SUMMARY OF Ph.D. DISSERTATION

School Science for Open and Environmental Systems	Student Identification Number	SURNAME, First name ITOU Tsutomu

Title

Research on the Thread Control Mechanism and the Calculation Mechanism for Responsive Multithreaded Processor for Real-Time Processing

Abstract

Real-time processing is required for realizing a lot of systems including the robot, the home automation, the factory automation, and so on now. Priority is added to each task based on the timing constraint in real-time processing. Each task is switched and executed in priority order to guarantee the timing constraint. Context switching occurs when tasks are switched. An overhead of context switching is large if context switching is performed by software. It becomes a problem when the frequency of task switching increases.

Responsive Multithreaded (RMT) Processor is designed for distributed real-time processing. RMT PU, a processing unit of RMT Processor, executes eight threads (tasks) in priority order without context switching. But Context switching arises as well as traditional processors if nine threads or more are executed.

Large amount of calculation is required for soft real-time processing. As a lot of computing resources are occupied by soft real-time processing, other operations are not likely to be executed. In this paper, we propose the thread control mechanism and the vector operation mechanism to resolve these problems and support hard real-time processing and soft real-time processing by hardware.

Each thread is held in on-chip memory called context cache in the thread control mechanism. Context switching between the hardware context of RMT PU and context cache is performed by hardware. Threads are automatically started by hardware based on the interrupt factor to improve the interrupt response. The started thread is automatically switched and executed by hardware based on priority. The time of context switching has reduced by 0.4% by the thread control mechanism compared with software. It showed that an overhead of context switching was hidden by the priority control mechanism of RMT PU even when the thread with short processing time was executed in the thread control mechanism.

Multiple threads share vector registers in the vector operation mechanism. A configuration of vector registers is changed flexibly to share vector registers efficiently. Multiple vector operations are performed with one instruction to continue vector calculations even if instruction slots are not allocated by the priority control mechanism of RMT PU. Lower priority threads are executed by using the remaining resources because the calculations for soft real-time processing are performed by the vector unit. So the throughput of entire system improved.

It is considered that more advanced real-time systems can be built by using the proposed mechanism because hard real-time processing and soft real-time processing are supported by hardware.