Low Cost PLD with High Speed Partial Reconfiguration

Naoki Hirakawa[1], Masanori Yoshihara[1], Masayuki Sato[2 3], Kazuya Tanigawa[1] and Tetsuo Hironaka[1]
[1]Graduated School of Information Sciences, Hiroshima City University,
3-4-1 Ozuka-Higashi, AsaMinami-Ku, Hiroshima, 731-3194, Japan
[2]Graduated School of Engineering, Tokyo Metropolitan University,
1-1 Minami-Osawa, Hachioji-shi,Tokyo, 192-0397, Japan
[3]TAIYO YUDEN CO.LTD
5607-2 NakaMuroda-machi, Takasaki-shi, Gunma, 370-3347, Japan
E-mail : [1]n-hirakawa@csys.ce.hiroshima-cu.ac.jp

Abstract: Recently, Field Programmable Gate Arrays (FPGAs) have been used for implementing various types of logic functions. But the conventional FPGAs have the following problems. The conventional FPGAs include switch matrixes for programmable connection, but the switch matrix occupies a significantly large area of the FPGA. As another problem, the configuration speed is slow because of the serial configuration method. To resolve these problems, we proposed MPLD as a new Programmable Logic Device (PLD) architecture which introduce MLUT instead of the conventional LUT. In MPLD each MLUT can be used as a combination logic, memory and switch matrix, on the demand of the mapped circuit. The merits of MPLD are the following. 1) MPLD can behave as both the reconfigurable device and the conventional parallel memory. 2) Implementing cost of MPLD is cheaper than the conventional FPGAs, because MPLD does not need switch matrixes as the conventional FPGA does. 3) The configuration speed is fast and partial configuration is easy because configuration method of MPLD is same as write access of the conventional parallel memory. In this paper, we present the MPLD architecture and its evaluation results of the prototype MPLD chip.

1. Introduction

Field Programmable Gate Arrays (FPGAs) have been used for implementing various types of logic functions. The FPGA is a flexible device, so we can configure various circuits for various applications. As problems of the conventional FPGAs, the switch matrix occupies a significantly large area, and the configuration speed is slow. To resolve these problems, we proposed MPLD as a new Programmable Logic Device (PLD) architecture, introducing MLUT instead of the conventional LUT and switch matrix[1][2]. MLUT realized the functions as logic, memory and switch matrix. Merits of MPLD are the following.

- Switch matrix is not necessary.
  - Implementation of MPLD requires less number of metal layers for wiring than the conventional FPGA because of absent of switch matrix.
  - Cost for implementing MPLD is cheaper than the conventional FPGA because of less number of metal layers.
- Configuration speed is fast because configuration method of MPLD is same as write access of the conventional parallel memory.
- Partial reconfiguration is easy and fast because the method of configuration is same as write access of the conventional parallel memory.
- MPLD can be reconfigured while acting as a logic circuit because MPLD is based om 2-port memory.
- MPLD can behave as the conventional parallel memory because of the structure of MPLD.

In this paper, we describe the structure and the behavior of MLUT which is a fundamental component of MPLD, and then present how MPLD is constructed by the MLUT. After that, we present the evaluation results of the prototype MPLD chip and then conclude and describe the future works.

2. MPLD Architecture

In this section, MPLD Architecture is described. Figure 1 shows basic structure of MPLD. In Figure 1, MLUT is the fundamental component which construct the MPLD. So, MLUT is required to have the following functions.

- LUT
- Switch Matrix
- Memory for Configuration

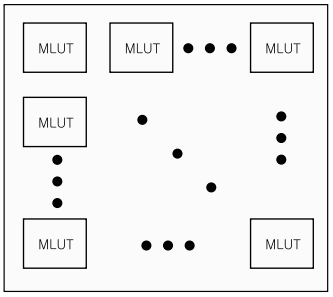

Figure 1. Basic Structure of MPLD

2.1. Structure of MLUT

Here, the structure of MLUT is explained. To meet the requirements, the basic design of MLUT is based on 2-port memory. Since 2-port memory is possible to read and write simultaneously, this means functions as LUT and memory can be realized simultaneously. Figure 2 shows the conventional 2-port memory. In Figure 2, let us assume that input LAD and output LDATA of port1 are used for logic, and input MAD and MDATA of port2 are used for memory access. The key idea to achieve the requirements of the MLUT is to pair LAD and LDATA as Figure 3. In Figure 3, each bit of LAD and

LDATA are grouped as Address Data pairs, such as a pair of LAD1 and LDATA1, a pair of LAD2 and LDATA2,⋯ and a pair of LAD*N* and LDATA*N*. Each Address Data pair works as a I/O port for the MLUT. Figure 4 is an example of MLUT with 4 Address Data pairs. Figure 5 shows the basic structure of MPLD which consists of MLUTs with 4 Address Data pairs. In Figure 5, each Address Data pair is connected to the MLUTs at the upper, lower, right and left respectively.

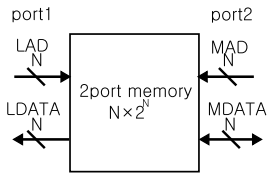

Figure 2. 2-port Memory

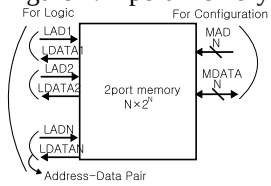

Figure 3. Address Data Pair

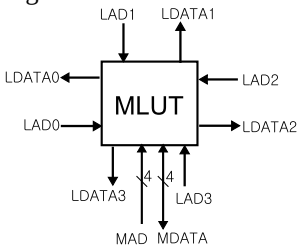

Figure 4. MLUT with 4 Address Data Pairs

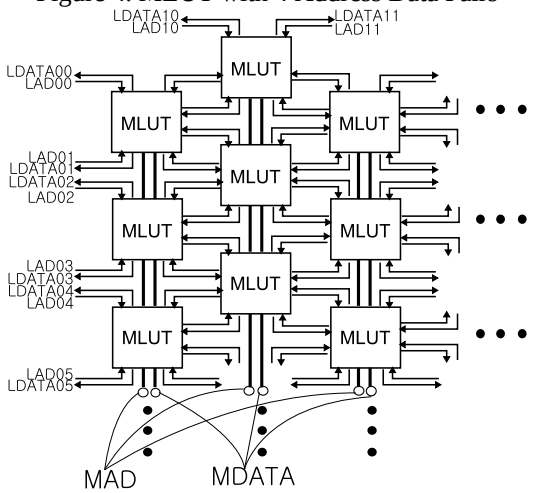

Figure 5. Basic Structure of MPLD

2.2. Behavior of MLUT

To explain the behavior of MLUT, the MLUT with 4 address data pairs shown in Figure 4 is used as an example.

- Configuration
  Figure 6 shows wires for configuration in MLUT. Since method of configuration is same as write access of the conventional parallel memory, MAD can be used to select the MLUT to reconfigure, with the data provided from MDATA in memory access. Configuration of the MLUT is provided as the truth table like the conventional LUT.

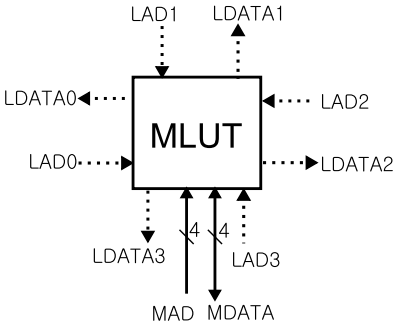

Figure 6. I/O to Configure MLUT

- Behavior as logic circuit
  Let us assume the MLUT behaves as a logic circuit like Figure 7. The logic circuit in Figure 7 is expressed as the truth table. The truth table of this logic circuit is shown in Table 1. In Table 1, each Input and Output are correspond to each bit of the address and data port of the conventional parallel memory. On configuration this truth table is written to the MLUT, then the MLUT works as the logic circuit as shown in Figure 7. Figure 8 shows the MLUT after configuration. In the Figure, LAD works as the input of the truth table, and LDATA works as the output of the truth table.

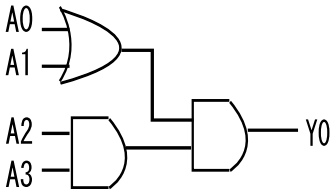

Figure 7. Logic Circuit

Table 1. Truth Table Written in MLUT for Logic Circuit

Input				Output			
LAD0	LAD1	LAD2	LAD3	LDATA0	LDATA1	LDATA2	LDATA3
0	0	0	0	0	*	0	0
0	0	0	1	0	*	*	*
0	0	1	0	0	*	*	*
0	0	1	1	0	*	*	*
0	1	0	0	0	*	*	*
0	1	0	1	0	*	*	*
0	1	1	0	0	*	*	*
0	1	1	1	0	*	*	*
1	0	0	0	0	*	*	*
1	0	0	1	0	*	*	*
1	0	1	0	0	*	*	*
1	0	1	1	1	*	*	*
1	1	0	0	0	*	*	*
1	1	0	1	0	*	*	*
1	1	1	0	0	*	*	*
1	1	1	1	1	*	*	*

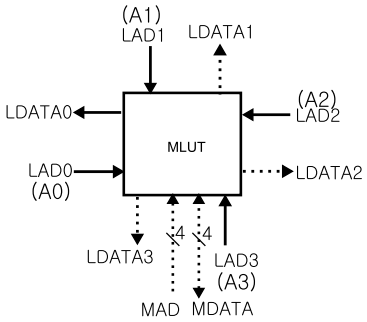

Figure 8. Behavior as Logic Circuit

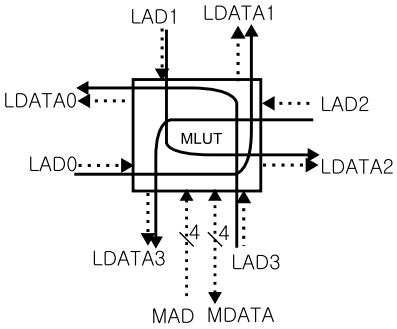

Figure 10. Behavior as Switch Matrix in MLUT

- Switch Matrix
  Assuming the MLUT behaves as the switch matrix like Figure 9, the switch matrix in Figure 9 is expressed as the truth table. The truth table of this switch matrix is shown in Table 2. In Table 2, Input LAD0, LAD1, LAD2 and LAD3 are corresponded to Output LDATA1, LDATA2, LDATA3 and LDATA0. Configuring this truth table into the MLUT, the MLUT works as shown in Figure 10.

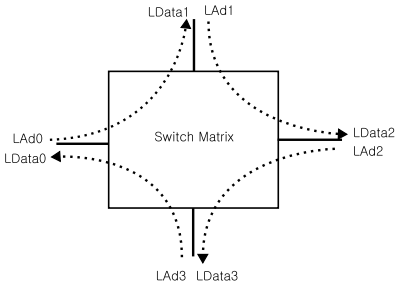

Figure 9. Switch Matrix

Table 2. Truth Table Written in MLUT for Switch Matrix

Input				Output			
LAD0	LAD1	LAD2	LAD3	LDATA0	LDATA1	LDATA2	LDATA3
0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0
0	0	1	0	0	0	0	1
0	0	1	1	1	0	0	1
0	1	0	0	0	0	1	0
0	1	0	1	1	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	1	0	1	1
1	0	0	0	0	1	0	0
1	0	0	1	1	1	0	0
1	0	1	0	0	1	0	1
1	0	1	1	1	1	0	1
1	1	0	0	0	1	1	0
1	1	0	1	1	1	1	0
1	1	1	0	0	1	1	1
1	1	1	1	1	1	1	1

3. Design of Prototype MPLD

To evaluate the MPLD architecture, prototype MPLD is designed. Figure 11 shows the structure of prototype MPLD. In Figure 11, prototype MPLD consists of the 16x4 MLUTs array, decoders for configuration and D-FFs for making MPLD possible to behave as the sequential circuit. Each MLUT which is neighboring the D-FF has its output connected to the D-FF, and also has its input connected to output of the D-FF, so the MLUT can feedback its output to itself. Each MLUT of the prototype MPLD has six address data pairs[2]. Figure 12 shows the MLUT with six address data pairs. In the Figure, SEL is 1bit input for single cycle full context switch and prototype MLUT can update 48bit of configuration data by one memory write access. So it needs sixteen times memory access to reconfigure the whole MLUT except for single cycle context. The six address data pairs of each MLUTs are connected to each other as seen in Figure 13 Figure 14 shows the structure of MLUT with six address data pairs. As seen in Figure 14, the MLUT with six address data pairs consists of six 2-port memory blocks, decoder for logic and MUX. Each memory block is implemented as a 16x8 bit 2-port memory. This means that the prototype MPLD has extra memory capacity for single cycle dynamic context switch. So, the capacity of this MLUT is twice of the single cycle dynamic context switch.

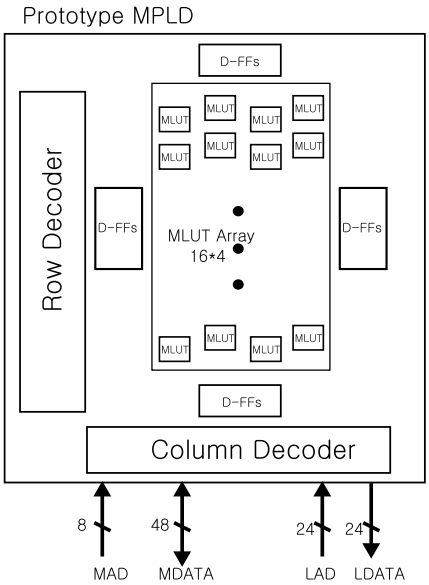

Figure 11. Structure of Prototype MPLD

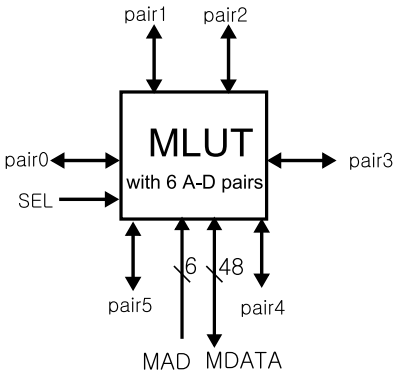

Figure 12. MLUT with 6 Address Data Pairs

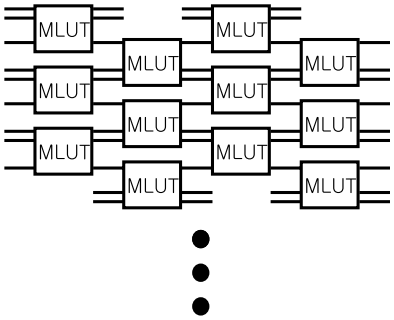

Figure 13. Connection of MLUTs with 6 Address Data Pairs

4. Evaluation

We implemented a prototype MPLD to confirm its function by using five metal layers ROHM 0.18μm CMOS technology, and confirmed its functions as memory and PLD by configuring it as a 32bit counter as an example application. Evaluation results are shown in Figure 15 and Table 3. As results, memory capacity of the prototype MPLD was 49152bit, and the core area was $1767.54 \times 1690.96 \mu m^2$ and the number of metal layers used for wiring was three. From evaluation results, latency of memory write access was 12.8nsec. This means that the configuration speed of MPLD is about 78.1 MHz because it depends on memory write access speed. Since configuration on each MLUT requires sixteen times of memory write accesses and prototype MPLD consists of 64 MLUTs, the achieved configuration time is about 6.6μsec for whole prototype MPLD.

Table 3. Evaluation Results

Behavior	Latency
Read	16.4nsec
Write	12.8nsec
32bit Counter	9.35nsec

5. Conclusions and Future Works

In this paper, we presented MPLD architecture as the low cost PLD with high speed partial reconfiguration. MPLD does not use switch matrix, which means it can be implemented with low cost. Since method of configuration is same as write access of the conventional parallel memory, configuration speed is faster than that of the conventional FPGA and on the same time fine grain partial reconfiguration is possible too.

In the future, we plan to improve the structure of MPLD from the results of prototype MPLD chip and also develop the compiler of the MPLD for practical use.

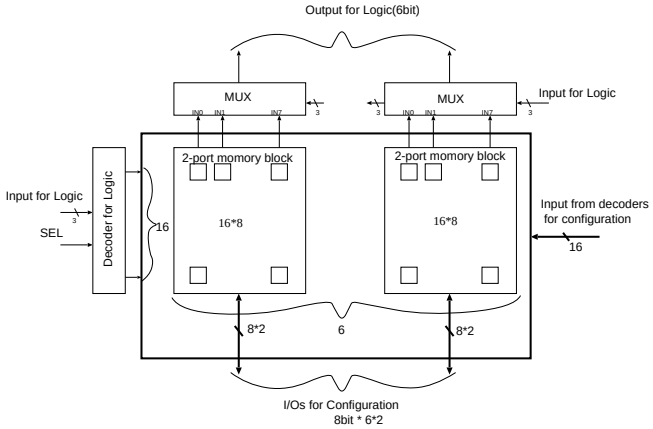

Figure 14. Structure of MLUT with 6 Address Data Pairs

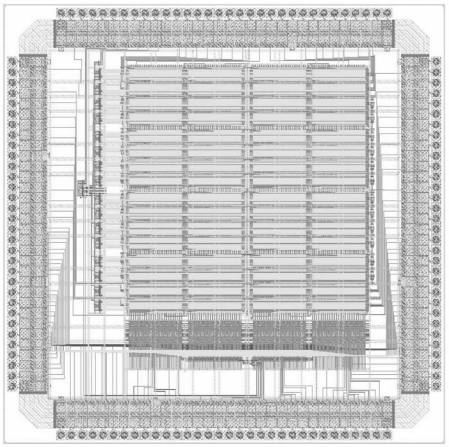

Figure 15. Prototype MPLD

Acknowledgement

The VLSI chip in this study has been fabricated in the chip fabrication program of VLSI Design and Education Center(VDEC), the University of Tokyo in collaboration with Rohm Corporation and Toppan Printing Corporation.

References

[1] Masayuki Sato and Hiroki Wakamatsu ”A logic design technique using SRAM blocks ” IEICE Technical Report RECONF2006-23 (in Japanese), pp.17-22, 2006.

[2] Masanori Yoshihara, Naoki Hirakawa, Kazuya Tanigawa, Tetsuo Hironaka and Masayuki Sato, ”Implementation of Memory(MPLD) with the Ability to Work as a Reconfigurable Device ”IEICE Technical Report RECONF2007-16 (in Japanese), pp.7-12, 2007.