EINDHOVEN UNIVERSITY OF TECHNOLOGY
Department of Mathematics and Computer Science

Examination of Real-time Systems (2IMN20)
on Friday, April 13, 2018, 13:30h-16:30h.

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. *Specification concepts*
   A terminal, such as a sensor and an actuator, can be modeled as a component with three interface elements for control as illustrated in Figure 1. Briefly describe
   (a) (0.5) the three elements of the interface;
   Answer: The control interface consists of *commands* (arrow down), *observers* (dashed line), and *events* (arrow up); see slides Specification concepts.
   (b) (0.5) the relevance of these elements for both sensors and actuators;
   Answer: The interface of a sensor typically contains a *command* (for initialization) and *observers* and/or *events* to inform the control system about state-changes. The interface of an actuator typically contains *commands*; see slides Specification concepts.
   (c) (0.5) the relation with event-triggered and time-triggered tasks.
   Answer: a so-called *event* will (re-) activate an event-triggered task, whereas a so-called *observer* is particularly useful for a polling (i.e. time-triggered) task.

   Note: This question appeared as Question 1 in the exam of 2014/01/27.

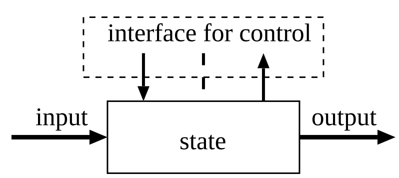

Figure 1: A component-model of a terminal.

2. *Response time analysis*
   (a) (0.5) Describe at least 2 reasons for best-case response time analysis.
   Answer: to determine
      i. whether or not a task meets its best-case deadline, i.e. $BD_i \leq BR_i$;
      ii. a bound on the response jitter and finalization jitter of a task, e.g.
         A. to check whether or not the task meets its control performance or
         B. to determine the activation jitter of a task or message triggered by the task.

   Note: This question appeared as Question 2(a) in the exam of 2017/07/05.
   (b) (1.0) Consider five tasks with characteristics as given below.

	BT	BD	AJ	BC
τ_1	5	1	1	1
τ_2	8	2	0	2
τ_3	9	3	0	3
τ_4	12	1	0	1
τ_5	44	8	0	4

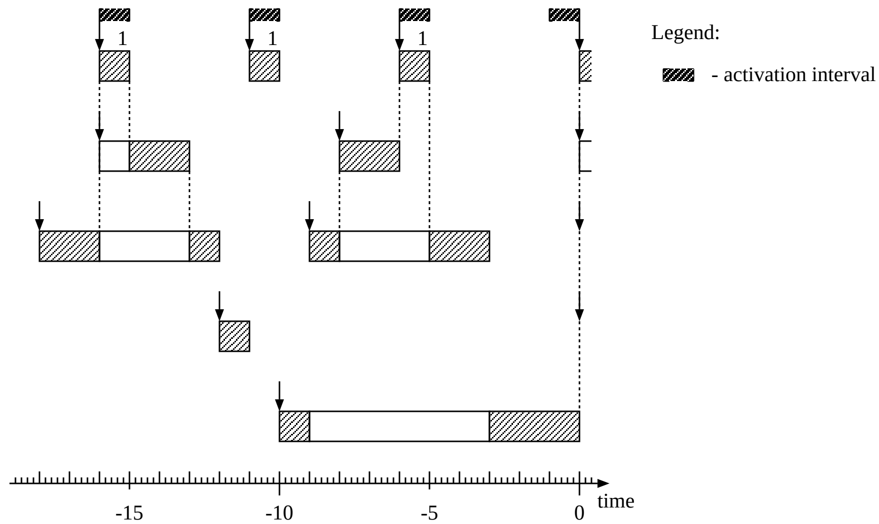

Figure 2: Timeline with an optimal instant for task τ_5.

Determine the best-case response time BR_5 of task τ_5 using a time line, based on the following assumptions:

i. tasks are scheduled by fixed-priority preemptive scheduling (FPPS),
ii. tasks have unique priorities and are given in order of decreasing priorities, i.e. τ_1 has the highest and τ_5 has the lowest priority,
iii. tasks have arbitrary phasing,
iv. tasks do not suspend themselves,
v. scheduling and context switching overheads are ignored,
vi. a job of τ_i is not started before its previous job has completed,
vii. the sum of worst-case deadline WD_i and the activation jitter of τ_i is at most equal to the minimal inter-arrival time WT_i of τ_i, i.e. $WD_i + AJ_i \leq WT_i$.

Answer: See Figure 2. Note that $BR_5 = 10$.
Note: This question appeared as Question 2(f) in the exam of 2017/07/05.

3. *Servers*

Consider a fixed-priority server, characterized by a period T_S, a capacity C_S, and a phasing ϕ_S. Let the server be scheduled at the highest priority in a system. Suppose an aperiodic request A of $2 \times C_S$ arrives at time a_A. Assume no pending load of aperiodic requests at time a_A and no arrival of aperiodic requests at a_A other than A. What is the best-case and worst-case response time of that aperiodic request when it is handled by

(a) (0.5) a polling server,
Answer: The best-case occurs when the aperiodic request arrives at $a_A = \phi_S + k \times T_S$, i.e. at the arrival of the server, and the entire capacity of the polling server is therefore still available at time a_A. The best-case response time BR_A of the request is subsequently identical to $T_S + C_S$.

The worst-case happens when the aperiodic request arrives at $a_A = \phi_S + k \times T_S + \epsilon$, i.e. an infinitesimal time ϵ *after* an activation of the server, and the entire capacity of the server has been depleted at time a_A. The worst-case response time WR_A of the request is now identical to $2 \times T_S + C_S - \epsilon$.
Note: This question appeared as Question 3(b) in the exam of 2017/07/05.

(b) (1.0) a deferrable server.
Answer: The best-case occurs when the aperiodic request arrives at $a_A = \phi_S + k \times$

$T_S - C_S$, i.e. *before* an activation of the server, and the entire capacity of the deferrable server is still available at time a_A. The best-case response time BR_A of the request is subsequently identical to $2 \times C_S$.

The worst-case happens when the aperiodic request arrives at $a_A = \phi_S + k \times T_S + C_S$, i.e. *after* an activation of the server, and the entire capacity of the deferrable server has been consumed at time a_A. The worst-case response time WR_A of the request is now identical to $2 \times T_S$.

Note: This question appeared as Question 4(b) in the exam of 2017/04/18.

4. *Resource Access Protocols*

   (a) Transitive priority adjustment may occur when applying the Priority Inheritance Protocol (PIP).

   i. (1.0) Describe at least three *conditions* that need to hold for transitive priority inheritance 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.

   Note: This question appeared as Question 6(a) in the exam of 2016/04/28.

   ii. (0.5) Illustrate transitive priority inheritance by means of an example, e.g. draw a time-line.

   Answer: See slides.

   Note: This question appeared as Question 3 in the exam of 2016/06/29.

   (b) (0.5) Give at least 1 advantage and at least 2 disadvantages of PIP compared to the Stack Resource Policy (SRP).

   Answer:

   Advantage: *transparency*.

   Disadvantages:

   - a task may experience chained blocking under PIP but not under SRP;
   - PIP does not prevent deadlocks, whereas SRP does;
   - computing the blocking time of a task for PIP is hard and for SRP is (relatively) simple;
   - implementing PIP is hard and implementing SRP is relatively simple.

   Moreover,

   - SRP can also be used in combination with DPS (Dynamic Priority Scheduling) and for multi-unit resources, whereas PIP cannot;
   - SRP gives rise to less pre-emptions than PIP;
   - SRP allows stack sharing, whereas PIP doesn't.

5. *Communication* and *Expected reading*

   This question is about the Controller Area Network (CAN).

   (a) (0.5) Let an original bit-pattern for a CAN message 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**0**", i.e. an extra bit after the first 5 original bits and subsequently an extra bit after 4 original bits.

   Note: This question appeared as Question 7(a)ii in the exam of 2016/04/08.

(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.
Note: This question appeared as Question 6 in the exam of 2014/01/27.

6. *Guest lectures*

(a) (1.0) Prof. Christian Hentschel illustrated that system control can be simplified by shifting control issues to a Scalable Video Algorithm (SVA). Describe the principle of SVAs with priority processing in your own words. *Hint*: priority processing is based on 4 essential elements.
Answer: See his slides.
Note: This question appeared as Question 7 in the exam of 2017/04/13.

(b) Prof. Saad Mubeen presented four different data propagation delays.

i. (0.5) Describe these four delays. *Hint*: Make a drawing and illustrate all four delays.
Answer: See his slide "Data Propagation Delays".

ii. (0.5) From these four delays, two are described in AUTOSAR, being *reaction delay* and *age delay*. Give two examples from the automotive domain, one where reaction delay plays a role and another where age delay plays a role.
Answer: See his slide "Data Propagation Delays: Reaction & Age".

Note: In total, 8.5 points are based on questions from earlier exams.