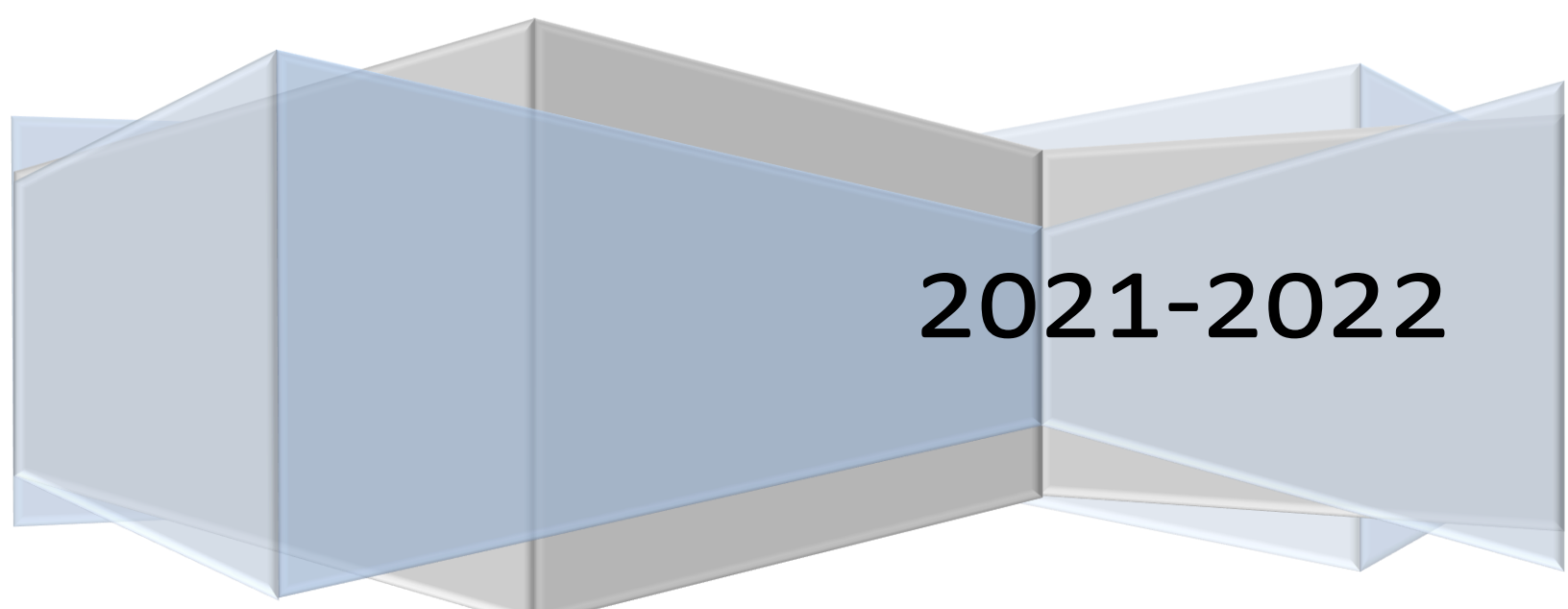


Intel® Quartus Prime

Simulación

Ignacio Blázquez



2021-2022

Contenido

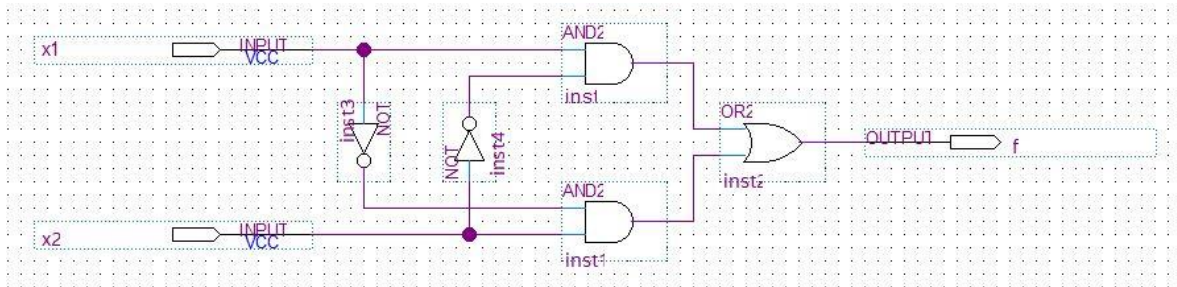
1. Introducción	2
2. Editor de formas de Onda	3
3. Simulación	13
4. Errores frecuentes.....	16

1. Introducción

Una vez introducido nuestro diseño, ya sea mediante esquemático o en VHDL, y verificado que no existen errores mediante la ejecución de un Análisis y Síntesis del mismo, es importante realizar simulaciones para comprobar que el diseño introducido se comporta en realidad como esperamos.

Para ello, el Quartus Prime ofrece la posibilidad de realizar simulaciones mediante software externo de simulación de sistema digitales, pero también incorpora un simulador interno, el ModelSim-Altera, que nos permite realizar simulaciones lógicas de nuestro diseño sin necesidad de utilizar otros simuladores externos.

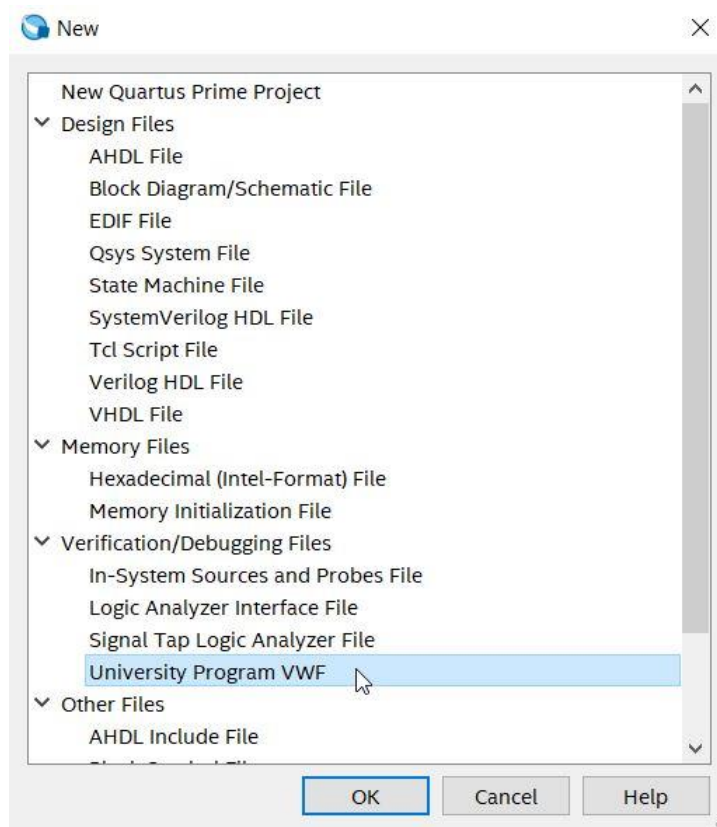
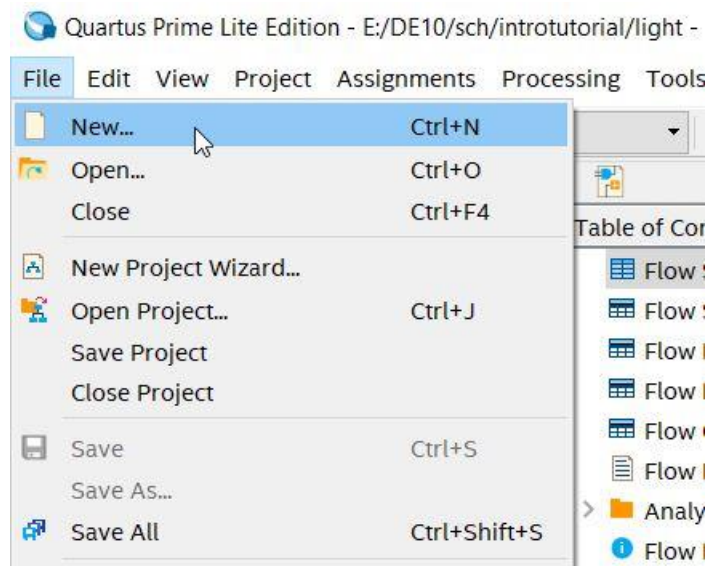
Para este tutorial, partiremos de un diseño de un sencillo sistema ya introducido en el Quartus mediante esquemático. El sistema dispone de dos entradas, x1 y x2, y una salida, f, y el diagrama de puertas lógicas es el de la figura:



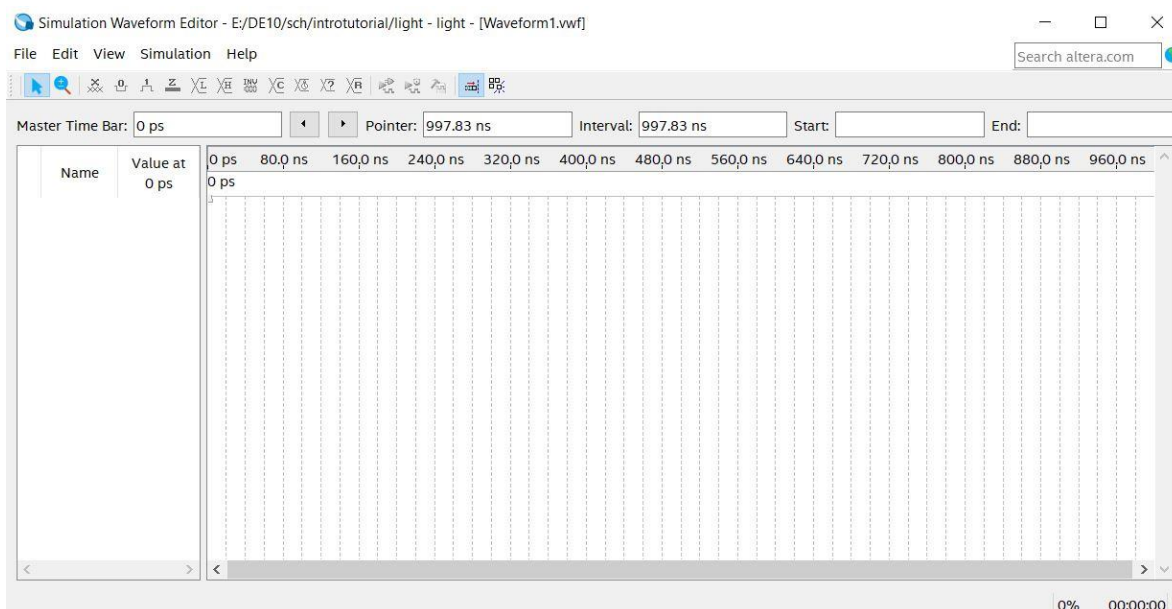
En los siguientes apartados explicaremos el proceso para generar un fichero de formas de onda con los estímulos (combinaciones de valores que queremos aplicar a las entradas de nuestro circuito, x1 y x2) y realizar una simulación del comportamiento del mismo, obteniendo como resultado el valor lógico que adopta la salida f para cada una de las combinaciones aplicadas en la entrada, lo que nos permitirá comprobar si el circuito se comporta como nosotros deseábamos.

2. Editor de formas de Onda

Para abrir el editor de formas de onda debemos seleccionar mediante los menús la opción **File > New > Verification/Debugging Files > University Program VWF**:

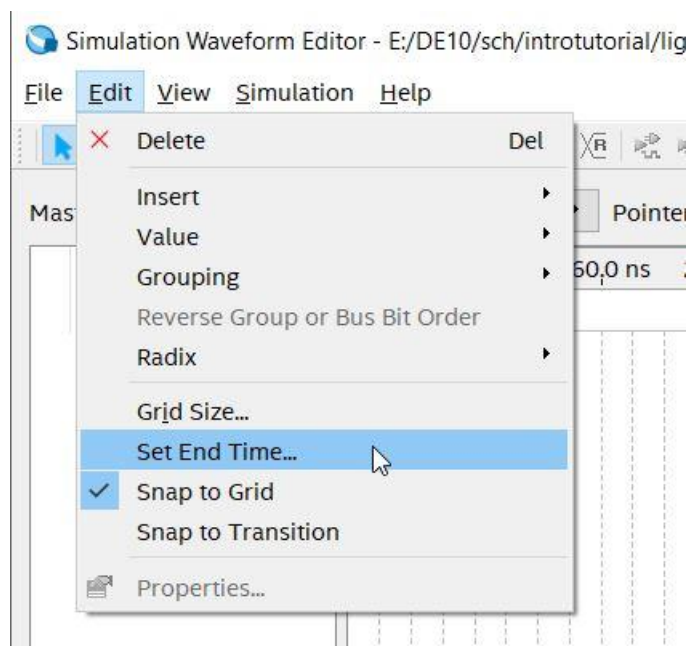


Al seleccionar **OK** se nos abrirá la ventana principal del Editor de Formas de Onda:

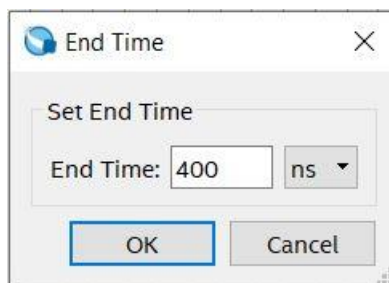


Al disponer nuestro circuito de sólo dos entradas, bastará con generar las cuatro combinaciones binarias de dos bits para verificar que el circuito siempre se comporta de forma correcta.

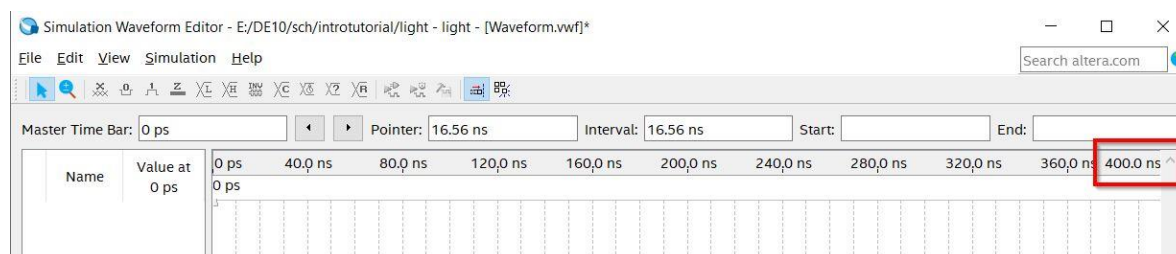
Podemos restringir la duración de la simulación a, por ejemplo, 400ns. Para ello seleccionamos en el menú **Edit > Set End Time...**:



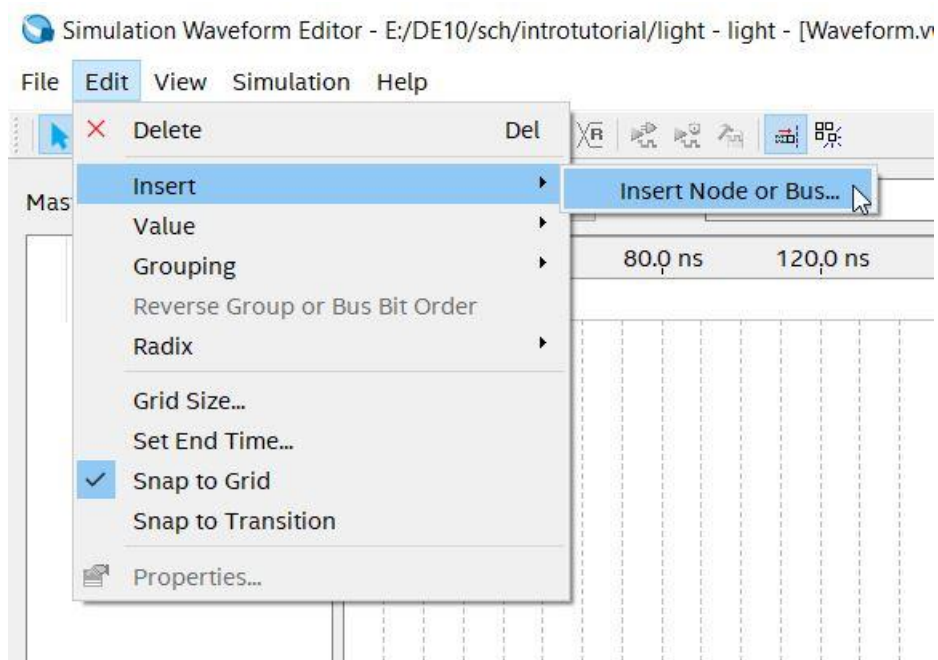
y en la ventana siguiente tecleamos 400 y escogemos nanosegundos (ns) como unidad de tiempo:



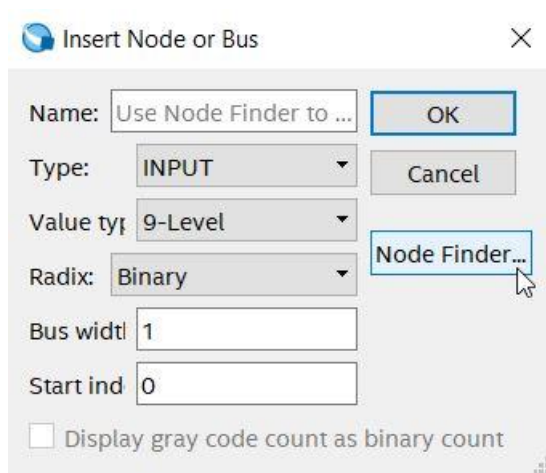
Al seleccionar OK la ventana principal del Editor de Formas de Onda habrá ajustado la escala de tiempos entre 0 y 400 ns:



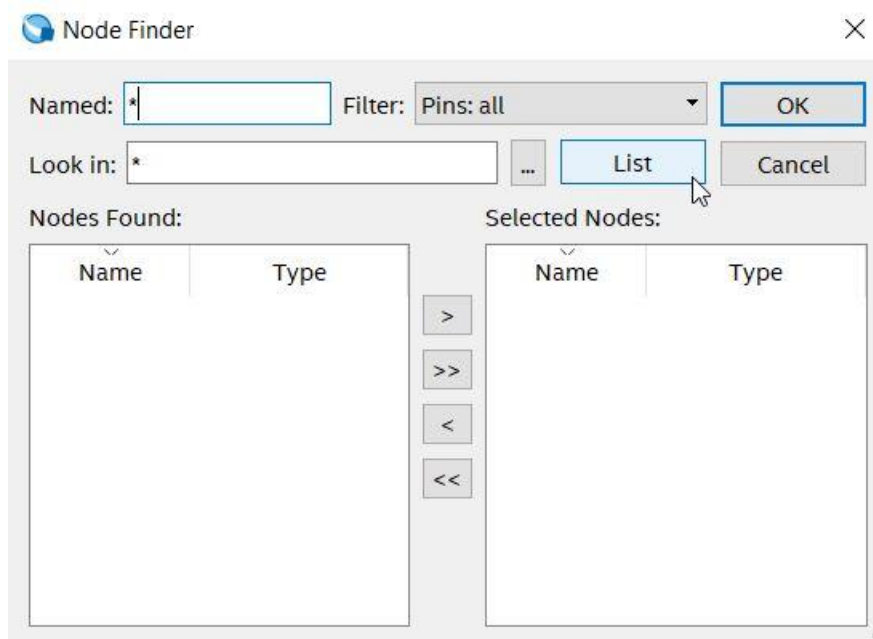
A continuación, debemos incorporar al editor las señales de entrada a las que queremos asignar valores y las señales de salida cuyo valor después de la simulación queremos observar. Para ello, seleccionaremos la opción del menú **Edit > Insert > Insert Node or Bus...**:



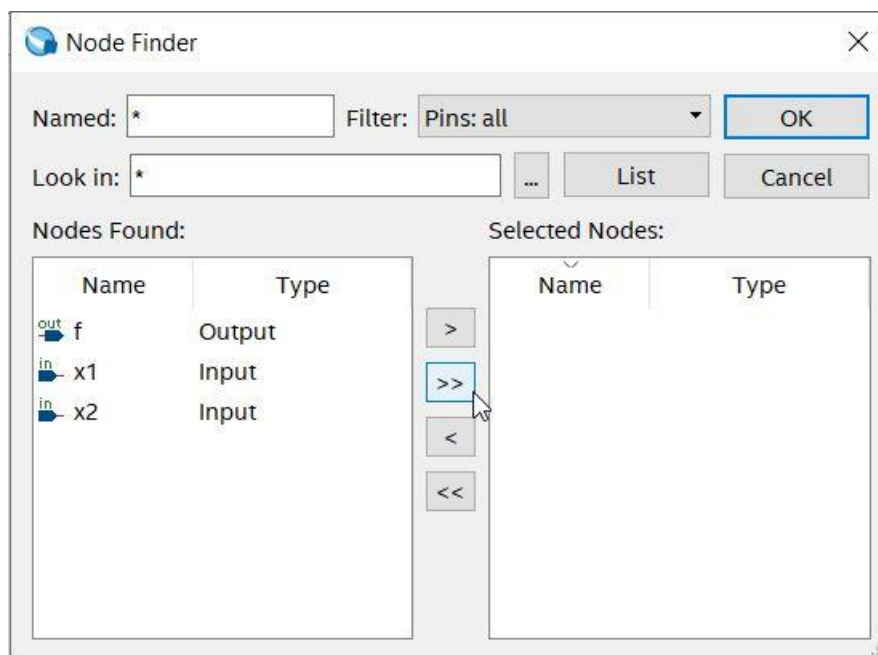
En la siguiente ventana seleccionamos **Node Finder**:



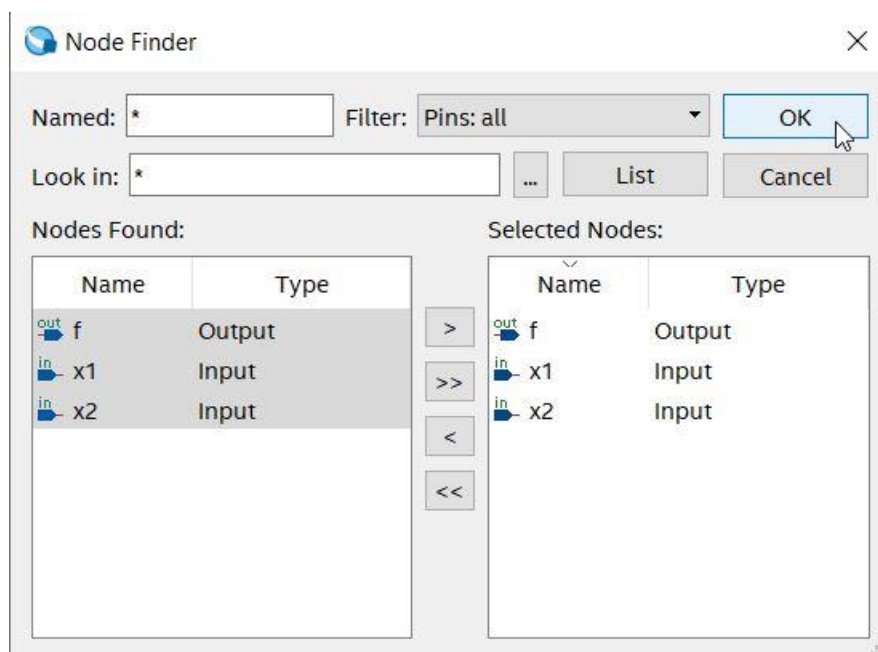
y a continuación **List**:



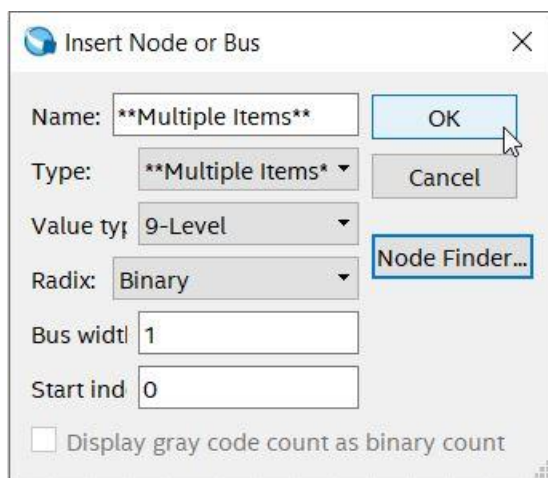
Nos aparecen a la izquierda las señales de entrada y de salida de nuestro circuito:



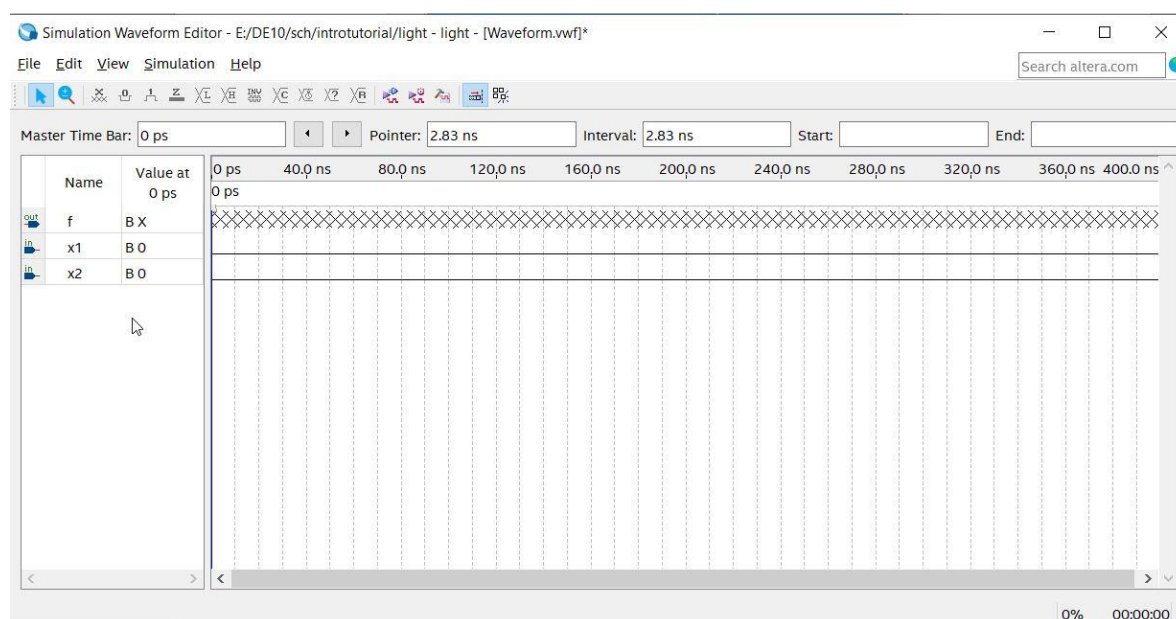
Podemos seleccionar o deseleccionar las que nos interese incorporar al editor de una en una mediante los botones > y <, o todas de golpe mediante >> y <<. Si usamos >> nos aparecen todas las entradas y salidas como seleccionadas en la ventana de la derecha:



Para volver a la ventana principal del editor, seleccionamos **OK** dos veces:

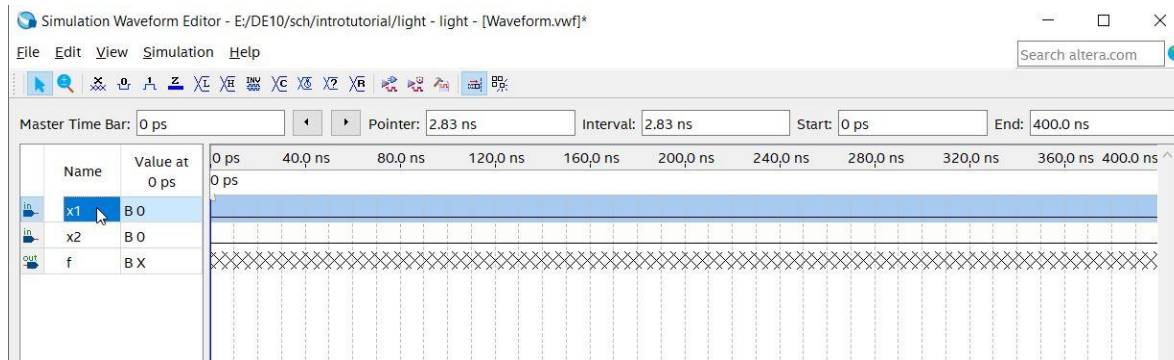


Ya tenemos las entradas y la salida de nuestro circuito incorporadas al editor:

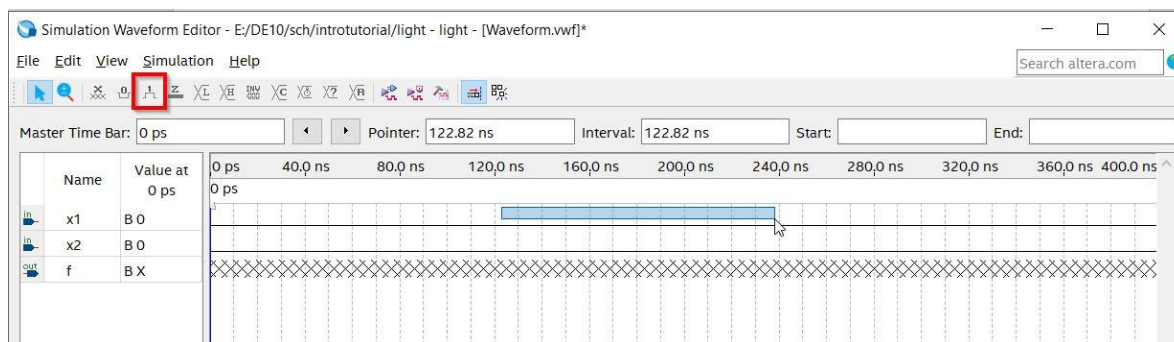


Observemos que se asigna el valor por defecto “0” a las entradas x1 y x2 durante todo el intervalo de simulación, y a la salida f un entramado de equis, “XXX”, indicando que es una salida y no se puede editar este valor.

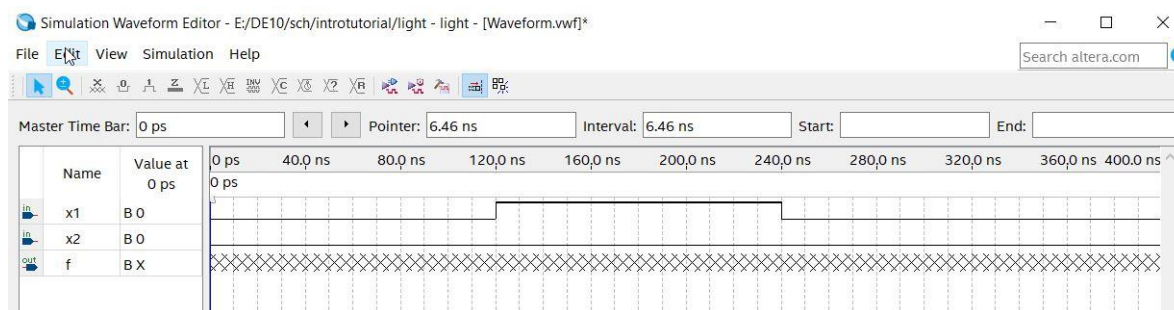
Normalmente nos convendrá situar primero, en la parte superior, las entradas, y las salidas en la parte inferior. Para ello podemos seleccionar una señal cualquiera con el ratón y arrastrarlo arriba o abajo hasta situarlo en la posición deseada. Las ordenamos para que aparezcan primero x1 y x2, y abajo de todo, la salida f:



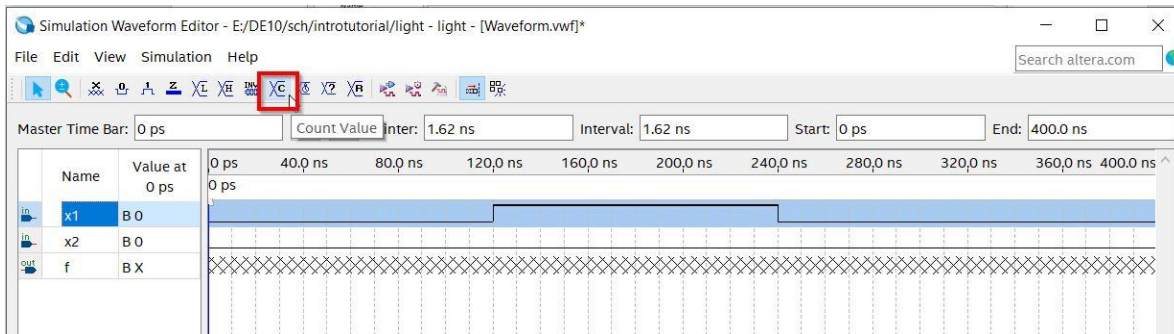
Las formas de onda de las entradas (valores lógicos que queremos que adopten en los diferentes intervalos de tiempo) se pueden editar de varias formas. Una de ellas es seleccionar con el ratón el intervalo de tiempo deseado y forzar los niveles lógicos de ese intervalo con los botones correspondientes del menú de iconos. Por ejemplo, para forzar un “1” lógico de la señal x1 durante el intervalo entre 120ns y 240ns, seleccionamos ese intervalo con el ratón y activamos el botón correspondiente a ese nivel lógico en el menú de iconos:



Como resultado obtenemos:



Para generar de forma rápida todas las combinaciones posibles de las dos señales de entrada podemos proceder de otra manera. Seleccionamos (click con el ratón encima del nombre de la señal) todo el intervalo de tiempo para esa señal, y mediante la opción de “Count Value” del menú de iconos hacemos que el valor de esta señal se vaya incrementando cada cierto intervalo de tiempo deseado. Hagámoslo con x1 y seleccionemos como intervalo de tiempo 200 ns:



Count Value

Radix: Binary

Start value: 0

Increment by: 1

Count type

☒ Binary

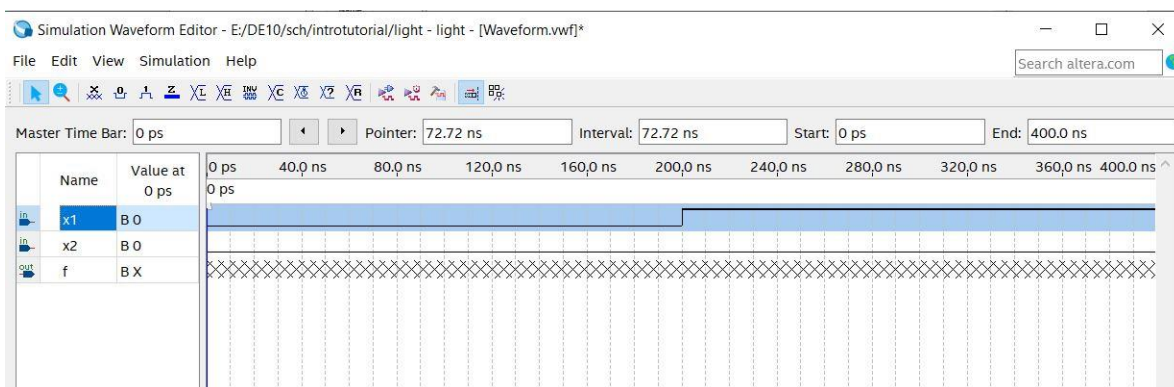
☐ Gray code

Transitions occur

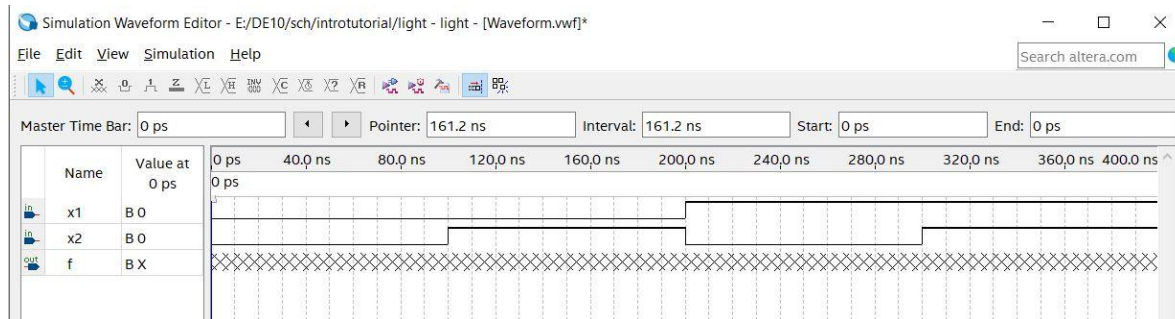
Count every: 200 ns

OK Cancel

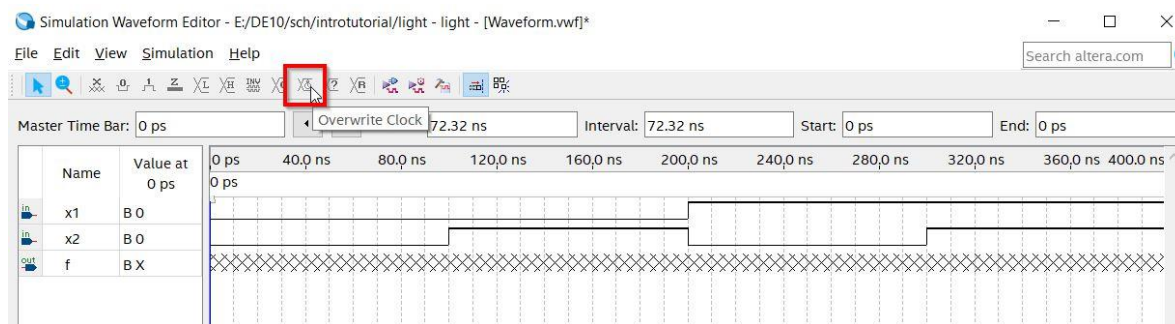
Obtenemos como resultado que la entrada x1 adopta el valor lógico “0” durante los primeros 200ns, y luego se incrementa y pasa a adoptar el valor lógico “1” durante los siguientes 200ns:



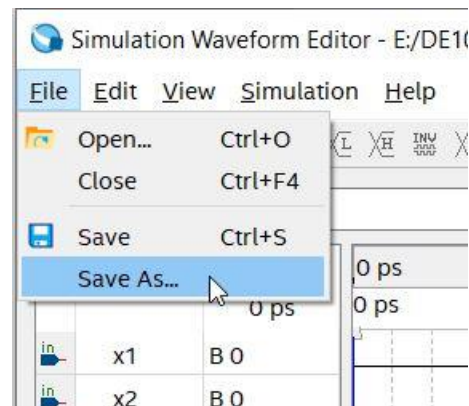
Procedamos de igual forma con x2, pero ahora seleccionando como intervalo de tiempo 100 ns. Co esto conseguiremos que durante el intervalo de 400ns se generen las cuatro combinaciones de dos bits de las señales x1 y x2, “00”, “01”, “10” y “11”, en intervalos de 100ns:

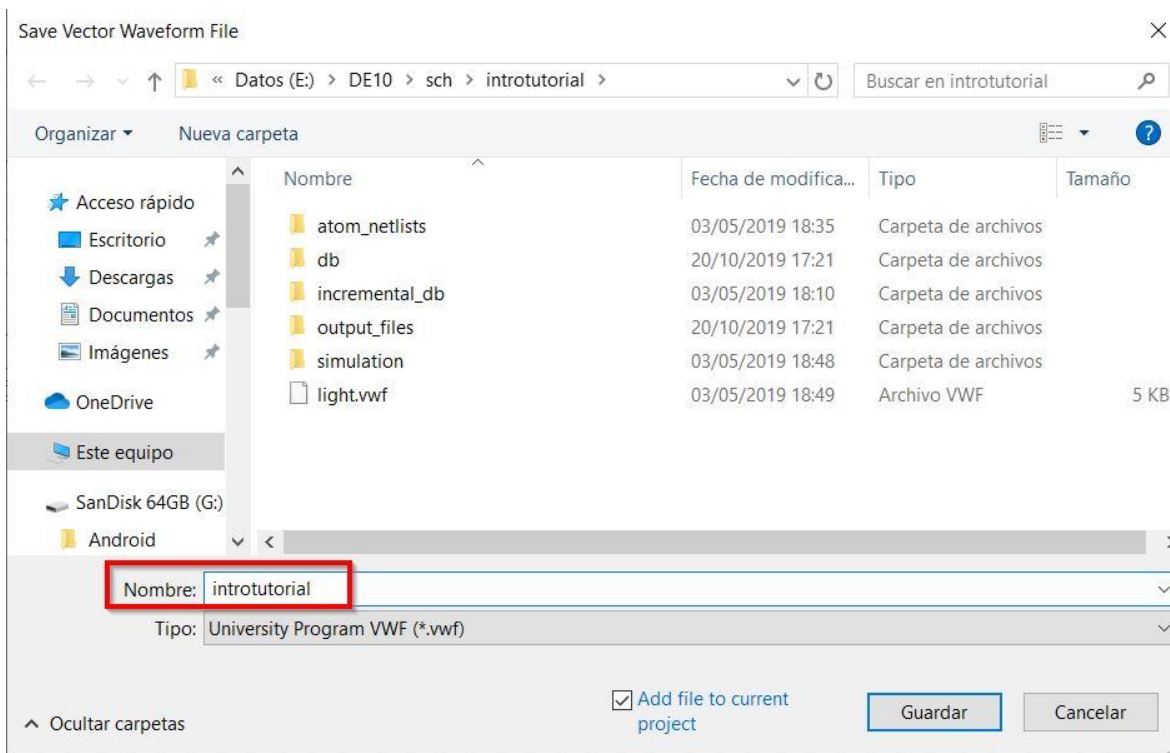


La opción de “Count Value” resulta útil especialmente cuando queremos asignar un valor a varias señales de golpe que forman un bus. Cuando las señales son de un único bit, se puede usar de forma similar la opción del menú de iconos “Overwrite Clock”:



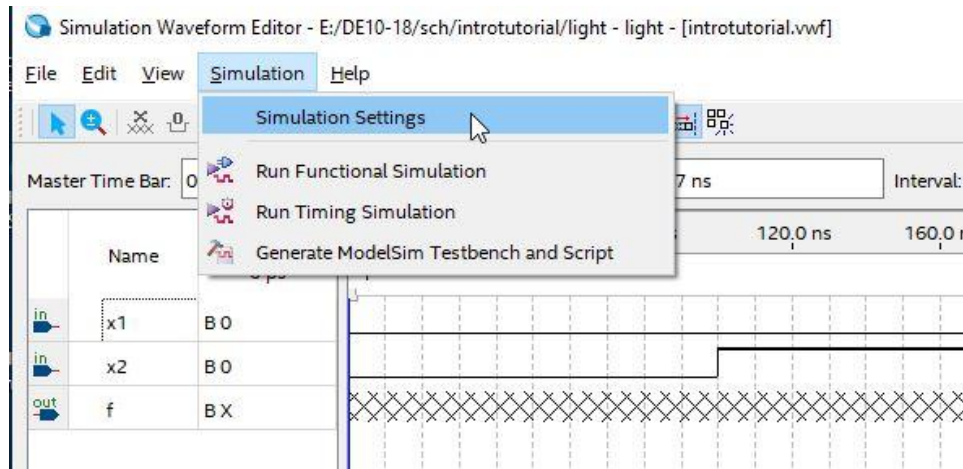
Una vez hemos asignado los valores deseados a nuestras señales de entrada, guardamos nuestro fichero con la opción **File > Save as...**, y le ponemos el nombre deseado (en el ejemplo, *introtutorial*):



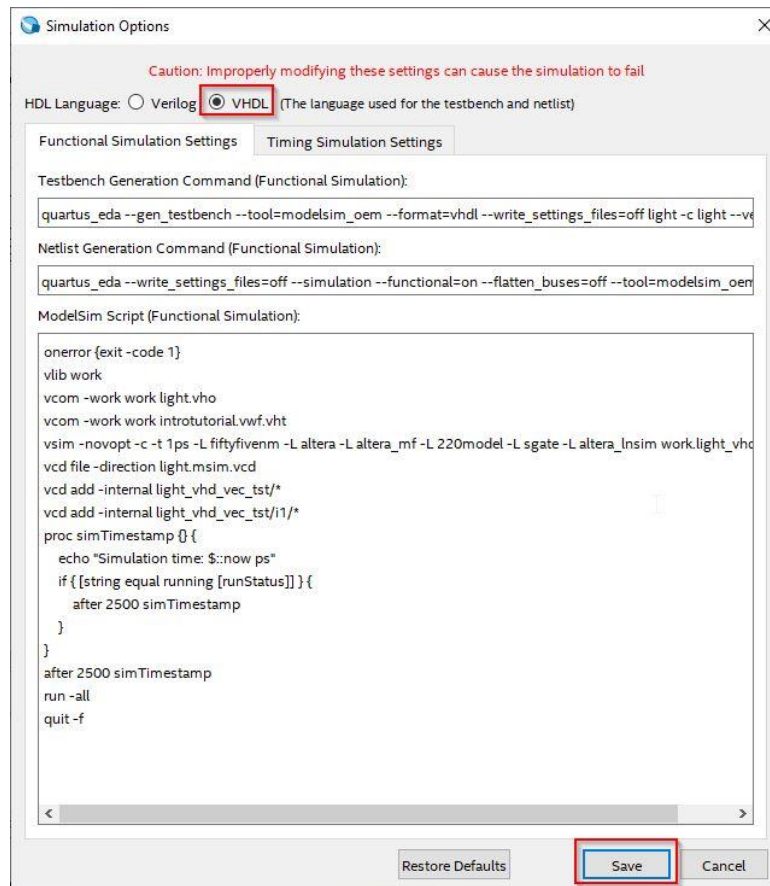


3. Simulación

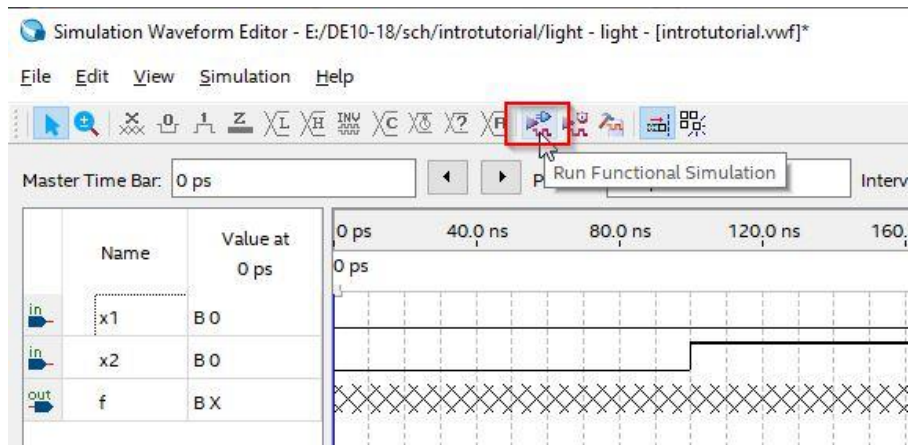
Una vez hemos seleccionado en el editor de formas de onda los valores de las señales que queremos aplicar a las entradas de nuestro circuito podemos proceder a realizar la simulación. Para ello, en primer lugar, en el editor de formas de onda, seleccionamos la opción de menú **Simulation > Simulation Settings**:



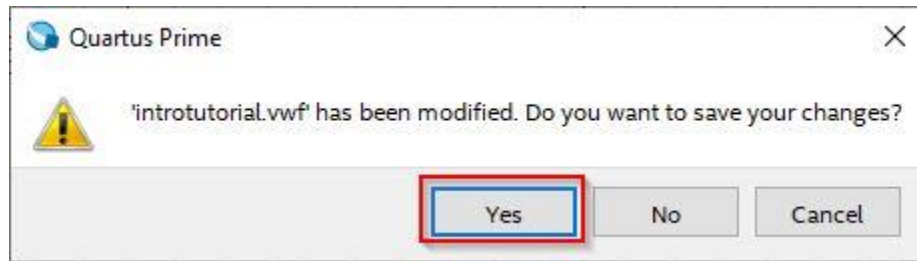
y escogemos la opción **VHDL**, y después **Save**:



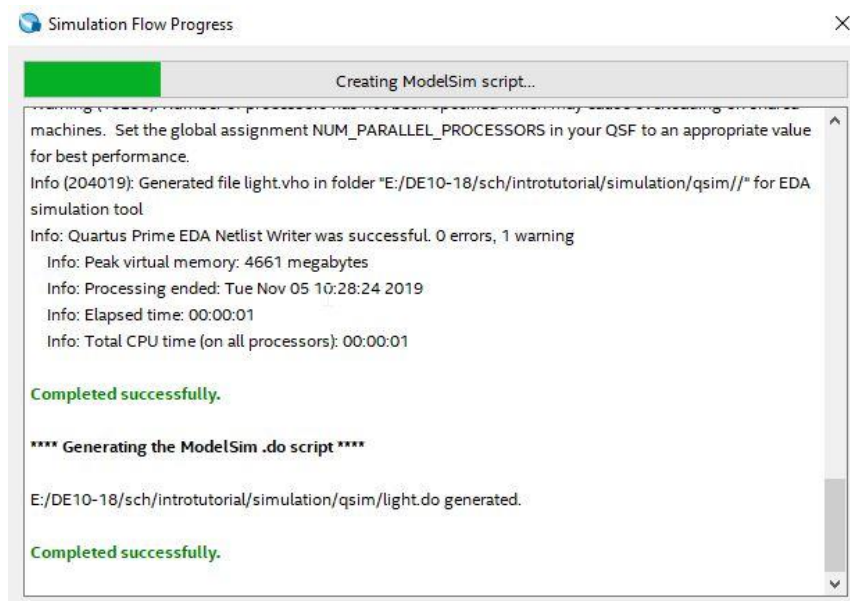
Ahora podemos ejecutar la simulación mediante el menú **Simulation > Run Functional Simulation** o mediante el icono correspondiente de la barra de iconos:



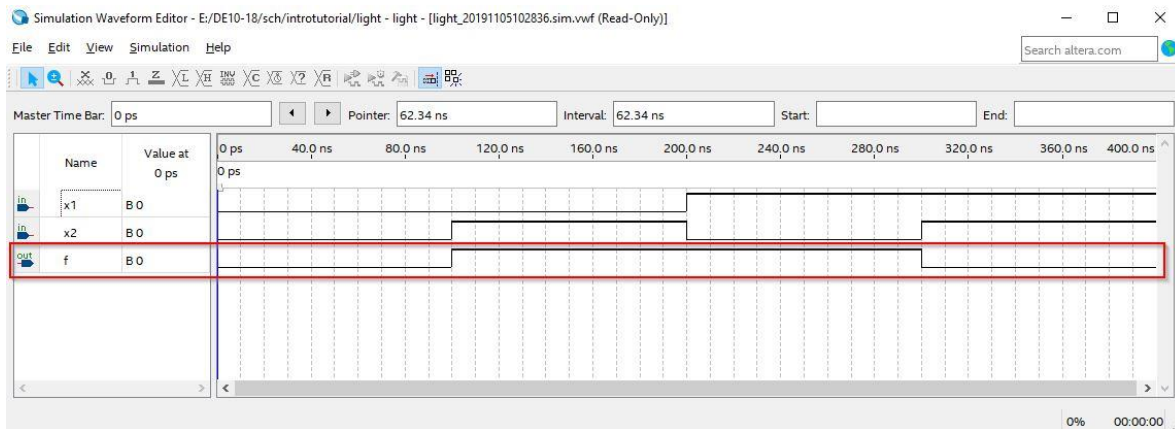
Si nos pregunta si queremos guardar el fichero porque ha cambiado decimos que sí:



Se abrirá una ventana indicando el proceso de la simulación.



Si ésta finaliza sin errores, se cerrará la ventana anterior y se abrirá una nueva similar a la del editor de formas de onda donde podremos observar el valor que ha adoptado la salida para los diferentes valores de las entradas que se han ido sucediendo en función del tiempo.

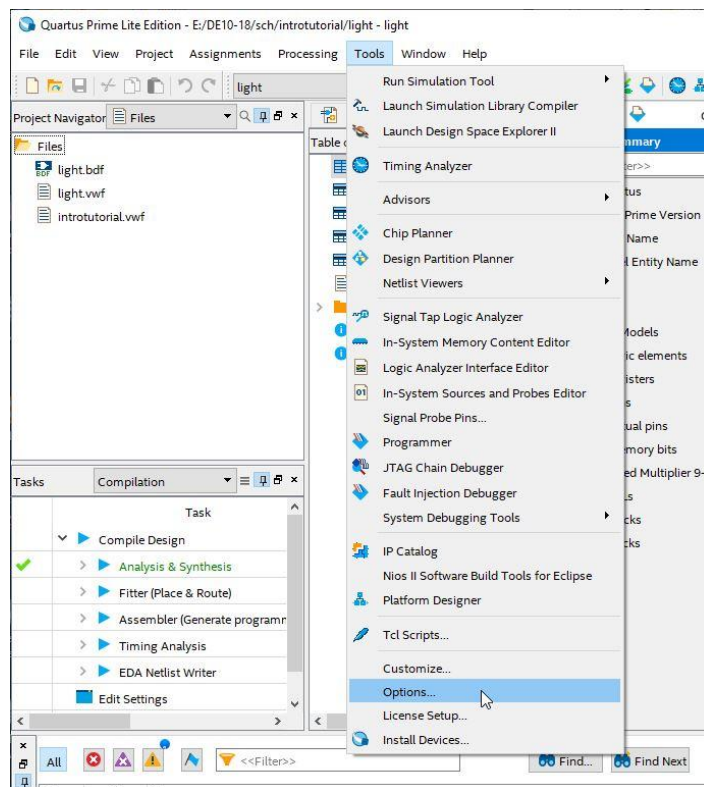


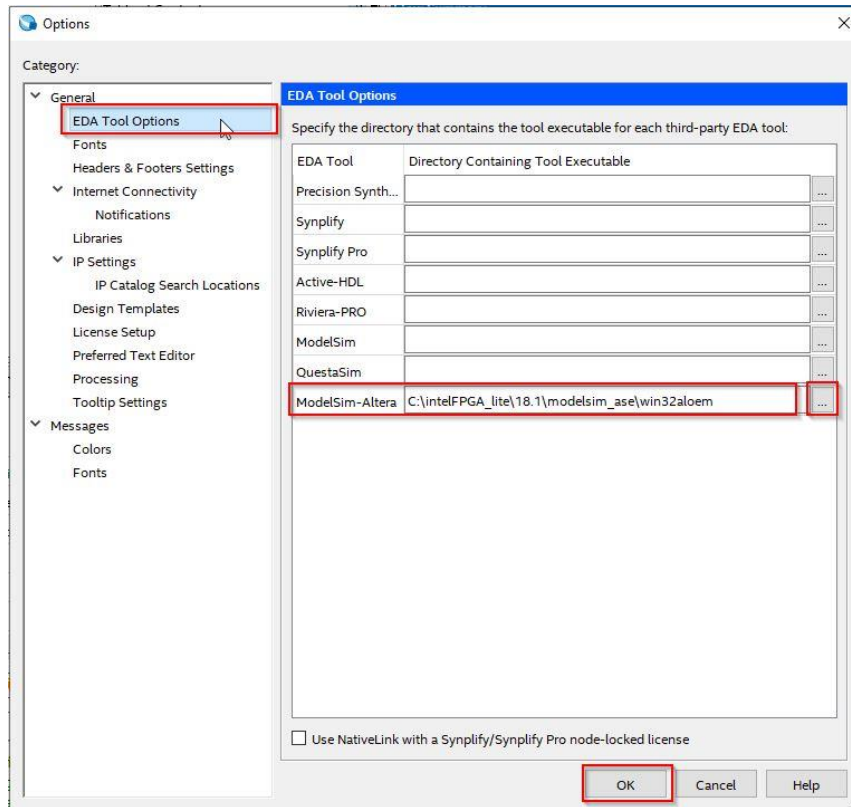
Podemos observar que, cuando la entrada x1 = '0' y la entrada x2 = '0', la salida f adopta el valor lógico '0'; que cuando x1 = '0' y x2 = '1', f = '1', y así sucesivamente podemos comprobar si nuestro circuito funciona como debería o no. En este último caso, deberemos repasar nuestro diseño y el esquema introducido y realizar los cambios oportunos.

4. Errores frecuentes

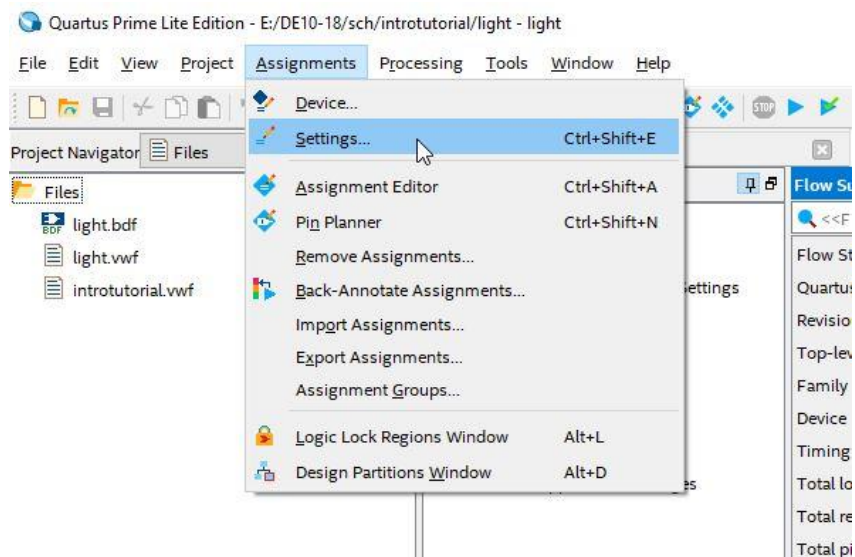
A la hora de simular un diseño, a veces el simulador se detiene informando que se ha producido un error. A continuación, ofrecemos algunos consejos para evitar errores en la simulación:

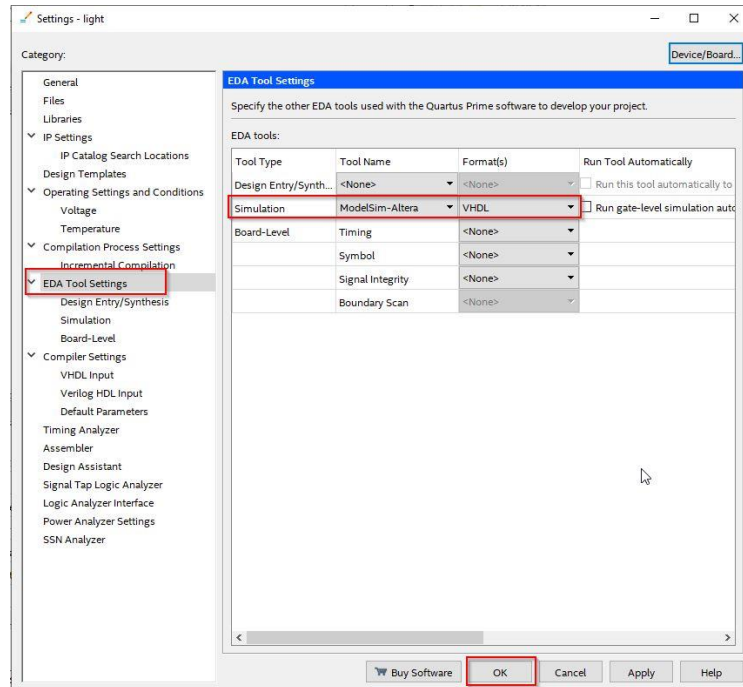
1. Se recomienda que las carpetas donde se realizan los proyectos del Quartus Prime estén cerca de la raíz (*paths* o rutas cortas), que los nombres de las carpetas no contengan espacios ni caracteres raros ni acentos (limitarse a letras, números y *underscore* o guión bajo).
2. El nombre de las entidades del diseño tampoco han de contener espacios ni caracteres raros ni acentos (limitarse a letras, números y *underscore* o guión bajo).
3. Evitar usar como nombres de entidades o de los puertos de entrada/salida de nuestros diseños palabras reservadas del lenguaje VHDL (la simulación que realiza el Quartus se basa en obtener una descripción en lenguaje VHDL de nuestro diseño, y si se usan palabras reservadas del lenguaje puede dar error). Por ejemplo, no llamar a las entradas “in” o “select”, o a las salidas “out”, ya que éstas son palabras reservadas en VHDL (si se produce un error y no se sabe el origen, consultar en algún manual de referencia de VHDL si los nombres de las entidades o señales usados en el diseño son palabras reservadas).
4. Asegurarse de haber indicado en las opciones del Quartus la ruta del ejecutable del ModelSim-Altera: en la ventana principal del Quartus Prime, en el menú **Tools > Options > EDA Tool Options** indicar la ruta del ejecutable (por defecto debería ser “C:\intelFPGA_lite\18.1\modelsim_ase\win32aloem”):





5. Asegurarse de haber indicado en las opciones de simulación del Quartus que se use el ModelSim-Altera y VHDL: en la ventana principal del Quartus Prime, en el menú **Assignments > Settings > EDA Tool Settings > Simulation** escoger “ModelSim-Altera” y “VHDL”:





6. A veces, después de realizar una simulación correcta, si se modifica el nombre del fichero de formas de onda o se crea uno nuevo, puede producirse un error porque el Quartus aún recuerda el nombre del anterior fichero usado. En este caso, probar a resetear los valores por defecto de simulación: en el editor de formas de onda seleccionamos la opción de menú **Simulation > Simulation Settings** y hacemos click sobre **Restore Defaults**:

