Module 1: Lecture Questions

Module 1 - Lecture 1 (Wednesday - 3/24/21)

None.

Module 1 - Lecture 2 (Thursday - 3/25/21)

1. What distinguishes Real-Time Software from Traditional Software?
2. What are three challenges of developing Real-Time Software?
3. What are the definitions of Latency, Execution time, response time, release time, relative deadline?
4. For a periodic task with a period of 1 msec, what is its relative deadline?
5. What is preemption?
6. How does adding priority and enabling preemption affect latency and execution times?
7. Under what conditions is a task considered schedulable?
8. In the ece3849_int_latency project, there is an event0 with a 6001 usec period and a clock running at 120 MHz.
    a. What is the value of the TIMER0_PERIOD we use to program the interval timer for TIMER0?

There are 120 counts in 1 usec. To program 6001 usec the timer count value = 120 * 6001 = 720,120.

    b. If the event handler reads a current time value at the very beginning of the service routine of 700,000. What is the latency in usec of the event 0 task?

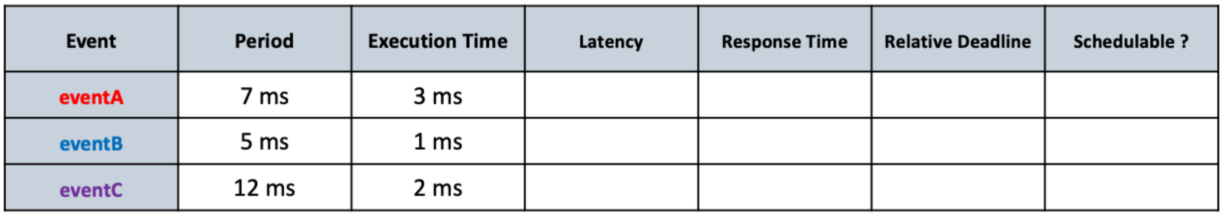
The latency will be the difference between the (TIMER0_PERIOD – the current timer value)/120. Dividing by 120 converts from timer ticks to usecs.
(720120-700,000)/120 = 167.67 usec.

    c. How can the service routine tell if it has met its deadline?
    d. If the clock were running at a 100 MHz, how would it affect these calculations?

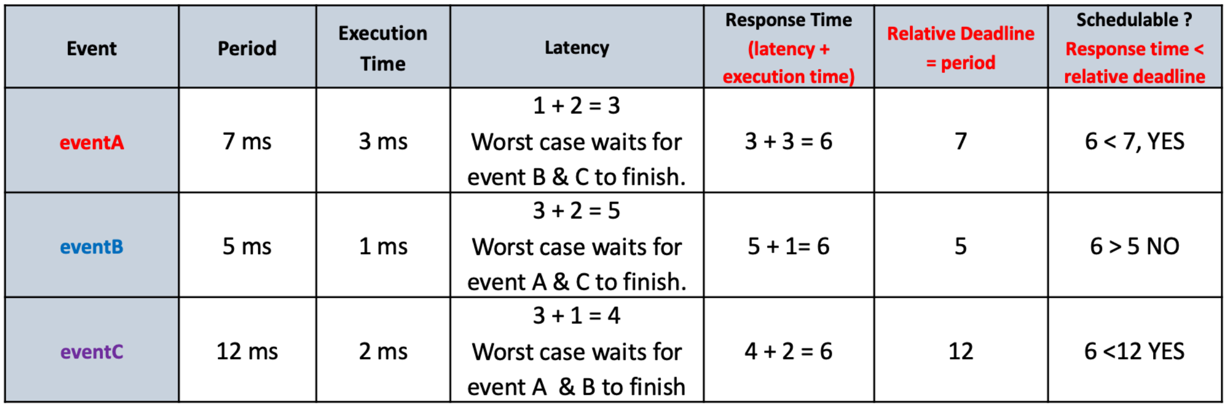
If the clock is running at 100 MHz, then there would only be 100 counts per usec instead of 120.

Module 1 - Lecture 3 (Friday - 3/26/21)

9. In a Canonical Real-Time Systems what are the three system assumptions do we make about the system.
10. Why is it important to use the maximum latency and execution values when calculating response time?
11. Describe the Round Robin and Priority Polling algorithms. List one pro and one con for each scheduling method.
12. In the Round Robin scheduling example, we had three event0 , event1, and event2.
    a. Which event is most likely to be un-schedulable in this method? Why?
    b. If we changed to Priority Polling which event is most likely to be un-schedulable? Under what conditions would this occur?
13. For the below tasks fill in the latency , response time, relative deadline and if it is schedulable using the Round Robin polling method.



Event	Period	Execution Time	Latency	Response Time	Relative Deadline	Schedulable ?
eventA	7 ms	3 ms				
eventB	5 ms	1 ms				
eventC	12 ms	2 ms				



Event	Period	Execution Time	Latency	Response Time (latency + execution time)	Relative Deadline = period	Schedulable ? Response time < relative deadline
eventA	7 ms	3 ms	1 + 2 = 3 Worst case waits for event B & C to finish.	3 + 3 = 6	7	6 < 7, YES
eventB	5 ms	1 ms	3 + 2 = 5 Worst case waits for event A & C to finish.	5 + 1= 6	5	6 > 5 NO
eventC	12 ms	2 ms	3 + 1 = 4 Worst case waits for event A & B to finish	4 + 2 = 6	12	6 <12 YES

Module 1 - Lecture 4 (Monday - 3/29/21)

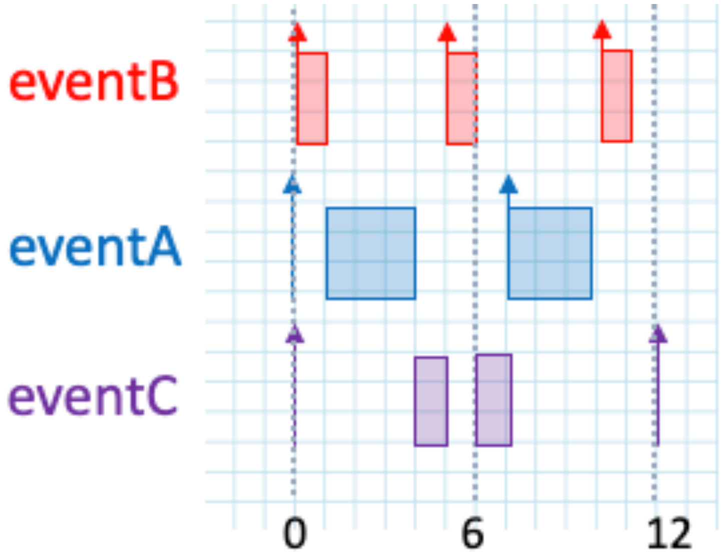
14. For the below tasks fill in the values for latency , response time, relative deadline and if it is schedulable using the Priority polling method.
    a. List in the priority column which task is high, mid and low priority and explain why.
    b. These are the same tasks from question 13. Did changing to Priority Polling improve the performance? Why / Why not?

Event	Period	Execution Time	Priority	Latency	Response Time	Relative Deadline	Schedulable ?
eventA	7 ms	3 ms					
eventB	5 ms	1 ms					
eventC	12 ms	2 ms					

Event	Period	Execution Time	Priority	Latency	Response Time (latency + execution time)	Relative Deadline = period	Schedulable ? Response time < relative deadline
eventA	7 ms	3 ms	Mid	2+1 = 3 ms Worst case waits for event B & C to finish.	3 + 3 = 6 ms	7	6 < 7 YES
eventB	5 ms	1 ms	High	3 ms Will wait for either eventA or Event B Worst case: Max(3,2)	1 +3 = 4 ms	5	4 < 5 YES
eventC	12 ms	2 ms	Low	3+1 = 4 ms Worst case waits for event A & B to finish	2 + 4 = 6 ms	12	6 < 12 YES

15. When using preemptive scheduling,
    a. What is the roll of the interrupt controller?
    b. What tasks run in the foreground? What tasks run in the background?
    c. How should the tasks be prioritized?
    d. What happens if all the interrupts have the same priority?
    e. Why does response time become more difficult to calculate?
16. Interrupts can be globally disabled for some or all of the ISR.
    a. Why would we want to disabled the interrupts?
    b. Why is it important to minimize the amount of time that interrupts are disabled for?
17. Below are the same tasks as in question 13 and 14. For each task fill in the priority, latency, relative deadline and if it is schedulable using preemptive scheduling / interrupts.
    a. Would you use preemptive scheduling for these tasks or stay with a polling strategy? Explain.

Event	Period	Execution Time	Priority	Latency	Response Time	Relative Deadline	Schedulable ?
eventA	7 ms	3 ms					
eventB	5 ms	1 ms					
eventC	12 ms	2 ms					



Event	Period	Execution Time	Priority	Latency	Response Time (latency + execution time + preemption time)	Relative Deadline	Schedulable ?
eventA	7 ms	3 ms	mid	1 ms Worst case waits for eventB	1 +3 = 4	7	4 < 7 YES
eventB	5 ms	1 ms	high	0 ms Highest priority can not be preempted.	0+1 =1	5	1 < 5 YES
eventC	12 ms	2 ms	low	3 + 1 = 4 ms Worst case waits for event A & B	4 + 2 + 1 = 7 4 ms latency + 2 ms execution time + 1 ms preemption from event0 =7 ms	12	7 < 12 YES

Module 1 - Lecture 5 (Tuesday - 3/30/21)

18. What are the conditions that must be met for Rate-Monotonic Scheduling Theory?
    a. How have we guaranteed by design or by measurement that these conditions are met in our in-class examples using the ece3849_int_latency project?
    b. We have a system running in lab with three tasks which we assume meets all RMS conditions. We have measured the average CPU utilization to be 50%. Which RMS condition may not be met by the measurement?
    c. We have a system running in lab with two tasks that forces all tasks to run at their maximum execution time all the time. Under these conditions we measure a CPU utilization of 66%. However, we notice that the highest priority interrupt is missing its deadline occasionally and suspect that one of the RMS conditions is not being met. Name one condition might cause this behavior, if it is not met? Explain.
19. A system is using Preemptive scheduling with interrupts for the three periodic real-time tasks with the characteristics below. The system satisfies all the RMS conditions.
    a. Calculate the CPU utilization for the system. Show your work.

CPU utilization = 4ms/12ms + 2ms/5ms + 4ms/19ms = 0.9439 = 94.39%

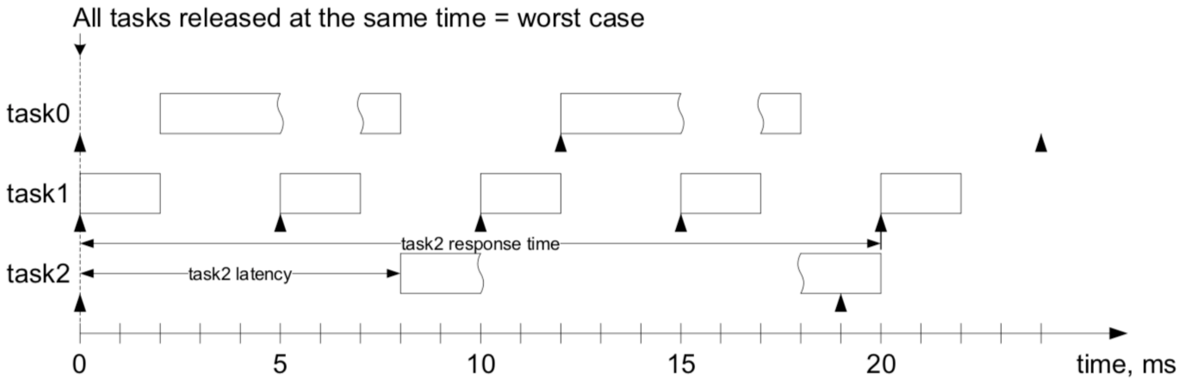
   b. Using the RMS utilization bound, are the tasks guaranteed to meet all the deadlines? Explain.

RMS CPU utilization bound for three tasks is 0.7798 = 77.98%
CPU utilization (0.9439) > RMS CPU utilization bound (0.7798)
Therefore the CPU utilization does not guarantee that the system is schedulable by RMS.

   c. Determine which is the high, mid and low priority task and fill in the table below. Explain your answer.
   d. How many msec do you need to draw out to determine the worst-case values? Explain.
   e. Using the Graphical Model draw the worst-case RMS schedule to determine the latency, response time and if it is schedulable and fill in the table below.

Task	Period	Execution time	Priority	Latency	Response time	Schedulable ?
task0	12 ms	4 ms				
task1	5 ms	2 ms				
task2	19 ms	4 ms				

Task	Period	Execution time	Priority	Latency	Response time	Schedulable ?
task0	12 ms	4 ms	**mid**	**2 ms**	**8 ms**	**Yes**
task1	5 ms	2 ms	**high**	**0 ms**	**2 ms**	**Yes**
task2	19 ms	4 ms	**low**	**8 ms**	**20 ms**	**No**


All tasks released at the same time = worst case
task0
task1
task2
task2 response time
task2 latency
0
5
10
15
20
time, ms