Final Exam for Real Time Systems

2005 Oct 21, 09-14

Wang Yi

Instructions:

1. You may use a mini-calculator, and a dictionary.
2. Do not write on the back side.
3. Put page number on each page.
4. You may write in English or Swedish.
5. State which problems you have solved in the following table.
6. Please handle in this coverage page together with your solutions.

Problem	Solved	Max. Points	Your Points
1		30	
2		10	
3		10	
4		10	
5		15	
6		15	
7		10	
	SUMMA:	100	

Name : ....

Pers.no. : ....

Problem 1 *(30p)*

1. *Give two methods to tolerate software faults. Describe briefly how they work.*

2. *Explain briefly how the "message scheduling mechanism" (i.e. the arbitration mechanism) of CAN works.*

3. *Are DMS and EDF optimal? If yes, explain in what sense. Are they "stable"? If yes, explain in what sense.*

4. *What are the 4 most important language constructs needed in real-time programming?*

5. *Describe two reasons explaining why testing is not good enough, and two reasons explaining why modeling and verification techniques may help in developing real-time systems.*

Problem 2 *(10p) Study the following Ada-like program. Model the two tasks using a network of two timed automata. Can the automaton for task B reach node STOP when it is running together with the automaton for task A?*

Task body A is	*Task body B is*

<pre>loop
  delay 100;
  if x = 1 then B.warning
  else x := 1
end loop</pre>
	<pre>loop
	    select
	        begin
	        accept warning;
	        goto foo
	        end
	     or
	        begin
	        delay 20;
	        x := 0;
	        end
	    end select
	end loop
	foo: STOP</pre>

Problem 3 *(10p)*

1. *Describe four methods to improve the average response times for non-periodic soft RT tasks in a system containing both periodic and non-periodic tasks.*

2. *Give a method to handle non-periodic hard RT tasks for a system running RMS. Describe the equation to calculate the worst case response time for a non-periodic hard RT task.*

Problem 4 *(10p) Assume a system with one processor and three periodic tasks:*

Task	T_i	C_i	D_i
A	52	12	52
B	40	10	12
C	30	10	25

where T stands for period, C for WCET, och D for deadline.

1. *Assume that* $D_i = T_i$ *(i.e. ignore the deadlines given in the table) and RMS is used to schedule the tasks:*
   (a) *What is the priority order?*
   (b) *Construct the run time schedule for the first 52 time units.*
   (c) *Is the task set schedulable? Motivate your answer.*

2. *Assume that DMS is used to schedule the tasks:*
   (a) *What is the priority order?*
   (b) *Construct the run time schedule for the first 52 time units.*
   (c) *Is the task set schedulable? Motivate your answer.*

3. *Assume that EDF is used to schedule the tasks:*
   (a) *Construct the run time schedule for the first 52 time units.*
   (b) *Is the task set schedulable? Motivate your answer.*

Problem 5 *(15p)*

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

2. *Describe the resource access protocols: BIP (Basic Priority Inheritance Protocol) and HLP (High Locker's Priority Protocols). Please use examples if needed.*

3. *Can BIP and HLP prevent deadlocks? Motivate your answer.*

4. *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 {
insert(current-task, S.queue);
schedule()};
Enable-interrupt
	Disable-interrupt;
	If empty(S.queue)
	then S.counter ++ 1
	else {
	new-task := get-first(S.queue);
	insert(new-task, Ready-queue);
	schedule()};
	else S.counter ++ 1;
	Enable-interrupt

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

Problem 6 *(15p) Given a set of periodic tasks some of which may have the same periods.*

1. *Describe how RMS works for the task set.*

2. *Describe how the RMS sufficient schedulability test (i.e. using the utilization bound) works.*

3. *Describe how to calculate the worst case response times for tasks.*

4. *Give a necessary utilization bound for the schedulability of the task set if a 5-processor system is used to compute the tasks (ignore the overheads).*

5. *Give a necessary utilization bound for the schedulability of the task set if the tasks are implemented using 5-version programming technique for fault-tolerance (ignore the overheads and assume that each version of the same task has the same worst-case execution time).*

Problem 7 *(10p) Assume a CAN bus running at 1Mbits per second, connecting three stations (nodes) A, B, and C.*

1. *On node A, there are two tasks. One is sending a message with identity 7 at most every 50ms and The other is sending a message with identity 9 at most every 60ms.*
2. *On node B, there are two tasks. One is sending a message with identity 10 at most every 100ms and the other is sending a message with identity 2 at most every 10ms.*
3. *On node C, there is a single task sending a message with identity 4 at most every 20ms.*

The transmitted messages are of fixed size (120 bits each). Assume that the CAN controllers have sufficient buffer capacity, no transmission errors, and no jitters. What is the worst case transmission delay (i.e. time from queuing to completed message transmission) for the messages with identity 9? Motivate your answer.