

Cuestión (20 minutos – 2 puntos)

Un microcontrolador de la familia STM32L152RB debe disponer de una memoria de datos de 256M x 16 bits. En esta memoria, la parte más baja es memoria ROM y la parte más alta es memoria RAM. También se sabe que hay 64M de memoria ROM y 192M de memoria RAM.

Para implementar esta memoria se dispone de los siguientes chips:

- Chips ROM de 32M x 16 bits
- Chips RAM de 64M x 8 bits
- Chips RAM de 128M x 16 bits

- a) Indica cuántos chips de cada tipo hacen falta para implementar la memoria.
- b) Dibuja el mapa de memoria del sistema, indicando la primera y última dirección de cada zona en hexadecimal. El diseño del microcontrolador obliga a que el chip de RAM más grande tenga que ir colocado en la parte más alta de la memoria.
- c) Diseña el decodificador de direcciones válido para esta memoria utilizando un decodificador 2:4 con salidas activas por nivel alto y las puertas lógicas AND, OR y NOT que consideres necesarias. Considera que las señales de Chip Select (CS) son activas por nivel bajo.

Problema 1 (60 minutos – 4 puntos)

Se desea instalar en una señal de tráfico luminosa un sistema que controle el brillo de sus LEDs de manera que, cuando la intensidad de luz ambiente sea mayor el brillo de la señal de tráfico también sea mayor y, cuando la luz ambiente baje (por ejemplo, de noche), el brillo de la señal sea menor para no deslumbrar.

Para implementar el prototipo se va a utilizar la placa de desarrollo con el microcontrolador de referencia del curso, una placa LightHz de Mikroelektronika que incorpora un integrado TSL230BR y una placa basada en el integrado A6260. La placa LightHz con el integrado TSL230BR genera una señal digital cuya frecuencia es proporcional a la cantidad de luz ambiente que recibe. El integrado A6260 se utiliza para entregar a los LEDs de la señal de tráfico la corriente necesaria, conectando su señal de entrada EN a una señal PWM, la corriente suministrada a los LEDs es proporcional al ciclo de trabajo de la misma. Al final de este enunciado se da un código de referencia, en base al cual ha de responder a las siguientes preguntas:

1. Represente el esquema del hardware utilizado para la aplicación, indicando claramente qué pines del microcontrolador se conectan a los elementos que componen el sistema (20%)
- 2.- Identifique qué timers se utilizan en este programa. Indique para cada uno de ellos si funciona como TOC, TIC o PWM y cuál es su estado (en marcha o parado) tras la rutina de configuración. (10%)
- 3.- Represente el diagrama de flujo de la función declarada como “void espera (unsigned int segundos);”. ¿Para qué se utiliza esta función en el programa principal? (10%)
- 4.- En este programa ¿se utilizan variables globales? En caso afirmativo diga cuales. Justifique la necesidad de utilizar variables globales. (5%)
- 5.- El programa configura un pin de salida. Identifique cual es este pin y dibuje la forma de onda que genera el microcontrolador en este pin durante los primeros instantes tras el Setup(). (15%)
- 6.- El integrado TSL230BR es un sensor de luz con salida en frecuencia. ¿Cuál es la frecuencia máxima que puede medirse en base al programa? (20%)
- 7.- Suponiendo que la luz ambiente es tal que la frecuencia de la señal digital generada por el sensor de luz es de 100kHz, indique cuál sería el ciclo de trabajo de la señal PWM que se aplica al integrado A6260. (20%)

CÓDIGO PROGRAMA CONTROL BRILLO LEDs

```
#include "Biblioteca_SDM.h"
#include "stm3211xx.h"

#define TMR0_1segundo 1000

void Setup(void);
void espera (unsigned int segundos);
unsigned cuenta = 0;

void EXTI0_IRQHandler(void) {
    if (EXTI->PR!=0) {
        cuenta++;
        EXTI->PR = 0x01;
    }
}

void main(void) {
    Init_SDM();
    Setup();
    while(1) {
        espera(1000);
        cuenta = 0;
        EXTI->PR = 0x01;
        EXTI->IMR |= 0x01;
        espera(2);
        EXTI->IMR &= ~(0x01);
        TIM4->CCR2 = cuenta/8;
    }
}

void Setup(void) {
    GPIOB->MODER|=0x00000001 << (2*7 +1);
    GPIOB->MODER&=~(0x00000001 << (2*7));
    GPIOB->AFR[0] |= (0x02 << (7*4));

    TIM2->CR1 = 0x0000;
    TIM2->CR2 = 0x0000;
    TIM2->SMCR = 0x0000;
    TIM2->PSC = 32000;
    TIM2->CNT = 0;
    TIM2->ARR = 0xFFFF;
    TIM2->CCR1 = TMR0_1segundo;
    TIM2->DIER = 0x0000;
    TIM2->CCMR1 = 0x0000;
    TIM2->CCER = 0x0000;
    TIM2->EGR |= 0x0001;

    GPIOA->MODER &= ~(1 << (0*2 +1));
    GPIOA->MODER &= ~(1 << (0*2));
    EXTI->FTSR |= 0x01;
    EXTI->RTSR &= ~(0x01);
    SYSCFG->EXTICR[0] = 0;
    NVIC->ISER[0] |= (1 << 6);

    TIM4->CR1 = 0x0080;
    TIM4->CR2 = 0x0000;
    TIM4->SMCR = 0x0000;
    TIM4->PSC = 0;
    TIM4->CNT = 0;
    TIM4->ARR = 32000;
    TIM4->DIER = 0x0000;
    TIM4->CCMR1 = 0x6800;
    TIM4->CCER = 0x0010;
    TIM4->CCR2 = 16000;
    TIM4->CR1 |= 0x0001;
    TIM4->EGR |= 0x0001;
    TIM4->SR = 0;
}
```

```
void espera (unsigned int segundos){
    unsigned int t = 0;
    TIM2->SR = 0;
    TIM2->CR1 |= 0x0001;
    while (t < segundos){
        TIM2->CCR1 = TIM2->CNT + TMR0_1segundo;
        while ((TIM2->SR&0x0002)==0);
        TIM2->SR &= ~(0x0002);
        t++;
    }
    TIM2->CR1 &= ~(0x0001);
}
```

Problema 2 (100 minutos – 4 puntos)

Se quiere controlar el sistema de sensado de una cámara limpia para la fabricación de circuitos fotónicos integrados, utilizando el microcontrolador de referencia del curso. Este sistema dispone de un sensor de temperatura, un sensor de humedad y un sensor de fuego y debe permite transmitir la información de estos sensores a un PC instalado en la propia cámara limpia y al puesto de control central que se conecta con la cámara limpia mediante un enlace de fibra óptica de plástico. La conexión con el PC se realizará mediante una comunicación serie asíncrona y el envío de datos al puesto central se realizará utilizando una codificación FSK (Frequency Shift Keying), que se detalla más adelante. El sistema dispone también de un altavoz que emite una alarma sonora en el caso de que se detecte fuego en la cámara.

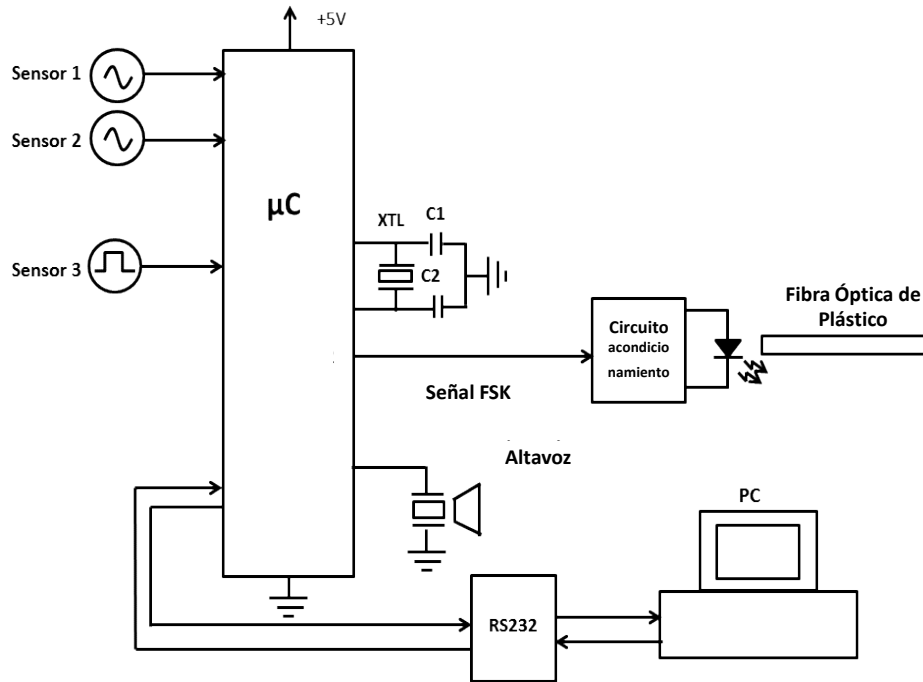


Fig 1.

A continuación se describen más detalladamente las características del sistema

- **El microcontrolador funciona con un oscilador de 8MHz.**
- **Sensores:** Los tres sensores del sistema tienen las siguientes características y prioridad en el sistema de control

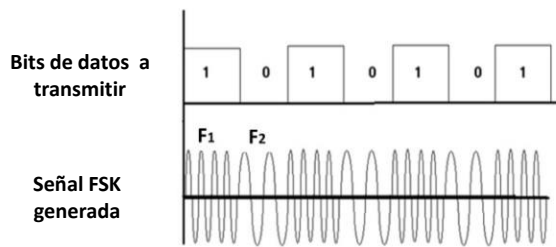
Sensor	Tipo	Resolución	Señal	Rango (V)	Nivel de Prioridad
1	Temperatura (0 – 99 °C)	20 mV / °C	analógica	0 - 2	No
2	Humedad Relativa (0 – 99 %)	20 mV / %	analógica	0 - 2	No
3	Fuego		digital	0 - 5	Alta

El sensor de incendios genera una señal digital a su salida. Cuando detecta fuego en la cámara se produce un flanco de subida en esta señal. La señal permanece a nivel alto mientras el fuego está activo y cuando el fuego se extingue se produce un flanco de bajada en la señal de salida del sensor. Esta señal permanece a nivel bajo mientras no haya fuego en la cámara.

- **Señal de alarma de fuego:** Cuando el sensor de incendios detecta fuego en la cámara el microcontrolador genera una señal cuadrada de 1ms de periodo para activar el altavoz. Cuando el fuego se extingue esta señal también desaparece.
- **La comunicación serie con el PC se realiza a 19200 baudios, 8 bits, con 1 bit de inicio y 1 bit de parada.** Esta comunicación debe permitir enviar y recibir datos del PC.
- **El envío de la información al puesto de control se realiza generando en uno de los pines del microcontrolador, una señal FSK cuya estructura se detalla a continuación.**

El generador FSK se utiliza para generar dos señales de diferentes frecuencias que representan los dos posibles valores binarios: Para transmitir un '1' lógico se genera una señal de una frecuencia f1 (100kHz) durante un tiempo (periodo de bit = 80µs) y para generar un '0' lógico una señal de frecuencia f2 (50kHz). Estas señales pueden ser sinusoidales (como en el dibujo de la fig. 2a) o cuadradas.

La estructura de los datos de los sensores que se envían al puesto de control central cada 2 segundos es la que aparece en la Fig. 2b.



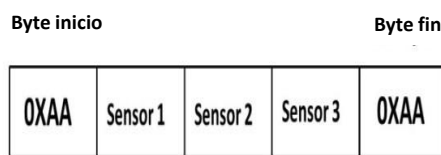
Características en el Sistema de la cámara limpia

- ✓ $f_1 = 100$ kHz (señal cuadrada)
- ✓ $f_2 = 50$ kHz (señal cuadrada)
- ✓ Ciclo de trabajo del 50%
- ✓ Período de bit = 80 μ s.

a

Características del paquete de datos a transmitir al puesto central de control

- El paquete empieza tiene un byte de inicio y uno de fin, ambos codificados con **0XAA**. Todos los bytes del paquete se empiezan a transmitir por el bit LSB.
- La información del sensor digital de incendios (Sensor 3) se transmite con **0XB4** y **0XB6** cuando su salida está a nivel alto y a nivel bajo, respectivamente.



b

Fig 2

El diagrama de flujo que se ha desarrollado para el programa principal es el que se muestra a continuación:

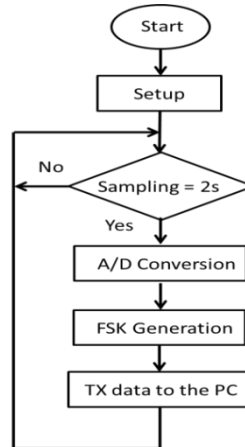


Fig. 3

Se pide:

1. Haga una tabla indicando las tareas que realiza el microcontrolador, los periféricos utilizados y los pines de conexión del microcontrolador (indicando si funcionan como analógicos o digitales y si son de entrada o salida). (20%)
2. Indique como configurar el convertidor A/D y la comunicación serie para esta aplicación. Indique de forma justificada los valores necesarios en todos los registros de configuración. (20%)
3. La rutina "A/D Conversion" que aparece en el diagrama de flujo de la Fig. 3 selecciona el canal analógico de entrada y realiza la conversión A/D de cada uno de los canales. Represente el diagrama de flujo de esta rutina. Considere que los resultados de las conversiones A/D se almacenan en un vector declarado como : char ADresult []; (15%)
4. La rutina "setup" que aparece en el diagrama de flujo de la Fig. 3 realiza la configuración inicial de todos los recursos del microcontrolador. Indique razonadamente qué recurso/s del microcontrolador atenderá mediante interrupción y cuál debe ser el contenido de los registros asociados a la/s interrupción/es en esta rutina de inicialización. (10%)
5. Realice el diagrama de flujo y el código C de la/s rutina/s de atención a la interrupción (20%)
6. Realice el diagrama de flujo de la rutina "FSK Generation" que aparece en el diagrama de flujo de la Fig.3. (15%)