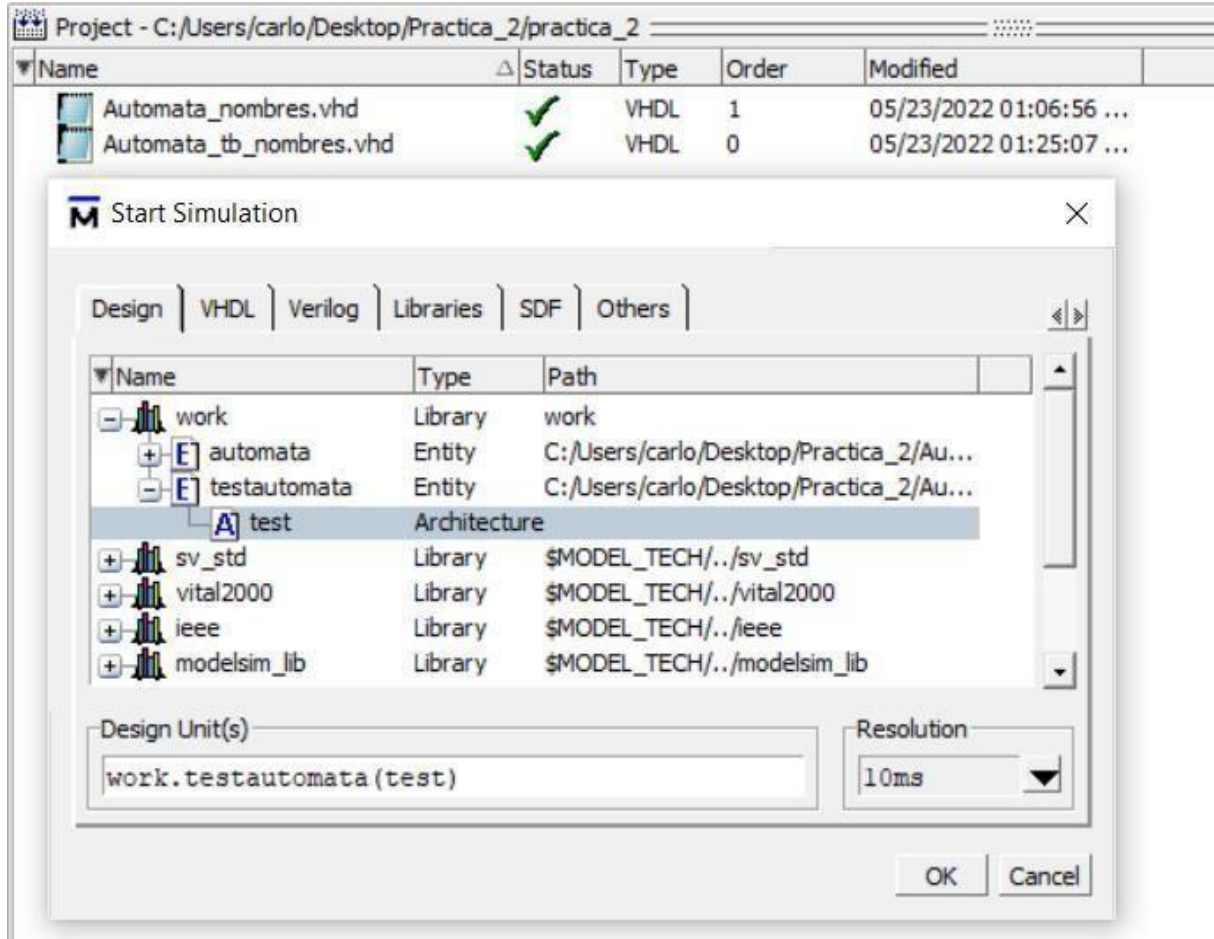


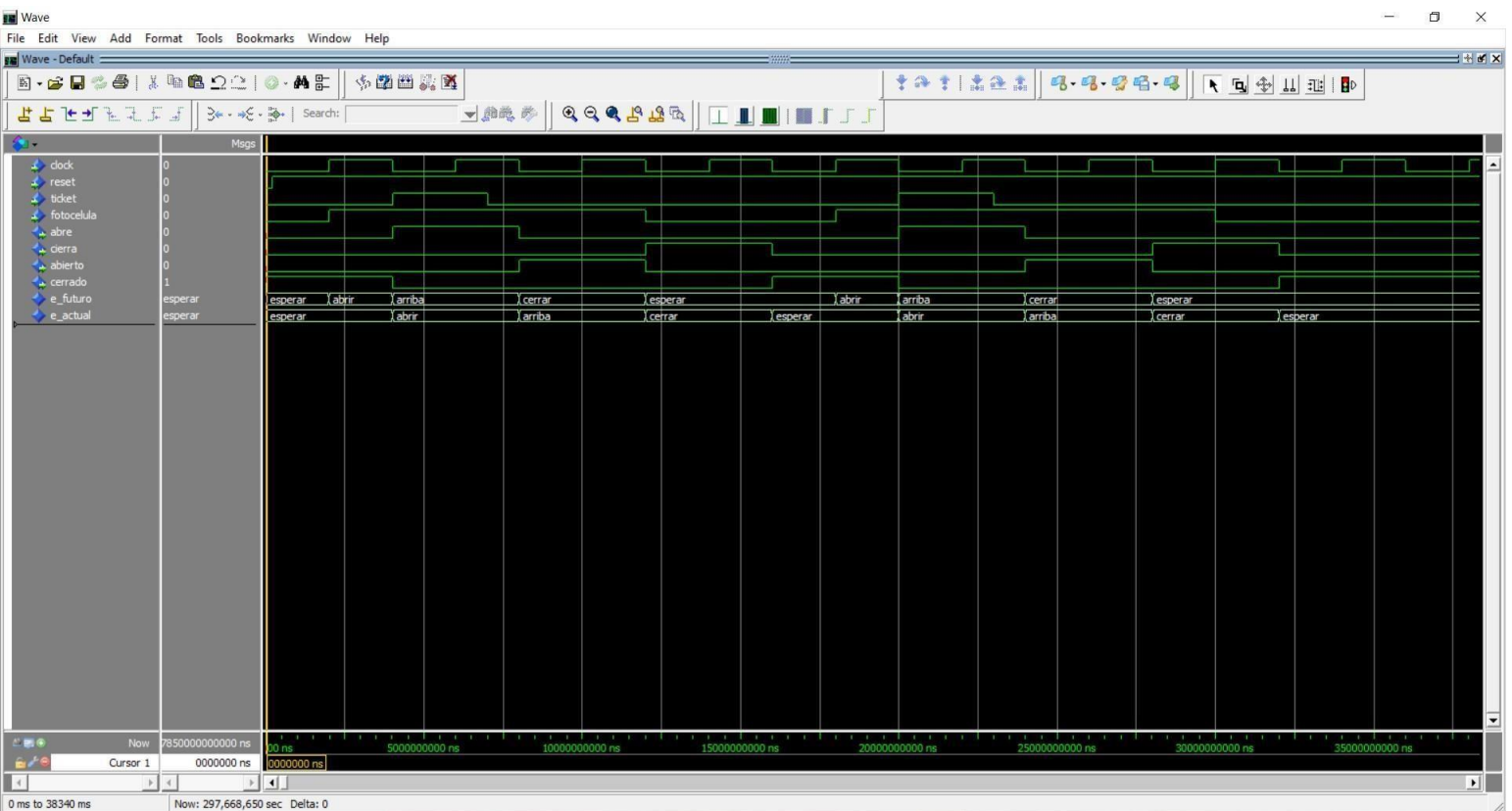
PRACTICA 2

MEMORIA DESCRIPTIVA DE LOS RESULTADOS

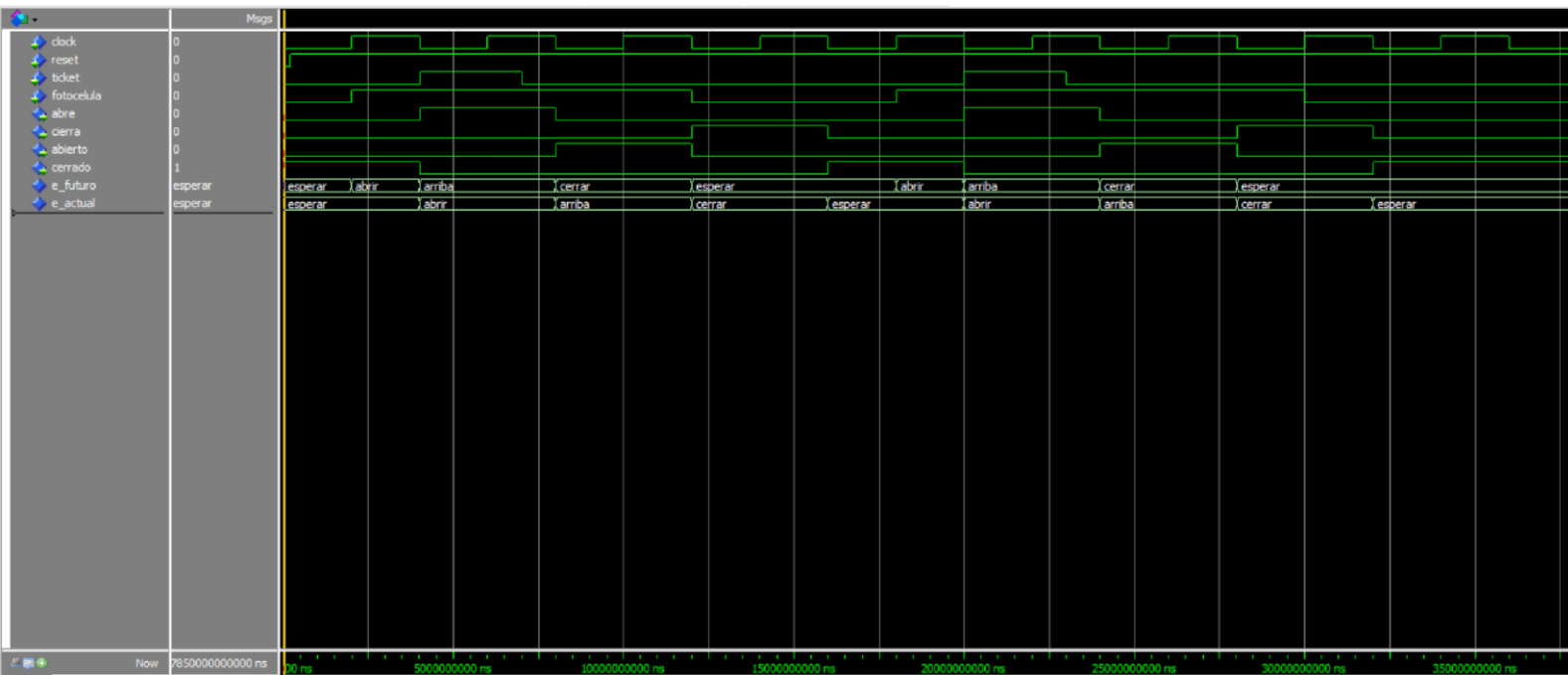
1. Captura de pantalla de la librería WORK una vez que los archivos estén compilados (Se deben visualizar todas las unidades de diseño del proyecto).



2. Captura de pantalla con los resultados de la simulación del circuito (visualizándose las formas de onda de las señales de entrada, salida, así como los nombres y secuenciamiento de los estados actual y futuro).



Gráfica más en detalle:



3. Breve análisis de los resultados anteriores donde se justifiquen los diferentes valores que toman las salidas para las diferentes combinaciones de las entradas.

Como se puede observar en la gráfica y acorde con las características del Autómata de Moore el valor de las salidas va a depender del estado actual, observándose así que para cada uno de los estados actuales que se actualiza en cada flanco de bajada pues le corresponde un '1' lógico a una y solo una de las salidas.

Por supuesto la designación de estados futuros también va acorde al valor de las entradas ticket y fotocélula que cumplen los requisitos de práctica y así se puede comprobar que cumple con las instrucciones o drivers del test en donde se introduce dos veces el ticket por lo que efectivamente vamos a tener dos procesos de esperar, abrir, arriba y cerrar como se puede ver en la gráfica.

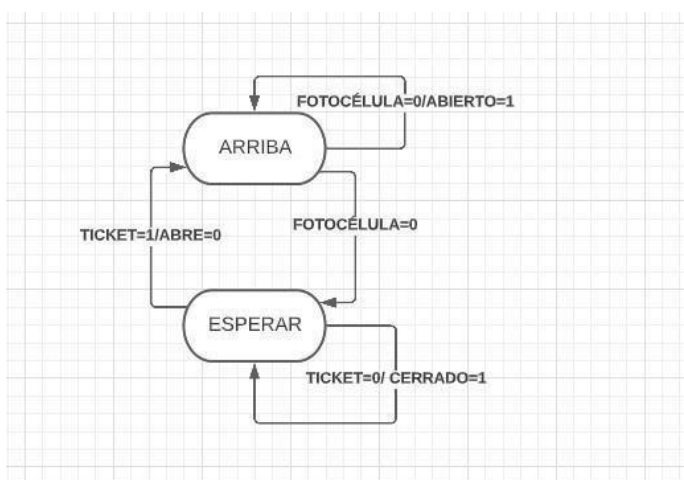
CUESTIONES:

1. Realizar el correspondiente autómata de Mealy con el menor número de estados posible
 - a. Especificar las variables de entrada, de estado y de salida

Entradas: Ticket, Fotocélula, Clock y Reset.
 Estados: Esperar, Abrir, Arriba y Cerrar.
 Salidas: Abierto, Cerrado.

Estas son las entradas, los estados y las salidas, propias del diagrama de flujo del autómata de Mealy que se piden en el enunciado.

- b. Representar el diagrama de flujo del autómata



- c. Al realizar la arquitectura de comportamiento del autómata de Mealy en VHDL, ¿Cuántos procesos utilizaría y qué señales incluiría en la lista de sensibilidad de estos?

PROCESS(TICKET, FOTOCELULA, E_ACTUAL)-- proceso para generar el estado futuro y salida

PROCESS(CLOCK, RESET)-- proceso para actualizar la memoria (estado actual)

2. Indicar de forma razonada las diferencias respecto al autómata de Moore.

En los autómatas de Moore las variables de salida toman sus valores en el estado actual, en cambio, en los autómatas de Mealy, hasta que las entradas no hacen que se produzca un cambio de estado o produzcan que vuelvas al mismo estado, las salidas no cambian . Es decir, las salidas en los autómatas de Mealy dependen del estado actual y de las entradas, mientras que en los autómatas de Moore las salidas solo dependen del estado actual.

Observaciones:

Cabe a mencionar que en nuestro código la declaración del Clock (Generación de la señal de reloj) la tenemos como “ clock <= NOT clock AFTER clock_perodo;” somos conscientes que el periodo de nuestro reloj es de 4 segundos o 4000 ms y el requerido en la práctica es de 2 segundos o 2000 ms y que se tiene que comprobar en la gráfica y que por ello el código debería ser “clock <= NOT clock AFTER clock_perodo/2;” pero al poner este periodo en nuestra gráfica se nos “duplican” todos los estados tanto de estado actual (e_actual) como de estado futuro(e_futuro) y que eso a parte de dejar la gráfica sin sentido alguno ya que “duplican” todos los procesos y nos hace imposible hacer la captura de la misma, por ello es que hemos decidido escoger ese periodo para generar la señal de reloj.

También queremos destacar que el código se ejecuta hasta una gran cantidad de ns debido a que en el test no especifica ninguna señal de terminar con la que podamos añadir un wait al código y detener así la simulación.