

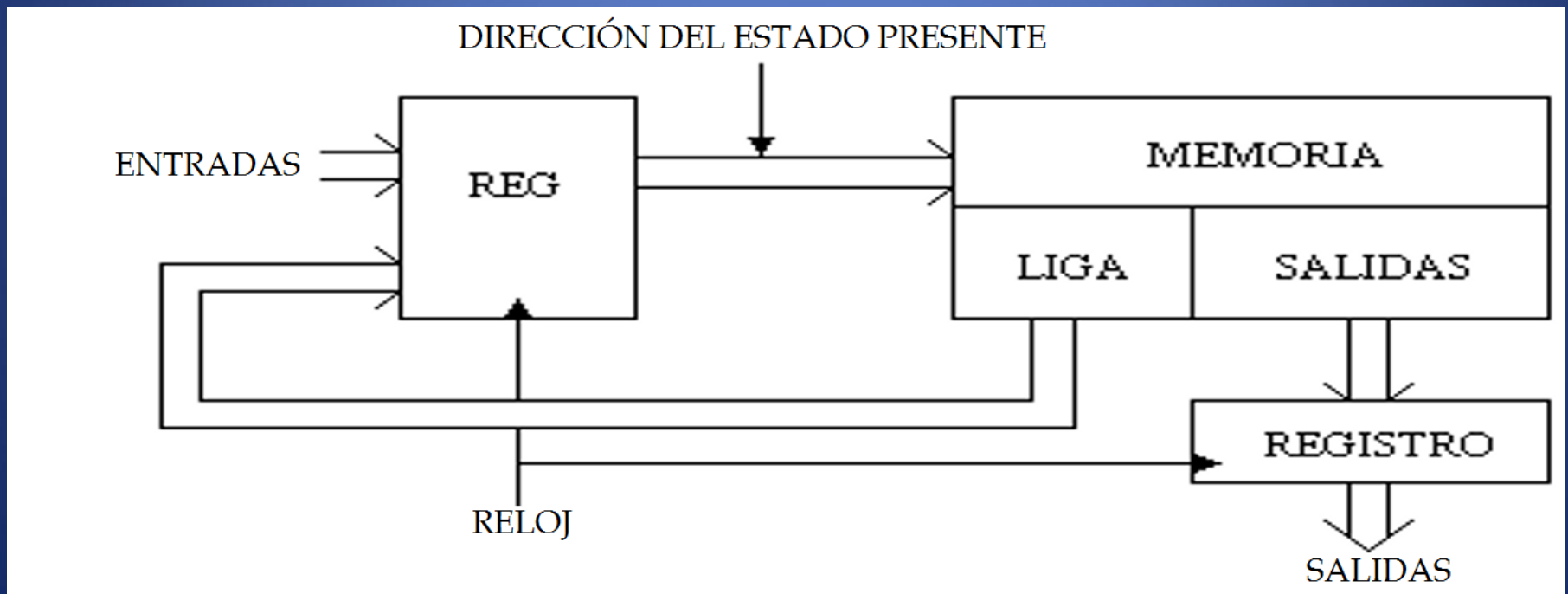
# CONSTRUCCIÓN DE MÁQUINAS DE ESTADOS USANDO MEMORIAS

Se explicará cómo instrumentar los algoritmos de máquinas de estados en configuraciones físicas de componentes ya definidos e independientes de los algoritmos a ser ejecutados.

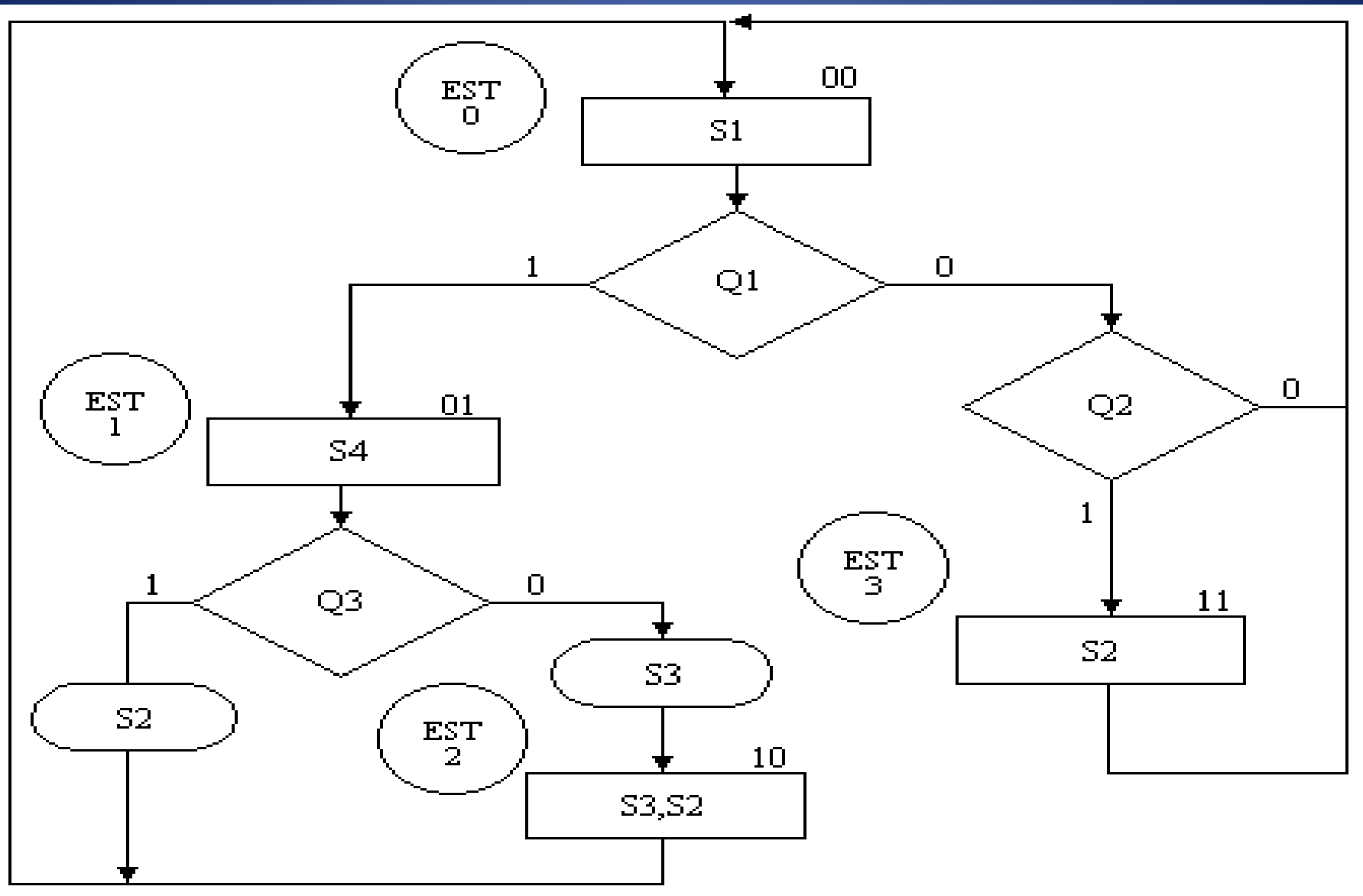
En particular, se construirán máquinas de estados utilizando memorias.

# DIRECCIONAMIENTO POR TRAYECTORIA

Este tipo de direccionamiento guarda el estado siguiente y las salidas de cada estado de la carta ASM en una localidad de memoria. La porción de la memoria que indica el estado siguiente es llamada "LIGA", mientras que la porción que indica las salidas es llamada "SALIDAS". Concatenando la liga junto con las entradas, en un registro que está conectado a las líneas de dirección de la memoria se forma la dirección de memoria que contiene la dirección del estado presente.



# EJEMPLO



# SOLUCIÓN

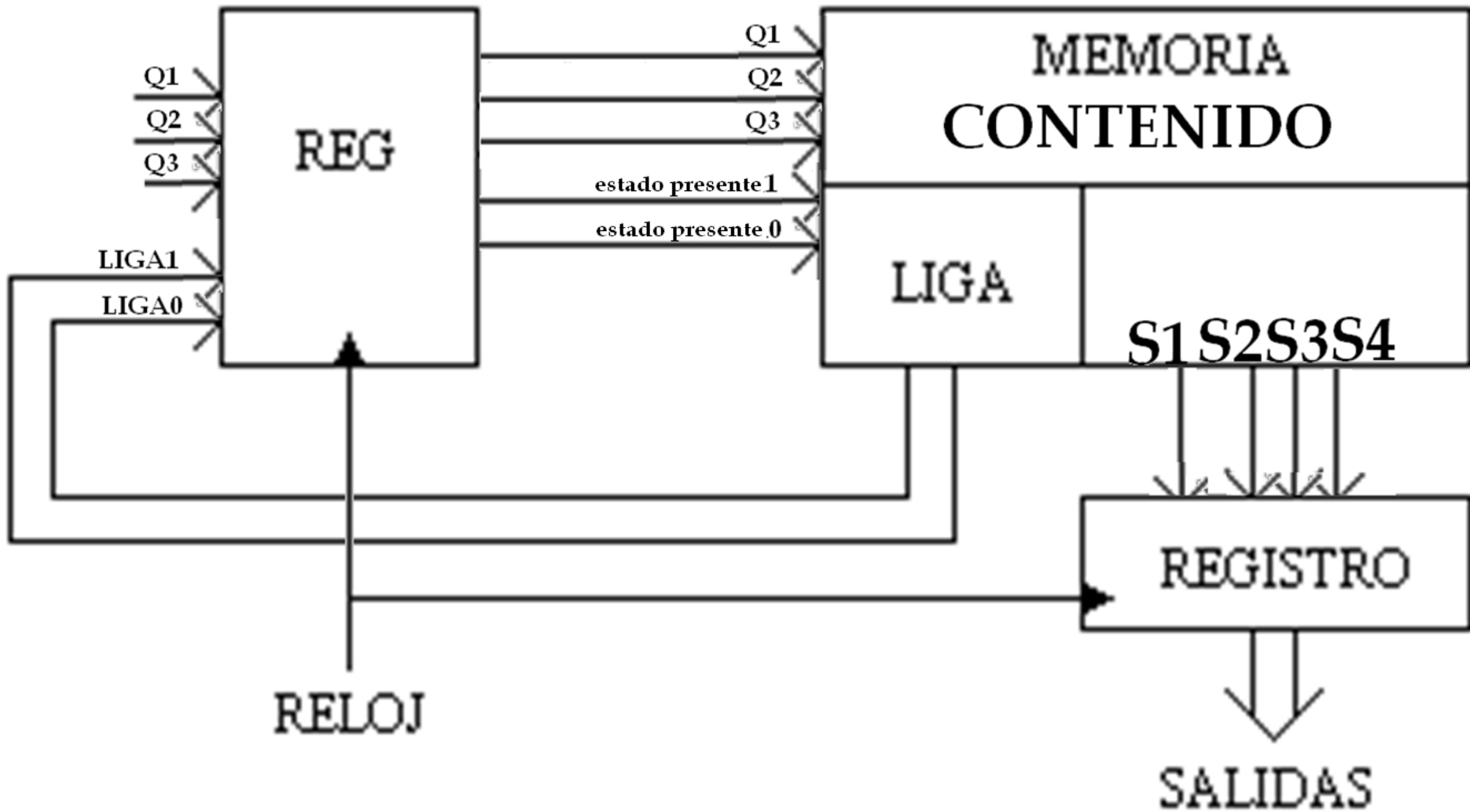
Para cada estado es necesario considerar todas las posibles combinaciones de las variables de entrada, aún cuando algunas de ellas no se utilicen. Por ejemplo, si en el estado EST0 la variable Q1 es igual a 1, la máquina de estados pasará al estado EST1 independientemente de los valores de las otras variables. Aún así, se deben considerar todas las combinaciones de las otras variables de entrada y colocar en las localidades de memoria correspondientes los valores adecuados.

Cuando una salida se activa, el bit correspondiente de memoria se coloca a 1, en caso contrario se coloca a 0.

# CONTENIDO MEMORIA

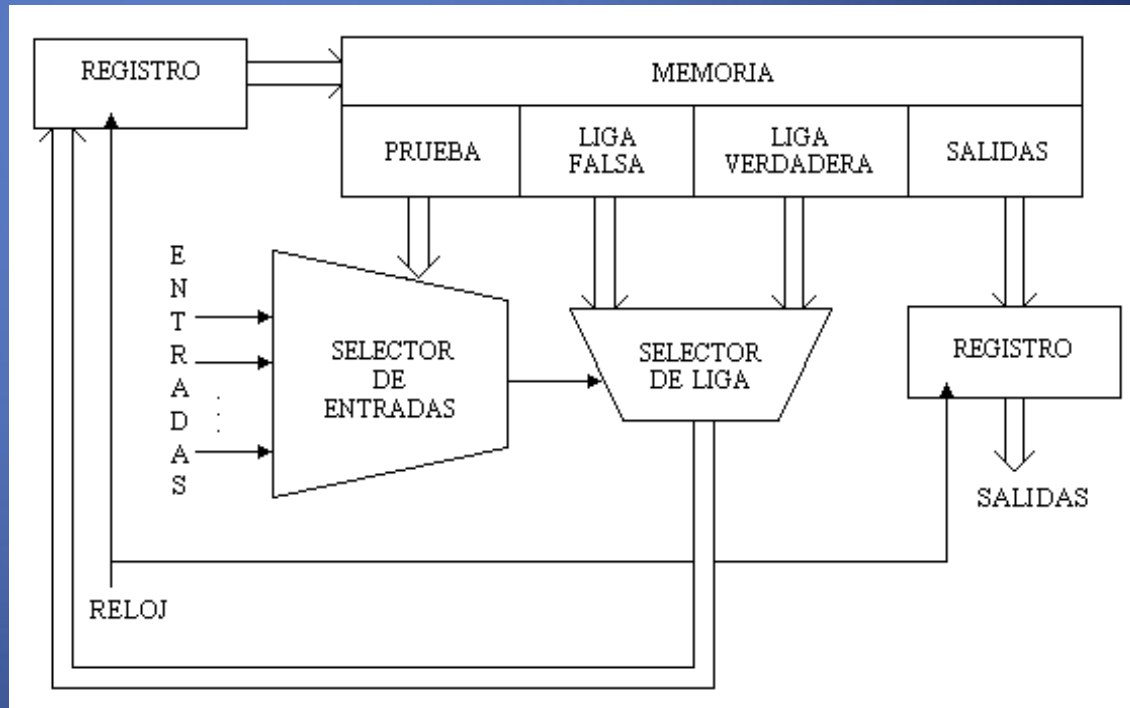
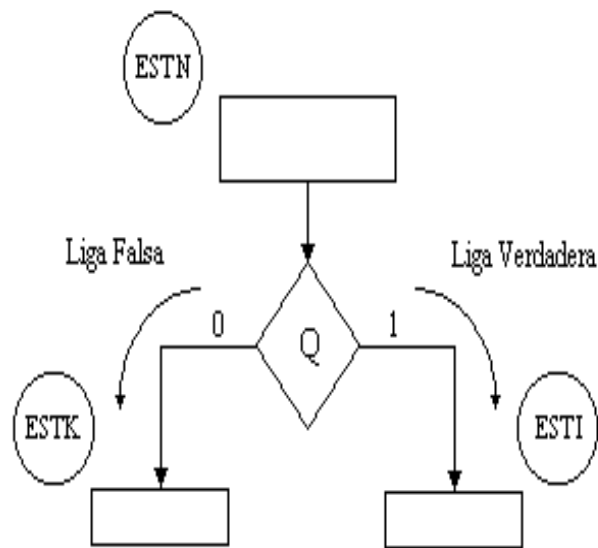
		Dirección de Memoria			Contenido de Memoria					
Edo. Presente		Entradas			Ligas		Salidas			
		Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>	Liga <sub>1</sub>	Liga <sub>0</sub>	S <sub>1</sub>	S <sub>2</sub>	S <sub>3</sub>	S <sub>4</sub>
EST0	0 0	0	0	0	0	0	1	0	0	0
	0 0	0	1	0	1	1	1	0	0	0
	0 0	0	1	1	1	1	1	0	0	0
	0 0	0	0	1	0	0	1	0	0	0
	0 0	1	0	0	0	1	1	0	0	0
	0 0	1	0	1	0	1	1	0	0	0
	0 0	1	1	0	0	1	1	0	0	0
	0 0	1	1	1	0	1	1	0	0	0
EST1	0 1	0	0	0	1	0	0	0	1	1
	0 1	0	0	1	0	0	0	1	0	1
	0 1	0	1	0	1	0	0	0	1	1
	0 1	0	1	1	0	0	0	1	0	1
	0 1	1	0	0	1	0	0	0	1	1
	0 1	1	0	1	0	0	0	1	0	1
	0 1	1	1	0	1	0	0	0	1	1
	0 1	1	1	1	0	0	0	1	0	1
EST2	1 0	0	0	0	0	0	0	1	1	0
	1 0	0	0	1	0	0	0	1	1	0
	1 0	0	1	0	0	0	0	1	1	0
	1 0	0	1	1	0	0	0	1	1	0
	1 0	1	0	0	0	0	0	1	1	0
	1 0	1	0	1	0	0	0	1	1	0
	1 0	1	1	0	0	0	0	1	1	0
	1 0	1	1	1	0	0	0	1	1	0
EST3	1 1	0	0	0	0	0	0	1	0	0
	1 1	0	0	1	0	0	0	1	0	0
	1 1	0	1	0	0	0	0	1	0	0
	1 1	0	1	1	0	0	0	1	0	0
	1 1	1	0	0	0	0	0	1	0	0
	1 1	1	0	1	0	0	0	1	0	0
	1 1	1	1	0	0	0	0	1	0	0
	1 1	1	1	1	0	0	0	1	0	0

# SOLUCIÓN EJEMPLO

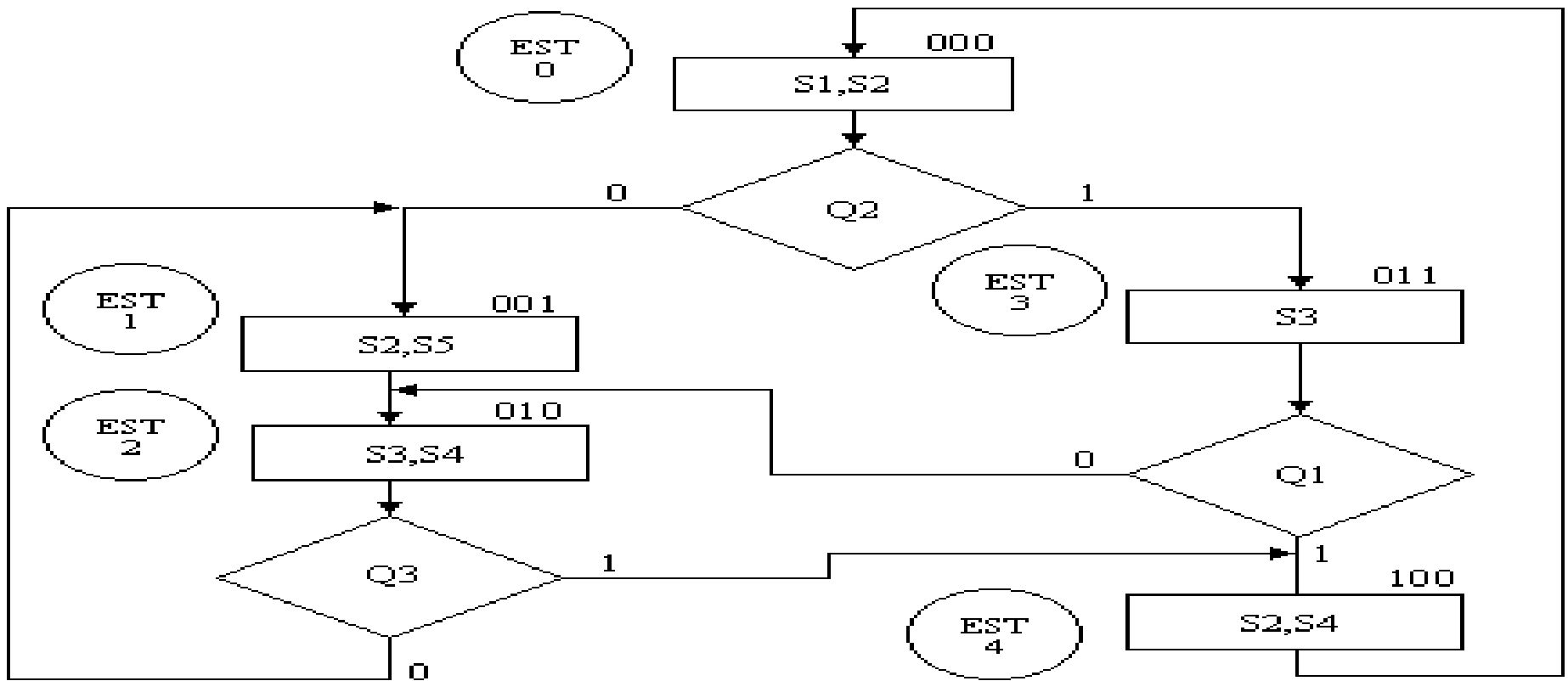


# DIRECCIONAMIENTO DE ENTRADA-ESTADO

Este tipo de direccionamiento se restringe a cartas ASM con una sola entrada por estado. Una nueva porción de la palabra de memoria contiene una representación binaria de la entrada a probar en cada estado, esta parte es llamada "PRUEBA". Con esta representación binaria un selector de entrada elige una de las variables de entrada. La parte de liga tiene dos estados siguientes, escogiéndose uno por el selector de liga, en base a la entrada seleccionada por la parte de prueba. Si el valor de la entrada seleccionada por el selector de entradas es igual a cero, entonces el selector de liga elegirá la liga falsa, en caso contrario se seleccionará la liga verdadera.



# EJEMPLO



Además de asignar una representación binaria a cada estado, también a cada variable de entrada se le asignará una representación binaria.

Para los estados que no tengan variable de entrada, se utiliza una variable auxiliar, la cual tiene un valor preestablecido de cero ó uno.



# SOLUCIÓN EJEMPLO

Para este ejemplo, se eligió la siguiente representación binaria de las entradas:

$Q_x$  = Variable Auxiliar = 00

$Q_1$  = 01

$Q_2$  = 10

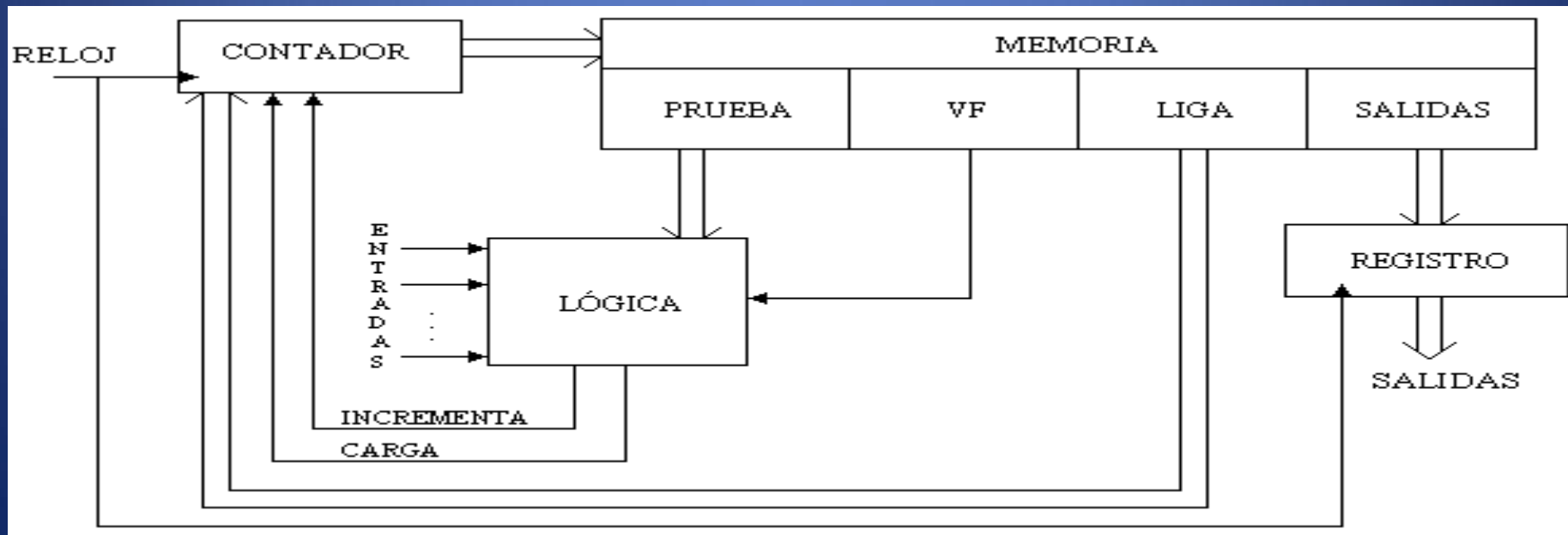
$Q_3$  = 11

En el estado EST0 se selecciona la entrada  $Q_2$ , por lo tanto, se coloca en el campo de prueba de la memoria su representación binaria, es decir, un 10. Si  $Q_2$  es igual a cero, el estado siguiente es EST1 y su representación binaria, 001, es colocada en el campo de la liga falsa. Si  $Q_2$  es igual a uno, el estado siguiente es EST3 y su representación binaria, 011, es colocada en el campo de la liga verdadera. Con los demás estados se procede de la misma forma.

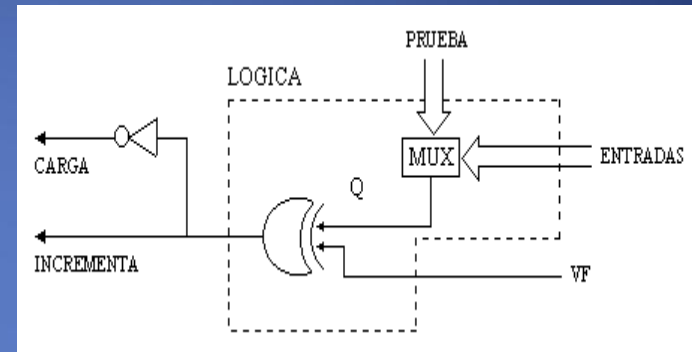
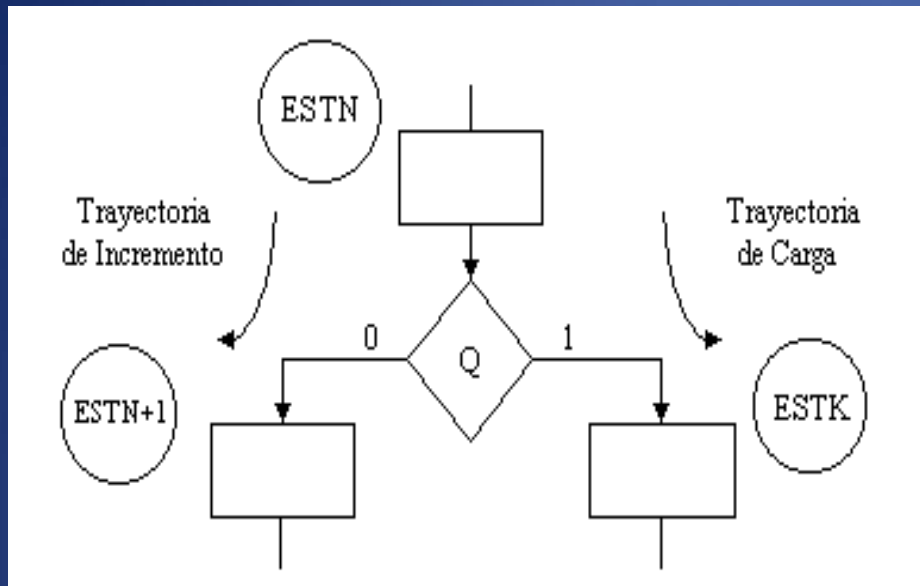
Dirección de Memoria			Contenido de la Memoria												
			Prueba		Liga Falsa			Liga Verdadera			S1	S2	S3	S4	S5
Edo.	Presente														
0	0	0	1	0	0	0	1	0	1	1	1	1	0	0	0
0	0	1	0	0	0	1	0	0	1	0	0	1	0	0	1
0	1	0	1	1	0	0	1	1	0	0	0	0	1	1	0
0	1	1	0	1	0	1	0	1	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

# DIRECCIONAMIENTO IMPLÍCITO

Este tipo de direccionamiento utiliza solamente un campo de liga. Una variable de entrada seleccionada por el campo de prueba, y VF, son las que deciden si se utiliza la dirección de liga (se carga el valor de liga en el contador) o no (se incrementa el contador en una unidad). El campo VF (Verdadero-Falso) sirve para indicarle a la lógica cuánto debe valer la variable de entrada, para así cargar en el contador el valor de la liga y hacer el salto. Además de asignar una representación binaria a cada estado (ABE), también a cada variable de entrada se le asignará una asignación binaria (ABEN).. Se utiliza también una variable auxiliar que sirve para los estados que no tengan variable de entrada, de manera que cuando en un estado no exista variable de entrada se probará la variable auxiliar, la cual tiene un valor preestablecido de cero ó uno.



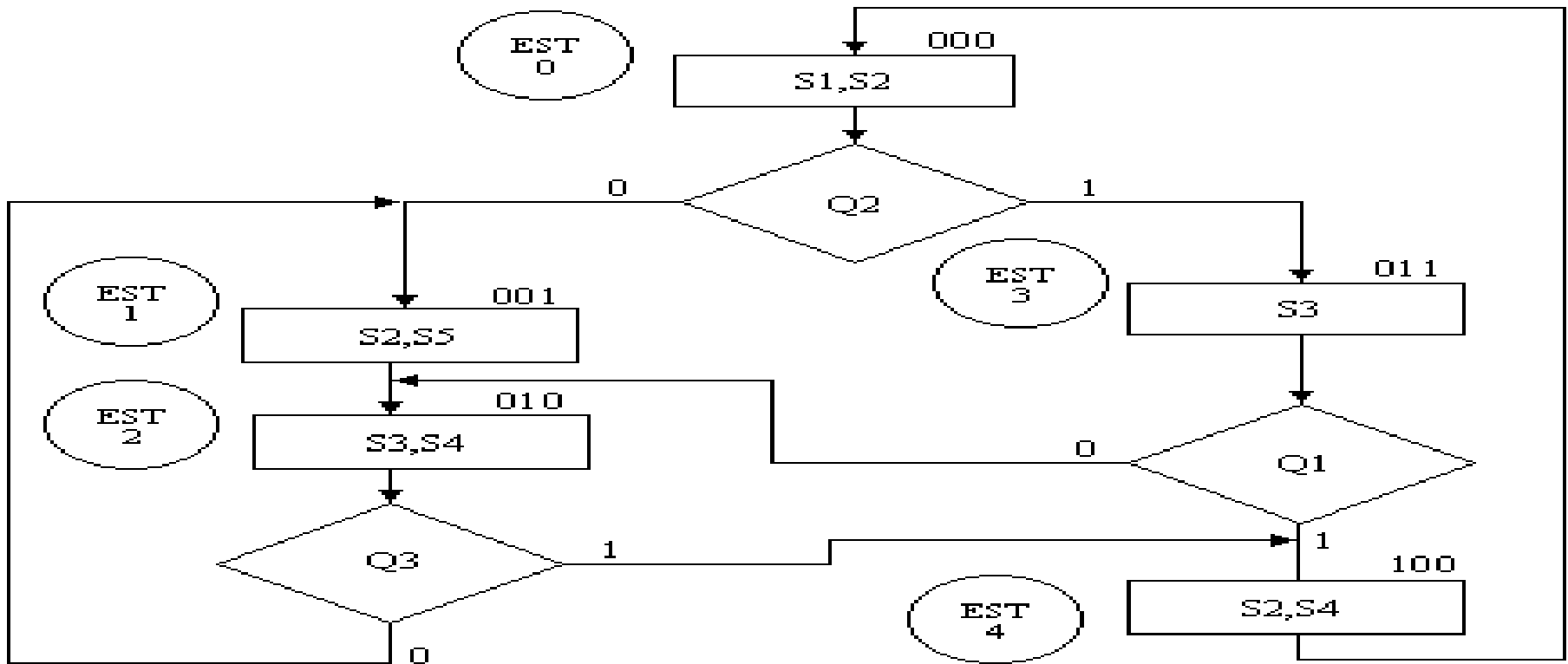
# DIRECCIONAMIENTO IMPLÍCITO



VF	Q	Incrementa	Carga
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

VF y Q se relacionan por medio de una XOR para generar la señal de INCREMENTA y por medio de una XNOR para generar la señal de CARGA.

# EJEMPLO

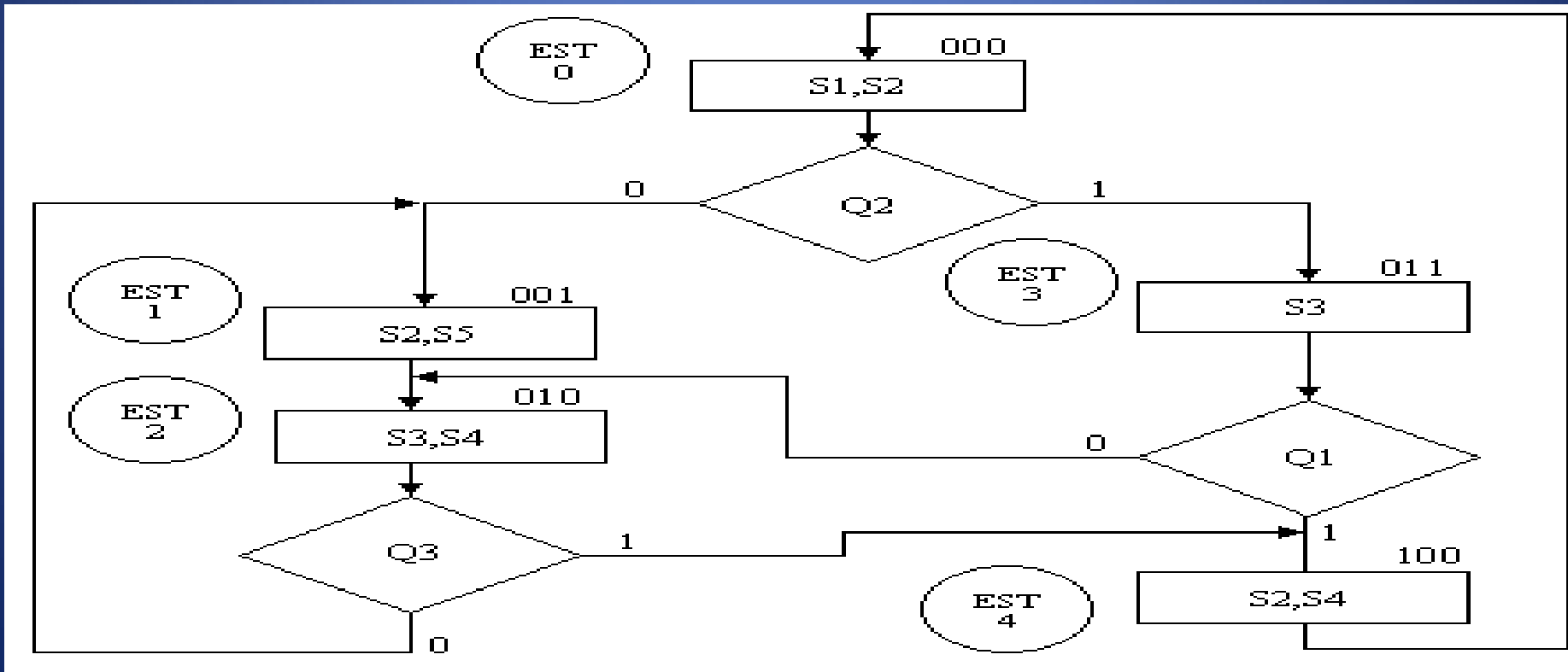


NOTA: Además de realizar una **Asignación Binaria** a cada **Estado (ABE)**, se debe hacer una asignación binaria a las entradas (**ABEN**). Se utiliza también una variable auxiliar que nos sirve para los estados que no tengan variable de entrada, de manera que cuando en un estado no exista variable de entrada se probará la variable auxiliar, la cual tiene un valor preestablecido de cero ó

## SOLUCIÓN EJEMPLO

Es necesario tomar precauciones al hacer la asignación binaria de los estados, porque se debe asegurar que por cada variable de entrada sensada existan dos estados siguientes: uno igual al estado presente más uno y el otro igual a cualquier otro valor. Por ejemplo, para la carta ASM anterior, es necesario cambiar la asignación binaria de los estados, ya que para pasar del estado EST2 al estado EST1 ó del estado EST2 al estado EST4 se necesitan hacer cargas en ambos casos.

La nueva asignación de estados se muestra a continuación..



# SOLUCIÓN EJEMPLO

Para este ejemplo, se eligió la siguiente representación binaria de las entradas:

$Q_x$  = Variable Auxiliar = 00

$Q_1$  = 01

$Q_2$  = 10

$Q_3$  = 11

<b>Dirección de la Memoria</b>	<b>Contenido de la Memoria</b>			
Estado Presente	Prueba	VF	Liga	Salidas $S_1S_2S_3S_4S_5$
000	10	1	010	1 1 0 0 0
001	00	1	011	0 1 0 0 1
010	01	1	100	0 0 1 0 0
011	11	0	001	0 0 1 1 0
100	00	1	000	0 1 0 1 0