

## Exercise 10 Event Flags

In this exercise we will look at the configuration of an event Flag object and use it to synchronise the activity of several threads

**In the Pack Installer select “Ex 10 Event Flags” and copy it to your tutorial directory.**

### Open main.c

The code in main.c creates an event flag object and instantiates it in appMain().

```
static const osEventFlagsAttr_t EventFlagAttr_LED = {  
    .name = "LED_Events",  
};  
  
void app_main (void *argument)  
{  
    LED_Initialize();  
    EventFlag_LED = osEventFlagsNew(&EventFlagAttr_LED);
```

The code then creates three threads. Two of the threads wait for an event flag to be set ;

```
_NO_RETURN void led_Thread1 (void *argument) {  
for (;;) {  
osEventFlagsWait (EventFlag_LED,0x01,osFlagsWaitAny,osWaitForever);  
LED_On(1);  
  
_NO_RETURN void led_Thread2 (void *argument) {  
for (;;) {  
    osEventFlagsWait (EventFlag_LED,0x01,osFlagsWaitAny,osWaitForever);  
    LED_On(2);
```

The remaining thread is used to set the flag;

```
__NO_RETURN void led_Thread3 (void *argument) {  
for (;;) {  
    osEventFlagsSet (EventFlag_LED,0x01);  
    LED_On(3);
```

### Build the code

### Start the debugger and run the code

### Observe the activity of the LED's

Why does the code not run as expected?

When the event flag is set one of the waiting threads will wake up and clear the flag. The second waiting thread is not triggered. Each thread should be waiting on a separate Event flag within the event flag object.

Change the code so that the waiting threads are waiting on separate flags and the remaining thread sets both flags

```
_NO_RETURN void led_Thread1 (void *argument) {  
    for (;;) {  
        osEventFlagsWait (EventFlag_LED,0x01,osFlagsWaitAny,osWaitForever);  
        LED_On(1);  
    }  
  
    _NO_RETURN void led_Thread2 (void *argument) {  
        for (;;) {  
            osEventFlagsWait (EventFlag_LED,0x02,osFlagsWaitAny,osWaitForever);  
            LED_On(2);  
        }  
    }  
}
```

The remaining thread is used to set both flags;

```
__NO_RETURN void led_Thread3 (void *argument) {  
    for (;;) {  
        osEventFlagsSet (EventFlag_LED,0x03);  
        LED_On(3);  
    }  
}
```