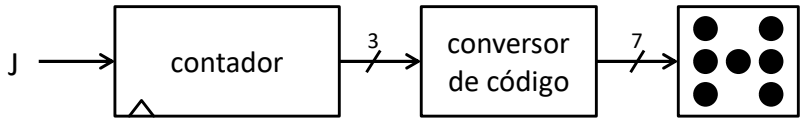
PROBLEMAS DE FUNDAMENTOS DE COMPUTADORES I
TEMA 7

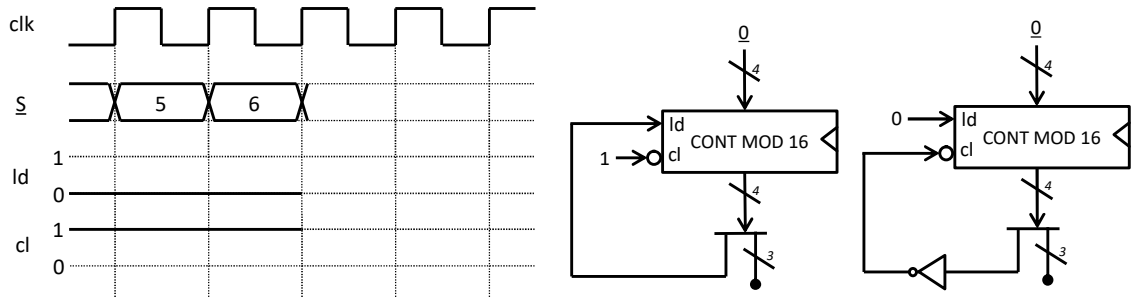
Problemas básicos:

1. Utilizando un contador con carga paralela módulo 8 y el menor número de puertas lógicas, diseñe un dado electrónico cuyo diagrama de bloques se muestra en la figura siguiente:



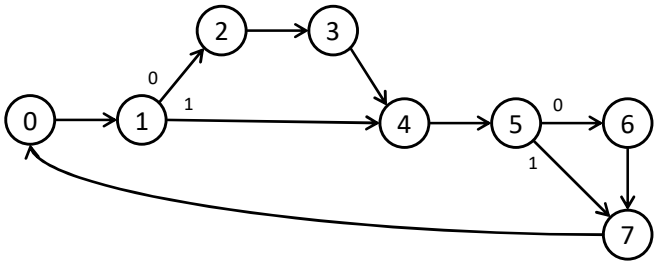
El sistema tiene una entrada, *J*, conectada a un pulsador. Cuando esta entrada vale '1', el contador sigue cíclicamente una secuencia de 6 valores distintos cada uno codificando una cara diferente del dado; cuando vale '0', el contador se detiene. La salida del contador está conectada a un conversor que, para cada valor de la secuencia, enciende los leds que correspondan según la cara del dado representada. Para evitar que el usuario pueda averiguar cuál es el valor que está marcando el dado mientras mantiene presionado el pulsador, la frecuencia de la señal de reloj debe ser lo suficientemente alta (por ejemplo, 1 KHz). Así, como el valor final del dado está determinado por el tiempo que la entrada permanece a '1' y este no puede ser controlado con exactitud por un humano, se consigue la deseada sensación de aleatoriedad.

2. Complete un cronograma, como el mostrado en la figura, para cada uno de los diseños basados en contadores mostrados en la misma.



3. Utilizando un contador con carga en paralelo y el mínimo número de puertas lógicas, implemente un sistema secuencial cuya salida repita la secuencia: 0, 1, 4, 4, 7, 7.

4. Utilizando un contador con carga en paralelo y el menor número de puertas lógicas, implemente un sistema secuencial con una entrada binaria que se comporte de acuerdo con el diagrama de la figura siguiente:



5. Usando puertas lógicas y contadores módulo 16 con capacitación de cuenta y carga paralela conectados a un reloj de 60 Hz, diseñe un temporizador de 1 segundo (su salida vale 1 durante un ciclo de cada 60). Ídem para el caso de que la frecuencia de reloj sea de 1 KHz.

6. Diseñar un registro de desplazamiento bidireccional de 3 bits con dos entradas de datos en paralelo, *E1* y *E2*. El registro tendrá una entrada de control, *S*, que indica la función a realizar según la siguiente tabla:

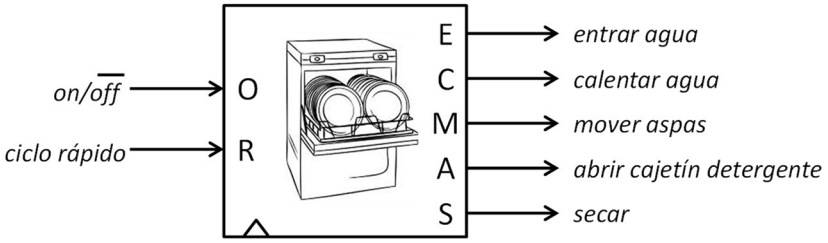
S	Q(t+1)
000	Q(t)
001	Q(t) × 2
010	Q(t) ÷ 2
011	C1(Q(t))
100	E1(t)
101	E2(t)
110	Q(t) and E2(t)
111	not Q(t)

7. Diseñe un sistema secuencial que controle el funcionamiento de un lavaplatos. El sistema tiene las 2 entradas y 5 salidas mostradas en la figura y debe comportarse según las siguientes especificaciones:

- En el estado inicial, todas las salidas valen '0'. Desde cualquier estado se va al estado inicial siempre que la tecla *on/off* vale '0', y allí se permanece hasta que *on/off* vale '1', momento en comienza desde el principio el ciclo de lavado.
- Durante su funcionamiento, el aparato pasa por 3 etapas: lavado (2 ó 4 ciclos dependiendo del valor de la tecla ciclo rápido), aclarado (1 ó 2 ciclos dependiendo del valor de la tecla ciclo rápido) y secado (1 ciclo). Después del secado se pasa siempre al estado inicial.
- Durante el lavado entra agua durante el primer ciclo, y durante el mismo se calienta. En el segundo ciclo se abre el cajetín del detergente. Todos los ciclos del lavado se mueven las aspas.
- Durante el aclarado entra agua el primer ciclo. Se mueven las aspas todos los ciclos del aclarado.
- Durante el secado se activa la salida secar.

Se pide:

a) Especifique el sistema mediante un diagrama de estados como máquina de Moore.
b) Diseñe el sistema usando un contador y el mínimo número de puertas posible.



Problemas adicionales:

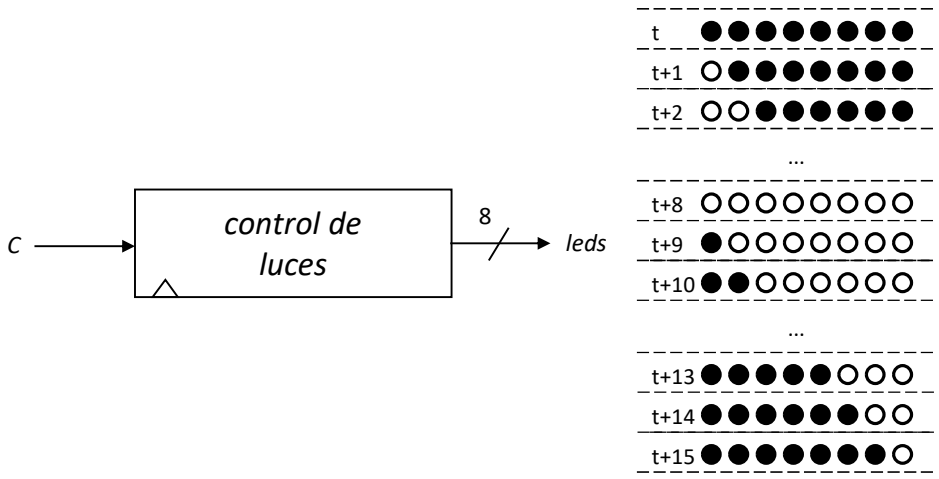
8. Utilizando como bloque básico un contador con carga paralela módulo 16, implemente un contador ascendente programable módulo m (siendo $1 < m < 16$). El sistema tendrá una entrada de capacitación de cuenta, una entrada de 4 bits por la que se indicará el valor máximo alcanzable en la cuenta y una salida de saturación que tomará el valor '1' cuando el contador alcance dicho valor máximo y adicionalmente su entrada de capacitación valga '1'.

9. Diseñe un cronómetro digital que visualiza sobre 5 displays 7-segmentos los minutos, segundos y décimas de segundo transcurridos. El cronómetro tiene una entrada asíncrona, *reset*, para puesta a cero y una entrada, *stop*, que cuando toma el valor '0' detiene la cuenta y cuando vale '1' la reanuda. En su diseño use contadores como los desarrollados en el ejercicio anterior y conversores de BCD a 7-segmentos. Suponga que la frecuencia del reloj del sistema es de 100 Hz.

10. Se desea implementar un sistema secuencial que controle un rótulo luminoso formado por 8 leds. El sistema tendrá una señal de control *C*. Si *C* vale '0', los 8 leds permanecerán apagados. Si *C* vale '1', el sistema hará que los leds se iluminen siguiendo cíclicamente la secuencia mostrada en la figura, al ritmo fijado por la frecuencia del reloj del sistema.

Se pide:

a) Implemente el sistema usando un contador y una ROM 16×8.
b) Amplíe el sistema para que los leds puedan iluminarse siguiendo 4 secuencias distintas, seleccionables a través de una entrada de 2 bits.
c) Amplíe el sistema para que el usuario, a través de un entrada de 2 bits, pueda seleccionar la velocidad a la que se sigue la secuencia (normal, ×2, ×4 y ×8).



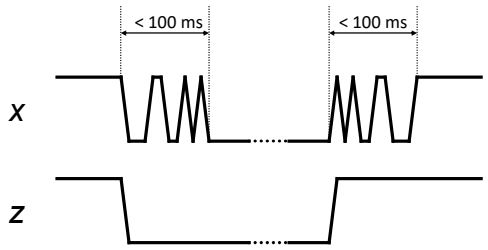
11. La luminosidad de un led puede regularse digitalmente mediante la técnica conocida como led dimming. Dado que la intensidad suministrada por la salida de un sistema digital es fija para encender un led y nula para apagarlo, esta técnica consiste modificar al porcentaje de tiempo durante el que sistema digital suministra intensidad al led cuando éste debe estar encendido. Así, el sistema digital en lugar de enviar una señal constante para encender el led (que daría lugar a un encendido de máxima luminosidad) envía una señal periódica de anchura de pulso variable cuyo factor de trabajo (el tiempo que está a '1') determina la luminosidad del led. Diseñe un circuito con una señal de carga paralela, una entrada para el valor de luminosidad (0-100) codificado en binario y una salida por la que generar la señal que regula la luminosidad de un led. Para diseñar el circuito

utilícese un contador, un registro y un comparador. La frecuencia de reloj es de 100 Hz (frecuencias inferiores hacen que el ojo humano detecte el parpadeo).

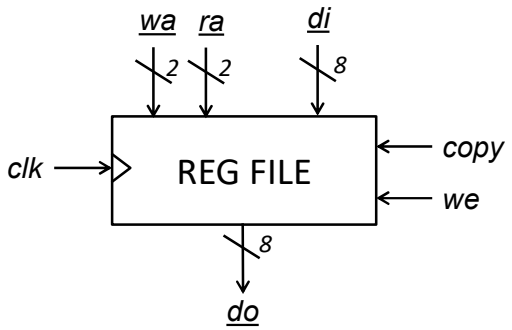
12. Diseñe un circuito secuencial que elimine los rebotes de un pulsador. El circuito funcionará a 1 MHz, tendrá una entrada, *x*, por la que entra la señal con rebotes y una salida, *z*, por la que sale la señal filtrada. Su comportamiento, que se muestra en la figura, será el siguiente:

- En reposo, mientras la entrada valga '1' (es decir, mientras el pulsador no se presione), la salida debe valer '1'.
- Tras la detección del primer flanco de bajada (provocado por la presión del pulsador), la salida debe permanecer estable a '0' durante 100 ms con independencia de los valores que tome la entrada durante ese intervalo.
- Transcurridos los 100 ms y mientras que la entrada valga '0' (es decir, mientras se mantenga el pulsador presionado), la salida debe valer '0'.
- Tras la detección del primer flanco de subida (provocado por la depresión del pulsador), la salida debe permanecer estable a '1' durante 100 ms con independencia de los valores que tome la entrada durante ese intervalo.
- Transcurridos los 100 ms, el sistema volverá al estado de reposo.

Para medir un intervalo de 100 ms utilícese un temporizador formado por un contador y un comparador. El sistema constará de una máquina de estados que convenientemente lea la entrada, escriba la salida, arranque el temporizador y espere su finalización.



13. Se desea ampliar la funcionalidad de un banco de 4 registros de 8 bits añadiéndole una función de copia. Para ello, dispondrá de una entrada de control adicional, *copy*. Cuando vale '0' el banco de registros debe comportarse normalmente y cuando vale '1' (y *we* también vale '1') debe copiar el contenido del registro indicado por <u>*ra*</u> en el registro indicado por <u>*ra*</u> ignorando el valor presente en <u>*di*</u>.



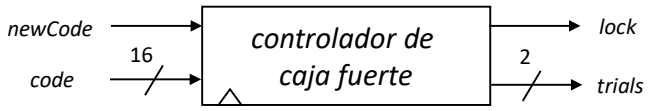
14. Un watchdog es un temporizador que se utiliza para la detección y recuperación de posibles errores de funcionamiento en un computador. Durante un funcionamiento normal, el computador reinicia regularmente el watchdog para evitar que este consuma su tiempo. Si el watchdog alcanza el time-out, activa una señal que dispara una acción correctiva (típicamente el reinicio del computador). Diseñe un watchdog con time-out

seleccionable (1s, 500ms, 250ms, 125ms) a través de una entrada de 2 bits. Además el sistema tendrá una entrada para reiniciar la cuenta y una salida que se activará al alcanzar el time-out. El sistema funcionará a 50 KHz y se diseñará usando un contador, un comparador y un multiplexor 4 a 1.

15. Diseñe el sistema para controlar la apertura y cierre de una caja fuerte mostrado en la figura. Las entradas, *code* y *newCode*, están conectadas a un teclado, la salida *trials* a un display y la salida *lock* al motor del cerrojo. El sistema debe comportarse según las siguientes especificaciones:

- El cerrojo permanece abierto mientras *lock* vale '0' y cerrado en caso contrario.
- Inicialmente la caja fuerte debe estar abierta.
- Cada vez que el usuario teclee una clave de 4 dígitos, el teclado pondrá en *code* un código de 16 bits (codificando en BCD las 4 teclas pulsadas) y pondrá la señal *newCode* a '1' durante un único ciclo de reloj para que el sistema lea en paralelo la clave introducida.
- El primer código enviado hará que el sistema almacene la clave y cierre el cerrojo.
- Los sucesivos códigos permitirán abrir el cerrojo siempre y cuando la clave leída coincida con la clave almacenada.
- Se tendrán un máximo de 3 intentos para acertar la clave almacenada.
- Tras fallar los 3 intentos el cerrojo quedará indefinidamente cerrado.
- El sistema indicará a través de *trials* el número de intentos restantes hasta el bloqueo del cerrojo.

Para diseñar el sistema utilícese un registro de 16 bits, un comparador de igualdad de 16 bits y una máquina de estados.



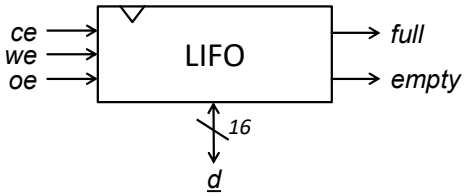
16. Una UART (Universal Asynchronous Receiver-Transmitter) es uno de los dispositivos de E/S serie más comunes. Consta de un transmisor que serializa los datos paralelos que se quieren enviar, un receptor que paraleliza los datos serie que se reciben y un controlador que comunica ambos elementos con el computador. Se desea diseñar el circuito que efectúa la transmisión serie asíncrona de datos de 8 bits. El circuito tiene las entradas y salidas mostradas en la figura y debe comportarse según las siguientes especificaciones:

- En inactividad, la señal *Tx* debe valer '1'.
- Los datos se transmitirán en una trama de 11 bits que comenzará con un bit de start a '0', los 8 bits de datos comenzando por el bit menos significativo, un bit de paridad impar (el patrón formado por este bit y los 8 bits de datos debe tener siempre un número impar de '1') y un bit de stop a '1'.
- El sistema, cada vez que detecte que la señal *start* vale '1', cargará de *data* el dato a transmitir y comenzará a transmitirlo en serie.
- La señal *ready* debe valer '0' mientras que se esté transmitiendo el dato y '1' en caso contrario

La frecuencia de reloj es de 5 MHz y los datos deberán transmitirse a 9600 baudios (bits/segundo). Para diseñar el circuito utilícese un registro de desplazamiento de 11 bits (para la conversión paralelo-serie de los datos), un contador 0-11 (para la cuenta

del número de bits desplazados), un contador del tamaño que considere oportuno (para la cuenta del número de ciclos que deben transcurrir entre un desplazamiento y otro calculados como el cociente entre la frecuencia de reloj y la velocidad de transmisión) y las puertas lógicas que estime necesarias.

17. Un buffer LIFO (last in, first out) es un almacén de datos no accesibles por dirección sino legibles en orden inverso al orden en que se escribieron (es decir, el primer dato que se puede leer es siempre el último que se ha escrito). Usando un contador ascendente/descendente, 2 RAM 256×8 bits y otros componentes que estime necesarios, diseñe una LIFO capaz de almacenar un máximo de 256 datos de 16 bits. El sistema tendrá una entrada/salida de datos, *d*, de 16 bits, 3 señales de control, *ce*, *we* y *oe*, para capacitar el módulo, la escritura y la lectura respectivamente y 2 señales de estado, *full* y *empty*, para indicar cuándo la LIFO está llena y cuándo está vacía.



18. Usando un sumador/restador y un registro, diseñe un acumulador genérico que opere con números enteros con signo codificados en C2 con *n* bits. El sistema acepta un operando, y lo suma/resta al resultado calculado en con anterioridad. El sistema tiene un puerto de datos de entrada, *x*, un puerto de datos de salida, *z*, dos entradas de control, *ld* y *op,* y una salida de estado, *ov*. Si *ld* vale '1', el dato en la entrada *x* debe acumularse; si vale '0', no. Si la entrada *op* vale '0', el dato debe sumarse; si vale '1', restarse. La salida *ov* debe activarse cuando el resultado no sea representable en C2 con *n* bits.

19. Usando multiplicadores, sumadores, multiplexores y registros, diseñe 3 rutas de datos sin controlador tal que cada una pueda realizar una de las siguientes expresiones, donde *x*, *a* y *b* son puertos de entrada de n bits y *z* es un puerto de salida de n bits:

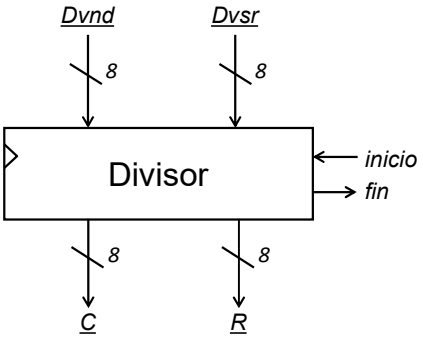
a) $\mathrm{z}(t) = \sum_{i=1}^{t}[\mathrm{a}(i)\cdot\mathrm{x}(i)] \equiv z(t-1) + \mathrm{a}(t)\cdot\mathrm{x}(t)$
b) $\mathrm{z}(t) = \sum_{i=1}^{t}[\mathrm{a}(i)\cdot\mathrm{x}(i) + \mathrm{b}(i)] \equiv z(t-1) + \mathrm{a}(t)\cdot\mathrm{x}(t) + \mathrm{b}(t)$
c) $\mathrm{z}(t) = \sum_{i=1}^{t}[\mathrm{x}(i)^2 + \mathrm{x}(i) + \mathrm{b}(i)] \equiv z(t-1) + \mathrm{x}(t)^2 + \mathrm{x}(t) + \mathrm{b}(t)$

Rediseñe las rutas de datos b) y c) de modo que utilicen un único sumador. Discuta por qué estos diseños con solo un sumador requieren tener controlador y no pueden aceptar datos nuevos y generar resultados en todos los ciclos.

20. Diseñar un circuito secuencial capaz de calcular iterativamente el elemento i-ésimo de la serie de Fibonacci. El circuito tendrá una entrada, *i* de 5 bits, para indicar el elemento de la serie; una entrada, *inicio*, para indicar el comienzo del cálculo; una salida, *fin*, para indicar el fin del cálculo y una salida *z* de 20 bits, para indicar el valor del elemento de la serie indicado. La serie de Fibonacci se define recursivamente de la siguiente manera: fib(0)=0, fib(1)=1, fib(*i*)=fib(*i*-1) + fib(*i*-2). El sistema deberá diseñarse usando circuito realizará el cálculo iterativamente. Tendrá 1 sumador de 20 bits, 2 registros para almacenar los 2 últimos elementos de la serie y un contador que llevará la cuenta del índice.

21. Diseñar un sistema secuencial que implemente la división entera de 2 números naturales codificados en binario utilizando el algoritmo de restas sucesivas siguiente:

```
begin
    Dividendo := Dvnd;
    Divisor := Dvsr;
    Cociente := 0;
    while Dividendo >= Divisor do begin
        Dividendo := Dividendo - Divisor;
        Cociente := Cociente + 1;
    end while;
    C := Cociente;
    R := Dividendo;
end;
```



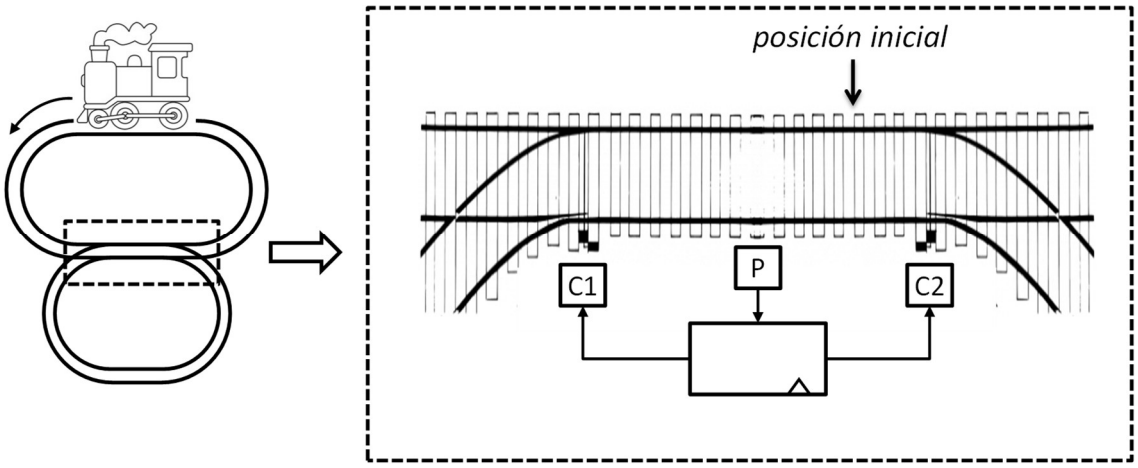
El sistema tendrá dos entradas, <u>*dvnd*</u> y <u>*dvsr*</u> de 8 bits, para operandos; una entrada, *inicio*, para indicar el comienzo del cálculo; una salida, *fin*, para indicar el fin del cálculo y dos salidas, <u>*c*</u> y <u>*r*</u> de 8 bits, para resultados. El sistema tendrá los puertos mostrados en la figura y utilizará 2 registros, 1 contador y 1 restador.

Problemas de examen:

22. (Febrero 2012) Sea un tren eléctrico de juguete con un trazado de vías en forma de ocho que dispone de un sensor de presencia (*P*) y dos cambios de agujas (*C1* y *C2*). Se desea diseñar un sistema que, leyendo el valor del sensor y controlando los cambios de agujas, permita que el tren realice indefinidamente la siguiente secuencia: dos vueltas por el tramo superior, dos vueltas por el tramo inferior, una vuelta en ocho.

El sistema tiene como entrada *P* y como salidas *C1* y *C2*. La entrada *P* se activa (vale '1' durante un ciclo de reloj) cuando el tren pasa por el tramo de vía que comparten los trayectos circulares superior e inferior. Por su parte el tren seguirá una trayectoria que es función del valor en las salidas *C1* y *C2*:

- Cuando *C1*='0' y *C2*='0', el tren realiza recorridos circulares por el tramo superior.
- Cuando *C1*='1' y *C2*='1', el tren realiza recorridos circulares por el tramo inferior.
- Cuando *C1*='0' y *C2*='1', el tren pasa del tramo superior al inferior.
- Cuando *C1*='1' y *C2*='0', el tren pasa del tramo inferior al superior.



Suponiendo que el tren se encuentra en la posición inicial y circula en el sentido indicados en la figura, se pide:

a) Especificar el sistema como máquina de Moore

b) Implementarlo utilizando un contador módulo-8 y el menor número de puertas lógicas.

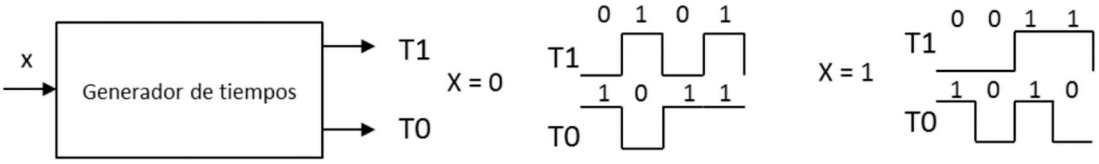
23. (Septiembre 2012) Se desea diseñar el sistema de control de una muñeca interactiva. El sistema tiene 2 entradas y 2 salidas, todas ellas binarias. La entrada *R* valdrá '1' cuando haya ruido y la entrada *C* lo hará cuando haya un chupete en la boca de la muñeca. Por su parte, la salida *G* habilita un generador de sonidos que reproduce o bien un llanto (si *L* es igual a '1') o bien algunas palabras (si *L* es igual a '0').

Una vez encendida, la muñeca se encontrará en estado "tranquila" donde, si no hay estímulos, ni habla, ni llora. Si se hace ruido, sigue "tranquila" y habla. Si se le pone el chupete (haya o no ruido), dejará de hablar (si lo estuviera haciendo) y pasará al estado "dormida". En el estado "dormida" no hace nada y permanecerá en él hasta que, sin tener el chupete puesto, se escuche un ruido. En ese caso llorará y pasará al estado "asustada". En el estado "asustada" permanecerá llorando mientras el ruido se mantenga. Cuando el ruido desaparezca dejará de llorar y pasará a estar "dormida" o "tranquila" en función de si tiene o no el chupete puesto.

Se pide:

a) Especificar el sistema como máquina de Mealy.
b) Implementarlo utilizando un contador mod-4 y el menor número de puertas lógicas.

24. (Junio 2017) Diseñar un generador de patrón de tiempo con una entrada de control binaria x y dos salidas binarias T1 y T0 que se comporte de la siguiente manera:

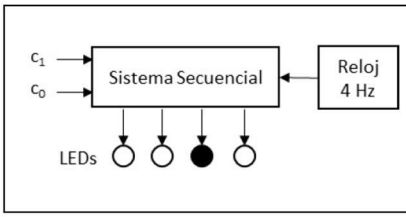


El sistema generará señales completas, de manera que sólo se tendrá en cuenta el valor de la entrada x cuando finaliza la generación de una señal. Inicialmente el sistema generará la señal correspondiente para x=0.

a) Dibujar el diagrama de estados del sistema como máquina Moore.
b) Implementarlo utilizando un contador binario módulo 8 y puertas lógicas.

25. (Septiembre 2017) Utilizando registros y el menor número de puertas lógicas, implementar un circuito secuencial que recibe números de 8 bits codificados en complemento a dos y que se comporta de la siguiente manera: Z(t)=1 cuando $\underline{x}$(t-3) es par y negativo, $\underline{x}$(t-2) es par y positivo, y $\underline{x}$(t-1) es impar y negativo; y vale 0 en el resto de los casos.

26. (Septiembre 2018) Se desea diseñar un sistema secuencial que actúe sobre un conjunto alineado de 4 LEDs de manera que realice cuatro funciones diferentes según el valor de dos entradas binarias de control c_1 y c_0:

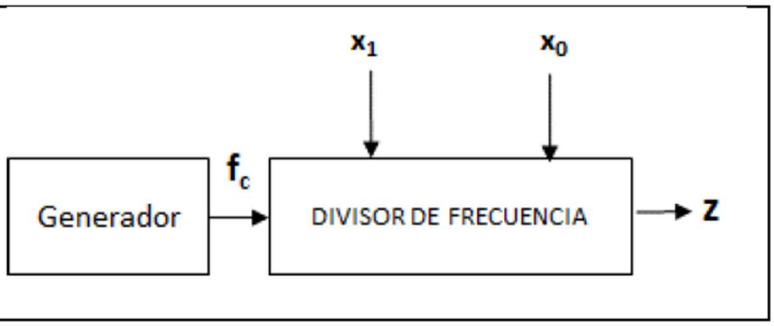


c_1	c_0	Función
0	0	Rotación derecha de un LED encendido durante 1 segundo
0	1	Rotación izquierda de un LED encendido durante 1 segundo
1	0	Todos los LEDs apagados
1	1	Todos los LEDs encendidos

Para ello se dispone de un reloj de 4 Hz, contadores ascendentes/descendentes de 2 bits, multiplexores, descodificadores, y puertas lógicas. Diseñar el sistema utilizando el menor número de componentes.

27. (Julio 2019) Diseña un divisor programable de frecuencia a partir de una señal digital de frecuencia f_c con factores de división 2, 4, 8 y 16. La selección del factor de división se hará con dos variables binarias x_1 y x_0, según la siguiente tabla. Para el diseño utiliza contadores binarios de 4 bits, decodificadores y el menor número de puertas lógicas.

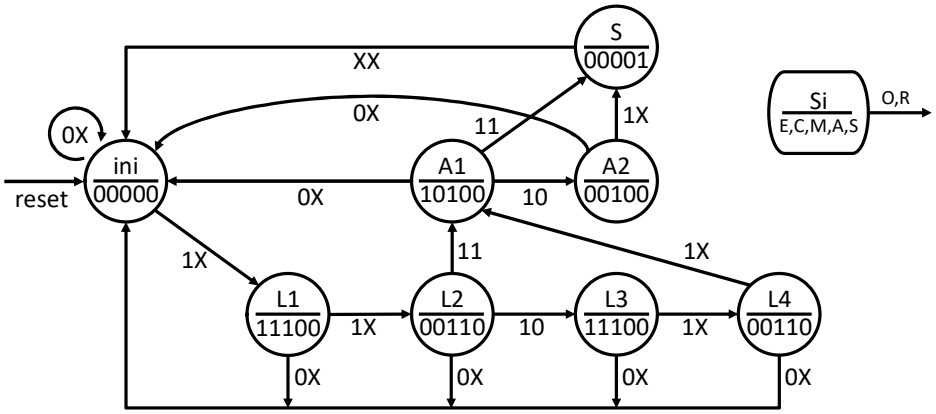
x_1	x_0	z
0	0	$f_c \div 2$
0	1	$f_c \div 4$
1	0	$f_c \div 8$
1	1	$f_c \div 16$



Soluciones

1. al **6.** Véanse las transparencias.

7.



$\underline{Q} = \{000: ini, 001: L1, 010: L2, 011: L3, 100: L4, 101: A1, 110: A2, 111: S\}$

$\overline{E}(Q_2, Q_1, Q_0) = \sum m(1,3,5) = \overline{Q_2}Q_0 + \overline{Q_1}Q_0$

$C(Q_2, Q_1, Q_0) = \sum m(1,3) = \overline{Q_2}Q_0$

$M(Q_2, Q_1, Q_0) = \sum m(1 \dots 6) = Q_2\overline{Q_1} + Q_1\overline{Q_0} + \overline{Q_2}Q_1 + \overline{Q_1}Q_0$

$A(Q_2, Q_1, Q_0) = \sum m(2,4)$

$S(Q_2, Q_1, Q_0) = \sum m(7)$

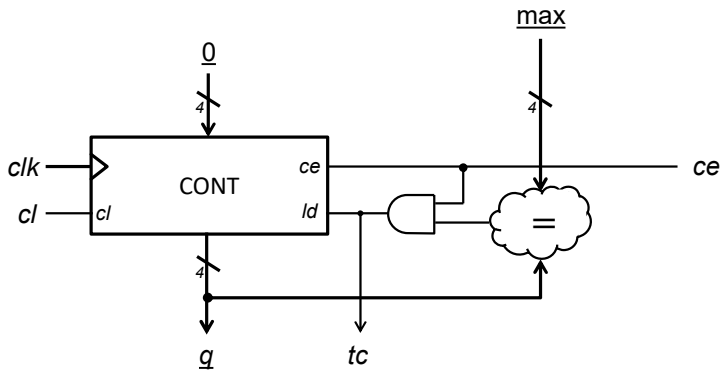
$ce(O, C, Q_2, Q_1, Q_0) = \sum m(16 \dots 25,27,28,30,31) + \sum d(0 \dots 15,26,29) = 1$

$ld(O, C, Q_2, Q_1, Q_0) = \sum m(0 \dots 15,26,29) = \overline{O} + C\overline{Q_2}Q_1\overline{Q_0} + CQ_2\overline{Q_1}Q_0$

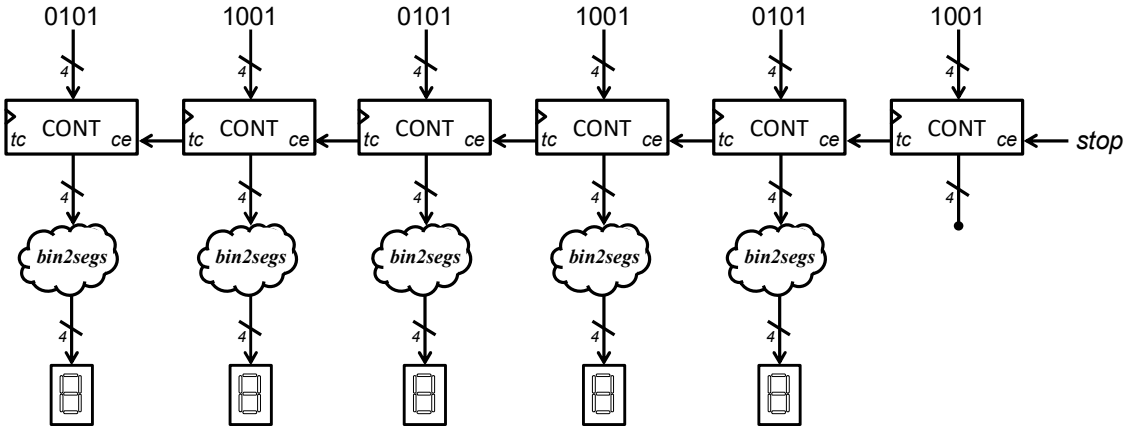
$e_2(O, C, Q_2, Q_1, Q_0) = e_0 = \sum m(26,29) + \sum d(16 \dots 25,27,28,30,31) = \overline{O}$

$e_1(O, C, Q_2, Q_1, Q_0) = \sum m(29) + \sum d(16 \dots 25,27,28,30,31) = OQ_2$

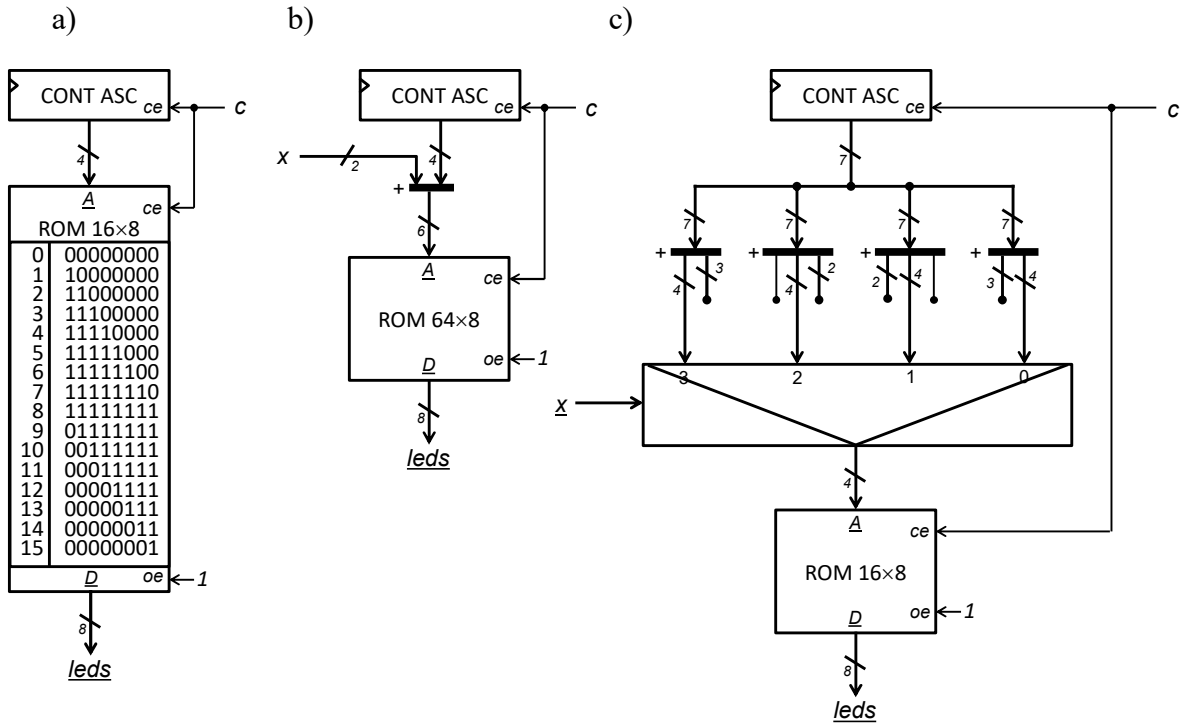
8.



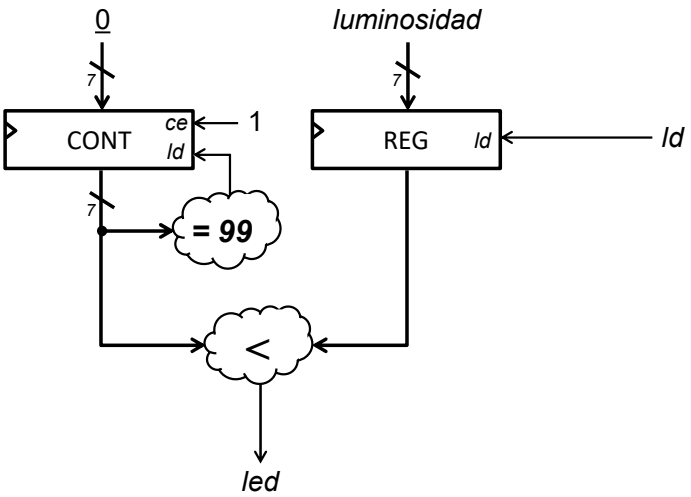
9.



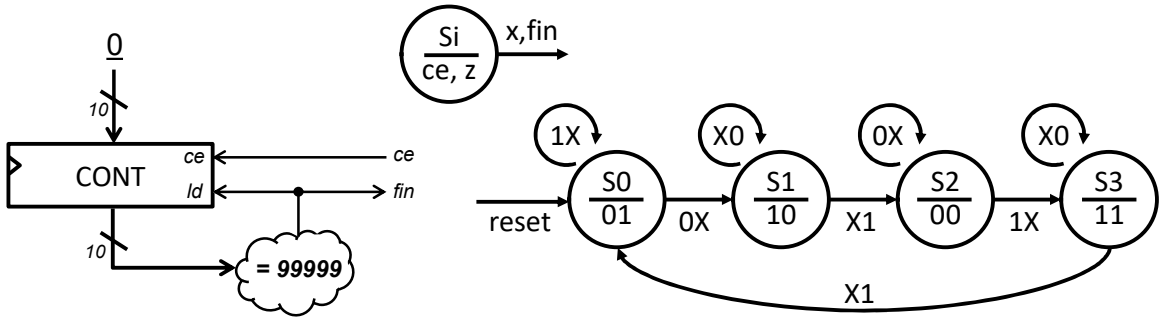
10.


a)
b)
c)
CONT ASC
ce
c
ROM 16×8
A
D
oe
1
leds
0 00000000
1 10000000
2 11000000
3 11100000
4 11110000
5 11111000
6 11111100
7 11111110
8 11111111
9 01111111
10 00111111
11 00011111
12 00001111
13 00000111
14 00000011
15 00000001
x
ROM 64×8
ROM 16×8

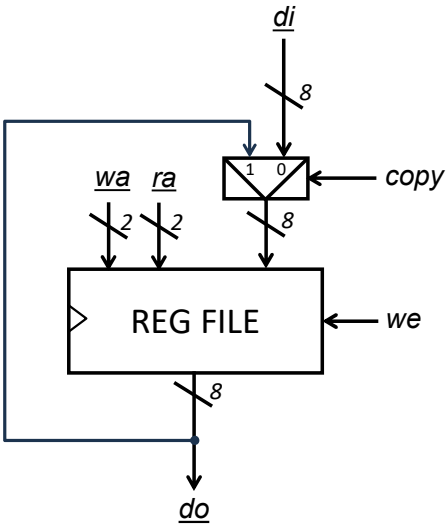
11.


0
luminosidad
CONT
ce
ld
1
REG
ld
= 99
<
led

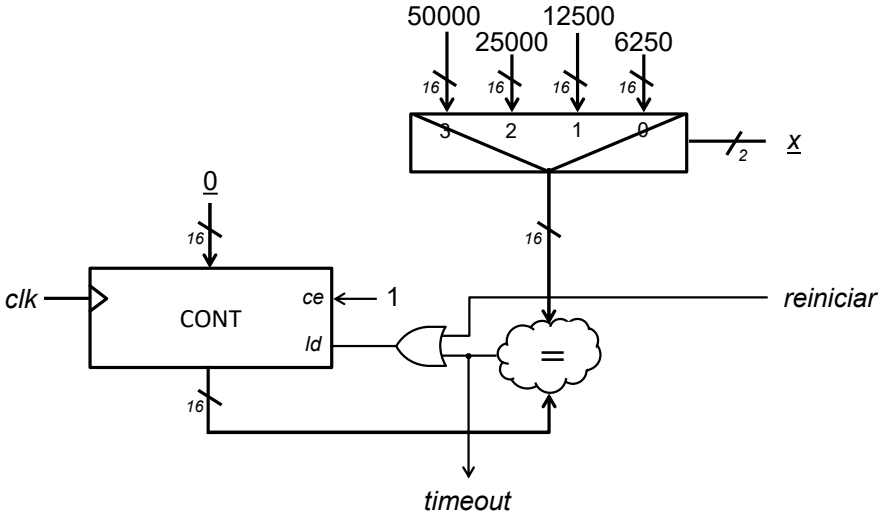
12.


0
CONT
ce
ld
fin
= 99999
Si
ce, z
x,fin
1X
X0
0X
X0
reset
S0
01
S1
10
S2
00
S3
11
0X
X1
1X
X1

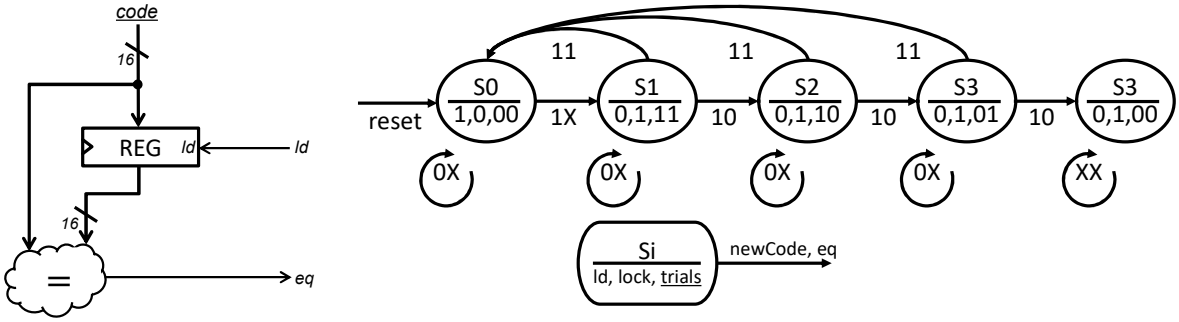
13.


di
8
1
0
copy
wa
ra
2
2
8
REG FILE
we
8
do

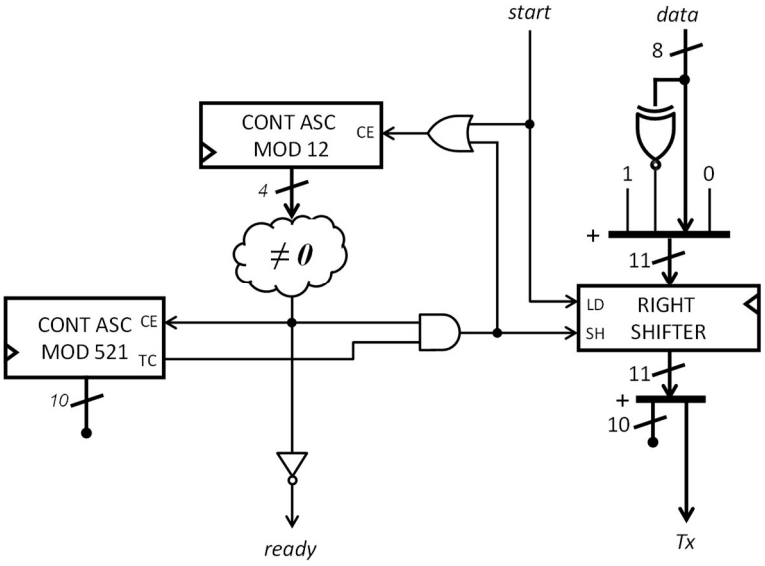
14.


50000
12500
25000
6250
16
16
16
16
3
2
1
0
2
x
0
16
16
clk
CONT
ce
ld
1
reiniciar
=
16
timeout

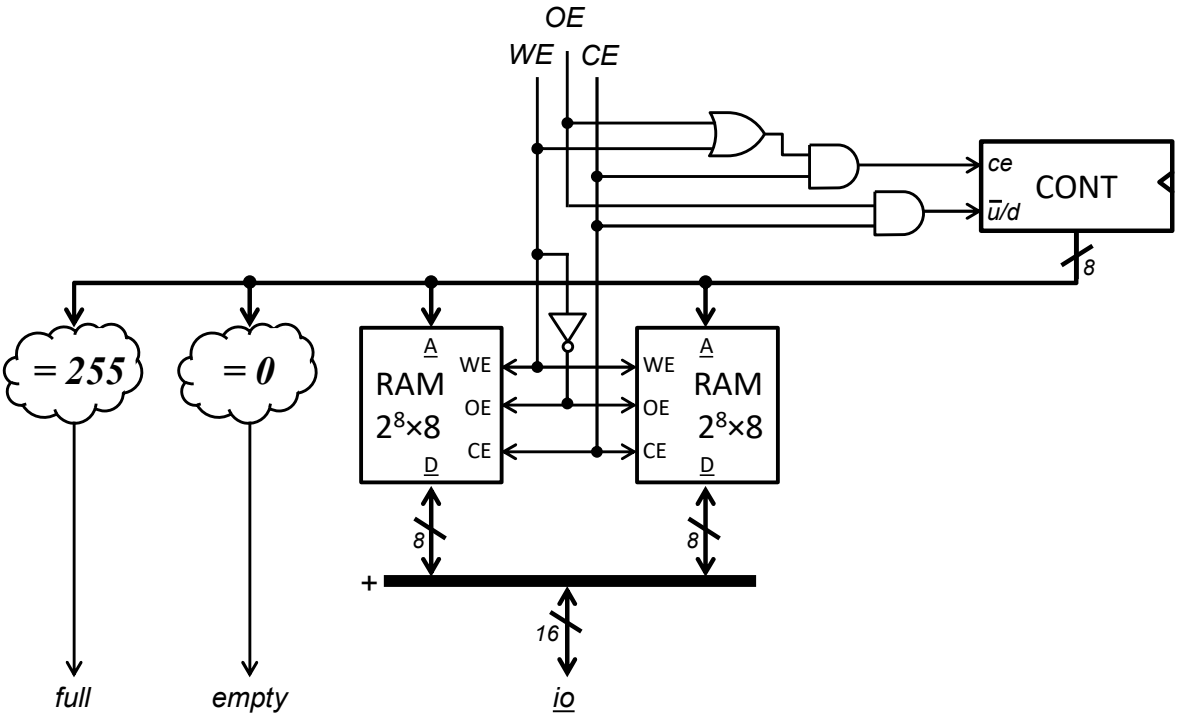
15.


code
16
REG
ld
ld
16
=
eq
reset
S0
1,0,00
1X
S1
0,1,11
10
S2
0,1,10
10
S3
0,1,01
10
S3
0,1,00
11
11
11
0X
0X
0X
0X
XX
Si
ld, lock, trials
newCode, eq

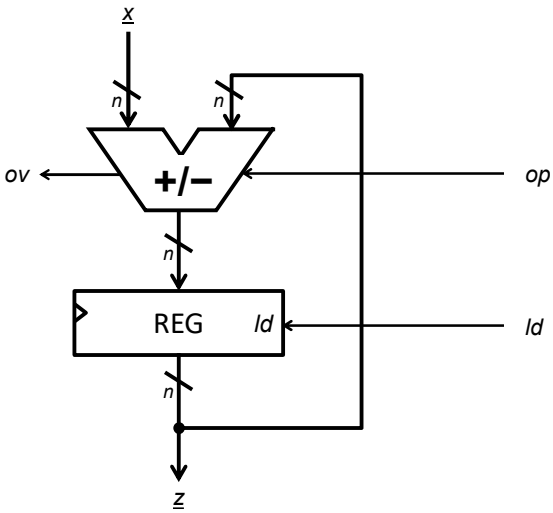
16.


start
data
8
CONT ASC
MOD 12
CE
4
≠ 0
1
0
+
11
CONT ASC
MOD 521
CE
TC
10
LD
SH
RIGHT
SHIFTER
11
+
10
ready
Tx

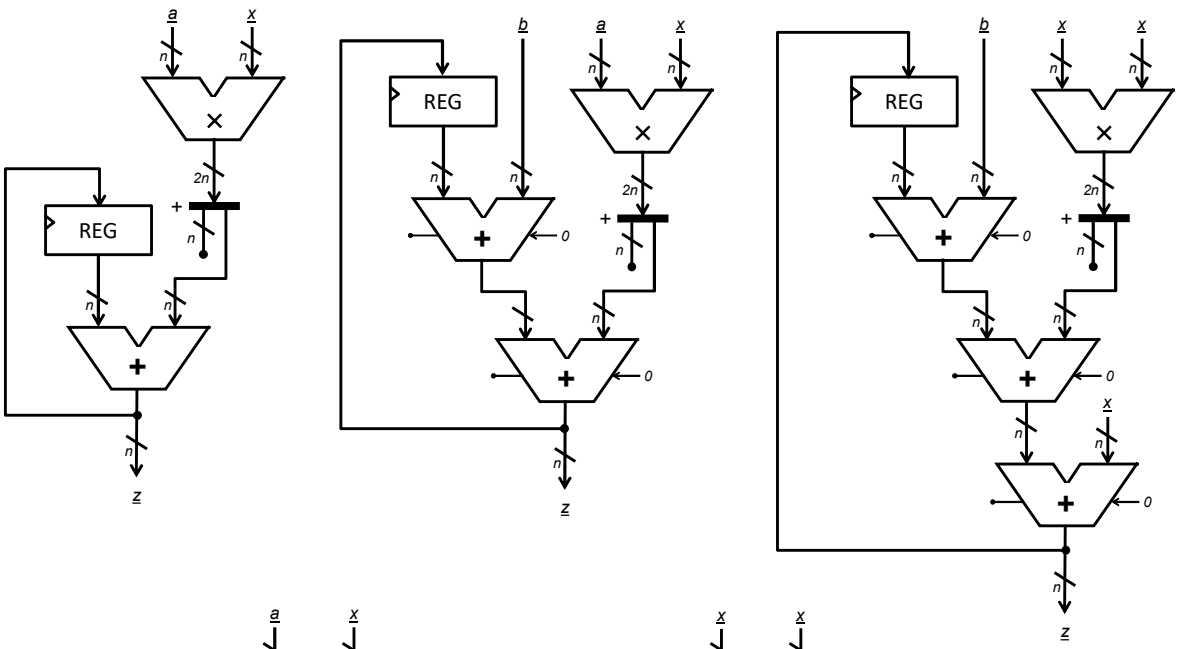
17.

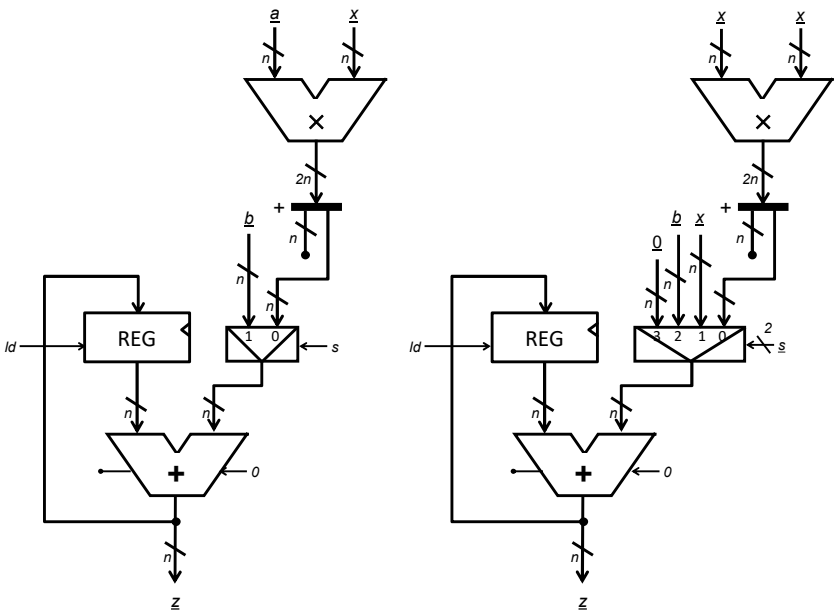

OE
WE
CE
ce
ū/d
CONT
8
= 255
= 0
A
RAM
2⁸×8
WE
OE
CE
D
A
RAM
2⁸×8
WE
OE
CE
D
8
8
+
16
full
empty
io

18.

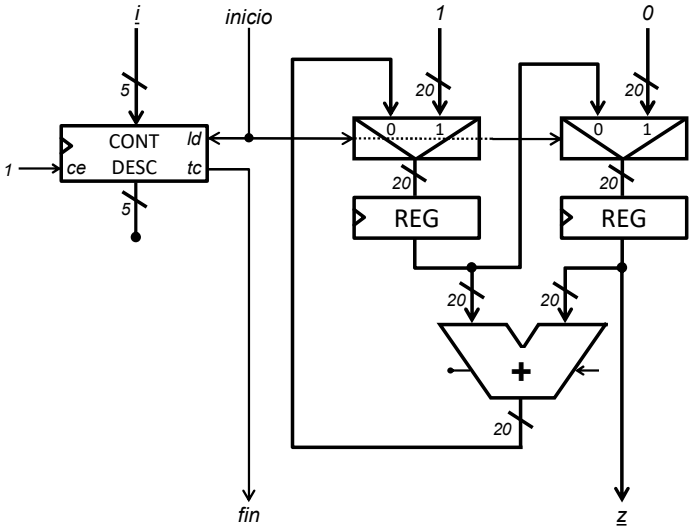

x
n
n
ov
+/−
op
n
REG
ld
ld
n
z

19.

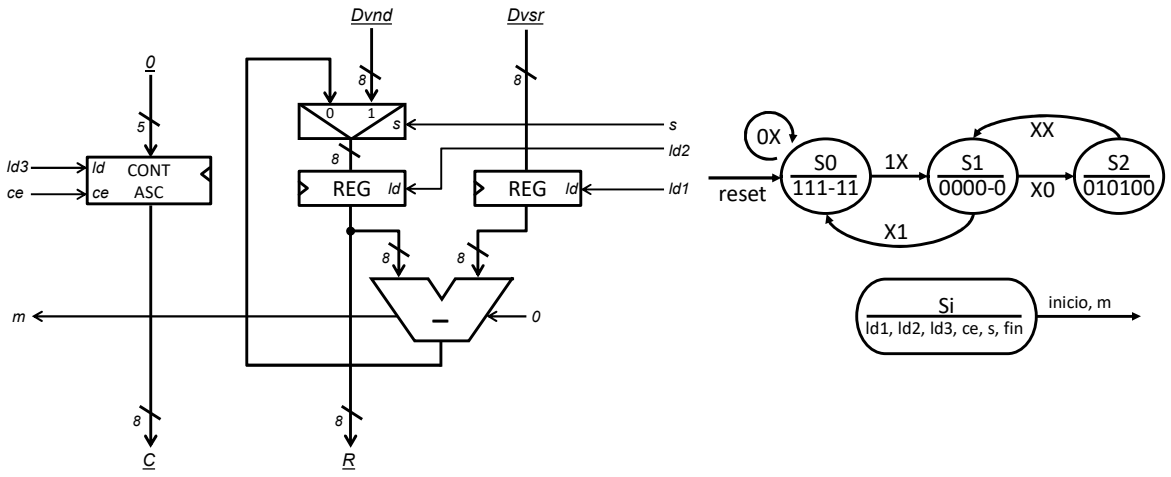

a
x
n
2n
REG
+
z
b
0


a
x
n
2n
b
REG
ld
1 0
s
0
z
0
b
x
3 2 1 0
2
s

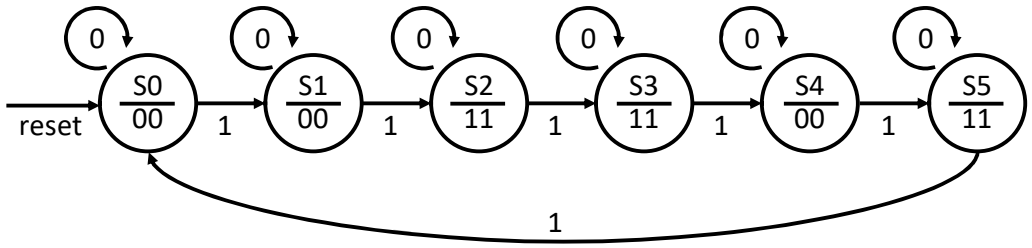
20.


i
inicio
1
0
5
20
CONT
DESC
ce
ld
tc
REG
fin
z

21.

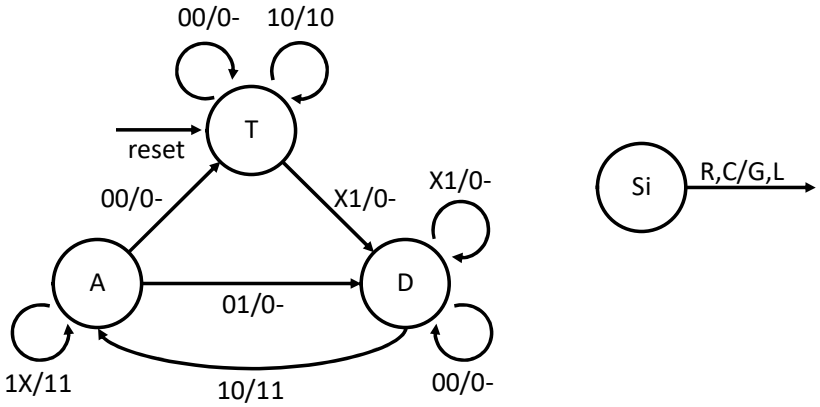


22.



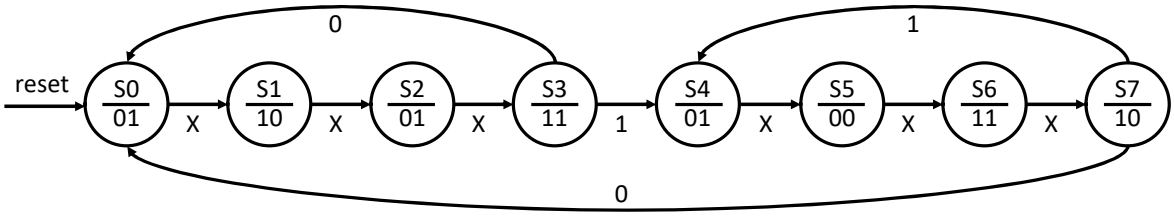
$C2(Q_2,Q_1,Q_0) = C1 = \sum m(2,3,5) + \sum d(6,7) = Q_1 + Q_2Q_0$
$ce(P,Q_2,Q_1,Q_0) = \sum m(8 \dots 12) + \sum d(6,7,13,14,15) = P$
$ld(P,Q_2,Q_1,Q_0) = \sum m(13) + \sum d(6,7,14,15) = PQ_2Q_1$
$e_2(P,Q_2,Q_1,Q_0) = e_1 = e_0 = \sum d(0 \dots 12,14,15) = 0$

23.



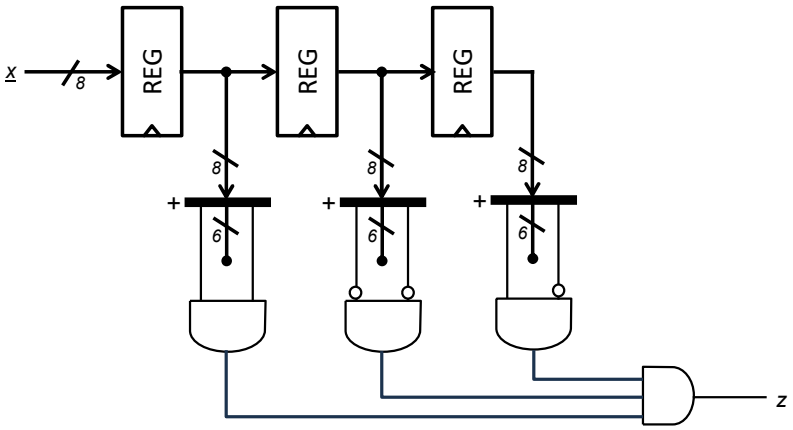
$\underline{Q} = \{00 = T, 01 = D, 10 = A\}$
$G(Q_1,Q_0,R,C) = \sum m(2,6,10,11) + \sum d(12 \dots 15) = Q_1R + R\bar{C}$
$L(Q_1,Q_0,R,C) = \sum m(6,10,11) + \sum d(0,1,3,4,5,7,8,9,12 \dots 15) = Q_1 + Q_0$
$ld(Q_1,Q_0,R,C) = \sum m(8,9) + \sum d(12 \dots 15) = Q_1\bar{R}$
$ce(Q_1,Q_0,R,C) = \sum m(1,3,6) + \sum d(8,9,12 \dots 15) = Q_0R\overline{C} + \overline{Q_1}\,\overline{Q_0}RC + \overline{Q_1}\,\bar{R}\,\bar{C}$
$e_1(Q_1,Q_0,R,C) = \sum d(0 \dots 7,10 \dots 15) = 0$
$e_0(P,Q_2,Q_1,Q_0) = \sum m(9) + \sum d(0 \dots 7,10 \dots 15) = C$

24.

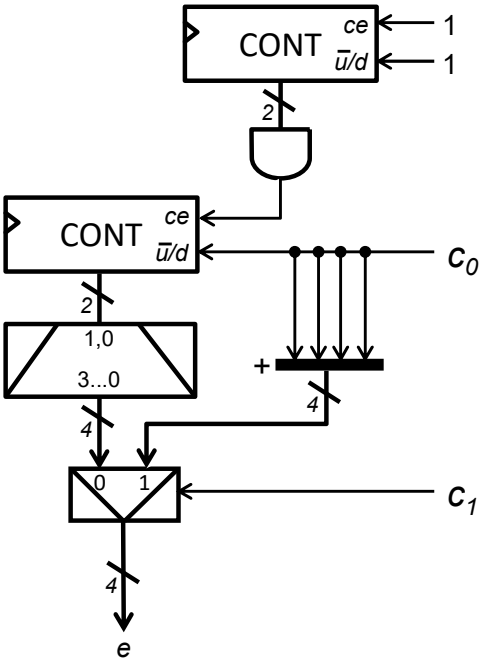
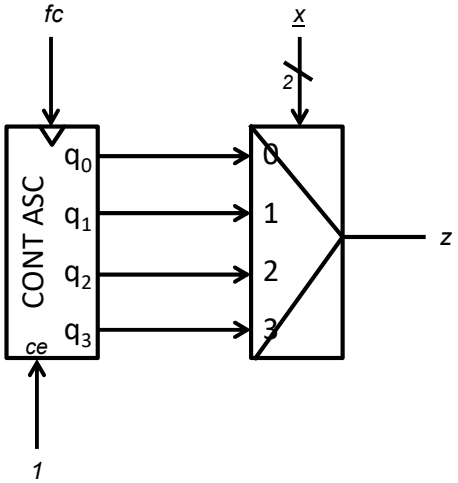


$t_1(Q_2,Q_1,Q_0) = \sum m(0,2,3,4,6) = Q_2Q_1 + \overline{Q_2}Q_0$
$t_0(Q_2,Q_1,Q_0) = \sum m(1,3,6,7) = \overline{Q_0} + \overline{Q_2}Q_1$
$ld(x,Q_2,Q_1,Q_0) = \sum m(3,15)$
$ce(x,Q_2,Q_1,Q_0) = \sum m(0,1,2,4 \dots 14) + \sum d(3,15) = 1$
$e_2(x,Q_2,Q_1,Q_0) = \sum m(15) + \sum d(0,1,2,4 \dots 14) = x$
$e_1(x,Q_2,Q_1,Q_0) = e_0 = \sum d(0,1,2,4 \dots 14) = 0$

25.



26.



27.