

Práctica No. 5. Máquinas de estado

Simón Sánchez Rúa

Juan Camilo Arteaga Ibarra

Equipo 8

Digitales 1

Universidad de Antioquia

Facultad de ingeniería

Medellín

2022

Objetivos:

- Comprender y aplicar el concepto de máquinas de estado para la solución de problemas.
- Diseñar máquinas de estado que den respuestas a diversos sistemas.

Materiales:

- FPGA Basys 3 y Software vivado.

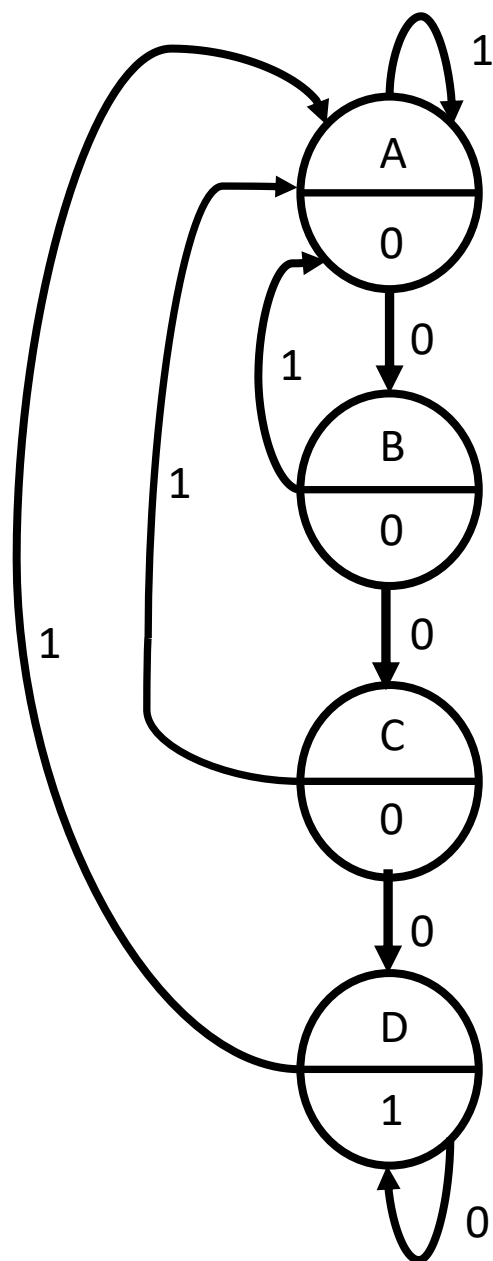
Descripción:

A partir de los ejemplos asignados, se realizan cambios los ejercicios propuestos en el github y se implementan haciendo uso del lenguaje VHDL una vez diseñados los diagramas de estado para cada uno de los ejemplos.

Ejemplo 1:

Del diseño planteado en [github](#) se realizan los cambios pertinentes y se diseñan los diagramas de estado para cada caso. La condición es que la respuesta sea 1 cuando la entrada es igual a 0 durante 3 o más intervalos de reloj.

- **Moore:**



Estado Inicial	Estado final		Z E.Inicial
	X=0	X=1	
A	B	A	0
B	C	A	0
C	D	A	0
D	D	A	1

Tabla 1: tabla de transición de estados de la máquina Moore Ej1.

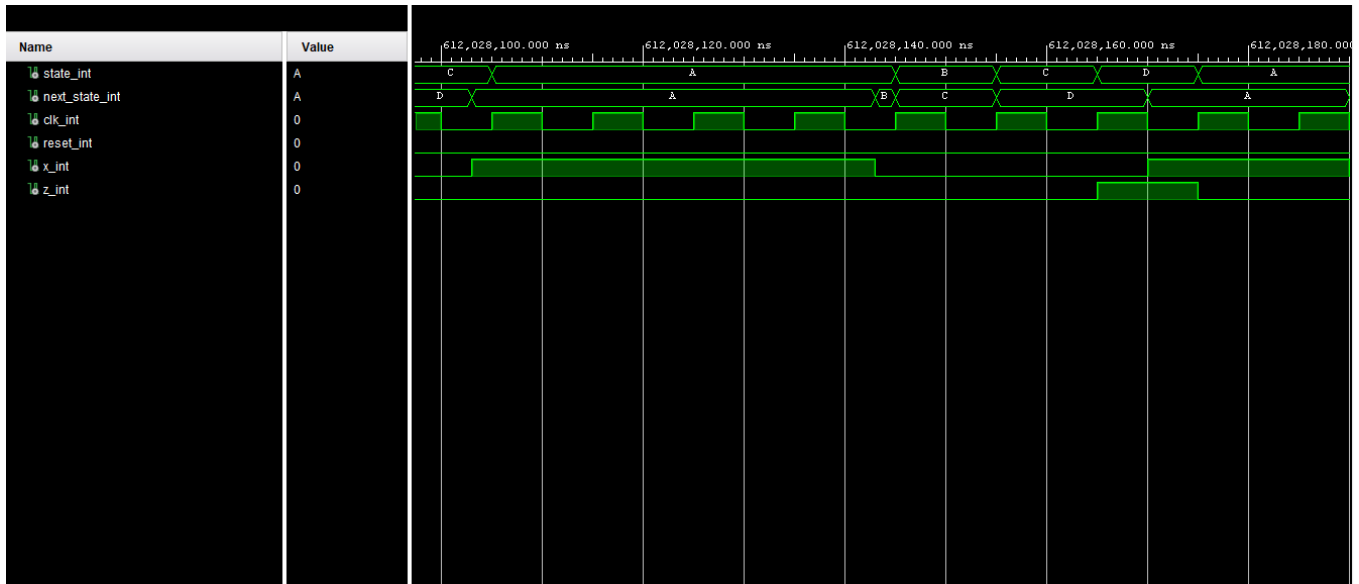


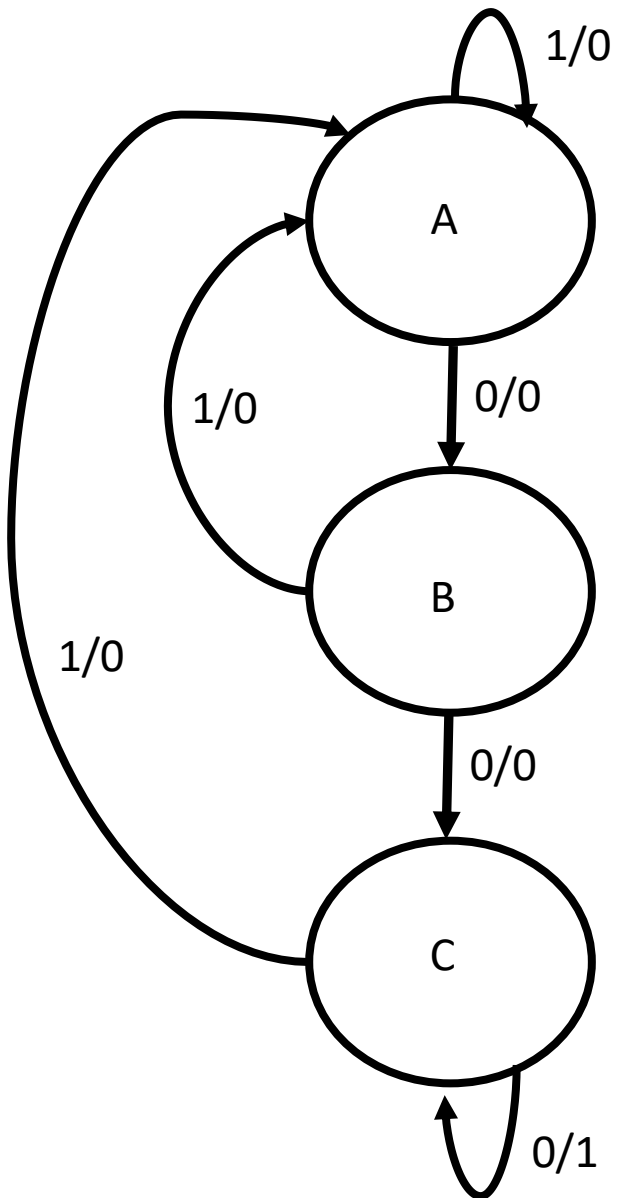
Ilustración 2: Waveform Moore Ejemplo 1.

Los cambios de estado se dan de la siguiente manera (estado actual es state_int, entrada x_int y salida z_int):

1. Cuando la entrada es 1, del estado actual C con salida 0 se pasa al estado A con salida 0.
2. Con la entrada manteniéndose como 1, el estado sigue en A con salida 0.
3. La entrada cambia a 0, donde se pasa del estado A con salida 0 al estado B con salida 0.
4. La entrada se mantiene en 0, así que del estado B con salida 0 se pasa al estado C con salida 0.
5. La entrada se mantiene en 0, el estado C con salida 0 pasa al estado D con salida 1.
6. La entrada pasa a ser 1, por ende el estado D con salida 1 pasa al estado A con salida 0.

Los cambios de estado funcionan como la máquina de estado planteada, por ende el código funciona como era esperado. [Ver código](#).

- Mealy:



Estado Inicial	Estado final/Z	
	X=0	X=1
A	B/0	A/0
B	C/0	A/0
C	D/1	A/0

Tabla 2: tabla de transición de estados de la máquina Mealy Ej1.

Ilustración 3: diagrama de estados Ej1 Mealy.

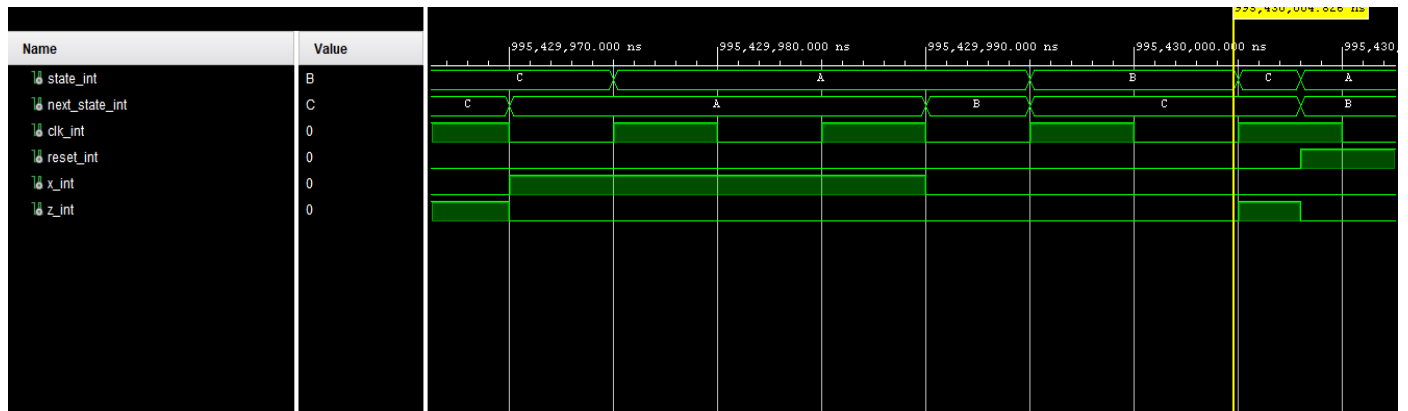


Ilustración 4: Waveform Mealy Ejemplo 1.

Los cambios de estado se dan de la siguiente manera (estado actual es state_int, entrada x_int y salida z_int):

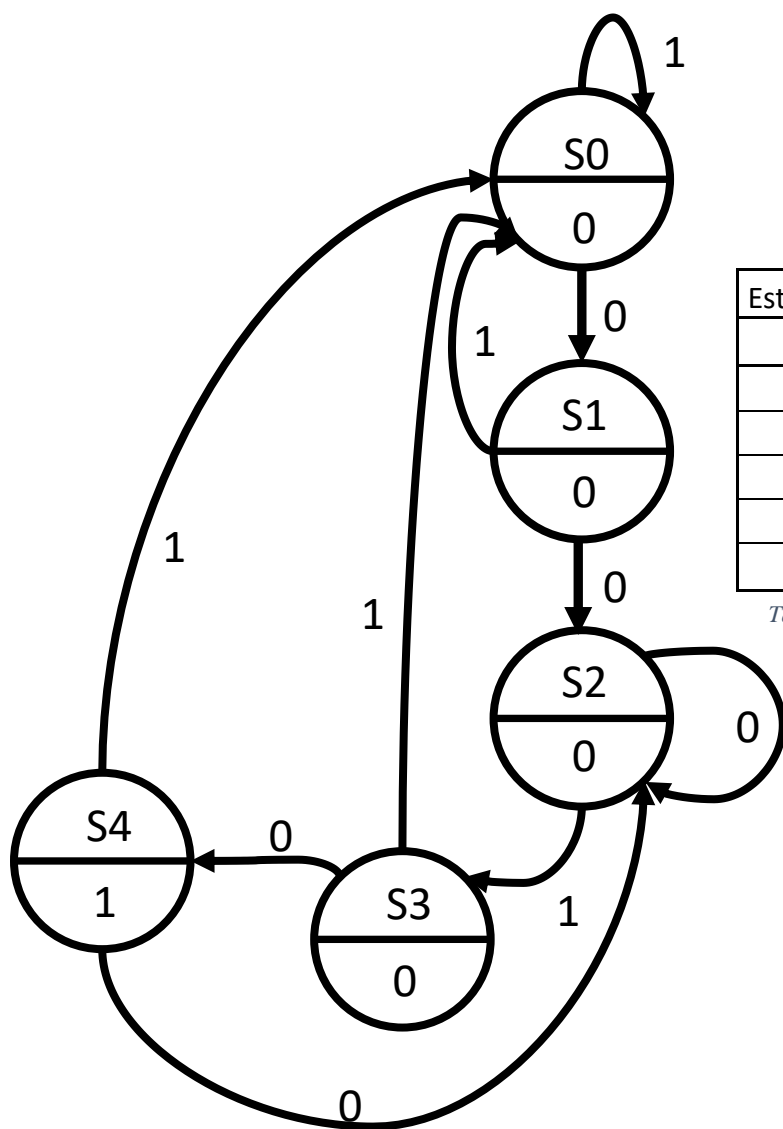
1. La entrada es 1, por eso el estado C con salida 1 pasa al estado A con salida 0.
2. La entrada se mantiene en 1, así que el estado A con salida 0 se mantiene igual.
3. La entrada pasa a 0, el estado A con salida 0 pasa al estado B con salida 0.
4. La entrada sigue en 0, entonces el estado B con salida 0 pasa al estado C con salida 1.
5. La entrada pasa a ser 1, el estado C con salida 1 pasa al estado A con salida 0.

Los estados y salida coinciden con el diagrama planteado y por ende el programa funciona como se deseaba. [Ver código](#).

Ejemplo 2:

Para este caso también se hace uso del ejemplo propuesto en [github](#), realizando las modificaciones pertinentes. La condición es que salga un 1 después de una secuencia de 0010, la repetición es permitida.

- **Moore:**



Estado Inicial	Estado final		Z E.Inicial
	X=0	X=1	
Moore			
S0	S1	S0	0
S1	S2	S0	0
S2	S2	S3	0
S3	S4	S0	0
S4	S2	S0	1

Tabla 3: tabla de transición de estados de la máquina Moore Ej2.

Ilustración 5: diagrama de estados Ej2 Moore.

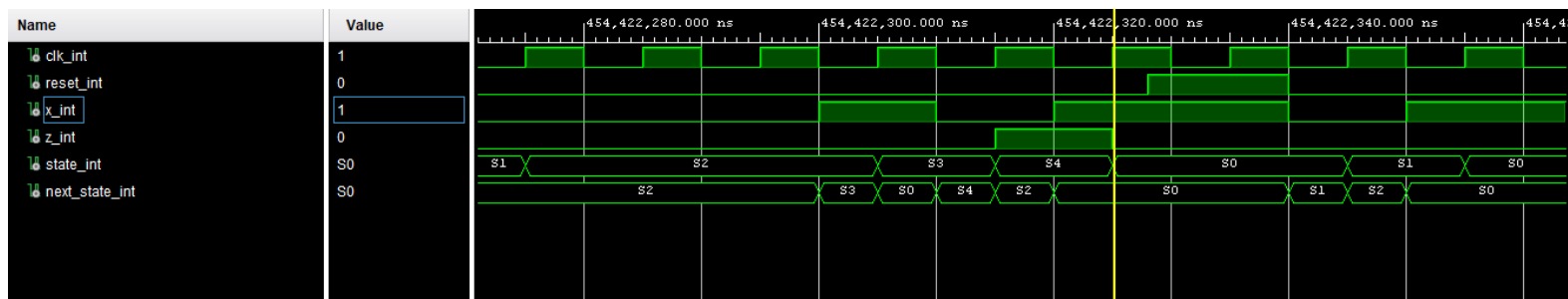


Ilustración 6: Waveform Moore Ejemplo 2.

Los cambios de estado se dan de la siguiente manera (estado actual es state_int, entrada x_int y salida z_int):

1. Con la entrada en 0, el estado S1 con salida 0 pasa al estado S2 con salida 0.
2. La entrada se mantiene en 0, así que el estado S2 con salida 0 se mantiene igual.
3. La entrada pasa a ser 1, entonces el estado S2 con salida 0 pasa al estado S3 con salida 0.
4. La entrada pasa a ser 0, entonces el estado S3 con salida 0 pasa al estado S4 con salida 1.
5. La entrada pasa a ser 1, entonces el estado S4 con salida 1 pasa al estado S0 con salida 0.
6. La entrada se mantiene en 1, así que el estado S0 con salida 0 se mantiene igual.
7. La entrada pasa a ser 0, el estado S0 con salida 0 pasa al estado S1 con salida 0.
8. La entrada pasa a ser 1, el estado S1 con salida 0 pasa al estado S0 con salida 0.

Las salidas y estados concuerdan con la máquina de estado planteada, por ende el código funciona según lo esperado. [Ver código](#).

- Mealy:

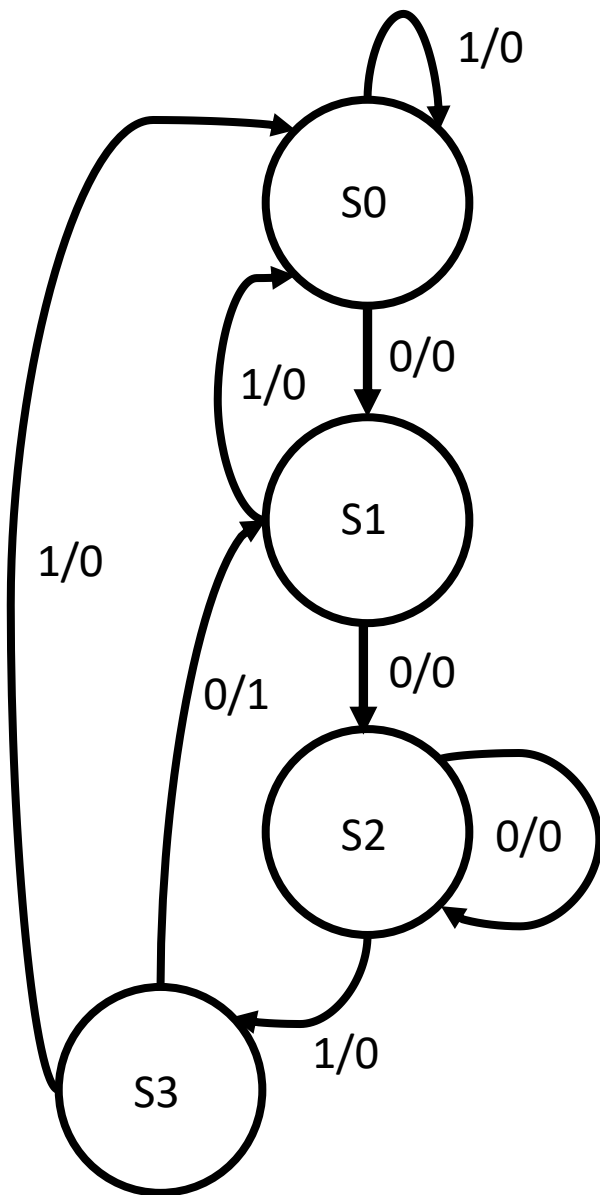


Ilustración 7: diagrama de estados Ej2 Mealy.

Estado Inicial	Estado final/Z	
Mealy	X=0	X=1
S0	S1/0	S0/0
S1	S2/0	S0/0
S2	S2/0	S3/0
S3	S1/1	S0/0

Tabla 4: tabla de transición de estados de la máquina Mealy Ej2.

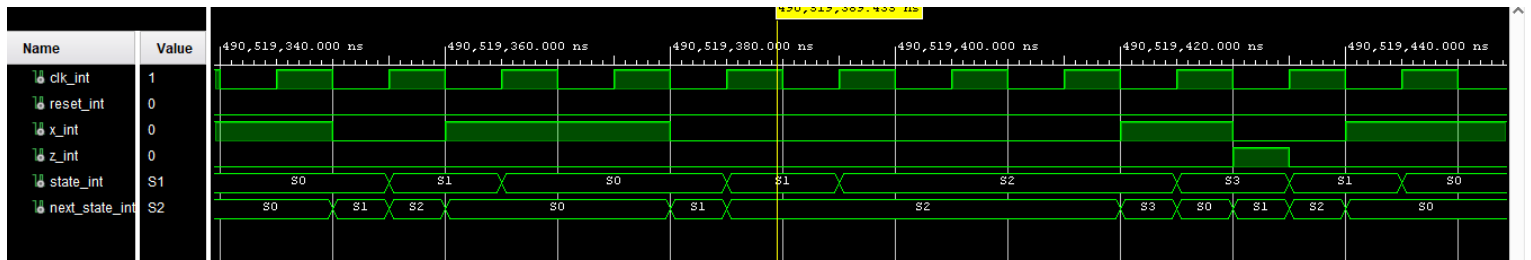


Ilustración 8: Waveform Mealy Ejemplo 2.

Los cambios de estado se dan de la siguiente manera (estado actual es state_int, entrada x_int y salida z_int):

1. La entrada está en 1, el estado S0 con salida 0 se mantiene igual.
2. La entrada pasa a 0, el estado S0 con salida 0 pasa al estado S1 con salida 0.
3. La entrada pasa a 1, el estado S1 con salida 0 pasa al estado S0 con salida 0.
4. La entrada se mantiene en 1, el estado S0 con salida 0 se mantiene igual.
5. La entrada pasa a 0, el estado S0 con salida 0 pasa al estado S1 con salida 0.
6. La entrada se mantiene en 0, el estado S1 con salida 0 pasa al estado S2 con salida 0.
7. La entrada se mantiene en 0, el estado S2 con salida 0 queda igual.
8. La entrada cambia a 1, el estado S2 con salida 0 pasa al estado S3 con salida 0.
9. La entrada cambia a 0, el estado S3 con salida 0 pasa a tener salida 1, luego pasa al estado S1 con salida 0.
10. La entrada cambia a 1, el estado S1 con salida 0 pasa al estado S0 con salida 0.

Las salidas y los estados son los mismos que se dan en el diagrama de estados, por ende el programa funciona correctamente. [Ver código](#).

Ejemplo 7:

Desde la propuesta en [github](#), se le añade un estado en el que se detiene el motor, adaptándolo al código propuesto y arreglando algunos errores que presenta el código base.

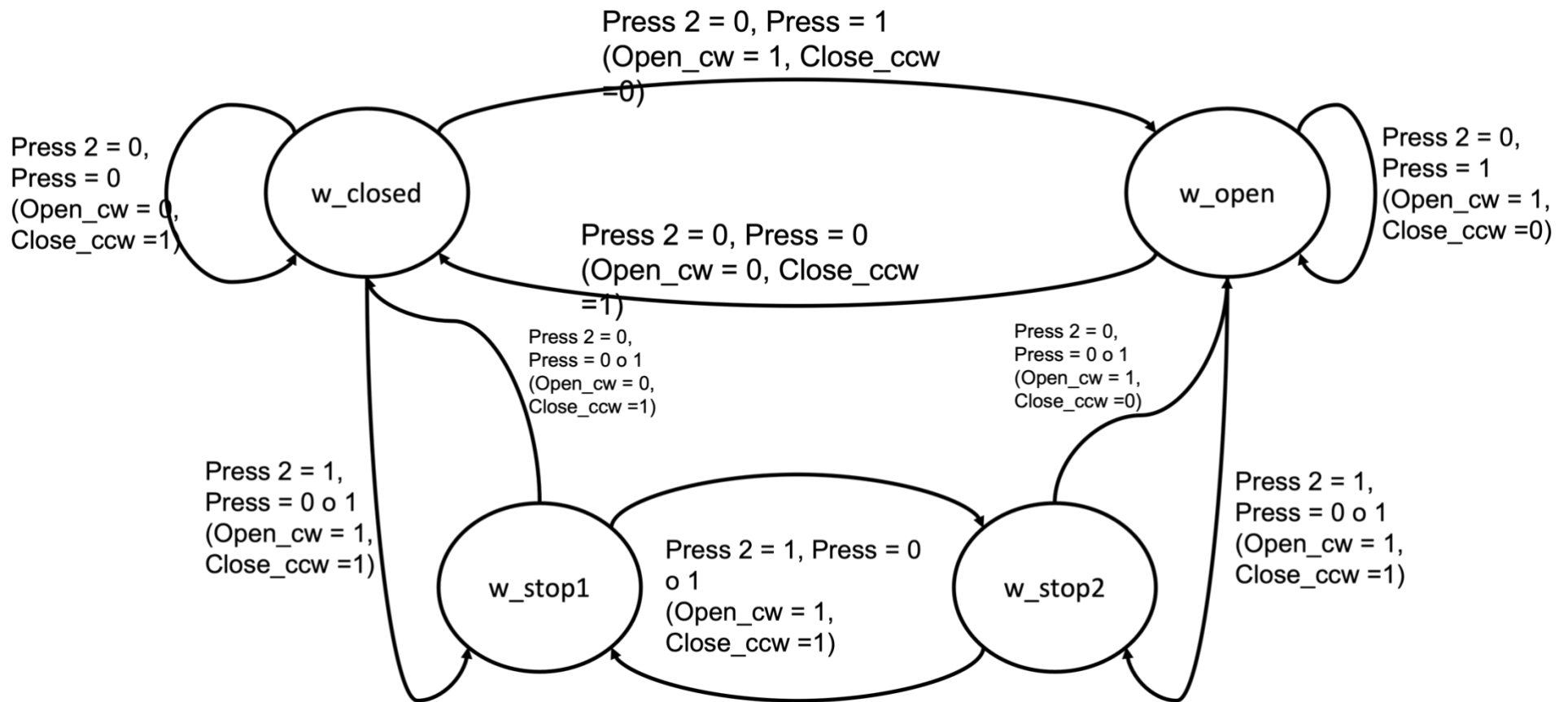


Ilustración 9: diagrama de estados Ej7.

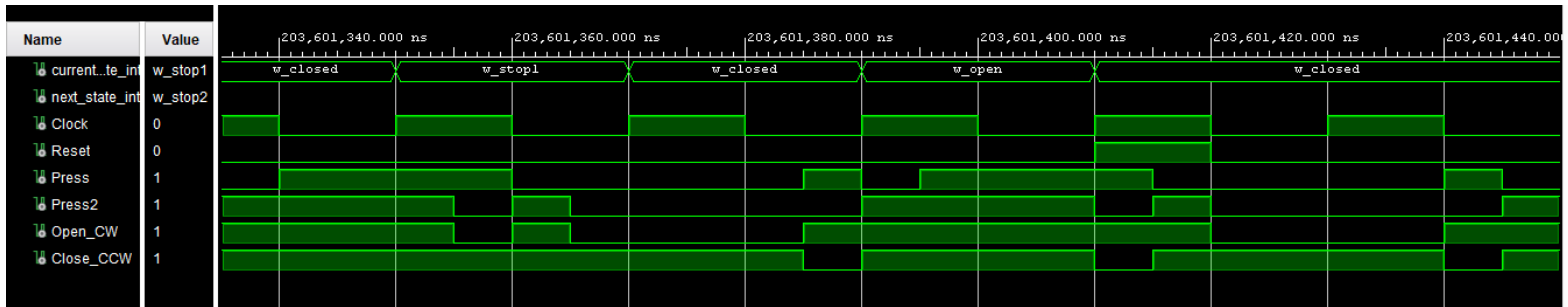


Ilustración 10: Waveform Motor Ejemplo 7.

El motor cambia de dirección de la siguiente forma (cambia de acorde a las subidas de reloj):

1. Cuando Press es 0 y Press2 pasa a 1, el estado w_closed pasa al estado w_stop1
2. Press2 pasa de 1 a 0, entonces el estado w_stop1 pasa al estado w_closed
3. Press2 se mantiene en 0 y Press pasa a 1, entonces el estado pasa de w_closed a w_open.
4. Press pasa a 1 y Press2 pasa a 0, entonces el estado pasa a w_closed porque momentos antes se dio un stop, por ende el estándar es que pase w_closed.
5. Se mantiene en w_closed debido a que Press y Press2 pasan a 0.
6. El Press pasa a 1 y hay un open.

Algo importante a recalcar es que al ser mealy las salidas cambian aun si no se ha dado el cambio de estado y por ende más de una vez los Open_CW y Close_CCW varían dependiendo de los botones Press.

El motor rota como se deseaba puesto que los estados se cumplen, así que el programa funciona correctamente incluso con los nuevos estados de stop. [Ver código](#).