

Apellidos:.....**SOLUCIÓN**.....

1	2	3	4	5

Nombre:.....

**DURACIÓN 2:00**

1.- [1 Punto] Describa para qué sirven los circuitos de las Figuras 1 y 2 y comente sus diferencias, ventajas y desventajas.

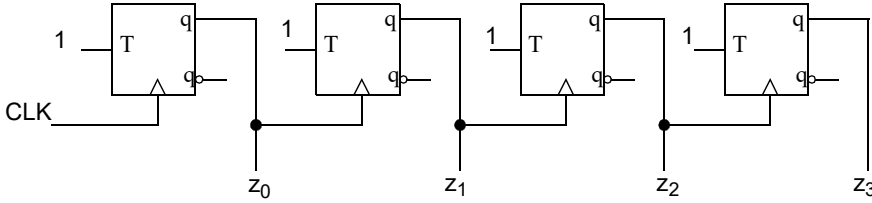


Figura 1

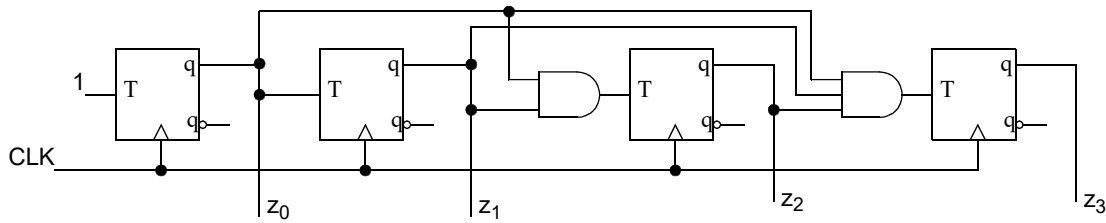


Figura 2

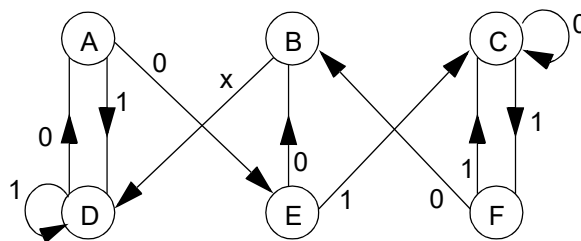
**SOLUCIÓN**

El circuito de la figura 1 es un contador asíncrono módulo 16 o contador *ripple*. En cambio, el de la figura 2 es un contador síncrono al tener todos los biestables la misma señal de reloj. La principal ventaja del contador asíncrono es que no usa puertas y que un biestable sólo se conecta con sus dos vecinos, lo que permite integrar decenas de biestables en un trozo pequeño de silicio, ya que es una estructura fácilmente repetible y hay pocas conexiones. Sin embargo, tiene el problema que las salidas  $z_i$  no cambian simultáneamente, sino que van acumulando retrasos conforme crece el subíndice. Aunque se podría hacer funcionar el contador a la máxima frecuencia de la tecnología, la lectura del estado daría problemas en función de cuándo se haga la misma. Es posible que algunos biestables ya hayan cambiado y el resto no.

El contador síncrono no tiene ese problema, ya que todos los biestable cambian a la vez. Eso permite que la velocidad a la que puede operar sea mayor que en el caso anterior, estando limitada sólo por la velocidad máxima de la tecnología y el retraso del biestable y el de una puerta. Presenta la desventaja de que hay que usar puertas AND a partir del tercer biestable y estas puertas van creciendo en tamaño. La puerta de la etapa  $i$  tendría  $i$  entradas y estaría conectada a las  $i$  salidas anteriores, lo que implica una dificultad mucho mayor de hacer el layout en el chip, lo que significa una mayor área y un mayor coste.

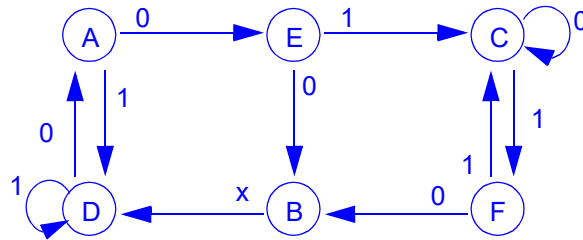
**CRITERIO DE CORRECCIÓN**

2.- [1 Punto] Dado el diagrama de estados, indique los asignamientos de estados y operaciones para una implementación con un contador reversible módulo 8, con inhibición y clear síncrono.

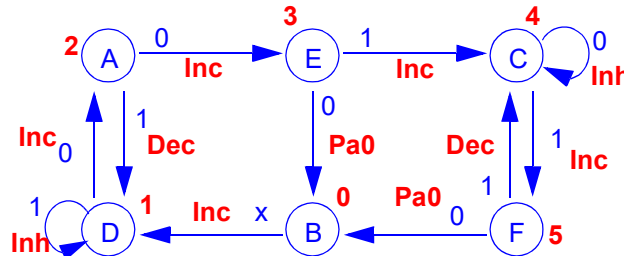


**SOLUCIÓN**

Reordenando el diagrama de estados para relacionarlo con el contador:



Con estas operaciones, e indentificando el estado 0 del contador con el B, nos quedaría:



Se podría plantear hacer D o C como estados 0, ya que son los estados que reciben más flechas. Pero ello supondría condenar la transición entre E y B a ser realizada con una carga en paralelo, no disponible. En cambio, si se hace B como estado 0, queda resuelta esa transición con una puesta a 0, y las que reciben D y C se pueden hacer con incremento, inhibición y decremento, todas disponibles.

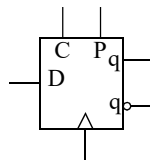
### CRITERIO DE CORRECCIÓN

3.- [2 Puntos] Tipo test. Sólo hay una respuesta correcta por pregunta. Cada fallo resta un tercio de un acierto:

1. Los biestables con reloj,

- \*\*CORRECTA\*\*** están condicionados por la señal de reloj y pueden tener otras entradas de tipo asíncrono;
- están condicionados por la señal de reloj y no pueden tener otras entradas de tipo asíncrono;
- pueden estar condicionados por la señal de reloj y no pueden tener otras entradas de tipo asíncrono;
- pueden estar condicionados por la señal de reloj y pueden tener otras entradas de tipo asíncrono.

2. Dada la siguiente figura, si las entradas C y P se conectan a un 1 lógico:

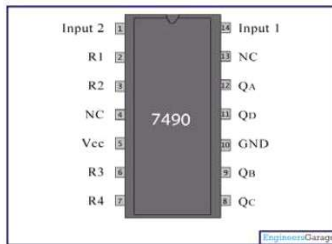


- \*\*CORRECTA\*\*** No puede darse dicho caso.
  - El biestable se establece a 0.
  - El biestable se establece a 1.
  - Ninguna de las respuestas es válida.
3. Un demultiplexor es:
- \*\*CORRECTA\*\*** Un circuito combinacional con una entrada de datos, n entradas de selección y m salidas donde  $m \leq 2^n$ .
  - Un circuito secuencial con una entrada de datos, n entradas de selección y m salidas donde  $m \leq 2^n$ .
  - Un circuito combinacional con m entradas de datos, n entradas de selección y una salida donde  $m \leq 2^n$ .
  - Ninguna de las respuestas es correcta.

4. La siguiente tabla de estados se corresponde con:

		x	
		0	1
q	0	0	1
	1	1	1
		Q	

- a) Un biestable tipo D.  
 b) Un biestable tipo T.  
 c) Un biestable tipo JK.  
 d) **\*\*CORRECTA\*\*** Ninguna de las respuestas es correcta.
5. Dada la siguiente figura, para que el circuito integrado 74LS90 funcione como un contador:



Encapsulado

Resets				Salidas			
R1	R2	R3	R4	QD	QC	QB	QA
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
X	L	X	L	Cuenta			
L	X	L	X	Cuenta			
L	X	X	L	Cuenta			
X	L	L	X	Cuenta			

Tabla de verdad Reset/Cuenta

- a) No importa el valor que tengan los resets para que implemente la cuenta.  
 b) **\*\*CORRECTA\*\*** Si se establece el reset R2 en valor bajo, no importa el valor de R3 para que el circuito implemente la cuenta.  
 c) Si se establece el reset R1 en valor alto, no importa el valor de R2 para que el circuito implemente la cuenta.  
 d) Si se establece el reset R1 en valor bajo, no importa el valor de R3 y R4 para que el circuito implemente la cuenta.

## SOLUCIÓN

### CRITERIO DE CORRECCIÓN

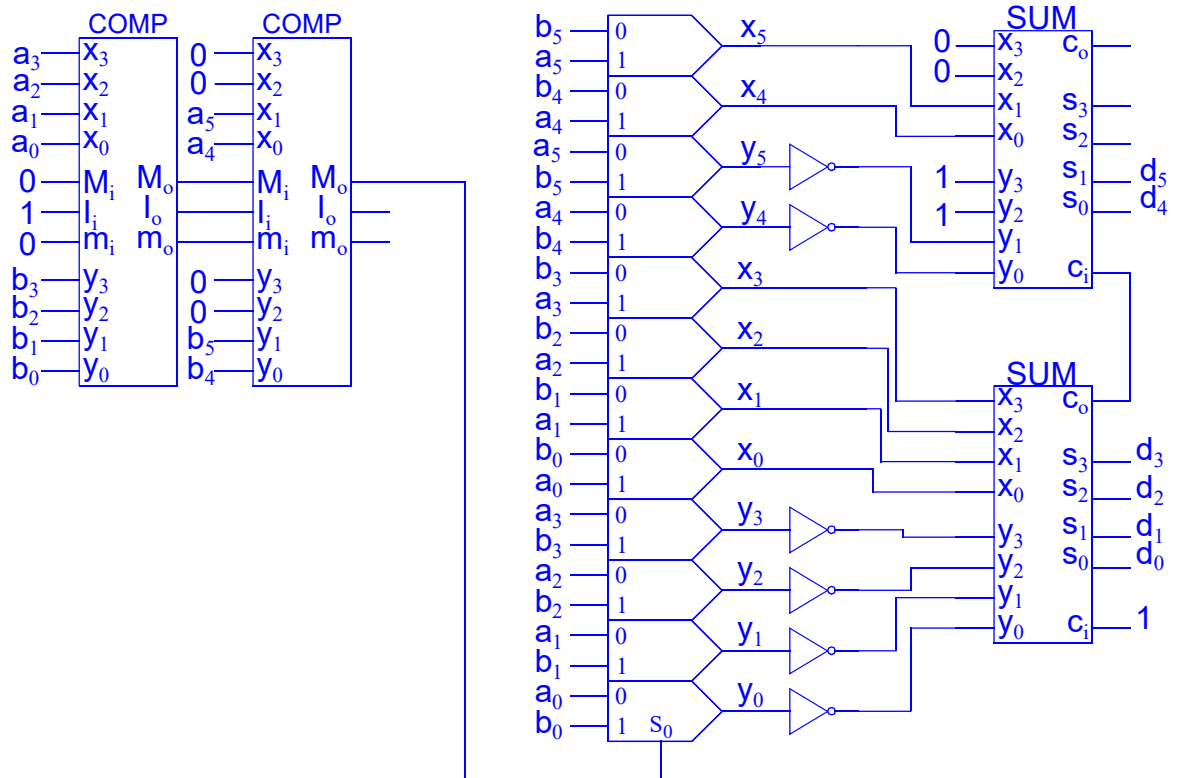
- 4.- [3 Puntos] Dadas la edades de dos personas menores de 64 años, calcule la diferencia de edad usando subsistemas combinacionales no programables de hasta 4 bits y puertas.

## SOLUCIÓN

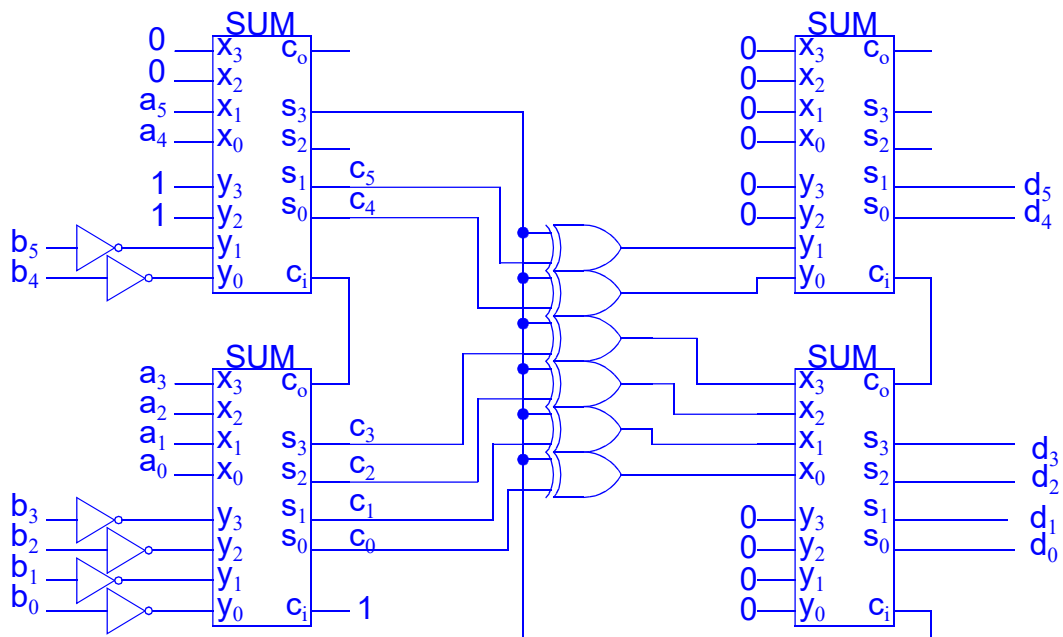
Se codificarán las edades con 6 bits ya que  $2^6=64$ . Sean  $a_{5:0}$  y  $b_{5:0}$ . Se trata de números SIN signo en binario natural. Es preciso calcular la diferencia de ambas edades asegurando que el resultado es un número positivo. Podemos usar diferentes soluciones:

- A priori decidir cuál de los dos es mayor usando un comparador de magnitud de 8 bits (realizado con comparadores de 4 bits). Después, usando dos baterías de 6 MUX2:1 y 6 inversores, obtener el mayor que llamaremos  $x_{5:0}$  y el complemento a 1 del menor, que será  $y_{5:0}$ . Formaremos un sumador de 8 bits conectando dos sumadores paralelos de 4 bits y conectaremos  $x_{5:0}$  e  $y_{5:0}$  a las entradas análogas del sumador que tendrá un carry de entrada de 1 para, junto con el complemento a 1 del segundo parámetro, restar el menor de los números al mayor. El resultado  $d_{5:0}$  es la diferencia de

edades.



- Primero se calcula  $c_{7:0} = 00a_{5:0} - 00b_{5:0}$  usando una ALU de 8 bits configurada para restar o un sumador paralelo de 8 bits al que se le conecta el segundo sumando en complemento a 1 y la entrada de carry a 1, para hacer una resta. Si  $c_{7:0}$  resulta negativo (es decir, si  $c_7=1$ ), es necesario calcular el complemento a 2 de  $c_{7:0}$ . En caso contrario, el resultado es correcto. Una forma óptima de realizar esa operación Ca2 condicional es usar una batería de 6 XOR2 controladas por  $c_7$  y usar un sumador al que se le conecta la salida de las XOR, un 0 en el otro operando y el bit de mayor peso como entrada de carry. Sólo son relevantes los 6 bits menos significativos del resultado. Se usa aritmética de 8 bits uniendo elementos de 4 bits convenientemente conectados (salida de carry del dispositivo de menor peso con la entrada de carry del de mayor peso). Al hacer la aritmética con 8 bits se garantiza que no habrá desbordamientos, ya que el resultado necesita, a lo sumo, 7 bits.



### CRITERIO CORRECCIÓN

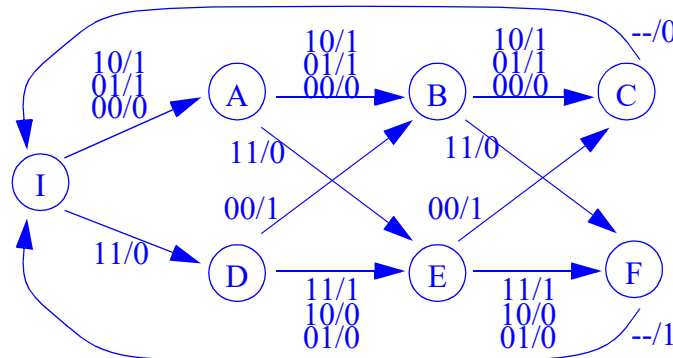
- Estructura: 50%
- Implementación: 50%

- 5.- [3 Puntos] Un circuito secuencial dispone de dos entradas (A y B) por la que se reciben dos números de 3 bits sin signo empezando por el LSB y realiza la suma algebraica de A y B. El resultado de 4 bits se va proporcionando simultáneamente a la llegada de los datos por la salida S. Entre número y número hay un bit que será ignorado a efectos de la suma. Obtenga el diagrama de estados del autómata de Mealy. Ejemplos:  
 A 011-111-  
 B 110-110-  
 S 10010101

### SOLUCIÓN

Como se trata de leer grupos de 4 bits (aunque el cuarto se deseché), la máquina de estado tendrá una estructura tal que cualquier camino que se siga siempre habrá de tener 4 estados. Además, puesto que hay que tener en cuenta el posible acarreo de la suma de los bits anteriores, la máquina tendrá que tener dos estados para cada bit. Los estados A, B y C recuerdan que el acarreo ha sido 0. Los estados D, E y F que fue 1. A y D recuerdan que se ha recibido un bit; B y E, dos; C y F, tres:

Entradas: AB  
 Salida: S



Estados:

- I: estado inicial. Esperando primer bit (LSB) de los números
- A: Recibido primer bit. Suma sin acarreo
- B: Recibido segundo bit. Suma sin acarreo
- C: Recibido tercer bit. Suma sin acarreo
- D: Recibido primer bit. Suma con acarreo
- E: Recibido segundo bit. Suma con acarreo
- F: Recibido tercer bit. Suma con acarreo

### CRITERIO CORRECCIÓN