

Apellidos:.....

Nombre:.....

P1	P2	P3

Duración 4:00 horas

1.- [3 Puntos] Preguntas cortas:

- a) **[15%]** Escriba un trozo de código que configure el SMCLK a una frecuencia de 3'5 MHz, el MCLK a 7Mhz y deje sin alterar el ACLK.
- b) **[15%]** Indique cómo leería el valor del TA0R si b.1) El timer está parado; b.2) El timer está corriendo con fuente ACLK; b.3) ídem con fuente SMCLK.
- c) **[30%]** Haga una tarea cooperativa (sólo la tarea y las definiciones que le afecten) que conmute el led1 del Launchpad (P1.0) cuando se pulse el sw1 (P1.1) sabiendo que el tiempo de rebote del mismo es de 5ms. No use interrupciones. Sólo la tarea cooperativa funcionando por sondeo. Dispone del módulo st desarrollado en prácticas.
- d) **[40%]** Analice el siguiente código e indique qué hace. Señale si sigue o no el convenio de llamada de C. En caso afirmativo, muestre el prototipo. En caso negativo, enumere los parámetros de entrada/salida, ubicación y tipo. Calcule el número de ciclos de ejecución, así como el tiempo total para $f_{MCLK}=1MHz$. Ensamble manualmente el programa y dé en hexadecimal el código máquina.

;	Emulación	CiclosCódigo máquina
;-----		
npi	mov.w r12, r14	;
npi1	mov.b @r12+, r15	;
	tst.b r15	;
	jz npi2	;
	cmp.b r13, r15	;
	jne npi1	;
	dec.w r12	;
	jmp npi3	;
npi2	clr.w r12	;
npi3	sub.w r14, r12	;
	ret	

2.- [3 Puntos] La subrutina bcd2bin usa el convenio de llamada de C y atiende al siguiente prototipo:

```
long int bcd2bin (unsigned long int bcd);
```

bcd2bin convierte el número que se pasa en formato BCD empaquetado de 32 bits (un dígito binario cada cuatro bits), a binario natural (32 bits). En caso de error devuelve -1. Ejemplos de entradas y su salida:

- BCD: 0000 0000 0000 0000 0000 0000 1001 0101 (95 en BCD)
Bin: 0000 0000 0000 0000 0000 0000 0101 1111 (95 en binario)
- BCD: 0000 0000 0000 0000 0000 0000 1011 0000
Bin: 1111 1111 1111 1111 1111 1111 1111 1111 (Error).

- a) Explique con palabras el algoritmo que va a emplear.
- b) Escriba la subrutina en ensamblador.

3.- [4 Puntos] Considere el circuito de la figura. Se trata de un sistema de luces con encendido/apagado y control de intensidad en una sola tecla con rebotes. Cada vez que se pulsa momentaneamente el switch, se conmuta el estado del led. Si se actúa sobre el switch al menos un cuarto de segundo, se entra en el modo de cambio de intensidad hasta que es liberado. La misma varía entre el 25% y el 100% a un ritmo del 100%/2s. Cada vez que se llega a los límites, la variación de brillo cambia de sentido. En el momento de la liberación del switch se paraliza el cambio de intensidad de la luz y se memoriza para posteriores encendidos/apagados. Suponga inicialmente el led apagado, con un brillo memorizado del 100%.

Haga un programa de mínimo consumo en ensamblador del MSP430 que gestione con el TA0 y por interrupciones este comportamiento.

Notas:

- Use una frecuencia de 128 Hz y 32 niveles de brillo.
- El pulsador tiene un tiempo de rebote de 10ms.
- Si la pulsación es menor al tiempo de rebote, se ignorará la misma. Si el tiempo de pulsación es inferior a 0'25s, se considerará una pulsación momentánea y se procederá a conmutar el estado de la luz. Si es mayor, se entra en modo de cambio de intensidad (si el led estaba apagado, se enciende para proceder al ajuste de la intensidad).
- Considere el perro guardián desactivado, los puertos desbloqueados, pila inicializada y LFXT en marcha con cristal de 32768Hz.
- La función primaria de P1.0 es TA0.1 (CCI1A para capturas y Out1 para salidas).
- La función primaria de P1.1 es TA0.2 (CCI2A para capturas y Out2 para salidas).

