Embedded Operating System Midterm, Chang Gung University, Autumn 2017

Name: Student ID:

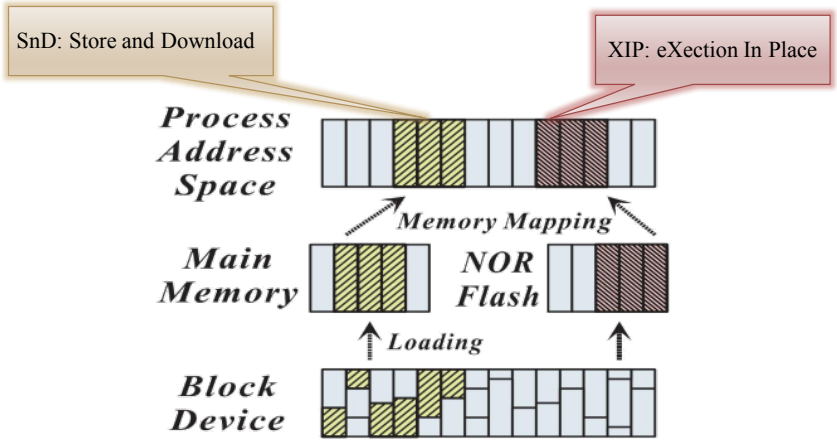
1. (12%) Please define the (a) Non-Recurring Engineering (NRE) Cost and (b) the Unit Cost of a system of products. (c) When will the unit cost be more important than the NRE cost? (d) When will the NRE cost be more important than the unit cost?

Answer: (a) Unit Cost: the monetary cost of manufacturing each copy of the system
(b) NRE Cost: the one-time monetary cost of designing the system
(c) If the number of sales is expected to be large, the unit cost is more important.
(d) If the number of sales is expected to be small, the NRE cost is more important. It could be some customized products.

2. (8%) To develop software on embedded systems, we usually need the cross-platform development environment consisting of some cross compiler, linker, and source-level debugger. (a) What is the cross compiler? (b) Why do we need it?

Answer: (a) A cross compiler is a compiler which can run on the host system, such as a PC, and can produce the binary which can run on the target embedded system.
(b) Some embedded systems do not have enough computing power, memory and/or system software to support the compiler. Thus, cross compiling is needed to build the embedded software.

3. (12%) For program execution, please define (a) Store and Download (SnD) and (b) eXection In Place (XIP). You can refer to the following figure to answer questions a and b. For each object in an embedded system, we can specify the load address for its address in ROM and the run address for its execution. (c) When can we use the same address for the load address and the run address of an object? (d) When should the load address and the run address of an object be different?



Answer: (a) SnD: It is to store the results in the storage before power-off and to download binary and data before running programs.
(b) XIP: It is to directly run programs on non-volatile and byte-addressable memory.
(c) If the object is read-only and will be used only once (or few times), we can keep it in ROM.
(d) If the object is not read-only and might be accessed many times, we should load it into DRAM to write some data and to improve the performance.

4. (8%) For real-time operating systems, please define (a) Pure Real-Time OS and (b) Real-Time Extension of a General OS.

Answer: (a) A pure real-time OS is designed for real-time requirements and should be completely real-t

ime compliant
(b) A real-time extension of a general OS is to extend an OS by real-time components so as to support real-time applications and non-real-time applications.

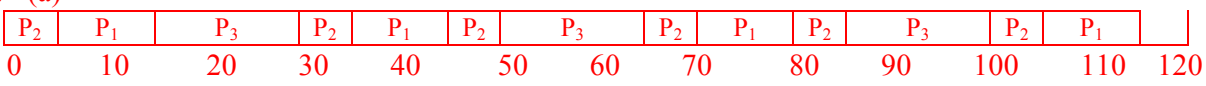
5. (8%) Please define (a) "Soft" real-time systems and (b) "Hard" real-time systems.
Answer: (a) Soft real-time systems: We want to meet the deadline constraint so as to guarantee the quality of applications, but deadline missing is not fatal, e.g., multimedia applications.
(b) Hard real-time systems: If the deadline is missed, critical data are permanently lost or people might get hurt. Thus, it does not allow any deadline missing, e.g., nuclear power plant controllers and anti-lock brake systems.

6. (8%) Context switching is an overhead of task scheduling. Thus, whenever we have a new task scheduling algorithm, we would like to analyze the number of context switching. Stack Discipline is a very useful rule for analyzing the context switching overhead of a task scheduling algorithm. Please provide the definition of Stack Discipline.
Answer: If process A preempts process B, process A must complete before process B can resume.

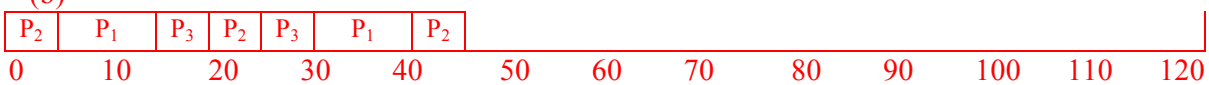
7. (8%) For many applications and systems, it is (computationally) impossible to derive the Worst Case Execution Time (WCET) of them. However, when conducting real-time task scheduling, we need to have a value for the execution time of each task. What should we do for getting some value for the execution time of each task?
Answer: We should get a safe upper bound of the real WCET and use the upper bound as the execution time for the scheduling algorithm. For the quality of the WCET analysis, we should push the upper bound as close to the real WCET as possible.

8. (8%) For an application thread, we can use Data Partition and/or Functional Partition for using multiple cores. Please define "Data Partition" and "Functional Partition."
Answer: Data partition: The method partitions data into several sets and runs each set on a core independently so as to explore the parallelism.
Function partition: The method partitions a function into several sub-functions and runs sub-functions on different core in a pipeline fashion.

9. (12%) For three periodic tasks P_1, P_2 and P_3. P_1 has its period 30 and execution time 10. P_2 has its period 20 and execution time 5. P_3 has its period 40 and execution time 15. Please draw the scheduling results of (a) the Earliest Deadline First scheduling and (b) the Rate Monotonic Scheduling from time 0 to time 120. If there is any deadline missing, please point it out and stop the scheduling when it has the deadline missing.
Answer: (a)



(b)



Task P_3 misses its deadline at time 40.

10. (12%) Consider 4 tasks, t_1, t_2, t_3, and t_4 which have priorities x_1, x_2, x_3, and x_4, respectively, and assume $x_1>x_2>x_3>x_4$ (x_1 is the highest priority). After we profile the programs of the 4 tasks, we have the following information:

- Task t_1 will lock semaphore S_1 for 3 ms.
- Task t_2 will lock semaphore S_2 for 4 ms and lock semaphore S_1 for 5 ms.

- Task t_3 will lock semaphore S_2 for 9 ms and lock semaphore S_3 for 7 ms.
- Task t_4 will lock semaphore S_1 for 8 ms and lock semaphore S_3 for 10 ms.

Please derive the priority ceiling of each semaphore. If the Priority Ceiling Protocol (PCP) is used to manage the semaphore locking, please derive the worst-case blocking time of each task.

Answer: Priority Ceiling: S_1: x_1, S_2: x_2, S_3: x_3
Blocking Time: t_1: 8 ms, t_2: 9 ms, t_3: 10 ms, t_4: 0ms

11. (12%) A sporadic server has a replenishment period 5 and the maximum execution budget 2. Let the sporadic server have the budget 2 at time 0. Assume that events arrive at 1, 3, 6, 9, 11, and each event consumes the execution time 1. Please draw a diagram to show the changing of the execution budget from time 0 to time 20.

Answer: