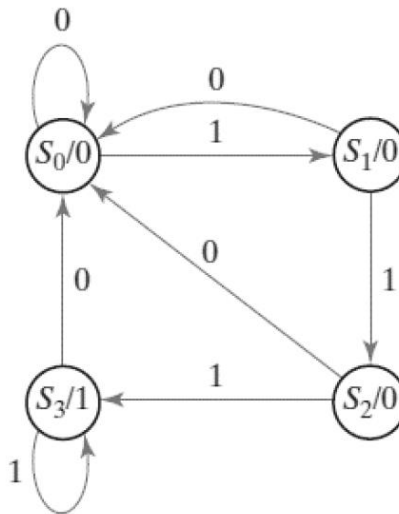


## Práctica 5. Máquina de estados.

### 5.1. Introducción.

Un 'estado' es la condición de una cosa en un tiempo determinado. Algunos que pueden realizar tareas y que utilizan estados como su núcleo son máquinas de estado. También son conocidas como máquinas de estado finitas, lo que significa que sabemos todos los posibles estados de ella. La clave para la máquina de estado es el concepto del tiempo y la historia. El estado de la máquina es evaluada periódicamente. Cada vez que es evaluada, un nuevo estado es elegido (el que podría ser el mismo estado nuevamente) y el resultado es presentado.

El modelo más general de un circuito secuencial tiene entradas, salidas y estados internos. Se acostumbra distinguir entre dos modelos de circuitos secuenciales: el modelo Mealy y el modelo Moore. Difieren en la forma en que se genera la salida. En el modelo Moore, la salida sólo es función del estado actual. En un modelo Moore, las salidas del circuito secuencial se sincronizan con el reloj porque sólo dependen de salidas de flip-flop que están sincronizadas con el reloj.



El diagrama de estados del circuito se presenta en la figura anterior. Se obtiene partiendo del estado  $S_0$ . Si la entrada es 0, el circuito permanece en el mismo estado, pero si es 1, pasa al estado  $S_1$  para indicar que se detectó un 1. Si la siguiente entrada es 1, el cambio es al estado  $S_2$ , para indicar que han llegado dos unos consecutivos, pero si la entrada es 0 volvemos al estado  $S_0$ . El tercer uno consecutivo envía al circuito al estado  $S_3$ . Si se detectan más unos, el circuito permanecerá en  $S_3$ . Cualquier entrada 0 devolverá el circuito a  $S_0$ . Así, el circuito permanecerá en  $S_3$  en tanto se hayan recibido tres o más unos consecutivos.

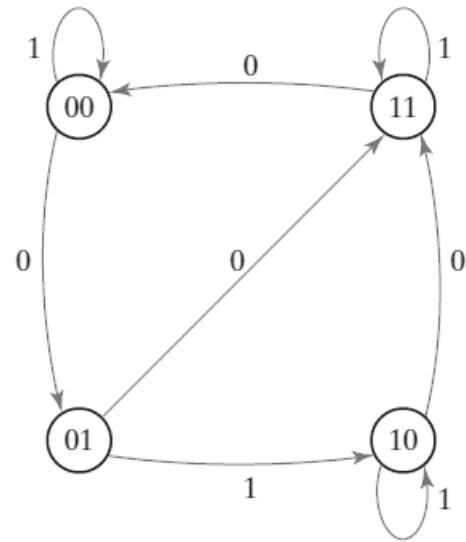


## 5.2. Previo Práctica 1.

- 1) ¿Qué es una máquina de estado finitos?
- 2) ¿Qué importancia tiene las máquinas de estados?
- 3) ¿Qué función cumple la instrucción **parameter** y cual es la sintaxis de declaración?

## 5.3. Ejemplo del Diagrama de estados de Moore

Estado Actual		Entrada	Siguiete estado	
A	B		A	B
0	0	0	0	1
0	0	1	0	0
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	0
1	1	1	1	1



Solución en Verilog:

```
module moore (x,AB,CLK,RST);
    input x,CLK,RST;
    output [1:0]AB;
    reg [1:0] state;
    parameter S0 = 2'b00, S1 = 2'b01, S2 = 2'b10, S3 = 2'b11;

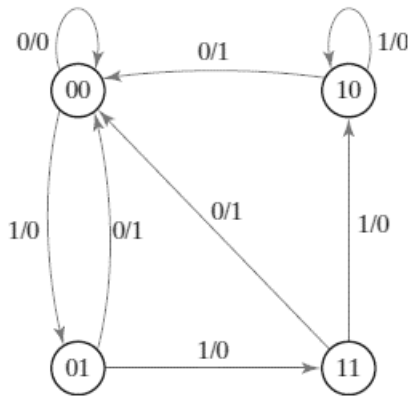
    always @ (posedge CLK )
        if (~RST) state = S0;
        else
            case (state)
                S0: if (~x) state = S1; else state = S0;
                S1: if (x) state = S2; else state = S3;
                S2: if (~x) state = S3; else state = S2;
                S3: if (~x) state = S0; else state = S3;
            endcase

        assign AB = state;
endmodule
```



## 5.4. Trabajo de laboratorio

- 1) Del ejercicio anterior realizar su implementación en la tarjeta de desarrollo y que funciones de forma **SECUENCIAL**.
- 2) De la siguiente Máquina de Estados obtener su tabla de verdad y su simulación funcional en verilog, así como su implementación en la tarjeta de desarrollo y que funciones de forma **SECUENCIAL**.



## 5.5. Conclusiones