Department of Electrical and Electronic Information Engineering	ID	M185201	Supervisor	Shuichi Ichikawa
Name	Seiya Ogido			Naoki Fujieda

Abstract

Title	Implementing Fault Tolerance Method Using Dynamic Partial Reconfiguration by Xilinx Zynq-7000 SoC

In the area of fault-tolerant systems, such as in the space industry, area redundancy techniques are often employed to sustain the operation of logic circuits. The area redundancy technology requires three or more equivalent units, which involves large area overhead. Thus, a fault-tolerant systems with reconfigurable devices have been studied in recent years.

Ogido et al. have proposed a fault tolerant system that adopts FPGA (Field Programmable Gate Array) with dynamic partial reconfiguration, where autonomous control methods were presented. They adopted FSBL (First Stage Boot Loader) to write the circuit design to the FPGA, which was not practical because the system has to be restarted whenever the circuit design is written.

This study examines the reconfiguration method from the processor of an SoC (System on Chip), which consists of a hard-core processor and an FPGA fabric. For dynamic partial reconfiguration, two methods are introduced; i.e., bare metal application and Linux OS. In case of the bare metal application, the system was implemented with the PRC provided by Xilinx on the evaluation board. In case of Linux OS, Linux is executed on ARM processor on the evaluation board to control dynamic partial reconfiguration via PCAP, which is the dedicated interface for the hard-core processor. In this process, we designed the hardware with multiple reconfigurable areas on the FPGA, and the software for reconfiguration with the corresponding device driver. The cyclic operation among the reconfigurable regions of the logic circuit were realized, which was proposed in the previous study. To reproduce the cyclic operation, the abstraction method was developed to access to the circuits moving in the reconfigurable area; the software firstly refers to a common database to retrieve the port addresses which are properly maintained when the circuits are moved in reconfigurable areas. As a result, it was confirmed that the dynamic partial reconfiguration can be controlled with the Linux OS. To realize the fault-tolerant system, we implemented the synchronization method using flags among the fault-tolerant circuits that operate asynchronously. The derived fault-tolerant system over multiple reconfigurable areas successfully operated. By controlling the fault-tolerant operation from the hard-core processor, the intermediate results are properly saved to restart the computation after reconfiguration.

Two sample applications were then designed, implemented, and verified on the evaluation board with embedded Linux. First, data transfer between two tiles were verified. Using handshaking between CPU and tiles, it was confirmed that data transfers are successfully maintained between CPU and tiles. Next, the conventional TMR (triple modular redundancy) was implemented with our scheme. The derived system operated properly, including logic roving.