

# CS452/552 - Real-Time Operating Systems

## Lab Assignment #7

### Spring 2016

The purpose of this assignment is to add semaphores, and the initial part of event queues to the UIKAPI.

As discussed in class, semaphores are an important capability in any operating system, and are especially important in an RTOS. Therefore, you are to implement a semaphore mechanism. At a minimum, you should implement three functions:

- `UIKSemCreate( ... );`
- `UIKSemPend( ... );`
- `UIKSemPost( ... );`

Additionally, it is sometimes useful to have a means for checking the value of the semaphore (perhaps called `UIKSemValue( ... );`);

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, the task can become ready. One initial capability associated with events is for a task to delay itself for a period of time, expressed in clock ticks:

```
UIKTask Delay(int16 count);
```

In this assignment, you are to implement the above functions. In addition, design an application that demonstrates the proper operation of your functions.