EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computer Science

Examination Real-time Systems (2IN26)
on Wednesday, January 23^{rd} 2013, 9.00h-12.00h.

First read the entire examination. There are 7 exercises in total. Grades are included between parentheses at all parts and sum up to 11 points. *Motivate all your answers.* Good luck!

1. Interrupts generated by I/O peripheral devices represent a big problem for the predictability of a real-time system. Three approaches are described in the book of Buttazzo to handle interrupts. Two of these approaches are based on disabling interrupts and using polling (by either application tasks or dedicated kernel routines). The third approach leaves the interrupts enabled and tries to minimize the interference due to interrupts.

   (a) (0.5) Describe the third approach. *Hint*: Distinguish between *device driver* and *device manager* and address the role of the operating system.
   Answer: See book Section 1.3.4.

   (b) (0.5) Describe at least 2 advantages of this third approach.
   Answer: Advantages (see book Section 1.3.4): (*i*) busy wait during I/O operations is eliminated; (*ii*) interrupt interference is minimized (i.e. to the *activation* of a proper task only); (*iii*) priority of the task managing the device is completely independent from other priorities and can be set according to the application requirements.

   (c) (0.5) The basic recursive equation for worst-case response time analysis of tasks under fixed-priority pre-emptive scheduling is given by

   $$x = B_i + C_i + \sum_{j<i} \left\lceil \frac{x + AJ_j}{T_j} \right\rceil C_j. \tag{1}$$

   Extend the equation with a term for interrupt handling.
   Hint: Assume a set of interrupts $\mathcal{T}^i$, a minimum inter-arrival time T_k^i of the interrupt identified by k, and a cost IH_k^i for handling the interrupt.
   Answer: See slides of the lecture.

2. For real-time systems, *priority* $\neq$ *importance.*

   (a) (0.5) Explain why the (relative) priority of a task may be higher than its (relative) importance for fixed-priority scheduling.
   Answer: The priority is typically used as a means to make a task-set schedulable, e.g. RM and DM are optimal priority assignments.

   (b) (0.5) When there is a mismatch between priority and importance, this may give rise to problems. Give an example of such a problem.
   Answer: Upon an overload, e.g. a less important task with a higher priority than a more important task needs more computation time than assumed, the more important task may miss its deadline.

   (c) (0.5) Describe a way to mitigate such problems.
   Answer: Overloads due to consumption exceeding the worst-case consumption time can be limited to the task experiencing the overload by using *reservations* (or servers) per task. This resolves this inter-task overload problem, and leaves the intra-task problem to the task itself. As an alternative, you can split tasks to remain optimal; see RTS.C6-Resource reservation.

3. In the following two fixed-priority scheduling (FPS) situations, the response times of all jobs in a so-called level-i active period have to be considered to determine the worst-case response time of task τ_i. Explain in your own words why looking at the first job alone is not sufficient for

   (a) (0.5) fixed-priority preemptive scheduling (FPPS) and *arbitrary deadlines*;

   (b) (0.5) fixed-priority scheduling with deferred preemption (FPDS) and *deadlines at most equal to periods.*

   Answer: See slides of the lectures.

4. There exist two types of periodic servers, *gain-time providing* and *idling.*

   (a) (0.5) Describe (i) the difference between both types in your own words and (ii) the difference between their "response time analysis parameters".
   Answer: See side-set RTS.B4-Policies-2-FP-servers and RTS.Exercises-5.

   (b) (0.5) Describe at least one practical situation where a gain-time providing periodic server is preferred above an idling periodic server.
   Answer: For hard real-time systems, resource allocation to tasks is typically based on worst-case parameters, and these tasks should therefore not need any additional resources. For greedy applications and (sub-) systems based on QoS (such as cost-effective media processing in resource-limited environments as digital TVs, digitally improved analogue TVs, and Set-Top Boxes, as presented by Prof. Hentschel), it is advantageous to make gain-time available to allow those applications and (sub-) systems to improve the quality of their output. Similarly, the robustness of a system can be improved when the gain-time becomes available rather than being idled away... In summary: in those situations where the quality of the system can be improved based on the availability of gain-time.

5. Consider two periodic tasks τ_1 and τ_2, with $T_1 = D_1 = 4$, $T_2 = D_2 = 6$, $C_1 = 1$, and $C_2 = 2$, and a deferrable server S_{DS} with $T_{DS} = 5$ and $C_{DS} = 2$. The tasks and server are scheduled by means of rate-monotonic scheduling, i.e. the server has an intermediate priority. Assume a simultaneous release of the tasks and the server (e.g. at time $t = 0$).

   (a) (1.0) Aperiodic requests arrive at $t = 3$ with $C_1^a = 3$, at $t = 8$ with $C_2^a = 1$, and $t = 13$ with $C_3^a = 2$, and are handled first-in-first-out. Draw a time line with the executions of the periodic tasks and due to the aperiodic requests, and the remaining capacity of the deferrable server in an interval of length 18.
   Answer: see Figure 1.

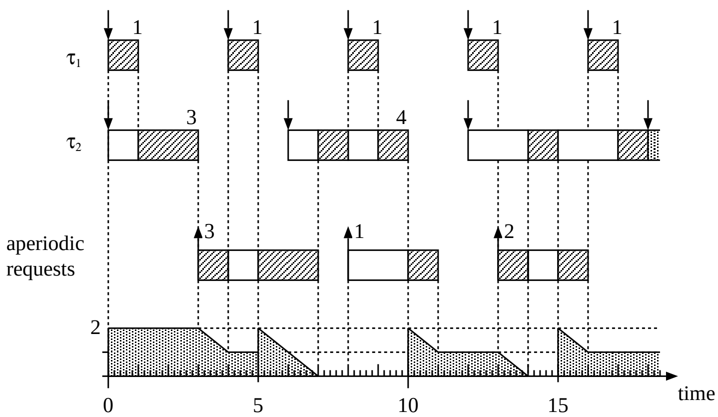

Figure 1: Example of a timeline with a Deferrable Server at intermediate priority.

(b) (0.5) Determine whether or not the periodic tasks are schedulable in the presence of the deferrable server under arbitrary phasing. Motivate your answer by means of *calculations*. *Hint*: Apply the response time analysis using equation (1) above.
Answer: Note that $AJ_{DS} = T_{DS} - C_{DS} = 3$. Based on (1), we find $WR_1 = C_1 = 1 < D_1 = 4$ and $WR_2 > 6 = D_2$. Hence, τ_2 is not schedulable.

6. Consider three periodic tasks τ_1, τ_2, τ_3 (having decreasing priority) that are scheduled by means of fixed-priority pre-emptive scheduling and share three resources, A, B, and C. The longest duration $\delta_{i,R}$ for a task τ_i on resource R is given by the following table (there are no nested critical sections).

	A	B	C
τ_1	3	0	3
τ_2	3	4	0
τ_3	4	3	6

(a) (1.0) Compute the maximum blocking time B_i for each task, assuming the Priority Inheritance Protocol (PIP).

(b) (0.5) Illustrate a situation where τ_2 experiences its maximum blocking time.

(c) (0.5) Compute the maximum blocking time B_i for each task, assuming the Priority Ceiling Protocol (PCP).

Answer: See 3^{rd} edition of the book of Buttazzo, exercises 7.2 till 7.4.

7. The following questions concerns "Expected reading".

(a) A so-called *resource supply bound function* $\mathtt{sbf}_\Gamma(t)$ is defined in "I. Shin and I. Lee, *Periodic resource model for compositional real-time guarantees*, In: Proc. 24^{th} IEEE Real-Time Systems Symposium (RTSS), pp. 2-13, Dec. 2003".

i. (0.5) What does this function represent?
Answer: The minimum resource supply of Γ in an interval of length t.

ii. (1.0) Given a periodic resource $\Gamma(\Pi, \Theta)$, draw $\mathtt{sbf}_\Gamma$ as a function of t for $0 \leq t \leq 4\Pi$, where $\Theta = \Pi/3$. Note that Π represents the period and Θ the capacity.
Answer: See paper. Note that the so-called "blackout duration" (i.e. the longest time without resource supply) is equal to $2(\Pi - \Theta) = \frac{4}{3}\Pi$. Further note that a similar question was asked in earlier exams.

(b) The original analysis for CAN was refuted in "R.I. Davis, A. Burns, R.J. Bril, and J.J. Lukkien, Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised, Real-Time Systems, 35(3): 239-272, April 2007."

i. (0.5) What was wrong in the original analysis and how was it resolved?

ii. (0.5) Which messages are at risk due to the flaw?

Answer: See paper.