

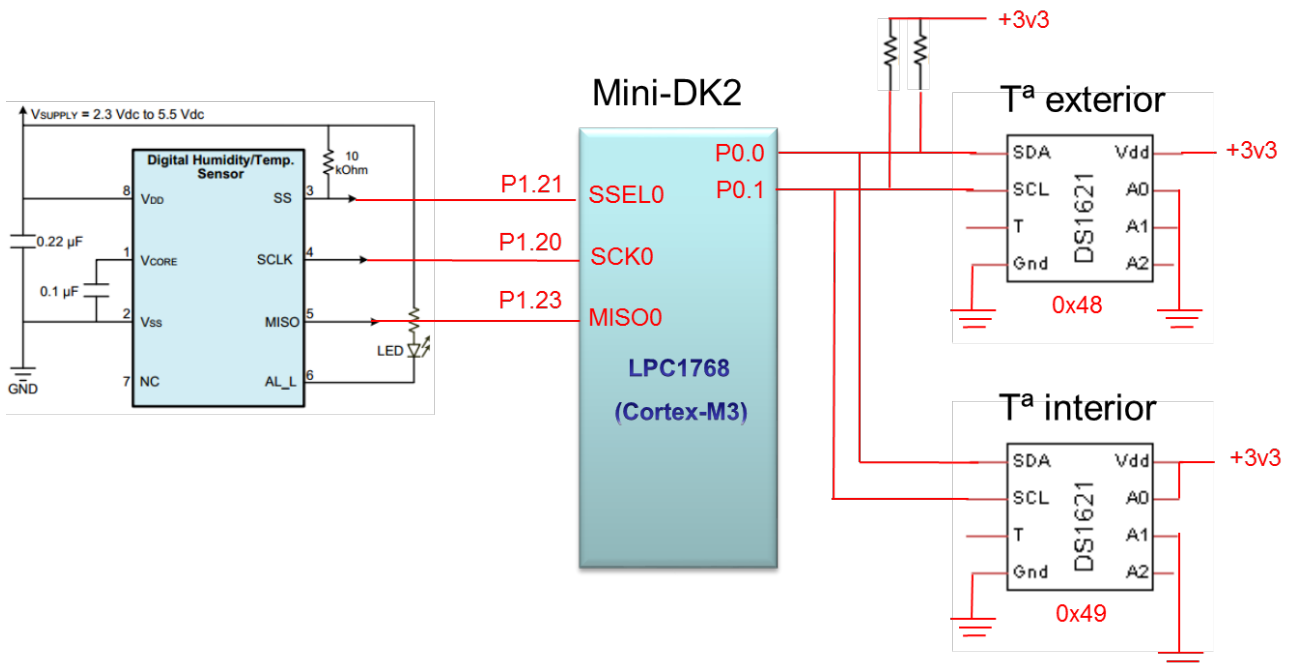


ASIGNATURA:	SISTEMAS ELECTRÓNICOS DIGITALES AVANZADOS	FECHA:	29-04-2016
APELLIDOS:	SOLUCIÓN		Nº lista
NOMBRE:		DNI:	

PRUEBA DE EVALUACIÓN INTERMEDIA 2

CUESTIÓN 1

Complete el conexionado de los sensores de temperatura y humedad de la figura y escriba la función de configuración de la temperatura máxima y mínima del termostato interno del DS1621 interior. Considere estos valores como enteros (T^a con precisión de 1 grado)



```
void set_termostato(uint8_t TH, uint8_t TL)
{
    I2CSendAddr(0x49,0);
    I2CSendByte(0xA1);           //comando Acceso TH
    I2CSendByte(TH);
    I2CSendStop();
    I2CSendAddr(0x49,0);
    I2CSendByte(0xA2);           //comando acceso TL
    I2CSendByte(TL);
    I2CSendStop();
    I2CDelay_50ms();             // Escritura EEPROM
}
```

CUESTIÓN 2

En un sistema de tiempo real basado en el LPC1768 funcionando con una frecuencia interna de 100MHz es necesario realizar las tareas que se indican a continuación. Indique, para cada una de las tareas, el **subsistema interno del microcontrolador que utilizaría** y el **valor de los parámetros T (Periodo Mínimo de interrupción) y D (Plazo Límite de Finalización)** de cada una de las tareas.

- Medir la frecuencia de una señal cuadrada cuyo valor puede oscilar de 10Hz a 10kHz.
- Medir el ciclo de trabajo de una señal cuadrada de 1kHz de frecuencia. El ciclo de trabajo oscilará entre un 10% y un 90%.
- Se desea enviar una señal digital a 100 baudios modulada en FSK (1200Hz y 2200Hz).
- Recibir datos por la UART0 a 19200 baudios.
- Muestrear **continuamente** una señal de audio a 8 kHz utilizando un único canal del DMA en bloques de 1024 muestras. El procesamiento de las muestras debe realizarse en bloques de 1024 muestras.
- Generar dos señales PWM para controlar la velocidad de dos motores con una frecuencia de 10kHz.

Apartado	T	D
a	100us	100us
b	100us	100us
c.1	227,27us	227,27us
c.2	10ms	10ms
d	520us	520us

Apartado	T	D
e	128ms	125us
f1 y f2	-	-

Para medir la frecuencia que se solicita en el **apartado a**, se utilizaría por ejemplo la captura CAP 0.0 del Timer 0 configurados por flanco de bajada. El caso peor sería para el caso a 10kHz. Se produciría una interrupción cada 100 us y, como el tiempo habría sido capturado, se dispondría de todo el tiempo entre capturas para poder leer el valor de la medida ($D = T$).

Si se decidiera utilizar para medir el periodo una interrupción externa configurada por flanco, y dentro de la interrupción se leyera el valor de un temporizador (sin captura hardware), el tiempo que se tardara en procesar la interrupción influiría en la precisión de la medida por lo que normalmente D debería en este caso elegirse mucho menor que T.

Para medir el ciclo de trabajo del **apartado b** se puede configurar, por ejemplo, la captura del CAP 0.2 por flanco de subida y, en la interrupción asociada, configurarlo por flanco de bajada y así repetidamente. El tiempo mínimo entre interrupciones sería del 10% de 1ms que serían 100us. Como el tiempo habría sido capturado, se dispondría de todo el tiempo entre capturas para poder leer el valor de la medida ($D = T=100us$).

Otra opción sería utilizar dos entradas de captura de un mismo temporizador, una configurada por flanco de subida y otra por flanco de bajada. Se configura sólo una de las dos como interrupción y en la rutina de atención se calcularía el periodo y el ciclo de trabajo con el valor de las dos capturas, almacenando el valor de la última captura para utilizarlo en el cálculo siguiente. En este caso no influye la limitación del 10% y 90% siendo válida también para ciclos de trabajo mucho más cercanos al 0% y al 100%. ($D = T=1ms$).

La señal FSK del **apartado c** se podría generar con el Temporizador 1 utilizando el MAT1.0. Para generar la señal cuadradas, el periodo de interrupción será la mitad del periodo de la señal a generar ya que hay que temporizar el tiempo entre cada conmutación. $T = (1/f)/2$. El peor caso sería cuando se enviara la frecuencia de 2200Hz que debería interrumpir cada $(1/2200)/2 = 227,27us$. Al utilizarse registros de comparación, en el caso de las dos interrupciones $D=T= 227,27us$.

La frecuencia de transmisión se podría generar con el Match 1 del mismo temporizador que interrumpiría cada 1/100 segundos ($D=T=10ms$).

En el **apartado d**, se utilizaría una de las UART disponibles y el tiempo entre interrupciones sería:

$$T = \frac{10 \text{ bits}}{19200 \text{ bits/seg}} = 520 \text{ us}$$

Se dispone de todo el tiempo entre interrupciones para leer el dato por lo que $D = T=520us$.

En el **apartado e**, se utilizaría un temporizador configurado para que cada 125us dispare la conversión del ADC (por ejemplo se podría utilizar el MAT0.1 asociado al TIM0). También habría que configurar un canal de DMA para ir guardando las 1024 muestras en memoria. De esta forma habría una interrupción asociada al DMA cada 1024 muestras ($1024 * 125 \text{ us} = 128 \text{ ms}$).

Debido a que las muestras deben ser procesadas en bloques y sólo se puede utilizar un canal de DMA, la rutina de atención del DMA que extraiga las muestras del buffer o que configure otra zona de memoria donde guardar los datos, debe realizarse antes de la siguiente conversión, es decir que el tiempo máximo en que debe ser atendida la interrupción sería de 125us. En este caso $T = 128 \text{ ms}$ y $D=125\text{us}$.

En el caso de las señales PWM del **apartado f**, se utilizaría el módulo PWM que no necesita una interrupción asociada, por lo que no influye en la ejecutabilidad.

CUESTIÓN 2.

Suponiendo que un sistema basado en el LPC1768 tiene cuatro tareas con los parámetros que se indican en la tabla y teniendo en cuenta que existe una **región crítica** en la Tarea 3 de **5ms**, y otra en el programa principal de **2ms**

Tarea	Prioridad	Subprioridad	C	T	D
Tarea 1	0	0	3	10	10
Tarea 2	1	0	10	40	40
Tarea 3	1	1	10	70	40
Tarea 4	2	1	15	100	80

Nota: Unidades en ms

Se pide:

- Analice si el sistema es ejecutable
- Si el sistema es ejecutable:
 - Calcule la duración máxima de la región crítica del programa principal que haría que el sistema dejara de ser ejecutable.
- Si el sistema no es ejecutable:
 - Calcule el valor del periodo mínimo de ejecución de la Tarea 2 que haría el sistema ejecutable.

- a) La Tarea 1 puede interrumpir a las tareas 2, 3 y 4. La Tarea 2 no puede interrumpir a la Tarea 3 pero sí a la 4. La Tarea 3 puede interrumpir a la 4.

El sistema es ejecutable cuando, para todas sus tareas, el tiempo de respuesta es menor que el plazo límite de finalización (deadline).

Tarea 1:

$$R_1 = C_1 + B_1 + I_1 = 3 + \max \left\{ \begin{matrix} RC_{Tarea\ 3} \\ RC_{ProgPr} \end{matrix} \right\} + 0 = 3 + 5 + 0 = 8\ ms \leq D_1 = 10\ ms$$

Tarea 2:

$$R_2 = C_2 + B_2 + I_2 = C_2 + \max \left\{ \begin{matrix} C_{Tarea\ 3} \\ RC_{ProgPr} \end{matrix} \right\} + C_1 \left\lceil \frac{R_2}{T_1} \right\rceil = 10 + 10 + 3 \left\lceil \frac{R_2}{10} \right\rceil$$

$$w^0 = 23; w^1 = 20 + 3 \left\lceil \frac{23}{10} \right\rceil = 29; w^2 = 20 + 3 \left\lceil \frac{29}{10} \right\rceil = 29;$$

$$R_2 = 29\ us \leq D_2 = 40\ ms$$

Tarea 3:

$$R_3 = C_3 + B_3 + I_3 = C_3 + RC_{ProgPr} + C_1 \left\lceil \frac{R_3}{T_1} \right\rceil + C_2 \left\lceil \frac{R_3 - C_3}{T_2} \right\rceil = 10 + 2 + 3 \left\lceil \frac{R_3}{10} \right\rceil + 10 \left\lceil \frac{R_3 - 10}{40} \right\rceil$$

$$w^0 = 25; w^1 = 12 + 3 \left\lceil \frac{25}{10} \right\rceil + 10 \left\lceil \frac{15}{40} \right\rceil = 31; w^2 = 12 + 3 \left\lceil \frac{31}{10} \right\rceil + 10 \left\lceil \frac{31 - 10}{40} \right\rceil = 34;$$

$$w^3 = 12 + 3 \left\lceil \frac{34}{10} \right\rceil + 10 \left\lceil \frac{34 - 10}{40} \right\rceil = 34;$$

$$R_3 = 34\ ms < D_3 = 40\ ms$$

Tarea 4:

$$R_4 = C_4 + B_4 + I_4 = C_4 + RC_{ProgPr} + C_1 \left\lceil \frac{R_4}{T_1} \right\rceil + C_2 \left\lceil \frac{R_4}{T_2} \right\rceil + C_3 \left\lceil \frac{R_4}{T_3} \right\rceil = 15 + 2 + 3 \left\lceil \frac{R_4}{10} \right\rceil + 10 \left\lceil \frac{R_4}{40} \right\rceil + 10 \left\lceil \frac{R_4}{70} \right\rceil$$

$$w^0 = 40; w^1 = 17 + 3 \left\lceil \frac{40}{10} \right\rceil + 10 \left\lceil \frac{40}{40} \right\rceil + 10 \left\lceil \frac{40}{70} \right\rceil = 49; w^2 = 17 + 3 \left\lceil \frac{49}{10} \right\rceil + 10 \left\lceil \frac{49}{40} \right\rceil + 10 \left\lceil \frac{49}{70} \right\rceil = 62;$$

$$w^3 = 17 + 3 \left\lceil \frac{62}{10} \right\rceil + 10 \left\lceil \frac{62}{40} \right\rceil + 10 \left\lceil \frac{62}{70} \right\rceil = 68; w^4 = 17 + 3 \left\lceil \frac{68}{10} \right\rceil + 10 \left\lceil \frac{68}{40} \right\rceil + 10 \left\lceil \frac{68}{70} \right\rceil = 68;$$

$$R_4 = 68\ ms < D_4 = 80\ ms \rightarrow \text{El sistema ES ejecutable}$$

b) Análisis de la RC_{ProgPr} que hace el sistema no ejecutable.

En la Tarea 1 hay un margen de 2ms, la RC_{ProgPr} podría aumentar hasta 7ms

En la Tarea 2, el bloqueo es debido a la duración de la Tarea 3 que es mayor que 7ms por lo que este valor de la RC_{ProgPr} sería aceptable.

En la Tarea 3,

$$R_3 = C_3 + B_3 + I_3 = C_3 + RC_{ProgPr} + C_1 \left\lceil \frac{R_3}{T_1} \right\rceil + C_2 \left\lceil \frac{R_3 - C_3}{T_2} \right\rceil = 10 + 7 + 3 \left\lceil \frac{R_3}{10} \right\rceil + 10 \left\lceil \frac{R_3 - 10}{40} \right\rceil$$

$$w^0 = 30; w^1 = 17 + 3 \left\lceil \frac{30}{10} \right\rceil + 10 \left\lceil \frac{20}{40} \right\rceil = 35; w^2 = 17 + 3 \left\lceil \frac{35}{10} \right\rceil + 10 \left\lceil \frac{25}{40} \right\rceil = 39;$$

$$w^3 = 17 + 3 \left\lceil \frac{39}{10} \right\rceil + 10 \left\lceil \frac{39 - 10}{40} \right\rceil = 39;$$

$$R_3 = 39ms < D_3 = 40ms \rightarrow \text{sigue siendo ejecutable}$$

En la Tarea 4,

$$R_4 = C_4 + B_4 + I_4 = C_4 + RC_{ProgPr} + C_1 \left\lceil \frac{R_4}{T_1} \right\rceil + C_2 \left\lceil \frac{R_4}{T_2} \right\rceil + C_3 \left\lceil \frac{R_4}{T_3} \right\rceil = 15 + 7 + 3 \left\lceil \frac{R_4}{10} \right\rceil + 10 \left\lceil \frac{R_4}{40} \right\rceil + 10 \left\lceil \frac{R_4}{70} \right\rceil$$

$$w^0 = 45; w^1 = 22 + 3 \left\lceil \frac{45}{10} \right\rceil + 10 \left\lceil \frac{45}{40} \right\rceil + 10 \left\lceil \frac{45}{70} \right\rceil = 22 + 15 + 20 + 10 = 67;$$

$$w^2 = 22 + 3 \left\lceil \frac{67}{10} \right\rceil + 10 \left\lceil \frac{67}{40} \right\rceil + 10 \left\lceil \frac{67}{70} \right\rceil = 22 + 21 + 20 + 10 = 73;$$

$$w^3 = 22 + 3 \left\lceil \frac{73}{10} \right\rceil + 10 \left\lceil \frac{73}{40} \right\rceil + 10 \left\lceil \frac{73}{70} \right\rceil = 22 + 24 + 20 + 20 = 86; \rightarrow \text{No es ejecutable}$$

Es decir, que una RC_{ProgPr} de 7ms haría sistema no ejecutable, es necesario buscar un valor inferior.

Si nos fijamos en la expresión original de R_4

$$w^3 = 17 + 3 \left\lceil \frac{62}{10} \right\rceil + 10 \left\lceil \frac{62}{40} \right\rceil + 10 \left\lceil \frac{62}{70} \right\rceil = 68; w^4 = 17 + 3 \left\lceil \frac{68}{10} \right\rceil + 10 \left\lceil \frac{68}{40} \right\rceil + 10 \left\lceil \frac{68}{70} \right\rceil = 68;$$

Se puede ver que la R_4 puede aumentar hasta 70 sin que se produzca el incremento de las interferencias. Si R_4 supera ese valor, la interferencia aumenta en 10, lo que hace que R_4 vaya a 80 que está en el límite de la ejecutabilidad. Con una RC de 4ms

$$R_4 = C_4 + B_4 + I_4 = C_4 + RC_{ProgPr} + C_1 \left\lceil \frac{R_4}{T_1} \right\rceil + C_2 \left\lceil \frac{R_4}{T_2} \right\rceil + C_3 \left\lceil \frac{R_4}{T_3} \right\rceil = 15 + 4 + 3 \left\lceil \frac{R_4}{10} \right\rceil + 10 \left\lceil \frac{R_4}{40} \right\rceil + 10 \left\lceil \frac{R_4}{70} \right\rceil$$

$$w^0 = 42; w^1 = 19 + 3 \left\lceil \frac{42}{10} \right\rceil + 10 \left\lceil \frac{42}{40} \right\rceil + 10 \left\lceil \frac{42}{70} \right\rceil = 19 + 15 + 20 + 10 = 64;$$

$$w^2 = 19 + 3 \left\lceil \frac{64}{10} \right\rceil + 10 \left\lceil \frac{64}{40} \right\rceil + 10 \left\lceil \frac{64}{70} \right\rceil = 19 + 21 + 20 + 10 = 70;$$

$$w^3 = 19 + 3 \left\lceil \frac{70}{10} \right\rceil + 10 \left\lceil \frac{70}{40} \right\rceil + 10 \left\lceil \frac{70}{70} \right\rceil = 19 + 21 + 20 + 10 = 70;$$

Si la RC_{ProgPr} fuera superior a 4ms, w^3 sería superior a 70 y esto haría que en la siguiente iteración el resultado fuera superior a 80 haciendo el sistema no ejecutable.



$$RC_{ProgPr} \leq 4ms$$

CUESTIÓN 4.

En una aplicación de monitorización remota de datos, se desea representar gráficamente la temperatura de una habitación. En la Figura 1 se muestra el código HTML correspondiente a la salida por el navegador representada en la Figura 2. Se desea realizar un programa que actualice el valor de la temperatura, así como en el color y el tamaño de la representación gráfica, siendo el máximo de temperatura representable 50°C correspondiente a un ancho de 100 pixels y el mínimo de 0°C, correspondiente a cero pixels. El color debe cambiar de negro para 0°C a rojo brillante para 50°C.

Indique el contenido del fichero `temperatura.cgi` a la que accedería el navegador y el contenido de la función `cgi_func(...)` necesario. Suponga que la temperatura se almacena en una variable global denominada `int temp`.

Nota: en el código HTML el color del dibujo se representa con un número hexadecimal de 24 bits donde los 8 bits de mayor peso corresponden con la cantidad de color rojo (0 indica que la imagen no tiene rojo y 255 indica que tiene todo el rojo posible).

<pre><table style="width: 350px; height: 64px;" border="0"> <tbody> <tr> <td style="width: 100px;">Temperatura:</td> <td style="width: 50px;">50&#186;C</td> <td style="width: 200px;"> <table style="width: 100px; height: 35px;" border="0"> <tbody> <tr> <td style="width: 100%; height: 29px; background-color: #ff0000;"></td> </tr> </tbody> </table> </td> </tr> </tbody> </table></pre>	<p>Temperatura: 50°C</p> 
<pre><table style="width: 350px; height: 64px;" border="0"> <tbody> <tr> <td style="width: 100px;">Temperatura:</td> <td style="width: 50px;">25&#186;C</td> <td style="width: 200px;"> <table style="width: 50px; height: 35px;" border="0"> <tbody> <tr> <td style="width: 100%; height: 29px; background-color: #7f0000;"></td> </tr> </tbody> </table> </td> </tr> </tbody> </table></pre>	<p>Temperatura: 25°C</p> 

Fichero `temperatura.cgi`:

```
t <table style="width: 350px; height: 64px;" border="0">
t <tbody>
t <tr>
t <td style="width: 100px;">Temperatura:</td>
c a <td style="width: 50px;">%d&#186;C</td>
t <td style="width: 200px;">
c b <table style="width: %dpx; height: 35px;" border="0">
t <tbody>
t <tr>
c c <td style="width: 100%; height: 29px; background-color: #%2x0000;"></td>
t </tr>
t </tbody>
t </table>
t </td>
t </tr>
t </tbody>
t </table>
```

CUESTIÓN 5.

Explique qué es un DSP y sus características fundamentales.

CUESTIÓN 6.

Explique en qué consiste el método de arbitración por interrupción en una DUAL-PORT cuya capacidad es de 64Kbytes.