



EXAMEN DE PROGRAMACIÓN DE SISTEMAS Y DISPOSITIVOS

CURSO 2017-18, PRIMERA CONVOCATORIA, 23 DE ENERO DE 2018

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.

- (1.5 puntos)** Codificar en C una aplicación que por encuesta periódica retransmita por la UART0 de la placa de prototipado S3CEV40 los caracteres recibidos por ese mismo interfaz. El microcontrolador S3C44B0X deberá configurarse para que su timer-0 interrumpa cada 20 ms. La aplicación constará de 2 hebras que usarán un *mailbox* para comunicarse: una hebra en *foreground* (`isr_timer0`) encargada de leer el estado de la UART0 y, en caso de detectar la recepción de un carácter, lo lea de la UART0 y lo comunique a la otra hebra; y una hebra en *background* (`main`) encargada de inicializar el sistema, instalar la hebra en *foreground* como RTI del timer-0 e indefinidamente esperar el aviso de la otra hebra para retransmitir por la UART0 los caracteres recibidos. Téngase en cuenta que, para el correcto funcionamiento de la aplicación, ninguna de las hebras debe llamar a la función `uart0_getchar`.
- (1 punto)** Codificar en C una función que espere la presión y depresión de una tecla del *keypad* de la placa de prototipado S3CEV40 mientras sea falso el valor del *flag* booleano cuya dirección se pasa como argumento. Si la función finaliza por presión y depresión de una tecla devolverá su *scancode*, si lo hace porque el *flag* vale cierto devolverá 0xff. Esta función permite que la lectura del keypad por encuesta continua no bloquee la respuesta del programa invocante al evento externo señalizado por el flag. El prototipo de la función será:

```
uint8 keypad_flag_getchar( volatile boolean *flag );
```
- (0.75 puntos)** Codificar en C una función que permita reproducir por encuesta (*polling*), a través del controlador de bus IIS del microcontrolador S3C44B0X, el sonido estéreo almacenado en un *buffer* de tamaño en bytes dado a partir de un instante determinado (indicado en décimas de segundo). El audio está almacenado en memoria en el formato habitual: alternando las muestras del canal izquierdo y del derecho a razón de 16 bits/muestra y 16000 muestras/segundo. El prototipo de la función será:

```
void iis_playFrom( int16 *buffer, uint32 length, uint32 decsegs );
```
- (0.75 puntos)** Es posible reducir la frecuencia efectiva de muestreo de un sistema de audio, sin reducir su frecuencia nominal, haciendo un cribado de las muestras transmitidas. El cribado más elemental consiste en tomar, en cada canal, la media aritmética de una serie de muestras consecutivas. Se pide codificar en C una función para realizar por encuesta (*pooling*) el cribado de 1 de cada 2 muestras recibidas a través del controlador de bus IIS del microcontrolador S3C44B0X. El prototipo de la misma será:

```
inline void iis_getSampleHalfRate( int16 *ch1, int16 *ch2 );
```