# Chapter 3: Using the WICED Real Time Operating System Abstraction Layer

## Objective

After completing chapter 3 you will have a fundamental understanding of the role of the WICED RTOS in building WICED projects. You will be able to use the WICED RTOS abstraction layer to create and use threads, semaphores, mutex, queues, and timers.

## Time: 2 Hours

## Fundamentals

### RTOS & Co-operative multitasking

#### Cyclic dependencies

#### Resource conflicts (sharing memory, sharing peripherals)

#### Interprocess communication

### WICED RTOS Abstraction Layer

### Thread

### Mutex

### Semaphore

### Queue

### Timer

## Exercise(s)

### (thread) Create a blinking led thread

### (semaphore) Button press in main thread -> semaphore to lock led thread

### (mutex) make a function to lock printing

### (queues) send a message to say # of times to blink

### (timers) make a blinking led based on timer

### (debugging) ?