PAPER

A Bit-Serial Reconfigurable VLSI Based on a Multiple-Valued X-Net Data Transfer Scheme

Xu BAI[†a)], ***Student Member and*** **Michitaka KAMEYAMA**[†], ***Fellow***

SUMMARY A multiple-valued data transfer scheme using X-net is proposed to realize a compact bit-serial reconfigurable VLSI (BS-RVLSI). In the multiple-valued data transfer scheme using X-net, two binary data can be transferred from two adjacent cells to one common adjacent cell simultaneously at each "X" intersection. One cell composed of a logic block and a switch block is connected to four adjacent cross points by four one-bit switches so that the complexity of the switch block is reduced to 50% in comparison with the cell of a BS-RVLSI using an eight nearest-neighbor mesh network (8-NNM). In the logic block, threshold logic circuits are used to perform threshold operations, and then their binary dual-rail voltage outputs enter a binary logic module which can be programmed to realize an arbitrary two-variable binary function or a bit-serial adder. As a result, the configuration memory count and transistor count of the proposed multiple-valued cell are reduced to 34% and 58%, respectively, in comparison with those of an equivalent CMOS cell. Moreover, its power consumption for an arbitrary 2-variable binary function becomes 67% at 800 MHz under the condition of the same delay time.

key words: *multiple-valued data transfer scheme, X-net, multiple-valued current-mode logic, MOS current-mode logic, fine-grain reconfigurable VLSI*

1. Introduction

Field-Programmable Gate Arrays (FPGAs) are cost-effective for small-lot production and flexible because functions and interconnections of logic resources can be directly programmed by end users. However, the disadvantages of the conventional FPGAs are large area and power consumption due to complex programmable switch and connection blocks [1], [2].

A bit-serial reconfigurable VLSI (BS-RVLSI) using an eight nearest-neighbor mesh network (8-NNM) has been proposed to reduce the complexity of the interconnections and switch blocks [3]–[6]. Multiple-valued signaling is effectively employed for reducing the switch blocks, and a binary-controlled differential-pair circuit has been proposed to realize low-power arithmetic logic functions, including a full-adder sum and an arbitrary two-variable binary function. Also, a current-source sharing technique between a differential-pair circuit and a current-mode D-latch has been proposed to reduce power consumption of the current-mode bit-serial pipeline [6].

In the BS-RVLSI using the 8-NNM shown in Fig. 1, each cell composed of a switch block and a logic block is connected to eight adjacent cells [6]. The switch block is not so compact because eight NMOS pass transistors and eight configuration memories are provided at each input/output of the cell to realize an eight-near neighborhood data transfer.

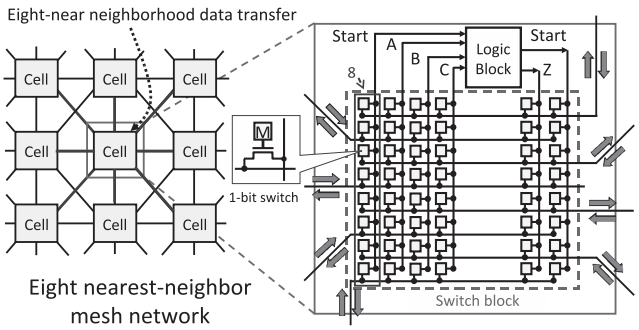


Fig. 1 Architecture of the bit-serial reconfigurable VLSI using an eight nearest-neighbor mesh network.

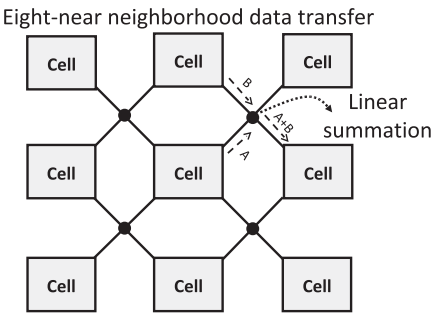


Fig. 2 Multiple-valued data transfer scheme using X-net.

As shown in Fig. 2, X-net is more sufficient than the 8-NNM to realize the eight-near neighborhood data transfer [7]. In this paper, X-net is employed for implementing area-efficient switch blocks without decreasing performance. In X-net, one cell is connected to four cross points, and each cross point is connected to the other three adjacent cells. Therefore only four NMOS pass transistors and four configuration memories are provided at each input/output of the cell to realize the eight-near neighborhood data transfer. Moreover, a multiple-valued data transfer scheme is proposed to improve the utilization of X-net. Linear summation of current signals transferred between cells can be realized at each cross point, which leads to high utilization of hardware resources.

HSPICE simulation of the proposed multiple-valued cell of the BS-RVLSI using X-net is done using a 65-nm

Manuscript received September 10, 2012.
Manuscript revised February 14, 2013.

[†]The authors are with the Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980–8579 Japan.
a) E-mail: baixu@ecei.tohoku.ac.jp

DOI: 10.1587/transinf.E96.D.1449

Copyright © 2013 The Institute of Electronics, Information and Communication Engineers

CMOS design rule. The performance evaluation of the proposed multiple-valued cell is compared with the previous multiple-valued cell and with the equivalent CMOS cell of the BS-RVLSI using the 8-NNM. The configuration memory count and the transistor count of the proposed multiple-valued cell are reduced to 34% and 58%, respectively, in comparison with those of the equivalent CMOS cell. The configuration memory count and the transistor count of the proposed multiple-valued cell are reduced to 61% and 80%, respectively, in comparison with those of the previous multiple-valued cell.

2. Multiple-Valued Data Transfer Scheme Using X-Net

Major single instruction multiple data (SIMD) machines appeared contain a neighborhood interconnection network allowing regular data communications. In [8] and [9], a processor array SIMD architecture, called Massively Parallel Computer (MasPar), is presented. X-net inspired by the MasPar gathers all the cells in a 2-D grid, allowing each cell to communicate with its eight neighbors using a binary data transfer scheme [10]. To transfer a data from $cell_i$ to its right adjacent $cell_{i+1}$, the $cell_i$ transmits out its northeast corner and the $cell_{i+1}$ reads from its northwest corner (one-to-one data transfer). Figure 3 shows three kinds of one-to-one data transfer modes at each "X" intersection. One binary data A can be transferred from the *cell* 1 to the adjacent *cell* 2, *cell* 3 and *cell* 4. However, if two binary data A and B are transferred from the *cell* 1 and *cell* 4 to the common adjacent *cell* 2 simultaneously (two-to-one data transfer), two "X" intersections are required in the binary data transfer scheme as shown in Fig. 4, which results in low utilization of X-net.

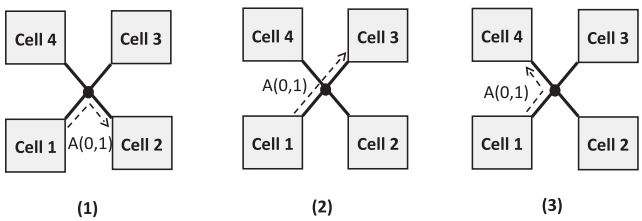


Fig. 3 One-to-one data transfer at each "X" intersection in a binary data transfer scheme.

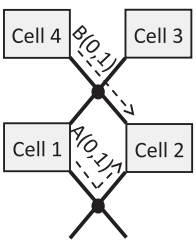


Fig. 4 Two-to-one data transfer in a binary data transfer scheme.

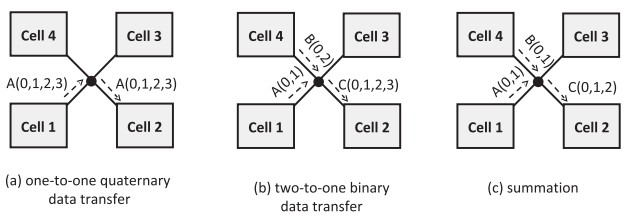


Fig. 5 Three data transfer patterns at each "X" intersection in a multiple-valued data transfer scheme.

To improve the utilization of X-net, we introduce a multiple-valued data transfer scheme shown in Fig. 2, where multiple-valued current signals are transferred between cells. Two binary data A and B from two adjacent cells can be transferred to one common adjacent cell at each "X" intersection (two-to-one data transfer) as shown in Fig. 5 (b). A and B should be (0, 1) and (0, 2), respectively, and C becomes a quaternary data (0, 1, 2, 3) which expresses two-bit information. On the other hand, summation of A and B can be realized at each "X" intersection as shown in Fig. 5 (c). A and B should be (0, 1) and (0, 1), respectively, and C becomes a ternary data (0, 1, 2). All the one-to-one quaternary data transfer, the two-to-one binary data transfer and the linear summation can be realized at each "X" intersection in the multiple-valued data transfer scheme as shown in Fig. 5, which leads to high utilization of X-net.

3. Architecture of the Bit-Serial Reconfigurable VLSI Based on the Multiple-Valued X-Net Data Transfer Scheme

3.1 Direct Allocation of a Control/Data Flow Graph (CDFG)

A behavioral description given by a control/data flow graph (CDFG) specifies the sequence of operations to be performed by the BS-RVLSI. To perform operations in the CDFG, they are mapped into the cells, which task is called allocation. A direct allocation of CDFG on the BS-RVLSI has been introduced to realize high utilization of the cells and reduce the complexity of interconnection networks [11], [12].

Figure 6 shows the direct allocation of the CDFG on

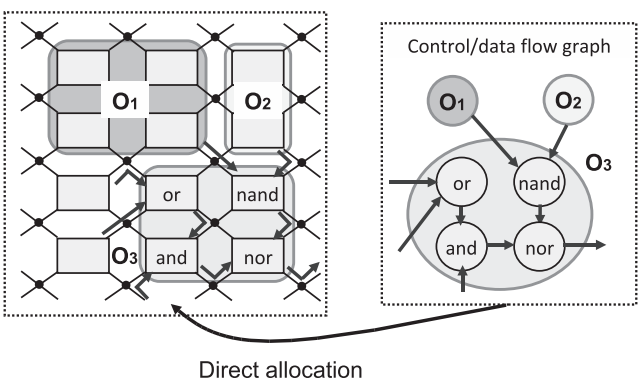


Fig. 6 Direct allocation of a control/data flow graph on the bit-serial reconfigurable VLSI using X-net.

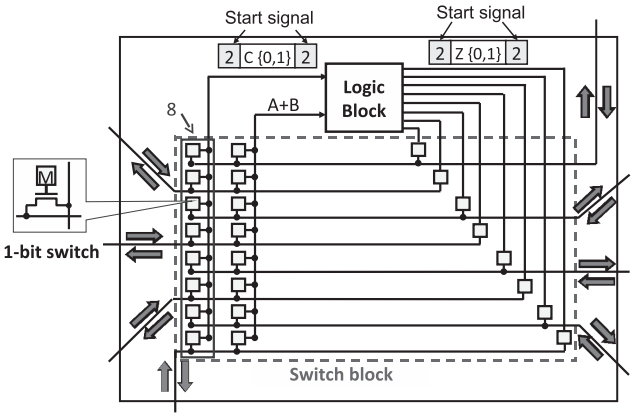


Fig. 7 Multiple-valued cell of the bit-serial reconfigurable VLSI using an eight nearest-neighbor mesh network.

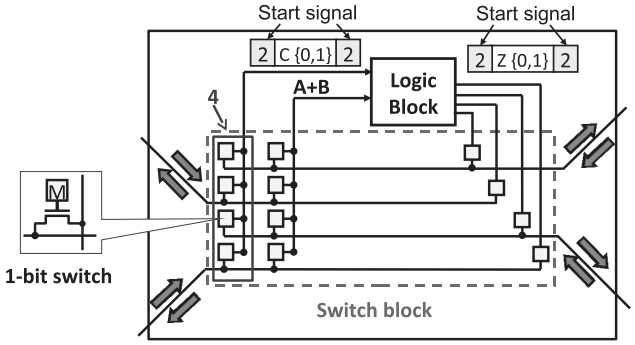


Fig. 8 Multiple-valued cell of the bit-serial reconfigurable VLSI using X-net.

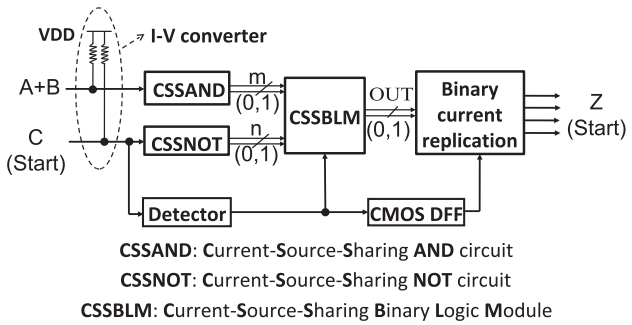


Fig. 9 Multiple-valued logic block.

the BS-RVLSI using X-net, each node in the CDFG corresponds to a macro-block in the BS-RVLSI and each edge corresponds to a data transfer path between the macro-blocks, where the macro-block consists of multiple cells. The complexity of logical connections between the macro-blocks becomes almost same as that of the CDFG. The architecture for the localized data transfer can be effectively employed for reducing the complexity of interconnections and delay due to data transfer between cells.

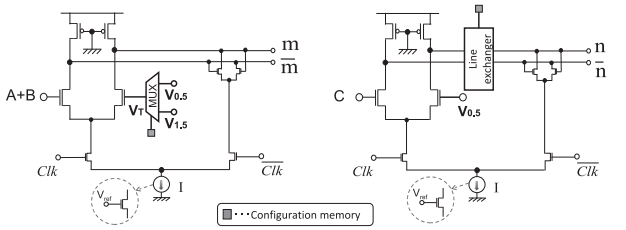
3.2 Multiple-Valued Cell of the Bit-Serial Reconfigurable VLSI Using X-Net

The multiple-valued current-mode logic circuit technology has attractive features to reduce circuit area in comparison with CMOS implementation [13]. In the multiple-valued current-mode logic circuit, linear summation can be performed simply just by wiring, which leads to reduction of the number of active devices as well as the wiring complexity. Moreover, the multiple-valued current-mode logic circuit can be effectively employed for reduction of the number of switch blocks in a reconfigurable VLSI [4]. Since the current flow in the multiple-valued current-mode logic circuit is independent of the operating frequency, its dynamic power dissipation becomes lower at the high operating frequency than that of a corresponding binary CMOS implementation whose dynamic power dissipation is proportional to the operating frequency. In the multiple-valued current-mode logic circuit, a differential-pair circuit (DPC) is effectively used to realize a threshold operation, because the DPC makes a signal-voltage swing small yet current-driving capability large.

As shown in Fig. 7, a multiple-valued cell of the BS-RVLSI using an 8-NNM has been proposed. It has been confirmed that the area of the multiple-valued cell is reduced to 78% in comparison with that of the equivalent CMOS cell [4]. The power consumption becomes 67% at 800 MHz under the condition of the same delay time [6]. Multiple-valued signaling is utilized to reduce the NMOS pass transistor count and configuration memory count in the switch block, and a programmable binary-controlled differential-pair circuit is introduced to realize high-performance low-power arithmetic logic operations including an arbitrary two-variable binary logic function and a full-adder sum [6]. To achieve further complexity reduction of interconnections and switch blocks, a multiple-valued data transfer scheme using X-net is proposed to reduce the NMOS pass transistor count and the configuration memory count in the switch block without decreasing performance.

Figure 8 shows the multiple-valued cell of the BS-RVLSI using X-net. Four NMOS pass transistors and four configuration memories are provided at each input/output of the multiple-valued cell which is connected to four cross points. There are two methods to realize the linear summation of the binary input currents A and B. One is that A and B are linearly summed at the "X" intersection, if A and B are transferred from a common "X" intersection. The other is that A and B are linearly summed in the switch block, if A and B are transferred from two different "X" intersections. In a bit-serial operation, a start signal indicating a head of a one-word data is required to initialize D flip-flops used for a state memory. Superposition of the binary input current C and the start signal in a single interconnection is introduced to implement compact switch blocks, where the logic value "1" and "0" is defined as C and the logic value "2" is defined as the start signal.

As shown in Fig. 9, the multiple-valued logic block consists of a current-to-voltage (I-V) converter, two current-source-sharing threshold logic circuits (CSSTLCs), a start



(a) Current-source-sharing AND circuit (b) Current-source-sharing NOT circuit

Fig. 10 Current-source sharing threshold logic circuits (CSSTLCs).

Table 1 Programmable operations of the current-source-sharing AND circuit.

(a) Dual-rail code

$A+B$	$(m,\overline{m})$
0	(0, 1)
1	(1, 0)

(b) AND type dual-rail code

$A+B$	$(m,\overline{m})$
0	(0, 1)
1	(0, 1)
2	(1, 0)

Table 2 Programmable operations of the current-source-sharing NOT circuit.

(a) Dual-rail code

C	$(n,\overline{n})$
0	(0, 1)
1	(1, 0)

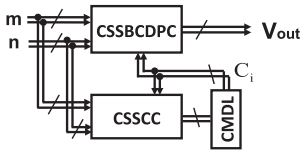
(b) NOT type dual-rail code

C	$(n,\overline{n})$
0	(1, 0)
1	(0, 1)

signal detector, a current-source-sharing binary logic module (CSSBLM), and a current replication circuit [6]. The multiple-valued current signals from the switch block are converted to multiple-valued voltage signals by the I-V converter, and then enter two CSS threshold logic circuits and the start signal detector. The start signal detector is implemented by a one-level differential-pair circuit. The threshold is set "1.5" to make the output "1" for the input logic value "2".

Figure 10 shows the CSSTLCs including a current-source-sharing AND circuit (CSSAND) and a current-source-sharing NOT circuit (CSSNOT). Both the CSSAND and CSSNOT are constructed by a differential-pair circuit. In the CSSAND, the programmable operation shown in Table 1 is performed if *Clk* is high, and the operation result is stored if *Clk* is low. The AND type dual-rail code is used to generate a partial product in a multiplication and the dual-rail code is used in other cases. In the CSSNOT, the programmable operation shown in Table 2 is performed if *Clk* is high, and the operation result is stored if *Clk* is low. The NOT type dual-rail code is used to convert a subtrahend to a 2's complement number in a subtraction and the dual-rail code is used in other cases.

Figure 11 shows the CSSBLM composed of a current-source-sharing binary-controlled differential-pair circuit (CSSBCDPC), a current-source-sharing carry circuit (CSSCC), and a current-mode D-latch (CMDL). The CSSBLM can be programmed to realize an arbitrary two-variable binary function or a bit-serial adder. The arithmetic logic operations are performed when *Clk* is low, and the operation results are stored when *Clk* is high. The CSSBCDPC



CSSBCDPC : **C**urrent-**S**ource-**S**haring **B**inary-**C**ontrolled **D**ifferential-**P**air **C**ircuit
CSSCC : **C**urrent-**S**ource-**S**haring **C**arry **C**ircuit
CMDL : **C**urrent-**M**ode **D**-**L**atch

Fig. 11 Current-source sharing binary logic module (CSSBLM).

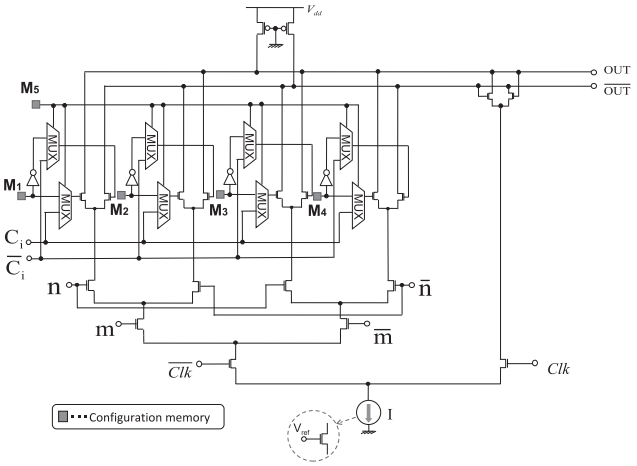


Fig. 12 Current-source-sharing binary-controlled differential-pair circuit (CSSBCDPC).

Table 3 Arbitrary 2-variable binary logic function.

A	B	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	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

can be programmed to realize an arbitrary two-variable binary function or generate the full-adder sum. The CSSCC is used to generate the full-adder carry. The CMDL is used to store the full-adder carry for the bit-serial adder.

Figure 12 shows the CSSBCDPC. The current I produced by the current source is steered into one of the branches in the CSSBCDPC according to the dual-rail binary input voltages. In the first-level differential pair, when *Clk* is low, the current I flows through the left path. The CSSBCDPC is programmed to realize an arbitrary two-variable binary function or generate the full-adder sum. When *Clk* is high, the current flows through the right path and the operation result is stored by two cross-coupled NMOS transistors. The binary voltages $(m,\overline{m})$ and $(n,\overline{n})$ generated by the CSSTLCs are used as the inputs of the second-level and third-level differential pairs, respectively. Multiplexers controlled by a configuration memory M_5 are used to select the inputs of the fourth-level differential pairs. An arbitrary two-variable binary function shown in Table 3

Table 4 Programming of an arbitrary 2-variable binary function.

Function	M_1	M_2	M_3	M_4
f_0	0	1	1	0
f_1	0	1	1	1
f_2	0	1	0	0
f_3	0	1	0	1
f_4	0	0	1	0
f_5	0	0	1	1
f_6	0	0	0	0
f_7	0	0	0	1
f_8	1	1	1	0
f_9	1	1	1	1
f_{10}	1	1	0	0
f_{11}	1	1	0	1
f_{12}	1	0	1	0
f_{13}	1	0	1	1
f_{14}	1	0	0	0
f_{15}	1	0	0	1

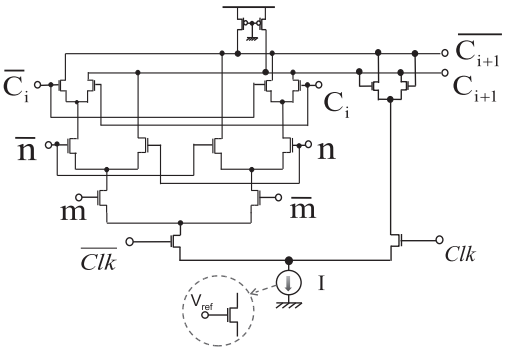


Fig. 13 Current-source-sharing carry circuit (CSSCC).

can be realized, if the configuration memories M_1, M_2, M_3 and M_4 are selected to connect with the third-level differential pairs. Table 4 shows the values of M_1, M_2, M_3 and M_4 and the corresponding function. The full-adder sum can be realized, if the carry signal $(c, \overline{c})$ is selected as the input of the third-level differential pairs. A dual-rail output has two values of V_{dd} and V_{dd}-$\triangle V$, where $\triangle$V is the output voltage swing and is equals to $I \times R$. R is the equivalent resistance of the pMOS load transistor.

Figure 13 shows the CSSCC. When Clk is low, the current I flows through the left path and the full-adder carry is generated. When Clk is high, the current I flows through the right path and the full-adder carry is stored by two cross-coupled NMOS transistors.

4. Evaluation

The evaluation of the proposed multiple-valued cell of the BS-RVLSI using X-net is done based on HSPICE simulation using a 65-nm CMOS design rule. The supply voltage and unit current I are 1.2 V and 10 μA, respectively.

The proposed multiple-valued cell is compared with the previous multiple-valued cell and with an equivalent CMOS cell shown in Fig. 14. The equivalent CMOS cell is designed using the library provided by VDEC. Table 5 shows the comparison result. The configuration memory

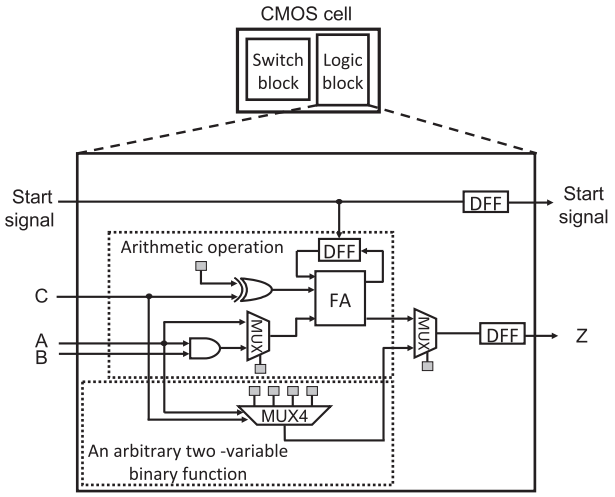


Fig. 14 Equivalent CMOS cell.

Table 5 Comparison results of the cells.

	Equivalent CMOS cell	Previous Multiple-valued cell	Proposed Multiple-valued cell
Supply voltage	1.2 V	1.2 V	1.2 V
Delay	0.4 nS	0.4 nS	0.4 nS
Configuration memory count	55	31	19
Transistor count of configuration memories	330	186	114
Total transistor count	566	412	328
Power consumption @800 MHz	181 μW	121.95 μW* 148.08 μW**	121.73 μW* 147.87 μW**

* An arbitrary 2-variable binary function
** Arithmetic operations

count and the transistor count of the proposed multiple-valued cell are reduced to 61% and 80%, respectively, in comparison with those of the previous multiple-valued cell. The delay and power consumption of the proposed multiple-valued cell are same as those of the previous multiple-valued cell. The configuration memory count and the transistor count of the proposed multiple-valued cell are reduced to 34% and 58%, respectively, in comparison with those of the equivalent CMOS cell. Moreover, its power consumption of an arbitrary 2-variable binary function becomes 67% under the condition of the same delay time.

Let us evaluate the BS-RVLSI using X-net and the BS-RVLSI using the 8-NNM in some applications.

Application 1: Let us consider a 6-input addition, which is one of the fundamental arithmetic operations. Figure 15 shows its data follow graph (DFG). Figures 16 and 17 show the allocation results for the BS-RVLSI using the 8-NNM and the BS-RVLSI using X-net, respectively. Table 6 shows the comparison result. The configuration memory count and the transistor count of the BS-RVLSI using X-net are reduced to 61% and 80%, respectively, in comparison with those of the BS-RVLSI using the 8-NNM. The computation time, power consumption and cell count of the

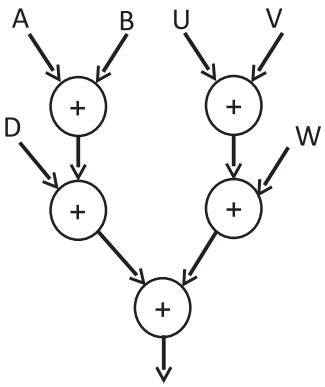


Fig. 15 Data flow graph for the 6-input addition.

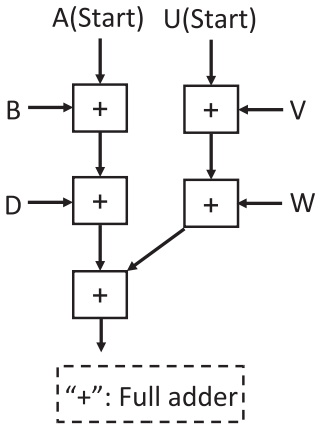


Fig. 16 Allocation of the 6-input addition onto the bit-serial reconfigurable VLSI using the eight nearest-neighbor mesh network.

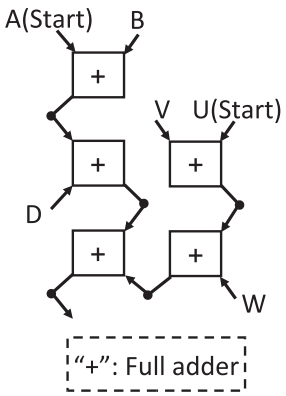


Fig. 17 Allocation of the 6-input addition onto the bit-serial reconfigurable VLSI using X-net.

BS-RVLSI using X-net are same as those of the BS-RVLSI using the 8-NNM.

Application 2: Let us consider a 2×2-bit multiplier, which is another one of the fundamental arithmetic operations. Figures 18 and 19 show the allocation results for the BS-RVLSI using the 8-NNM and the BS-RVLSI using X-net, respectively. Table 7 shows the comparison result. The configuration memory count and the transistor count of

Table 6 Comparison of the 6-input addition modules in bit-serial reconfigurable VLSIs.

	Reconfigurable VLSI using 8 nearest-neighbor network	Reconfigurable VLSI using X-net
Supply Voltage	1.2 V	1.2 V
Computation time	2.4 nS	2.4 nS
Cell count	5	5
Configuration memory count	155	95
Transistor count	2060	1640
Power consumption @800 MHz	740.4 μW	739.35 μW

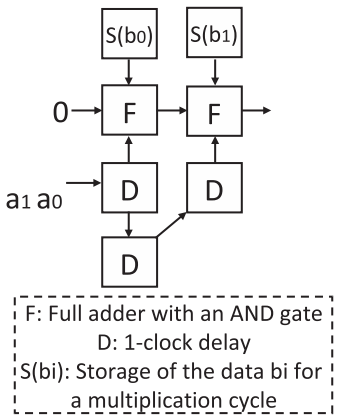


Fig. 18 Allocation of the 2×2-bit multiplier onto the bit-serial reconfigurable VLSI using the eight nearest-neighbor mesh network.

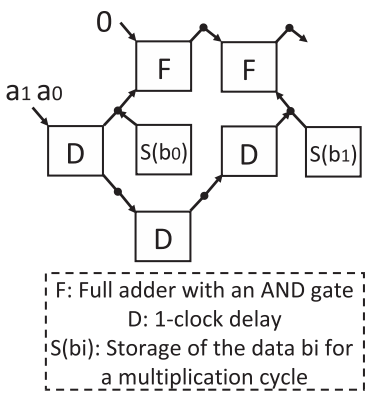


Fig. 19 Allocation of the 2×2-bit multiplier onto the bit-serial reconfigurable VLSI using X-net.

the BS-RVLSI using X-net are reduced to 61% and 80%, respectively, in comparison with those of the BS-RVLSI using the 8-NNM. The computation time, power consumption and cell count of the BS-RVLSI using X-net are same as those of the BS-RVLSI using the 8-NNM.

Application 3: Let us consider a sum-of-absolute-differences operation which is widely used as a similarity measure in template matching. The sum-of-absolute-differences operation is expressed as

$$SAD = |A1 - B1| + |A2 - B2| + \cdots + |A16 - B16| \quad (1)$$

Table 7 Comparison of the 2×2-bit multiplier modules in bit-serial reconfigurable VLSIs.

	Reconfigurable VLSI using 8 nearest-neighbor network	Reconfigurable VLSI using X-net
Supply Voltage	1.2 V	1.2 V
Computation time	1.6 nS	1.6 nS
Cell count	7	7
Configuration memory count	217	133
Transistor count	2884	2296
Power consumption @800 MHz	905.91 μW	904.39 μW

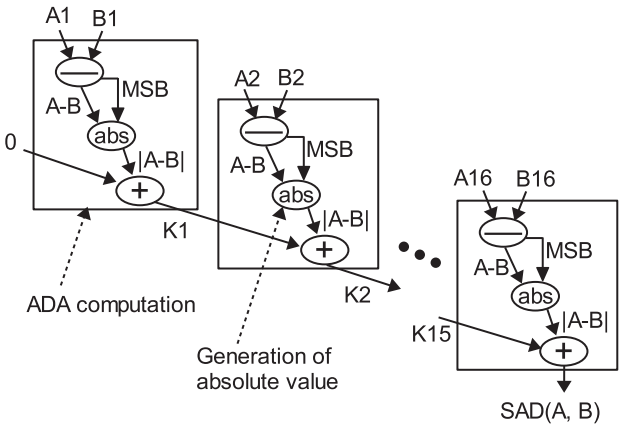


Fig. 20 Control/data flow graph for a sum-of-absolute-differences operation.

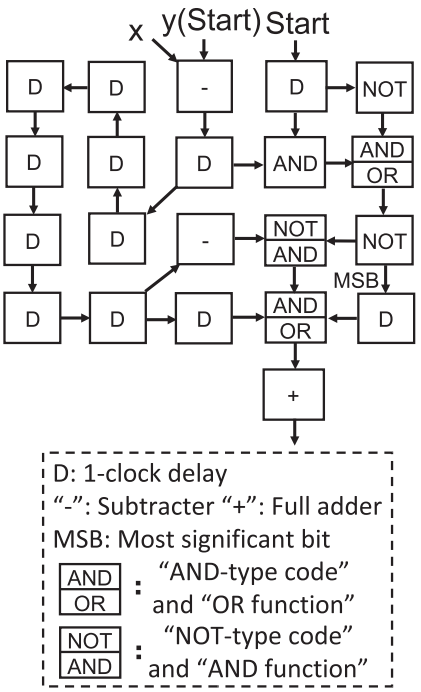


Fig. 21 Allocation of the absolute difference operation and addition onto the bit-serial reconfigurable VLSI using the eight nearest-neighbor mesh network.

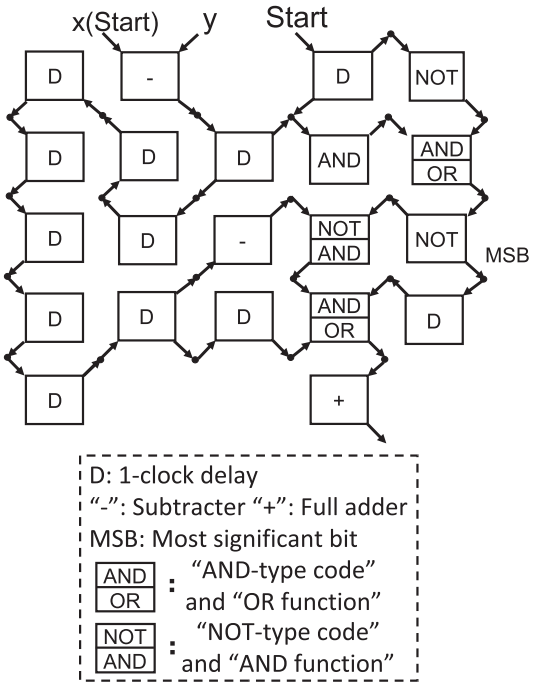


Fig. 22 Allocation of the absolute difference operation and addition onto the bit-serial reconfigurable VLSI using X-net.

Table 8 Comparison of the absolute absolute difference operation and addition modules in bit-serial reconfigurable VLSIs.

	Reconfigurable VLSI using 8 nearest-neighbor network	Reconfigurable VLSI using X-net
Supply Voltage	1.2 V	1.2 V
Computation time	8 nS	8 nS
Cell count	21	21
Configuration memory count	651	399
Transistor count	8652	6888
Power consumption @800 MHz	2639.34 μW	2634.75 μW

where the CDFG is shown in Fig. 20. The sum-of-absolute-differences operation is performed by iteration of an absolute difference operation and addition. Figures 21 and 22 show the allocation results of the 8-bit absolute difference operation and addition for the BS-RVLSI using the 8-NNM and the BS-RVLSI using X-net, respectively. Table 8 shows the comparison result. The configuration memory count and the transistor count of the BS-RVLSI using X-net are reduced to 61% and 80%, respectively, in comparison with those of the BS-RVLSI using the 8-NNM. The computation time, power consumption and cell count of the BS-RVLSI using X-net are same as those of the BS-RVLSI using the 8-NNM.

5. Conclusion

This paper presented a multiple-valued data transfer scheme using X-net and its application to a bit-serial reconfigurable

VLSI. The key advantage is that the multiple-valued X-net data transfer scheme was effectively introduced for reducing the complexity of the interconnections and switch blocks in the bit-serial reconfigurable VLSI without decreasing performance. It was demonstrated that the configuration memory count and the transistor count of the multiple-valued bit-serial reconfigurable VLSI using X-net are reduced to 61% and 80%, respectively, in comparison with those of the multiple-valued bit-serial reconfigurable VLSI using an eight nearest-neighbor mesh network.

Acknowledgements

This work is supported by VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with STARC, e-Shuttle, Inc., Fujitus Ltd., Cadence Design Systems, Inc. and Synopsys, Inc.

References

[1] S. Vassiliadis and D. Soudris, ed., Fine- and coarse-grain reconfigurable computing, Springer, 2007.
[2] I. Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.26, no.2, pp.203–215, 2007.
[3] H.M. Munirul and M. Kameyama, "Architecture of a fine-grain field-programmable VLSI based on multiple-valued source-coupled logic," IEICE Trans. Electron., vol.E87-C, no.11, pp.1869–1875, Nov. 2004.
[4] 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.
[5] A. Ishikawa, N. Okada, and M. Kameyama, "Low-power multiple-valued reconfigurable VLSI based on superposition of bit-serial data and current-source control signals," Proc. 40th IEEE International Symposium on Multiple-Valued Logic, pp.179–184, 2010.
[6] X. Bai and M. Kameyama, "Current-source-sharing differential-pair circuits for a low-power fine-grain reconfigurable VLSI architecture," Proc. 42nd IEEE International Symposium on Multiple-Valued Logic, pp.208–213, 2012.
[7] X. Wang and L. Bandi, "X-network: An area-efficient and high-performance on-chip wormhole-switching network," 12th IEEE International Conference on High Performance Computing and Communications, pp.362–368, 2010.
[8] J. Nickolls, "The design of the maspar MP-1: a cost effective massively parallel computer," 35th IEEE Computer Society International Conference, pp.25–28, 1990.
[9] J.N. Kalamatianos and E. Manolakos, "Parallel computation of higher order moments on the MasPar-1 machine," International Conference on Acoustics, Speech and Signal Processing, vol.3, pp.1832–1835, 1995.
[10] Mouna Baklouti, Mohamed Abid, Philippe Marquet, and Jean Luc Dekeyser, "Study and integration of a parametric neighboring interconnection network in a massively parallel architecture on FPGA," International Conference on Computer Systems and Applications, pp.368–373, 2009.
[11] N. Ohsawa, M. Hariyama, and M. Kameyama, "High-performance field programmable VLSI processor based on a direct allocation of a control/data flow graph," Proc. IEEE Computer Society Annual Symposium on VLSI, pp.86–91, 2002.
[12] 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.
[13] T. Ike, T. Hanyu, and M. Kameyama, "Fully source-coupled logic based multiple-valued VLSI," Proc. 32nd IEEE International Symposium on Multiple-Valued Logic, pp.270–275, 2002.

Xu Bai received the B.E. degree in Communication Engineering from Southwest Jiaotong university, Chengdu, China in 2008 and M.S. degree in Information Sciences from Tohoku University, Sendai, Japan in 2011, respectively. He is currently working toward the Ph.D. in Graduate School of Information Sciences, Tohoku University. His research interests include reconfigurable computing, multiple-valued VLSI computing and MOS current-mode logic. He received the Best Paper Prize of IEEE Sendai Section Student Award in 2010. He is a student member of the IEEE.

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 Dean and Professor in the Graduate School of Information Sciences, Tohoku University. His general research interests are intelligent integrated systems for real-world applications and robotics, advanced VLSI architecture, and new-concept VLSI including multiple-valued VLSI computing. Dr. Kameyama 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. Dr. Kameyama is IEEE Fellow and IPSJ Fellow.