Development of Reconfiguration unit for General Purpose Microcomputer with Dynamic Partial Reconfiguration

Sota Kono, Gen Nishizawa, and Akira Yamawaki

Kyushu Institute of Technology, Kitakyushu, 804-8550 Japan
Second affiliation, Address, City and Postcode, Country

Corresponding Author: yama@ecs.kyutech.ac.jp

Abstract

Microcomputers are used in many kinds of the embedded devices. Developers of the microcomputer need to meet the requests from the makers of the embedded products. Tailoring to each application leads to the higher development cost. However the general purpose microcomputer so as to include all functions is not feasible. This paper proposes a GEDY (General purpose micro-computer with dynamic partial reconfiguration). It is possible to switch the circuit as needed by using the DPR (Dynamic Partial Reconfiguration). Also, it is possible to add a new circuit after shipment without stopping the entire system. In this paper, we develop the reconfigurable parts in the GEDY and verify its operation. In the experiment, we prepare the LED lighting circuit and a serial communication circuit. By switching them dynamically, we verify the functionality of the reconfigurable parts. The result of the FPGA implementation shows that the circuit scale of GEDY is small. This facts indicates that the most of FPGA resources can be used for the reconfigurable parts.

Keyword: DPR, reconfigurable parts, FPGA

1. Introduction

Microcomputers (MCUs) are used in many kinds of the embedded devices. Across many applications, some MCU has limited peripherals specialized for the applications. Some equip with various peripheral devices for the general purpose usages. Developers of the microcomputer need to meet the requests from the makers of the embedded products. Tailoring to each application leads to the higher development cost. But the general purpose microcomputer so as to include

all functions is not feasible. These problems can be solved by the use of a dynamic partial reconfiguration (DPR)⁽¹⁾ on a microcomputer. Because any circuits can be swapped on the single unified MCU. Also, a new circuit can be installed after shipment without stopping the entire system like software program.

Researchers have studied about the DPR⁽²⁻⁴⁾. But, in many cases, DPR is used for data processing circuit. That is, dynamically switching the peripheral circuit (UART, SPI, PWM, Ethernet, Timer etc...) in reconfigurable part is not considered. Also, demonstration experiment has not been about the feasibility of function by switching of the peripheral circuit. Further, it has not been evaluated impact on performance. Therefore, in this paper, we propose a general purpose micro-computer with dynamic partial reconfiguration, GEDY. Also, we shows a design method of DPR in processor-based FPGA. Then, we develop the reconfigurable part in the GEDY and verify its operation.

2. Overview of GEDY

2.1 Structure of GEDY

Figure 1 shows an overview of the GEDY organization

Figure 1. GEDY Organization

The GEDY consists of CPU, RCP, MMR, RP, and OFCM as shown in Figure 1

The CPU is a processor to manage the whole of the MCU. It also handles the dynamic partial reconfiguration on the reconfigurable parts (RPs). The CPU reconfigures the dedicated RP by storing the circuit configuration bitstream into the reconfiguration port (RCP).

The memory mapped register (MMR) is a characteristic component of the GEDY. The MMR decouples the on-chip interconnect and the reconfigurable circuit reconfigured on a RP by the unified memory mapped register interface. Thus, the circuit designer have only to consider the simple memory mapped registers when designing the interface to the CPU. From the viewpoint of the CPU, the CPU has only to access the memory mapped registers to manage the RP similar to the conventional I/O peripherals. In addition, the memory map for each hardware to be configured on a RP are remapped according to the hardware swapping. Thus, the CPU can uniformly access the hardware through the fixed memory map even if many hardware modules are dynamically switched on the different RPs.

The mailbox (MB) is a register file to communicate any data between the reconfigurable part and the CPU. The mailbox hides the complexity of the interconnect among the RP and CPU by the simple readable/writable register file. The CPU has only to load/store the data to the MB to manage the hardware module on the RP. The hardware module has only to read/write the dedicated registers in the MB to execute following the request of the CPU. The base address of the MB can be remapped dynamically on the MMR according to the hardware module to be reconfigured into the RP as mentioned above.

The off-chip memory (OFCM) holds many bitstreams to be configured on the RPs. The other useful data can be stored into the OFCM.

2.2 Flow of Reconfiguration

Figure 2 shows the Flow of Reconfiguration. In the flow of (1), The CPU writes address of the peripheral circuit mailbox to address of the RP. In the flow of the (2), The CPU writes the circuit data from OFCM to RCP. In the flow of (3), peripheral circuit is configured in the RP. In the flow of (4), The CPU writes necessary parameters to the MB. Also, The CPU operates the peripheral circuit by sets the boot flag on

Figure 2. Flow of Reconfiguration

the MB. In the flow of (5), The CPU knows the operation completion of the peripheral circuit by seeing the completion flag of the MB. Then, The CPU reads the output results from the MB or OFCM

3. Design Flow for GEDY

3.1 Design Entry

In the GEDY, the partial reconfiguration module has to be resigned following the design method for an FPGA supporting DPR. This section summarizes the design method of the hardware modules supporting DPR on the GEDY.

The fixed part (FP) is provided as the top module in hardware description language (HDL). The partial reconfigurable modules (PRMs) are described as the components in the RPs embedded to the fixed part in HDL. The fixed part and the partial reconfigurable modules are converted to the netlists dedicated to the physical GEDY.

3.2 Floor Planning

Floor planning is to place the netlists onto the physical GEDY. A master floor plan consisting of the fixed part and the RPs is provided. The master floor plan is dedicated to the physical GEDY. In the master floor plan, the fixed part and the RPs are previously deployed to certain areas physically. The user has only to assign the netlists of the PRMs to the physicals RPs.

3.3 Generation of Circuit Data

After the floor planning, the bitstreams configuring the fixed part and the RPs are generated. The full bitstream containing the fixed part is configuration data of the entire GEDY. The partial bitstream is configuration data of each RP.

Figure 3. Image of the Floor Planning

The data is stored in an external storage (DDR memory, flash memory, network, etc...).

After the power is turned on, the GEDY configures the initial system by the full bitstream stored in an external storage. During system execution, GEDY loads the circuit data in accordance with the management program, and then reconfigures RP by writing the partial bitstreams to RCP.

4. Prototype of GEDY, GEDY-1

4.1 Overview of Prototype

To verify the functionality of the reconfigurable part, we have developed the prototype hardware of GEDY, GEDY-1. Figure 4 shows an overview of the GEDY-1.

Figure 4. Prototype Organization

The GEDY-1 consists of CPU, RCP, two RPs, OFCM, LED, and I/O port corresponding to the components as shown in Figure 4.

We have prepared the three types of circuit data into the OFCM. The first circuit data for the LED driving circuit. The second circuit data for the serial communication circuit. The third circuit data is the empty PR.

We assigned the identify numbers for three circuits tagging with the RP number. By sending the ID from the PC to the

GEDY-1, the GEDY-1 can configure the specified circuit into the corresponding RP.

4.2 Implementation of Prototype

In this paper, we design a dynamic partial reconfiguration hardware by using the Vivado2015.2 Xilinx, Inc. Also, an implementation of the circuit to the FPGA (XC7Z010-1CLG-400) on the Digilent zybo board.

Table 1 shows the utilization of resources in the FP. In the FP, utilization of the flip-flop (FF) is the only 4 percent. Also, utilization of the look up table (LUT) is the only 5percent. In

Table 1. Utilization of Resources in the FP

	Available	Used	Utilization
FF(個)	35200	1475	4.19%
LUT(個)	17600	884	5.02%

Table 2. Correspondence of the Number and Circuit

Number	Circuit
1	RM0_blank
2	RM0_LED
3	RM0_UART
4	RM1_blank
5	RM1_LED
6	RM1_UART

other words, we can assign most of FPGA resources to the RPs. The achievable clock frequency of the FP is about 145MHz. This operation frequency is good performance on the FPGA used. By the other words, the FP may not negatively effect the total performance of the GEDY-1 including several partially reconfigurable modules.

4.3 Operation verification of Prototype

First, we saved each circuit data of the RPMs to the DDR3 SDRAM. Then, when transmitting the number from the PC, we confirmed the switching of the circuit corresponding to the number.

Table 2 shows correspondence of the number and the circuit. Figure 5 shows the result when configured the LED lighting circuit. Figure 6 shows the result when configured the serial communication circuit.

We can confirm the lighting LED from figure 5. Also, we can confirm the serial communicating with the PC. In this experiment, serial communication is directed to echo back circuit that directly send the ones received.

5. Conclusion

We have proposed a general purpose micro-computer with dynamic partial reconfiguration (GEDY). Then, we implemented the reconfigurable parts on the FPGA (XC7Z010-1CLG-400). As a result of the operation verification, we confirmed that the reconfigurable parts was switched correctly. Also, Utilization of the source was only 4% in the flip-flop and 5% in the look up table. Therefore, we can assign most of FPGA resources to the reconfigurable parts.

In future work, we will development of the memory mapped register. Thereby, The CPU can uniformly access the

Figure 5. Result when Configured the LED Lighting Circuit

Figure 6. Result when Configured the Serial Communication Circuit

hardware through the fixed memory map even if hardware modules are dynamically switched on the different reconfigurable parts.

References

- (1) M.Taher and T.El-Ghazawi : "Virtual Configuration Management: A Technique for Partial Runtime Reconfiguration", IEEE Transactions on Computers, vol.58, No.10, pp.1398-1410, 2009
- (2) M.Majer and J.Teich : "The Erlangen Slot Machine: A Dynamically Reconfigurable FPGA-based Computer", Journal of VLSI Signal Pro-cessing, vol.47, No.1, pp. 15-31,2007
- (3) Yosuke Shiga, Haruki Imanaka, Takahumi Ueno : "PCI interface technology by FPGA of dynamic reconfigurable Hw/Sw mixed system", Information Processing Society of Japan Vol.2002, No.5, pp. 85-90, 2002
- (4) Shuhei Igari, Junji Kitamichi, Yuichi Okuyama : "Implementation of Multi-accelerator type dynamically reconfigurable processor", The Institute of Electronics Information and Communication Engineers, Vol.110, No.360, pp.163-168, 2011