

-
- Cada hoja debe tener Nombre y CI.
 - Deben estar numeradas y la primer hoja debe decir el total de hojas
 - Utilice solo un lado de las hojas
 - Incluya un solo problema por hoja
 - **Sea prolijo**
-

Problema 1

Se tienen dos sistemas basados en microprocesadores Z80. El sistema con el Z80-1 debe transferir datos al sistema con el Z80-2, para ello se propone el siguiente mecanismo.

La comunicación se implementa mediante una interfaz que utiliza un PIO (que será manejado por el Z80-2) y hardware adicional.

Se desea que los datos escritos por el Z80-1 en su dirección de E/S 0F0h puedan ser leídos por el Z80-2 en su dirección de E/S 00h. La sincronización entre escrituras del Z80-1 y las lecturas del Z80-2 se hacen mediante una señal auxiliar READY. READY le indicara al Z80-1 que puede transferir un dato al Z80-2. El Z80-1 lee READY como el bit menos significativo de la dirección de E/S 0F0h

Cada vez que el Z80-1 escribe un nuevo dato, debe desactivarse la señal READY y generar una interrupción del Z80-2. Una vez leído el dato por el Z80-2 la señal READY debe activarse para indicarle al Z80-1 que puede escribir nuevamente. Este es el único dispositivo que solicita interrupciones al Z80-2.

Se pide:

- a) Diseñar el hardware de la interfaz. Suponer que ambos micros tienen 32K de ROM y 32K de RAM. No se pide implementar dicha decodificación, solo la del espacio de E/S.
- b) Escribir el código que debe de ejecutar el Z80-2 luego de un RESET para asegurar un correcto funcionamiento del sistema.
- c) Escribir una subrutina que transfiera desde el Z80-1 NUM bytes que se encuentran en memoria a partir de la dirección BASE.

La subrutina recibe BASE en los registros HL y NUM en el registro B.

NUM puede valer entre 1 y 255. La subrutina debe testear la señal READY y en caso de esperar más de 2ms para transferir un byte debe suspender la transferencia y retornar devolviendo 0FFH en el acumulador. Si tiene éxito en transferir los NUM bytes debe retornar 00H en el acumulador. La subrutina no tiene porque preservar los registros.

Sugerencia: Para controlar los 2ms tener en cuenta cuanto tiempo transcurre entre sensado y sensado de READY y ver cuantas veces debe hacerse como máximo.

NOTA: fclk = 4MHz para ambos Z80.

Problema 2

Se desea diseñar un reproductor de mensajes de audio pre-grabados. Este aparato será capaz de reproducir uno de dos mensajes de hasta 4 segundos de duración.

El sistema tendrá las siguientes entradas y salidas de control:

Entradas: **play:** Para reproducir el mensaje.
 sel: Para seleccionar el mensaje a reproducir (sel=0 -> mensaje0, sel=1 -> mensaje1)
 reloj: Para sincronizar el sistema

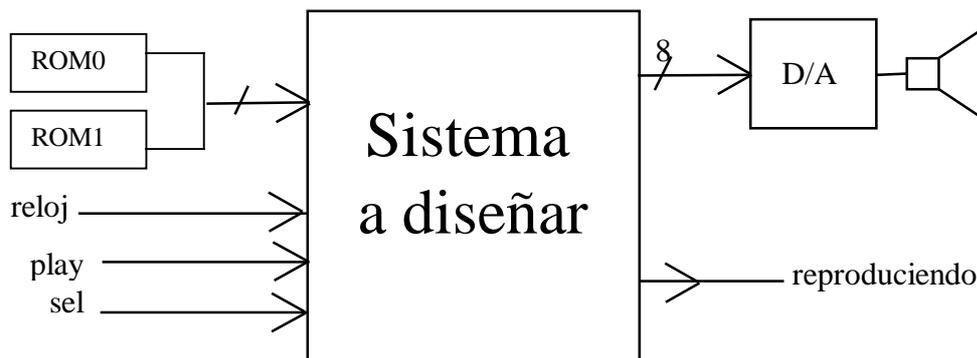
Salidas: **reproduciendo:** Mientras se está reproduciendo permanece en 1

El funcionamiento es el siguiente:

- Con un flanco de subida en **play** debe comenzar a reproducirse el mensaje seleccionado por **sel** en ese momento. Debe reproducirse **desde el comienzo y hasta el fin del mensaje**. (Notar que el mensaje puede durar menos de 4 segundos).
- Mientras se reproduce un mensaje se ignora la entrada **play**.
- La frecuencia de muestreo de la señal de voz es de 8 kHz, y las muestras son de 8 bits. Se dispone de una señal periódica **reloj** de 8kHz que se utiliza para interrumpir al sistema.

Los mensajes están previamente almacenados en dos memorias ROM0 y ROM1 de 32Kx8. La longitud en muestras de cada mensaje se graba en las dos primeras posiciones de la ROM correspondiente. Las muestras se graban a continuación en posiciones consecutivas.

El esquema general del sistema se muestra en la figura.



Convertor D/A: Reconstruye la señal analógica proporcional al valor de la entrada.

Se pide el diseño completo de un sistema basado en Z80 que se comporte de acuerdo a lo especificado, incluyendo todo el hardware y software necesario para el funcionamiento del sistema a partir de un RESET. Se dispone de chips de RAM de 16Kx8, EPROM de 16Kx8 y 32Kx8 y la lógica habitual (FF, compuertas, buffers, latches, etc). Se deberá diseñar un mecanismo de paginado para poder acceder a las memorias de mensaje en la mitad superior del espacio de memoria del Z80.

Ejercicio 1

Un sintonizador de TV implementa las funciones UP y DOWN para sintonizar el siguiente canal dentro del conjunto de canales preprogramados. Para eso se mantiene en la variable CANAL el nro. de canal actualmente sintonizado (desde 0 a 255) y se lleva una tabla en memoria con una posición para cada canal. Si el canal está preprogramado en su posición en la tabla se almacena un 0FFh, en caso contrario se almacena 00h.

```

tmp = canal
tmp = (tmp + 1) mod 256
mientras tabla[tmp] == 0
entonces {
    tmp = (tmp + 1) mod 256
}
canal = tmp
    
```

- a) El pseudocódigo del recuadro implementa la subrutina UP. Codificarla en assembler. Incluir además las directivas de reserva de memoria para CANAL y la tabla. Se supone que el sistema tiene 32K de ROM y 32K de RAM.
- b) ¿Qué sucede con la subrutina de (a) si no hay ningún canal preprogramado? Modificarla para que en ese caso retorne manteniendo incambiada la variable CANAL.

Ejercicio 2

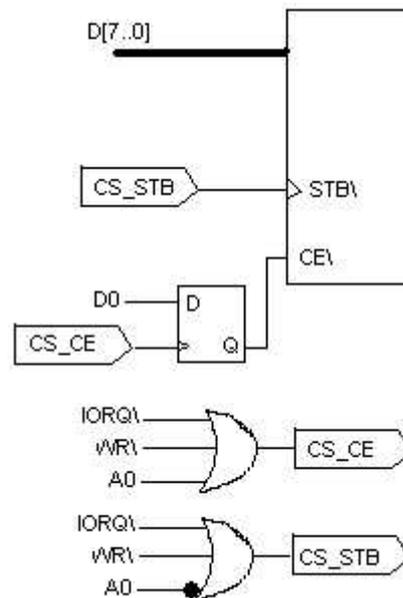
Se tiene un sistema con un Z80A funcionando a 4MHz y los puertos del diagrama.

T* es el tiempo que transcurre desde que CE=0 hasta que ocurre el primer flanco de subida en la entrada STB.

Indicar en función de los parámetros del Z80, cuanto es el mínimo valor de T* si los puertos son manejados con el siguiente código:

```

LD A,00h
OUT (00h), A
LD A,03h
OUT (01h), A
    
```



Suponer que CE = 1 al comenzar la ejecución del código.

Tiempo de propagación de FF(tp): tp_max = 10 ns, tp_min = 0 ns
 Tiempo de propagación compuerta (td): td_max = 5 ns, td_min = 0 ns