SERVER DESIGN FOR μC/OS-II

Our current research efforts go towards the implementation of hierarchical scheduled real-time systems. Hierarchical scheduling frameworks (HSFs) allows independent application development by postponing the analysis of the global real-time system behaviour until system integration [1].

An example implementation of a HSF built on top of a commercial real-time operating system (RTOS), VxWorks, is presented by Behnam et al. [2]. Similarly, we currently extend μC/OS-II [3] with hierarchical scheduling support. Opposite to the approach in VxWorks taken by Behnam et al. [2], μC/OS-II provides the kernel sources. The availability of kernel sources might lead to a different design and expectably a better performance.

The industry standard in real-time systems is FPPS, although EDF has shown better processor utilization bounds [4]. An idea is to provide FPPS at the application level, while scheduling servers with EDF.

This assignment comprises to investigate which servers, scheduled under EDF, are suitable to service critical sporadic tasks within HSFs. A sporadic server is not suitable to be scheduled using EDF. As a alternative, Buttazzo introduced a constant bandwidt server (CBS) [5].

Please, describe the following in sufficient detail in a report:

1. Explain why a sporadic server is not suitable for EDF scheduling.
2. Compare the CBS with the servers introduced in the RTA course.
3. Which server type would you implement for serving critical a-periodic tasks? (motivate your decision: implementation complexity versus efficiency etc.)

Additionally, choose one server and outline implementation directions for μC/OS-II. Describe the following in sufficient detail:

1. a task-model for a HSF
2. representation of servers and their attached tasks
3. implementation directions for a server type of your choice, i.e. describe the event handling of all relevant events such as server activations and replenishments
4. analyse the complexity of your implementation directions

One might assume availability of support for periodic tasks. An extension to μC/OS-II is available for handling event queues [6]. A copy of [3] is available in HG 5.04. You don't have to do the actual implementation, as long as the outlined descriptions deal sufficiently with the potential pitfalls.

The group is expected to give a presentation about the outlined solutions of approximately 45 minutes, including 10 minutes for questions.

In case you have any questions concerning this assignment, please contact:

Martijn van den Heuvel (m.m.h.p.v.d.heuvel@tue.nl)
HG 5.04
http://www.win.tue.nl/~mheuvel/

REFERENCES

[1] I. Shin and I. Lee, "Periodic resource model for compositional real-time guarantees," in *24th IEEE Real-Time Systems Symposium (RTSS)*, Dec. 2003, pp. 2–13.

[2] M. Behnam, T. Nolte, I. Shin, M. Åsberg, and R. J. Bril, "Towards hierarchical scheduling on top of VxWorks," pp. 63–72, July 2008.

[3] J. J. Labrosse, *Microc/OS-II*. R & D Books, 1998.

[4] G. C. Buttazzo, "Rate monotonic vs. edf: judgment day," *Real-Time Syst.*, vol. 29, no. 1, pp. 5–26, 2005.

[5] ——, *Hard Real-time Computing Systems: Predictable Scheduling Algorithms And Applications (Real-Time Systems Series)*. Santa Clara, CA, USA: Springer-Verlag TELOS, 2004.

[6] M. Holenderski, W. Cools, R. J. Bril, and J. J. Lukkien, "Multiplexing real-time timed events," *WiP session of the 14th IEEE Int. Conference on Emerging Technologies and Factory Automation (ETFA)*, July 2009. [Online]. Available: http://www.win.tue.nl/~mholende/ucos/