EINDHOVEN UNIVERSITY OF TECHNOLOGY
Department of Mathematics and Computer Science

Examination of Real-time Systems (2IMN20)
on Thursday, April 8th 2016, 13.30h-16.30h.

First read the entire examination. There are 7 exercises in total. Grades are included between parentheses at all parts and sum up to 10 points. *Motivate all your answers.* Good luck!

1. (0.5) The specification of a system may not include timing requirements. Where do the (hard real time) timeliness constraints in such a case come from? Give at least two examples.
   Answer: *Timeliness constraints* on a *system* are typically *derived* from the functional requirements and *determined* by (the environment and) design choices in the system; see slides `RTS.B3-Specification concepts`.

2. (0.5) Fixed-priority scheduling (FPS) and dynamic-priority scheduling (DPS) both have their relative strength and weaknesses. Give two strength of FPS and two strength of DPS.
   Answer: FPS is the de facto standard in industry, supported by most commercial off-the-shelf (COTS) real-time operating systems (RTOSs), and behaves "*reasonable*" under overloads, i.e. only the task experiencing the overload and all tasks with a priority lower than the task experiencing the overload are at risk.

   EDF can schedule more task sets than FPS, and gives rise to less pre-emptions than FPS, i.e. has a lower overhead.

3. This question concerns *arrival patterns* of tasks.

   (a) (0.5) Let a task τ have a period T and activation jitter AJ. Describe the minimum inter-arrival time $T^{\min}$ and the maximum inter-arrival time $T^{\max}$ of τ in terms of T and AJ. *Hint*: Make a drawing.
   Answer: $T^{\min} = T - AJ$ and $T^{\max} = T + AJ$.

   (b) (0.5) Suppose a *periodic* task τ *with activation jitter* is modelled as an *elastic* task when determining worst-case response times under fixed-priority pre-emptive scheduling. What is the consequence for the calculated worst-case response time of τ itself and the worst-case response times of tasks with a lower priority than τ? *Hint*: Assume the minimum inter-arrival time $T^{\min}$ is equal to the deadline D, i.e. $T^{\min} = D$.
   Answer: Elastic tasks are *characterized* by a minimum and maximum inter-arrival time. For the worst-case response time analysis, the minimal inter-arrival time will be used.

   The worst-case response time WR of τ when modelled as an elastic task will be the same as the worst-case response time when modelled as a periodic task with activation jitter.

   The worst-case response times of tasks with a lower priority than τ may become *larger*, because the interference of τ may increase.

   (c) (0.5) Sporadic tasks were introduced by Al Mok in his PhD-thesis in 1983. Describe why Mok introduced sporadic tasks and how a system can guarantee the characteristic property of sporadic tasks.
   Answer: see slides `RTS.B3-Reference Model`.

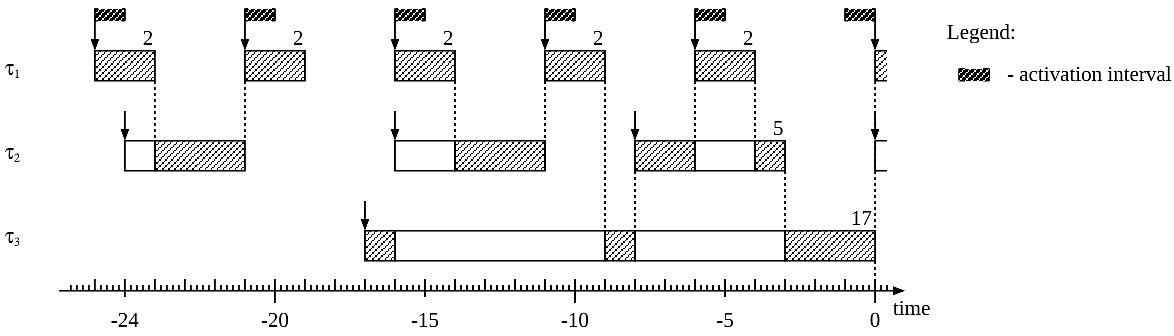

Figure 1: Timeline with an optimal instant for task τ_3.

4. This question concerns *servers*.

   (a) (0.5) Describe the purpose of a server in your own words.
   Answer: See slides `RTS.B4-Policies-2-FP-servers`.
   The "*classic*" answer to this question is to provide good average response times for soft and non-real-time activities, whilst guaranteeing hard real-time tasks. Servers provide an *improved* average response time of aperiodic tasks compared to so-called *background scheduling*.

   With the emergence of hierarchical scheduling and virtual platforms, servers are also used as a means to prevent (or limit) temporal interference between tasks and/or components, as described in `RTS.C6-Resource reservation`.

   (b) (0.5) Explain the difference between an *idling* periodic server and a *gain-time providing* (or *consumption*) periodic server.
   Answer: See slides `RTS.B4-Policies-2-FP-servers`.

   (c) (0.5) Let a system consist of set $\mathcal{T}$ of hard real-time tasks and a deferrable server S_{DS}. Which parameters of the deferrable server are needed for both best-case and worst-case response time analysis of the tasks? *Hint*: Think about period and capacity, amongst others.
   Answer: See slides `RTS.B4-Policies-2-FP-servers`.

5. 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
τ_2	8	8	3	0	3
τ_3	25	25	15	0	5

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

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

   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 $BR_3 = 17$.

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 $BR_3 = 17$.

6. This exercise concerns resource access protocols (RAPs).

(a) (1.0) Transitive priority adjustment may occur when applying the Priority Inheritance Protocol (PIP). Describe at least three conditions that need to hold for transitive priority adjustment to occur.
Answer: We essentially need to describe conditions that may give rise to transitive blocking: (1) at least three tasks, e.g. τ_h, τ_m, and τ_l, with distinct priorities, where τ_h has highest and τ_l has lowest priority; (2) at least two mutual exclusive resources, e.g. R_1 and R_2; (3) τ_h uses a resource R_1, τ_l uses R_2, and τ_m uses R_1 and R_2 in a nested fashion, i.e. it first locks R_1 and than locks R_2 before releasing R_1.

(b) (0.5) How many times can a job J of a task be blocked under PIP?
Answer: At most $\min(n, m)$, where n is the number of lower priority tasks that can block J and m is the number of distinct semaphores that can block J.

(c) (0.5) Give at least 1 advantage and at least 2 disadvantages of PIP compared to SRP.
Answer:
Advantage: *transparency*.
Disadvantages: a task may experience chained blocking, PIP does not prevent deadlocks, and computing the blocking time of a task is hard.

7. This question concerns the guest lectures of Dr. Moris Behnam.

(a) In his 1^{st} lecture on real-time communication, Dr. Behnam addressed CAN.

i. (0.5) Give at least two advantages of CAN compared to the traditional point-to-point wiring.
Answer: Although this was not explicitly mentioned, the reasons are similar to the introduction of computers replacing mechanical subsystems, e.g. (1) it reduces cables and connectors, (2) it saves cost, weight, and space, and (3) it reduces problems with connectors.
Alternatively, from the RTSJ-paper, CAN has been "*designed to provide simple, efficient, and robust communication for in-vehicle networks*", and "*traditional point-to-point wiring became increasingly expensive to manufacture, install, and maintain due to the hundreds of separate connections and tens of kilograms of copper wire required*".

ii. (0.5) Let an original bit-pattern be given by "1111 1000 0111 1000 0111 1". What will be the sequence after bit-stuffing?
Answer: "1111 1**0**000 0**1**111 1**0**000 0**1**111 1", i.e. an extra bit after the first 5 original bits and subsequently an extra bit after 4 original bits.

(b) In his 2^{nd} lecture on real-time communication, Dr. Behnam addressed Ethernet.

i. (0.5) Why is Ethernet not suitable for real-time?

ii. (0.5) Describe at least two techniques to make Ethernet applicable for real-time.

Answer: See his slides.