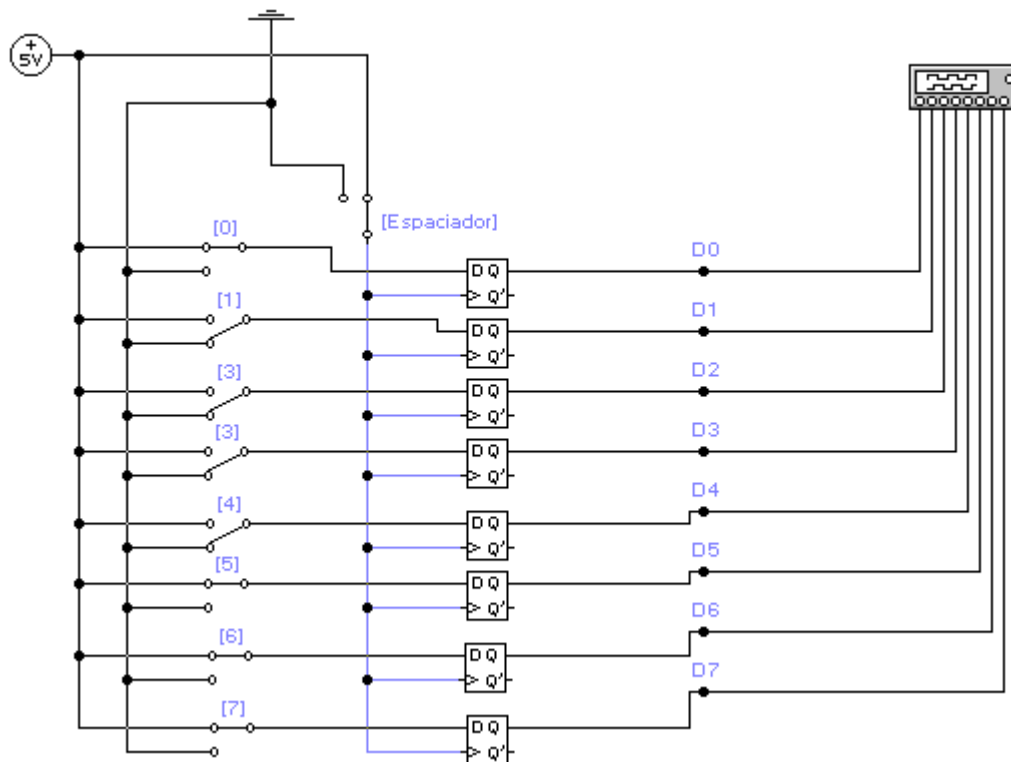


PRÁCTICA 4: CIRCUITOS SECUENCIALES

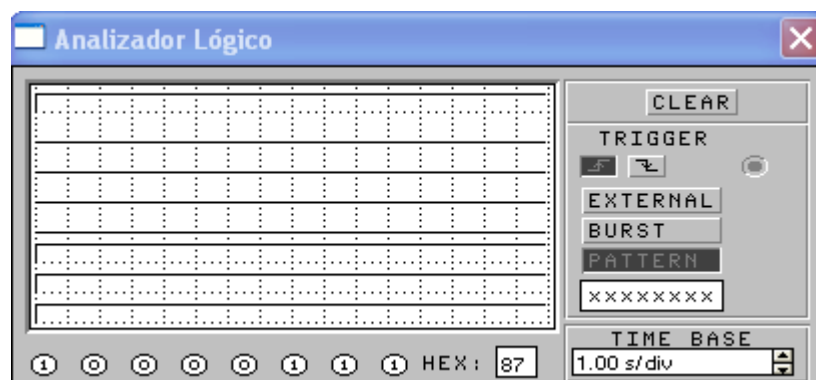
1. Queremos diseñar el circuito que realice la función de “Introducción a la Memoria” (tecla “Min”) y “Borrado de Memoria” (tecla “MC”) de una calculadora. Para implementarlo, seguiremos los siguientes pasos:

a) Utilizando biestables tipo D del simulador, realizaremos un registro de almacenamiento de números de 8 bits. Colocaremos una señal “Min” que cuando se active haga efectiva la carga de los datos en el registro. El dato de 8 bits estará representado por serán señales que tendrán el nombre desde D0 hasta D7. Debemos mostrar el esquema del circuito y probar su correcto funcionamiento en el simulador.

Para activar los biestables y que pueda almacenar el número, necesitamos al ser de tipo D, un pulso por flanco de subida tal y como se ve en la imagen.



Comprobamos su correcto funcionamiento mediante el analizador lógico, consiguiendo ya tener almacenados los números en el registro.



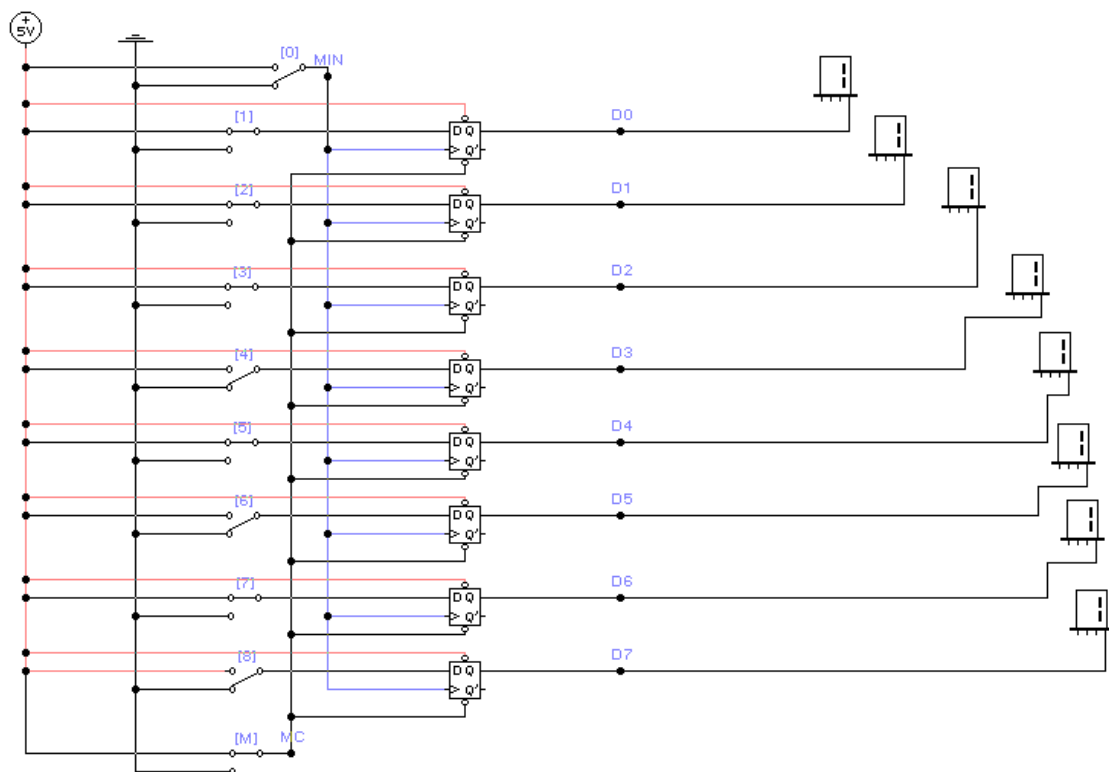
PRÁCTICA 4: CIRCUITOS SECUENCIALES

b) En el circuito anterior sustituiremos los biestables tipo D por biestables D con entradas ASINCRONAS y añadiremos una nueva señal “MC” que, cuando se active, borre el contenido del registro. Debemos mostrar el esquema del circuito y probar su correcto funcionamiento en el simulador.

Para el siguiente esquema, cambiamos los biestables síncronos por otros con entradas asíncronas.

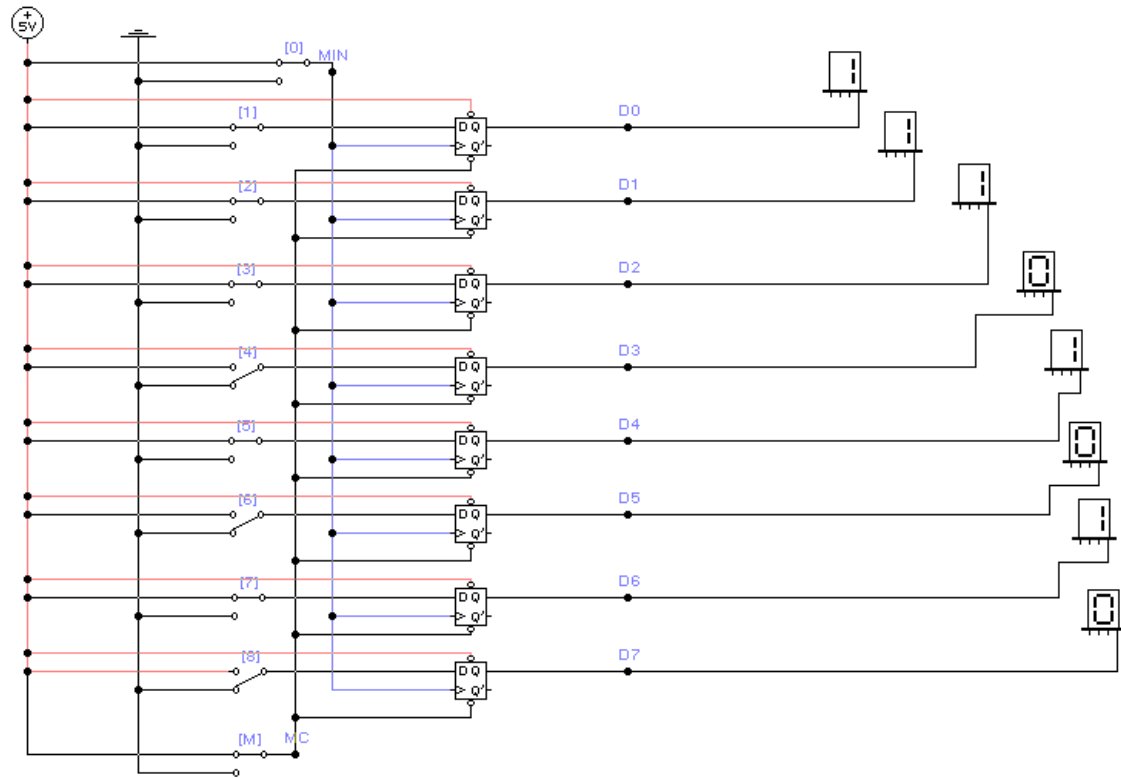
Vemos que las entradas asíncronas están negadas y como no tenemos que utilizar la señal PRESET, la conectamos en nivel ALTO fijo.

Cuando activamos el simulador, observamos que MC esta activado y MIN desactivado el registro aparece a Nivel Alto:



Comprobamos su funcionamiento con un pulso por flanco de subida y ya se observa en la imagen que se nos almacena el registro del numero que introducimos.

PRÁCTICA 4: CIRCUITOS SECUENCIALES



PRÁCTICA 4: CIRCUITOS SECUENCIALES

2.- Se desea implementar, para nuestra calculadora, un circuito secuencial que pueda sumar dos números A y B de cualquier longitud de bits. Para ello, nuestro sistema dispondrá de dos líneas A y B por donde van llegando, de menor a mayor peso, los bits de ambos sumandos a cada pulso de reloj. Además, el sistema dispondrá de una señal adicional “R” que indicará (poniéndose a 1) que comienza una nueva suma y, por tanto, que llegan nuevos sumandos por las líneas serie. El sistema deberá proporcionar, en todo momento, información acerca del resultado de la suma actual (señal “S”) y el acarreo para la etapa siguiente “C”.

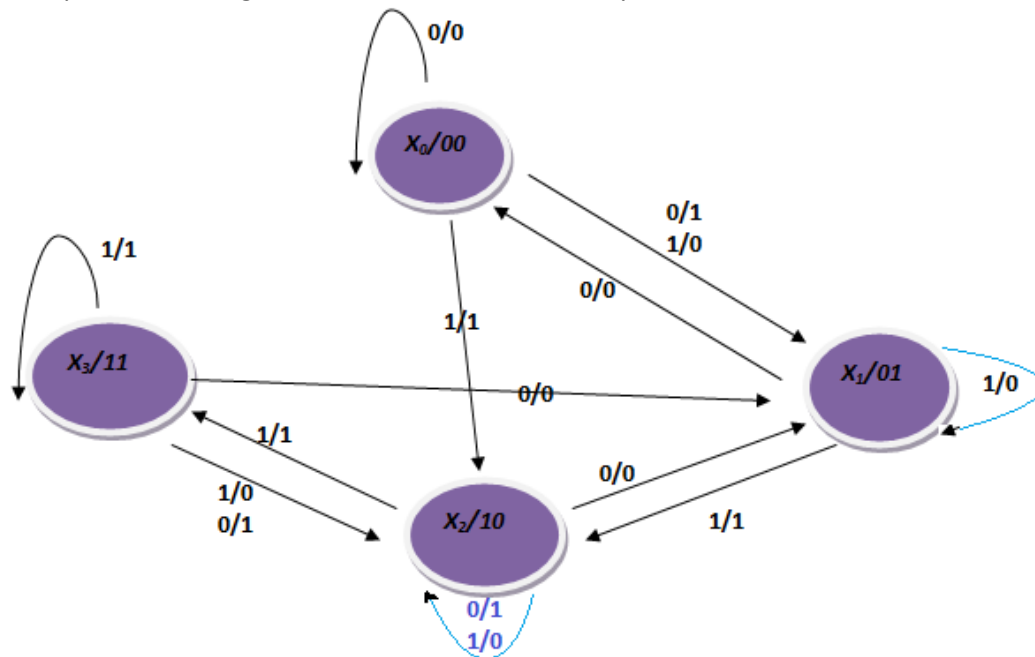
Se pide:

1. Diseñar el sistema secuencial mediante un autómata de Moore, utilizando biestables de tipo JK.
2. Esquema del circuito
3. Implementar el circuito en el simulador y probar su correcto funcionamiento.

El **biestable JK** puede considerarse como el biestable universal. Dispone de tres entradas síncronas J y K, para especificar la operación y CLK, para disparar el biestable. También consta de dos entradas asíncronas PR y CLR, y por supuesto dos salidas complementarias.

Para diseñar el sistema secuencial, vamos a utilizar diagramas que representan el cambio de estado en función del estado previo y de las entradas.

La Figura representa el diagrama de transición de estados y salidas.



Una vez realizado el grafo, realizamos su correspondiente tabla de transacciones simbólicas.

| AB | 00 | 01 | 10 | 11 | Z |
|----------------|----------------|----------------|----------------|----------------|----|
| X ₀ | X ₀ | X ₁ | X ₁ | X ₂ | 00 |
| X ₁ | X ₀ | X ₁ | X ₁ | X ₂ | 01 |
| X ₂ | X ₁ | X ₂ | X ₂ | X ₃ | 10 |
| X ₃ | X ₁ | X ₂ | X ₂ | X ₃ | 11 |

PRÁCTICA 4: CIRCUITOS SECUENCIALES

Para codificar los biestables lo relacionamos de la siguiente manera, donde:

$$X_0 = 00$$

$$X_1 = 01$$

$$X_2 = 10$$

$$X_3 = 11$$

Para poder codificar 2 biestables de 4 estados diferentes necesitamos 2 bits, es decir dos variables, el cual vamos a definir como q_1 y q_0 .

Por tanto el siguiente paso es reemplazar los valores de la tabla transacción a codificación binaria, quedando de la siguiente manera:

| $q_1 q_0 / AB$ | 00 | 01 | 10 | 11 | Z |
|----------------|----|----|----|----|----|
| 00 | 00 | 01 | 01 | 10 | 00 |
| 01 | 00 | 01 | 01 | 10 | 01 |
| 10 | 01 | 10 | 10 | 11 | 10 |
| 11 | 01 | 10 | 10 | 11 | 11 |

Una vez reemplazado los valores realizamos la tabla de la verdad del biestable JK.

| $q(t) \rightarrow Q(t)$ | JK |
|-------------------------|----|
| $0 \rightarrow 0$ | 0X |
| $0 \rightarrow 1$ | 1X |
| $1 \rightarrow 0$ | X1 |
| $1 \rightarrow 1$ | X0 |

Por último realizamos las respectivas tablas de Karnaugh de las dos variable, para posteriormente sacar la función.

Para ello necesitamos la tabla de codificación binaria y la tabla del biestable JK, en el cual se relacionan con la tabla de la verdad del biestable JK dependiendo del valor, esta se pone en los siguiente mapas de karnaugh.

PRÁCTICA 4: CIRCUITOS SECUENCIALES

BIESTABLE q_1

| $q_1 q_0 / AB$ | 00 | 01 | 10 | 11 | Z |
|----------------|----|----|----|----|----|
| 00 | 00 | 01 | 01 | 10 | 00 |
| 01 | 00 | 01 | 01 | 10 | 01 |
| 10 | 01 | 10 | 10 | 11 | 10 |
| 11 | 01 | 10 | 10 | 11 | 11 |

| $q(t) \rightarrow Q(t)$ | JK |
|-------------------------|----|
| $0 \rightarrow 0$ | 0X |
| $0 \rightarrow 1$ | 1X |
| $1 \rightarrow 0$ | X1 |
| $1 \rightarrow 1$ | X0 |

| $q_1 q_0 / AB$ | 00 | 01 | 11 | 10 |
|----------------|----|----|----|----|
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 0 | 1 | 0 |
| 11 | X | X | X | X |
| 10 | X | X | X | X |
| $J_1 = AB$ | | | | |

| $q_1 q_0 / AB$ | 00 | 01 | 11 | 10 |
|----------------|----|----|----|----|
| 00 | X | X | X | X |
| 01 | X | X | X | X |
| 11 | 1 | 0 | 0 | 0 |
| 10 | 1 | 0 | 0 | 0 |
| $K_1 = A'B'$ | | | | |

BIESTABLE q_0

| $q_1 q_0 / AB$ | 00 | 01 | 10 | 11 | Z |
|----------------|----|----|----|----|----|
| 00 | 00 | 01 | 01 | 10 | 00 |
| 01 | 00 | 01 | 01 | 10 | 01 |
| 10 | 01 | 10 | 10 | 11 | 10 |
| 11 | 01 | 10 | 10 | 11 | 11 |

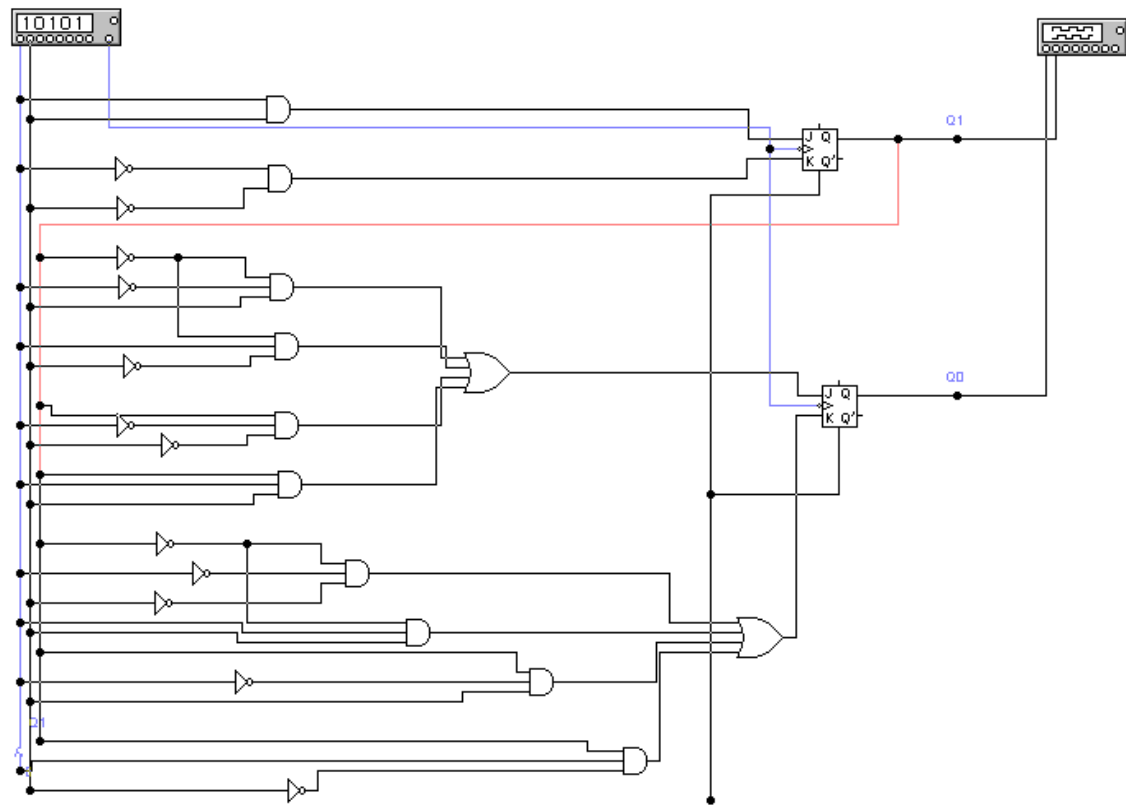
| $q(t) \rightarrow Q(t)$ | JK |
|-------------------------|----|
| $0 \rightarrow 0$ | 0X |
| $0 \rightarrow 1$ | 1X |
| $1 \rightarrow 0$ | X1 |
| $1 \rightarrow 1$ | X0 |

| $q_1 q_0 / AB$ | 00 | 01 | 11 | 10 |
|---|----|----|----|----|
| 00 | 0 | 1 | 0 | 1 |
| 01 | X | X | X | X |
| 11 | X | X | X | X |
| 10 | 1 | 0 | 1 | 0 |
| $J_1 = q_1 A'B + q_1 AB' + q_1 A'B' + q_1 AB$ | | | | |

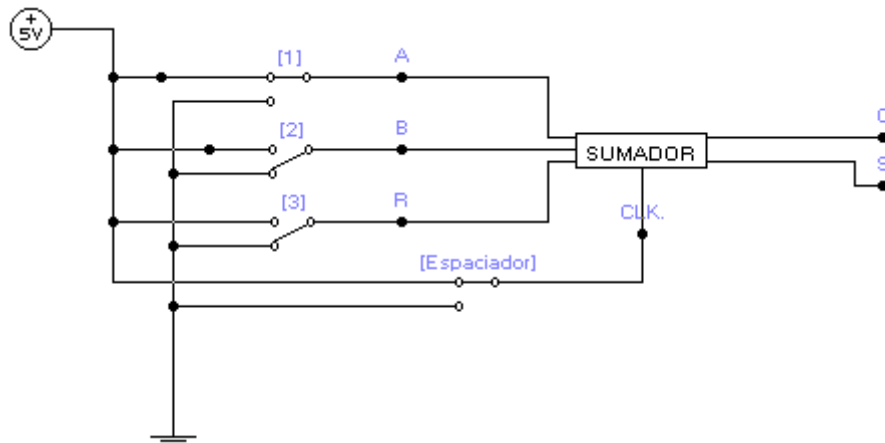
| $q_1 q_0 / AB$ | 00 | 01 | 11 | 10 |
|---|----|----|----|----|
| 00 | X | X | X | X |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 0 | 1 | 0 | 1 |
| 10 | X | X | X | X |
| $K_1 = q_1 A'B' + q_1 AB + q_1 A'B + q_1 AB'$ | | | | |

PRÁCTICA 4: CIRCUITOS SECUENCIALES

Una vez obtenidas las funciones mediante la simplificación de Karnaugh ya podemos implementar el circuito.



Esquema del circuito.



PRÁCTICA 4: CIRCUITOS SECUENCIALES

3.- Utilizando como base el sumador serie diseñado en el apartado anterior (que previamente habremos convertido en subcircuito) y registros de desplazamiento de 4 bits, se pide diseñar un circuito que sume dos números de 4 bits.

Para ello el sistema contará con dos registros de desplazamiento de 4 bits donde estén almacenados los sumandos (A y B) y un tercero para almacenar el resultado (S). La cuenta de los bits que van llegando al sumador serie será controlada por un contador de 4 bits que se usará para detectar que ya ha terminado la llegada de los operandos, inhabilitando la señal de reloj.

El circuito 74194 es un registro de desplazamiento que puede realizar la carga en paralelo de los datos y luego desplazarlos a izquierda o derecha, su funcionamiento es la siguiente:

74194 (Registro de Desplazamiento Bidireccional de 4-bits)

Tabla de verdad del Registro de Desplazamiento:

| CLEAR | MODO | | CLK | SERIE | | PARALELO | | | | SALIDAS | | | |
|-------|------|----|-----|-------|------|----------|---|---|---|---------|-----|-----|-----|
| | S1 | S0 | | IZQ. | DER. | A | B | C | D | QA | QB | QC | QD |
| 0 | X | X | X | X | X | X | X | X | X | 0 | 0 | 0 | 0 |
| 1 | X | X | 0 | X | X | X | X | X | X | QA0 | QB0 | QC0 | QD0 |
| 1 | 1 | 1 | POS | X | X | a | b | c | d | a | b | c | d |
| 1 | 0 | 1 | POS | X | 1 | X | X | X | X | 1 | QAn | QBn | QCn |
| 1 | 0 | 1 | POS | X | 0 | X | X | X | X | 0 | QAn | QBn | QCn |
| 1 | 1 | 0 | POS | 1 | X | X | X | X | X | QBn | QCn | QDn | 1 |
| 1 | 1 | 0 | POS | 0 | X | X | X | X | X | QBn | QCn | QDn | 0 |
| 1 | 0 | 0 | X | X | X | X | X | X | X | QA0 | QB0 | QC0 | QD0 |

- POS = transición de bajo a alto

- a, b, c, d = el nivel de estado estable para las entradas A, B, C, o D respectivamente

- QA0, QB0, QC0, QD0 = el nivel de QA, QB, QC, o QD, respectivamente, anterior al cual se habían sido establecidas las condiciones indicadas de régimen estacionario

- QAn, QBn, QCn, QDn = el nivel de QA, QB, QC, o QD anterior a la más reciente transición negativa del reloj.

Se pide:

a) esquema del circuito usando el sumador serie del apartado 2, registros de desplazamiento de 4 bits 74194.

b) Montaje y prueba del funcionamiento en el simulador.

