

Add Swi to Task Example

The first screenshot shows the 'SYS/BIOS - System Overview' window. A red arrow points to the 'Swi' button in the 'Threads' section, with the label 'click'.

The second screenshot shows the 'SYS/BIOS - Scheduling - Swi - Module Settings' window. A red arrow points to the checkbox 'Add the software interrupt threads module to my configuration', with the label 'click'.

The third screenshot shows the 'SYS/BIOS - Scheduling - Swi - Instance Settings' window. A red circle highlights the 'Required Settings' section, and a red arrow points to it with the label 'edit'.

```

/*
 * 28x specific Task example, with Swi also
 */
#include <xdc/std.h>
#include <xdc/runtime/Log.h>
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/knl/Task.h>
#include <ti/sysbios/knl/Semaphore.h>
#include <ti/sysbios/knl/Swi.h> ← add this line

/* Semaphore handle defined in task.cfg */
extern const Semaphore_Handle mySem;

/* Swi handle defined in swi.cfg */
extern const Swi_Handle mySwi; ← add this line

/* Counter incremented by timer interrupt */
volatile UInt tickCount = 0;

/*
 * ===== main =====
 */
Int main()
{
    /*
     * Print "Hello world" to a log buffer.
     */
    Log_info0("Hello world\n");

    /*
     * Start BIOS.

```

```

    * Begins task scheduling.
    */
    BIOS_start();    /* does not return */
    return(0);
}

/*
 * ===== myTickFxn =====
 * Timer ISR function that posts a Swi
 */
Void myTickFxn(UArg arg)
{
    tickCount += 1;    /* increment the counter */

    /* every 10 timer interrupts post the Swi */
    if ((tickCount % 10) == 0) {
        Swi_post(mySwi);
    }
}

/*
 * ===== myTaskFxn =====
 * Task function that pends on a semaphore until 10 ticks have
 * expired.
 */
Void myTaskFxn(Void)
{
    /*
     * Do this forever
     */
    while (TRUE) {
        /*
         * Pend on "mySem" until swi0 says
         * it's time to do something.
         */
        Semaphore_pend(mySem, BIOS_WAIT_FOREVER);

        /*
         * Print the current value of tickCount to a log buffer.
         */
        Log_info1("via SWI, 10 ticks. Tick Count = %d\n", tickCount);
    }
}

Void mySwiFxn(Void)
{
    Semaphore_post(mySem);
}

```

Can add more Swi's to project by graphically instantiating more and adding function code for each.