

Title	アプリケーションに適応した組込みプロセッサとリアルタイムOSの自動生成に関する研究
Author(s)	宮内, 哲夫
Citation	
Issue Date	2021-03
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/17480
Rights	
Description	Supervisor: 田中 清史, 情報科学研究科, 博士

氏名	宮内 哲夫		
学位の種類	博士(情報科学)		
学位記番号	博情第 453 号		
学位授与年月日	令和 3 年 3 月 24 日		
論文題目	A Study on Automatic Generation of Embedded Processors and Real-Time OS Adapted to Applications		
論文審査委員	主査	田中 清史	北陸先端科学技術大学院大学 教授
		井口 寧	同 教授
		金子 峰雄	同 教授
		丹 康雄	同 教授
		中條 拓伯	東京農工大学 准教授

論文の内容の要旨

In recent years, the use of embedded microprocessors has been increasing with prevailing IoT (Internet of Things) and RTOSs are commonly used to develop a real-time system effectively. By using an RTOS, a complicated procedure can be divided into tasks with real-time scheduling based on the preemptive and priority-based task scheduling, and a real-time application can be developed easily with using RTOS functions such as task management, task dependent synchronization, synchronization and communication. While there are advantages for using an RTOS, an RTOS itself consumes additional memory, computational resources and power. Based on these points, our research objectives are as follows: (1) Build an application adaptive processor core. (2) Remove unused codes in RTOS kernel, while leaving necessary functions such as checking possible errors. (3) Implement hardware RTOS to reduce the amount of software resources and execution time. (4) Remove unused codes in a hardware RTOS as well as a software-only RTOS. (5) Build automatic development environment with which we can perform the items above.

In order to achieve the objectives, we adopted MIPS32 architecture for a processor core and illustrated the method for analyzing an application program and generating the application adaptive processor core circuit. In addition, we implemented two- to eight-core multi core processor on an FPGA and showed eight-core processor can be implemented on a relatively small FPGA with application adaptive processor cores.

Regarding RTOS, we proposed a framework to generate application adapted hardware RTOS and software-only RTOS. For the specification of an RTOS, we adopted μ ITRON4.0 for the research as it is widely used and its specification is open in public. We propose the methods, “Removing Unnecessary Codes Caused by Fixed Attributes” and “Removing Unnecessary Codes Caused by the Way of Calling”, for generating an application adaptive RTOS kernel. For the former method, as each system call is specified with attributes through parameters in a configuration file, functions which are not specified in the configuration file can be deleted from the RTOS kernel. For the latter method, error codes for system calls are defined in the RTOS

specification whereas codes for checking errors which never occur in the application program remain in some cases. Since those codes are redundant when an application program is fixed, it is shown that how unnecessary error checking can be removed. In addition, we explained the structure of the hardware RTOS, which consists of RTOS Hardware Wrapper and RTOS Hardware Core. We propose an environment to generate an application adaptive processor core and a hardware/software-only RTOS kernel in a fully automatic manner.

For the evaluation of the effect of the proposal, we applied the proposed methods to several application programs and measured FPGA resources, RTOS kernel execution time and the size of the software parts. As a result, it can be seen that the hardware resources and the size of a software part of an RTOS kernel are reduced, and that the system call execution time is improved.

Keyword: processor, MIPS, RTOS, μ ITRON, configuration, system call, FPGA, adaptation

論文審査の結果の要旨

本論文はリアルタイム組込みシステムにおいて共通コンポーネントとして使用されるプロセッサおよびリアルタイム OS に対し、アプリケーションに応じて構成を細粒度で最適化する手法とそれを実現する自動適応化環境を提案し、いくつかのアプリケーションプログラムに対して提案手法を適用した結果を議論したものである。論文では (1) アプリケーションに適応化した (対象アプリケーションが必要とする命令群とそれを実行するための最小構成の論理回路のみを持つ) マルチコアプロセッサコアの自動構築、(2) リアルタイム OS 内で対象アプリケーションが必要としない機能をコード断片レベルで削除する手法、(3) ソフトウェアサイズと実行時間を削減するハードウェアリアルタイム OS の実装、(4) 上記の全てをアプリケーションソースコードから完全自動的に実現する開発環境の構築、および (5) 提案手法を組込みアプリケーションに適用した場合の評価について記述している。

プロセッサ適応化手法の効果を実証するために、MIPS32 命令セットに基づくマルチコアプロセッサをハードウェア記述言語で設計し、実際に FPGA を利用した評価においてハードウェア資源量を 53~76%削減可能であることが示されている。また、リアルタイム OS の適応化手法に関して、 μ ITRON4.0 仕様に準拠したソフトウェア RTOS (システムコール群) およびハードウェア RTOS を開発し、提案手法を適用する実験において、ソフトウェア RTOS の場合は最大 63% のコードサイズ削減、ハードウェア RTOS の場合は最大 42% のコードサイズ削減が可能であることが示されている。加えて、ハードウェア RTOS 導入によるシステムコールの実行時間の短縮効果と各種適応化によるプロセッサおよびハードウェア RTOS の動作周波数向上効果も示されている。

提案内容と同様の適応化は開発者が手作業で行うことも可能であるが、相当の知識・経験と時間が必要である。一方、本研究の完全自動化環境はこれを極めて短時間でいい、かつ手作業の場

合のような不注意によるバグ注入の恐れがない点が優れている。しかしながら、近年注目されている埋め込みプロセッサ内蔵型 FPGA を利用するシステムに対する優位性に関して十分な考察は行われておらず、またハードウェア／コードサイズの削減が実際のコスト削減にどの程度貢献可能であるかは実証されていなことなどが課題として残されている。

以上、本論文は、コスト制約の厳しい組込みシステム開発分野においてハードウェア量とソフトウェアコードサイズを自動的に削減する方式を提案したものであり、学術的意義のみならず産業界に対する貢献も見込まれる。よって博士（情報科学）の学位論文として十分価値あるものと認めた。