

Práctica 5: Máquinas de estado y carta ASM

Arnold Torres Maldonado

Resumen—En esta práctica el objetivo fue diseñar e implementar el sistema de seguridad de una caja fuerte. Después diseñar e implementar en lenguaje VHDL un semáforo usando máquinas de estado. Aquí se plasman los conceptos básicos y razonamientos que hicieron posible completar esta tarea.

I. INTRODUCCIÓN

Las máquinas de Mealy y Moore son circuitos síncronos. Un circuito síncrono es un circuito digital en el cual sus partes están sincronizadas por una señal de reloj.

En un circuito síncrono ideal, cada cambio en los diferentes niveles lógicos es simultáneo. Estas transiciones se realizan después de un cambio de nivel de una señal llamada reloj. Idealmente la entrada a cada elemento de almacenamiento alcanza su valor final antes de que la siguiente señal de reloj ocurra, por lo tanto, el comportamiento de un circuito se puede predecir exactamente. Se requiere de cierto retardo para cada operación lógica, por lo que existe una máxima rapidez en el que cada sistema síncrono puede responder. El análisis de un diagrama de tiempos puede darnos esta rapidez.

Una máquina de Mealy es una máquina de estados finita, donde las salidas están determinadas por el estado actual y la entrada. Esto significa que en el diagrama de estados se incluye una señal de salida para cada arista de transición. Por ejemplo, en la trayectoria de un estado 1 a un estado 2, si la entrada es cero la salida puede ser uno, y se debe poner sobre la arista la etiqueta “0/1”.

En contraste, la salida de una máquina de estado finito Moore (máquina de Moore), depende solo del estado actual y no depende de la entrada actual. Por lo tanto, los estados de una máquina de Moore son la unión de los estados de la máquina de Mealy y el producto cartesiano de estos estados y alfabeto de entrada (posibles entradas).

Un diagrama de estados nos muestra gráficamente la secuencia de estados en un sistema y las condiciones de cada estado y de las transiciones entre cada uno de ellos. Por otra parte, el diagrama ASM es muy similar a un diagrama de estados. Cada cuadro de estado equivale a un estado de un circuito secuencial. El cuadro de decisión equivale a la información binaria que se escribe sobre las flechas que conectan dos estados de un diagrama de estados. Por ello, hay ocasiones en que conviene convertir el diagrama ASM en un diagrama de estados y luego usar procedimientos de circuitos secuenciales para diseñar la lógica de control. Considerar que:

- Los tres estados se indican con círculos, con su valor binario escrito adentro.
- Las flechas indican las condiciones que determinan el siguiente estado.
- Las operaciones incondicionales y condicionales que deben efectuarse no se indican en el diagrama de estados.

II. DESARROLLO DE LA PRÁCTICA

II-A. Diseño del sistema de seguridad de la caja fuerte

A partir de los requerimientos se llegó a la conclusión de que existían los estados y transiciones presentados en la Figura 1. Todo se trató de detectar una secuencia de pulsaciones en determinado momento.

El estado A hace referencia al estado inicial donde NO hay código. El B es el estado donde se recibe el primer uno, seguido del C donde ya se han recibido dos unos. Por último, pasamos a D donde se ha detectado que la secuencia se ha introducido de manera correcta.

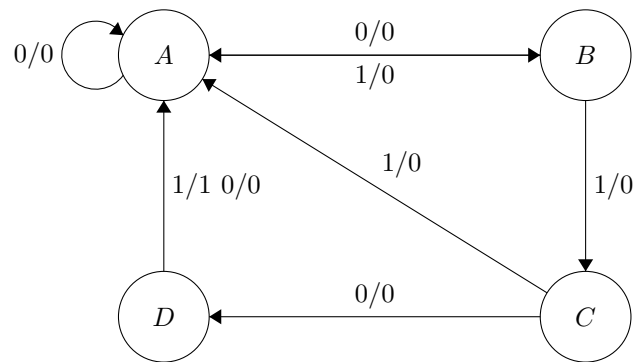


Figura 1. Diagrama de estados del sistema de seguridad

El diagrama de estados se transformó en la tabla de estados del Cuadro 1.

Cuadro I
TABLA DE ESTADOS

Estado siguiente	Estado presente		Salida	
	$w = 0$	$w = 1$	$w = 0$	$w = 1$
$y_1 y_0$	$Y_1 Y_0$	$Y_1 Y_0$	z_1	z_0
$A = 00$	$A = 00$	$B = 01$	0	0
$B = 01$	$A = 00$	$C = 10$	0	0
$C = 10$	$D = 11$	$A = 00$	0	0
$D = 11$	$A = 00$	$A = 00$	0	1

Luego se sacaron las expresiones Y_0 , Y_1 , z_0 y z_1 usando la técnica de minimización del mapa K, cuyo procedimiento se muestra en el Apéndice A. Las expresiones minimizadas son:

$$Y_0 = y_1 \overline{y_0} w + \overline{y_1} \overline{y_0} w \quad (1)$$

$$Y_1 = y_1 \overline{y_0} w + \overline{y_1} y_0 w \quad (2)$$

$$z_0 = y_1 y_0 w \quad (3)$$

$$z_1 = 0 \quad (4)$$

Para terminar, el diagrama ASM (Fig. 2) que se generó a partir del diagrama de estados (Fig. 1) facilitó describir el funcionamiento del circuito en VHDL para implementarlo en el FPGA. Lo único que se programó y que no se encuentra en la tabla es el contador de cinco minutos requerido para volver a activar el sistema de seguridad cuando este llega a cero.

II-B. Diseño del semáforo

Partiendo de los estados de la tabla mostrada en el Cuadro II, se llegó a la tabla mostrada en el Cuadro III, que se simplificaron y acondicionaron los nombres de los estados y las salidas para poder obtener las ecuaciones por minimización, lo que luego se plasma en el diagrama de estados de la Figura 2.

Cuadro II
ESTADOS/TIEMPOS ESPECIFICADOS PARA EL SEMÁFORO

Estado	Norte - Sur	Este - Oeste	Retraso (en segundos)
$A = 0$	Verde	Rojo	5
$B = 1$	Amarillo	Rojo	1
$C = 2$	Rojo	Rojo	1
$D = 3$	Rojo	Verde	5
$E = 4$	Rojo	Amarillo	1
$F = 5$	Rojo	Rojo	1

En la siguiente tabla la salida z se determinó al pensar en las luces que determinado semáforo debía tener encendidas en determinado momento. Las salidas z_6, z_5 y z_4 fueron para el semáforo Norte-Sur y z_3, z_2, z_1 fueron para el semáforo Este-Oeste. Los últimos dos estados fueron puestos a porque son necesarios $2^3 = 8$ flip-flops porque existen 3 dígitos para representar los nombres de cada estado. El valor de estos estados no influye en el comportamiento del circuito.

Cuadro III
TABLA DE ESTADOS DEL SEMÁFORO

Estado presente	Estado siguiente		Salida z
	$w = 0$	$w = 1$	
$y_3y_2y_1$	$Y_3Y_2Y_1$	$Y_3Y_2Y_1$	$z_6z_5z_4z_3z_2z_1$
000	001	000	100001
001	010	001	010001
010	011	010	001101
011	100	011	001100
100	101	100	001010
101	000	101	001001
110	xxx	xxx	xxxxxx
111	xxx	xxx	xxxxxx

De la tabla de estados se obtuvieron los valores de Y_3, Y_2, Y_1 y z_6, z_5, \dots, z_1 haciendo los mapas de Karnaugh correspondientes (ver Apéndice B).

$$Y_3 = y_2y_1\bar{w} + y_3\bar{y}_1 + y_3w \quad (5)$$

$$Y_2 = \bar{y}_3\bar{y}_2y_1\bar{w} + y_2\bar{y}_1 + y_2w \quad (6)$$

$$Y_1 = \bar{y}_1\bar{w} + y_1w \quad (7)$$

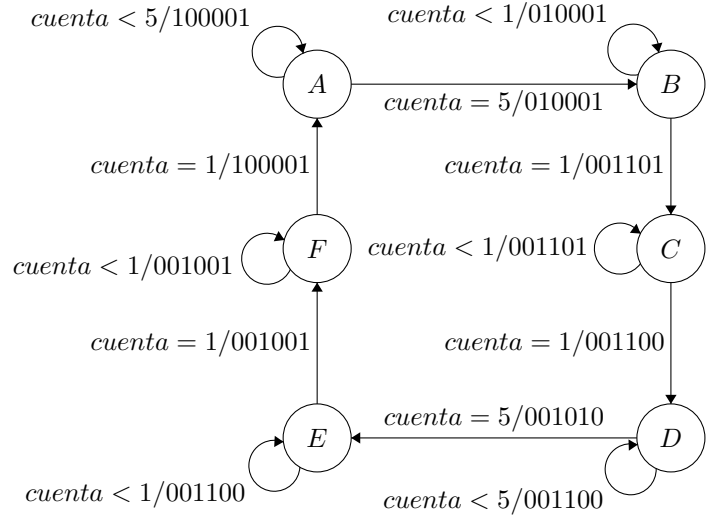


Figura 2. Diagrama de estados del semáforo

$$z_6 = \bar{y}_3y_2y_1 \quad (8)$$

$$z_5 = \bar{y}_3y_2y_1 \quad (9)$$

$$z_4 = \bar{y}_3y_2 + y_3\bar{y}_2 \quad (10)$$

$$z_3 = \bar{y}_3y_2 \quad (11)$$

$$z_2 = y_3\bar{y}_2y_1 \quad (12)$$

$$z_1 = \bar{y}_3\bar{y}_1 + \bar{y}_2y_1 \quad (13)$$

III. SIMULACIONES

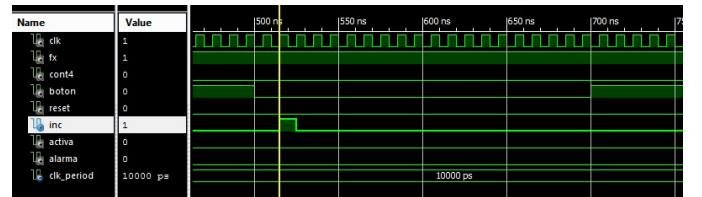


Figura 3. Simulación del sistema de seguridad



Figura 4. Simulación del semáforo

IV. CONCLUSIONES

Si definimos el comportamiento de un circuito como un sistema que va saltando de un estado a otro según ciertas condiciones facilita su implementación en VHDL. Esto es gracias a que en VHDL existen las señales de tipo estado y solo hace falta meter las condiciones para cambiar entre estos y las consecuencias que tiene el cambio a un CASE dentro de un PROCESS.

Veo que hacer una máquina de estado es una manera distinta de abordar un tipo de problema a mera conveniencia, esta manera de ver las cosas puede facilitar el comprender el funcionamiento de un algoritmo.

		w	
		0	1
y_1y_0	00	0	0
	01	0	0
	11	0	1
	10	0	0

Figura 7. Mapa para z_0

APÉNDICE A

MINIMIZACIÓN DE LA TABLA DE ESTADOS DEL SISTEMA DE SEGURIDAD

		w	
		0	1
y_1y_0	00	0	0
	01	0	1
	11	0	0
	10	1	0

Figura 5. Mapa para Y_0

		w	
		0	1
y_1y_0	00	0	1
	01	0	0
	11	0	0
	10	1	0

Figura 6. Mapa para Y_1

APÉNDICE B

MINIMIZACIÓN DE LA TABLA DE ESTADOS DEL SEMÁFORO

		y_1w			
		00	01	11	10
y_3y_2	00	0	0	0	0
	01	0	0	0	1
	11	-	-	-	-
	10	1	1	1	0

Figura 8. Mapa para Y_3

		y_1w			
		00	01	11	10
y_3y_2	00	0	0	0	1
	01	1	1	1	0
	11	-	-	-	-
	10	0	0	0	0

Figura 9. Mapa para Y_2

		$y_1 w$			
		00	01	11	10
$y_3 y_2$	00	1	0	1	0
	01	1	0	1	0
	11	-	-	-	-
	10	1	0	1	0

Figura 10. Mapa para Y_1

		y_1	
		0	1
$y_3 y_2$	00	1	1
	01	1	0
	11	0	0
	10	0	1

Figura 13. Mapa para z_1

REFERENCIAS

- [1] T. L. Floyd, *Fundamentos de Sistemas Digitales* (9a. Ed.). Pearson Educacion, 2007.
- [2] D. D. Gajski y P. G. Carlos, *Principios de Diseno Digital*:. Prentice Hall, 2000.

		y_1	
		0	1
$y_3 y_2$	00	0	0
	01	1	1
	11	0	0
	10	1	1

Figura 11. Mapa para z_4

		y_1	
		0	1
$y_3 y_2$	00	0	0
	01	1	1
	11	0	0
	10	0	0

Figura 12. Mapa para z_3