

DISEÑO DE PROCESADORES DEDICADOS

SIMULACIÓN CON ISIM

M. en C. Juan Carlos Herrera Lozada
jlozada@ipn.mx

Instituto Politécnico Nacional



Centro de Innovación y Desarrollo
Tecnológico en Cómputo

CIDETEC

Instala ISE WebPack contenido en Suite 14.7. Utiliza el formato de la práctica 1 para generar un diseño y llegar hasta el diagrama esquemático del mismo. Consideremos los archivos disponibles para la práctica, ALU4.vhd y CONTA.vhd, ambos contenidos en el archivo p1.zip disponible en la página web de nuestro curso.

1. Simulando ALU4.vhd (Circuito de Lógica Combinatoria)

Una vez generado el símbolo esquemático de la ALU a partir del código VHDL, se crea el diagrama esquemático con la macro y sus respectivos *pads* de entrada/salida. Observa la figura 1. Para llegar a este punto, auxiliate de la primera parte de la práctica 1.

- 1 Accede a la ventanilla *Design* y cambia de *Implementation* a *Simulation*. En este último modo debe estar habilitada la simulación *Behavioral*.
- 2 Recuerda que se trabajará sobre el archivo *sch*, correspondiente al diagrama esquemático que es el nivel jerárquico mayor en nuestro diseño. Marcando con un click del mouse el archivo esquemático podremos acceder a los procesos disponibles para este archivo.

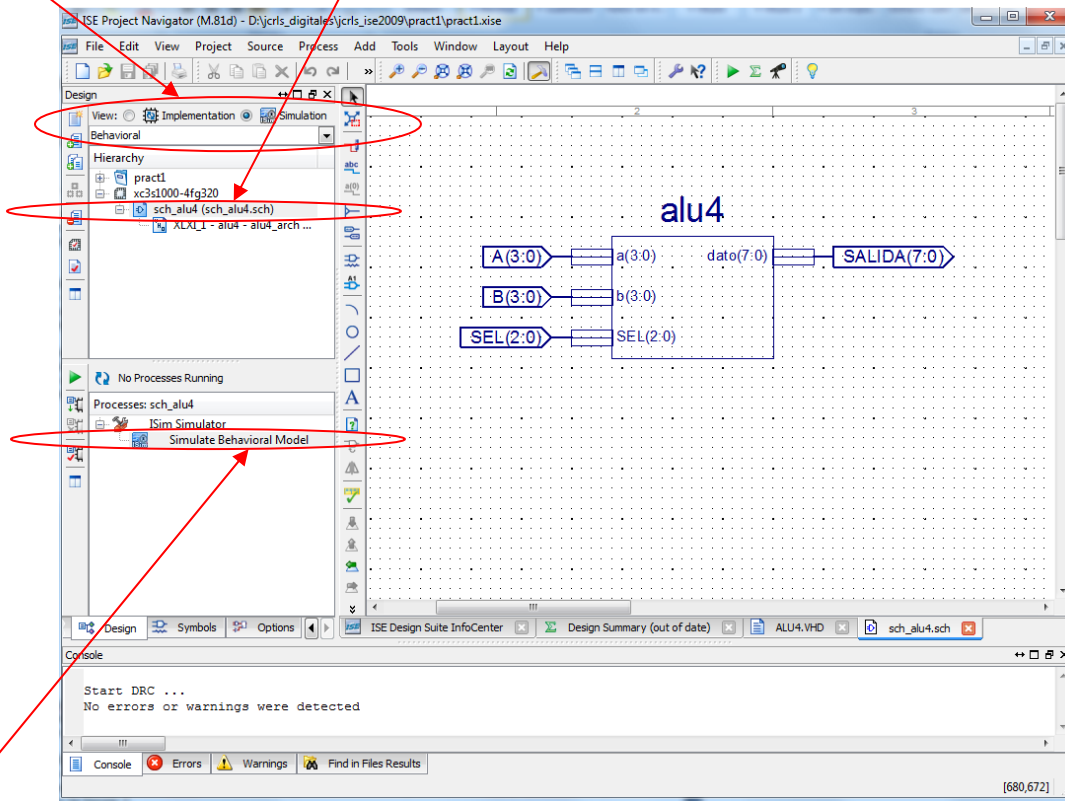


Figura 1.

- 3 Daremos doble click sobre la opción *Simulate Behavioral Model*. En este punto es importante que observes que no es necesario crear un archivo de simulación a diferencia de lo que hacemos con ModelSim (el simulador para la versión 10.1i de WebPack), aunque claro, si tu idea es conservar el archivo, lo recomendable es que lo hagas. Se abrirá ISIM como se aprecia en la figura 2. La ventana completa de ISIM está dividida por omisión en tres columnas: *name*, *value* y *wave form*. En la columna *name* están listadas todas las variables de entrada y de salida que declaraste en tu diseño. En la columna *value* aparecen los posibles valores de las variables; en este

mismo rubro, *UUUU* hace referencia a un dato de 4 bits al que no se le ha asignado algún valor. Finalmente la columna *wave form* muestra las formas de onda de la simulación.

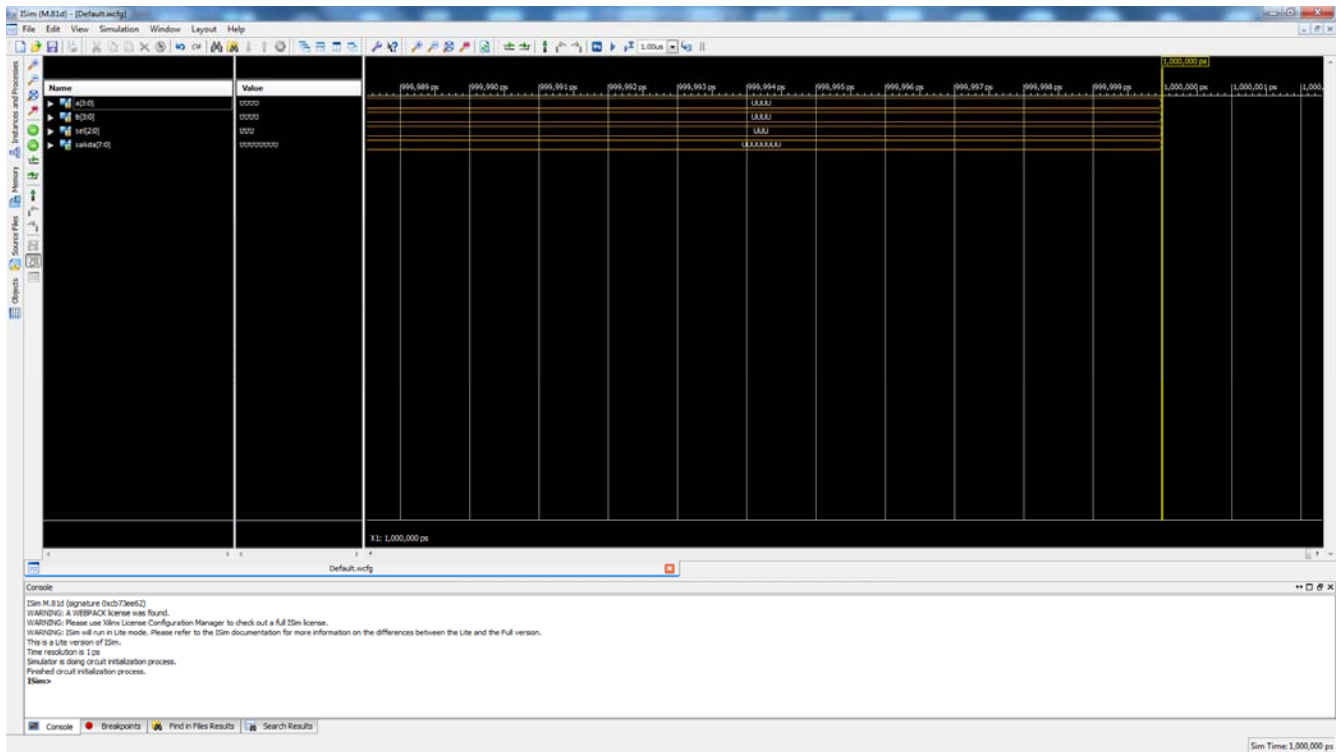


Figura 2.

4 Para escribir los estímulos para la simulación es necesario posicionarse sobre la variable de entrada que te interesa trabajar y con el botón derecho del mouse sobre la columna *value* de esta misma variable, accede a las opciones de edición y selecciona *Force constant*, tal y como se aprecia en la figura 3.

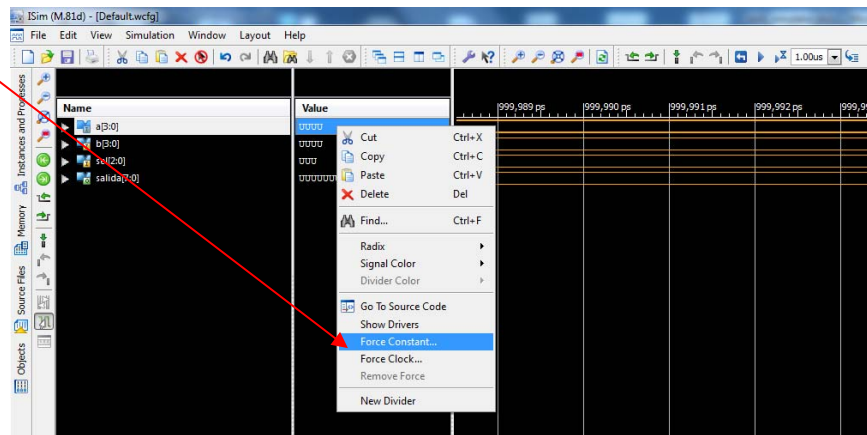


Figura 3.

5 En la ventana emergente escribiremos el valor de la variable. Observa que por omisión te dejará capturar sólo un valor binario de 4 bits (en el caso de la variable A de nuestra ALU). En la figura 4 se aprecia la captura de este dato, que para ejemplificar es 1010. Los últimos campos de este cuadro de diálogo se mantendrán sin cambio. Aplica y acepta las modificaciones.

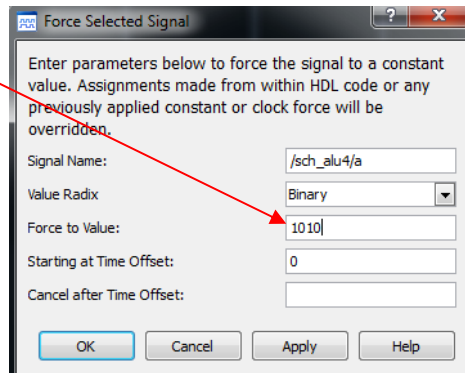


Figura 4

6 No notarás que el dato introducido se actualice inmediatamente en la columna *value*, sin embargo puedes observar que en la consola se indica la actividad realizada, tal y como lo muestra la figura 5.

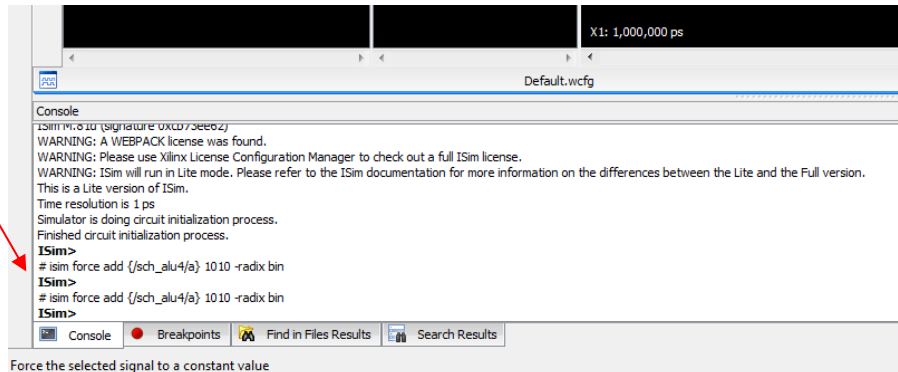


Figura 5.

7 Debes completar los estímulos para todas las variables de entrada. De acuerdo al código VHDL de la ALU, la variable B también es de 4 bits y la selección es de 3 bits. Recuerda que no modificaremos la salida pues es el resultado que nos interesa monitorear. La consola debe mostrar las modificaciones de la figura 6.

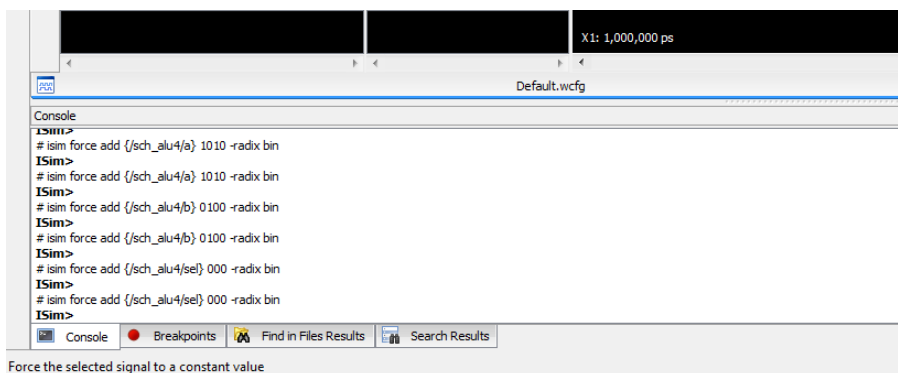


Figura 6.

- 8) Para proceder a la simulación con los estímulos capturados debes presionar en una sola ocasión el botón *Run for the time specified on the toolbar*, indicado en la figura 7.

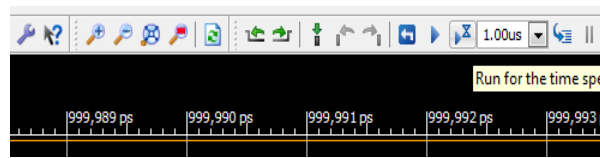


Figura 7.

- 9) Las formas de onda se actualizarán mostrando el valor capturado para cada variable de entrada y la salida mostrará su correspondiente resultado. Observa la figura 8. Recuerda que el resultado estará desfasado un ciclo en la simulación. Para seguir adicionando estímulos, repite desde el paso 4 sobre el mismo archivo. Para el caso particular de la ALU puedes dejar fijos los valores de A y de B (considerando siempre $A \geq B$) y sólo cambiar la selección de la operación desde 000 hasta 111. Cada vez que captures un nuevo dato de selección, presiona el botón *Run for the time specified on the toolbar* para avanzar en la simulación.

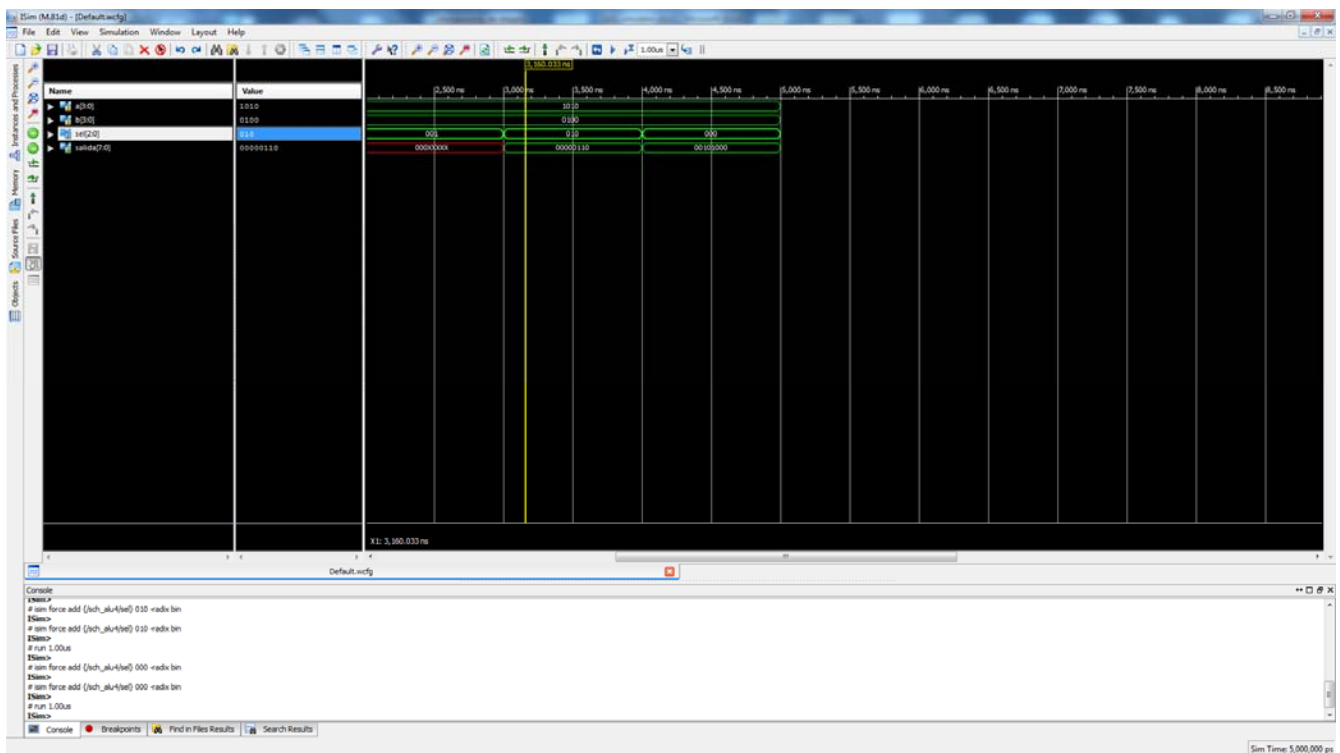


Figura 8.

2. Simulando CONTA.vhd (Circuito de Lógica Secuencial)

El procedimiento es el mismo que se utilizó para la ALU, sólo se adecuará el estímulo para considerar una señal de reloj. Comencemos considerando la figura 9.

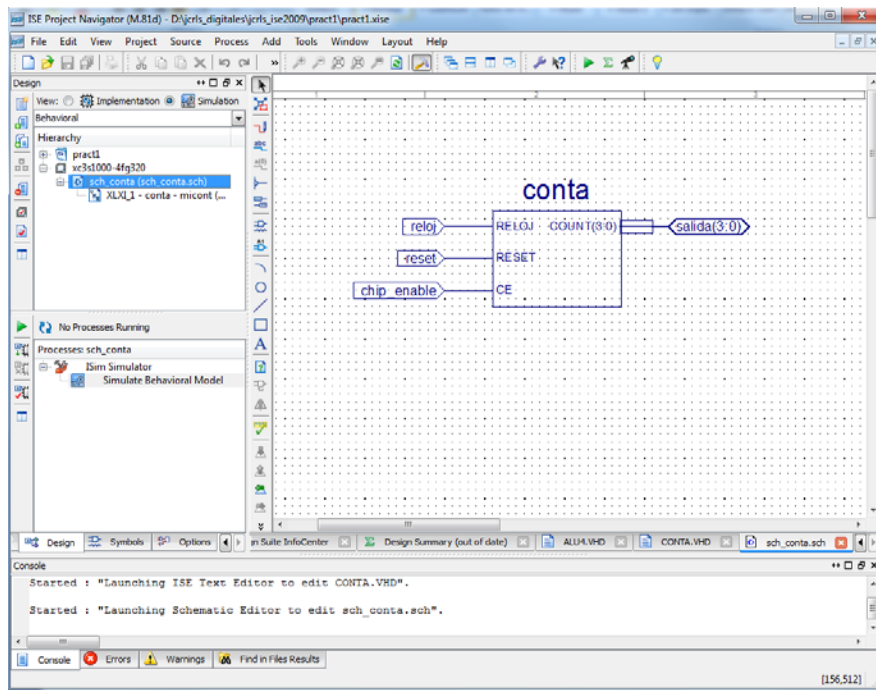


Figura 9.

Cuando presionemos *Simulate Behavioral Model* y se abra ISIM, capturaremos el dato para CE y para Reset (analiza el código VHDL para saber cómo funcionan) tal y como se hizo para la ALU. Para el caso de la señal de reloj, posicionándonos en la columna value de la variable reloj y con el botón derecho del mouse, accederemos a las opciones de edición y seleccionaremos Force Clock, como se aprecia en la figura 10. Aplica y Acepta las modificaciones.

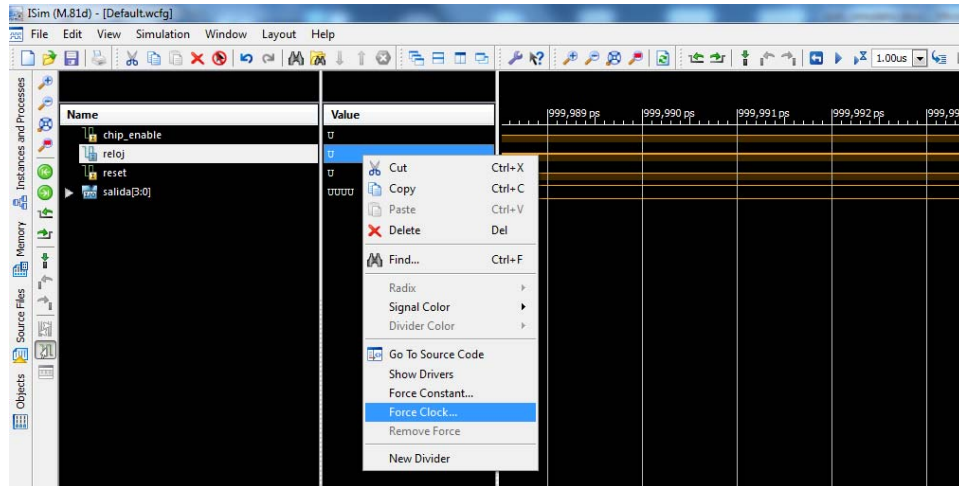


Figura 10.

En la ventana emergente escribiremos los siguientes datos:

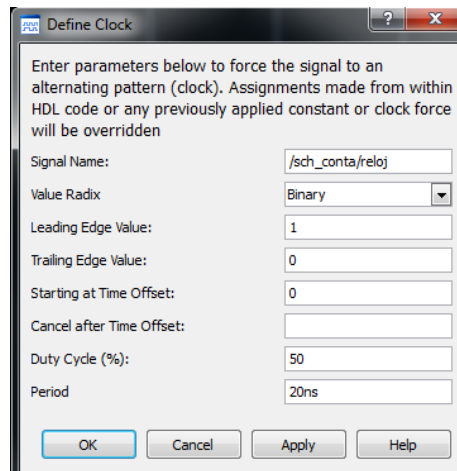


Figura 11.

Posteriormente ejecuta tu simulación como se hizo con la ALU. Tu simulación resultará como se aprecia en la figura 12.

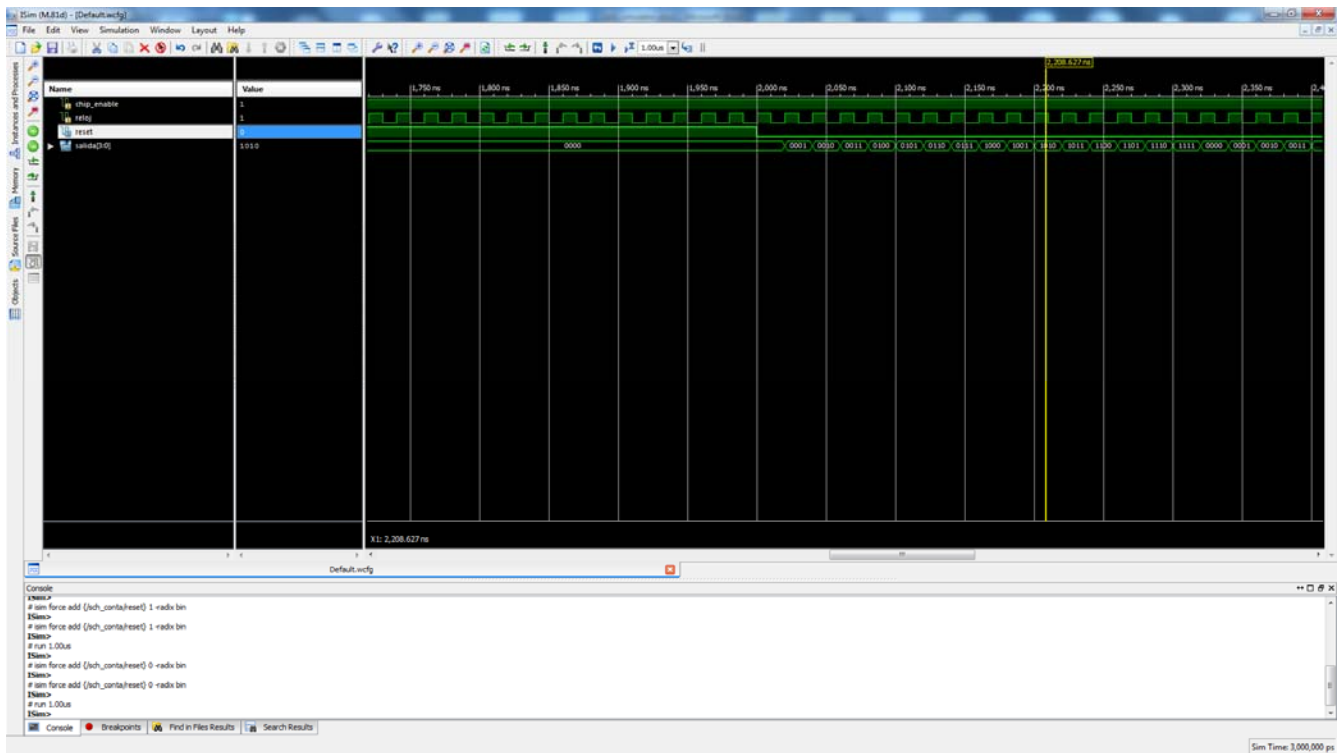


Figura 12.

Para que funcione correctamente esta simulación, te recomiendo un reset inicial utilizando CE=1 y Reset=1 para la primera ejecución y posteriormente CE=1 con Reset=0 que permitirá el incremento normal en el contador con la segunda ejecución de la simulación.