EECS 473, Homework #1

Due 10/11 at 11pm. This should be good preparation for the midterm, you should also look at old exams on the website when studying for the exam. In particular, expect a design question much like those found in the prior exams. I expect this will take about 6 hours to do.

1. Everything, Everywhere, All at Once

Q1. Multiple choice—Circle the best answer. **[12]**

a) Which of the following is an advantage of RM scheduling over EDF scheduling?

- ***RMS handles deferred interrupts properly (as long as priority inheritance is enabled) while EDF often does not.RMS can schedule certain sets of periodic tasks EDF cannot.***
- ***RMS doesn't require dynamic priorities for periodic tasks, while EDF generally does.***
- ***When RMS fails to schedule, it will always fail to schedule the task with the largest CPU needs, while EDF could fail to schedule other tasks.***

b) Which of the following statements about linear regulators is FALSE?

- ***A 5V output linear regulator with a 10V input would be expected to waste less power than that same regulator with a 12V input.***
- ***An LDO is a type of linear regulator.***
- ***A linear regulator is generally expected to have a less-noisy output than a switching regulator.***
- ***A real linear regulator with a 10V input and 5V output would be expected to waste no more than 50% of the input power.***

c) Which of the following statements about the GPL is FALSE?

- ***It stands for the GNU Public License.***
- ***Things which use the GPL are open source and in the public domain.***
- ***The GPL is sometimes referred to as being a "viral license" because if you use code licensed under the GPL you likely need to license that code under the GPL.***
- ***Linux and gcc are both licensed under the GPL.***

d) The term "rat's nest" is best understood to refer to what with respect to PCBs?

- ***The layout when most/all connections are still "air wires".***
- ***The schematic once the connections have been made.***
- ***The (potentially excessive) use of "neck downs" to reduce PCB trace resistance.***
- ***The use of an excessive number of vias on a PCB.***

2. Scheduling Algorithms

The following questions ask you to consider different scheduling schemes.

Q2. Say you have the following groups of tasks. For each group find the CPU utilization. Figure out which groups are RM schedulable. ***Only*** where needed are you to show the critical instant analysis. **[12]**

Group	*T1 Execution Time*	*T1 Period*	*T2 Execution Time*	*T2 Period*	*T3 Execution Time*	*T3 Period*	*CPU Utilization?*	*RM Schedulable?*
A	2	7	2	7	2	7		
B	1	10	5	12	4	21		
C	4	8	3	9	5	27		
D	1	3	2	5	2	8		

Q3. Consider an embedded application which consists of 3 tasks named A, B, and C. Each task is CPU bound (that is, there is no I/O or memory operations which take significant time to execute) and periodic. Each task must complete before the next instance of that task is ready to start. These tasks have the following properties and requirements. You are to assume there is no overhead of any type (including scheduling overhead) and that this machine runs any given instruction in exactly the same amount of time.

Task	**Max. number of instructions executed**	**Period**
A	1 million	100ms
B	2 million	150ms
C	4 million	500ms

a. Given a choice of a processors with MIPS ratings of 25, 30, 35, and 50, which is the slowest that will be able to schedule these tasks using EDF? Show your work. **[5]**

b. As part a but for RMS. **[10]**

Q4. Answer the following: **[10]**

a. Provide two tasks that together are not schedulable under Round Robin where one has exactly a 10% CPU utilization and the total CPU utilization is less than 75% or state why no such tasks exist.

b. Provide two tasks that together are schedulable under Round Robin where one has exactly a 10% CPU utilization and the total CPU utilization is greater than 75% or state why no such tasks exist.

Q5. Let's look back and see how EDF does with the scheduling problems that RM struggles with.

a. Consider the tasks found in group D of problem 1.
    i. Show how EDF would schedule the tasks through time 15 if all three were released at time 0. **[4]**
    ii. Explain how having dynamic priorities is helpful here. **[3]**
b. Consider the tasks found in group C of problem 1. Either show at what time a task would first fail to make its deadline OR show that EDF could schedule up to time 40 (whichever is true) assuming all tasks were first released at time zero. If it schedules through time 40, will it ever fail at scheduling? Justify your answer. **[8]**

Shared Resources and Their Problems

Q6. Consider the following set of jobs:

i. Task A has a period of 10ms, a worst-case execution time of 3ms of CPU time, and needs to use resource "X" for the entire time.
ii. Task B has a period of 21ms and a worst-case CPU time of 5ms
iii. Task C has a period of 33ms, a worst-case CPU time of 5ms, and needs to use resource "X" for the entire time.

a. If there were no resource dependencies, would the above tasks be schedulable under RMS? EDF? Justify your answer. **[4]**
b. With the dependencies, but assuming no priority inheritance, show an example of RMS and EDF both failing. Assume that task "A" will not even start if task C is running (as resource X isn't available) and via versa. **[8]**
c. Explain how priority inheritance addresses the problem of priority inversion in the general case. **[5]**
d. Will the above tasks be schedulable if priority inheritance is in use? Consider both EDF and RMS. **[6]**

3. Licensing

Q7. In your own words answer the following questions:

a. What does it mean when a license is said to be "viral"? **[4]**
b. Which of the Creative Commons licenses are viral? **[2]**
c. What is the difference between the GPL and the LGPL? **[2]**
d. Describe the CC BY-NC-SA license. **[3]**

Q8. Read the Wikipedia article on Tivoization. Explain in your own words:

a. What Tivo is/was and its basic business model. **[2]**
b. What Tivo did with respect to the GPL. **[2]**
c. How the GPLv3 is a reaction to it. **[4]**
d. What are the major changes from GPLv2 to GPLv3 other than what you mentioned in part b? **[4]**

4. RTOS and Linux

Q9. Answer the following questions. You may need to use resources beyond the lecture material.

a. Jitter **[8]**
   i. What is jitter in the context of an RTOS?
   ii. Say you have an aperiodic task that always takes 1ms of CPU time to run and is triggered on an interrupt (say a GPIO pin going high). Now say you have a task that disables interrupts for no more than 10ms.
      1. What is the worst-case response time the system will have to the interrupt?
      2. Assuming this is the only source of jitter, how much jitter might we have?

b. In the C language, what is a "void *" and why does FreeRTOS use it? Provide an example. **[4]**

c. Sometimes a variable is declared as "volatile". **[4]**
   i. Explain why we often have MMIO locations declared as volatile. What might happen if we didn't do so.
   ii. You hear someone say: "The volatile keyword is often thought to have something to do with caches—that isn't the case. It is directed to the complier, not the hardware." Explain what they mean.

d. Define the term "critical section". **[2]**

Q10. Deferred interrupts and Mutexes

a. Explain the advantages of using a "deferred interrupt". **[4]**

b. The term "top half" and "bottom half" (sometimes "upper-half" and "lower-half") is often used when discussing deferred interrupts. Explain that terminology. **[4]**

Q11. Say we have the following datatypes and functions in a pseudocode-like language:

```
semaphore SemaphoreDeclare(state init);
void SemaphoreTake(semaphore);
void SemaphoreGive(semaphore);
```

where state is a two-state type with states FREE and BUSY

a. Consider the following code in this pseudocode-like language:

```
Init:
  semaphore a=SemaphoreDeclare(BUSY);
  int i=0; j=0;
  int x[8]=0; //all values in the array initialized to 0.

ISR:
  i++;
  SemaphoreGive(a);

Task1:
  SemaphoreTake(a);
  if(i>7) // Change made on 10/9.
      {i=0; j++}
      x[i]=j;
```

   i. Why was the semaphore initialized to BUSY? What would happen if it were instead initialized to FREE? **[3]**

ii. What would be the value of array x, i , and j after the ISR was called 20 times? **[7]**

b. Say you have a function named "work()" that takes some time and in which only one task is allowed to run at a time. Using a similar format to what we have in part a, write an Init, Task1, and Task2 function which uses a Mutex to insure that Task1 and Task2 are never both running "work()" at the same time. (Use the function names as above but replace the work "Semaphore" with "Mutex" in all cases). **[10]**

Q12. Consider the following code found as the read function member of the file_operations struct for a Linux kernel module. It is associated with the device file "/dev/txx2" (so a read of the file /dev/txx2 will result in this function being called). Assume that everything is set up appropriately beforehand. Ignore the fact that copy_to_user's return value is being ignored (it's just a warning…).

```
const char s[] = ,"TheABCsofDRL";
ssize_t memory_read(struct file *filp, char *buf,
                    size_t count, loff_t *f_pos) {


    if(*f_pos>=6)
          return 0;
    int x=count>2?2:count;
    /* Transferring data to user space */
    copy_to_user (buf, s+(*f_pos+1), x);
    printk("<1> *fpos= %d\n",*f_pos);
    /* Changing reading position as best suits */

    *f_pos+=x;
     return x;
}
```

a. Say a user program is run that opens the file and then reads 1 byte at a time until it finds the end-of-file. Each time it reads the data, it immediately prints what it reads.
   i. What will appear in the log file? **[5]**
   ii. What will be printed? **[5]**

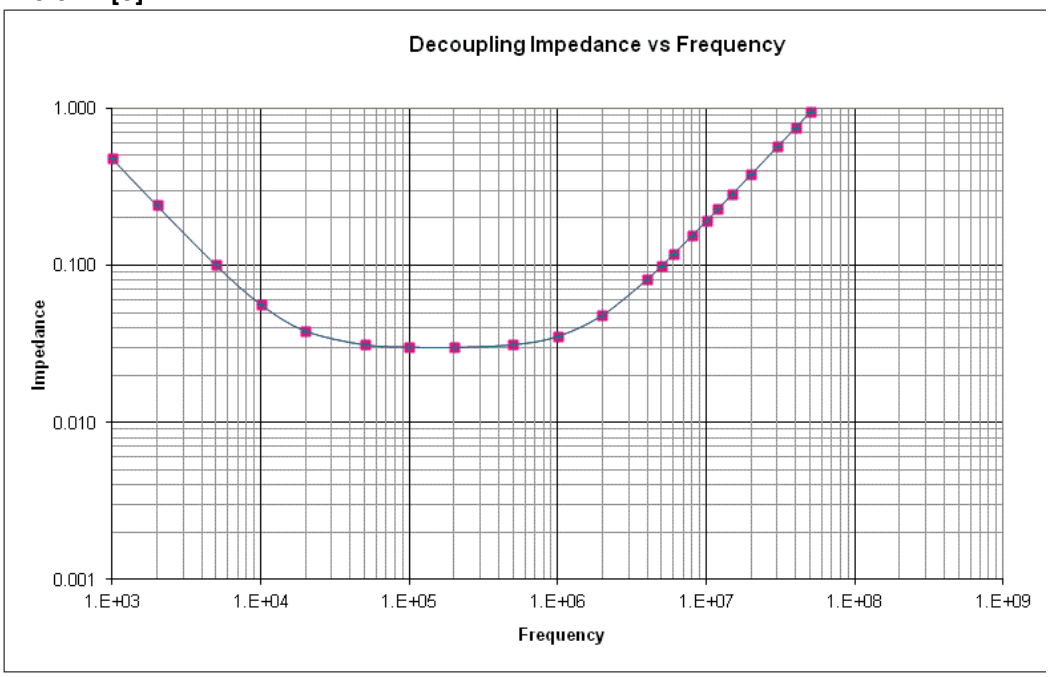
b. Say that someone does a cat of /dev/txx2.
   i. What will appear in the log file? **[5]**
   ii. What will be printed? **[5]**

5. Power Integrity etc.

Q13. In your own words, define the following terms in the context of PCB design: **[8]**

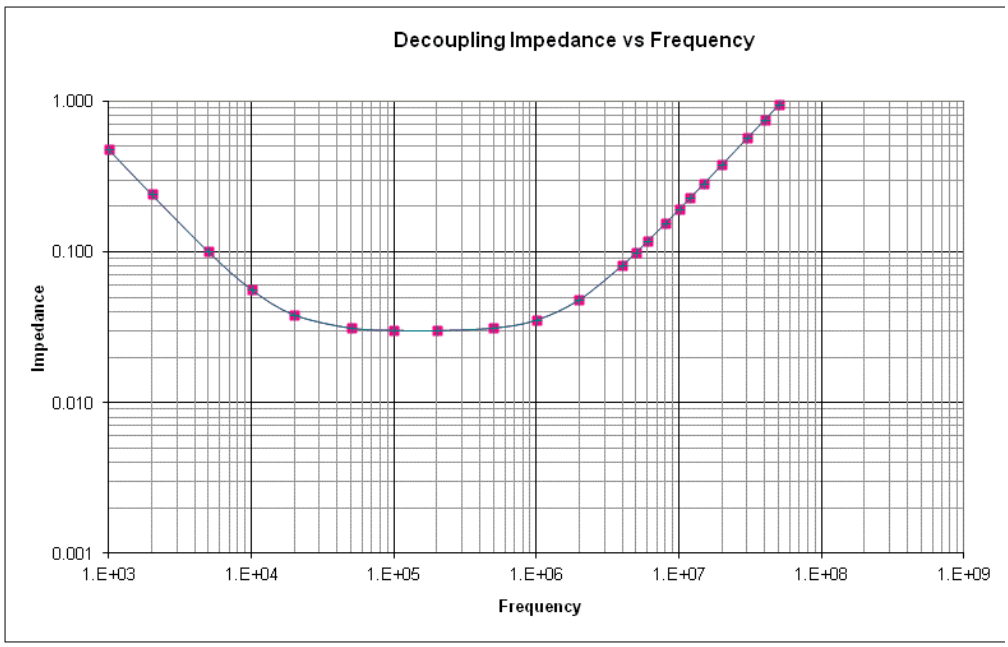
a. Power Integrity (PI)
b. Signal Integrity (SI)
c. Electromagnetic Interference (EMI)
d. Electromagnetic Compatibility (EMC)

Q14. The following is a graph of the effective impedance of a 330µF capacitor with an ESR of 0.03Ω and an ESL of 3nH at a wide range of frequencies. Modify the curve to show what it would look like if it was replaced with a 33µF capacitor with an ESL of 300pH, and an ESR of 0.02Ω **[6]**



Q15. In your own words, explain why putting two capacitors in parallel is often more useful for maintaining power integrity than one capacitor of twice the size. **[5]**

Q16. This is again the graph of the effective impedance of a 330µF capacitor with an ESR of 0.03Ω and an ESL of 3nH at a wide range of frequencies. Show how the curve would change if you instead had two of those capacitors in parallel. **[5]**



Q17. Say you have an FPGA where the Vcc to ground value is required to be in the range of 3.27 to 3.33V. Say that FPGA can draw up to 1.5A and that the voltage regulation module generates a solid 3.3V.

a) What is the maximum impedance your PDN (power distribution network) can have? **[3]**

b) Use the spreadsheet found on the course website with this assignment to generate a set of capacitors from those listed that will meet these requirements. You may assume you only need to worry about frequencies in the 10KHz to 900MHz range and you may include the PCB power/ground plane. Your solution should use as few capacitors as possible while meeting these requirements. **[12]**

6. Batteries and Power Supplies

(Note: we won't be done covering some of these topics until around 10/6, maybe later…)

Q18. Say you have a processor named "Bob" which requires 3.3-4.5V and has the following run modes

Run mode	MIPS	Current
Fast	6.0	200mA
Slow	3.0	92mA
Stopped	0	10mA

If you have a task that runs once a second for 100K instructions and otherwise does nothing. It is running at room temperature. Assuming waking up and sleeping have no cost.

a. How would you use the run modes to achieve the lowest energy utilization? **[4]**
b. Say you are using one of these batteries http://www.powerstream.com/thin-lithium-ion.htm (specifically the PGEB0054050[1]) to power your device. How long would you expect the power to last? (hint, think carefully about limits and assume we are directly powering the processor) **[6]**
c. What if you moved to using two of those in parallel? **[5]**
d. What if we put two of the batteries in series and used a linear regulator? **[5]**

Q19. Peukert Effect

a. Explain what the Peukert effect is. **[4]**
b. How it is the Peukert effect reflected in the Powerstream website (from the question above)? **[6]**

Q20. Say you have a linear regulator with an 8V input and a 6V output. If the load being driven by the regulator is using 4 Watts and the quiescent current is 15mA, how much power is being wasted as heat by the regulator? Show your work. **[8]**

[1] It really is found on that page though it's in an image so you'll not be able to find it with a search…