EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computer Science

Examination Real-time Architectures (2IN25)
on Tuesday, January 6^{th} 2009, 14.00h-17.00h.

First read the entire examination. There are 5 exercises in total. Grades are included between parentheses at all parts and sum up to 9 points. Good luck!

1. The hyperbolic bound $HB(n) : \prod_{1 \le i \le n}(U_i^{\mathcal{T}} + 1) \le 2$, where $U_i^{\mathcal{T}}$ denotes the processor utilization of the task τ_i, is an example of a *sufficient* schedulability test.
   (a) (0.5) What can be concluded concerning schedulability of a set $\mathcal{T}$ of n task when the hyperbolic bound does not hold for $\mathcal{T}$?
   Answer: The task set $\mathcal{T}$ may but need not be schedulable. Hence, nothing can be concluded yet...
   (b) (1.0) Construct an example set of three tasks for which the left-hand side of $HB(3)$ is equal to 2.
   Answer: See RTA.Rehearse-080918.
   (c) (0.5) Is the task set that you constructed schedulable according to the Liu & Layland bound? Motivate your answer.
   Reminder: $LL(n) : U^{\mathcal{T}} \le n(2^{1/n} - 1)$, where $U^{\mathcal{T}}$ denotes the processor utilization of the set $\mathcal{T}$.
   Answer: The Liu & Layland bound will also hold if and only if the utilizations of all three tasks are the same; see RTA.Exercises-2 "Solutions LL-bound". Because the Liu & Layland bound is also a sufficient condition, just like the hyperbolic bound, nothing van be concluded when the bound does not hold.

2. Consider two periodic tasks τ_1 and τ_2, with characteristics as given in the following table, which are scheduled by means of fixed-priority scheduling with deferred pre-emption (FPDS), where τ_1 has a higher priority than τ_2.

	$T = D$	C
τ_1	3	$1\frac{1}{2}$
τ_2	5	$1 + 1\frac{1}{4}$

   (a) (1.0) Determine the worst-case length WL_2 of the level-2 active period.
   Answer: Draw a timeline with a simultaneous release of both tasks at time $t = 0$. Because the pending load becomes zero at time $t = 9$, the worst-case length $WL_2 = 9$.
   (b) (1.0) Determine the worst-case response times WR_1^{D} of τ_1 and WR_2^{D} of τ_2 using

$$WR_{ik}^{\mathrm{D}} = \begin{cases} WR_i^{\mathrm{P}}(B_i^{\mathrm{D}} + kC_i - F_i) + F_i - (k-1)T_i & \text{for } i < n \\ WO_i^{\mathrm{P}}(kC_n - F_n) + F_n - (k-1)T_n & \text{for } i = n \end{cases}, \tag{1}$$

   and the following recursive equations for $WR_i^{\mathrm{P}}(C)$ and $WO_i^{\mathrm{P}}(C)$, respectively.

$$x = C + \sum_{j<i} \left\lceil \frac{x}{T_j} \right\rceil C_j, \tag{2}$$

$$x = C + \sum_{j<i} \left(\left\lfloor \frac{x}{T_j} \right\rfloor + 1 \right) C_j. \tag{3}$$

Answer: For task τ_1, it is only necessary to consider the first job: $WR^{\mathrm{D}}_1 = 2\frac{3}{4}$. Because the worst-case length WL_2 of the level-2 active period contains $\left\lceil \frac{WL_2}{T_2} \right\rceil = 2$ jobs, we need to consider two jobs for task τ_2. $WR^{\mathrm{D}}_{2,1} = 3\frac{3}{4}$ and $WR^{\mathrm{D}}_{2,2} = 4$, hence $WR^{\mathrm{D}}_2 = 4$.

3. Consider four periodic tasks τ_1, τ_2, τ_3 and τ_4 (having decreasing priority), which share five resources, A, B, C, D, and E. Compute the maximum blocking time B_i for each task for the following three protocols, knowing that the longest duration $D_i(R)$ for a task τ_i on resource R is given in the following table (there are no nested critical sections).

	A	B	C	D	E
τ_1	6	7	0	10	3
τ_2	0	0	0	8	0
τ_3	4	14	8	0	0
τ_4	0	11	0	9	7

(a) (1.0) Priority Inheritance Protocol (PIP).
Answer: Similar to Exercise 7.5 of the book of Buttazzo. Compared to that exercise, the columns have been exchanged ($A \to E \to B \to A$ and $C \leftrightarrow D$), and all non-zero values have been increased by 1.

(b) (0.5) Priority Ceiling Protocol (PCP).
Answer: Similar to Exercise 7.6 of the book of Buttazzo.

(c) (0.5) Highest Locker Protocol (HLP).
Answer: Same as for PCP.

4. The following questions concern presentations of assignments.

(a) (0.5) *Hierarchical scheduling of independent applications*: Two approaches were presented for two-level hierarchical scheduling of independent applications, one termed "Hierarchical Fixed-Priority Pre-emptive Scheduling" by Davis and Burns and the other "Periodic resource model for compositional real-time guarantees" by Shin and Lee. Although the analysis for the 2nd approach is more pessimistic than the former, the analysis for 2nd approach also has a major advantage. Explain that advantage.
Answer: Given the parameters of its server, the schedulability of a subsystem can be analyzed in isolation, i.e. without any knowledge about periods or capacities of other servers.

(b) *Hierarchical scheduling of dependent applications*: Two approaches were presented for global resource access, one approach termed H-SRP and the other termed SIRAP. The approaches differed with respect to their behavior when the amount of remaining capacity of a reservation was less than the amount of time needed for accessing the global resource.

i. (0.5) Explain the difference between both approaches.
Answer: H-SRP assumes budget *overrun* (optionally with *pay-back*) and SIRAP assumes (*self-*) *blocking*.

ii. (0.5) What are the consequence of SIRAP for a supporting mechanism?
Answer: Upon entering a critical section, a check is needed whether or not there is still enough remaining capacity to use the global resource, i.e. to execute the critical section.

5. The following questions concern guest-lectures.

   (a) (0.5) Prof. Christian Hentschel gave a guest lecture entitled *Scalable video algorithms*. Briefly describe his motivation for scalable video algorithms.
   Answer: See his slides.

   (b) (0.5) Mike Holenderski gave a guest lecture entitled *Swift mode changes*. He considered two scheduling approaches for mode changes, fixed-priority pre-emptive scheduling (FPPS) and fixed-priority scheduling with deferred pre-emption (FPDS). Which approach is better and why?
   Answer: Assuming that processing a mode corresponds with a single sub-job, the worst-case lead-time of a mode-change is larger for FPPS than for FPDS, because for FPDS we need to wait for at most 1 sub-job to complete.

   (c) (0.5) Alina Weffers-Albu gave a guest-lecture entitled *Behavioural Analysis of Real-Time Systems with Interdependent Tasks*. She explained that the behaviour of a chain of components scheduled by means of fixed-priority pre-emptive scheduling (FPPS) assumes a repetitive pattern after an finite amount of time. How can that amount of time for the so-called initial phase be minimized?
   Answer: By giving the first task in the chain the lowest priority and/or by minimizing the size of the buffers prior to the lowest priority component in the chain.