MITx 6.004.3x

Computation Structures 3: Computer Organization

Help selfpoised

Course Progress Dates Discussion

Course / 18. Devices and Interrupts / Lecture Videos (46:07)

< Previous Next >

LE18.4

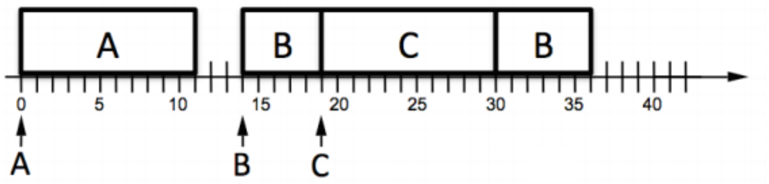
Bookmark this page

Calculator

LE18.4.1: Strong Priorities

0.0/1.0 point (ungraded)

A real-time operating system with priority interrupts has three interrupt handlers (A, B, C), each of which, when invoked by the appropriate interrupt request (marked as ↑ in the execution timelines), takes 11 time units to execute. For example, the following execution timeline shows the A handler running to completion after an A interrupt request, followed by execution of the B handler, which is itself interrupted by execution of the C handler.

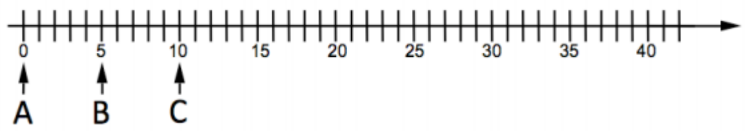


Another way of representing this execution timeline is using a table like the one below:

Interval start time	Interval end time	Device
0	11	A
11	14	NONE
14	19	B
19	30	C
30	36	B
36	40	NONE
NONE	NONE	NONE

Note that the intervals begin at time 0 and end at time 40. If no handler is running in a particular interval, then NONE should be entered as the device name. All intervals are consecutive such that the end time of one interval is the start time of the next interval. If there are unused rows in the table, then NONE should be entered for all three fields.

For the following question, assume that the interrupt requests arrive as shown in the execution timeline below.



Specify the execution times of the A, B, and C handlers assuming a **strong** priority system with the priorities C > B > A. Fill in the table below using the same conventions as the sample table shown above. Remember to show the complete execution (all 11 time units) for each handler.

Strong priority execution timeline:

Interval start time	Interval end time	Device
0		

Submit

Calculator

LE18.4.2: Strong Priorities

0.0/1.0 point (ungraded)

Three devices, D1, D2, and D3, are serviced by interrupt handlers in an interrupt system that may choose to use either strong (preemptive) priorities or weak (non-preemptive) priorities. Their service times and maximum request rates are shown in the following table:

Device	Service time	Interrupt frequency	Deadline
D1	400us	1/(800us)	800us
D2	250us	1/(1000us)	300us
D3	100us	1/(1000us)	400us

Service time indicates how long it takes to run the interrupt handler for each device. The maximum time allowed to elapse between an interrupt request and the end of the execution of the interrupt handler is indicated by the deadline.

1. Assuming each device was interrupting at its maximum rate, what fraction of the CPU time would be used to service **D1**? What fraction of the CPU time would be left for background computing? Answer in percent. **Enter your response as "N%".**

   Give PERCENTAGE of CPU used to service D1:

   Give PERCENTAGE of unused CPU:

2. Can the requirements given in the table above be met using a **strong** priority ordering among the interrupt requests? If so give priority ordering for D1, D2, D3 or select device(s) whose deadlines cannot be met. Select all that apply.

   Strong priority ordering or list device(s) with missed deadlines:

   - □ D1 > D2
   - □ D1 > D3
   - □ D2 > D1
   - □ D2 > D3
   - □ D3 > D1
   - □ D3 > D2
   - □ D1 misses deadline
   - □ D2 misses deadline
   - □ D3 misses deadline

Submit

Discussion

Hide Discus

Calculator

Topic: 18. Devices and Interrupts / LE18.4

Add a Post

Show all posts ⌄ by recent activity ⌄

☑ Misleading explanation in LE18.4.1 part B? 5
> In part A, we determined that given the frequency and service times of these interrupts, there is enough time to schedule them all ...

[STAFF] minor typo: there are 2 LE18.4.1 questions in this section. 2
One more for the list with small typo's. :-)

‹ Previous Next ›

edX

About
Affiliates
edX for Business
Open edX
Careers
News

Legal

Terms of Service & Honor Code
Privacy Policy
Accessibility Policy
Trademark Policy
Sitemap

Connect

Blog
Contact Us
Help Center
Media Kit
Donate

GET IT ON Google play

Download on the App Store

Calculator