We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

BOOK CITATION INDEX
CLARIVATE ANALYTICS
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

A Dynamically Reconfigurable Device

Minoru Watanabe
Shizuoka University,
Japan

1. Introduction

To the present day, the performance of microprocessors has progressed dramatically. Recently, almost all computer systems use reduced instruction set computer (RISC) architectures. However, about 30 years ago, complex instruction set computer (CISC) architectures were widely used for almost all computer systems. The advantages and successes of RISC architectures are attributable to their simplified structures.

Conventional complex instruction set computer (CISC) architectures invariably included various and numerous instruction sets. Each instruction was able to execute a complicated multi-step operation. For that reason, the CISC architectures were useful in assembler-based programming environments and in systems with small amounts of memory. However, such complicated architectures prevent increases in clock frequency or a processor's processing power.

Therefore, RISC architectures—which use simple architectures based on single-step instruction sets—have been developed. The RISC architectures present advantages in terms of higher clock frequency, smaller implementation area, and lower power consumption than conventional complex instruction set computer (CISC) architectures. Observation of many examples reveals that, in circuit implementations, a simple structure is best to increase the overall performance. That principle is also applicable to programmable devices.

If clock-by-clock reconfigurable devices are used, even a single instruction set computer (SISC) can be implemented onto them. A single instruction set computer is one in which a processor has only a single instruction. During production, various single instruction set computers are prepared: a single instruction set computer with an AND logic function, a single instruction set computer with an adder function, and so on. These processor units are implemented at necessary times and at necessary places of a programmable device. In CISC and RISC architectures, the hardware is fixed. Its operations are switched using software commands, as portrayed in Fig. 1(a). In contrast, in a single instruction set computer, the operation changes are executed by hardware reconfigurations, as shown in Figs. 1(b) and 1(c). Therefore, in a single instruction set computer, a processor with a certain function itself can be reconfigured to another processor with another function.

The implementation of such single instruction set computers provides the following advantages under programmable device implementations. A single instruction set computer with the simplest architecture can operate at the highest clock frequency among all processor architectures. In RISC architectures, many selectors to change functions are implemented; such selectors have a certain delay. However, single instruction set computers
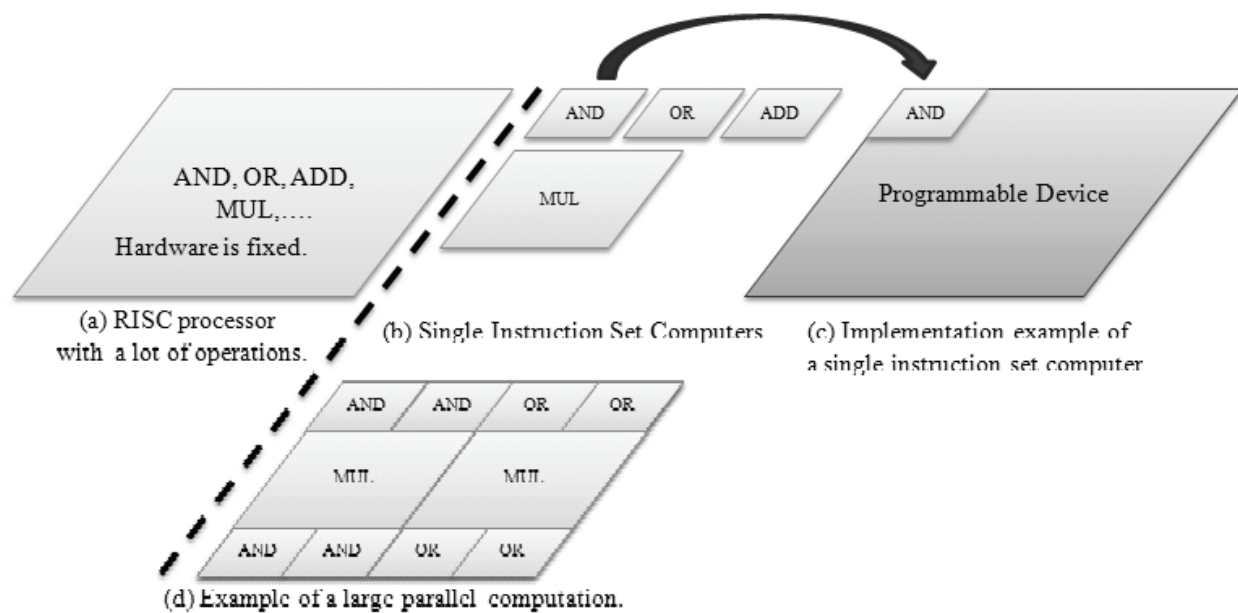
Fig. 1. RISC architecture and SISC architecture.

require no selector for use in function changes. Moreover, the inherent circuit complexity invariably increases the load capacitance and wiring capacitance at each circuit point. Large capacitance always decreases the maximum clock frequency. Therefore, the clock frequencies of simple architectures of single instruction set computers are higher than those of RISC and CISC architectures. As a result, the performance of single instruction set computers is superior to those of multi-instruction set computers.

Figure 1(d) shows that, since such a single instruction set computer can be implemented in a small area, large parallel computation can be achieved. Thereby, the total performance can be increased dramatically. However, to increase processing power using this concept, programmable devices must have a high-speed reconfiguration capability and a capability with numerous reconfiguration contexts to continue high-speed reconfigurations.

Currently, field programmable gate arrays (FPGAs) are widely used for many applications (1)–(3). Such FPGAs are always implemented with an external ROM. At power-on, a configuration context is downloaded from the external ROM to an internal configuration memory. However, such FPGAs have been shown to be unsuitable for dynamic reconfiguration applications because FPGAs require more than several milliseconds' reconfiguration time because of their serial transfer configuration mechanism.

On the other hand, high-speed reconfigurable devices have been developed, e.g. DRP chips (4). They include reconfiguration memories and a microprocessor array on a single chip. The internal reconfiguration memory stores the reconfiguration contexts of 16 banks, which can be substituted for one another during a clock cycle. Consequently, the arithmetic logic unit can be reconfigured on every clock cycle in a few nanoseconds. Unfortunately, increasing the internal reconfiguration memory while maintaining the number of processors is extremely difficult.

As with other rapidly reconfigurable devices, optically reconfigurable gate arrays (ORGAs) have been developed, which combine a holographic memory and an optically programmable gate array VLSI, as portrayed in Figs. 2 (5)–(9). Many configuration contexts can be stored in a holographic memory. Thereafter, they can be read out optically and programmed optically onto a gate array VLSI using photodiodes perfectly in parallel.

Therefore, high-speed configuration is possible in addition to numerous reconfiguration contexts. Such ORGA architectures present the possibility of opening the implementations of single instruction set computers.

This chapter introduces a VLSI design of an ORGA architecture: a dynamic ORGA architecture suitable for implementations of single instruction set computers.
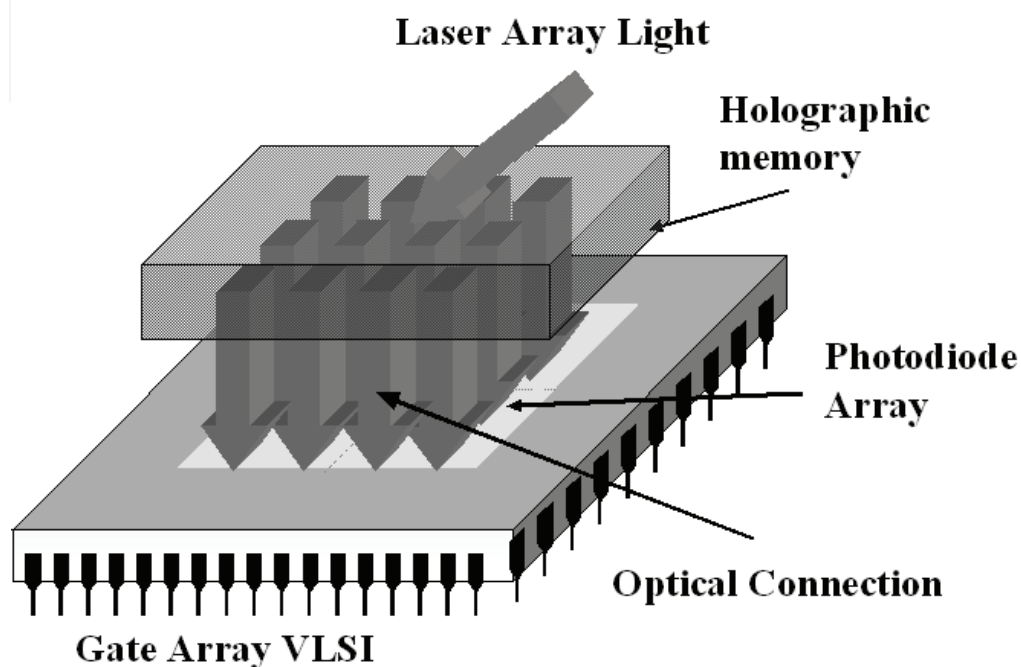
2. ORGA architecture

Fig. 2. Overall construction of an optically reconfigurable gate array (ORGA).

An overview of an Optically Reconfigurable Gate Array (ORGA) is portrayed in Fig. 2. An ORGA comprises a gate-array VLSI (ORGA-VLSI), a holographic memory, and a laser diode array. The holographic memory stores reconfiguration contexts. A laser array is mounted on the top of the holographicmemory for use in addressing the reconfiguration contexts in the holographic memory. One laser corresponds to a configuration context. Turning one laser on, the laser beam propagates into a certain corresponding area on the holographic memory at a certain angle so that the holographicmemory generates a certain diffraction pattern. A photodiode-array of a programmable gate array on an ORGA-VLSI can receive it as a reconfiguration context. Then, the ORGA-VLSI functions as the circuit of the configuration context. The reconfiguration time of such an ORGA architecture reaches nanosecond-order (5),(6). Therefore, very-high-speed context switching is possible. In addition to it, since the storage capacity of a holographicmemory is extremely high, numerous configuration contexts can be stored in a holographic memory. Therefore, the ORGA architecture can dynamically implement single instruction set computers.

3. Dynamic ORGA architecture

A configuration context is optically applied in ORGAs. In ORGA-VLSIs, a certain detection circuit must be used in addition to a programmable gate array. The detection circuit is called

an optical reconfiguration circuit. Such an optical reconfiguration circuit is connected to each programming point of a programmable gate array. Therefore, the number of reconfiguration circuits can be as large as those of FPGAs. The resultant reduction of the implementation area of optical reconfiguration circuits is extremely important in ORGAs.

In major ORGAs (5),(6), each optical reconfiguration circuit consists of a photodiode, a refresh transistor, and a single-bit static configuration memory, as portrayed on the left side of Fig. 3. A reconfiguration procedure is initiated by charging the junction capacitance of the photodiode using refresh transistors. After charging, an optical configuration context is provided from a holographic memory and is received on the photodiodes. The electric charge in the junction capacitance of each light-received photodiode is discharged and the electric charge in the junction capacitance of each photodiode receiving no light is retained. The resultant difference is detectable by sensing the voltage between the anode and cathode of the photodiode. The sensed information is temporarily stored on a single-bit static configuration memory. Then, the context information is provided to each programming point of a gate array. Using this technique, a configuration context can be retained indefinitely in the ORGA-VLSI so that the state of the gate array can be maintained statically.
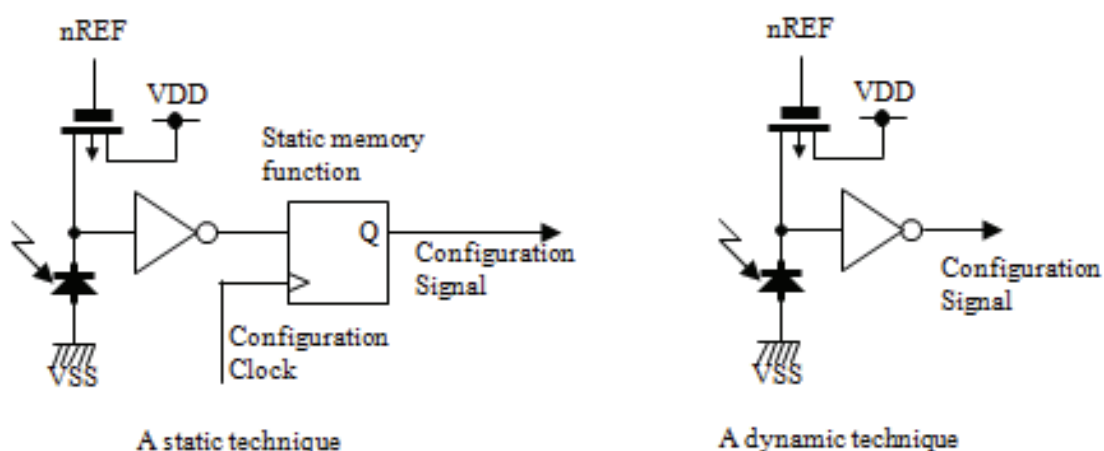

Fig. 3. Optical reconfiguration circuits of static and dynamic techniques.

However, the static configuration memory prevents realization of high gate count ORGA-VLSIs. The static configuration memory comes to occupy about 25% of the area of an entire VLSI chip. Moreover, using the memory function for storage during an indefinite period can be considered as over-capacity for implementation in single instruction set computers because a processor of a single instruction set computer is dynamically reconfigured. For that reason, its lifetime is very short. In addition, the configuration information is stored on a holographic memory; the information can therefore be read out anytime. Because of that feature, even when long-term functions are required, a certain refresh cycle enables such function implementations. Therefore, a Dynamic Optically Reconfigurable Gate Array (Dynamic ORGA) architecture without a long-term storable configuration memory was proposed (7). A photodiode invariably has junction capacitance. Therefore, the junction capacitance can maintain the state of a gate array for a certain time. The dynamic ORGA perfectly removes the static configuration memory to store a context and uses the junction capacitance of photodiodes as dynamic configuration memory, as shown on the right side of
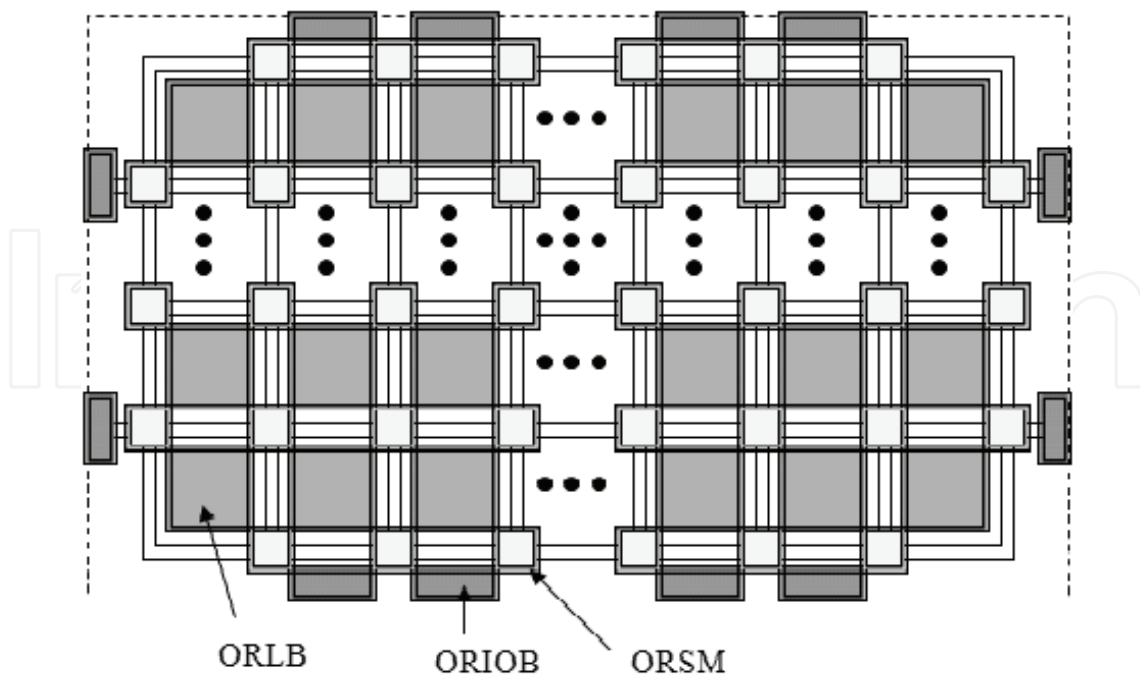
Fig. 4. An island-style gate array constructed by optically reconfigurable logic blocks (ORLBs), optically reconfigurable switching matrices (ORSMs), and optically reconfigurable I/O blocks (ORIOBs).

Fig. 3. Following such a concept of single instruction set computers, the junction capacitance of photodiodes is sufficient to retain the state of a gate array. This architecture is called a dynamic ORGA architecture. The dynamic ORGA architecture is a very advanced ORGA architecture in terms of gate density in ORGAs.

4. VLSI design with 51,272 gates

This section presents a description of the design of a 51,272 gate DORGA-VLSI. The 51,272-gate-count DORGA-VLSI chip was designed using a 0.35 μm standard complementarymetal oxide semiconductor (CMOS) process. The basic functionality of the DORGA-VLSI is fundamentally identical to that of currently available field programmable gate arrays (FPGAs). The DORGA-VLSI takes an island-style gate array or a fine-grain gate array.

4.1 Photodoide cell design

Always, the depletion depth of a photodiode between an N-well and a P-substrate is deeper than that of a photodiode between an N-diffusion and a P-substrate. However, the minimum size of a photodiode between an N-well and a P-substrate is always larger than that of a photodiode between an N-diffusion and a P-substrate. Since an ORGA requires many photodiodes, the implementation area reduction is very important. For that reason, photodiodes were constructed between the N-diffusion and the P-substrate. The acceptance surface size of the photodiode is $8.8 \times 9.5 \ \mu m^2$. In addition, the photodiode cell size is $21.0 \times 16.5 \ \mu m^2$. Such a cell was designed as a full custom design. The fourth metal layer is used for guarding transistors from light irradiation; the other three layers were used for wiring.

Technology	0.35 μm double-poly four-metal CMOS process
Chip size [mm^2]	14.2 ×14.2
Supply Voltage [V]	Core 3.3 , I/O 3.3
Photodiode size [μm^2]	9.5 ×8.8
Horizontal distance between photodiodes [μm]	28.5-42
Vertical distance between photodiodes [μm]	12-21
Number of photodiodes	170,165
Number of logic blocks	1,508
Number of switching matrices	1,589
Number of I/O bits	272
Gate count	51,272

Table 1. Specifications of a high-density DORGA.

4.2 Optically reconfigurable logic block

A block diagram of an optically reconfigurable logic block of the DORGA-VLSI chip is presented in Fig. 5. Each optically reconfigurable logic block consists of 2 four-input one-output look-up tables (LUTs), 10 multiplexers, 8 tri-state buffers, and 2 delay-type flip-flops
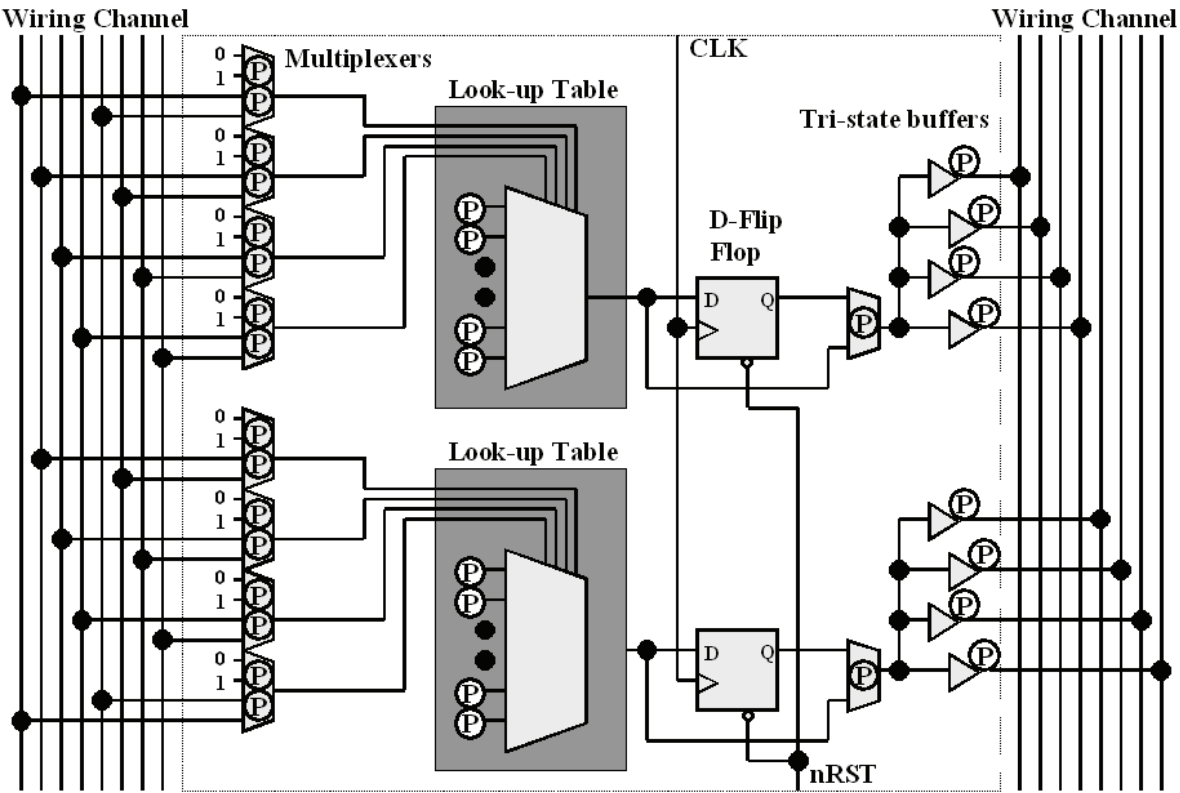
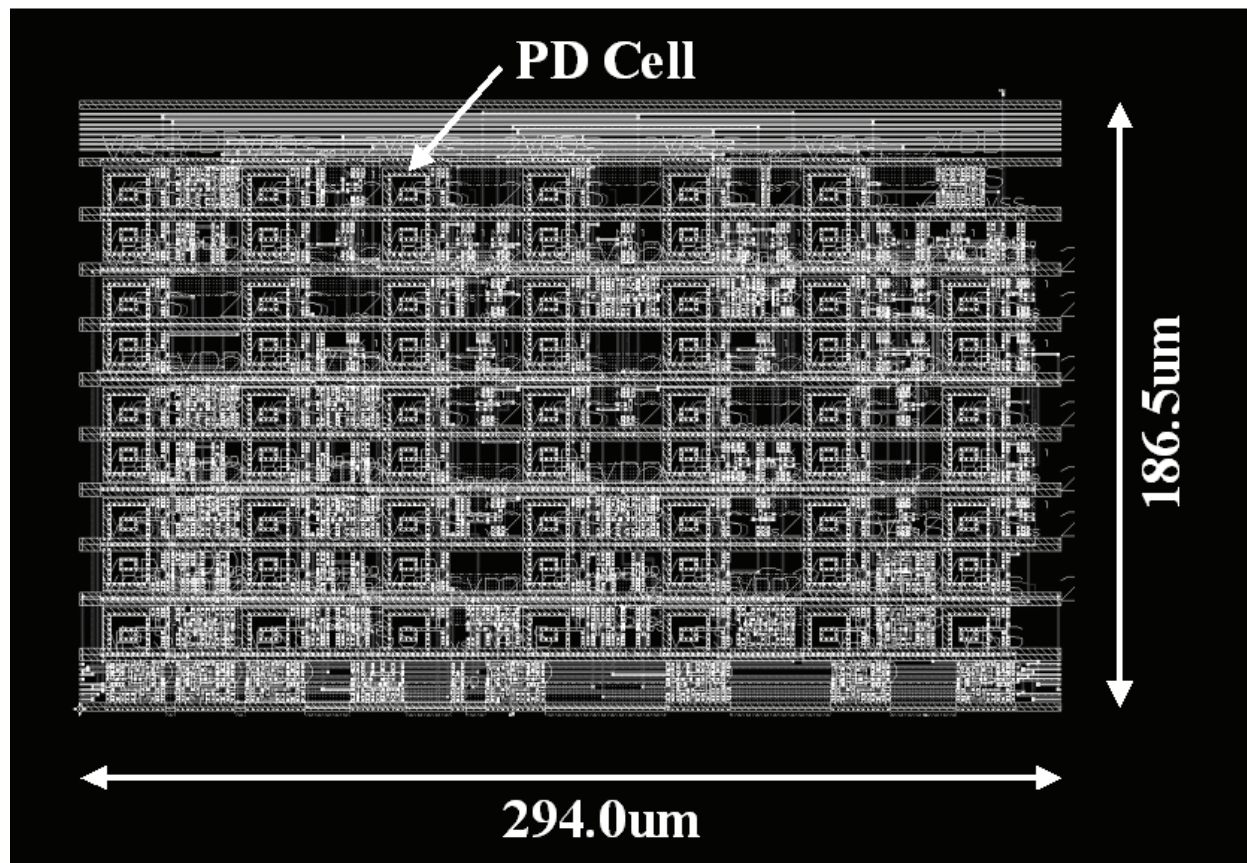Fig. 5. Block diagram of an optically reconfigurable logic block (ORLB).

Fig. 6. CAD layout of the optically reconfigurable logic block (ORLB).

with a reset function. The input signals from the wiring channel, which are applied through some switching matrices and wiring channels from optically reconfigurable I/O blocks, are transferred to LUTs through eight multiplexers. The LUTs are used for implementing Boolean functions. The outputs of an LUT and of a delay-type flip-flop connected to the LUT are connected to a multiplexer. A combinational circuit and sequential circuit can be chosen by changing the multiplexer, as in FPGAs. Finally, outputs of the multiplexers are connected to the wiring channel again through eight tri-state buffers. As a result, each four-input one-output LUT, multiplexer, and tri-state buffer has 16 photodiodes, 2 photodiodes, and 1 photodiode, respectively. In all, 58 photodiodes are used for programming an optically reconfigurable logic block. The optically reconfigurable logic block can be reconfigured perfectly in parallel. The CAD layout is depicted in Fig. 6. This is a standard-cell based design. The cell size is 294.0 × 186.5 μm^2. Wiring between cells was executed using the first to the third metal layers while avoiding the aperture area of the photodiode cell. Such optically reconfigurable logic block design is based on a standard cell design, except for custom designs of transmission gate cells and photodiode cells. Each photodiode is arranged at 42.0 μm horizontal intervals and at 12.0-21.0 μm vertical intervals.

4.3 Optically reconfigurable switching matrix
Similarly, optically reconfigurable switching matrices are optically reconfigurable. The block diagram of the optically reconfigurable switching matrix is portrayed in Fig. 7. Its basic construction is the same as that used by Xilinx Inc. Four-directional switching matrices with 48 transmission gates were implemented in the gate array. Each transmission gate can be
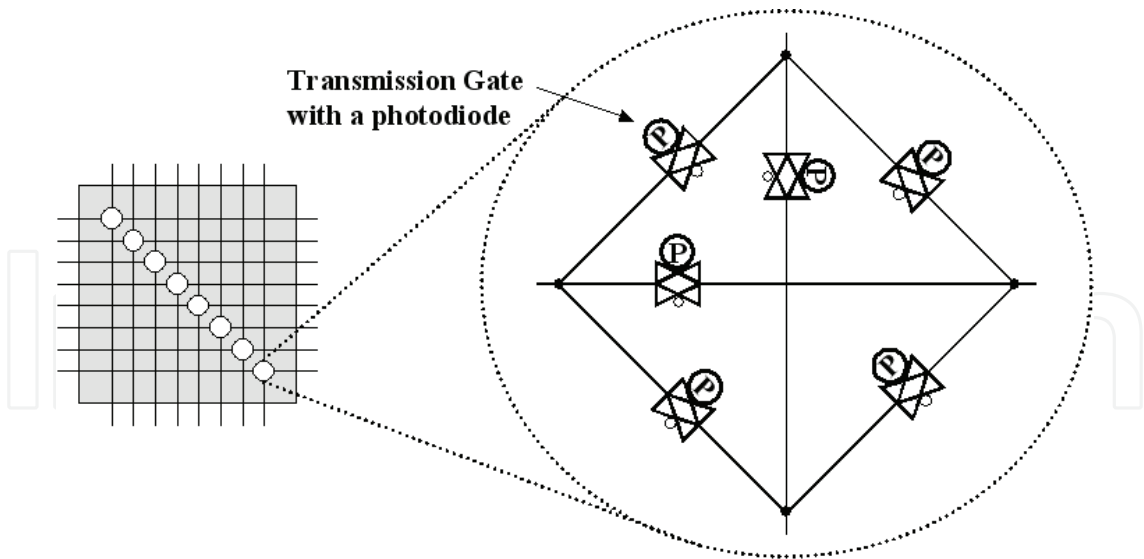
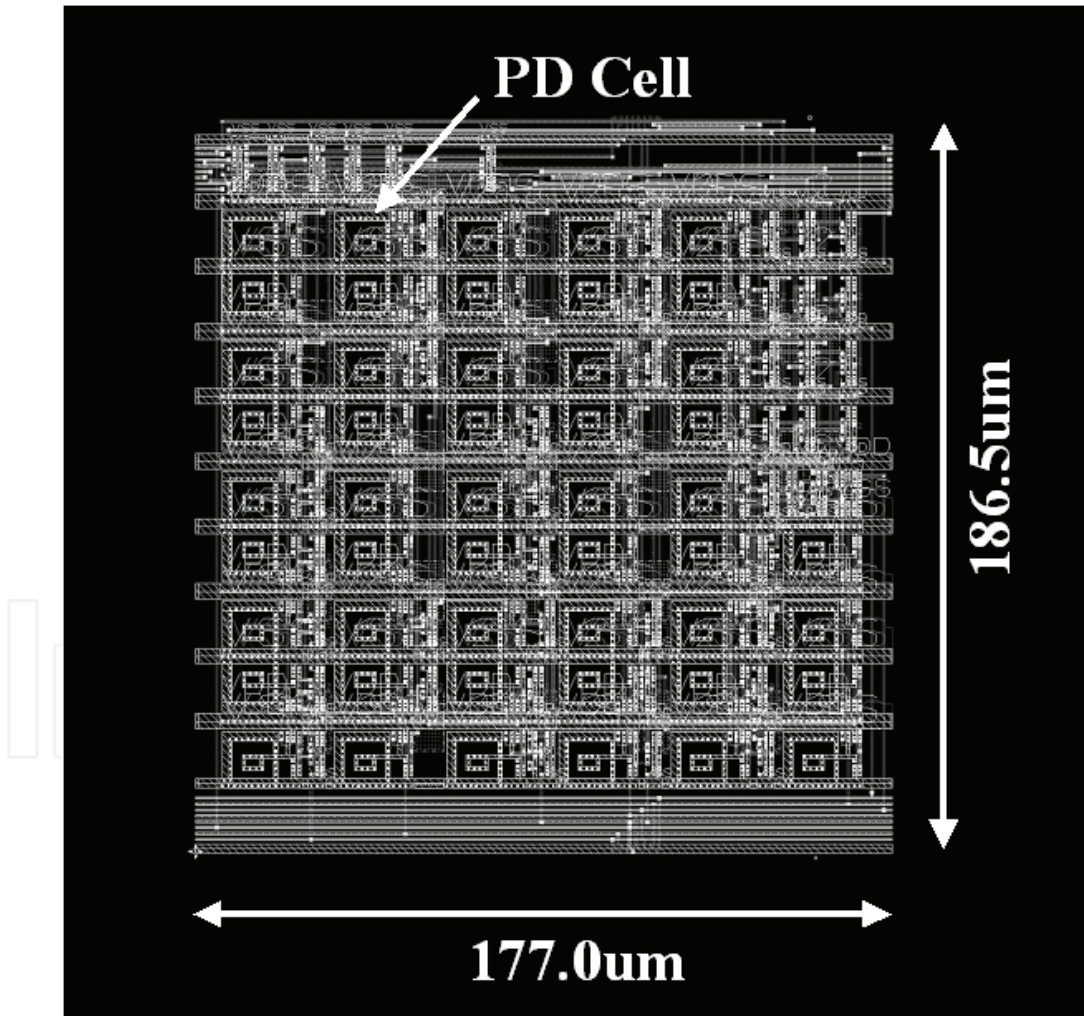Fig. 7. Block diagram of an ORSM.

Fig. 8. CAD layout of an ORSM.

considered as a bi-directional switch. A photodiode is connected to each transmission gate; it controls whether the transmission gate is closed or not. Based on that capability, four-

direction switching matrices can be programmed as 48 optical connections. The CAD layout is portrayed in Fig. 8. The cell size is 177.0 × 186.5 μm^2. As with the ORLBs, wiring was executed using the first to the third metal layers, thereby avoiding the aperture area of the photodiode cell. Such an optically reconfigurable switching matrix was designed using custom cells of photodiode cells and transmission gate cells, except for some buffers. Each photodiode is arranged at 28.5 μm horizontal intervals and at 12.0-21.0 μm vertical intervals.

4.4 Gate array

Figure 4 depicts the gate array structure. Table 1 presents its specifications. The gate array was designed using the Design Compiler logic synthesis tool and the Apollo place and route tool (Synopsys Inc.). The ORGA-VLSI chip consists of 1,508 optically reconfigurable logic blocks (ORLB), 1,589 optically reconfigurable switching matrices (ORSM), and 272 optically reconfigurable I/O bits (ORIOB). Each optically reconfigurable logic block is surrounded by wiring channels. In this chip, one wiring channel has eight connections. Switching matrices are located on the corners of optically reconfigurable logic blocks. Each connection of the switching matrices is connected to a wiring channel.

The accepted surface size of the photodiode and photodiode-cell size, including an optical reconfiguration circuit are, respectively, 8.8 × 9.5 μm^2 and 21.0 × 16.5 μm^2. The photodiode cells were arranged at 28.5-42.0 μm horizontal intervals and at 12.0–21.0 μm vertical intervals: in all, 170,165 photodiodes were used. The fourth metal layer is used for guarding transistors from light irradiation; the other three layers were used for wiring.

4.5 Reconfiguration performance

The retention time and configuration time of photodiode memory architecture in a DORGA-VLSI were estimated experimentally using another DORGA-VLSI chip. That other VLSI chip was fabricated using the same CMOS process. In addition, the VLSI chip has identical photodiode construction and characteristics. Therefore, although a 51,272 DORGA-VLSI chip has never been fabricated, its characteristics were measured using the other DORGA-VLSI chip. As a result, the retention time of photodiode was measured as longer than 45 s. That retention time is much longer than that of current DRAMs. Consequently, the storage time is sufficient for the implementation of single instruction set computers. Additionally, the product of the photodiode response time and laser power for each photodiode was measured as $T_{reconfiguration} \cdot P_{laser} = 12.7 \ pJ$. That measurement demonstrates that nanosecond-order configuration is possible.

5. Conclusion

This chapter has introduced and explained important concepts related to single instruction set computers. Such single-instruction set computers constitute an acceleration method used with microprocessor operations. To implement them, clock-by-clock dynamically reconfigurable devices are desired. However, using current VLSI technologies, simultaneous realization of fast reconfiguration and numerous reconfiguration contexts is impossible. To realize such clock-by-clock dynamically reconfigurable devices, another technology must be developed. As one possibility, this chapter has introduced and described an optically reconfigurable gate array VLSI. Currently, the gate count and performance of such ORGA-VLSIs are insufficient. Nevertheless, such architecture presents the possibility of overcoming

current VLSI limitations. Realizing a device to overcome those limitations remains as a subject for future works.

6. References

[1] Altera Corporation, "Altera Devices," http://www. altera.com.

[2] Xilinx Inc., "Xilinx Product Data Sheets," http://www. xilinx.com.

[3] Lattice Semiconductor Corporation, "LatticeECP and EC Family Data Sheet," http://www. latticesemi.co.jp/products, 2005.

[4] H. Nakano, T. Shindo, T. Kazami, M. Motomura, "Development of dynamically reconfigurable processor LSI," NEC Tech. J. (Japan), vol. 56, no. 4, pp. 99–102, 2003.

[5] M. Miyano, M. Watanabe, F. Kobayashi, "Optically Differential Reconfigurable Gate Array," Electronics and Computers in Japan, Part II, Issue 11, vol. 90, pp. 132-139, 2007.

[6] M. Nakajima, M. Watanabe, "A four-context optically differential reconfigurable gate array," IEEE/OSA Journal of Lightwave Technology, Vol. 27, No. 24, 2009.

[7] M. Watanabe, F. Kobayashi, "Dynamic Optically Reconfigurable Gate Array," Japanese Journal of Applied Physics, Vol. 45, No. 4B, pp. 3510-3515, 2006.

[8] D. Seto, M. Watanabe, "A dynamic optically reconfigurable gate array – perfect emulation," IEEE Journal of Quantum Electronics, Vol. 44, Issue 5, pp. 493-500, 2008.

[9] D. Seto, M. Watanabe, "An 11,424 Gate-Count Dynamic Optically Reconfigurable Gate Array with a Photodiode Memory Architecture," Asia and South Pacific Design Automation Conference, pp. 117-118, 2009.
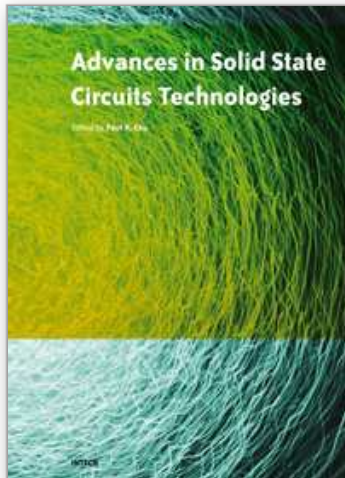
Advances in Solid State Circuit Technologies
Edited by Paul K Chu

This book brings together contributions from experts in the fields to describe the current status of important topics in solid-state circuit technologies. It consists of 20 chapters which are grouped under the following categories: general information, circuits and devices, materials, and characterization techniques. These chapters have been written by renowned experts in the respective fields making this book valuable to the integrated circuits and materials science communities. It is intended for a diverse readership including electrical engineers and material scientists in the industry and academic institutions. Readers will be able to familiarize themselves with the latest technologies in the various fields.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds