EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computer Science

Examination Real-time Architectures (2XN26)
on Friday, July 3^{rd} 2015, 13.30h-16.30h.

First read the entire examination. There are 5 exercises in total. Grades are included between parentheses at all parts and sum up to 10 points. Good luck!

1. For the schedulability analysis, it is typically assumed that tasks do not suspend themselves.

   (a) (0.5) What does that mean?
   Answer: If a task didn't complete its work yet, i.e. there is still work pending, the task is not blocked, and is granted the processor, it will execute rather than wait.

   (b) (1.0) Give an example illustrating the problem of suspension on the applicability of the analysis.
   Answer: Any example illustrating that if a task suspends itself, (*i*) a lower priority task (under FPPS) may miss its deadline due to the delayed interference and (*ii*) the task itself may miss its deadline. Note that self-suspension has a similar effect on lower priority tasks as jitter.

2. (0.5) The hyperbolic bound $HB(n) : \prod_{1\leq i\leq n}(U_i^\tau + 1) \leq 2$, where U_i^τ denotes the processor utilization of the task τ_i, is an example of a *sufficient* schedulability test. Discuss whether or not the following conditions are *necessary* and/or *sufficient*, for the application of the *HB*-bound, or instead that the condition is irrelevant for the *HB*-bound.

   (a) Deadlines are at most equal to periods, i.e. $D_i \leq T_i$.
   Answer: The hyperbolic bound is meant for the rate monotonic (RM) algorithm, i.e. $D_i = T_i$. When the *HB* holds for $D_i = T_i$, the task set remains schedulable when deadlines are increased. Hence, when the assumptions for RM hold with the exception that $D_i \geq T_i$, the *HB*-bound is still a *sufficient* condition. The given condition $D_i \leq T_i$ is too weak for the *HB*-bound. In particular, it is *not necessary*, because $D_i \geq T_i$ is allowed, and it is *not sufficient*, because $D_i < T_i$ is not allowed.

   (b) The sum of the utilizations is at most equal to 1.
   Answer: This need not hold to *apply* the *HB*-bound. When the condition does not hold, the *HB*-bound will not hold either. Formally: because $\sum_{1\leq i\leq n} U_i^\tau + 1 \leq \prod_{1\leq i\leq n}(U_i^\tau + 1)$ we immediately see that when $\sum_{1\leq i\leq n} U_i^\tau > 1$ the *HB*-bound will not hold. The condition is therefore *irrelevant*.

3. Consider three tasks that are scheduled by means of FPPS, where τ_1 has highest and τ_3 has lowest priority, with arbitrary phasing and characteristics as given below.

	$WT = BT$	WD	BD	AJ	WC	BC
τ_1	5	3	1	1	2	2
τ_2	8	8	3	0	3	3
τ_3	25	24	8	0	5	4

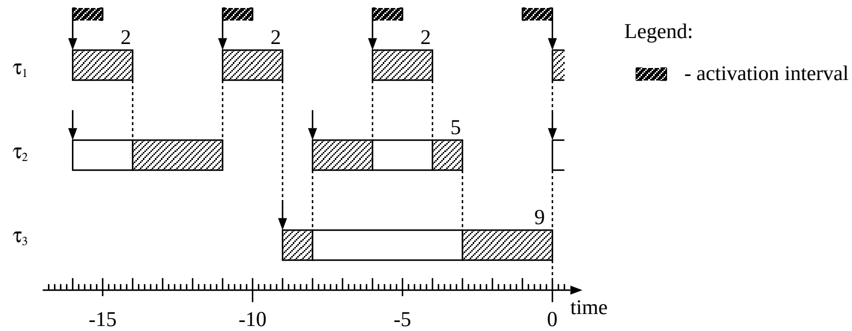

Figure 1: Timeline with an optimal instant for task τ_3.

(a) (0.5) Determine the worst-case response time of τ_3 by means of the following recursive equation.

$$x = B_i + WC_i + \sum_{1 \leq j < i} \left\lceil \frac{x + AJ_j}{WT_j} \right\rceil WC_j. \tag{1}$$

Answer: Note that the *lowest* priority task is *never* blocked. Hence, $B_3 = 0$ irrespective of potential resource sharing. Using an iterative procedure, we find $WR_3 = 24$.

(b) Determine the best-case response time of task τ_3

i. (1.0) by drawing a time line with an optimal instant for τ_3.
Answer: See Figure 1. Note that $BR_3 = 9$.

ii. (0.5) by means of the following recursive equation.

$$x = BC_i + \sum_{1 \leq j < i} \left(\left\lceil \frac{x - AJ_j}{BT_j} \right\rceil - 1 \right)^{+} BC_j \tag{2}$$

Answer: Using $WR_3 = 24$ as initial value for the iterative procedure to determine the best-case response time BR_3, we find $BR_3 = 9$.

(c) (1.0) Let $AJ_3 = 2$. Does this have an influence on the worst-case response time of task τ_3?
Answer: First, observe that $WD_3 + AJ_3 = 26 > T_3 = 25$, hence using (1) is not appropriate in general. Next, because $WR_3 + AJ_3 = 26 > T_3 = 25$, using (1) does not necessarily yield WR_3, i.e. a next job of τ_3 may have a larger response time than the first job due to additional interference of its previous jobs. By drawing a time-line (see Figure 2), we find that the second job has a response time of 25, which is larger than WD_3. From the timeline, we also derive that the level-3 active period ends at the finalization of the second job, hence $WR_3 = 25 > WD_3 = 24$. Hence, the value found by means of (1) is not the worst-case response time of task τ_3 for $AJ_3 = 2$.

4. Servers were presented as a means to schedule aperiodic tasks:

(a) (0.5) Give an advantage and a disadvantage of using *background scheduling* for aperiodic tasks rather than servers.
Answer: See slides.

(b) (0.5) Give an advantage and a disadvantage of using a *polling server* for aperiodic tasks when compared to other types of servers.
Answer: See slides.

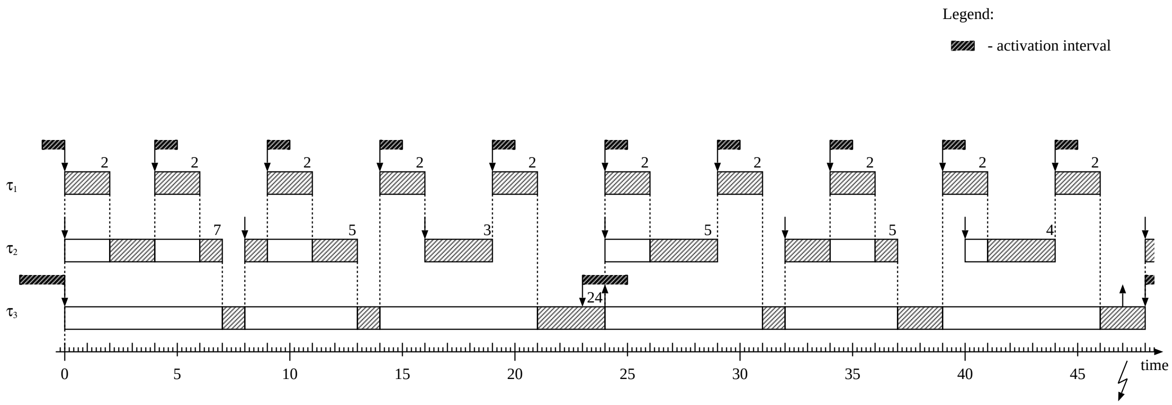

Figure 2: Timeline with a critical instant for task τ_3.

(c) (1.0) The description of the *sporadic server* in Buttazzo is refuted. Explain the problem in your own words.
Answer: See paper: "M. Stanovich, T.P. Baker, A.I. Wang, and M. Gonzlez Harbour, *Defects of the POSIX Sporadic Server and How to Correct Them*, In: Proc. 16^{th} IEEE RTAS, pp. 35 45, April 2010."

5. 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 $\delta_i(R)$ 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) (1.5) 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) (1.0) 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.