

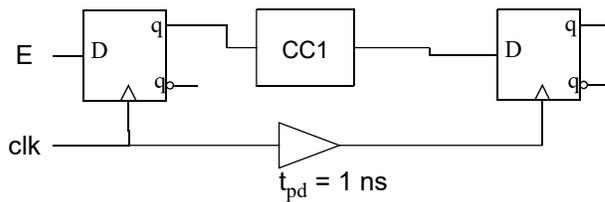
Apellidos:.....**SOLUCIÓN**.....

T.1	T.2	T.3	T.4

Nombre:..... Aula:..... Puesto:.....

TEORÍA (Cada pregunta vale 1 punto. Entregue TEORÍA y PROBLEMAS por separado)

- 1.- El circuito de la figura representa un circuito síncrono con las características temporales que se muestran en la figura. El circuito CC1 representa un circuito combinacional cuyo retraso se desconoce. ¿Existe alguna limitación en el tiempo de retraso que CC1 puede tener?. Justifique adecuadamente la respuesta.



Característica	tiempo
clk -> q	2 ns
t_{setup}	0,5 ns
t_{hold}	0,1 ns
periodo clk	10 ns

Solución

Para que el funcionamiento sea correcto, se tiene que cumplir que la salida del primer biestable tiene que llegar al segundo biestable antes del flanco activo de reloj y respetando el tiempo de setup. Esto implica una restricción sobre el retraso máximo que puede tener CC1. Como el periodo es de 10 ns, pero además entre el reloj del primer y del segundo biestable hay un retraso de 1 ns, el tiempo total es de 11 ns. Sin embargo como el retraso del biestable es de 2 ns y el tiempo de setup es de 0,5 ns, el tiempo disponible es de 8,5 ns. Este el tiempo máximo de retraso que puede tener el circuito CC1.

- 2.- Explique cómo construir un decodificador 3 a 8 con salidas activas en baja, utilizando únicamente decodificadores 2 a 4. Indique cualquier otra característica los decodificadores 2 a 4 deben tener.

Solución

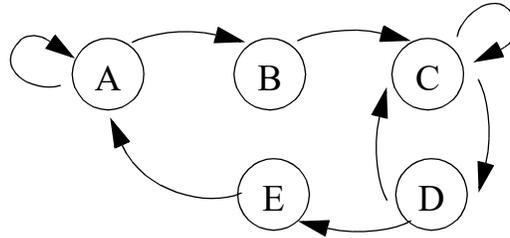
Se necesitan decodificadores 2 a 4 con salidas activas en baja y con entrada de habilitación activa en baja también. La forma de realizar la conexión puede encontrarse en los apuntes de teoría.

Apellidos:.....**SOLUCIÓN**.....

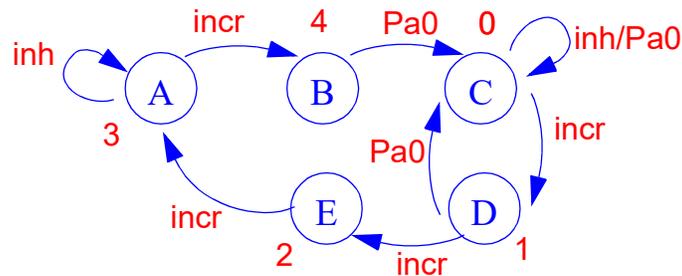
P1	P2

Nombre:..... Aula:..... Puesto:.....

- 3.- Se desea realizar el siguiente diagrama de estados con un contador. Indique las características mínimas que debe tener dicho contador (módulo y entradas y salidas especiales). Indique además la mejor asignación de cuenta para cada estado así como la operación a realizar en cada transición.

Solución

Se necesita un contador módulo 8 con entrada de puesta a cero y de inhibición de cuenta. La mejor asignación de cuenta y las operaciones a realizar en cada transición se muestran en la siguiente figura:



- 4.- Se desea almacenar en el registro R0 del CS3 el contenido de una posición de la memoria de datos. La dirección de esta posición de memoria está almacenada en otra posición de memoria, que está almacenada en el registro R1. Escriba en primer lugar un programa que haga esta operación. En segundo lugar escriba la secuencia de microoperaciones necesarias para realizar esta operación. Indique los ciclos de reloj necesarios, así como si algunas de estas microoperaciones pueden hacerse en el mismo ciclo de reloj.

SOLUCIÓN

Hay que hacer dos lecturas de la memoria de datos. En la primera memoria leemos el contenido de la dirección que tiene almacenada R1. Esta lectura la podemos almacenar en R2. La instrucción sería:

```
LD R2, (R1)
```

Después hay volver a leer la memoria de datos en la dirección almacenada en R2. Este es el dato que queremos almacenar en R0, por lo que la instrucción es:

```
LD R0, (R2)
```

Estas dos instrucciones son el programa que realiza esta operación. Las microoperaciones son:

```
AC <- R1
MAR <- AC
MDR <- MEM(MAR)
R2 <- MDR
AC <- R2
MAR <- AC
MDR <- MEM(MAR)
R0 <- MDR
```

Apellidos:.....**SOLUCIÓN**.....

Nombre:..... Aula:..... Puesto:.....

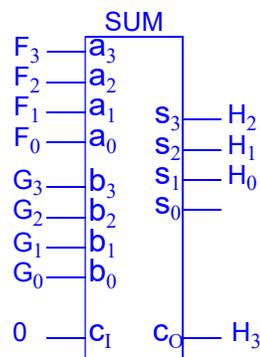
PROBLEMAS (Cada pregunta vale 3 puntos. Entregue TEORÍA y PROBLEMAS por separado)

1.- Un circuito tiene como entradas cuatro números de n bits, A, B, C y D. Usando sólo subsistemas combinatoriales no programables y puertas se desea diseñar un circuito que dé a su salida qué número (C o D) se encuentra más cerca del valor medio de A y B. Se recomienda seguir el siguiente esquema:

- [25%]** Diseñe un circuito MEDIA que calcule la media aritmética de dos números F y G.
- [25%]** Diseñe un circuito ABS que calcule el valor absoluto de la diferencia de J y K.
- [25%]** Diseñe un circuito MENOR que seleccione a su salida el menor de M y N.
- [25%]** Implemente el circuito propuesto usando los módulos anteriores.

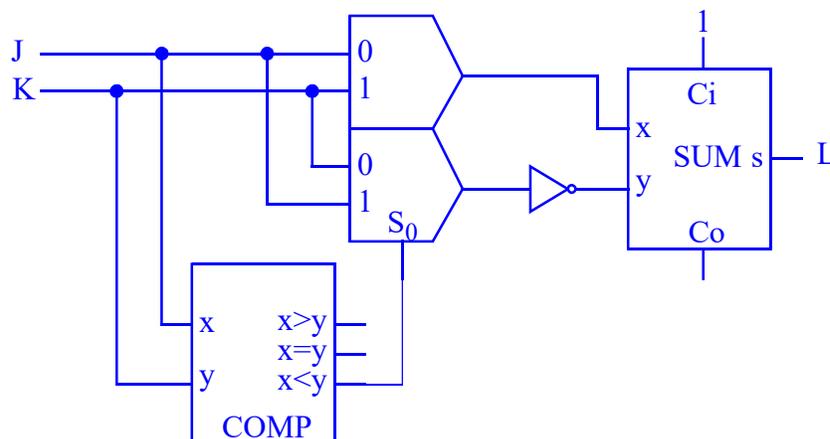
SOLUCIÓN

a) Para calcular la media de F y G sólo hay que implementar $(F+G)/2$. La suma puede llevarse a cabo con un sumador paralelo y la división por dos ignorando el LSB. Hay que tener en cuenta que es posible que la suma se desborde activando el carry de salida. Dicho bit se usará como MSB en el resultado, evitando posibles errores. Por ejemplo, si $n=4$, $F=9$, $G=15$, $F+G=24$, que no es representable en 4 bits, pero sí en 5. $F+G=11000_2$. $(F+G)/2=12=1100_2$.



b) Se trata de calcular la diferencia $|J-K|$. La resta se puede hacer sumando el complemento a dos de K, es decir, con un sumador paralelo e introduciendo el complemento a uno del sustraendo y la entrada de carry activa ($J-K = J + Ca_2(K) = J + Ca_1(K) + 1$). Para obtener el valor absoluto de la diferencia, previamente se seleccionará cuál de los dos números a restar es mayor, poniéndolo siempre como primer parámetro (con un comparador).

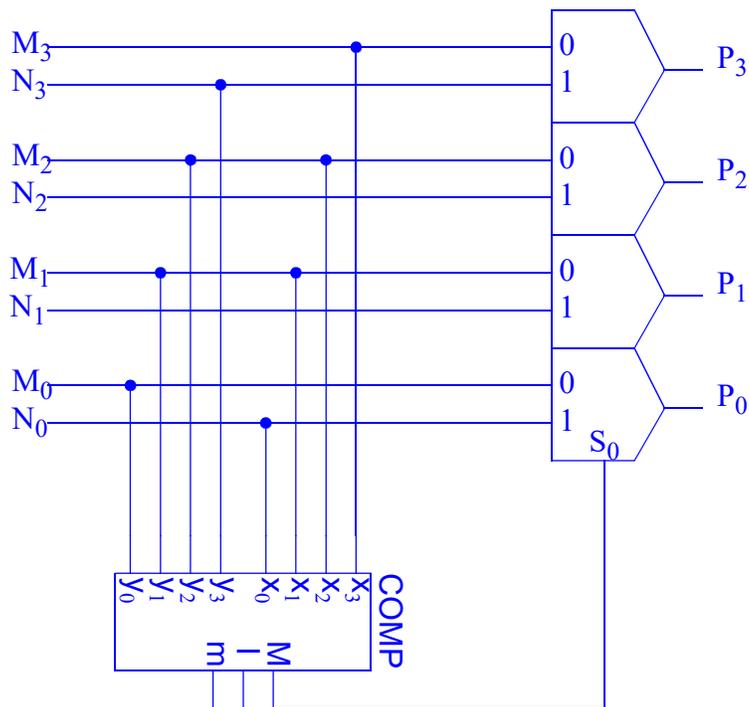
En el circuito siguiente, todas las líneas son buses de n bits, excepto las finas que son líneas individuales. Los MUX2:1 y los inversores son en realidad baterías de n elementos, uno para cada bit.



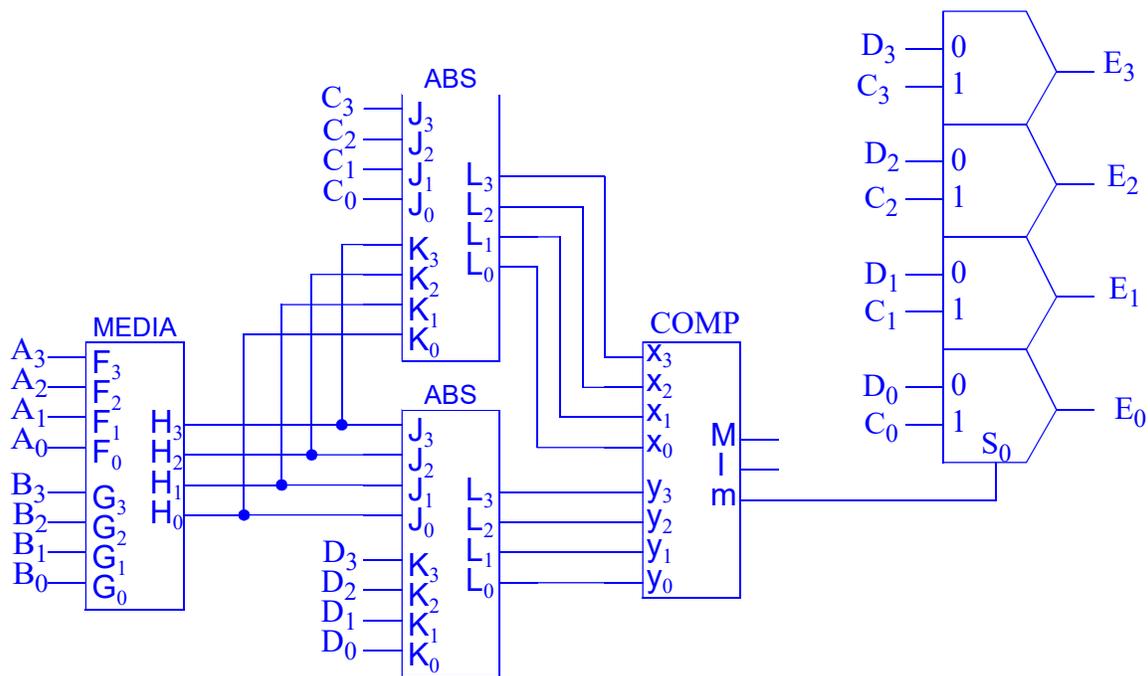
Apellidos:.....**SOLUCIÓN**.....

Nombre:..... Aula:..... Puesto:.....

c) El circuito se hará con un comparador de magnitud de n bits y n MUX2:1 (en la figura para n=4). La entrada de selección estará gobernada por la salida M del comparador.



d) Ensamblando los módulos anteriores, tenemos el siguiente circuito (se muestra para n=4). Nótese que no se ha usado el módulo MENOR, ya que ha sido necesario adaptarlo para conseguir el efecto deseado:



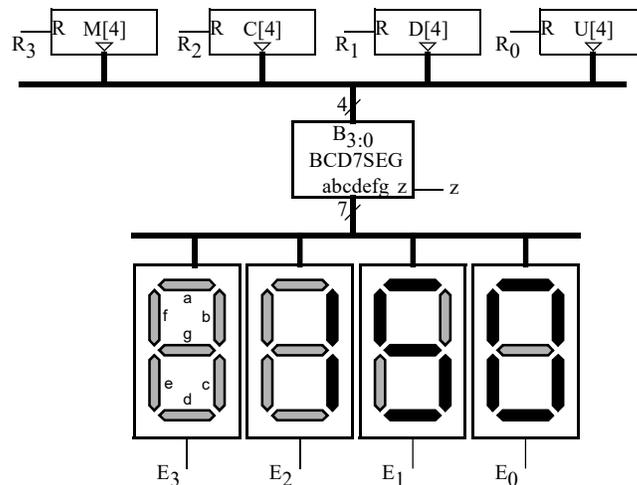
2.- Considere la siguiente unidad de datos que se usa para visualizar información numérica en decimal de hasta 4 dígitos. Dichos dígitos se encuentran almacenados en BCD en 4 registros (M para las unidades de millar, C para las centenas, D para las decenas y U para las unidades) que disponen de salida triestado con acceso a un bus de 4 bits. También conectado a ese bus hay un codificador de BCD a 7 segmentos

Apellidos:.....**SOLUCIÓN**.....

Nombre:..... Aula:..... Puesto:.....

(con una salida adicional z que se activa cuando el número que entra es 0), el cual se conecta a su vez a 4 visualizadores de 7 segmentos (son dispositivos combinatoriales sin memoria) mediante un bus dedicado de 7 bits. Por último, los visualizadores disponen de una entrada de control para habilitarlos (E_i). Si la entrada E_i está a cero, el visualizador está apagado. Si vale 1, se enciende mostrando los segmentos activos que haya en el bus.

El procedimiento para mostrar un número de 4 dígitos consiste en visualizar sólo un dígito cada vez, pero en rápida sucesión. Si la frecuencia de cambio de dígito es mayor de 100 Hz, el ojo humano no es capaz de notar el parpadeo y se consigue la ilusión de que están encendidos todos simultáneamente. Además, y para mayor claridad, se eliminan de la visualización los ceros no significativos (por ejemplo, se visualiza “ 50” y no “0050”; se visualiza “ 0” y no “0000”; ver figura donde se muestra el ejemplo suponiendo que $M=0$, $C=1$, $D=5$ y $U=0$).

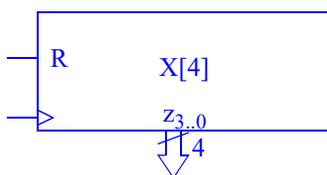


- Describa a nivel RT uno de los registros (M, C, D o U).
- Sabiendo que dispone de una señal de reloj de 100 Hz, haga el diseño de la unidad de control del sistema que visualice en los 7 segmentos los cuatro valores almacenados en los registros M, C, D y U. Obtenga la carta ASM y la implementación con la técnica de un biestable por estado).

SOLUCIÓN

a)

Registros. Los 4 son iguales y tienen lectura condicional. Se trata de dispositivos secuenciales, pero no se ha descrito bus de entrada ni señal de control asociada:

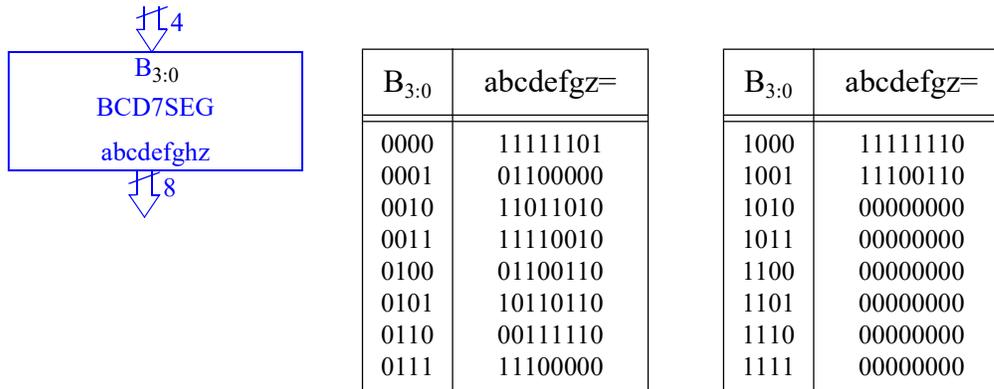


R	z=
0	HiZ
1	[X]

Apellidos:.....**SOLUCIÓN**.....

Nombre:..... Aula:..... Puesto:.....

El codificador es un dispositivo combinacional. Al ser la entrada un número BCD, los valores 10 a 16 quedan sin definir, pero se han puesto las salidas a cero.



b)

Carta ASM de la unidad de datos (y de control):

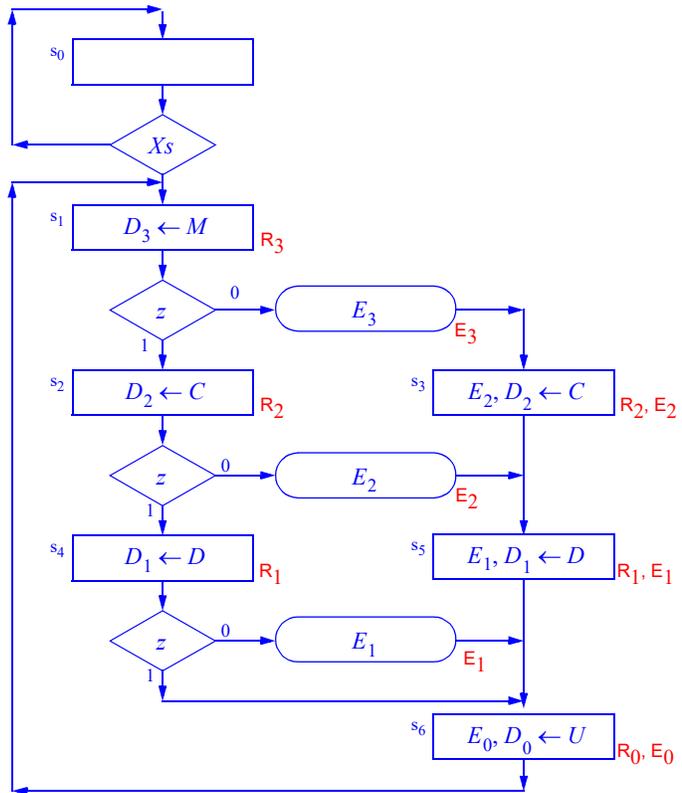
La máquina recorre un bucle de 4 estados una vez que se recibe el comando de inicio (Xs). Para poder eliminar los ceros no significativos, será necesario duplicar algunos estados para recordar si el dígito previo se visualizó o no.

Partimos del estado s_1 , responsable de visualizar las unidades de millar. En dicho estado se ordena la lectura del registro M, que será decodificado a 7 segmentos. También se analiza si dicho dígito es cero o no. En caso de que no lo sea ($z=0$) se activa habilitación del dígito D_3 ($E_3=1$) y se irá a la cadena de estados en el que se visualizan el resto de los dígitos incondicionalmente (s_3, s_5). En caso contrario ($z=1$), el dígito permanece apagado y se va a la cadena de estado en los que evalúa z

Apellidos:.....**SOLUCIÓN**.....

Nombre:..... Aula:..... Puesto:.....

después de activar la lectura de los dígitos (s_2, s_4). Nótese que el estado s_6 es especial al ser el último, ya que las unidades se visualizan siempre.



b)

