

CS452/552 - Real-Time Operating Systems

Lab Assignment #8

Spring 2016

The purpose of this assignment is to add event queues and the associated support routines to the UIKAPI. This is the last piece of the UIK we will implement - yeah!

A common requirement is for a task to wait until an "event" occurs. An event is usually some sort of externally-triggered operation, such as an I/O device requiring service or a timer going off. One possibility is for an interrupt service routine to set an event flag, then allow tasks to wait until that flag is set. Once the flag is set, a task that is waiting on that flag can become ready.

You are to implement the following event functions:

- `void UIKAssocEvent(int Event);` - will cause the task to become "associated" with the flag(s) specified by `Event`. `Event` is a value that specifies the bits representing one or more events that the task should be associated with. The task should then block until the event occurs.
- `void RaiseEvent(int Event);` - This allows a task or ISR to cause an event or events to occur. Note that since this task can be callable from an ISR, it is important for it to be efficient and never cause undue blocking.
- `void UIKDisassocEvent(int Event);` - causes the task to no longer be associated with the event. In other words, the task will not be scheduled when the event occurs.