	568	689			

Performance comparison

Table 3





becomes 1.2 times higher than that of the encoder-less cell, and the number of transistors becomes 1.2 times larger. On the other hand, the complexity of interconnections in a control circuit can be greatly reduced.

Let us consider a simple example of the state transition diagram of Fig. 14 (a). Only five cells are sufficient to construct the control circuit in the proposed reconfigurable VLSI as shown in Fig. 14 (b). On the other hand, there are many cells used to relay data signals in a reconfigurable VLSI using the encoder-less cells as shown in Fig. 14 (c),



Fig. 15 Ordinary sequential logic circuit.



Fig. 16 Cell used to relay data signals.



Fig. 17 Fine-grain bit-serial FPVLSI.

where an ordinary sequential logic circuit is designed as the control circuit of Fig. 15 [14]. Moreover, the data transfer is done through the switch block in the cell as shown in Fig. 16, which leads to serious degradation of delay time. Therefore, it is clear that the performance of the control circuit can be improved in the proposed reconfigurable VLSI.

The number of transistors of the control circuit in the proposed reconfigurable VLSI is compared with those in a conventional FPGA and a fine-grain FPVLSI. We select the architecture of Xilinx XC4000E as a typical conventional FPGA. The logic block contains two 4-input look-up-tables (4-LUTs), a 3-LUT and two registers. As one of the highest performance fine-grain reconfigurable VLSIs, we select the bit-serial fine-grain FPVLSI reported in [4]. In the FPVLSI, a cell can be connected to its four adjacent cells as shown in Fig. 17. An arbitrary 2-variable binary function including the full-adder sum function can be programmed, and one D-FF can be used. The ordinary sequential logic circuit of Fig. 15 is designed by using the FPGA and the FPVLSI as the control circuit.

Figure 18 and Table 4 are the evaluation results under the assumption that the number of cells are equal to that of states only in the proposed reconfigurable VLSI, where a sequential logic input is applied to the control circuit. The transistor counts of the FPGA and the FPVLSI are evaluated by counting the number of transistors, where those logic



Fig. 18 The number of transistors comparison.

Table 4Ratio of the number of transistors.

	2 states	4 states	8 states	16 states	Average
FPGA	1	1	1	1	1
FPVLSI	0.23	1.79	2.08	1.86	1.49
Proposed	0.44	0.88	0.88	0.88	0.77

blocks, switch/connection blocks are shown in Refs. [15] and [4], respectively. In the typical sequential circuit, the total number of state and external input variables is m + k, if the number of states and external inputs are 2^m and k, respectively. An m + k-variable logic functions are required to be provided for the state-transition and output functions of the sequential circuit, which can be designed using multiple LUTs in logic blocks. In the FPGA design, a cell is composed of a logic block, a switch block and two connection blocks. In the evaluation, we assume that the whole cell is used as a part of the sequential circuit if the LUT or the register in the cell is used, because interconnections between the logic circuits and registers in the sequential circuit will be complex. The average transistor count of the proposed reconfigurable VLSI can be reduced to 77% and 47% in comparison with those of the FPGA and the FPVLSI, respectively. The number of transistors in the FPVLSI increases greatly as the number of states becomes large, because many cells are required to relay data signals.

6. Conclusion

In this paper, we propose the reconfigurable VLSI based on logic-in-control architecture to reduce propagation delay of control signals and hardware complexity of a control circuit. The multiple-valued current-mode circuit technology is introduced in implementing of the fine-grain logic block. To simplify the complexity of interconnections between cells in the control circuit, direct allocation of a state transition diagram is introduced. Moreover, high-utilization ratio of the logic block will be achieved in both of the arithmetic/logic and the control modes.

As future work, the number of interconnection switches in the switch block will be reduced if the inputs and outputs of the logic block can be represented by multiplevalued current signals. Also, it is necessary to make system level evaluation through practical applications such as image and signal processing. To make CAD tools for high utilization ratio of hardware resources in the hybrid programming is an interesting issue.

Acknowledgment

This work is supported by VLSI Design and Education Center (VDEC), the University of Tokyo with the collaboration with Cadence Corporation, Synopsys Corporation and Mentor Graphics Corporation.

References

- P. Chow, S.O. Seo, J. Rose, K. Chung, G. Paez-Monzon, and I. Rahardja, "The design of an SRAM-based field-programmable gate array-Part I, Architecture," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.7, no.2, pp.191–197, 1999.
- [2] S. Vassiliadis and D. Soudris, ed., Fine- and Coarse-Grain Reconfigurable Computing, Springer, 2007.
- [3] I. Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," IEEE Trans. Computer-Aided Des. Integr. Circuits Syst., vol.26, no.2, pp.203–215, 2007.
- [4] M. Hariyama, W. Chong, and M. Kameyama, "Field-programmable VLSI based on a bit-serial fine-grain architecture," IEICE Trans. Electron., vol.E87-C, no.11, pp.1897–1902, Nov. 2004.
- [5] A. Ohta, T. Isshiki, and H. Kunieda, "A new FPGA architecture for high performance bit-serial pipeline datapath," IEICE Trans. Fundamentals, vol.E83-A, no.8, pp.1663–1672, Aug. 2004.
- [6] N. Ohsawa, O. Sakamoto, M. Hariyama, and M. Kameyama, "Program-counter-less bit-serial field-programmable VLSI processor with mesh-connected cellular array structure," Proc. IEEE Computer Society Annual Symposium on VLSI, pp.258–259, 2004.
- [7] J. Henkel and S. Parameswaran, ed., Designing Embedded Processors: A Low Power Perspective, Springer, 2007.
- [8] T.L. Van and N.V. Houtte, "Delayed universal logic modules and sequential machine synthesis," IEEE Trans. Comput., vol.C-24, no.8, pp.853–855, 1975.
- [9] T. Ito and M. Kameyama, "Universal VLSI based on a redundant multiple-valued sequential logic operation," Proc. 37th IEEE International Symposium on Multiple-Valued Logic, CDROM-no.32, 2007.
- [10] T. Ike, T. Hanyu, and M. Kameyama, "Fully source-coupled logic based on multiple-valued VLSI," Proc. 32nd IEEE International Symposium on Multiple-Valued Logic, pp.270–275, 2002.
- [11] H. Shirahama, A. Mochizuki, T. Hanyu, M. Nakajima, and K. Akimoto, "Design of a processing element based on quaternary differential logic for a multi-core SIMD processor," Proc. 37th IEEE International Symposium on Multiple-Valued Logic, 2007.
- [12] N. Okada and M. Kameyama, "Fine-grain multiple-valued reconfigurable VLSI using series-gating differential-pair circuits and its evaluation," IEICE Trans. Electron., vol.E91-C, no.9, pp.1437–1443, Sept. 2008.
- [13] H.M. Munirul, T. Hasegawa, and M. Kameyama, "Implementation and evaluation of a fine-grain multiple-valued field programmable VLSI based on source-coupled logic," Proc. 35th IEEE International Symposium on Multiple-Valued Logic, pp.120–125, 2005.
- [14] M.A. Karim and X. Chen, Digital Design, CRC Press, 2008.

[15] [Online]. Available:

http://www.xilinx.com/support/documentation/data_sheets/4000.pdf



Nobuaki Okada received the B.E. degree in Information Engineering and M.S. degree in Information Sciences from Tohoku University, Sendai, Japan, in 2006 and 2008, respectively. He is currently working toward the Ph.D. in Graduate School of Information Sciences, Tohoku University. His research interests include reconfigurable computing and multiplevalued VLSI computing.



Michitaka Kameyama received the B.E., M.E. and D.E. degrees in Electronic Engineering from Tohoku University, Sendai, Japan, in 1973, 1975, and 1978, respectively. He is currently a Dean and a Professor in the Graduate School of Information Sciences, Tohoku University. His general research interests are intelligent integrated systems for real-world applications, advanced VLSI architecture, and newconcept VLSI including multiple-valued VLSI computing. He received the Outstanding Paper

Awards at the 1984, 1985, 1987 and 1989 IEEE International Symposiums on Multiple-Valued Logic, the Technically Excellent Award from the Society of Instrument and Control Engineers of Japan in 1986, the Outstanding Transactions Paper Award from the IEICE in 1989, the Technically Excellent Award from the Robotics Society of Japan in 1990, and the Special Award at the 9th LSI Design of the Year in 2002. He is an IEEE Fellow.