

MAQUINAS DE ESTADO

1. DEFINICION

Son ciertos circuitos secuenciales que tienen un número determinado de estados (2^n). Pueden ser retroalimentados (flip flops, biestables) o máquinas síncronas temporizadas cuando utilizan las primeras para crear circuitos cuyas entradas son examinadas y cuyas salidas cambian con respecto a una señal de reloj controlada. En cualquier caso, se tienen unas entradas, unas salidas y unos estados.

2. ESTRUCTURA

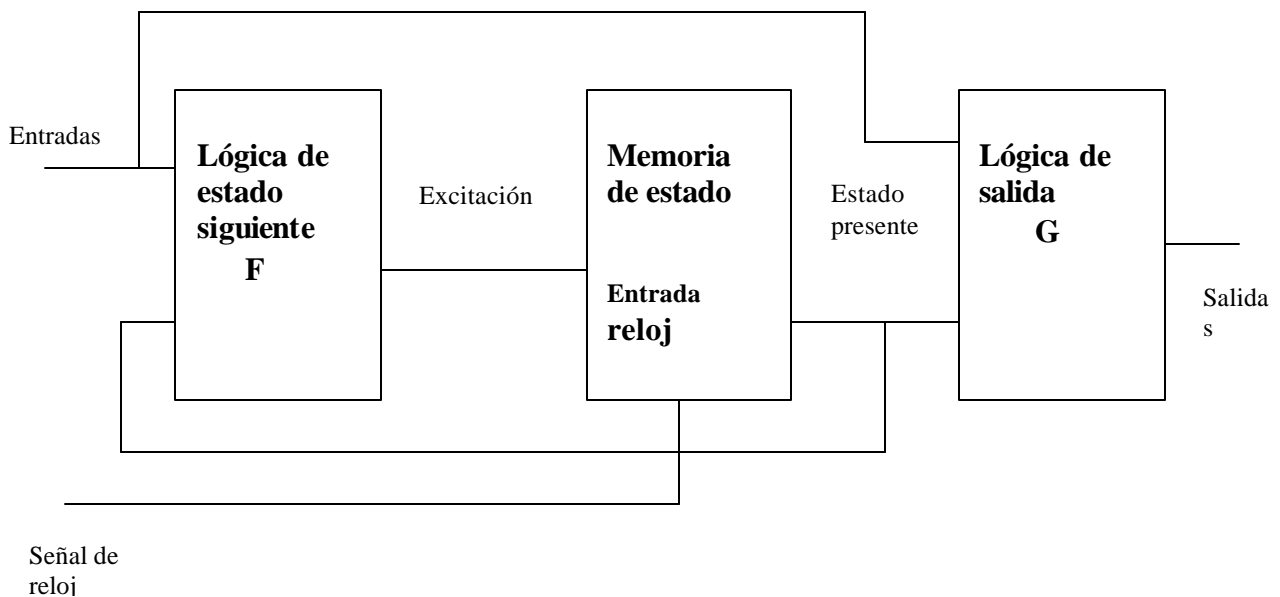


Fig 1. Estructura máquina de estados sincronizada por reloj(Mealy)

- ? **Lógica de estado siguiente(F):** Una función de las entradas y del estado actual.
- ? **Memoria de estados:** Es un conjunto de n flip flops que almacenan el estado presente de la máquina, que tiene 2^n estados diferentes. La señal de reloj controla el cambio de estado en tales flip flops.
- ? **La señal de reloj:** dispone el funcionamiento de los flip flops ya sea por disparo de flanco o por disparo de pulso
- ? **Lógica de salida(G):** Una función del estado actual y/o de las entradas

2.1. Máquina de Mealy: Es la máquina de estado en la cuál la salida depende tanto del estado presente como de las entradas externas. (Es el representado en la figura 1).

2.2. Máquina de Moore: Es la máquina de estado en la cual las salidas solo dependen del estado presente. Su estructura se muestra en al figura 2.

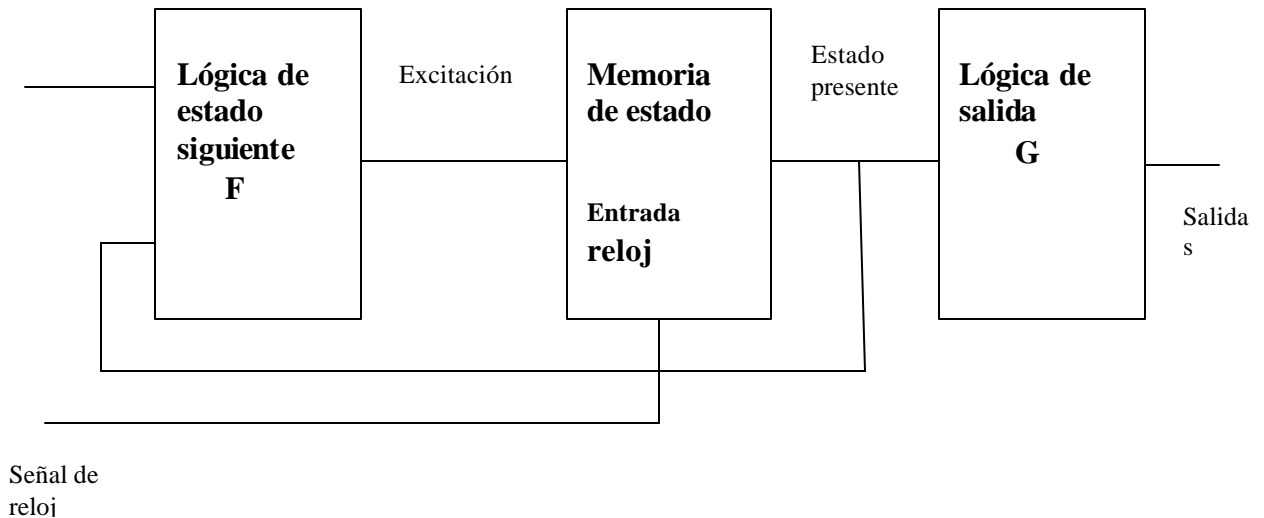


Fig. 2 Estructura máquina de Moore

3. ANALISIS CON FLIP FLOPS TIPO D

Asumiendo la definición formal de máquina de Mealy:

$$\begin{aligned}\text{Estado siguiente} &= F(\text{estado actual}, \text{entrada}) \\ \text{Salida} &= G(\text{estado actual}, \text{entrada})\end{aligned}$$

La primera ecuación dice que el estado siguiente estará determinado por la entrada presente y el estado presente del circuito.

La segunda precisa que la salida de la máquina la determinan las dos mismas variables.

El propósito del análisis de los circuitos secuenciales es determinar el estado siguiente y las funciones de salida para poder predecir el comportamiento del circuito.

Este análisis tiene tres pasos básicos:

- (1) Determinar el estado siguiente y las funciones de salida F y G.
- (2) Usar F y G para construir una tabla de *estados/salidas* que especifique por completo el estado siguiente y la salida del circuito para cada combinación posible de estado actual y entrada.

- (3) Diagrama de estados que presente en forma gráfica la información anterior(opcional).

4. ANALISIS CON FLIP FLOPS TIPO JK

5. DISEÑO DE MAQUINAS DE ESTADO SINCRONIZADAS CON RELOJ

Los pasos para diseñar una máquina de estados sincronizada con reloj, comenzando con una descripción verbal o especificación escrita, son los contrarios de los pasos de análisis.

- (1) Construir una tabla de estado/salida correspondiente a la descripción o especificación, mediante nombres mnemotécnicos para los estados. (Puede partirse del diagrama de estados correspondiente.
- (2) Minimizar el número de estados en la tabla de estado/salida(opcional)
- (3) Elegir un conjunto de variables de estado y asignar combinaciones de variables de estado a cada uno de los estados.
- (4) Sustituir las combinaciones de variable de estado en la tabla de estado/salida para crear una tabla de transición/salida que muestre la combinación de variable de estado siguiente y la salida para cada combinación de estado/salida
- (5) Elegir el tipo de flip flop que hará la memoria de estado.
- (6) Construir una tabla de excitación que muestre los valores de excitación requeridos para obtener el estado siguiente deseado para cada combinación de estado/entrada.
- (7) Derivar las ecuaciones de excitación de la tabla de excitación
- (8) Derivar las ecuaciones de salida de la tabla de transición/salida
- (9) Dibujar el diagrama lógico que muestre los elementos de almacenamiento de las variables de estado y realice las ecuaciones requeridas de excitación y salida.

5.1. Diseño de tablas de estado

El proceso es semejante a escribir un programa de computador:

- ? Descripción de entradas y salidas, más bien precisa, pero con descripción ambigua de las relaciones deseadas entre ellas y sin ninguna idea de cómo obtener y sin ninguna idea de cómo obtener en realidad las salidas deseadas a partir de las entradas.
- ? Determinación de procedimiento de diseño
- ? Nuevas identificaciones y manejo de elementos no tenidos en cuenta en el inicio
- ? Prueba

? Depuración.

Ejemplo.

Diseñar una máquina de estados sincronizada con reloj con dos entradas A y B, y una sola salida Z. La salida debe ser uno sí A tuvo el mismo valor en las dos pulsaciones previas del reloj, o sí B ha sido uno desde la última vez que fue cierta la primera condición. De lo contrario la salida debe ser cero.

? Construir plantilla para la tabla de estado.

SIGNIFICADO	S	AB				Z
		00	01	11	10	
Estado inicial	INI					0

Tabla 1

De la descripción dada se sabe que es una máquina de Moore pues la salida solo depende del estado presente, es decir, de los periodos previos del reloj. En este caso este caso la plantilla es la de la tabla 1, sí hubiese sido de Maely no habría columna de salidas y los estados siguientes se colocarían al lado de cada combinación de entrada.

? Primer llenado de la plantilla

5.2. Minimización de estados

5.3. Asignación de estados

Asumamos como alternativas de asignación las siguientes:

Nombre del estado	Asignación			
	El más simple	Compuesta Q1-Q3	Un solo uno Q1-Q5	Casi un 1 Q1-Q4
INI	000	000	00001	0000
A0	001	100	00010	0001
A1	010	101	00100	0010
OK0	011	110	01000	0100
OK1	100	111	10000	1000

Se desarrollará el ejemplo con las dos primeras alternativas, la primera por obvia, cumplir con el requisito de minimizar el número de variables de estado que cambian y maximizar el número de variables de estado que no cambian en un grupo de estados relacionados la otra por que, además, facilita el análisis del circuito.

SIGNIFICADO	S	AB				Z
		00	01	11	10	
Estado inicial	INI	A0	A0	A1	A1	0
A=0	A0	OK0	OK0	A1	A1	0
A=1	A1	A0	A0	OK1	OK1	0
A 00	OK0	OK0	OK0	OK1	A1	1
A 11	OK1	A0	OK0	OK1	OK1	1
S*						

Con la primera opción los estados quedarán:

S	Asignación
INI	000
A0	001
A1	010
OK0	011
OK1	100

Así que reemplazando la tabal de estado/salida quedaría

SIGNIFICADO	S	AB				Z
		00	01	11	10	
Estado inicial	000	001	001	010	010	0
A=0	001	011	011	010	010	0
A=1	010	001	001	100	100	0
A 00	011	011	011	100	010	1
A 11	100	001	011	100	100	1
S*						

Como los estados están controlados por la memoria de estados

Con la segunda opción

S	Asignación
INI	000
A0	100
A1	101
OK0	110
OK1	111

SIGNIFICADO	S	AB				Z
		00	01	11	10	
Estado inicial	000	100	100	101	101	0
A=0	100	110	110	101	101	0
A=1	101	100	100	111	111	0
OK0	110	110	110	111	101	1
OK1	111	100	110	111	111	1
S*						

- 5.4. Sustituir las combinaciones de variable de estado en la tabla de estado/salida para crear una tabla de transición/salida que muestre la combinación de variable de estado siguiente y la salida para cada combinación de estado/salida

5.5.

5.6. Síntesis con flip flops tipo D

5.7. Síntesis con flip flops tipo JK