ECE720T5 Final exam Information, Applied Track

The exam length is 150 mins. Closed book, outside of provided papers. You can bring a (non-programmable) calculator. The exam will include a mix of questions – examples are provided below for your reference.

Example questions: based on course slides

Note: questions can be based on any set of slides (Lecture 1 to 11 included) presented in class.

Question: Consider the following set of aperiodic real-time tasks, ordered by decreasing priorities and using two resources A, B.

Task	Activation Time	Relative Deadline	Worst-case exec time
τ_1	8	6	3
τ_2	5	13	6
τ_3	3	19	5
τ_4	0	24	9

a) τ_1 : (executes 1 time unit, locks B, executes 1 time unit, releases B, executes 1 time unit)

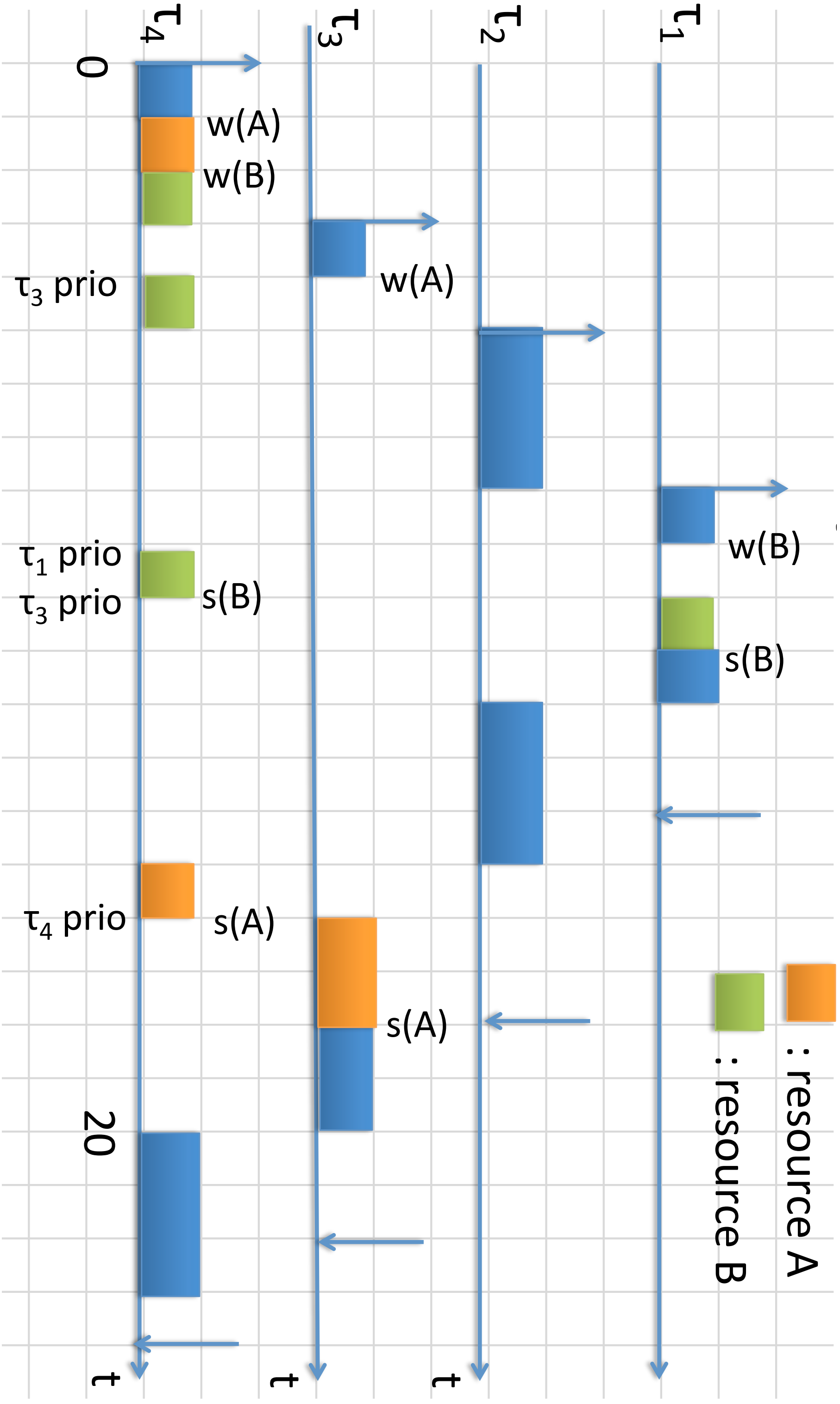
b) τ_2 : (executes 6 time units)

c) τ_3 : (executes 1 time unit, locks A, executes 2 time units, releases A, executes 2 time units)

d) τ_4 : (executes 1 time unit, locks A, executes 1 time units, locks B, executes 3 time units, releases B, executes 1 time unit, releases A, executes 3 time units)

Draw the task schedule using the Priority Inheritance Protocol.

Solution: see the description of Priority Inheritance in Lecture 2. The schedule is drawn below.


τ4
τ3
τ2
τ1
0
20
t
t
t
t
w(A)
w(B)
τ3 prio
w(A)
w(B)
τ1 prio
τ3 prio
s(B)
s(B)
τ4 prio
s(A)
s(A)
: resource A
: resource B

Question: Explain the following concepts: 1) Verification 2) Validation 3) Certification.

Solution: see the corresponding discussion in Lecture 3.

Question: List and discuss existing cache partitioning solutions. Which ones can be implemented in HW? Which in SW?

Solution: see the corresponding discussion in Lecture 4.

Question: What are lock-free and wait-free data structures? How do they help the implementation of real-time multiprocessors?

Solution: see the discussion in Lecture 8 – in either cases, they avoid the need to acquire a lock on a shared resource (by either spinning or suspending). Whether the performance is improved or not tends to depend on the complexity of the lock-free/wait-free implementation.

Example questions: based on assigned papers, concepts only

Note: questions are based only on **key concepts** of the papers in the "required reading list". No questions from the papers in the "presentation list", unless they cover concepts explained in the course slides (for which see the previous category of questions).

Question: The main contribution of the paper "Cyber-Physical Modeling of Implantable Cardiac Medical Devices" is the creation of a heart model. Explain why this is important.

Solution: In general, this is an open question, as the paper lists several reasons, which we discussed in Lecture 3. The key issue is that validating medical devices (in this case, pacemaker) is difficult without a model of the "physical" part of the system (in this case, the heart); in particular, in this case the model allows close-loop verification of the pacing system. The paper concretely shows that there are possible failure modes that cannot be detected using an open-loop system.

Question: The paper "HW Support for WCET Analysis of Hard Real-Time Multicore Systems" implements a Round-Robin arbitration scheme for access to both bus and memory resources. The authors of this research work claim that their approach is more suitable to systems running multiple safety-critical tasks on different cores compared to the arbitration scheme used in the Predator memory controller (which assign different bandwidth and burstiness to each requestor). Explain why.

Solution: We discussed this issue in both Lecture 4, when discussing Predator and memory controller (see slides 71/72), as well as in Lecture 5 when discussing the paper. The key point: if all requestors are cores running safety-critical tasks, then they are all equally important – so it does not make a lot of sense to differentiate among them like Predator does, since this can result in some cores suffering very high delay.

Example questions: based on assigned papers, more detailed

Note: for these questions, the paper will be made available together with the exam. Only papers in the "required reading list" are covered.

Question: consider the provided paper "Predator: A Predictable SDRAM Memory Controller". Assume that we want to use a DRAM device for which the t_RC parameter in Table 1 has a value of 31 instead of 11; all other parameters remain the same. As a reminder, this parameter determines the minimum separation between ACT commands to the same bank. Discuss how you could change the execution of the read and write groups in Figure 3 to work with this DRAM device.

Solution: the groups in Figure 3 have been explicitly discussed during Lecture 4 (see Slide 67). Each group is composed by ACT and RD/WR commands on 4 banks in a row (0 to 3), plus No OPeration to properly separate the commands by the required amount of clock cycles. Each group has a length of 16 cycles and can be repeated immediately after, since the distance between ACT commands to the same bank is equal to 16 > t_RC for the device considered in the paper. If the t_RC constraint is changed to 31 clock cycles, then the groups themselves do not need to be changed, but we can only execute one group every 31 clock cycles rather than 16.