



ARQUITECTURA DE COMPUTADORAS

Trabajo Práctico N° 3 Circuitos Secuenciales

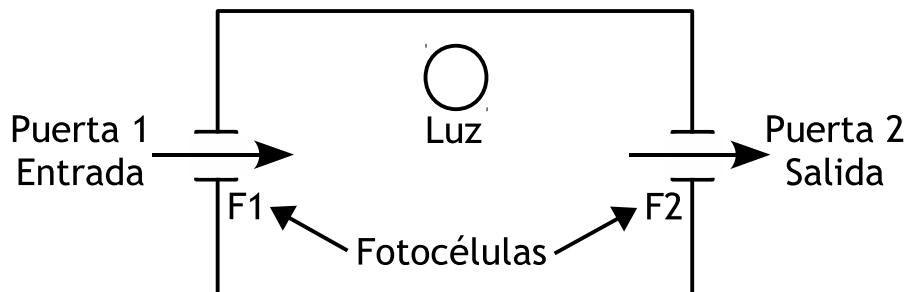
Primer Cuatrimestre de 2010

Ejercicios

1. En un cierto sistema de comunicaciones el comienzo de un nuevo mensaje se indica mediante la aparición de tres valores 1 consecutivos en una línea de entrada x , donde los datos en ésta se encuentran sincronizados con una fuente de reloj. Se desea diseñar un circuito secuencial con salida pulso z , tal que tenga pulso de salida justamente al comienzo de un nuevo mensaje (esto es, sólo en el tiempo de reloj que coincida con el tercero de una secuencia de tres valores 1 en la línea x). El propósito de este circuito es advertirle al sistema receptor acerca de la iniciación de un nuevo mensaje.

En este contexto, bosquejar el diagrama de estados y la tabla de estados correspondientes a este circuito.

2. En el contexto de un circuito donde hay dos pulsos de entrada x_1 y x_2 y uno de salida z , tal que el pulso de salida es coincidente con el segundo de dos pulsos consecutivos x_2 cuando ocurren a continuación de exactamente dos pulsos consecutivos x_1 :
 - a) Dibujar el diagrama de estados que representa el comportamiento del circuito anterior para luego construir la tabla de estados correspondiente.
 - b) Realizar una asignación de estados.
 - c) Obtener la expresión mínima de la función de salida z según la asignación de estados propuesta.
3. Para el próximo *Gran Hermano* se está diseñando un moderno confesionario, dispuesto como se indica en la siguiente figura:



En este confesionario, en lugar de contar con interruptores para la luz se han instalado sendas fotocélulas, una en cada puerta de acceso al mismo. El confesionario tiene una capacidad máxima de dos personas. Si una o dos personas están dentro, la luz debe permanecer encendida; caso contrario, debe estar apagada. Los participantes de la casa sólo pueden entrar al confesionario por la *Puerta 1* y salir por la *Puerta 2*. Además, las

entradas y las salidas nunca pueden ocurrir simultáneamente, ni tampoco puede salir una persona cuando no haya nadie en la habitación, ni puede entrar una tercera cuando ya haya dos personas dentro de la misma. Las fotocélulas se activan cada vez que su haz de luz es interrumpido por alguna persona ya sea entrando o saliendo. El resto del tiempo las señales de las fotocélulas son 0.

En este contexto:

- a) Dibujar el diagrama de estados que representa el comportamiento del circuito controlador de la luz, junto con su correspondiente tabla de estados.
- b) Reducir la cantidad de filas simplificando los estados equivalentes.
- c) Analizar la existencia de estados compatibles en este escenario.
- d) Realizar dos asignaciones de estados que resulten diferentes a nivel de implementación.

4. Considerando el circuito secuencial caracterizado por la siguiente tabla de estados,

<i>estado</i>	x_1	x_2	x_3	x_4	z_1	z_2
1	2	–	6	3	0	0
2	1	–	4	5	1	1
3	–	5	–	5	–	0
4	6	–	–	–	0	–
5	6	3	3	–	1	–
6	–	2	–	–	–	–

llevar adelante las siguientes tareas:

- a) Construir la *tabla de implicados* asociada a este circuito a fin de determinar los pares de estados que resultan compatibles entre sí.
 - b) Determinar si el conjunto de estados $\{1, 3, 4, 6\}$ es o no *maximalmente compatible*.
 - c) Calcular los restantes conjuntos de estados maximalmente compatibles mediante el método algebraico sobre los pares identificados como incompatibles.
 - d) Tomando como punto de partida el conjunto de conjuntos de estados maximalmente compatibles, encontrar al menos un conjunto con cuatro estados y otro con tres estados tales que que constituyan un cubrimiento cerrado de los estados del circuito original.
5. Se desea diseñar un circuito con dos niveles de entrada x_1 y x_2 y una señal de reloj, tal que un nivel de salida z se ponga a 1 con un pulso de reloj ocurriendo con $x_1 x_2 = 0 1$ y que z se ponga a 0 con el segundo de una secuencia de pulsos de reloj ocurriendo con $x_1 x_2 = 1 0$, siguiendo a un pulso de reloj ocurriendo con $x_1 x_2 = 0 1$. Ninguna otra secuencia de estados debe causar cambios en la salida.

En este escenario:

- a) Bosquejar el diagrama de estados y la tabla de estados del circuito.
- b) Realizar una asignación de estados.
- c) Determinar las funciones de excitación para la implementación con flip-flops **D**.
- d) Calcular la expresión mínima de la función de salida z .

6. Minimizar la cantidad de filas de las siguientes tablas de transición de estados para luego realizar una asignación de estados. Finalmente, determinar si el proceso de minimización permitió economizar la cantidad de elementos de memoria requeridos por estos circuitos.

a)

<i>estado</i>	x_1	x_2	z
1	3	2	0
2	3	1	0
3	5	4	0
4	3	4	1
5	1	2	1

b)

<i>estado</i>	x_1	x_2
1	4/0	7/0
2	5/0	3/1
3	5/0	2/1
4	3/0	1/0
5	6/0	1/0
6	4/0	6/0
7	5/0	3/0

7. Emular los siguientes dispositivos:

- Un flip-flop **S-R** a partir de un flip-flop **D**.
- Un flip-flop **J-K** a partir de un flip-flop **D**.
- Un flip-flop **D** a partir de un flip-flop **T**.
- Un flip-flop **J-K** a partir de un flip-flop **T**.
- Un flip-flop **D** a partir de un flip-flop **J-K**.

8. Implementar un contador binario síncrono módulo 10 (esto es, de 0 a 9) a partir de los siguientes componentes:

- Con flip-flops individuales tipo **J-K**, controlando sus entradas.
- Con un contador de cuatro etapas *up-down* con carga en paralelo.

9. Un *contador Moebius* (esto es, un contador que que modifica un solo bit por cada cambio de estado), puede ser fácilmente implementado usando un registro de desplazamiento. Teniendo esto en cuenta:

- Bosquejar un *contador Moebius* módulo seis a partir de un registro de desplazamiento implementado con flip-flops **J-K**.
- Desarrollar las seis señales periódicas de salida.
- Verificar que dividen el período de partida en seis intervalos iguales y que cada señal tiene un *duty cycle* de exáctamente 50 %.

OBS: En un tren de pulsos ideal (uno compuesto de pulsos rectangulares), el *duty cycle* es la razón entre la duración del pulso y el período del mismo. Por caso, para un tren de pulsos en el que la duración del pulso es de $1\mu s$ y el período es de $4\mu s$, el *duty cycle* resulta ser del 25 %.

- Indicar como se decodifican los estados 001, 110, 000 y 111.
- Suponiendo que la transición del estado 011 al estado 001 recién deba realizarse cuando aparezca un 1 en una cierta señal e (esto es, e hace las veces de señal de *enable* para esa transición), ¿cómo hay que modificar al contador para que contemple a la señal e ?
- Modificar el control sobre la transición entre estados propuesta en el inciso anterior para que la señal e ahora regule el paso del estado 110 al estado 111.

10. Un inventor produce un dispositivo de memoria con una única entrada, el cual satisface las siguientes propiedades:

- Cuando la línea de entrada I es 0 en el tiempo n , el dispositivo pasará al estado 0 en el tiempo $n + 1$.
- Cuando la línea de entrada I es 1 en el tiempo n , el dispositivo cambiará de estado en el tiempo $n + 1$.

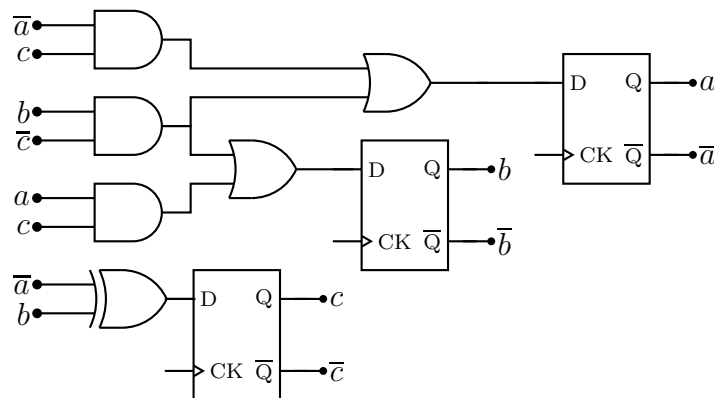
Este inventor desea construir un contador con su dispositivo, usando dos instancias Q_1 y Q_2 del mismo, para que cuente la siguiente secuencia:

Q_1	Q_2
0	0
0	1
1	0
0	0

Teniendo en cuenta el funcionamiento de este dispositivo:

- a) ¿Cómo tienen que ser las entradas I_{Q_1} I_{Q_2} ?
- b) ¿Qué se le debería advertir al inventor sobre su dispositivo?

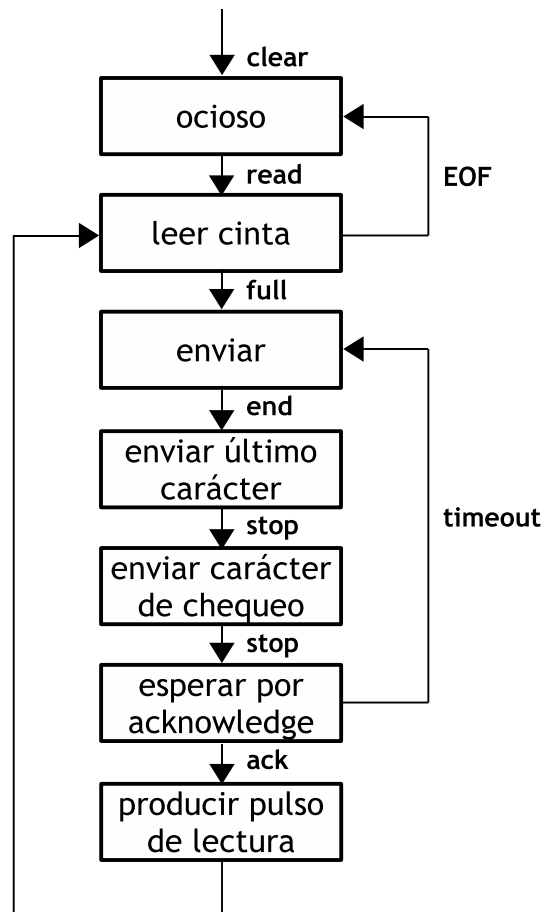
11. Dada la siguiente mecanización, utilizando flip-flops **D**, encontrar la secuencia de salida que se produce partiendo de un estado inicial 000.



12. Usando un contador binario *up-down* con carga paralela, implementar los estados de control para un sistema que luego de leer registros de datos desde una cinta magnética en un buffer, los envíe a través de una línea de transmisión de datos, junto con un carácter de chequeo. Entre la transmisión de un registro y otro, debe esperar por una señal de recepción correcta **ack**. Si no se recibe esta señal dentro de un cierto tiempo **timeout**, el mensaje es repetido. Este ciclo se repite indefinidamente, hasta que se recibe el **ack**.

El diagrama de estados correspondiente al sistema es el siguiente:

A partir del estado inicial (**ocioso**), ante una señal de lectura (**read**) se alcanza el estado **leer cinta**. Por otro lado, si se detecta fin de archivo (**EOF**), denotando que no hay más información en la cinta, se retorna al estado **ocioso**. Cuando el buffer está lleno (**full**) con los datos obtenidos de la cinta, se pasa al estado de **enviar**, y se transmite la información hasta que la señal **end** indique que se ha llegado al último dato en el buffer.



En ese momento se avanza al estado de **enviar último carácter**. Cuando se ha enviado el último bit del último dato, un bit de **stop** pone al sistema en condiciones de **enviar carácter de chequeo**. Luego, cuando se ha enviado el último bit del código de chequeo, una nueva señal de **stop** provoca que el sistema avance a **esperar por ack**. Si se recibe un **ack**, se entra en el estado de **producir un pulso de lectura**, el cual, con un pulso de reloj, genera una señal de **read** que causa que se vuelva a leer otro registro desde la cinta, en el estado **leer cinta**.

Esta secuencia se repite hasta que todos los registros son enviados, y una señal de **EOF** retorna al sistema a estado **ocioso**. Si no se recibe un **ack** dentro de $\frac{1}{3}$ de segundo de haber entrado en el estado **esperar por ack**, la señal **timeout** hace que el registro sea retransmitido.

Referencias

- [Bla79] BLAKESLEE, T. R. *Digital Design with Standard MSI and LSI*, second ed. John Wiley & Sons, 1979.
- [Mar76] MARCUS, M. P. *Switching Circuits for Engineers*, third ed. Prentice-Hall, 1976.