



EXAMEN DE PROGRAMACIÓN DE SISTEMAS Y DISPOSITIVOS

CURSO 2021-22, PRIMERA CONVOCATORIA, 19 DE ENERO DE 2022

En la realización de los ejercicios se considerarán implementadas y, por tanto, podrán usarse todas las funciones y macros públicas incluidas en el BSP desarrollado en los laboratorios. No podrá usarse ninguna función perteneciente a bibliotecas estándar de C. Si fuera necesaria alguna función auxiliar adicional, deberá ser codificada explícitamente. Se usará únicamente aritmética entera.

La página Web de la asignatura es <https://web.fdi.ucm.es/profesor/mendias/PSyD/PSyD.html>

1. **(1 punto)** En una arquitectura tipo cola de funciones se desean ejecutar cooperativamente 3 tareas periódicas `Task1()`, `Task2()` y `Task3()` con periodos respectivos de activación de 10, 20 y 30 ms. Codifique en C una aplicación que esté formada por 2 hebras: una hebra en *foreground* encargada de encolar periódicamente las tareas en los instantes requeridos; y una hebra en *background* (`main`) encargada de inicializar el sistema, instalar la hebra en *foreground* como RTI por interrupción periódica del `timer0` e indefinidamente ejecutar las tareas que se vayan encolando. Supónganse implementadas tanto las tareas, como las funciones para la gestión de la cola de funciones usadas en los laboratorios.

2. **(0.5 puntos)** Codificar en C una función que, usando una fuente 8×16 , escriba sobre el LCD de la placa de prototipado S3CEV40 un carácter a partir del pixel (x,y) al revés. Es decir, para que pudiera leerse correctamente al hacer un giro de 180° en la pantalla. El prototipo de la función será:

```
void lcd_putchar_upsidedown( uint16 x, uint16 y, uint8 color, char ch );
```

3. **(1 punto)** Codificar en C una función que realice o restaure una captura de la pantalla LCD de la placa de prototipado S3CEV40 utilizando el controlador de DMA ZDMA0. La función tendrá dos argumentos: el primero indicará si debe realizarse la captura (valor 1) o debe restaurarse (valor 0); el segundo será puntero al buffer en donde debe almacenarse o está almacenada la captura. El prototipo de la función será:

```
void lcd_snapshot( uint8 action, uint8 *capture );
```

4. **(1.5 puntos)** En una aplicación se desea tener la posibilidad de generar un zumbido a través del audio códec de la placa de prototipado S3CEV40. El zumbido se implementará enviando por el interfaz IIS las muestras correspondientes a una onda cuadrada de máxima amplitud y de frecuencia dada. Esta generación de muestras debe ser autónoma para que no bloquee la ejecución de la aplicación, por lo que deberá implementarse por interrupción. Para que sea posible controlar el envío de muestras, deberá codificarse en C una función que configurará el `timer0` para que interrumpa 16000 veces por segundo (la frecuencia de muestreo del interfaz IIS) e instalará como rutina de tratamiento otra función (que también deberá codificarse) encargada de enviar en cada momento la muestra que corresponda para generar la onda cuadrada deseada, es decir, muestras que alternaran entre `MAX_INT16` y `-MAX_INT16` cada $16000/(2 * \text{freqHz})$ muestras. Si el argumento es 0, el zumbido deberá cesar. El prototipo de la función será:

```
void iss_buzz( uint16 freqHz );
```