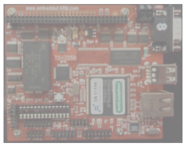
The University of Missouri - Columbia
Electrical & Computer Engineering Department
ECE4220 – Real-Time Embedded Computing



Final

1) Clear your desk top of all **handwritten** papers and personal notes. You may keep **only** the textbook, your test paper, and a pencil.

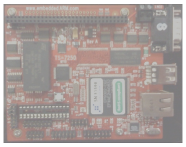
2) Read through the test completely and work the problems you can, leaving the difficult ones until last.

3) Keep your eyes on your own paper. Cheating will not be tolerated!

4) Work problems on the back of the previous page if necessary.

5) **Show your work!**

NAME: ____

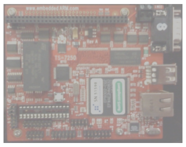


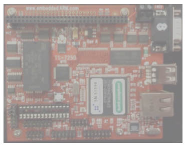
Question 1

Five tasks are to be scheduled using the *Earliest Deadline First* algorithm. The tasks' arrival times (a_i) – i.e. the times at which the tasks become *Ready* – the tasks' execution times (e_i) and the tasks' deadline (d_i) are displayed below.

	T_1	T_2	T_3	T_4	T_5
a_i	0	0	2	3	6
e_i	1	3	2	2	2
d_i	2	5	4	10	9

(a) Assuming a Non-preemptive OS, draw the time diagram with the execution of the five tasks scheduled by the *EDF* algorithm. (Explain why!)
(b) Did any task not meet the deadline?
(c) Re-do the two parts above assuming a preemptive scheduling.





Question 2

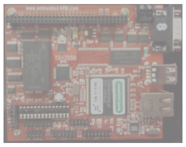
Verify the schedulability of the following two sets of tasks according to the Rate Monotonic algorithm. If the tasks are schedulable, construct the schedule according to RM.

(a)

	T_1	T_2	T_3
e_i	2	2	2
p_i	6	8	12

(b)

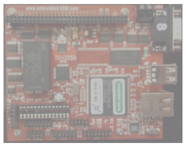
	T_1	T_2	T_3
e_i	1	2	3
p_i	4	6	8

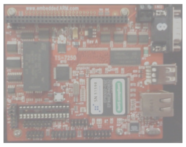


Question 3

A shared memory scheme is to be used as a means of exchanging blocks of data between two tasks, T_0 and T_1. One of the tasks is on the main processor, and the other is executing on an external device. The shared area occupies 2K of memory in a 16Kx8 SRAM; however, the number of bytes written with each exchange is variable.

(a) Present a design for the shared memory system.
(b) Using a UML sequence diagram or another clearly defined diagram, explain how your memory system works by describing a complete cycle that includes the following: a write by T_0; a read by T_1; a write by T_1; and a read by T_0.
(c) How does each task know when data is available and how much data is available?
(d) Are there any potential problems with your design?
(e) How would your design change if three tasks were involved in the exchange?

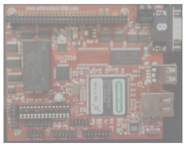




Question 4

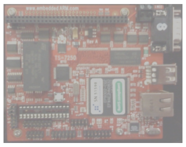
1. Which type of socket communication is considered to be more reliable: Stream or Data Gram? Explain how each type works and the reason for one being more reliable.

2. What function is used to map a variable/pointer to a register (such as port B) in kernel space?



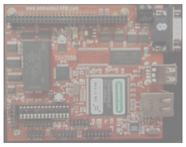
3. Instead of a main section for the case of a user space program, kernel modules have two generic sections of code. What are their names? What is their purpose/function? When are they run?

4. What is it called when a process is perpetually denied necessary resources? Give an example.



5. What mechanism can be used to pass data between a realtime task and a Linux process? Explain how it works.

6. If there is a realtime task that periodically writes to a FIFO, and two Linux processes open that same FIFO and try to read data from it, what is the behavior?



7. If I create a thread with id = tid and call pthread_join(&tid,0) in main, what happens?

8. Please order the following by their priority (1 highest priority 4 lowest priority)

____ Realtime Task

____ linux process

____ Realtime Interrupt

____ Linux interrupt