学位報告 4

報告番号	※甲　　第　　号

主　論　文　の　要　旨

論文題目　PREEMPTIVE SOFTWARE-HARDWARE MULTITASKING ON DYNAMICALLY PARTIALLY RECONFIGURABLE FPGAS
(動的部分再構成FPGAにおけるプリエンプティブ・ソフトウェア・ハードウェアマルチタスキング)

氏　　名　JOZWIK, Krzysztof

論　文　内　容　の　要　旨

This thesis describes a design framework, reconfigurable hardware platform and Operating System for Reconfigurable Systems (OS4RS) supporting Software-Hardware (SW-HW) multitasking on Dynamically Partially Reconfigurable (DPR) Field Programmable Gate Arrays (FPGAs).

DPR technology allows to partially change contents of the initial FPGA's configuration at run-time, without disturbing the operation of the rest of the system. This allows for implementation of a concept of a Hardware (HW) Task, i.e., a flow of execution running on an FPGA, which shares the reconfigurable resources with other HW tasks in a time-multiplexed manner. For full realization of the concept, an OS4RS supporting SW-HW multitasking is essential. The OS4RS allows the HW tasks to be active computing objects which contend for the computing resources and request OS services just like SW tasks do.

Concept of a HW task and OS4RS is of a great significance for the FPGA industry and research community. It allows an application programmer to take advantage of the computational power of reconfigurable resources without going into all the intricacies related to the underlying DPR technology. Moreover, when coupled with the High-Level Synthesis (HLS) technology, it significantly decreases a barrier-to-entry for software developers with weak or no background in hardware design for FPGAs.

The main contributions proposed by this thesis can be divided into two wide research areas. One of them is related to the HW task preemption mechanisms and base reconfigurable hardware platform for HW multitasking. The second one is related to the OS4RS.

In the first defined area of research, this thesis proposes two HW task preemption schemes giving a better support for preemption than the previous research works. It also presents a base reconfigurable hardware platform for HW multitasking which utilizes these schemes and significantly reduces the overall reconfiguration and preemption time. Finally, it presents a comprehensive quantitative and qualitative analysis of the HW task preemption, which previous research works failed to do. The preemption itself is a necessary mechanism to support blocking inter-task communication and synchronization semantics often found in embedded systems applications and gives an opportunity to improve system responsiveness and utilization of reconfigurable resources.

In the second defined area of research, this thesis makes contributions to the state-of-the-art in architecture of the OS4RS, its scheduling mechanism as well as inter-task communication and synchronization.

Regarding the first contribution, this thesis shows a complete model and implementation of the lightweight SW-HW co-designed OS supporting preemptable HW tasks with a dynamically scalable clock frequency. The majority of previous works either concentrated on the dynamic clock scaling feature of the FPGAs without applying it in the context of the OS or, on the contrary, only assumed its existence in their scheduling policies for SW-HW multitasking.

There has been a multitude of research works devoted to preemptive scheduling policies for HW multitasking systems, however, they did not show the performance benefits of applications executed entirely in a preemptive manner. We believe that, while support for preemption in the SW-HW Multitasking system is important, it should be restricted to cases when it is either required functionally or brings the benefits to the performance of the system. This thesis, rather than assuming a specific scheduling policy, gives a novel scheduling mechanism for efficient future implementation of the policy. It is based on timely and priority-based reservation of reconfigurable resources and allows for use of preemption only at the time it brings benefits to the total performance of the system. Furthermore, the architecture of the scheduler and the way it schedules allocations and deallocations of the HW tasks on the FPGA results in shorter latency of API calls, thereby reducing the overall OS overhead.

Finally, when compared to previous works, this thesis shows a novel model and implementation of a channel-based inter-task communication and synchronization suitable for SW-HW multitasking with preemptable and clock-scalable HW tasks. It allows for optimizations of the communication on per task basis and takes advantage of more efficient point-to-point message passing rather than shared-memory communication, whenever it is possible.