Final Exam for Real Time Systems

2004 Oct 19

Wang Yi

Instructions:

1. You may use a mini-calculator, and a dictionary.
2. No other books (or lecture notes) are allowed.
3. Do not write on the back side.
4. Put page number on each page.
5. You may write in English or Swedish.
6. Mark which problems you have solved in the following table.
7. Please handle in this coverage page together with your solutions.

Problem	Solved	Max. Points	Your Points
1		50	
2		10	
3		10	
4		10	
5		10	
6		10	
	SUMMA:	100	

Name : .....

Pers.no. : .....

Problem 1 *(50p)*

1. *Explain what are "backwards error recovery" and "forwards error recovery".*
2. *Describe what are Fault, Failure and the difference.*
3. *Explain roughly how CAN bus works, and how TTP (timed triggered protocol) works.*
4. *Describe the* Ada *semantics of* Delay 10, *and* Delay until 10.
5. *What is the main difference between hard and soft real time tasks?*
6. *What are the main differences between verification and testing?*
7. *Give examples to show non-preemptive EDF and Shortest Job First are not optimal in scheduling tasks with deadlines.*
8. *Describe informally what are safety and liveness properties, and the difference.*
9. *Describe what are "feasible" and "optimal" schedules.*
10. *Describe a static and a dynamic method to tolerate software faults.*

Problem 2 *(10p) Rate-Monotonic Scheduling will be used to schedule the following task set:*

Task	*WCET*	*Period*
T1	*1*	*5*
T2	*2*	*10*
T3	*4*	*20*
T4	*8*	*40*
T5	*100*	*5000*

1. *Is the task set schedulable? Why?*
2. *If we implement a 2-version system to tolerate software faults using the same hardware, is the task set still schedulable? Why?*

Problem 3 *(10p) Assume two non-preemptive tasks:* $T_1 = (C_1, D_1)$ *and* $T_2 = (C_2, D_2)$ *where* C_i *is the computing time and* D_i *is the deadline for task* T_i*. Assume further that* T_1 *arrives between time* 2 *and* 10 *and* T_2 *arrives between time* 2 *and* 12*.*

1. *Model the two tasks and a scheduler running the FIFS scheduling algorithm (First-In-First-Served) as a network of three timed automata.*

2. *Formalize the schedulability problem of the two tasks as a reachability problem for the network of timed automata.*

Problem 4 *(10p)*

1. *Describe the un-bounded priority inversion problem.*

2. *Describe how BIP (Basic Priority Inheritance Protocol) and HLP (Highest Locker's Priority Protocol) work.*

3. *Can they prevent deadlocks? How?*

The two standard operations P *and* V *on semaphores are implemented according to the following pseudo-code:*

P(S)	V(S)

Disable-interrupt;
if S.counter $>$ 0 **then**
S.counter -- 1
else
begin
insert(current-task, S.queue);
schedule()
end;
Enable-interrupt
	Disable-interrupt;
	If non-empty(S.queue) **then**
	begin
	new-task := first(S.queue);
	insert(new-task, Ready-queue);
	schedule()
	end
	else S.counter ++ 1;
	Enable-interrupt

Modify the above code to implement HLP. You should also describe what information should be kept in the TCB (task control block) and SCB (semaphore control block) for your implementation.

Problem 5 *(10p)*

1. *Are RMS and DMS optimal? If yes, in what sense?*

2. *Describe the sufficient and precise schedulability tests for RMS and DMS.*

3. *It is said that RMS and DMS are stable. What does it mean?*

4. *It is said that EDF (Earlest Deadline First) is non-stable. What does it mean?*

Problem 6 *(10p) Assume that RMS is used to schedule a set of preemptable periodic tasks (C_i, T_i), and the fixed time for storing or loading the context of a task is CS. Assume that whenever a task instance is started, preempted or terminated, its context must be loaded or stored. Modify the following standard equation to take the overheads for context switch into account in calculating the worst case response times:*

$$R_i = C_i + B_i + \sum_{j \in HP(i)} \lceil R_i/T_j \rceil * C_j$$

Describe breifly its informal meaning.