

APELLIDOS : \_\_\_\_\_  
NOMBRE: \_\_\_\_\_ D.N.I.: \_\_\_\_\_

- LEER ATENTAMENTE EL ENUNCIADO.
- RAZONAR CADA UNA DE LAS DECISIONES ADOPTADAS TANTO EN EL HW COMO EN EL SW.
- CUIDAR LA PRESENTACIÓN EVITANDO ESQUEMAS ININTELIGIBLES, LETRA INDESCIFRABLE Y EXPRESIONES MALSONANTES.
- UTILIZAR PARA EL DISEÑO LOS CRITERIOS DE INGENIERÍA (MÁXIMA EFICIENCIA, MÍNIMO COSTO).
- UTILIZAR COMENTARIOS EN LA ELABORACIÓN DEL CÓDIGO SW.

### CUESTIONES:

C1. Comentar la limitación que tiene la familia de microcontroladores PIC referente al anidamiento de subrutinas (0.5 puntos).

C2. Cuando utilizamos el Timer0 como contador del PIC 16F84, realizar un dibujo de su configuración y comentar de donde puede provenir la señal de incremento del contador, en que caso se usa cada fuente y los registros afectados (0.5 puntos).

C3. Comentar las diferencias entre los sistemas de control basados en PLC, en PC y en microcontrolador y en que tipo de aplicaciones se emplea cada solución (0.5 puntos).

C4. Explicar en que consisten y para que se emplean los protocolos de comunicación (0.5 puntos).

### PROBLEMAS:

P1. (5 puntos) Disponemos de un nodo esclavo en un sistema de alarma distribuido formado por un 8051 en el cual se han conectado los siguientes elementos:

- Una salida RS485C para comunicar con nodo central, empleando P1.6 como dirección.
- Tres salidas, que activarán a tres led (rojo en P1.3, amarillo en P1.4 y verde en P1.5).
- Tres entradas en las que hay conectados tres interruptores (en P1.0, P1.1 y P1.2).

Realizar un programa que efectúe la siguiente secuencia:

- Al recibir el comando "Q" por el enlace de comunicación se enciende el led rojo.
- Al recibir el comando "A" por el enlace de comunicación se apague el led rojo.
- Al recibir el comando "W" por el enlace de comunicación se encienda el led amarillo.
- Al recibir el comando "S" por el enlace de comunicación se apague el led amarillo.
- Al recibir el comando "E" por el enlace de comunicación se encienda el led verde.
- Al recibir el comando "D" por el enlace de comunicación se apague el led verde.

- Al recibir el comando “I” por el enlace de comunicación, envíe el estado de los tres terminales de entrada, como un número en código ASCII.
- Al recibir el comando “C” por el enlace de comunicación, haga eco del siguiente byte recibido.

Emplear la interrupción del puerto serie para recibir los comandos de la interfaz de comunicación y la interrupción del Timer0 para muestrear las entradas del sistema cada 500 ms.

Usar el Timer1 en modo autorrecarga como generador de baudios. El pulsador conectado a la patilla de interrupción /INT0 provoca que se envíe al nodo maestro inmediatamente la información de las entradas, sin que fuese necesario una petición por parte del maestro.

El protocolo de comunicación consiste en mandar en secuencia: un byte de comienzo, un byte que indica el código del destinatario, otro que indica el código del remitente y un byte de comando y otro opcional de datos. Se finaliza con un byte de fin :

BYTE de Comienzo (0AH)	Destinatario (Binario)	Remitente (Binario)	BYTE de Instrucción (ASCII)	Datos Opcional (ASCII)	BYTE de Fin (0DH)
------------------------------	---------------------------	------------------------	-----------------------------------	------------------------------	-------------------------

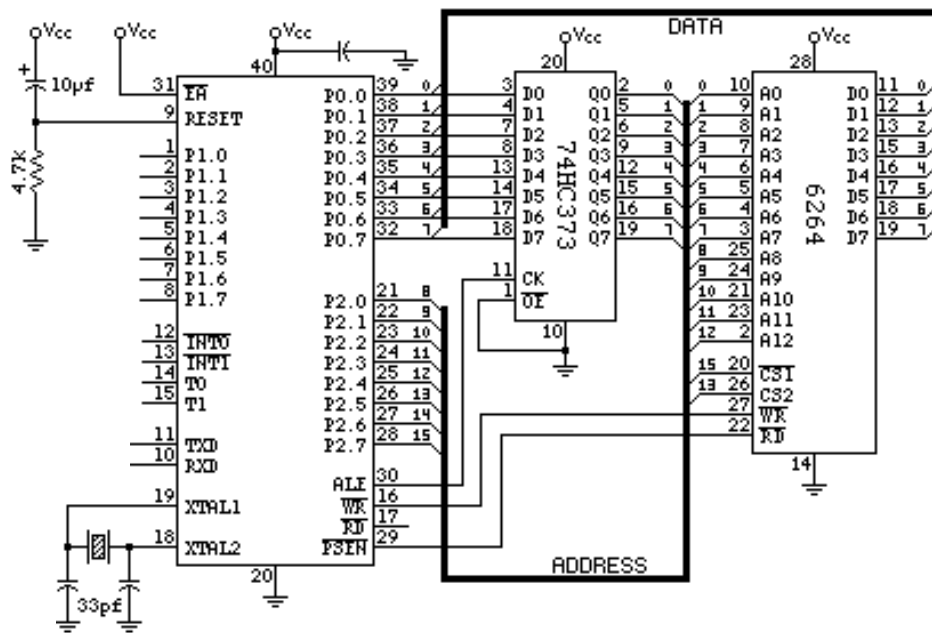
El programa podría comenzar así:

```
;*****
; Programa para la comunicación del bus RS-485
;*****
; Declaración de constantes y de variables
;*****
ID_MOD      EQU  #02H ;Número de identificación de este módulo
ID_Master   EQU  #00H ;Número de identificación del maestro

Orden_in    EQU  10H ;Variable que indica el n° de byte recibido
Haz_Eco     EQU  13H ;Variable que indica la acción de eco
Dato_Eco    EQU  14H ;Variable que contendrá el dato enviado por
el maestro
;*****
; Rutina de Vectorización
;*****
ORG  0H
LJMP  Inicio
ORG  023H      ;Vectorización del puerto de comunicación serie
LJMP  RSI_RS

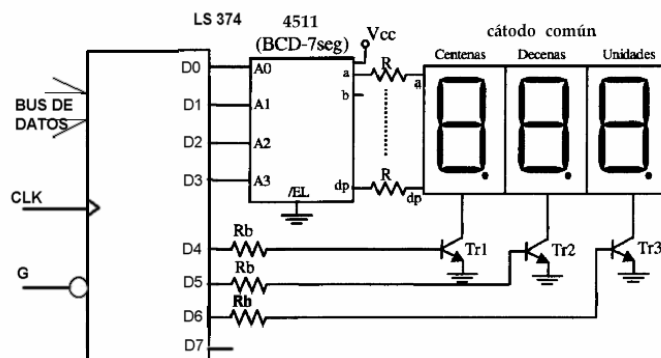
.....
```

P2. El siguiente diagrama esquemático representa un sistema basado en un 8051, donde los números del bus de direcciones representan bits de direcciones.



- a) (1 punto) Añadir al diseño básico anterior un puerto de Salida (registro LS374) conectado al bus de datos y direcciones empleando la dirección FF00H y la decodificación necesaria. Detallar el mapa de memoria, la dirección y el método de decodificación empleado. Se pretende que el programador pueda actualizar el registro de salida mediante las instrucciones:

```
MOV    DPTR, #0FF00H
MOVBX  @DPTR, A
```



El puerto de E/S se conecta a tres display de cátodo común con un decodificador BCD a 7 segmentos.

Con este hardware se va a realizar una aplicación donde se controle el turno de atención a los clientes, común en tiendas y locales comerciales.

- b) (2 puntos) Realizar una pequeña rutina que a lea de las direcciones de memoria interna 24H y 25H los números a representar en el display en código BCD empaquetado y refresque la pantalla.