EINDHOVEN UNIVERSITY OF TECHNOLOGY
Department of Mathematics and Computer Science

Examination Real-time Architectures (2XN26)
on Monday, November 4th 2013, 9.00h-12.00h.

First read the entire examination. There are 6 exercises in total. Grades are included between parentheses at all parts and sum up to 10 points. Motivate all your answers. Good luck!

1. Cyclic executives:

   (a) (0.5) The "Single rate - AFAP" approach has as main advantages that it is simple, fast, and tasks experience no preemptions. Give two *disadvantages* of this approach.
   Answer: Potentially unbounded jitter and energy inefficient; see `RTS.B3-Cyclic executives`.

   (b) (0.5) How can these two disadvantages be resolved?
   Answer: By means of a time-driven approach, putting the processor in sleep-mode while waiting for the timer to expire; see `RTS.B3-Cyclic executives`.

2. EDF and RMS:

   (a) (0.5) EDF and RMS are both optimal in some sense. Explain in which sense.
   Answer: See book.

   (b) (0.5) Let a set of hard periodic tasks with given characteristics be schedulable by means of both EDF and RMS. Assume an overrun happens for a particular task, e.g. that task requires more computation time than its given worst-case computation time or arrives more frequently than its minimum inter-arrival time. Which tasks can potentially miss their deadline due to such an overrun under EDF and RMS?
   Answer: Under EDF all tasks, including the task experiencing the overload. Under RMS the task experiencing the overload, and all tasks with a lower priority.

3. Analysis for fixed-priority preemptive scheduling:

   (a) (0.5) Which relation between WD_i and WT_i must hold to determine the worst-case response time of a task τ_i using the following recursive equation and why must this relation hold?

$$x = B_i + WC_i + \sum_{j<i} \left\lceil \frac{x + AJ_j}{WT_j} \right\rceil \cdot WC_j. \qquad (1)$$

   Answer: $WD_i + AJ_i \leq WT_i$. When this relation is not satisfied, the smallest positive solution of (1) need not be the worst-case response time of task τ_i, because a next job of τ_i may experience interference from a previous job.

   (b) (0.5) What do the left-hand side and the right-hand side of (1) represent?
   Answer: Assuming a simultaneous release of all tasks τ_j with $j \leq i$ at time $t = 0$, the LHS represents the amount of time *available* in $[0, x)$ and the RHS represents the *max.* amount of time *requested* by all tasks τ_j with $j \leq i$ in $[0, x)$ plus the *max.* blocking B_i due to tasks with a lower priority than τ_i.

   (c) (0.5) Extend (1) with a term reflecting the immediate interrupt service handling of event-triggered tasks.
   Answer: See `RTS.B5-Analysis-9-practical factors`.

(d) (0.5) Express finalization jitter of a task in terms of (*i*) best-case and worst-case response times and (*ii*) best-case and worst-case finalization times. Is the finalization jitter in either case exact?
Answer: See `RTS.B5-Analysis-2-FPPS`.

4. Fixed-priority servers:

   (a) (1.0) In "M. Stanovich et al, *Defects of the POSIX Sporadic Server and How to Correct Them*, In: Proc. 16^{th} IEEE RTAS, pp. 35 - 45, April 2010", the problem of "Premature Replenishments" is described. Describe the problem in your own words.
   Answer: See paper and `RTS.B4-Policies-2-FP-servers`.

   (b) Consider a set $\mathcal{T}$ of n periodic tasks, scheduled by means of fixed-priority preemptive scheduling, and a server S scheduled at an intermediate priority to service a-periodic requests. Let the server be characterized by a period T_S and a capacity C_S. Consider three options for S, i.e. a polling server, a deferrable server, or an idling periodic server. Which type of server would give rise to

      i. (0.5) the highest amount of interference for tasks with a lower priority than S for worst-case analysis?

      ii. (0.5) the lowest amount of interference for tasks with a lower priority than S for best-case analysis?

      iii. (0.5) the shortest average response time for a-periodic requests serviced by S?

   Answer: See `RTS.B4-Policies-2-FP-servers`.

5. Resource access protocols:

   (a) (0.5) Assume fixed-priority scheduling and let tasks share resources requiring mutually exclusive access. Can priority inversion in such a setting be prevented?
   Answer: In general, priority inversion is inevitable; see `RTA.B4-Policies-3-RAP`. Note that for some shared resources, *abortion* of critical sections is conceivable (effectively preventing priority inversion), albeit at the cost of additional overhead.

   (b) (0.5) The Priority Inheritance Protocol (PIP) is stated to be *transparent protocol*. What does that mean?
   Answer: See book.

   (c) (1.0) Give an example illustrating transitive priority inheritance for PIP.
   Answer: See `RTA.B4-Policies-3-RAP` or book Fig. 7.10.

6. Guest lectures given by Prof. C. Hentschel and related topics:

   (a) (0.5) Around the year 2000, video-processing for consumer terminals was based on ASICs and focussing on *cost-effectiveness*. Which additional requirement for future consumer terminals was envisioned in 2000, next to cost-effectiveness, and why?
   Answer: *Flexibility*; see `SVAs Lectures 2008-01`.

   (b) (0.5) To fully exploit programmable platforms, *scalable applications* were combined with *dynamic resource management*. Describe a technique (or *mechanism*) at the level of an RTOS providing a *virtual platform* to (scalable) applications.
   Answer: *Resource reservation*; see `RTA.C6-Resource reservation`.

   (c) (0.5) Briefly describe *priority processing* and its advantages.
   Answer: See `SVAs with Priority Processing`.