Master Thesis

“Simulation of Multi-core Scheduling in Real-Time Embedded Systems”

Background

The consequences of different scheduling strategies are often poorly understood. The simulation of different scheduling paradigms and their visualisation can significantly increase the degree of understanding. Therefore, a simulation tool supporting several conventional scheduling strategies was developed. Since the tool has a number of limitations by focusing mostly on single-core strategies, its extension is most desirable. Lately, the advent of multi-core systems in the real-time arena has added to the need of understanding the consequences of scheduling decisions, since many properties of the traditional approaches no longer hold. The simulation of different scheduling paradigms and the visualization of system behaviour for systems with several cores would be highly advantageous.

Tasks of the Thesis

The objective of the thesis is to extend the available simulation software to support different multi-core scheduling algorithms for task sets of varying properties, and to enhance the provided visualisation component while keeping it suitable for teaching and researching scheduling concepts and consequences.

The thesis is expected to meet at least the following requirements

- A study of the literature in search of promising multi-core real-time scheduling approaches.
- Extension of the simulator and the graphical visualiser written in Java to support several multi-core scheduling mechanisms including the use of shared resources and the consequences of blocking on several cores. The graphics library to be used is provided by the Eclipse platform.
- The following scheduling paradigms must be supported for multi-core scheduling: Pfair, Partitioned EDF, FMLP. Other paradigms for multi-core and single-core scheduling are optional.
- Design and implementation of a Response Time Analysis for the given task sets for single-core and partitioned strategies where each partition is assigned to a single core.
- Support for multi-core interrupt handling should be included.
- Clusters and groups of tasks should be supported.
- Task suspension should be supported.
- The resulting system is expected to be well documented and easily maintainable so that later extensions are possible. Therefore, high-quality design and documentation are imperative.
- An intermediate presentation is expected to be held 3.5 months after the starting date and the final presentation no later than 1.5 months after the submission of the written report.

The following additional capabilities are optional tasks for this thesis:

- Design of a user interface that enables the user to change simulation models from within the software.
- Design and implementation of a Response Time Analysis for the given task sets for select multi-core strategies.
- Exporting of the results as real-time tasks for a testbed (Litmus RT is one possible choice) is desirable. Decisions should be coordinated with the Consultant.

Supervising Professor: Prof. Dr. Erhard Plödereder
Consultant: Mikhail Prokharau
Starting Date: now