

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE0523 – Circuitos Digitales II

I ciclo 2023

Tarea 03:

Diseño de un Controlador para Cajero Automático.

Andrés Chaves Vargas - B92198

Grupo 01

Profesor: Enrique Coen

Mayo 2023

Índice

1. Descripción Arquitectónica.	1
1.1. Diagrama de Bloques	1
1.2. Diagrama ASM	2
2. Plan de Pruebas.	3
3. Instrucciones de Utilización de la Simulación.	6
4. Resultados.	6
4.1. Resultados Prueba 1.	6
4.2. Resultados Prueba 2.	7
4.3. Resultados Prueba 3.	7
4.4. Resultados Prueba 4.	8
4.5. Problemas con la Simulación.	9
5. Conclusiones.	9
6. Recomendaciones.	9

Resumen

En esta tarea se obtuvo las simulaciones realizadas a una máquina de estados de un controlador de cajero automático. Dichas simulaciones fueron analizadas utilizando la herramienta GTK-wave. Para comprobar el funcionamiento correcto de la máquina de estados se realizó un plan de cuatro pruebas, de las cuales se pasaron tres completamente y una de ellas de forma parcial. A partir de dichas simulaciones se generaron varias conclusiones, entre ellas que los retardos incluidos en el tester.v pueden generar problemas en las lecturas de los dígitos ingresados.

1. Descripción Arquitectónica.

1.1. Diagrama de Bloques

A continuación se muestra el diagrama de Bloques del cajero automático:

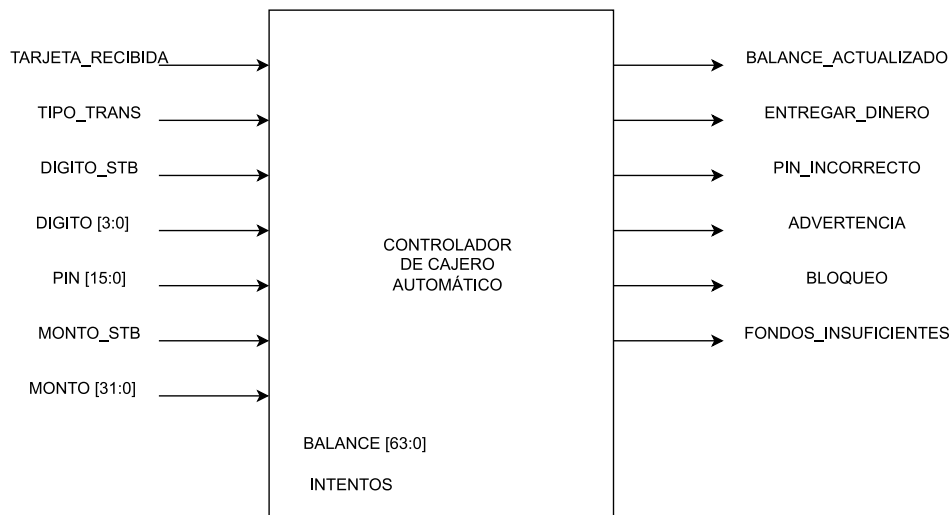


Figura 1: Diagrama de Bloques Cajero Automático.

Los estados del controlador son los siguientes:

- Estado 1: Esperando Tarjeta.
- Estado 2: Esperando Primer dígito.
- Estado 3: Esperando Segundo dígito.
- Estado 4: Esperando Tercer dígito.
- Estado 5: Esperando Cuarto dígito.
- Estado 6: Verificando PIN correcto.
- Estado 7: Bloqueo. Esperando RESET.
- Estado 8: Depósito.
- Estado 9: Retiro.

1.2. Diagrama ASM

A continuación se muestran el diagrama ASM del cajero automático:

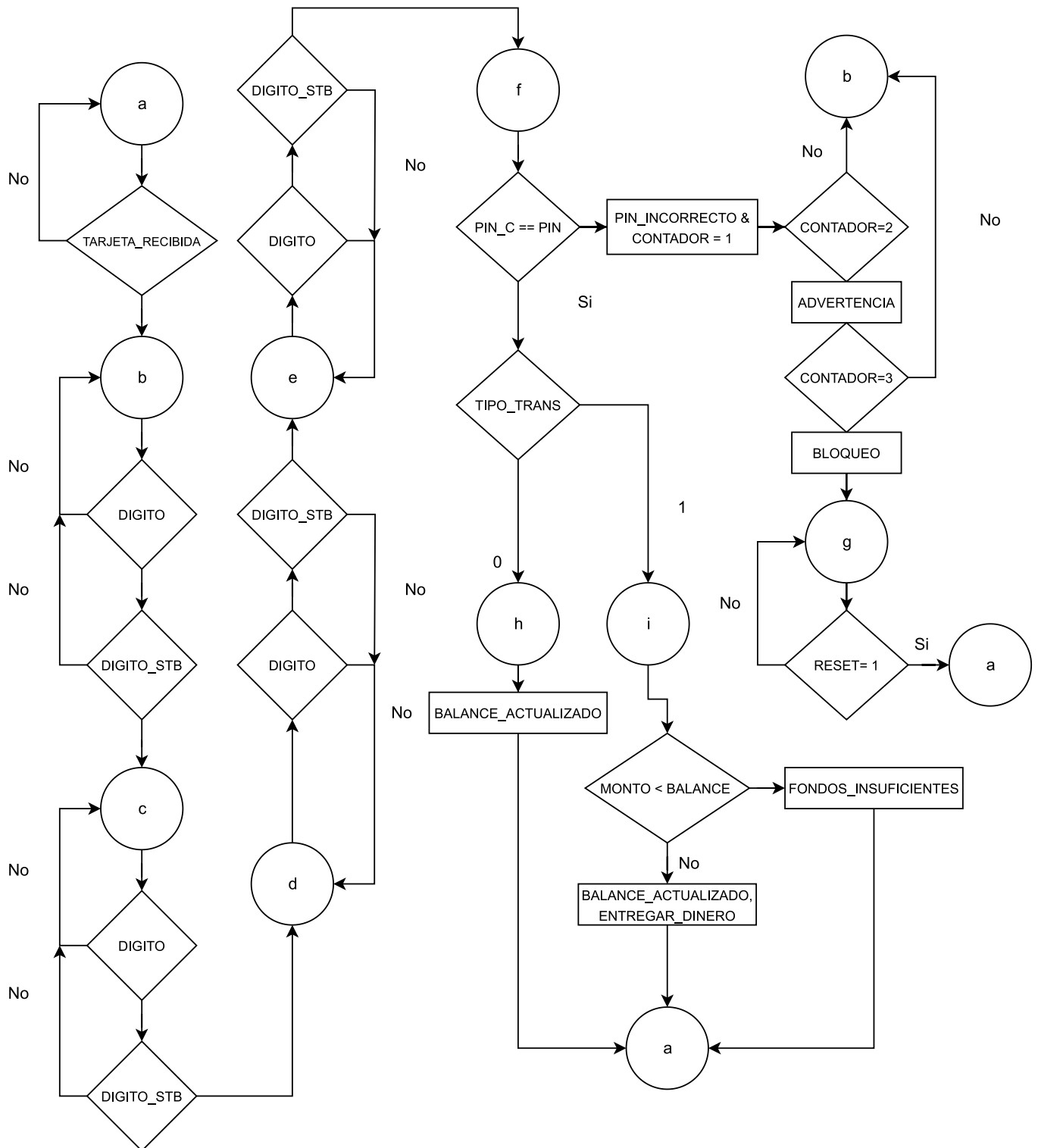


Figura 2: Diagrama ASM Cajero Automático.

2. Plan de Pruebas.

El plan de pruebas implementado es bastante simple, consiste en cuatro pruebas las cuales permiten ingresar a todos los estados posibles dentro de la máquina de estados para el cajero automático. A continuación se muestra el plan de pruebas a implementar:

- **Prueba 1: Proceso Normal de Funcionamiento. Acción: Depósito.**

Esta prueba consiste en ingresar los dígitos correctos y el valor TIPO_TRANS con el fin de llegar al estado de Depósito. La idea es que el valor asignado a MONTO se sume al valor asignado a BALANCE y se ponga en alto la señal BALANCE_ACTUALIZADO. A continuación se muestran los valores ingresados:

Primer dígito:

DIGITO = 2

DIGITO_STD = 1

DIGITO_STD = 0

Segundo dígito:

DIGITO = 1

DIGITO_STD = 1

DIGITO_STD = 0

Tercer dígito:

DIGITO = 9

DIGITO_STD = 1

DIGITO_STD = 0

Cuarto dígito:

DIGITO = 8

DIGITO_STD = 1

DIGITO_STD = 0

BALANCE = 64'h00000000

MONTO = 32'h5000000

MONTO_STB = 1

TIPO_TRANS = 0

- **Prueba 2: Proceso Normal de Funcionamiento. Acción: Retiro (Fondos Suficientes).**

Al igual que la prueba anterior prueba consiste en ingresar los dígitos correctos y el valor TIPO_TRANS con el fin de llegar al estado de Retiro. La idea es que el valor asignado

a MONTO sea menor que el valor asignado a BALANCE y se ponga en alto la señal BALANCE_ACTUALIZADO y ENTREGAR_DINERO. A continuación se muestran los valores ingresados:

Primer dígito:

DIGITO = 2

DIGITO_STD = 1

DIGITO_STD = 0

Segundo dígito:

DIGITO = 1

DIGITO_STD = 1

DIGITO_STD = 0

Tercer dígito:

DIGITO = 9

DIGITO_STD = 1

DIGITO_STD = 0

Cuarto dígito:

DIGITO = 8

DIGITO_STD = 1

DIGITO_STD = 0

BALANCE = 64'h1000000

MONTO = 32'h500000

MONTO_STB = 1

TIPO_TRANS = 1

■ **Prueba 3: Proceso Normal de Funcionamiento. Acción: Retiro (Fondos Insuficientes).**

Esta prueba es casi idéntica a las anteriores, con la diferencia de que se busca que el valor asignado a MONTO sea mayor que el valor asignado a BALANCE y se ponga en alto la señal FONDOS_INSUFICIENTES. A continuación se muestran los valores ingresados:

Primer dígito:

DIGITO = 2

DIGITO_STD = 1

DIGITO_STD = 0

Segundo dígito:

DIGITO = 1

DIGITO.STD = 1

DIGITO.STD = 0

Tercer dígito:

DIGITO = 9

DIGITO.STD = 1

DIGITO.STD = 0

Cuarto dígito:

DIGITO = 8

DIGITO.STD = 1

DIGITO.STD = 0

BALANCE = 64'h00000000

MONTO = 32'h5000000

MONTO.STB = 1

TIPO.TRANS = 1

■ Prueba 4: Proceso Pin Incorrecto: Bloqueo.

Finalmente, en esta prueba se ingresan los dígitos incorrectos a propósito. Con el fin de que se muestre el aumento de CONTADOR y que enciendan las señales PIN_INCORRECTO, ADVERTENCIA y BLOQUEO. Para esta prueba es necesario escribir tres veces los dígitos incorrectos para obtener las tres señales, sin embargo, debido a que las tres veces se escriben los mismos dígitos en este plan de pruebas solamente se incluirá una de ellas.

Además, únicamente se necesita ingresar los dígitos, ya que al bloquearse, las demás entradas son innecesarias. A continuación se muestra la prueba correspondiente:

Primer dígito:

DIGITO = 8

DIGITO.STD = 1

DIGITO.STD = 0

Segundo dígito:

DIGITO = 1

DIGITO.STD = 1

DIGITO.STD = 0

Tercer dígito:

DIGITO = 2

DIGITO_STD = 1

DIGITO_STD = 0

Cuarto dígito:

DIGITO = 3

DIGITO_STD = 1

DIGITO_STD = 0

3. Instrucciones de Utilización de la Simulación.

Para poder observar la simulación se creó un archivo Makefile que permite generar dicha simulación. Al utilizar el comando *make* en la carpeta Tarea03 del repositorio de GitHub al que se puede ingresar más adelante, es posible ingresar a la herramienta GTKwave y observar el resultado de la simulación realizada. Cabe resaltar que para poder observar cada prueba tal y como se muestran en los resultados, se deben comentar las pruebas que no se quieran realizar. El no hacer esto provoca problemas en la lectura de los dígitos.

A continuación se muestra el enlace al repositorio de GitHub con los archivos necesarios:

https://github.com/andresnv12/Tareas_CircuitosII

4. Resultados.

A continuación se muestran los resultados de las distintas pruebas realizadas:

4.1. Resultados Prueba 1.

La siguiente figura muestra el resultado obtenido para la primera prueba:

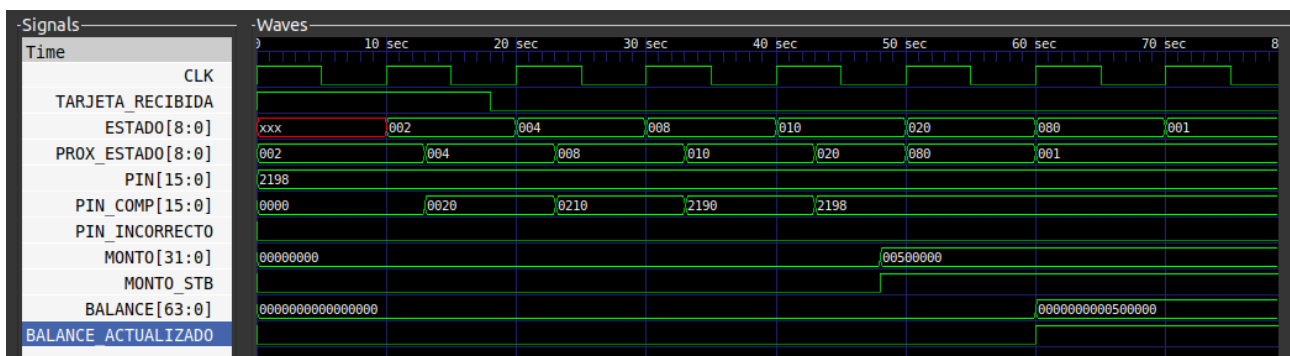


Figura 3: Prueba 1: Depósito.

Como se puede observar, en esta prueba se pasa por cada estado hasta llegar a un estado de Depósito. Los dígitos ingresados son correctos por lo que permite que PIN_COMP vaya

moviendo cuatro posiciones hacia la izquierda, de tal forma que al llegar al estado: Esperando cuarto dígito se obtenga el resultado correcto.

Ahora bien, nótese que en cuanto al valor de MONTO y BALANCE se obtienen los valores esperados y que además, se encienden las señales MONTO_STB y BALANCE_ACTUALIZADO en los momentos esperados, por lo tanto, dicha prueba fue exitosa.

4.2. Resultados Prueba 2.

A continuación se muestra la simulación realizada para la prueba 2:

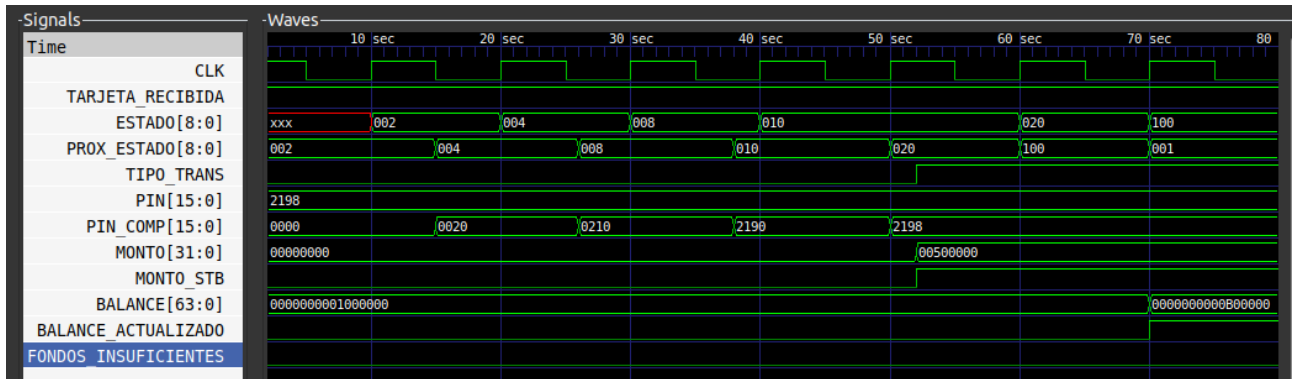


Figura 4: Prueba 2: Retiro (Fondos Suficientes).

Al igual que para la prueba anterior, se ingresaron los valores esperados con el fin de recorrer los primeros estados de forma satisfactoria. En esta prueba hubo un cambio el valor de TIPO_TRANS de tal forma que se cambiara a un estado de Retiro. Tal y como se puede observar en la figura anterior, se realiza la resta de forma correcta al realizar el retiro correspondiente, la señal de BALANCE_ACTUALIZADO se pone en 1 y la de FONDOS_INSUFICIENTES se mantiene en bajo como es esperado. Gracias a estos resultados se puede decir que esta prueba fue superada con éxito.

4.3. Resultados Prueba 3.

Para esta prueba se sigue una línea similar a la de la prueba anterior, sin embargo, se puso el valor de BALANCE en 0 para observar que se encienda correctamente la señal de de FONDOS_INSUFICIENTES. Como era de esperarse, el resultado fue exitoso, ya que dicha señal se encendió según lo esperado. A continuación se muestra la simulación que confirma dicho resultado:

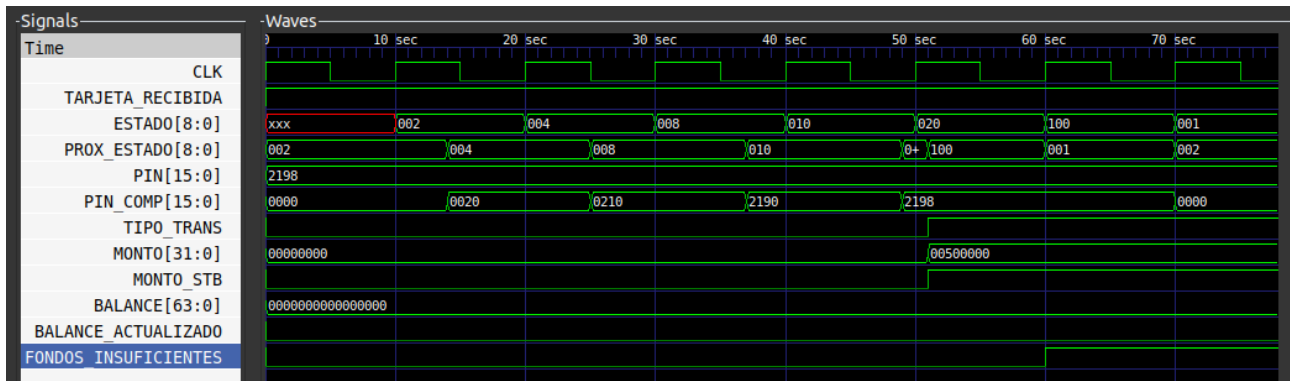


Figura 5: Prueba 3: Retiro (Fondos Insuficientes).

4.4. Resultados Prueba 4.

Finalmente se realizó la prueba de Bloqueo, la siguiente figura muestra la simulación obtenida:

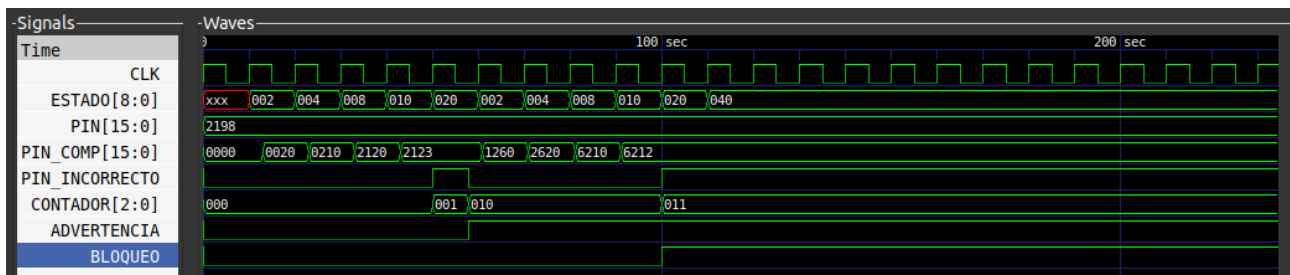


Figura 6: Prueba 4: Bloqueo.

A comparación de las anteriores pruebas, en esta se ingresaron tres veces los dígitos de forma errónea. Con el fin de llegar al estado de bloqueo del que únicamente se puede salir si la señal de RESET es 1. Según lo que se observa en la simulación anterior, se puede decir que dicha prueba se cumple parcialmente. Ya que sí encienden las señales esperadas de PIN_INCORRECTO, ADVERTENCIA y BLOQUEO, además de que CONTADOR aumenta hasta 3 según lo esperado.

Sin embargo, la señal de advertencia se activa antes de lo esperado, por lo que el tercer intento no llega a ser mostrado por la simulación, ya que se bloquea luego del segundo intento. Ahora bien, a pesar de no aprobar del todo la prueba realizada, se pensó en dos posibles soluciones a este problema. La primera consiste en crear un estado adicional al que se ingrese luego de que CONTADOR es mayor a 0, o bien, la segunda solución pretende utilizar un case para cada momento en que CONTADOR aumente.

4.5. Problemas con la Simulación.

Al realizar la simulación de las cuatro pruebas juntas se obtuvo problemas. Ya que existe una lectura errónea de los dígitos, lo que no permite obtener los casos de forma correcta. Este error se debe a la implementación errónea de los retardos. Que no se pudo resolver por falta de tiempo.

5. Conclusiones.

1. Mediante el uso de la herramienta GTKwave es posible analizar el resultado de las simulaciones para el controlador de Cajero Automático.
2. Los retardos incluidos pueden afectar el resultado de los dígitos ingresados y por ende, de la simulación en GTKwave al realizar todas las pruebas juntas.
3. El uso de Verilog permite implementar una máquina de estados de forma sencilla.

6. Recomendaciones.

- Se debe tener cuidado al añadir los retardos en las pruebas ya que al tener tantas entradas se pueden generar errores en las lecturas de los dígitos, lo que impide que se obtengan los resultados esperados.
- Con el fin de analizar mejor la simulación de la máquina de estados, es buena idea simular las pruebas una a una (en caso de que sean pocas) Para percibir mejor los cambios en las entradas y salidas.