

Laboratorio de sistemas en Tiempo Real

MANUAL PARA INSTALACION DE DRIVER USB-BLASTER Y CREACION DE PROYECTOS EN DE2i-150

Prestador es de Servicio Social:

RODRIGO BARRAGÁN AYALA, 210224772, E-mail: yoniein@hotmail.com

Cynthia Oliva Hernández, E-mail: cbabyo@hotmail.com

Profesores:

José Juan Meza Espinosa, E-mail: jose.meza@red.cucei.udg.mx

Salomón Eduardo Ibarra Chávez, E-mail: salomon.ibarra@red.cucei.udg.mx

INDICE

1 INSTALACION DE SOFTWARE -----	2
2 INSTALAR EL DRIVER USB-BLASTER -----	3
3 INICIAR UN PROYECTO CON QUARTUS II -----	10
4 ENTRADA DE DISEÑO -----	14
5 COMPILA Y VERIFICA TU DISEÑO -----	37

1 INSTALACION DE SOFTWARE

1.1 Introducción

Esta sección explica como instalar el siguiente software:

- Altera Complete Design Suit
- DE2i-150 Development Board Contents

1.2 Instalación de Altera

Altera provee las herramientas necesarias para desarrollar hardware y software para Altera FPGAs. El software Quartus II será la herramienta principal para crear proyectos de sistemas embebidos, el cual se encuentra en un DVD incluido en la caja. También es posible descargar esta aplicación en la página web de altera:

www.altera.com/download

El kit contiene el CD Quartus II con una edición de suscripción y un editor web. La edición web de Quartus II no requiere licencia.



1.3 Instalando el contenido del DE2i-150

Para instalar los componentes necesarios para desarrollar en la DE2i-150, es necesario copiar los contenidos de la carpeta (DE2i-150_SYSTEM_CD) localizada en el DE2i-150 system CD a tu computadora.

Contenido de al DE2i-150 system CD:

<i>Directory Name</i>	<i>Description of Contents</i>
User Manual	Contains the DE2i-150 documentation
Demonstrations	Contains design examples of DE2i-150
Datasheet	Contains the datasheets of the components on DE2i-150
Schematic	Contains the schematic of DE2i-150
Tools	Contains the design and testing tools for DE2i-150

2 INSTALAR EL DRIVER USB-BLASTER

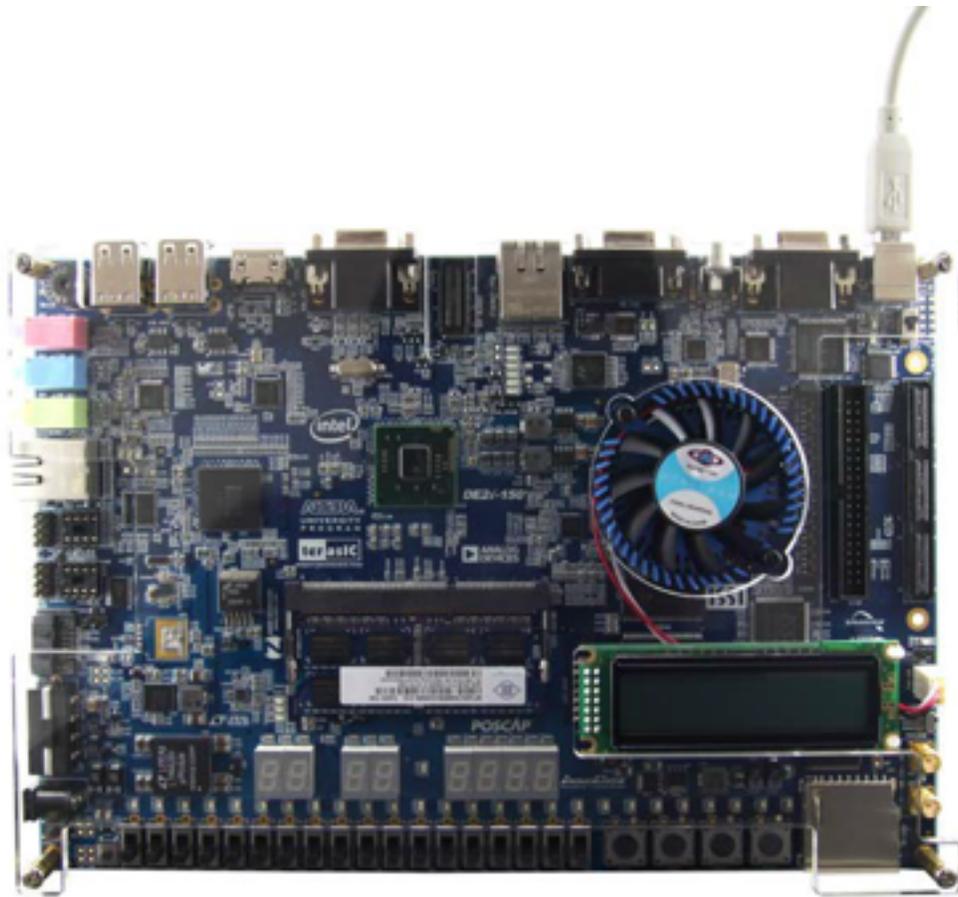
2.1 Introducción

La tarjeta de desarrollo incluye un circuito USB blaster para programación FPGA. Para poder comunicar la computadora huésped y la tarjeta de desarrollo se requiere instalar de drive USB-Blaster en la computadora huésped. Antes de comenzar la instalación, se debe verificar den se encuentra el driver con el directorio: \<Quartus II installation directory>\drivers\usb-blaster. Si el driver USB-Blaster no se encuentra, es necesario que verifique la correcta instalación del software Quartus II.

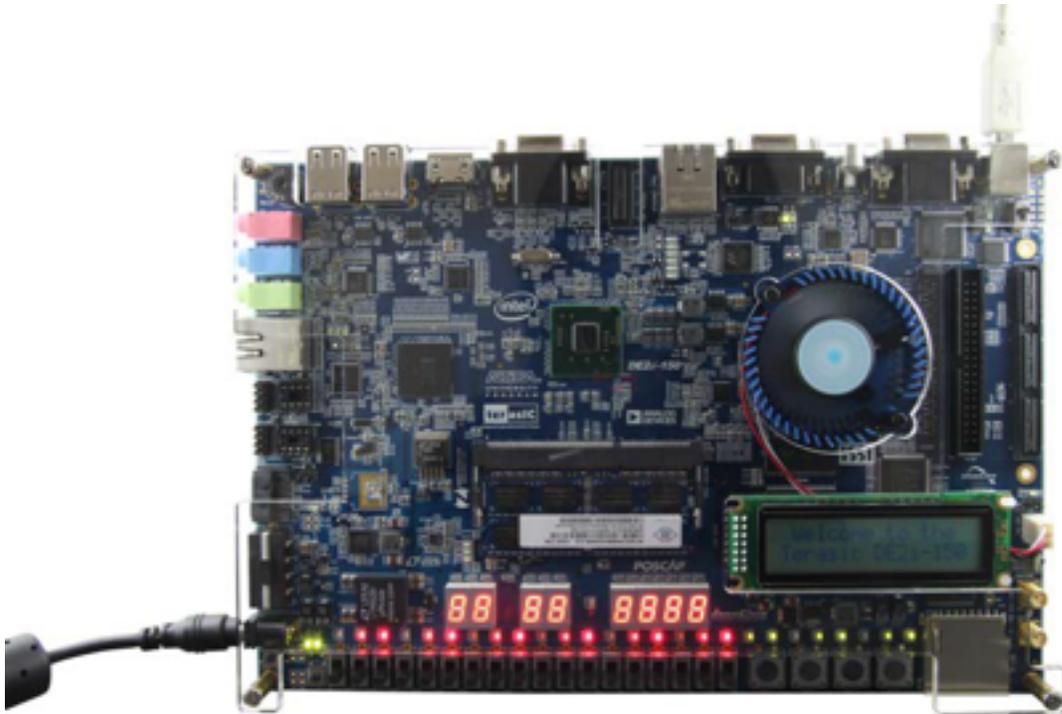
2.2 Preparando el hardware

Esta sección describe la instalación del driver USB-Blaster.

1. Conecte la computadora huésped a la tarjeta mediante el cable USB en el conector USB (J9) del DE2i_150



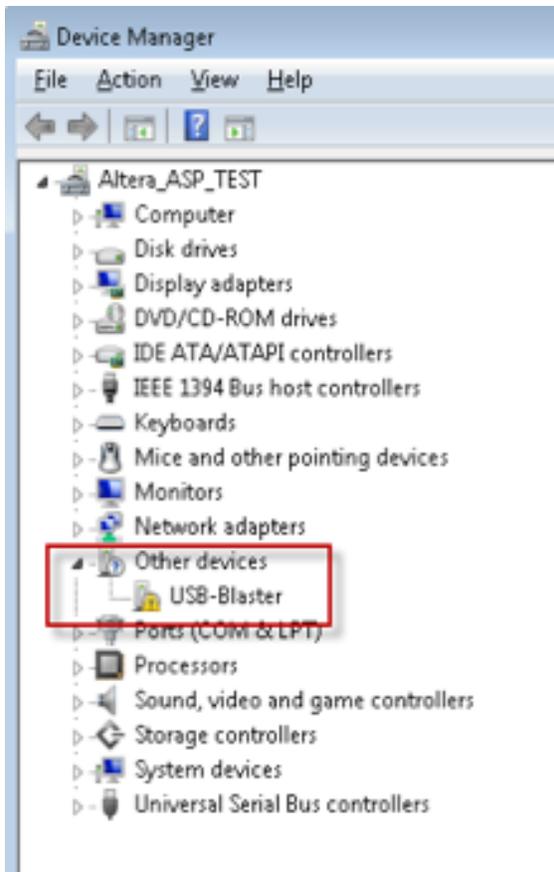
2. Conecte a la corriente la tarjeta DE2i-150.



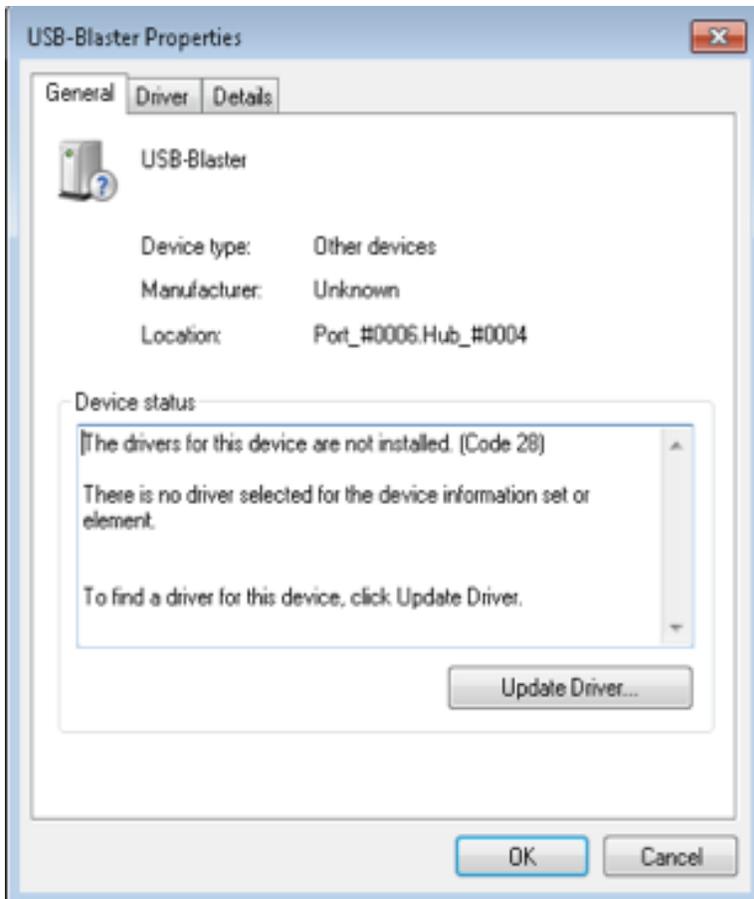
2.3 Instalando el USB-Blaster Driver para Windows 7

1. Nuevo hardware encontrado.

Una vez que la tarjeta esté conectada a la corriente y encendida, el sistema operativo debería reconocer un nuevo hardware, pero será incapaz de proceder si no tiene el driver instalado. La tarjeta DE2i_150 es programada al usar el mecanismo USB_Blaster de Altera. Instala el driver dirigiéndote al Device Manager y da doble-click en USB-Blaster bajo OTHER devices.

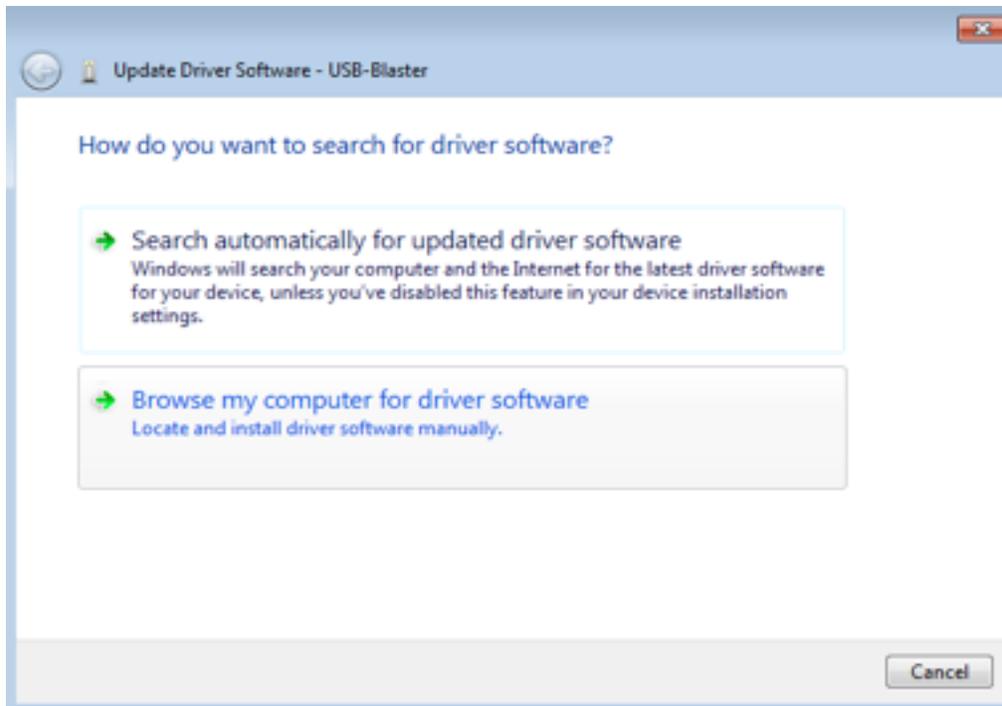


2. En la ventana de propiedades de USB-Blaster, da click en Update Driver.



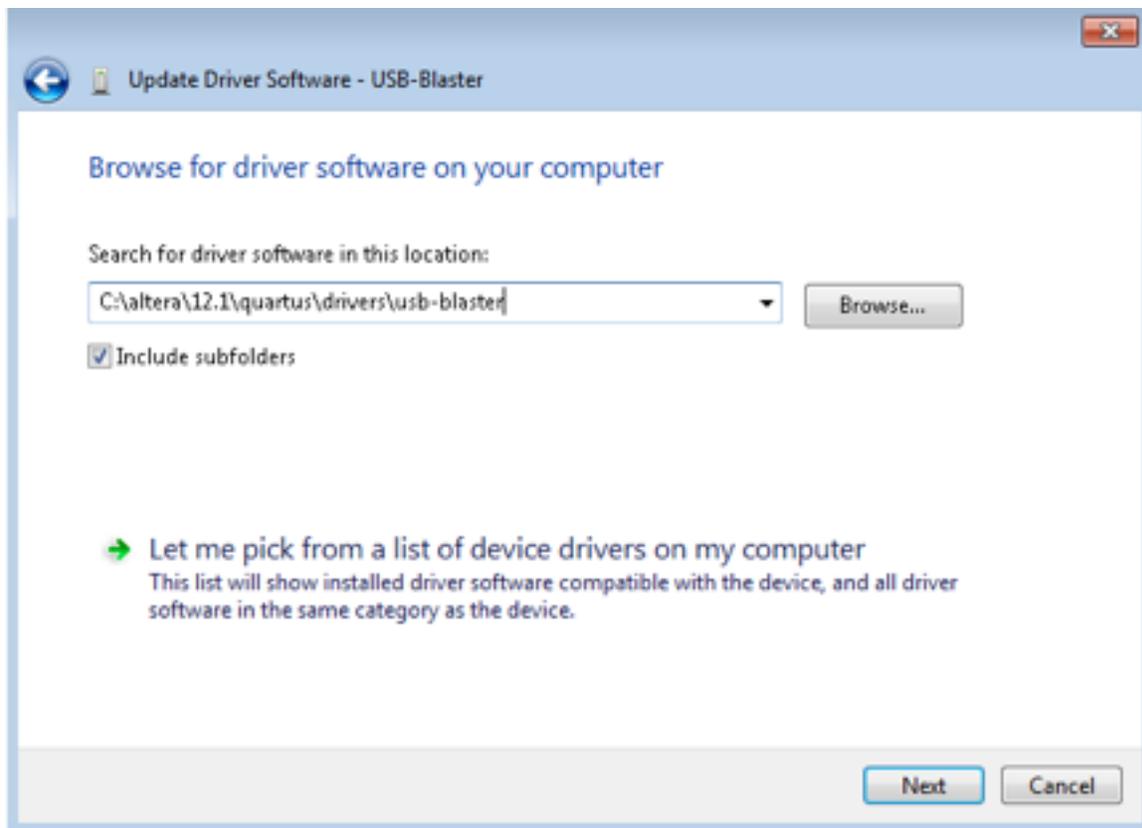
3. Especifica la ruta para el driver USB-Blaster:

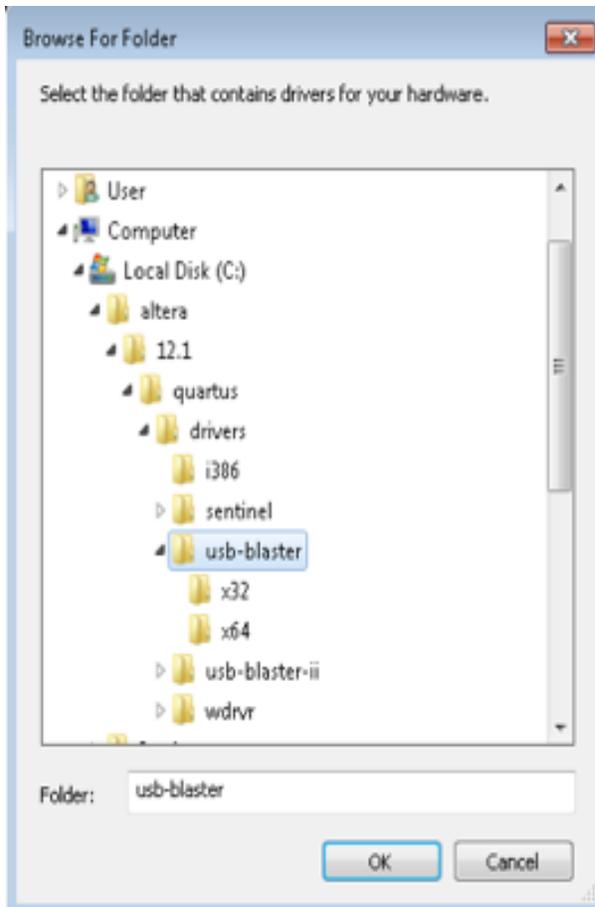
En la ventana Update Driver Software window, click Browse my computer for device software



4. Selección la versión apropiada para el driver USB-Blaster:

Encuentra el driver apropiado localizado en \<Quartus II system directory>\drivers\usb-blaster. Click OK y desde la ventana de retorno. Click Next para instalar el driver.

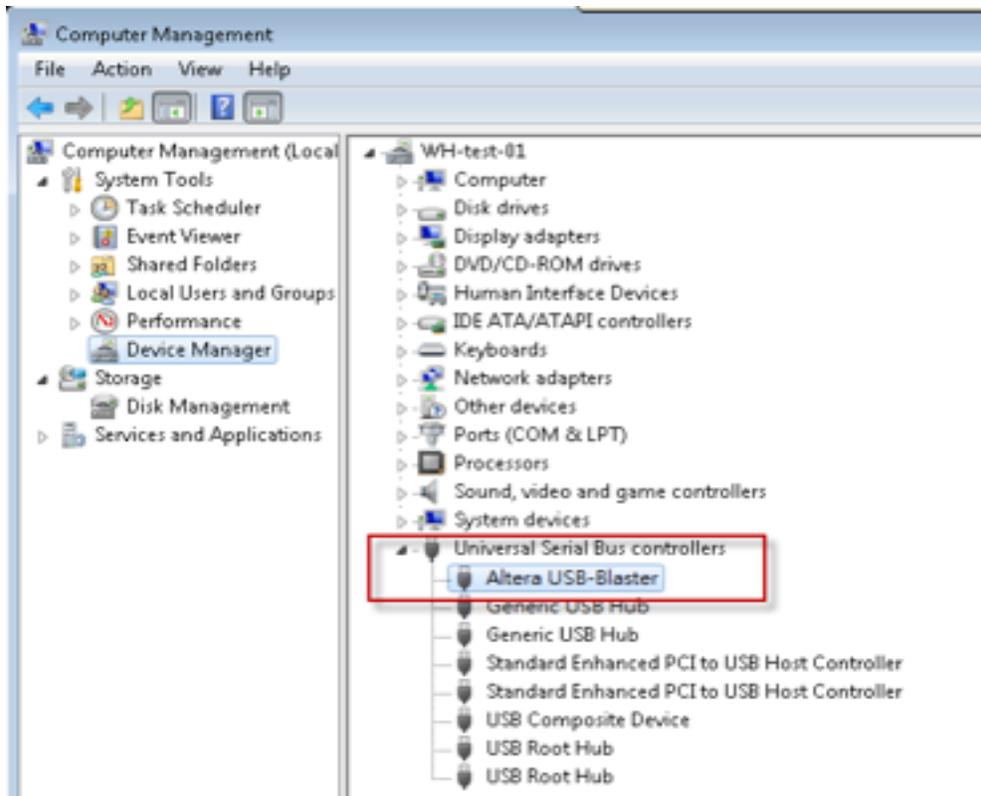




5. El driver USB blaster está listo para usarse.

El driver ya está instalado. Click en Finish para comenzar a usar el DE2i-150.

6. Para confirmar la instalación, busca el Altera USB-Blaster bajo el Universal Serial Bus controllers en el Device Manager.

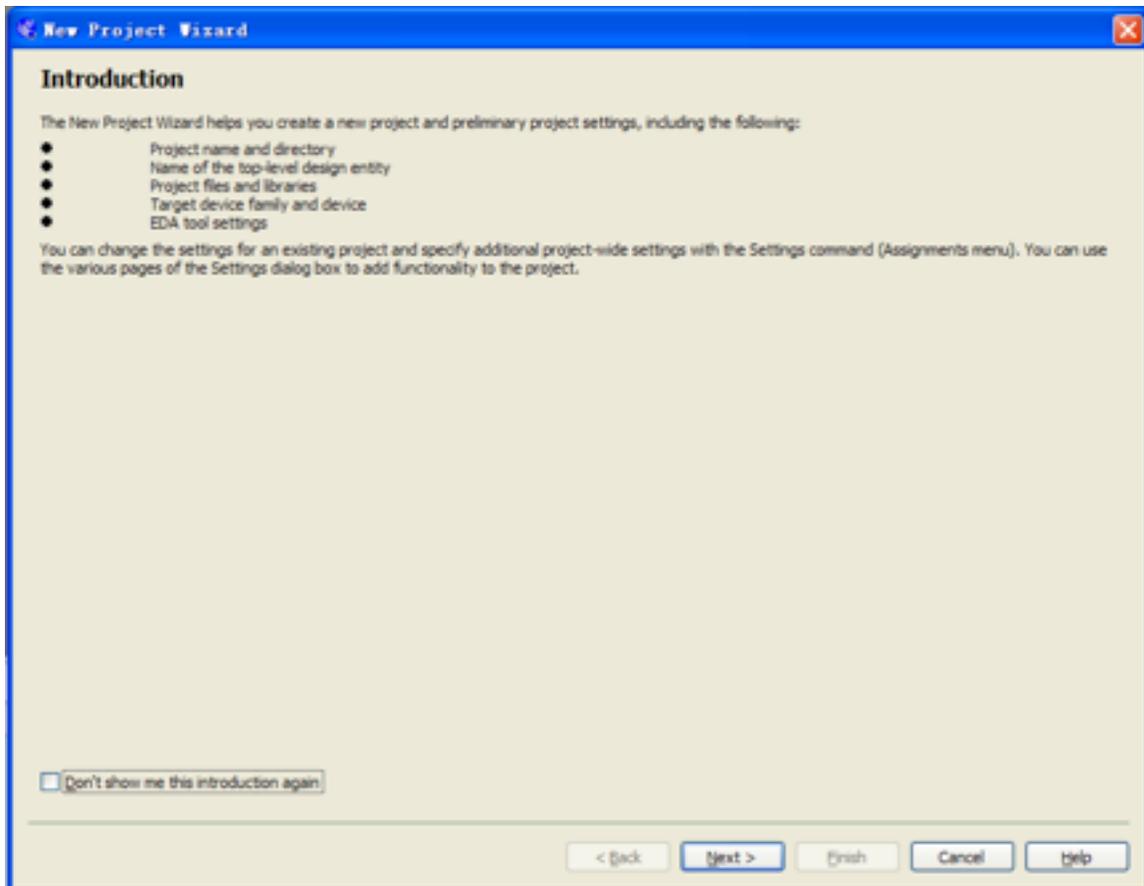


3 INICIAR UN PROYECTO CON QUARTUS II

Lo primero que hay que hacer es iniciar un nuevo proyecto en Quartus II. Un proyecto es un conjunto de archivos que mantienen la información del diseño FPGA. El Quartus II crea dos tipos de archivos los (.qsf) y los (.qpf).

3.1 Asignando el dispositivo

1. En el software Quartus II, select File > New Project Wizard. Se abre la página de introducción.



2. Click Next.

3. Ingresa la siguiente información acerca de tu proyecto:

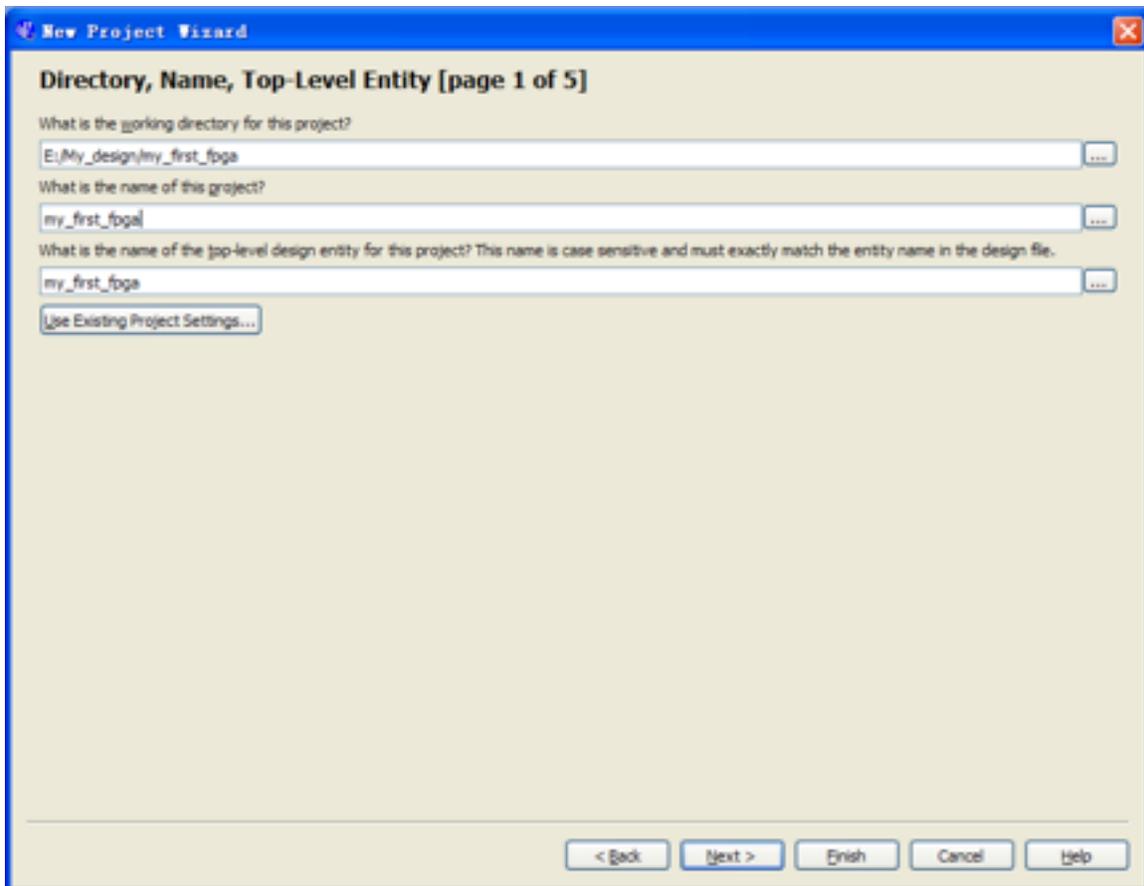
a. What is the working directory for this project? Ingresa un directorio en el cual vas a ingresar los archivos de tu proyecto del Quartus II para este diseño.

b. Por ejemplo, E:\My_design\my_fpga.

c. Los nombres de archivos, nombres de proyectos y directorios del Quartus II no pueden tener espacios.

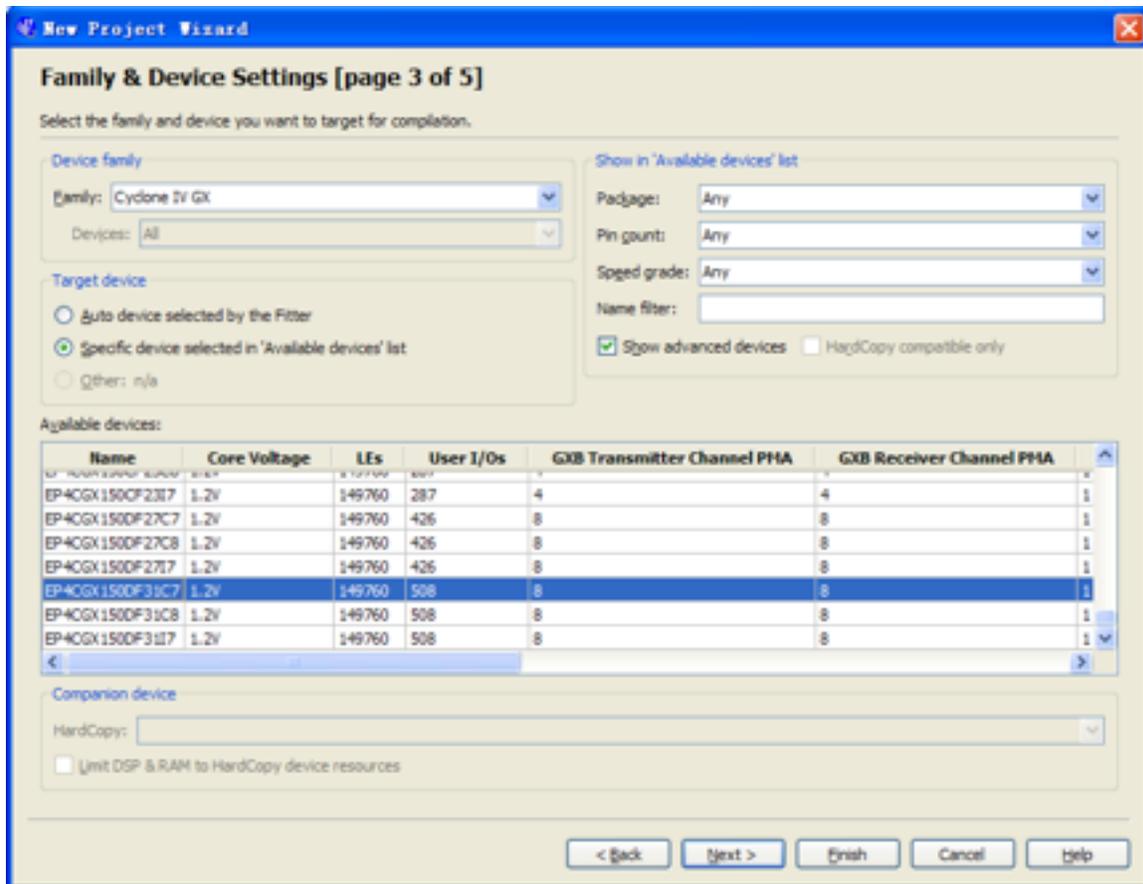
d. What is the name of this project? ingresa my_first_fpga.

e. What is the name of the top-level design entry for this project? Ingresa my_first_fpga.



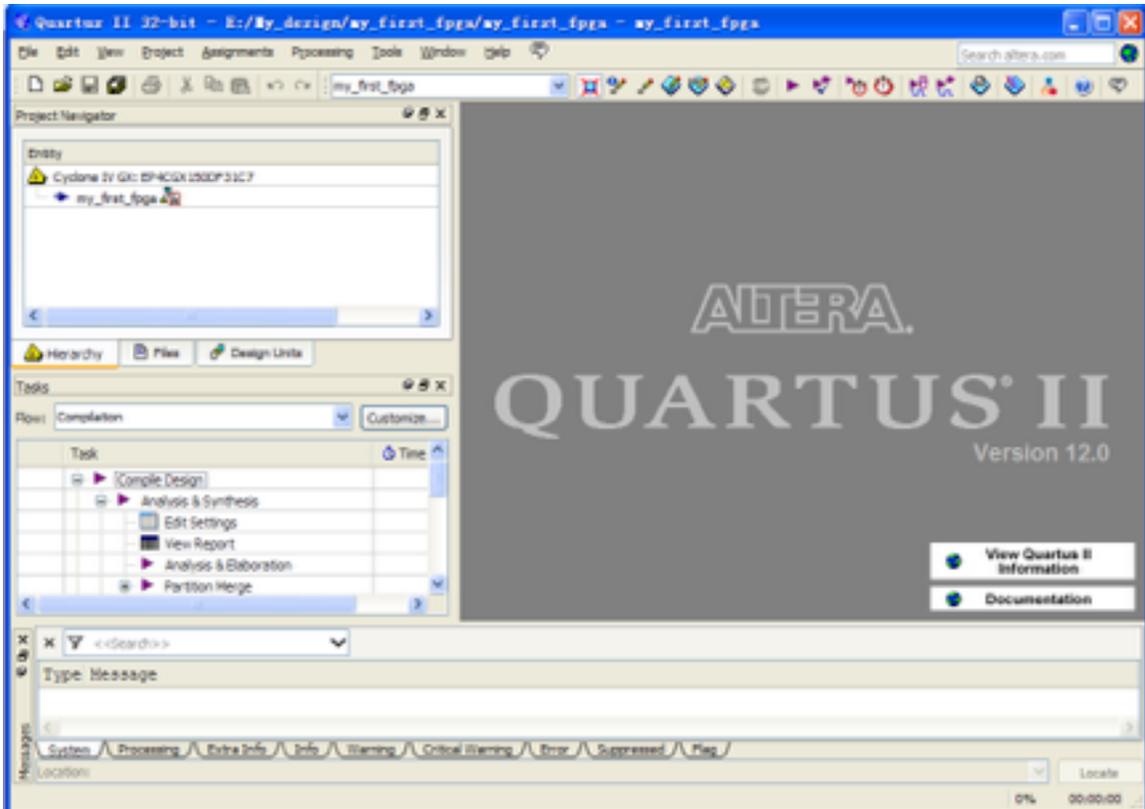
f. Click Next.

g. vas a asignar un dispositivo FPGA específico al diseño y asignar pines.



h. Click Finish.

4. Escoge si para crear el directorio del proyecto my_first_fpga. Acabas de crear tu primer proyecto FPGA en Quartus II.



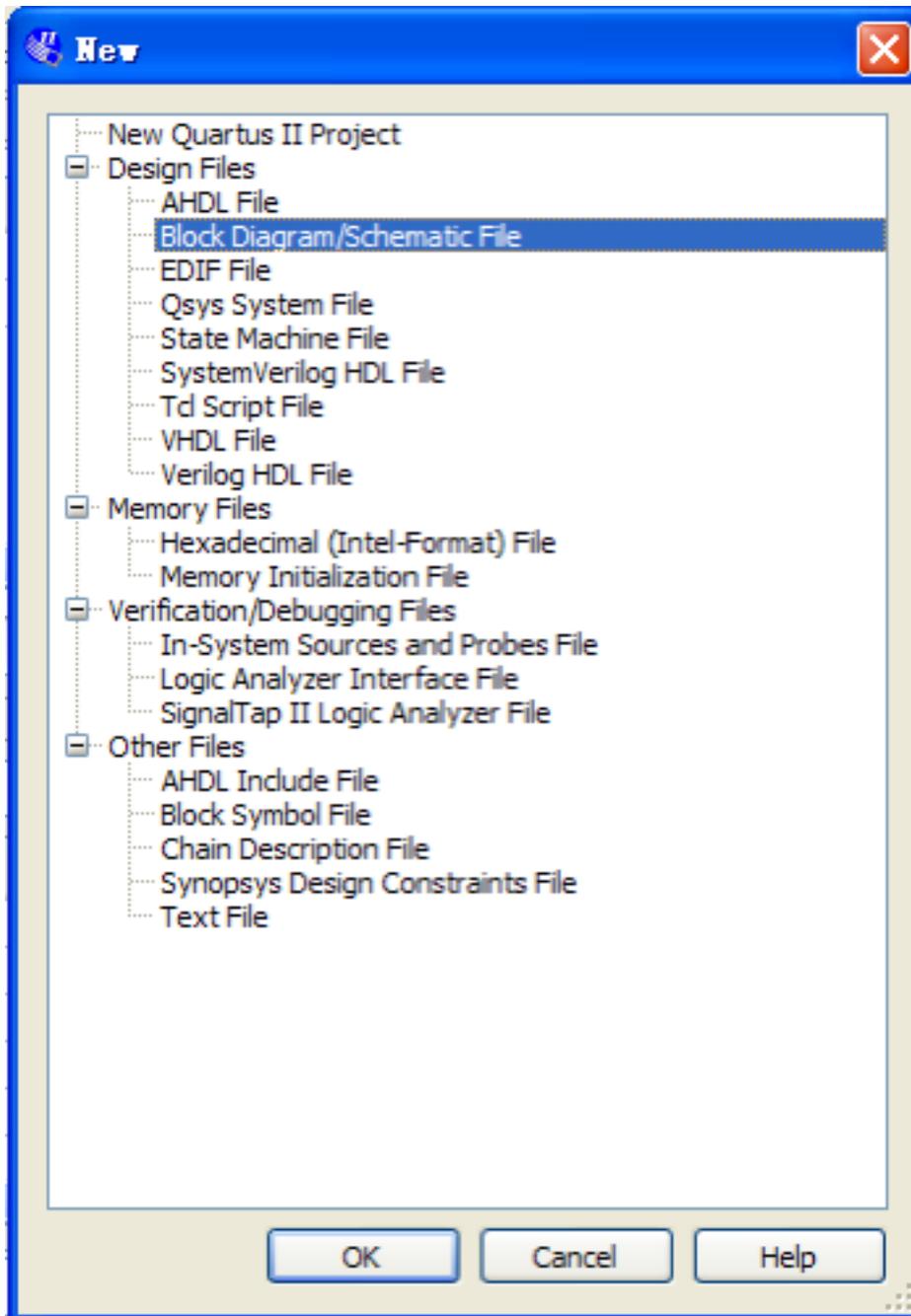
4 ENTRADA DE DISEÑO

4.1 Agregar una Mega Función PLL

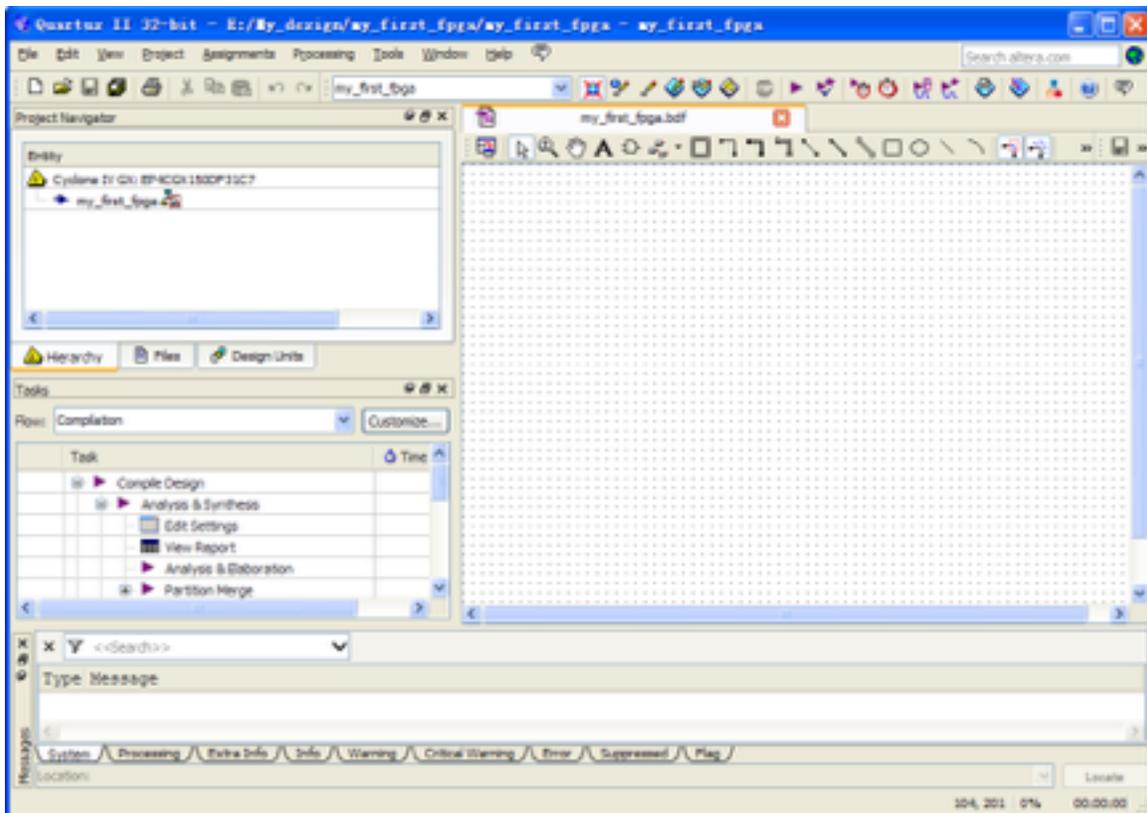
Esta sección describe como agregar una mega función PLL.

En el diseño de entrada creas un esquema o archivo de bloque de diseño (.bdf) este es el diseño top-level. Agregaras funciones de librerías de módulos parametizados (LPM) y asaras código Verilog HDL para agregar un bloque lógico.

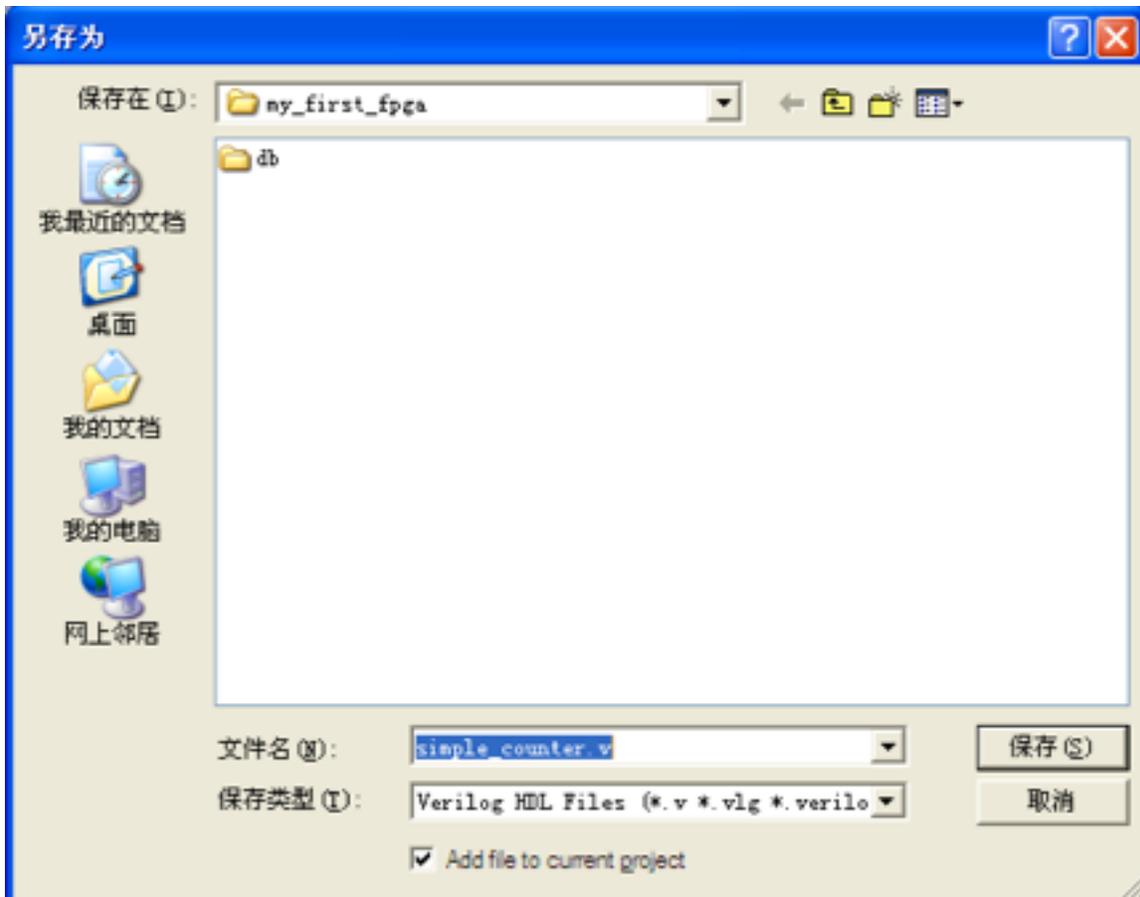
1. Escoge File > New > Block Diagram/Schematic File para crear un nuevo archivo Block1.bdf, el cual guardaras como diseño top-level.



2. Click OK.
3. Escoge File > Save se guarda tal como indica la información siguiente.
 - * File name: my_first_fpga
 - * Save as type: Block Diagram/Schematic File (*.bdf)
4. Click Save. El nuevo archivo de diseño aparece en el bloque editor.



5. Agrega código HDL al blank block diagram al escoger File > New > Verilog HDL File.
6. Click OK para crear un nuevo archivo Verilog1.v, el cual guardaras como simple_counter.v.
7. Selecciona File > Save como se indica a continuación:
 - * File name: simple_counter.v
 - * Save as type: Verilog HDL File (*.v, *.vlg, *.verilog)

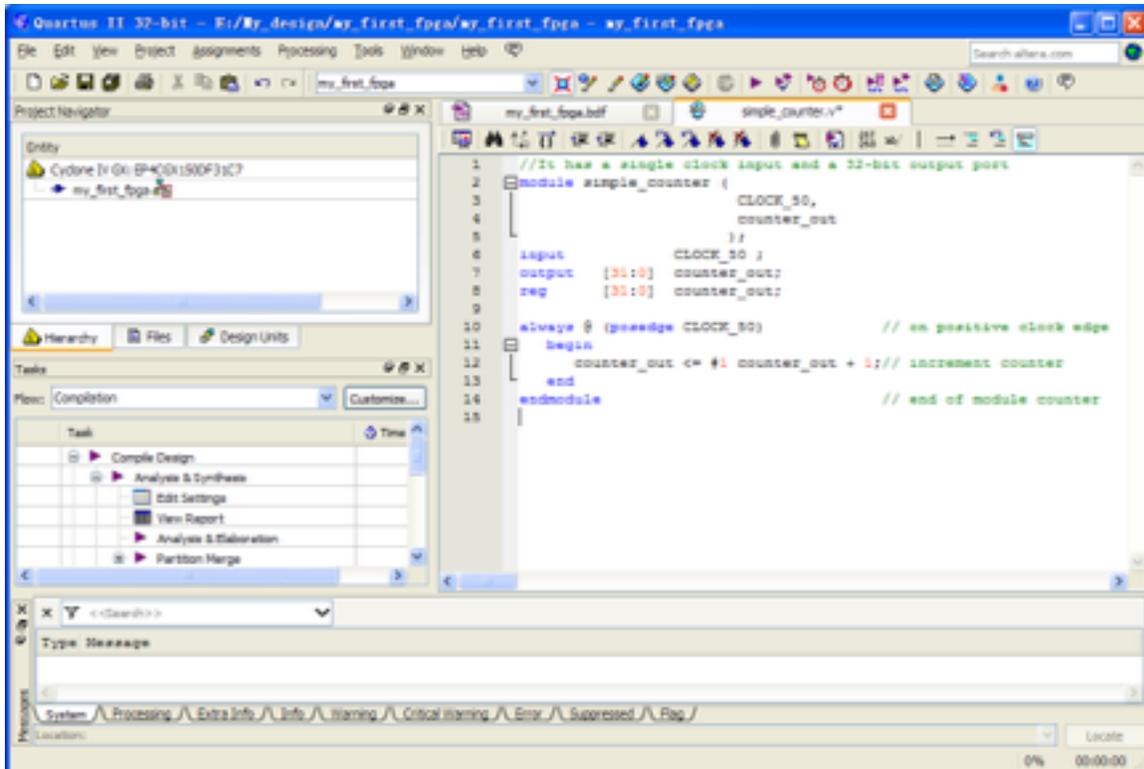


El archivo vacío resultante está listo para que le sea ingresado el código Verilog HDL.

8. Teclea el siguiente código en Verilog HDL en el archivo vacío simple_counter.v

```
//It has a single clock input and a 32-bit output port
module simple_counter (
  CLOCK_50,
  counter_out
);
input CLOCK_50 ;
output [31:0] counter_out;
reg [31:0] counter_out;

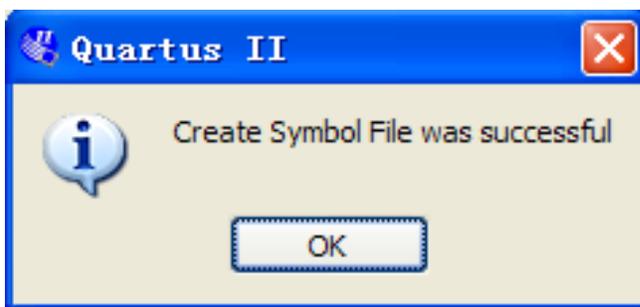
always @ (posedge CLOCK_50) // on positive clock edge
begin
counter_out <= #1 counter_out + 1;// increment counter
end
endmodule // end of module counter
```



9. Guarda el archivo al escoger File > Save, precionando Ctrl + s, o al dar click en el icono del floppy disk.

10. Escoge File > Create/Update > Create Symbol Files for Current File to convert the simple_counter.v file to a Symbol File (.sym). Veras este archivo símbolo para agregar el código HDL a tu BDF schematic.

El software Quartus crea un archivo símbolo y despliega un mensaje.



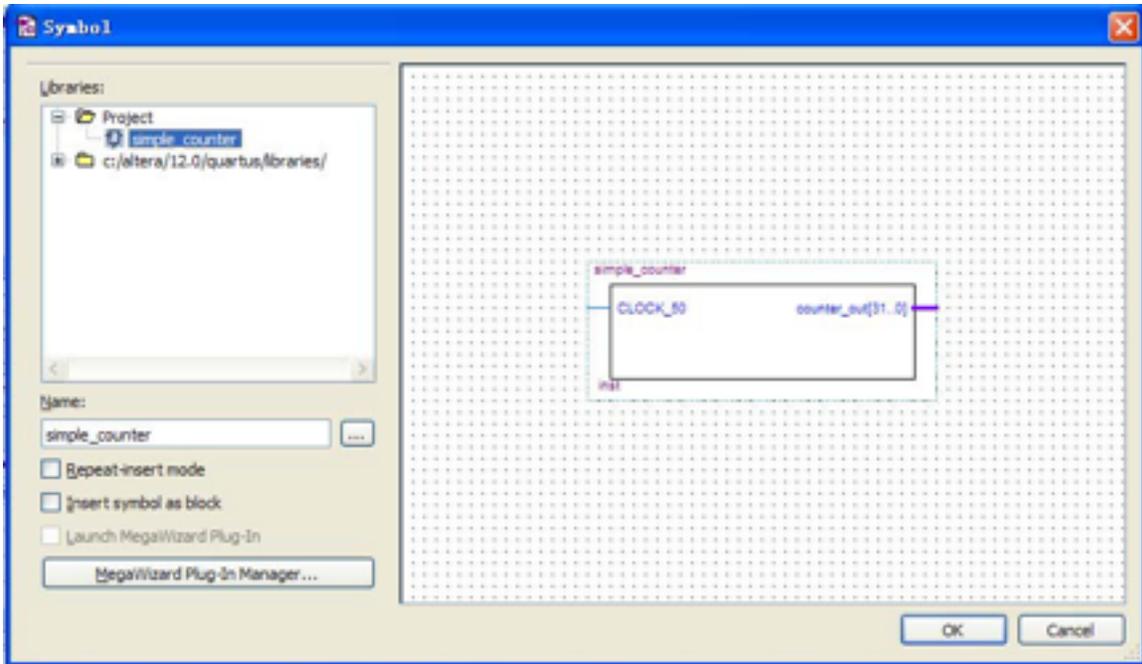
11. Click OK.

12. Para agregar el símbolo simple_counter.v al diseño top-level, click el my_first_fpga tab.

13. Escoge Edit > Insert Symbol.

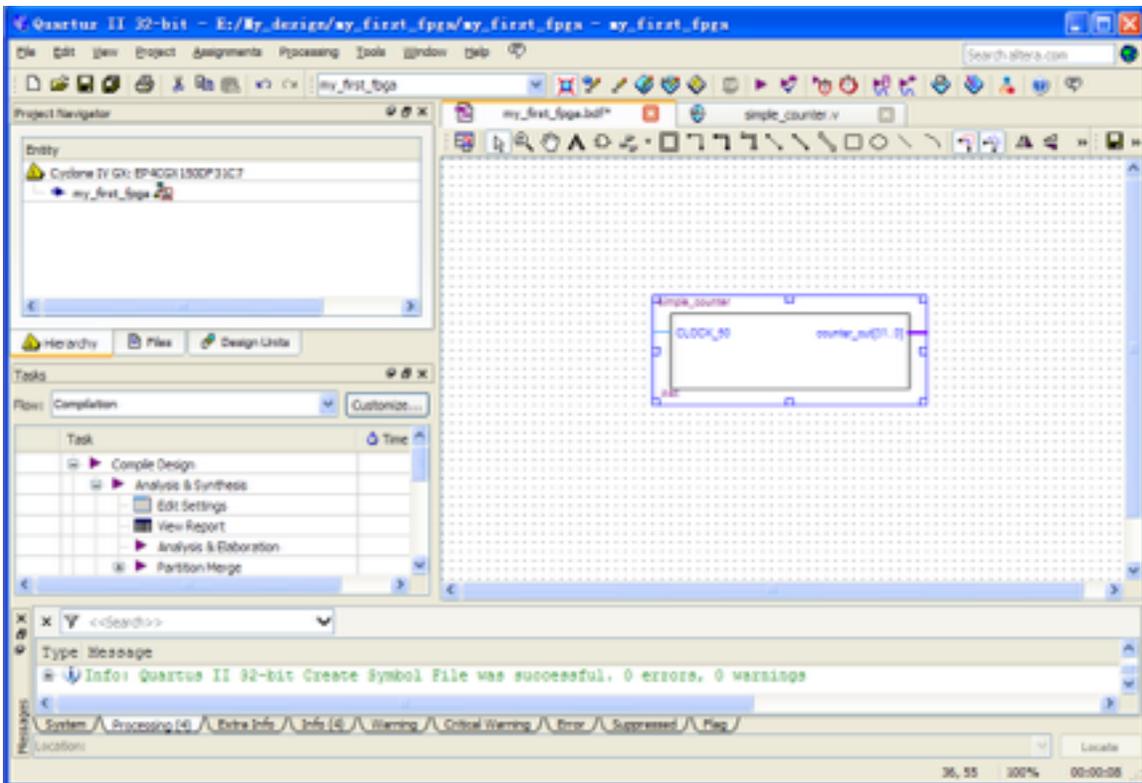
14. Double-click el directorio del proyecto para expandirlo.

Incluso puedes dar doble_click en un área vacía del BDF para abrir el Symbol dialog box



16. Click OK.

17. Mueve el cursor hacia el BDF grid; la imagen símbolo se mueve con el cursor. Click para colocar el símbolo simple_counter en el BDF. Puedes mover el bloque después de colocarlo al dar un simple click y arrastrarlo hacia donde tú quieras al soltar el botón del mouse.



18. Presiona la tecla Esc o click en un lugar vacío en el esquema grid para cancelar.

19. Guarda regularmente tu proyecto.

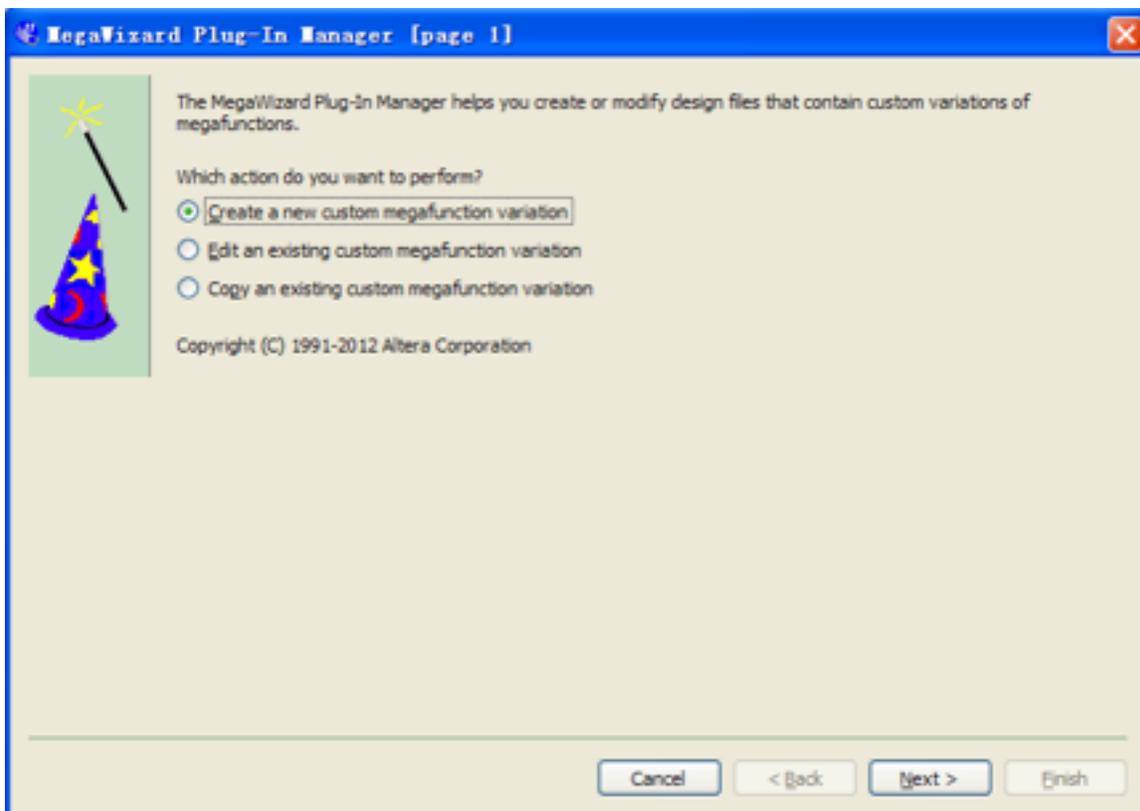
Usando Quartus para agregar un PLL Mega Función

Mega Funciones, tal y como las disponibles en el LPM, son módulos pre-diseñados que puedes usar en el diseño FPGA. Estas mega funciones Altera son optimizadas por velocidad, área y familia de dispositivos.

Puedes agregar eficiencia al usar una mega función en lugar de escribir la función tú mismo. Altera incluso provee funciones más complejas, llamadas funciones MegaCore, las cuales puedes adquirirlas gratis pero requiere una licencia para usar en el diseño de algún producto. Este diseño tutorial usa un reloj PLL para dirigir un contador simple. Un PLL usa el on-board oscillator para crear una frecuencia de reloj constante como ingreso al contador. Para crear el reloj, vas a agregar una mega función LPM pre construida llamada ALTPLL.

1. Escoge Edit > Insert Symbol o click Add Symbol on the toolbar---

2. Click Megawizard plug-in Manager. The MegaWizard Plug-in Manager aparece.

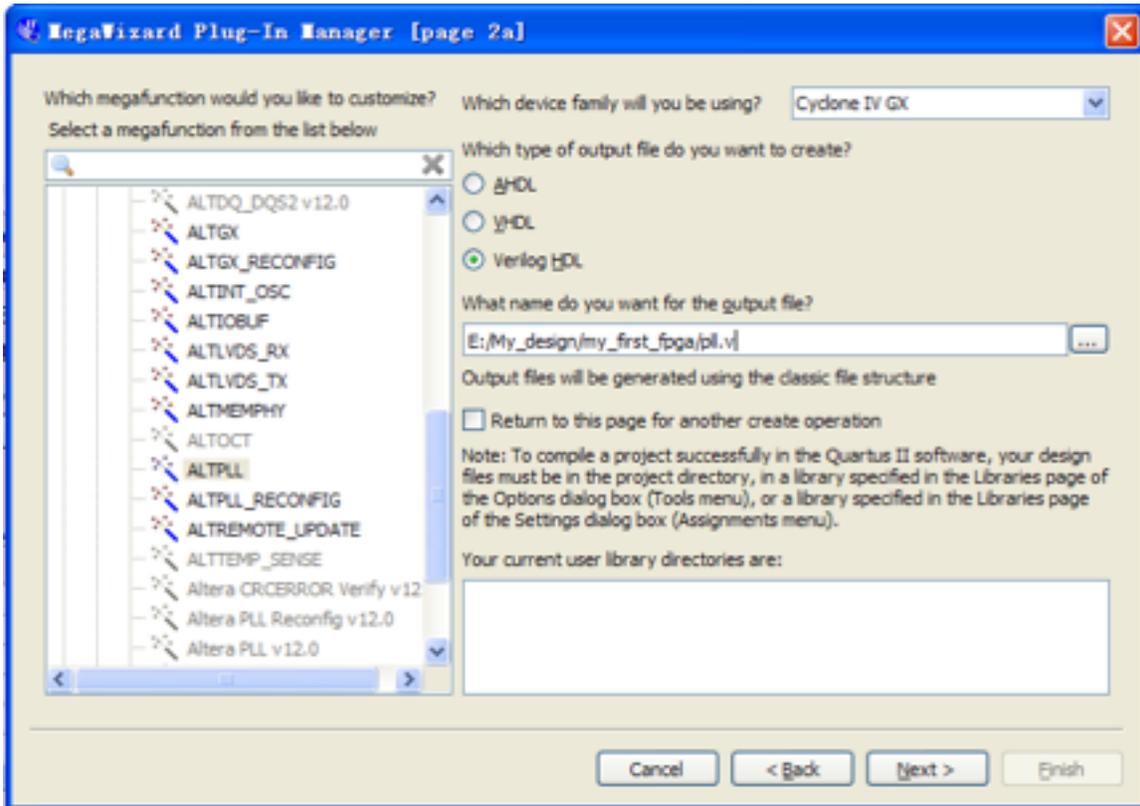


3. Click Next.

4. En MegaWizard Plug-in Manager [pagina 2a], especifica las siguientes secciones.

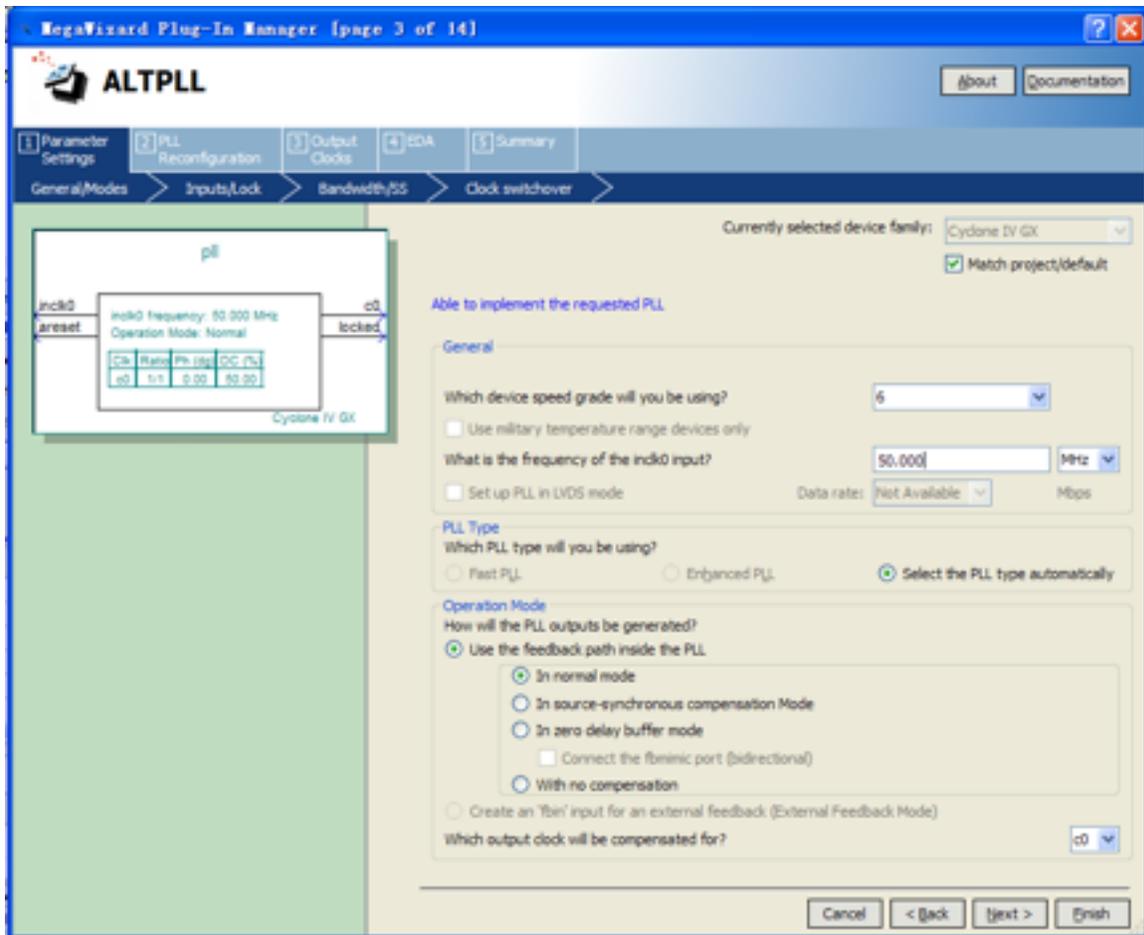
a. Escoge I/O > ALTPLL.

- b. Under which device family will you be using? Escoge the Cyclone IV GX para la tarjeta DE2i-150.
- c. Under Which type of output file do you want to create? Escoge Verilog HDL.
- d. Under What name do you want for the output file? Escribe pll al final del directorio ya creado.
- e. Click Next.

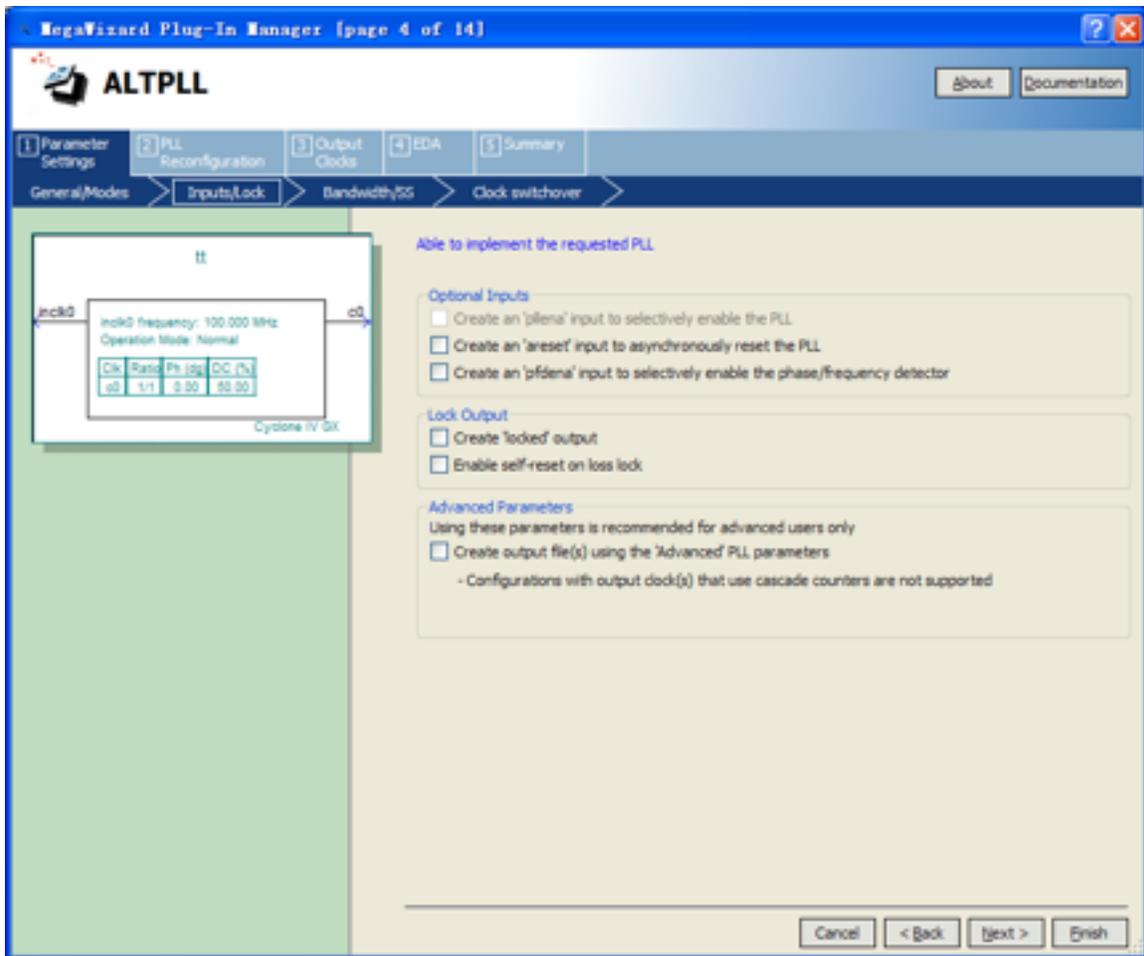


5. En la ventana del Manager del MegaWizard Plug-In [pagina 3 fe 14], crea las siguientes secciones.

- a. Confirma que la opción de la familia del dispositivo seleccionado muestre el dispositivo que corresponde a la tarjeta que estas usando.
- b. Escoge 6 en el grado de velocidad para DE2i-150.
- c. Ingresa 50 MHz. a la frecuencia del inclock0.
- d. Click Next.

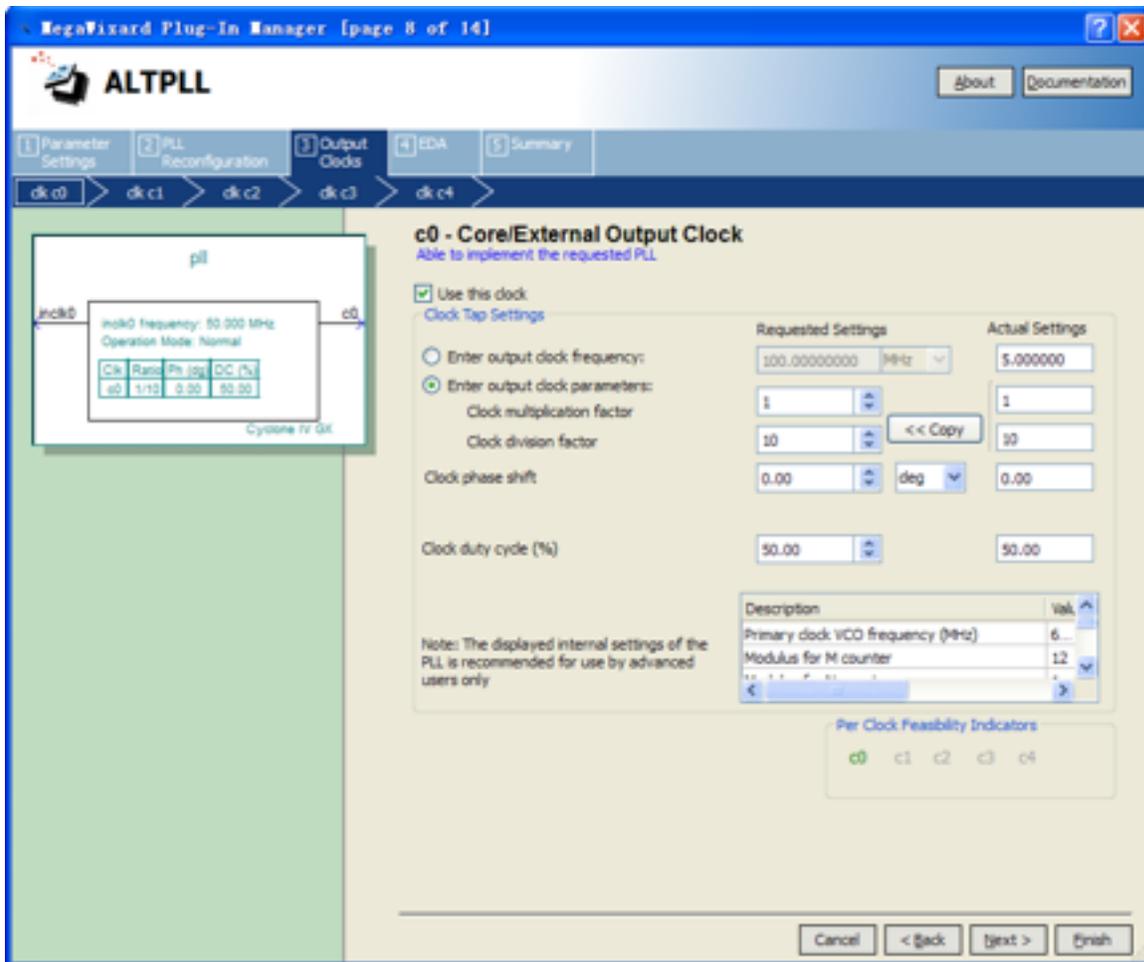


6. Apaga todas las opciones en el MegaWizard página 4. Cuando las estés apagando, los pines desaparecerán del bloque PLL.



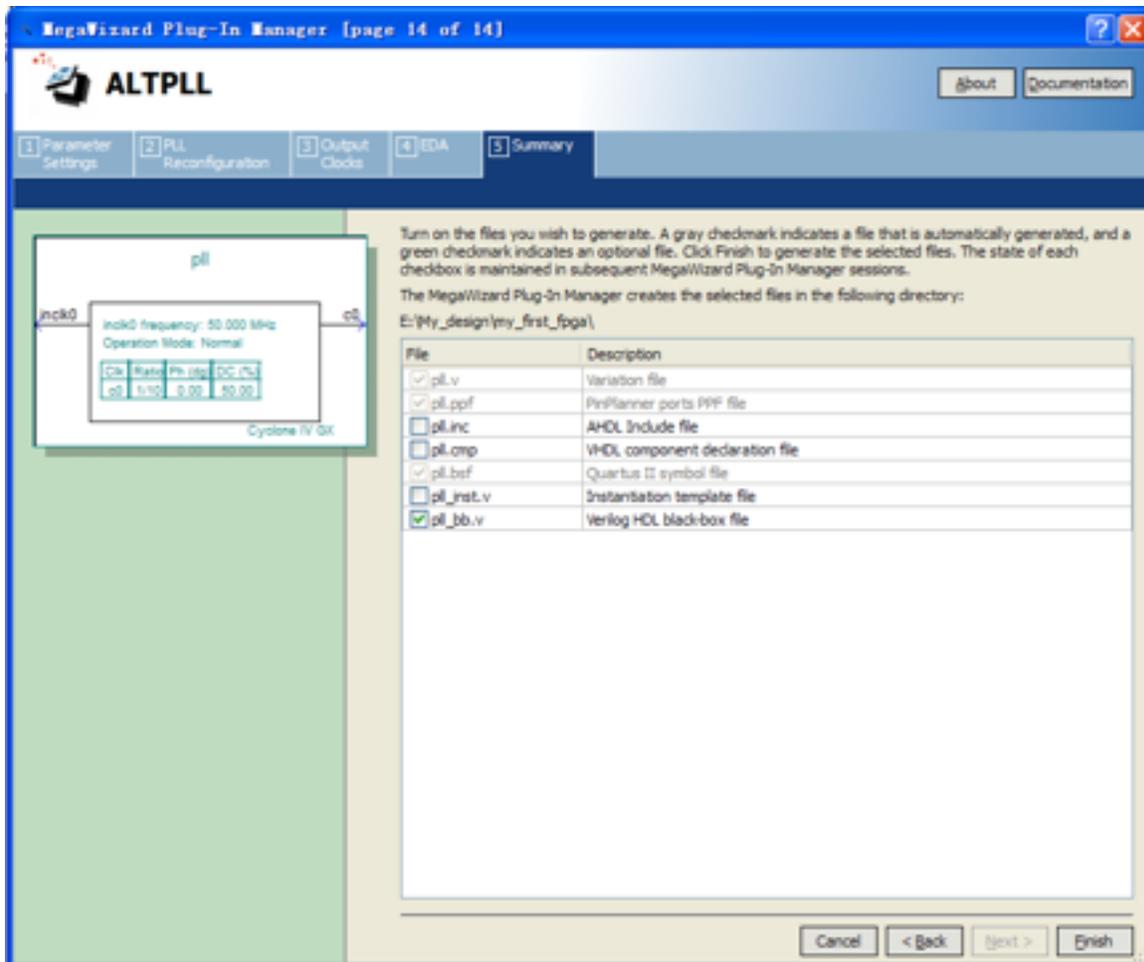
7. Click Next cuatro veces más, en el la ventana del MegaWizard Plug-in Manager [page 8 of 14], has las siguientes selecciones.

Clock División Settings input 10

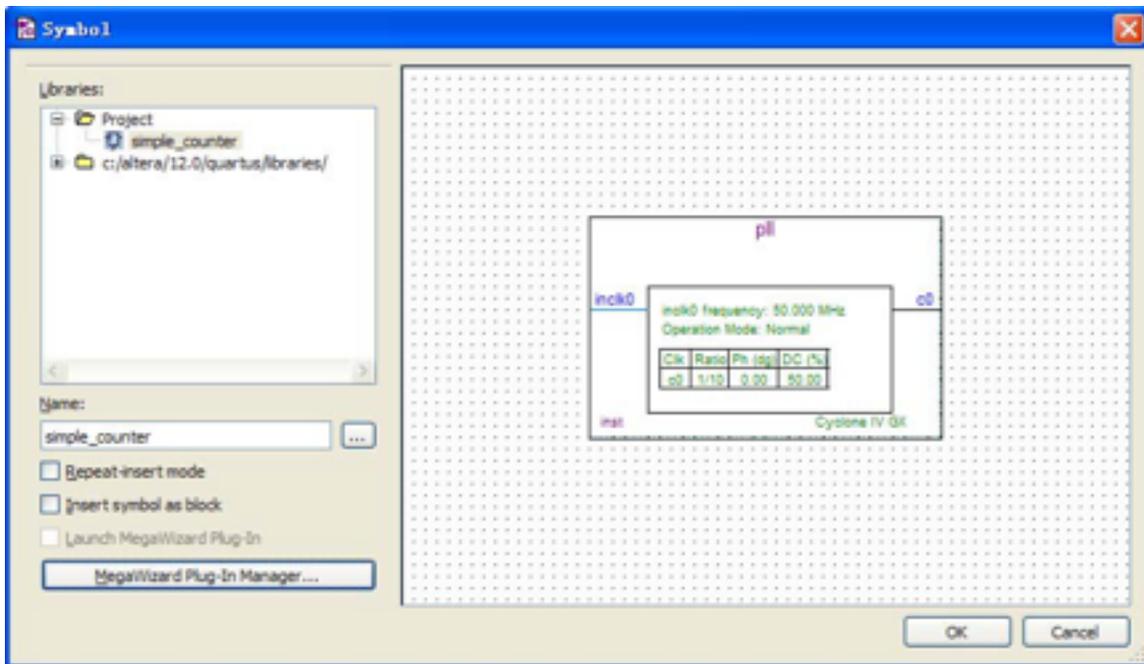


9. Click Next.

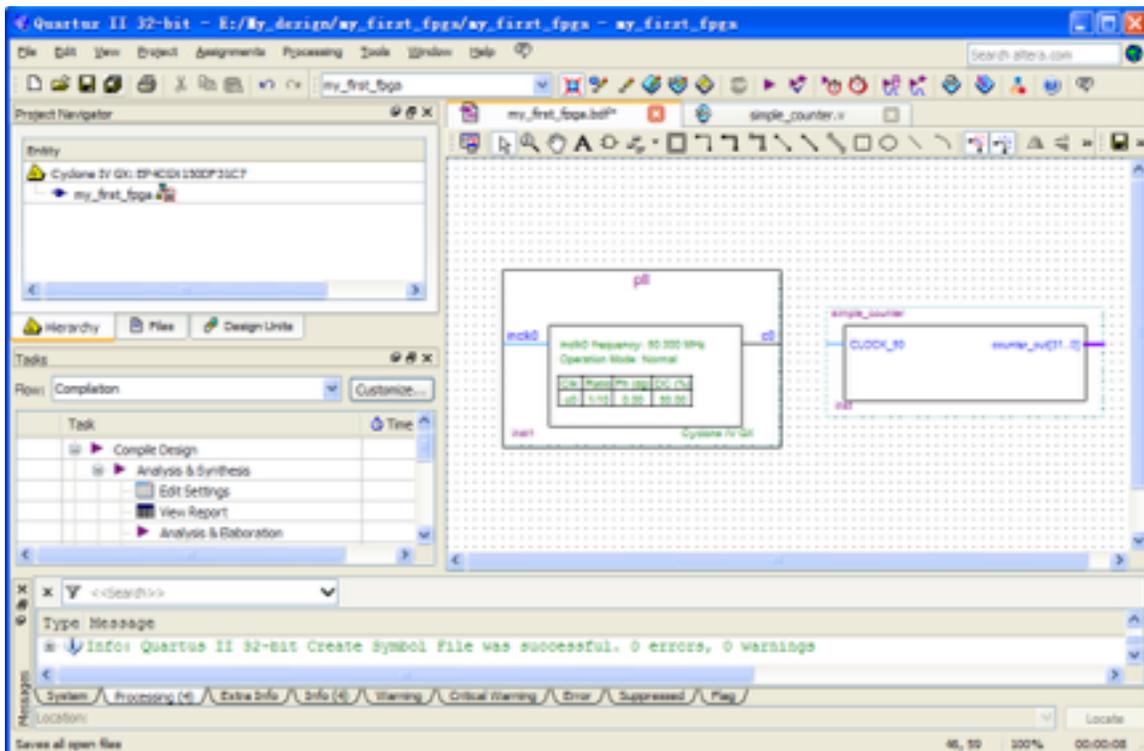
10. El wizard despliega un sumario de los archivos creados. Click Finish.



La ventana símbolo aparece, mostrando la nueva megafunción PLL creada.

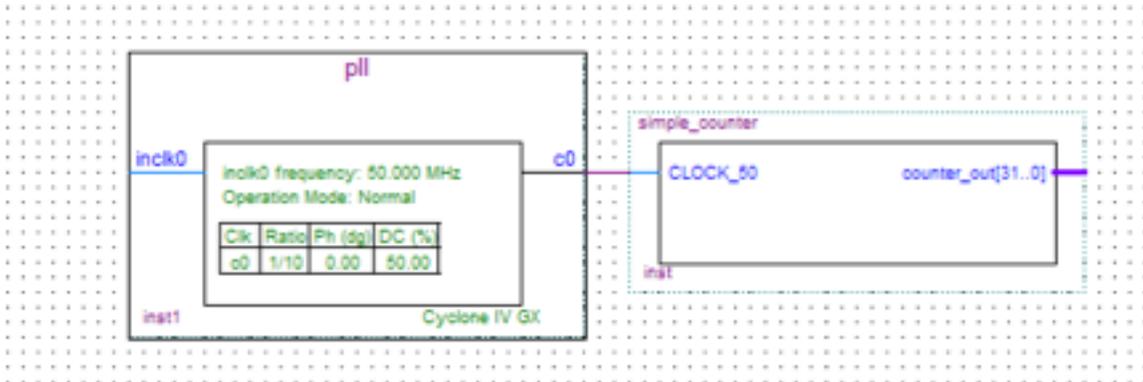


11. Click OK and place the pll symbol to the left of the simple_counter symbol. You can move the symbols around by clicking the left mouse button. Use this to make sure they are aligned correctly.



12. Move the mouse so that the cursor is over the pll c0 output pin. The icon of the orthogonal node tool appears.

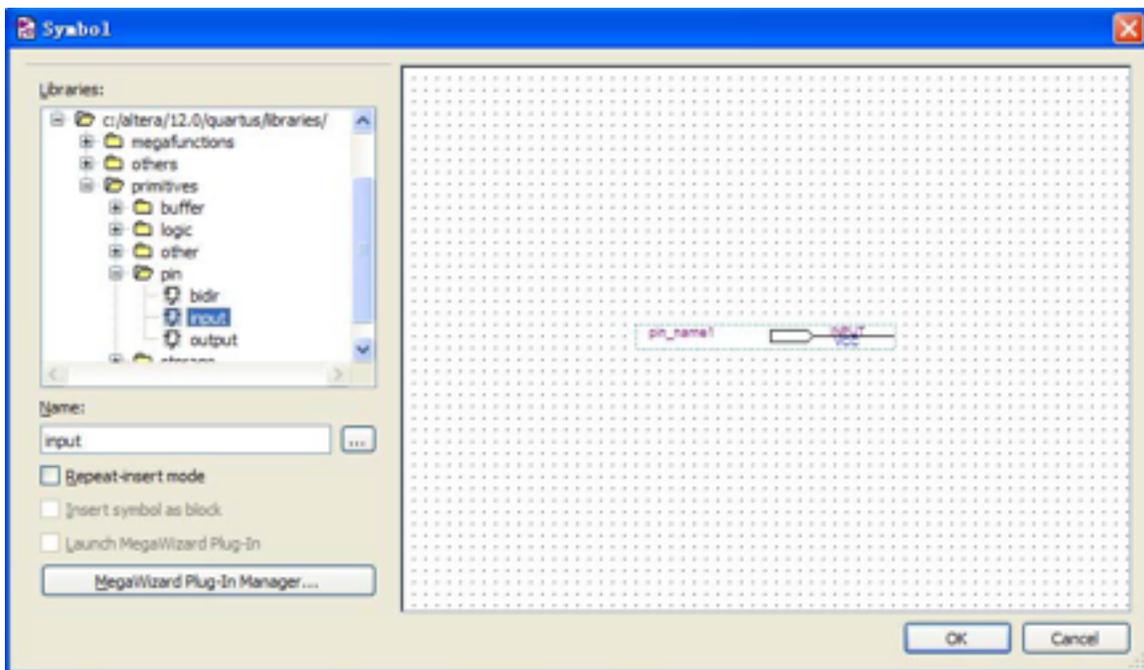
13. Click y arrastra un bus line desde el c0 output hacia el simple_counter clock input. Esta acción ata la salida del pll a la entrada del simple_counter.



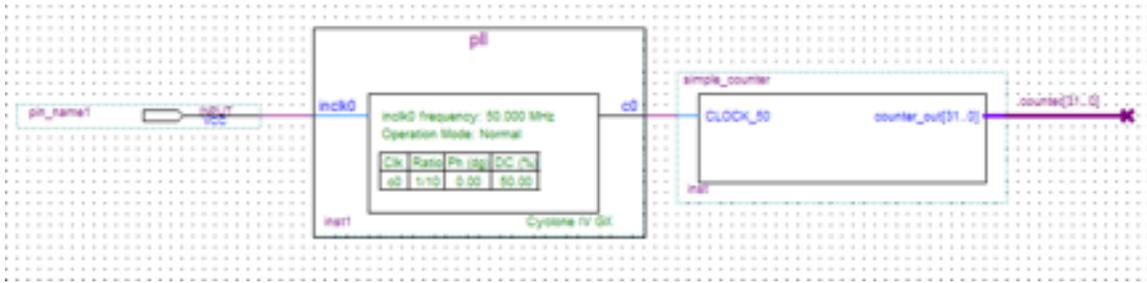
14. Agrega un pin de entrada y un bus de salida con los siguientes pasos:

- Escoge Edit > Insert Symbol.
- Under Libraries, select quartus/libraries > primitives > pin > input.
- Click OK

Si necesitas más espacio para colocar símbolos, puedes usar la barra horizontal y vertical a las orillas de la ventana BDF.

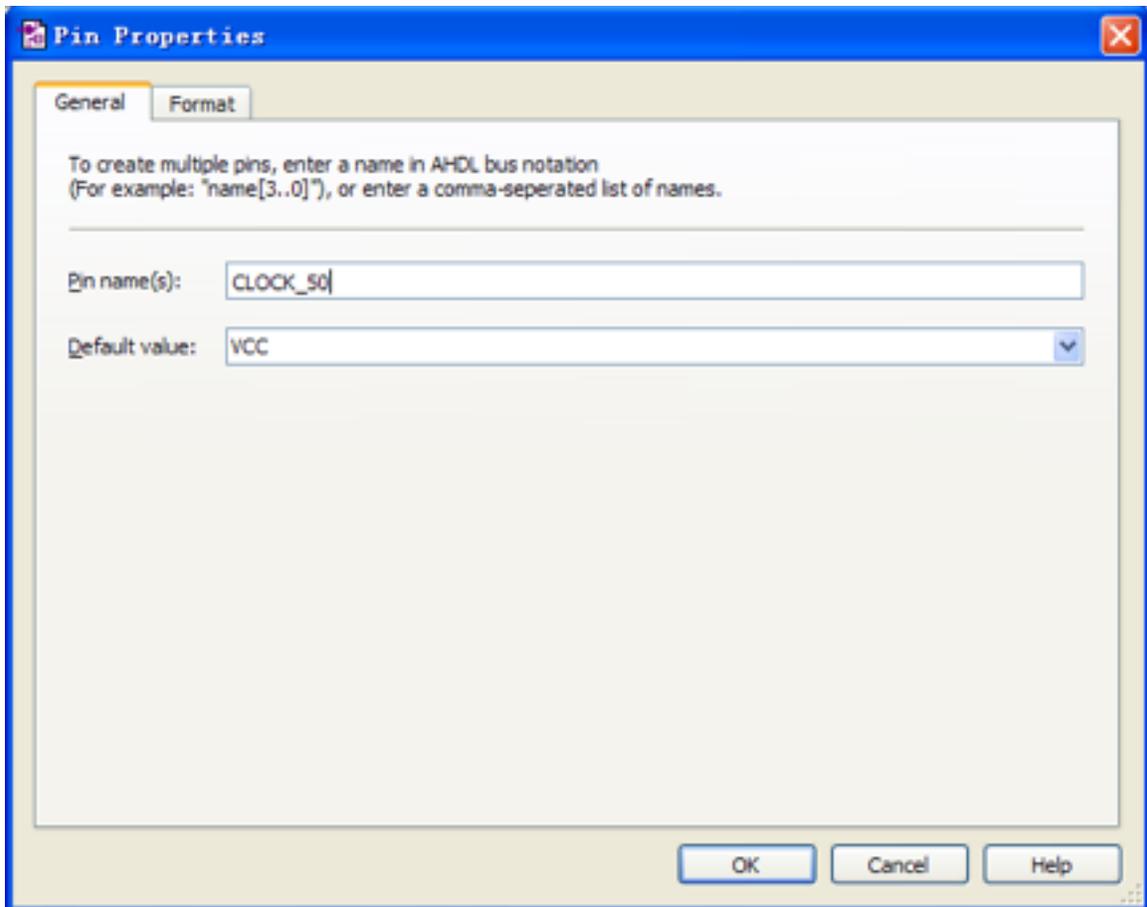


- coloca el nuevo pin dentro del BDF de manera que este tocando la entrada del símbolo pll.
- Utiliza el mouse para dar click y arrastrar la nueva salida hacia la izquierda.



f. Cambia el nombre del pin al hacer doble click en pin_name y teclear CLOCK_50 este nombre se relaciona con el reloj oscilador que está conectado a el FPGA.

g. Usando el Orthogonal Bus tool, dibuja una línea de bus conectado en un lado de la salida del simple_counter, y deja el otro extremo desconectado por 4 a 8 grid espacios hacia la derecha del simple_counter.



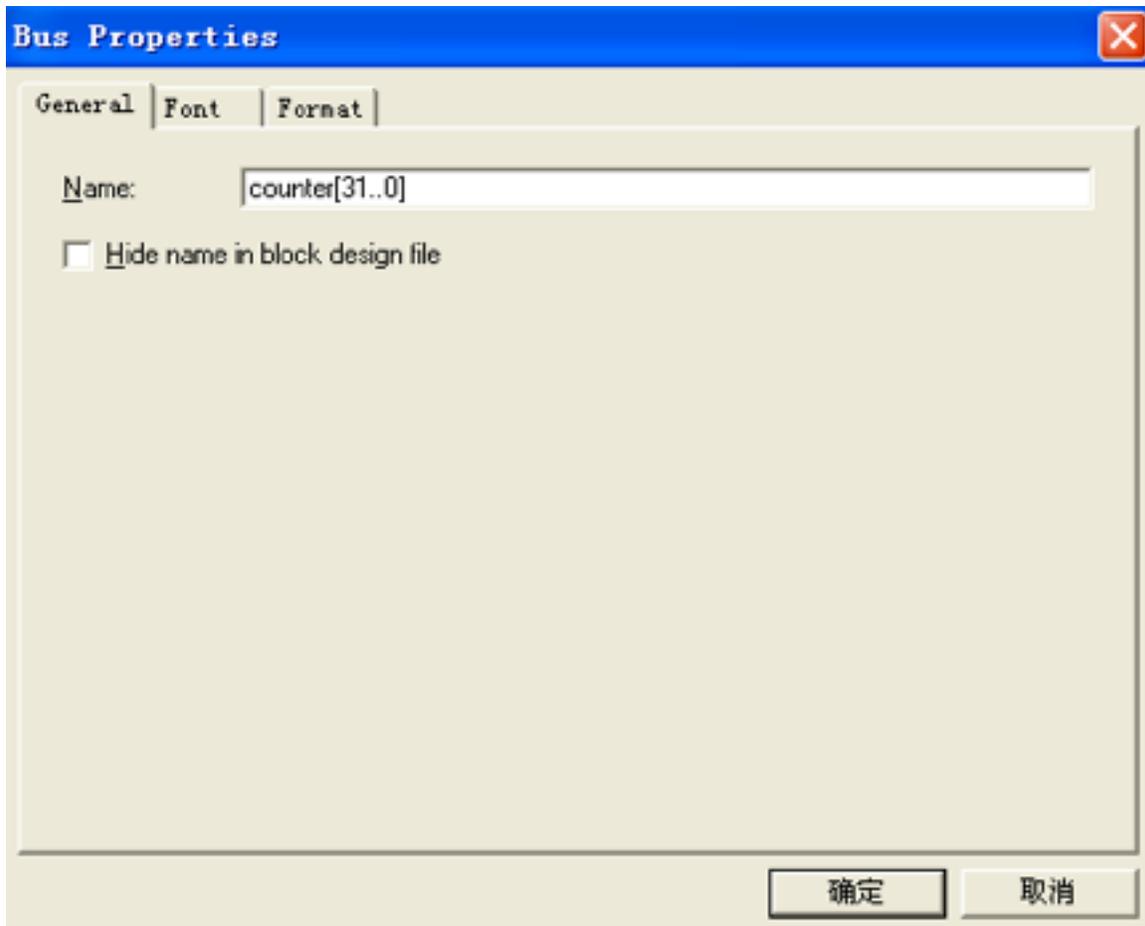
h. Click derecho en la nueva línea de salida y escoge propiedades.

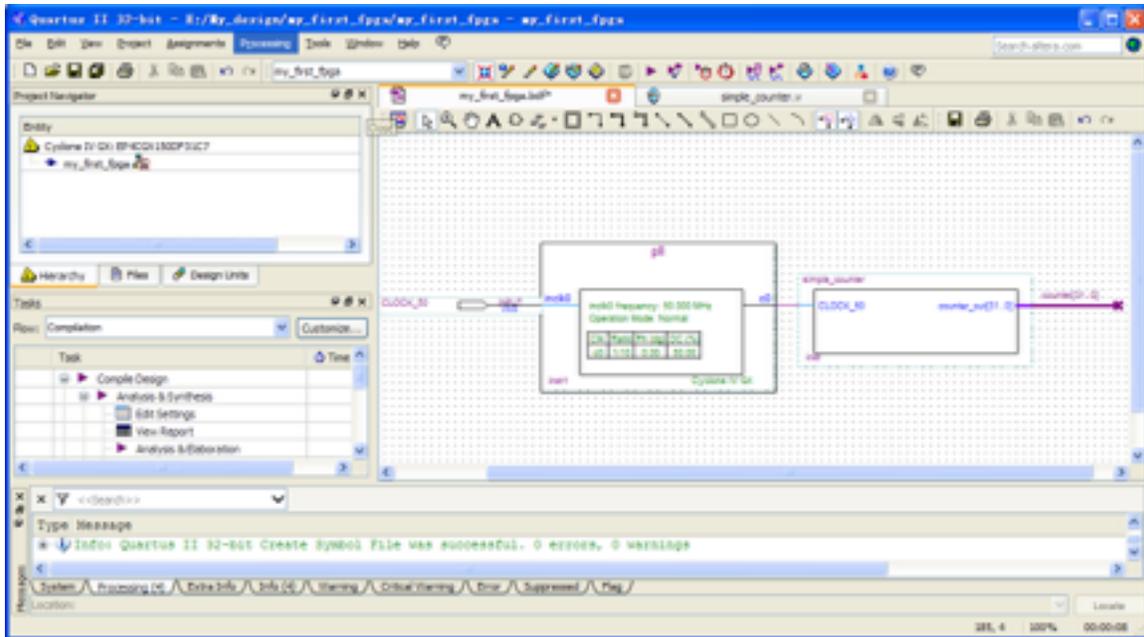
i. Usando el Orthogonal Bus tool, dibuja una línea de bus conectada a un lado de la salida del simple_spunter, y deja el otro extremo desconectado de 6 a 8 grid espacios hacia la derecha del simple_counter.

j. Escribe contador [31..0] como un nombre de bus. La notación [X..Y] es el método que utiliza

Quartus II para especificar el bus con esquemas BDF, donde X es el bit más significativo (MSB) y Y es el menos significativo (LSB).

k. Click OK

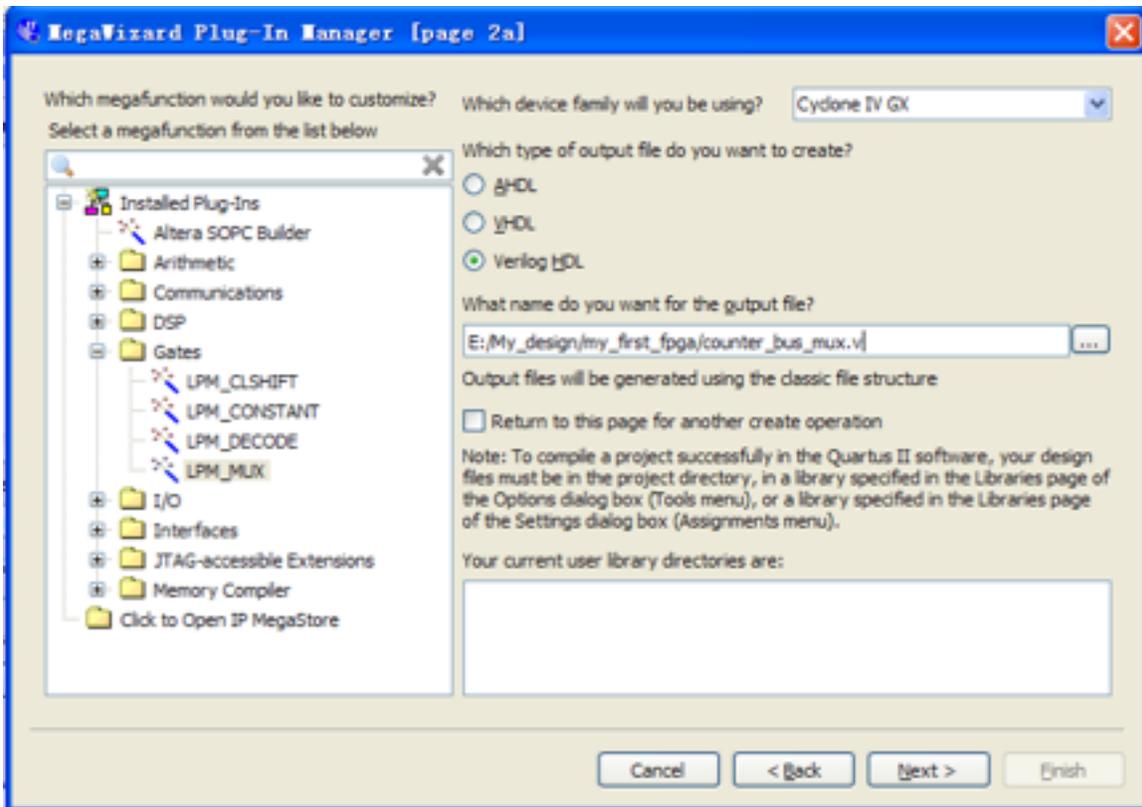




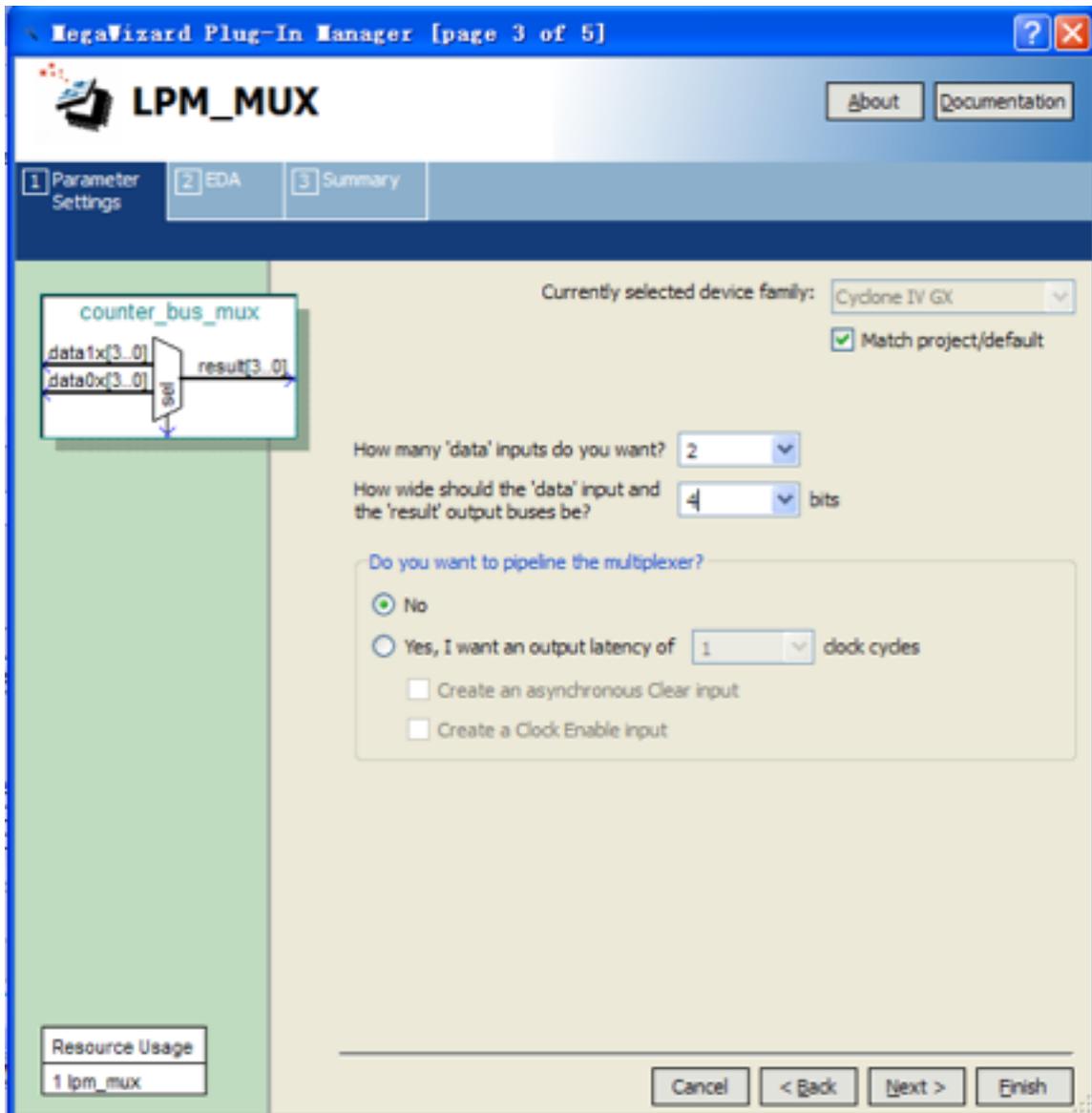
4.2 Agregar Un Multiplexor

Este diseño utiliza un multiplexor para enrutar la salida del simple_counter hacia los pines del LED en la tarjeta DE2i-150. Vas a usar el MegaWizard Plug-In manager para agregar un multiplexor, lpm_mux. El diseño de multiplexores tiene dos variaciones del bus contador a cuatro LEDs en la tarjeta DE2i-150.

1. Escape Edit > Insert Symbol.
2. Click MegaWizard Plug-in Manager.
3. Click Next.
4. Escoger Installed Plug-Ins > Gates > LPM_MUX.
5. Escoger la familia del dispositivo que corresponde al dispositivo en la tarjeta que estas usando, escoge Verilog HDL como el tipo de archivo de salida, y nombra el archivo de salida como counter_bus_mux.v.
6. Click Next.

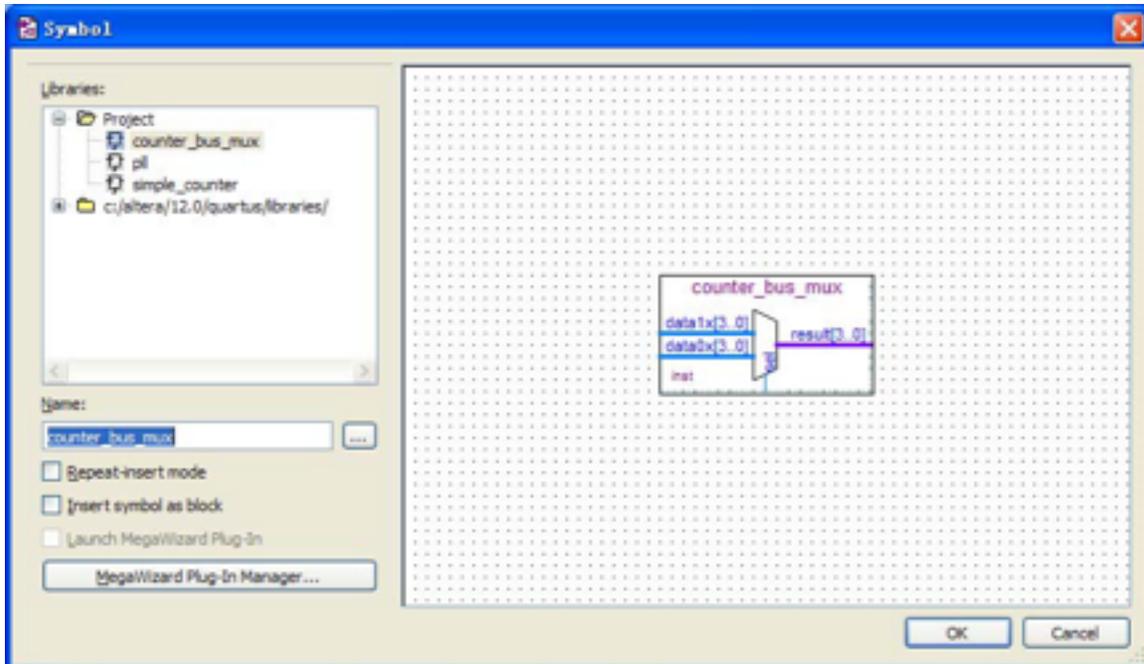


7. Under How many 'data' inputs do you want? Selecciona 2 entradas (default).
8. Under how 'wide' should the data input and result output be? Selecciona 4



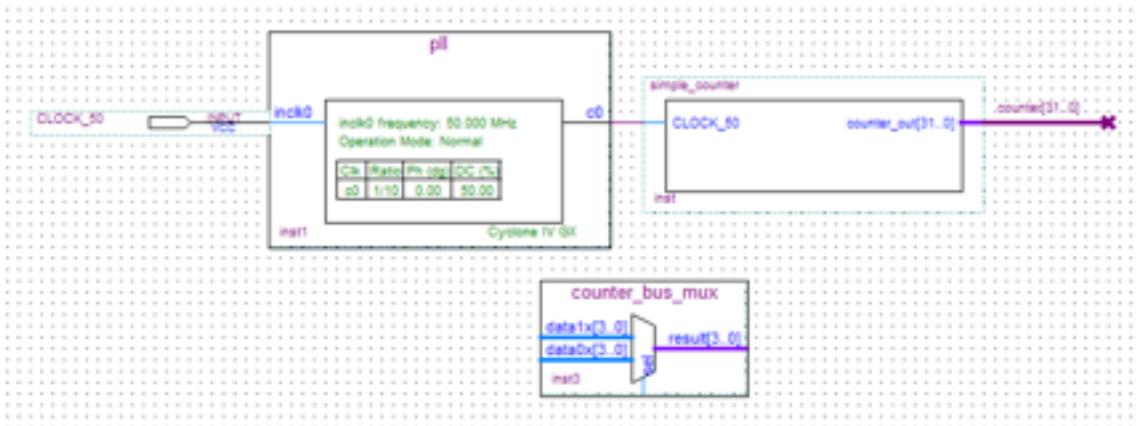
9. Click Next.

10. Click Finish dos veces. La ventana del símbolo aparece.



11. Click OK

12. Coloca el símbolo del counter_bus_mux bajo los símbolos existentes en el BDF.



13. Agrega buses de entrada y pines de salida para el símbolo counter_bus_mux como se muestra a continuación:

a. Usando el Orthogonal Bus tool, dibuja líneas de bus del data1x[3..0] y datax0[3..0]

Los puertos de entrada de 8 a 12 grid espacios hacia la izquierda del counter_bus_mux.

b. Dibuja una línea de bus de la salida resultante del puerto [3..0] de 4 a 8 grid espación hacia la derecha del conunter_bus_mux.

c. Click derecho en la línea de bus conectada a data1x[3..0] y escoge propiedades.

d. Nombra el bus counter[26..23], el cual selecciona solo aquellos counter output bits para

conectar los cuatro bits de la data1x input.

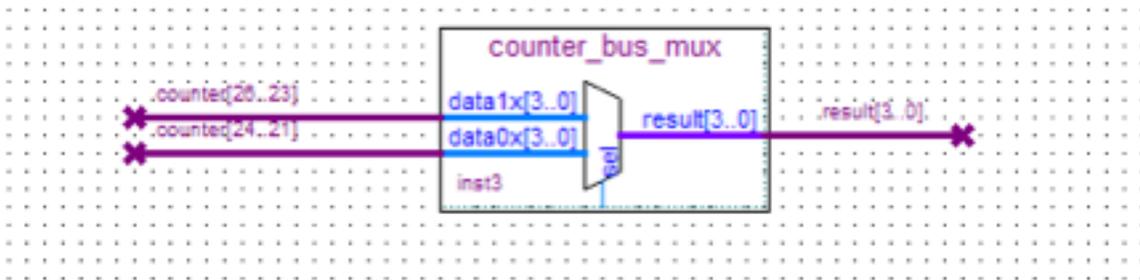
Debido a que los buses de entradas al counter_bus_mux tienen los mismos nombres que el bus de salida de simple_counter, (counter[x.y]) el software del Quartus II sabe conectar estos buses.

e. Click OK

f. Click derecho en la línea del bus conectada a data0x[3..0] y escoge propiedades.

g. Nombre los buses counter[24..21], de los cuales selecciona solo los bits de las salidas para conectar a los cuatro bits de entrada del data1x.

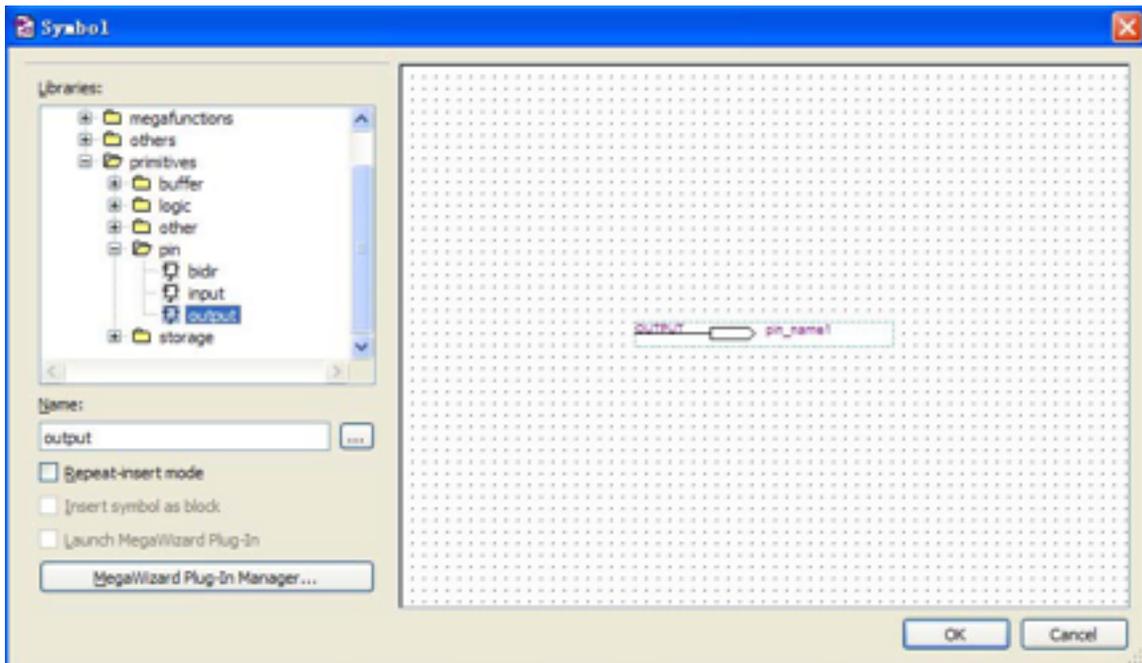
h. Click OK



Si no lo has hecho ya, guarda el proyecto antes de continuar.

14. Escoge Edit > Insert Symbol

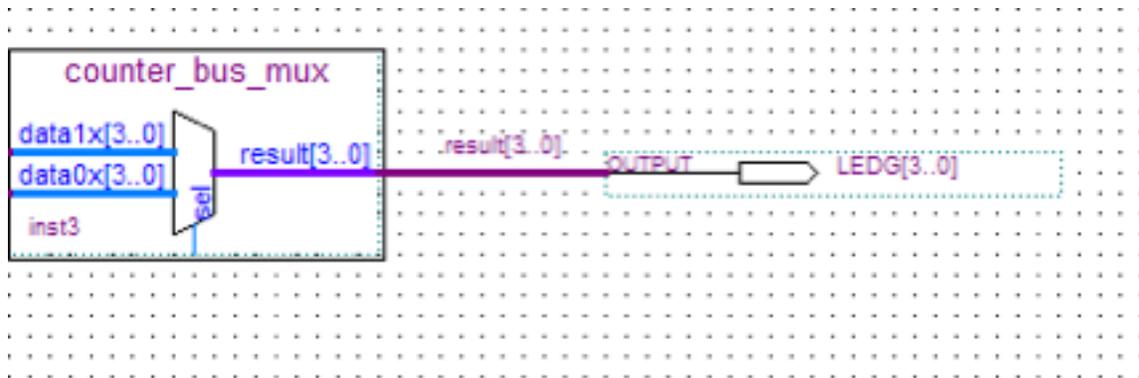
15. Bajo las librerías, doble click quartus/libraries/> primitives > pin > output



16. Click OK

17. Coloca pin de salida de manera que se conecte al resultado de la línea del bus counter_bus_mux [3..0].

18. Renombra el pin de salida como LEDG [3..0] como se describe en el paso 13 c y d.



19. Coloca un pin de salida a la línea de selección multiplexor usando un pin de salida:

a. Escoge Edit > Insert Symbol.

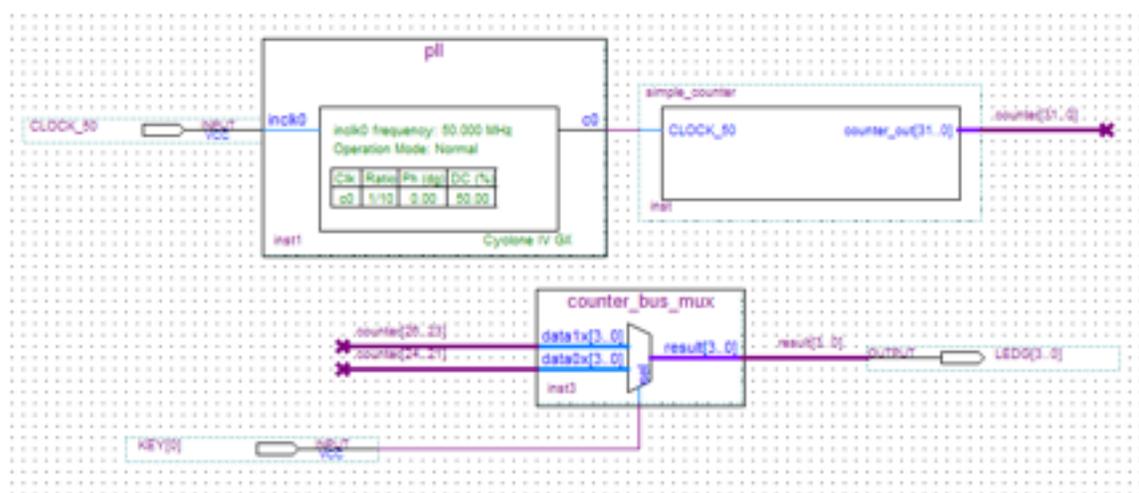
b. Bajo las librerías, doble click en quartus/libraries/ > primitives > pin > input.

c. Click OK.

20. Coloca este pin de salida bajo counter_bus_mux.

21. Conecta el pin de salida al pin del counter_bus_mux sel.

22. Renombra el pin de salida como KEY [0].



Has terminado de agregar los símbolos a tu diseño.

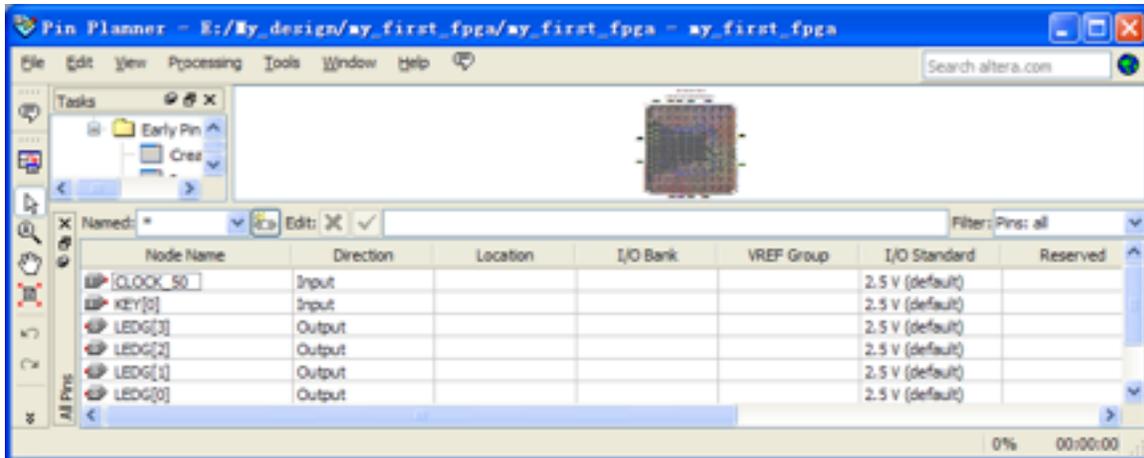
4.3 Asignar los pines.

En esta sección vas a asignar los pines. Antes de asignar los pines, realiza los siguientes pasos.

1. Escoge Processing > Start > Start Analysis & Elaboration in preparation for assigning pin locations.
2. Click OK en la ventana de mensaje que aparece después del análisis y elaboración completados.

Para hacer las asignación de pines que se relaciona con KEY [0] y CLOCK_50 input pins y LEDG[3..0] output pin, realiza los siguientes pasos.

1. Escoge Assignments > Pins, donde se abre el Pin Planner, una tabla spreadsheet-like de pines específicos. El planeador de pines muestra el diseño de seis pines.



2. En la locación de la columna junto a cada uno de los nombres de los nodos, agrega las coordenadas.

Pin Name FPGA Pin Location

KEY[0] AA26

LEDG[3] F26

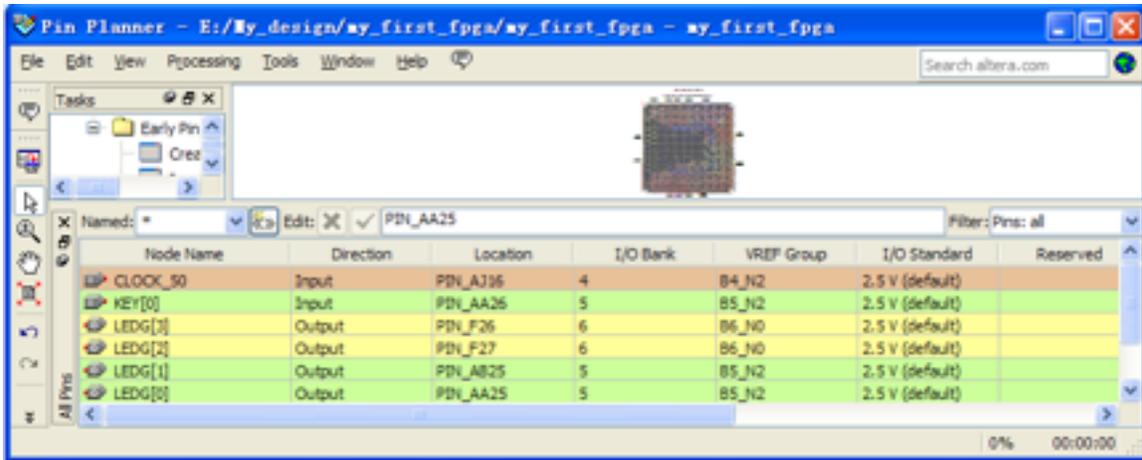
LEDG[2] F27

LEDG[1] AB25

LEDG[0] AA25

CLOCK_50 AJ16

Doble click en Location column para cualquiera de los seis pines para abrir una lista y escribir la locación mostrada en la tabla, puedes seleccionar el pin de una lista. Por ejemplo, si presionas F1 y presionas la tecla enter, el Quartus II rellena toda la locación PIN_F1 por ti. El software mantiene un seguimiento de los datos del FPGA correspondientes así como el banco I/O y el grupo VREF. Cada banco tiene un color distinto, el cual corresponde a lazo dibujado en la parte superior de la ventana.



Ahora, has terminado de crear tu diseño en Quartus II.

4.4 Crea un archivo TimeQuest SDC default

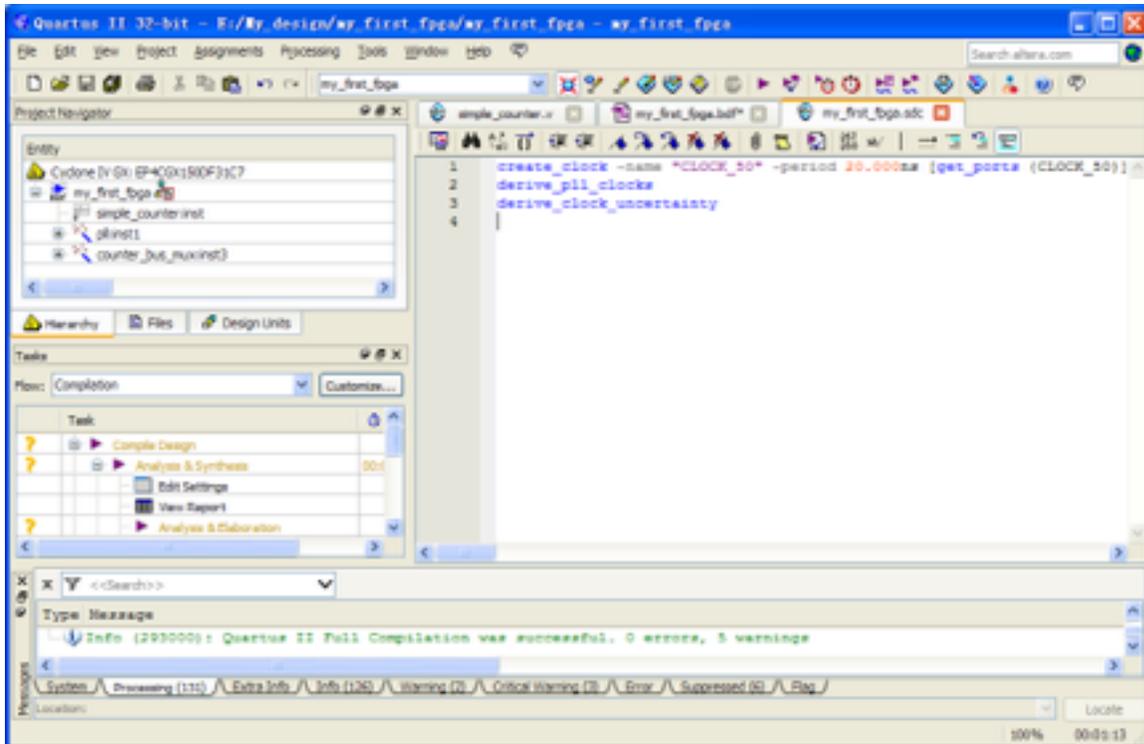
La sincronización es importante para un diseño exitoso. Para este tutorial vas a crear un archivo Synopsys Design Constraints básico (.sdc) que el Quartus II TimeQuest Timing Analyzer usa durante el compilado del diseño. Para diseños más complejos, necesitaras considerar los requerimientos de sincronización.

Para crear un SDC, realiza los siguientes pasos:

1. Abre el TimeQuest Timing Analyzer seleccionando Tools > TimeQuest Timing Analyzer.
2. Selecciona File > New SDC file. El SDC editor se abrirá.
3. Escribe el siguiente código en el editor:

```
create_clock -name "CLOCK_50" -period 20.000ns [get_ports {CLOCK_50}]
derive_pll_clocks
derive_clock_uncertainty
```

4. Guarda este archivo como my_first_fpga.sdc



Nombrando el SDC con el mismo nombre que el archivo a excepción de la extensión .sdc causa que el Quartus II utilice el análisis de sincronización por default. Si necesitas otro nombre, necesitaras agregar el SDC a la lista de asignaciones de archivos.

5 COMPILA Y VERIFICA TU DISEÑO

Después de crear tu diseño debes compilarlo. La compilación convierte el diseño es bitstream que puede ser descargado en el FPGA. La salida más importante de la compilación es el archivo objeto SRAM (.sof), el cual se utiliza para programar el dispositivo. Además el software genera los reportes que proveen información acerca del código que se está compilando.

5.1 Compila tu diseño

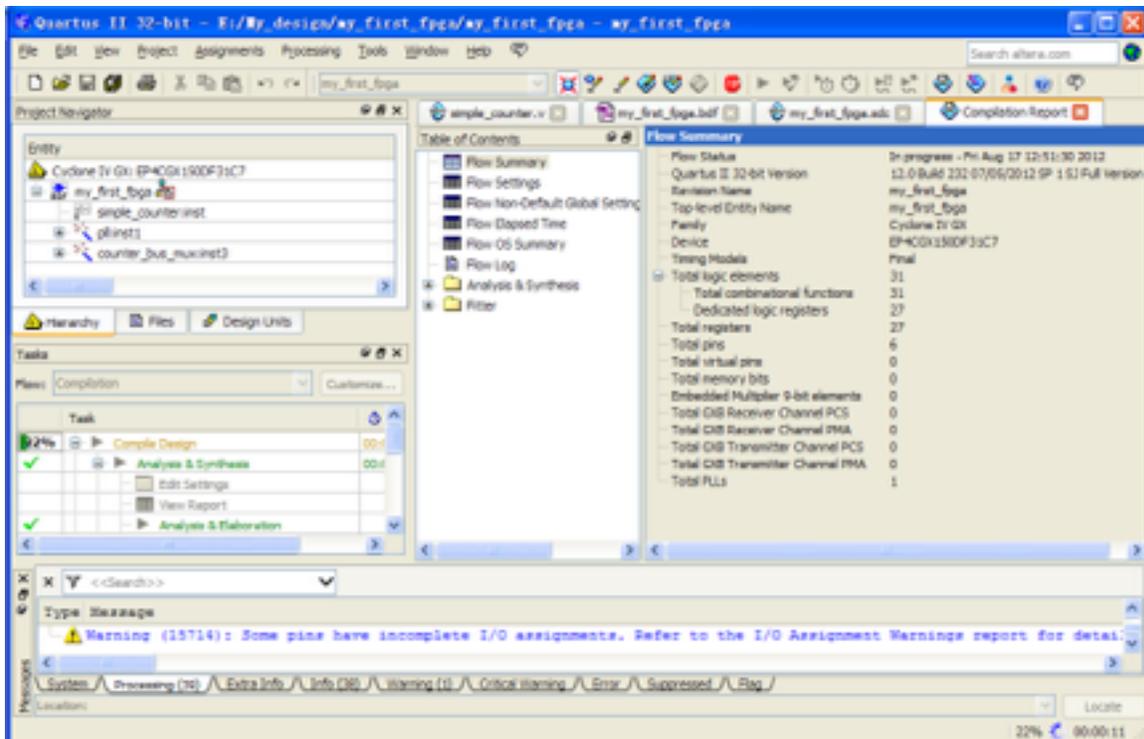
Si quieres guardar el .SOF en la memoria del dispositivo, debes primero convertir el SOF al tipo de archivo para al dispositivo de memoria.

Ahora que has creado un proyecto de Quartus II completo e ingresado todas las asignaciones, puedes compilar el diseño.

En el menú de proceso, escoge Start Compilation o click el botón Play en la barra de herramientas.

Si se te pide guardar los cambios a tu BDF, click YES.

Mientras se compila tu diseño, el Quartus II provee información útil acerca de la compilación.



Cuando la compilación este completa, el Quartus II despliega un mensaje. Click OK para cerrar el mensaje.

La ventana del Quartus II despliega muchos mensajes durante la compilación. No debería desplegar ningún mensaje crítico; debería desplegar algunas advertencias que indicaran que la sincronización del dispositivo es preliminar o que algunos parámetros de entrada/salida de los pines usados para los LEDs no fueron usados. El software provee los resultados de compilación en el reporte de compilación.

Flow Summary	
Flow Status	Successful - Fri Aug 17 12:52:25 2012
Quartus II 32-bit Version	12.0 Build 232 07/05/2012 SP 1 S1 Full Version
Revision Name	my_first_fpga
Top-level Entity Name	my_first_fpga
Family	Cyclone IV GX
Device	EP4CGX1500F31C7
Timing Models	Final
Total logic elements	31 / 149,760 (< 1%)
Total combinational functions	31 / 149,760 (< 1%)
Dedicated logic registers	27 / 149,760 (< 1%)
Total registers	27
Total pins	6 / 508 (1%)
Total virtual pins	0
Total memory bits	0 / 6,635,520 (0%)
Embedded Multiplier 9-bit elements	0 / 720 (0%)
Total GXB Receiver Channel PCS	0 / 8 (0%)
Total GXB Receiver Channel PMA	0 / 8 (0%)
Total GXB Transmitter Channel PCS	0 / 8 (0%)
Total GXB Transmitter Channel PMA	0 / 8 (0%)
Total PLLs	1 / 8 (13%)

5.2 Programa el dispositivo FPGA

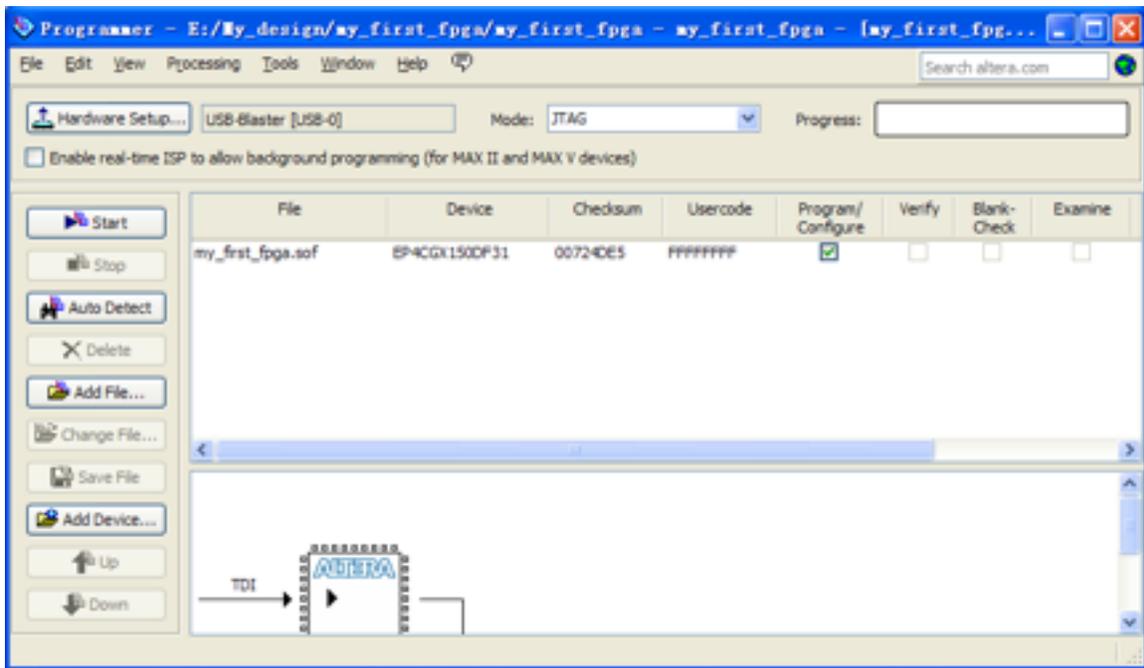
Después de compilar y verificar tu diseño estás listo para programar el FPGA en la tarjeta.

Descargas el SOF que acabas de crear en el FPGA usando el circuito USB-Blaster en la tarjeta. Prepara tu hardware para programar siguiendo los siguientes pasos:

- a) Conecta la fuente de poder a la tarjeta y a un enchufe.
- b) Para la tarjeta DE2i-150, conecta la USB-Blaster al j9 y el cable USB al USB-Blaster. Conectado al otro extremo del cable USB a la computadora huésped.
- c) Enciende la tarjeta usando al switch on/off.

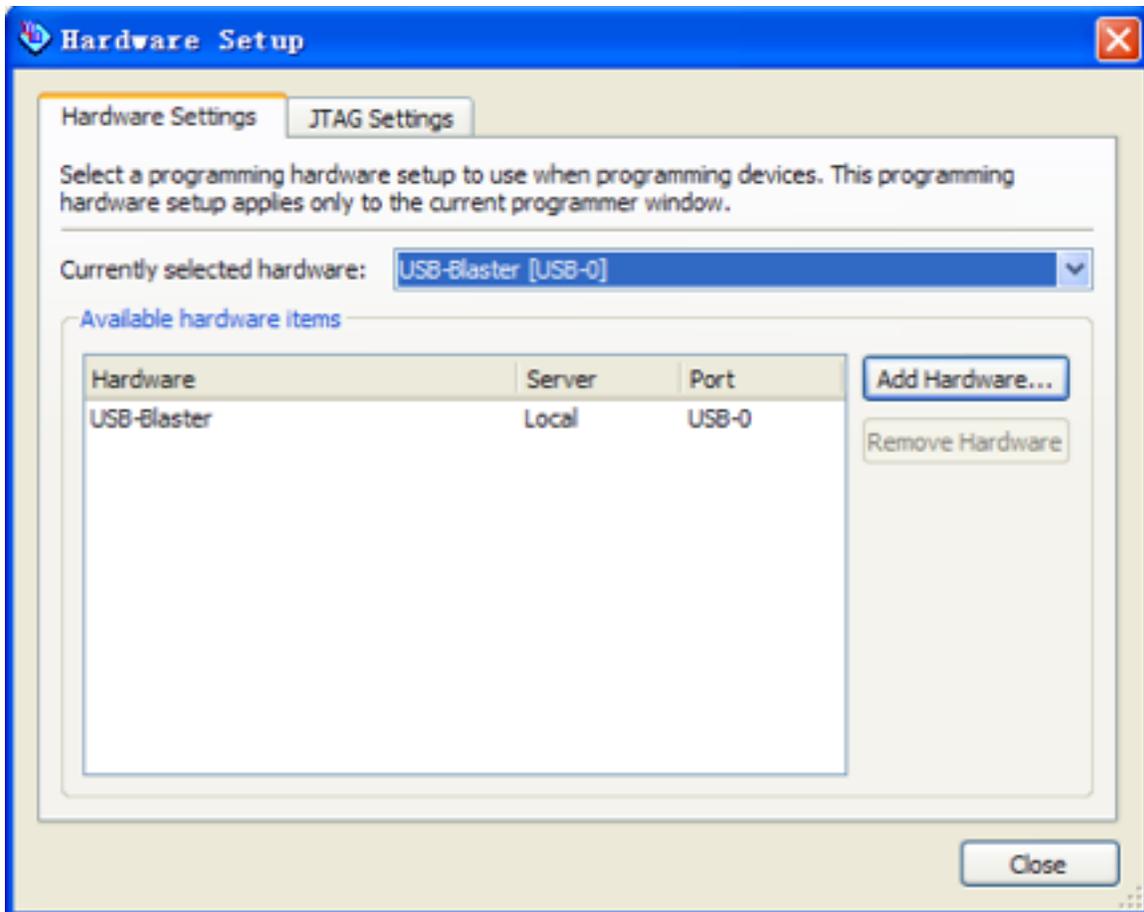
Programa el FPGA siguiendo los siguientes pasos.

1. Escoge Tools > Programmer. La ventana del programador se abre.



2. Click Hardware Setup.

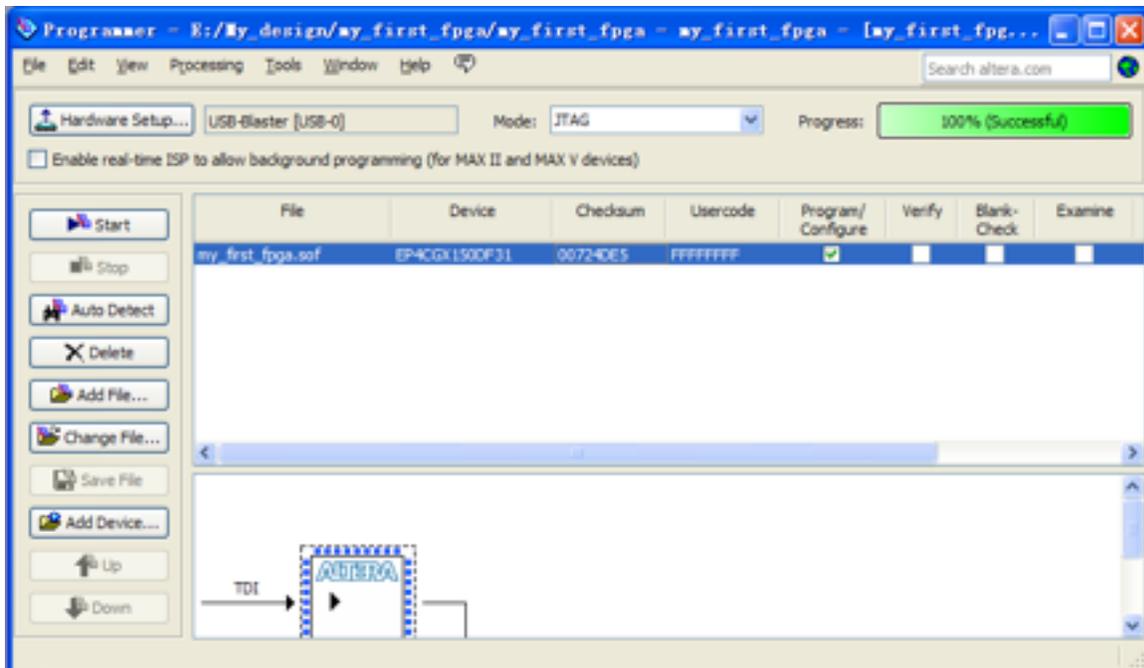
3. Si aún no está encendido, asciende la opción USB-Blaster [USB.0] bajo el actual hardware seleccionado.



4. Clicik Close.

5. Si el archivo en el programador no muestra my_first_fpga.sof, click agregar archivo.

6. Selecciona el archivo my_first_fpga.sof del directorio del proyecto.



5.3 Verifica el hardware

Cuando verificas el diseño en hardware, observas el comportamiento del tiempo de ejecución del diseño del hardware FPGA y te aseguras de que esté funcionando apropiadamente.

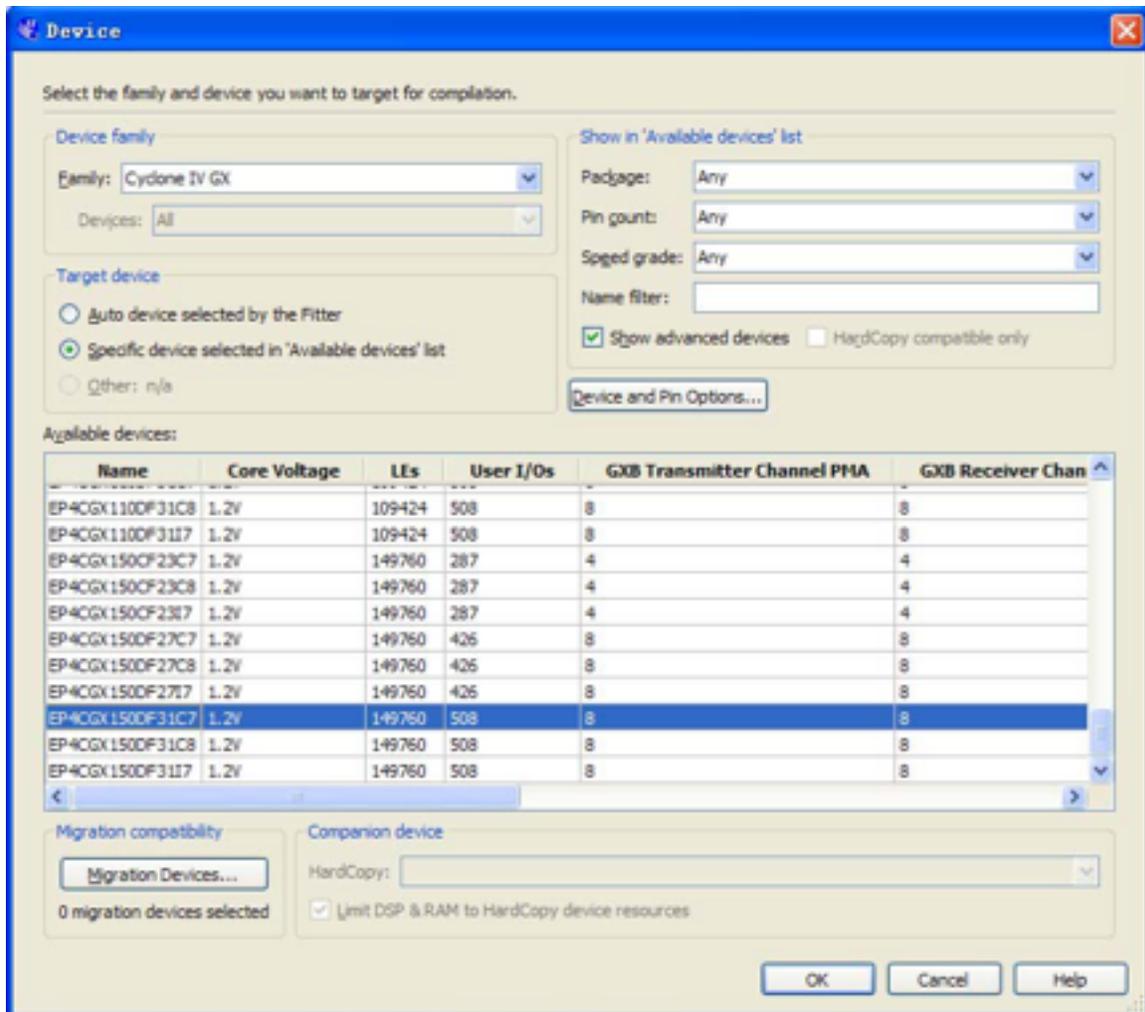
Verifica el diseño siguiendo los siguientes pasos:

1. Observa que los cuatro LEDs de la tarjeta aparentas estar avanzando lentamente en un patrón de cuenta binaria, el cual está controlado por los bits del simple_counter [26..23].

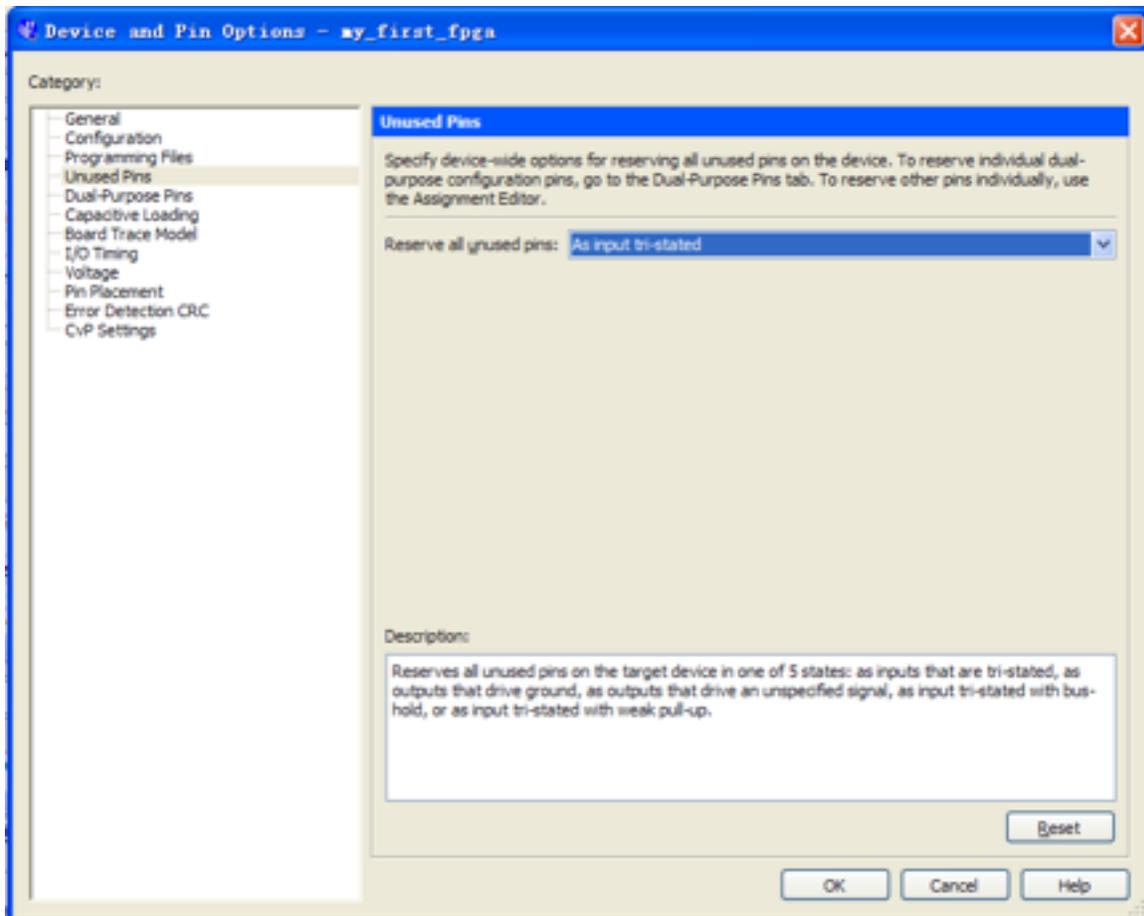
Los LEDs tienen baja actividad, por eso, cuando la cuenta comienza todos los LEDs son encendidos.

2. Mantén presionado KEY [0] en la tarjeta y observa que los LEDs avanzan más rápido. Presionando KEY causa la multiplexion del diseño usando la parte más rápida del counter.

3. Escoge Assignments > Device. Click Device and Options.



Escoge pines sin usar. Reserva todos los pines no utilizados: Escoge la opción As input tri-state.



Has doble click en OK.

4. En el menú de proceso, escoge Start compilation. Después de la compilación, Escoge Tools > Programmer. Selecciona el archivo my_first_fpga.sof del directorio del proyecto. Click Start. Para entonces podrás notar en los otros LEDs apagados.