Written Exam in Real-Time Programming (Realtidsprogrammering)
Normal Exam

Date: 2015-06-01
Time: 08:15 – 12:15

Responsible Teacher: Farhang Nemati
Phone: 019-303264

Points:
The exam contains 5 questions. In total 40 points.
20 points is required for grade 3, 30 for grade 4, and 35 for grade 5.

Notice:
Please use a new page for every question.
Using a calculator is allowed.

Good Luck!

1. (10 points)

Briefly answer the following questions (You may answer a question by an example if it makes sense):

a) What is the difference between Hard Real-Time Systems and Soft Real-Time Systems
b) What is Task Preemption?
c) What is a Semaphore?
d) What is Priority Inversion?
e) When a Deadlock may happen?
f) What is a Periodic, Aperiodic, and Sporadic task?
g) What is POSIX Standard?
h) In a Real-Time Operating System (RTOS) a task can be in different states. When is a task in READY state? When is it RUNNING state?
i) What is a Client-Server Communication?
j) What is the difference between Synchronous and Asynchronous message passing?

2. (12 points)

Assume you are required to design and implement a real-time system which consist of 4 periodic tasks; task1, task2, task3, and task4. Every 4ms task1 performs some work and then sends a command to be received by task2 or task3. task2 and task3 run with periods 8 ms and 10 ms respectively. task2 and task3 wait for a command from task1 and when they a get a command they perform some work and write their results on an output screen and also send the results to task4. task4 runs with period 6ms. At any time only one task can write its results to the output screen (either task2 or task3). The command queue can hold at most 10 commands. For each task it takes 1ms to perform its work.

a) Design the system using a Petri net model using transitions and timed places. Consider using semaphores and message queues.

b) Implement the system using VxWorks. Please notice that the screen has to be protected from being used by more than one task at any time. In the implementation define a start function, e.g., start(), where all tasks are created. For each task define a start function, e.g. startTask1() for task1, etc. You can use the following function calls:

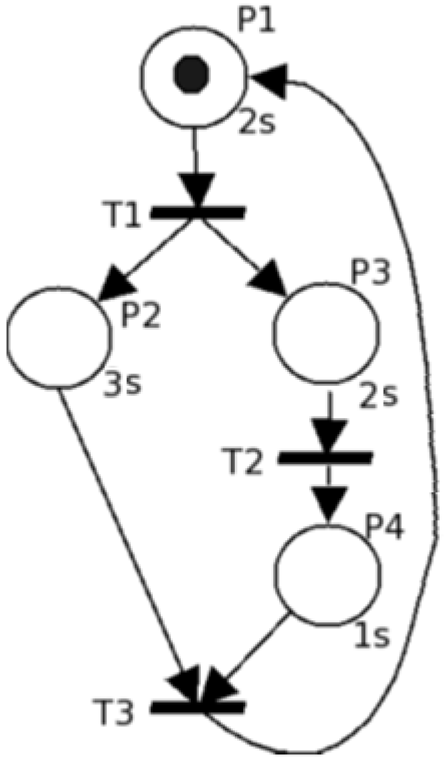
void work(int i); // performs the work for task i
void writeOnScreen(int i); // writes the results of task i on the screen
*int taskSpawn(char *name, int prio, 0, FUNCPTR function,int arg1, …);* // creates a task
sem = semCGreate(SEM_Q_FIFO, int initValue); // creates a semaphore
semTake(sem,WAIT_FOREVER); // takes a semaphore
semGive(sem); // gives a semaphore
q = msgQCreate(int maxQLen, int maxMsgLen, MSG_Q_FIFO); // creates a message queue
msgQSend(q,msg,int maxMsgLen,WAIT_FOREVER,MSG_PRI_NORMAL);//sends a message
msgQReceive(q,msg, int maxMsgLen,WAIT_FOREVER); // receives a message

3. (6 points)

A system has been designed by the following P-timed Petri net.

a) Draw the reachability graph of the P-timed Petri net

b) Assuming the Petri net executes with maximum speed does the Petri net has a period (Is it repeated)? If yes what is the period of the Petri net?



4. (6 points)

a) Assume a task model where tasks are periodic and preemptive and the tasks are independent. Explain briefly how the following scheduling algorithms work for such a task model?
   i. Timeline Scheduling (Cyclic Executive)
   ii. Earliest Deadline First (EDF)

b) What is a Resource Access Protocol for? How the following resource access protocols work?
   i. Priority Inheritance Protocol (PIP)
   ii. Highest Locker Priority Protocol (HLP)

5. (6 points)

Assume the following independent preemptive periodic tasks are scheduled by Rate Monotonic Scheduling (RMS) algorithm. The deadline of each task is equal to its period.

a) Calculate the response times for all three tasks.
b) Is the task set schedulable by RMS?

	e_i (execution time)	p_i(period) = d_i(deadline)
task1	1	3
task2	2	5
task3	2	10

Notice: The following equation is used for calculation of response times:

$$R_i = e_i + \sum_{\tau_j \in H_i} \left\lceil \frac{R_i}{p_j} \right\rceil e_j$$