

UNIVERSIDAD INDUSTRIAL DE SANTANDER
Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones

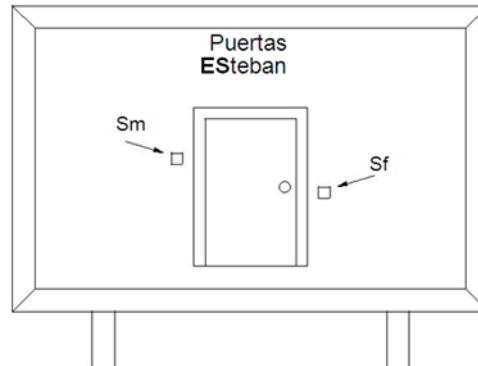


Sistemas Digitales I
Taller No 3
Máquinas de Estados Finitos (FSM)

Profesor:
Carlos A. Fajardo

Bucaramanga, Colombia
(Actualizado marzo de 2019)

1. El profesor Bacterio le ha pedido ayuda para que le diseñe el circuito de control de la nueva puerta secreta que se va a instalar en el cuartel general de la T.I.A. La puerta secreta ha sido convenientemente disimulada en un cartel publicitario de un conocido fabricante de puertas, tal como se muestra en la figura siguiente:



La puerta secreta está diseñada para ser usada única y exclusivamente por la pareja de súper agentes más famosa de la agencia: Mortadelo y Filemón. Para ello se han colocado dos sensores a ambos lados de la puerta justo a la altura de la cabeza de cada uno de los dos agentes, representados en la figura mediante Sm y Sf.

Funcionamiento de la puerta:

- Para activar el mecanismo Mortadelo y Filemón han de colocarse a ambos lados de la puerta, justo delante de su sensor correspondiente. Cuando el sistema detecte esta situación se encenderá una luz que se ha disimulado justo detrás de la letra “E” del cartel.
- A continuación Filemón tendrá que colocarse delante de la puerta para que el sensor Sf deje de detectarlo. Para confirmar este paso el circuito apagará la letra “E”.
- Cuando Mortadelo vea que se ha apagado la luz se pondrá también delante de la puerta para que el sensor Sm deje de detectarlo.
- El circuito entonces activará una señal para abrir la puerta, esta señal sólo deben durar un ciclo de reloj
- Una vez iniciada la secuencia, si en algún paso se realiza una acción equivocada, como por ejemplo que sea Mortadelo en lugar de Filemón el primero en ponerse delante de la puerta, se activará una alarma para alertar a todos los agentes de la T.I.A. de un posible intento de asalto a su sede. Dicha alarma seguirá activa hasta que se vuelva a inicializar el circuito con la señal de reset.

Para tener en cuenta:

- ✓ Los sensores Sm y Sf dan un 1 cuando detectan una persona enfrente de ellos y un 0 en caso contrario.
- ✓ El sistema cuenta con un reloj de **50MHz**.

2. **Sketch a state diagram** for a control of an analog-to-digital converter, as shown in the block diagram of Figure P10:

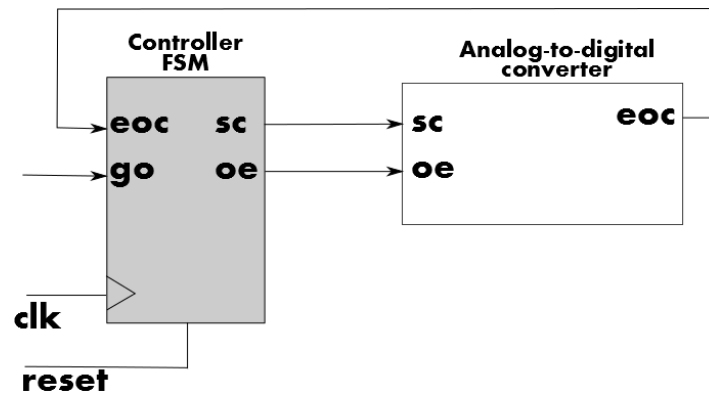


Figura P11

The controller has four states as follows: idle, start, waiting, and read. There are two outputs: sc (Start Conversion; active-HIGH) and oe (Output Enable; active LOW). There are four inputs: clock, go (active-LOW) eoc (End of Conversion), and asynchronous reset (active LOW). The machine operates as follows:

In the idle state, the outputs are: sc = 0, oe = 1. The machine defaults to the idle state when the machine is reset. Upon detecting a 0 at the go input, the machine makes a transition to the start state. In this transition, sc=1, oe = 1. The machine makes an unconditional transition to the waiting state; sc = 0, oe= 1. It remains in this state, with no output change, until input eoc=1. When eoc =1, the machine goes to the read state; sc=0, oe=0. The machine makes an unconditional transition to the idle state; sc=0, oe=1. (Tomado de Duek, 2000)

3. La empresa Roca le ha encargado el diseño del circuito de control de una cisterna automática. El circuito detectará cuando alguien se acerque al WC a hacer sus necesidades y descargará la cisterna cuando éste haya terminado y se vaya del servicio. Para ello el WC contará con un Sistema Infrarrojo, tal como se muestra en la siguiente Figura P3.

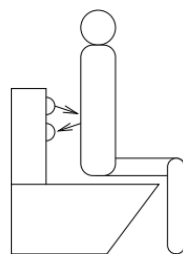


Figura P3

El Sistema Infrarrojo tendrá una entrada para activar el emisor de infrarrojos (A_E) para activarlo y una salida (R) que se pondrá a 1 lógico cuando reciba la luz infrarroja reflejada en el usuario a se pondrá en 0 cuando no reciba nada. La cisterna dispondrá de una electroválvula que se controlará con la señal Cisterna (C), de forma que cuando dicha señal esté a 1 se abrirá la electroválvula para descargar la cisterna.

El funcionamiento del circuito será el siguiente: en el estado inicial, para evitar consumir demasiada energía, el emisor de infrarrojos estará desactivado. De este estado se irá a otro en el que se activará el emisor. Si no se detecta a nadie se volverá al estado inicial para volver a repetir el ciclo. Si se detecta a alguien, se esperará a que éste se marche para descargar la cisterna y volver al estado inicial. Tenga en cuenta que el reloj del sistema es de 50MHz.

El diseño debe contener:

- ✓ Diagrama de bloques.
 - ✓ Si el diseño incluye una *Máquina de Estados*, se debe incluir el diagrama de estados, donde indique claramente: estados, entradas, salidas y condiciones para la transición entre estados.
4. Para unas olimpiadas le encargan el diseño de un circuito que controle un sintetizador para automatizar las salidas de las carreras de velocidad. El sistema completo consta del control y de un sintetizador de voz para generar las frases “a sus puestos”, “listos” y “ya” (Ver figura P10).

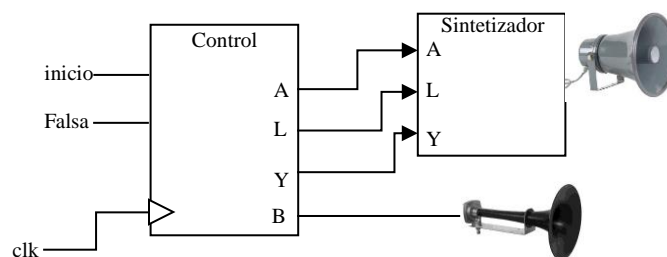


Figura P4.

El sintetizador dispone de tres entradas (A, L, Y), cada una de las cuales hace que se genere una de las frases. El sistema también consta de una bocina (B) para avisar de salidas falsas. Para detectar estas salidas falsas, se cuenta con la señal (Falsa), la cual estará en 1 cuando todos los corredores estén pisando los tacos y un 0 cuando no lo estén. El circuito dispondrá de un pulsador de *inicio* (sin rebotes) que mientras esté a cero mantendrá al circuito en un estado de reposo. Cuando el administrador de un pulso en la señal de inicio, el sistema dará el aviso de “a sus puestos”. Cuando todos los corredores hayan pasado a la pista y estén pisando sus tacos se dará el aviso de “listos”. Un segundo después, si no se ha detectado ninguna salida falsa, se dará la salida con el aviso de “ya”. Si por el contrario se ha producido alguna salida falsa, el circuito activará la bocina (B) durante un

segundo. En cualquiera de los casos, el circuito pasará al estado inicial de reposo.

Para tener en cuenta:

- Si una de las entradas al sintetizador está activada durante más de un segundo, el mensaje se repetirá varias veces. Por tanto, si no quiere que la organización quede mal delante de todo el mundo tendrá que evitar esta situación.
- Cuenta con un reloj de 1Hz.

El diseño debe contener:

- ✓ Diagrama de bloques.
- ✓ Si el diseño incluye una *Máquina de Estados*, se debe incluir el diagrama de estados, donde indique claramente: estados, entradas, salidas y condiciones para la transición entre estados

5. Una lámpara dispone de dos bombillas, una de 40 W y otra de 60 W. La finalidad de poner dos bombillas es la de poder regular el nivel de luminosidad, encendiendo o bien una de ellas o bien las dos a la vez, para conseguir una potencia total de 100 W. Para controlar la lámpara se dispone de 2 pulsadores (sin rebotes), denominados *s* (*subir*) y *b* (*bajar*). Cuando se conecte la alimentación la lámpara permanecerá apagada. Cada pulsación del botón *s* irá aumentando la luminosidad de la lámpara y cada pulsación del botón *b* la decrementará. La secuencia de encendido/apagado de las lámparas será 40 W, 60 W y 100W. Así, la primera vez que se pulse el botón *s* se encenderá sólo la bombilla de 40 W, la segunda vez que se pulse *up* se encenderá sólo la bombilla de 60 W y la siguiente pulsación del botón *up* encenderá las dos bombillas. Si se sigue pulsando el botón *up* no ocurriría nada. De la misma forma las pulsaciones del botón *down* irán bajando la luminosidad y si una vez apagadas las dos bombillas se sigue pulsando el botón *down* no pasará nada. Si se pulsan ambos botones a la vez la luminosidad no cambiará.

Notas:

Dispone de un reloj de 50MHz.

El diseño debe contener:

- ✓ Diagrama de bloques.
- ✓ Si el diseño incluye una *Máquina de Estados*, se debe incluir el diagrama de estados⁷, donde indique claramente: estados, entradas, salidas y condiciones para la transición entre estados.

6. Diseñe el control circuito para controlar un relé térmico de un motor. El circuito tendrá un interruptor **start** que se pondrá a 1 para arrancar el motor y a cero para pararlo. Además el circuito dispone también de una entrada T que se activa a nivel alto cuando el motor sufra un sobrecalentamiento. El circuito dispondrá de dos salidas: una para arrancar el motor (M) y otra de alarma (A) para indicar que ha ocurrido un sobrecalentamiento; ambas activas a nivel alto. El circuito a diseñar ha de arrancar el motor cuando se

active la entrada **start = 1** y mantenerlo arrancado hasta que se desactive dicha entrada **start = 0** o hasta que se produzca un sobrecalentamiento ($T=1$). En este caso (sobrecalentamiento) el motor se para y se activara la salida de alarma. Aunque el motor se enfríe ($T=0$) no se volverá a arrancar hasta que el usuario no ponga a 0 y luego a 1 la entrada **start**. La señal de alarma se mantendrá hasta que la entrada de **start** se ponga a 0.

El diseño debe contener:

- Diagrama de bloques.
- Si el diseño incluye una *Máquina de Estados*, se debe incluir el diagrama de estados, donde indique claramente: estados, entradas, salidas y condiciones para la transición entre estados.
- Asuma que el interruptor NO tiene rebotes.

7. Modifique el ejercicio anterior suponiendo que la señal **start** es un pulsador (sin rebotes), es decir, que para arrancar el motor se debe pulsar (llevarlo a 1 y él vuelve a ser cero automáticamente) start y para detenerlo se debe pulsar nuevamente start. Suponga que usted dispone de un reloj de 50kHz.

El diseño debe contener:

- Diagrama de bloques.
- Si el diseño incluye una *Máquina de Estados*, se debe incluir el diagrama de estados , donde indique claramente: estados, entradas, salidas y condiciones para la transición entre estados.

8. Una importante constructora contrata a un estudiante de la prestigiosa Universidad Industrial de Santander, para que le diseñe un sistema de iluminación para los pasillos de una construcción de la ciudad. El diseño debe cumplir con las siguientes especificaciones:

- El pasillo dispone de dos **pulsadores (sin rebotes)**, uno al lado de cada puerta, de manera que se pueda encender y apagar la luz desde cada extremo. Cada pulsador produce un '1' lógico mientras está pulsado y un '0' lógico cuando no lo está.
- Se desea que cada vez que se pulse cualquier pulsador, la luz cambie de estado: si está apagada se debe encender, y viceversa.
- Se debe tener en cuenta el caso en el que mientras se pulsa un pulsador, se pulse el otro. Por ejemplo, si estando apagada la luz, alguien pulsa P1 se enciende la luz. Pero si mientras está pulsado P1 alguien pulsa P2, entonces se apagará nuevamente la luz. Sin embargo, se puede considerar que la frecuencia del reloj es lo suficientemente alta como para que sea imposible un cambio simultáneo de los dos pulsadores (en el mismo ciclo de reloj).

Se espera que su diseño sea funcional, es decir, que cumpla con el funcionamiento esperado.

El diseño debe contener:

- Diagrama de bloques.
- Si el diseño incluye una *Máquina de Estados*, se debe incluir el diagrama de estados, donde indique claramente: estados, entradas, salidas y condiciones para la transición entre estados.

9. Asuma que usted es un sintetizador de Verilog. Dibuje el diagrama de estados de la siguiente descripción.

```
`timescale 1ns / 1ps

module Mi_FSM(
    input clk,
    input reset,
    input A,
    input B,
    output reg [2:0] salidas
);

localparam [2:0]
    S0 = 3'b000,
    S1 = 3'b001,
    S2 = 3'b010,
    S3 = 3'b011,
    S4 = 3'b100;

reg [2:0] Q;
reg [2:0] D;

/* Registro de Estado */
always @(posedge clk)
    if (reset) begin
        Q <= S0;
    end else begin
        Q <= D;
    end
end

/*Logica del Estado siguiente*/
always @(Q,A,B)
begin
    case (Q)
        S0:
            if (A & B)
                D = S1;
            else
                D = S0;

        S1:
            if (A & B)
                D = S1;
            else if (A & !B)
                D = S2;
            else
                D = S3;

        S2 :
            if (A & !B)
                D = S2;
            else if (!A & !B)
                D = S4;
```

```

        else
            D = S3;

S3    :
        D = S3;

S4    :
        D = S0;

default:
        D = S0;

    endcase
end

/*Logica de salida*/
always @(Q)
begin
    case (Q)
        S0: salidas = 3'b000;
        S1: salidas = 3'b100;
        S2: salidas = 3'b000;
        S3: salidas = 3'b001;
        S4: salidas = 3'b010;
        default: salidas = 3'b000;
    endcase
end

endmodule

```