



PEC 3 – Prueba de Evaluación Continuada

Presentación

En esta tercera PEC se amplían los conocimientos teóricos sobre sistemas empuotrados, tanto a nivel hardware como a nivel software, y se sigue con el desarrollo del sistema empuotrado de ejemplo.

Competencias

Propias

- Conocer las aplicaciones y la metodología de desarrollo de les sistemas empuotrados
- Conocer la arquitectura hardware de un sistema encastado
- Conocer los modelos de desarrollo de software de un sistema encastado
- Saber desarrollar una aplicación básica que haga uso de las funcionalidades mínimas de un sistema encastado
- Conocer las herramientas mas utilizadas para desarrollar y evaluar una aplicación para un sistema encastado

Transversales

- Capacidad de comunicación en lengua extranjera.
- Capacidad de adaptarse a nuevas tecnologías y entornos.

Objetivos

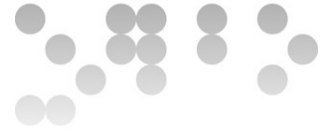
Los objetivos de esta PEC son dos. Por un lado, ampliar los conocimientos sobre los aspectos teóricos de la asignatura, tanto a nivel hardware como a nivel del sistema operativo. Por otro lado, empezar a trabajar los aspectos mas prácticos de la asignatura, incluyendo el modulo de desarrollo de NXP basado en la arquitectura ARM Cortex-M3, el módulo de comunicaciones WiFly basado en 802.11g y el sistema operativo FreeRTOS.

Cuestión 1

Qué es el consumo dinámico y estático de un microcontrolador o microprocesador? De que depende dicho consumo (frecuencia, tamaño de los transistores, etc.). Como se puede reducir el consumo estático? Y el consumo dinámico? Busca información en el *datasheet* del microcontrolador que utilizamos en la asignatura.

Cuestión 2

En esta segunda cuestión empezaremos a utilizar el modulo WiFly para enviar datos vía Wi-Fi (IEEE 802.11) a Internet. El primer paso es conectar el



conversor USB-Serie con el módulo WiFly conectando los pines de TX y RX, así como alimentación y tierra. De este modo será posible enviar datos desde el PC a la web [Arp@Lab](http://cv.uoc.edu/webapps/xwiki/wiki/matembeddedssystemslabhome/view/Material/WebCharts) (<http://cv.uoc.edu/webapps/xwiki/wiki/matembeddedssystemslabhome/view/Material/WebCharts>) través de un programa terminal.

Se pide escribir los comandos necesarios para enviar datos a la web [Arp@Lab](http://cv.uoc.edu/webapps/xwiki/wiki/matembeddedssystemslabhome/view/Material/WebCharts). Importante: no es necesario que especifiquéis vuestra contraseña de la red WiFi!

Cuestión 3

En esta cuestión añadiremos a nuestro *workspace* un conjunto de ejemplos para gestionar diferentes periféricos del microcontrolador, por ejemplo *timers*. Seguid las instrucciones que encontraréis en:

<http://www.lpcware.com/content/project/lpcopen-platform-nxp-lpc-microcontrollers/lpcopen-v200-quickstart-guides/lpcopen-1>

En particular el fichero zip hace referencia a <Directorio instalacion lpcxpresso>\lpcxpresso\Examples\LPCOpen\lpcopen_2_10_lpcxpresso_nxp_lpcxpresso_1769.zip.

Cogiendo como base el proyecto *freertos_blinky*, modificadlo para que:

a) El debug se realice a través de la consola del lpcxpresso tal como se hacía en la segunda PEC:

- Project Properties → C/C++ Build → Settings → MCU Linker → Managed Linker Scripts → Library escaped NewlibNano (semihost)
- Al principio del fichero *freertos_blinky.c* es necesario añadir la siguiente línea antes de los `#include` del principio:

```
#define DEBUG_SEMIHOSTING 1
```

Comprobad que efectivamente el proyecto saca la información de debug por la consola.

b) Añadid a este proyecto el código necesario para leer del ADC (Conversor Analógico-Digital) del microcontrolador. Para eso os podéis basar en el proyecto de ejemplo *periph_adc*.

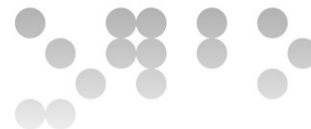
c) Desarrollad un código de ejemplo donde haya dos tareas que se comuniquen por una cola. La primera tarea debe leer un valor del ADC cada segundo y enviarla a la cola. La segunda tarea debe leer de la cola e imprimir por la consola de debug el valor leído. Podéis comprobar que estáis leyendo valores reales conectando el pin correspondiente al canal del ADC de manera alternativa a GND (valor 0) y a VCC (valor máximo).

Recursos

Para desarrollar correctamente esta PEC son necesarios los siguientes recursos:

Básicos

- Material didáctico 2, 3 y 4.



- Página web de FreeRTOS: <http://www.freertos.org/>

Complementarios

- Wiki de la asignatura <http://cv.uoc.edu/webapps/xwiki/wiki/matembeddedsystemslabhome/view/Material/IniciCortexM3>
- <http://www.lpcware.com/content/faq/lpcopen-debug-inputoutput-support>
- <http://www.lpcware.com/content/faq/lpcxpresso/library-variants>

Criterios de valoración

Se valorará la precisión y claridad de las respuestas a las cuestiones 1 y 2, y del correcto funcionamiento de la cuestión 3. La evaluación será:

- La cuestión 1 cuenta el 25% de la nota final de la PEC.
- La cuestión 2 cuenta el 35% de la nota final de la PEC.
- La cuestión 3 cuenta el 40% de la nota final de la PEC.

Para cuestiones sobre el enunciado, debéis dirigiros al consultor responsable de vuestra aula.

Formato y fecha de entrega

La entrega se realizara con un fichero compatible con ODT, PDF o RTF. Es necesario adjuntar el fichero en un mensaje dirigido al buzón de entrega de actividades.

El nombre del fichero tiene que ser Apellido1Apellido2Nombre_EXC_PEC1 con la extensión que corresponda, por ejemplo pdf.

Para ejercicios con múltiples documentos (por ejemplo, ficheros de configuración), es necesario comprimir todos los ficheros en un único documento comprimido mediante el formato zip.

La fecha de entrega de la PEC3 es el 2 de diciembre de 2014 a las 23:59h.