EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computer Science

Examination Real-time Architectures (2YN26)
on Thursday, January 20^{th} 2011, 9.00h-10.30h.

First read the entire examination. There are 8 exercises in total. Grades are included between parentheses at all parts and sum up to 10 points. Good luck!

1. (0.5) Explain why the response time of a task τ (of a system) can be considerably smaller than the response time of the system to an external event, when τ handles the event.
   Answer: There can be both a *delay* and *jitter* (*i*) between the arrival of the external event and the activation of the task and (*ii*) between the completion of the task and the response of the system. Reasons include processing at I/O devices, information that needs to be transported across a bus between these devices and the CPU, and intervals of times during which interrupts are disabled at the CPU.

2. 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.

   (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.

3. This question concerns fixed-priority scheduling with deferred preemption (FPDS).

   (a) (1.0) Why is the exact worst-case response time analysis for FPDS non-uniform?
   Answer: All tasks can be blocked by a lower priority task, except for the lowest priority task. Because a lower priority task τ_{L} has to start its execution of a sub job *before* the release of a higher priority task τ_{H} in order to block τ_{H}, the blocking time is a *supremum* rather than a *maximum.* The fact that blocking is a supremum is in itself not enough to cause the non-uniformity, however; see for example blocking in the context of fixed-priority pe-emptive scheduling, which results in a uniform analysis. The distinguishing characteristic for FPDS is the fact that the last sub job can't start before all pending work of previous sub jobs and all higher priority tasks has been completed. For a fixed (maximum) blocking time, this starting time of the final sub job is expressed using a floor-function. Because (non-zero) blocking gives rise to a supremum rather than a maximum, we have to consider a limit-case, resulting in a change from a floor-function (plus 1) to a ceiling-function. The latter causes the analysis to become non-uniform. See the slides for more details.

   (b) (0.5) Which constraint must hold to allow resource sharing between tasks scheduled by FPDS without an additional resource access protocol?
   Answer: Critical sections are not allowed to cross sub job boundaries.

4. This question concerns the CAN-bus.

   (a) (0.5) Why is "*bit-stuffing*" applied and how does it work?

   (b) (1.0) Briefly explain the arbitration mechanism on a CAN-bus.

   Answers: See slides.

5. Consider a system consisting of two nodes N_A and N_B and a bus. Assume a strictly periodic external event E triggers a task τ_A on node N_A. Just before its completion, task τ_A sends a message m to a task τ_B on node N_B. Upon reception of m, task τ_B performs some processing, and generates a system response just before its completion. Let the best-case and worst-case response times of τ_A, m, and τ_B be given by BR_A, WR_A, BR_m, WR_m, BR_B, and WR_B, respectively.

   (a) (0.5) Describe the activation jitter AJ_m of message m in terms of the best-case and worst-case response times of τ_A.
   Answer: The activation jitter AJ_m is equal to the finalization jitter FJ_A of task τ_A, i.e. $AJ_m = WR_A - BR_A$.

   (b) (0.5) Describe the activation jitter AJ_B of τ_B in terms of the best-case and worst-case response times of τ_A and m.
   Answer: The activation jitter AJ_B is equal to the finalization jitter FJ_m of message m. If we assume that the worst-case response time of m is determined with respect to the *start* of the interval of its activation jitter (as is done in the lecture on communication), then $AJ_B = WR_m - BR_m$.

6. The following questions concern Hierarchical Scheduling Frameworks (HSFs) and independent applications.

   (a) (1.0) In "[Shin et al 03] 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, December 2003", a *resource supply bound function* $\mathtt{sbf}_\Gamma(t)$ of a time interval of length t is defined that calculates the minimum resource supply of Γ during t units. Given a periodic resource $\Gamma(\Pi, \Theta)$, draw $\mathtt{sbf}_\Gamma$ as a function of t for $0 \leq t \leq 5\Pi$, where $\Theta = \Pi/3$.
   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 the exam of April 16^{th}, 2010.

   (b) (0.5) The worst-case response time analysis of tasks described in "[Davis et al 05] R.I. Davis and A. Burns, *Hierarchical Fixed Priority Pre-Emptive Scheduling*, In: Proc. 26^{th} IEEE Real-Time Systems Symposium (RTSS), pp. 389-398, December 2005." typically yields better results than the analysis in [Shin et al 03]. What is the major *disadvantage* of the approach described in [Davis et al 05]?
   Answer: The approach in [Davis et al 05] does not support independent analysis of subsystems, i.e. the characteristics of all other subsystems (in terms of periods and capacity) must be known to determine the schedulability of the tasks of a subsystem.

   (c) (0.5) The so-called *availability function* in "[Almeida et al 04] L. Almeida and P. Peidreiras, *Scheduling with temporal partitions: response-time analysis and server design*, In: Proc. 4^{th} ACM International Conference on Embedded Software (EMSOFT), pp. 95 - 103, September 2004" is similar to the supply bound function defined in [Shin

et al 03]. A difference is the definition in [Alemeida et al 04] of a so-called *maximum latency* $\Delta = (1+\beta) * (\Pi_S - \Theta_S)$, where $\beta = (WR_S - \Theta_S)/(\Pi_S - \Theta_S)$. Explain the motivation for this definition and its use.
Answer: The term *latency* is similar to the term *blackout duration* of [Shin et al 03], i.e. the longest time a subsystem may not receive any (CPU) resources. Rewriting of the equations yields: $\Delta = \Pi_S + WR_S - 2\Theta_S$. In order to determine Δ, we need to derive WR_S, and therefore need to know all the periods and capacities of higher priority subsystems, similar to [Davis et al 05]. By reducing the blackout duration, the analysis in [Almeida et al 04] improves on the analysis of [Shin et al 03], albeit at the cost of no longer supporting independent analysis of subsystems. Because the test does not consider the specific interference, the (local) schedulability test based on the availability function is more pessimistic than the test of [Davis et al 05]. See paper for more details.

7. One of the lectures concerned "*A QoS approach for multimedia consumer terminals with media processing in software*". The aim of the QoS approach was *cost-effective high-quality video processing in software for multimedia consumer terminals*, motivated by the requirements for *openness* and *flexibility* of these systems, and having as boundary condition that *the existing system qualities should be preserved.*

   (a) (0.5) Explain which real-time problems were addressed.

   (b) (0.5) Explain how these problems have been solved.

   Answer: See slides. The question has been taken from the exam of April 16^{th}, 2010.

8. The following questions concern resource sharing protocols for Hierarchical Scheduling Frameworks (HSFs).

   (a) (0.5) Both SIRAP and H-SRP prevent budget depletion during global resource access. Describe in your own words why budget depletion during global resource access is a problem.
   Answer: Budget depletion during global resource access may add up to the so-called *blackout duration* (i.e. $2(\Pi - \Theta)$) to (*i*) the blocking time of tasks of higher priority subsystems and (*ii*) the interference of lower priority subsystems, effectively reducing the schedulability of the system considerably.

   (b) (0.5) Describe how SIRAP prevents budget depletion during global resource access.
   Answer: Before accessing a global resource R_l, it is checked whether or not there is sufficient budget left to execute the critical section, including pre-emptions by tasks with a priority higher than the local resource ceiling of R_l. When there is enough budget left, the resource is accessed. Otherwise access is delayed till the next replenishment.

   (c) (0.5) Describe how H-SRP prevents budget depletion during global resource access.
   Answer: H-SRP allows a subsystem accessing a global resource to "overrun" its budget, i.e. a subsystem receives a so-called *overrun* budget in addition to its normal budget. The maximum duration of the overrun is determined by the length of the longest critical section accessing a global resource.