

Introducción al ISE WebPACK 14.7

Introducción

El Xilinx ISE WebPACK es un conjunto completo de diseño lógico programable FPGA/CPLD que proporciona:

- Especificación de lógica programable mediante captura esquemática o Verilog/VHDL
- Síntesis y “Place & Route” de la lógica especificada para varios FPGA y CPLD de Xilinx
- Simulación funcional (comportamental) y de sincronización (“post-Place & Route”)
- Descarga de datos de configuración en el dispositivo de destino a través del cable de comunicaciones

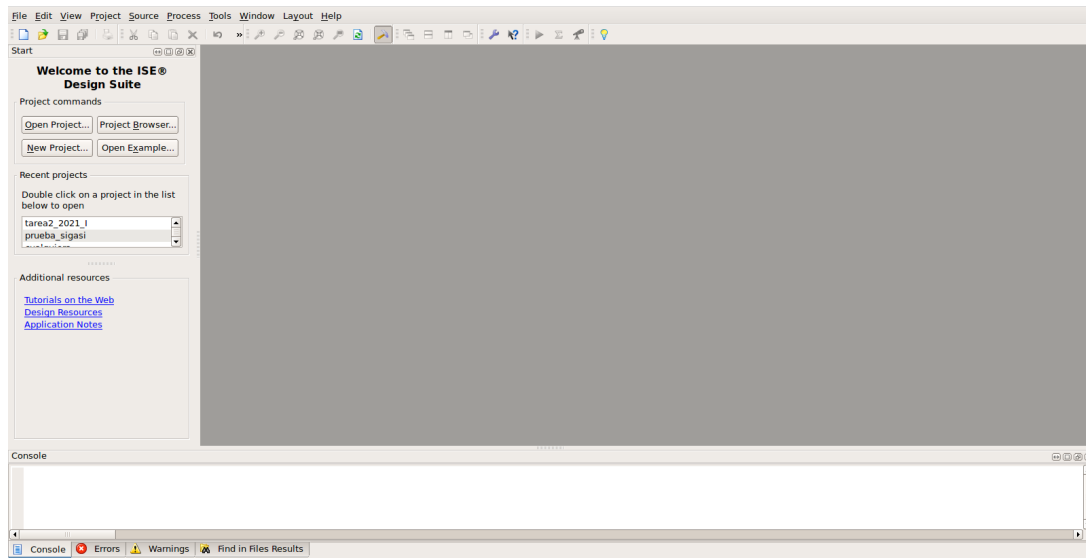
El desarrollo de ISE WebPACK se ha detenido a favor de Vivado Suite, pero sigue siendo útil desarrollarlo para dispositivos más antiguos que no son compatibles con la nueva suite.

Procedimiento

Iniciar Ise WebPack 14.7:

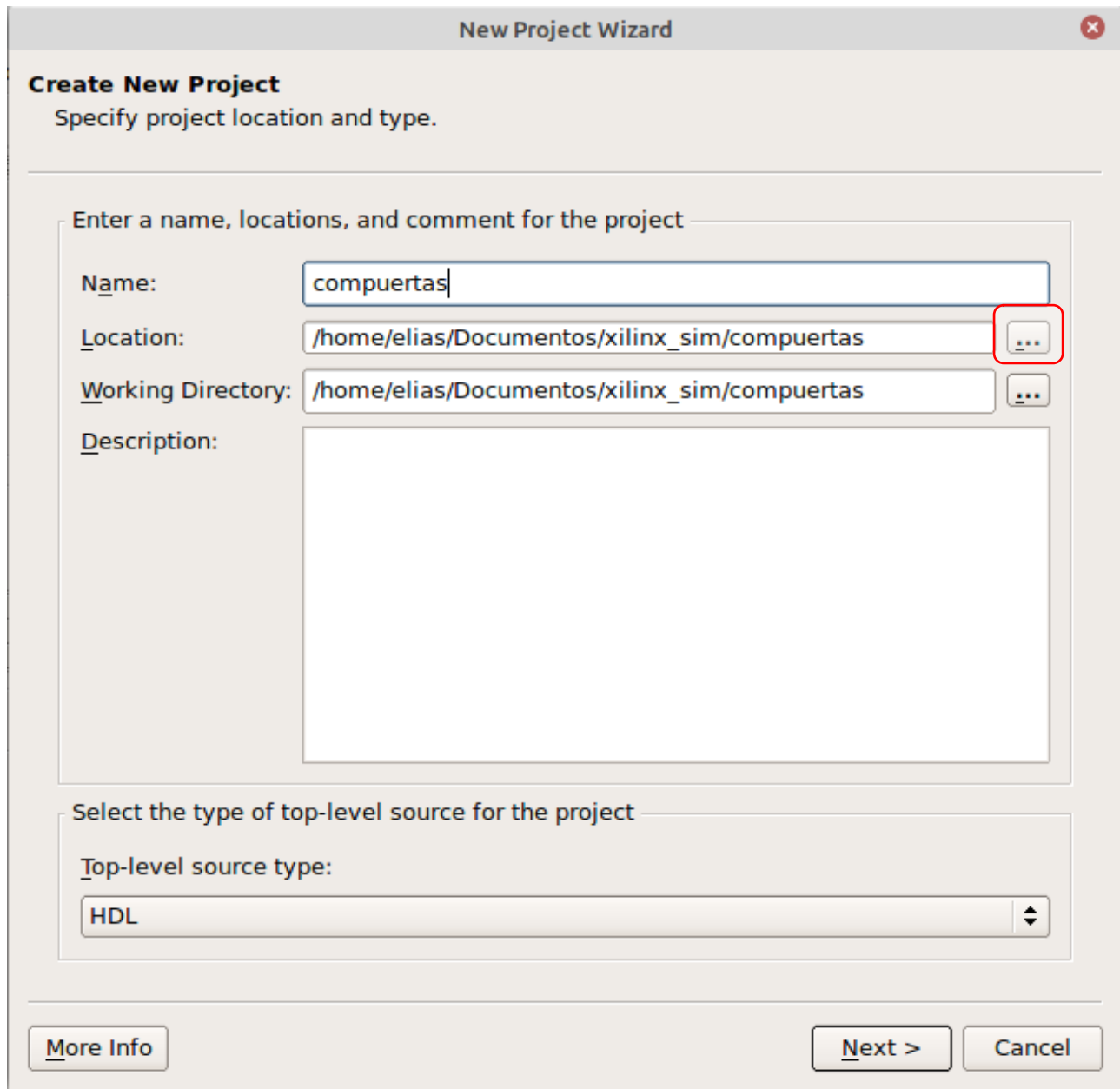


Aparece la ventana del ISE Project Navigator:



Click en New Project... (También File --> New Project)

En esta ventana dar el nombre del proyecto: compuertas, pero antes de esto deberás especificar la ruta en donde quieres almacenar los proyectos.



New Project Wizard

Create New Project
Specify project location and type.

Enter a name, locations, and comment for the project

Name: compuertas

Location: /home/elias/Documentos/xilinx_sim/compuertas ...

Working Directory: /home/elias/Documentos/xilinx_sim/compuertas ...

Description:

Select the type of top-level source for the project

Top-level source type:
HDL

More Info Next > Cancel

Click en Next >.

Aparece el diálogo para elegir la tarjeta que vamos utilizar que es la Nexys2, asegúrese que las opciones queden como se muestra.

New Project Wizard

Project Settings
Specify device and project properties.

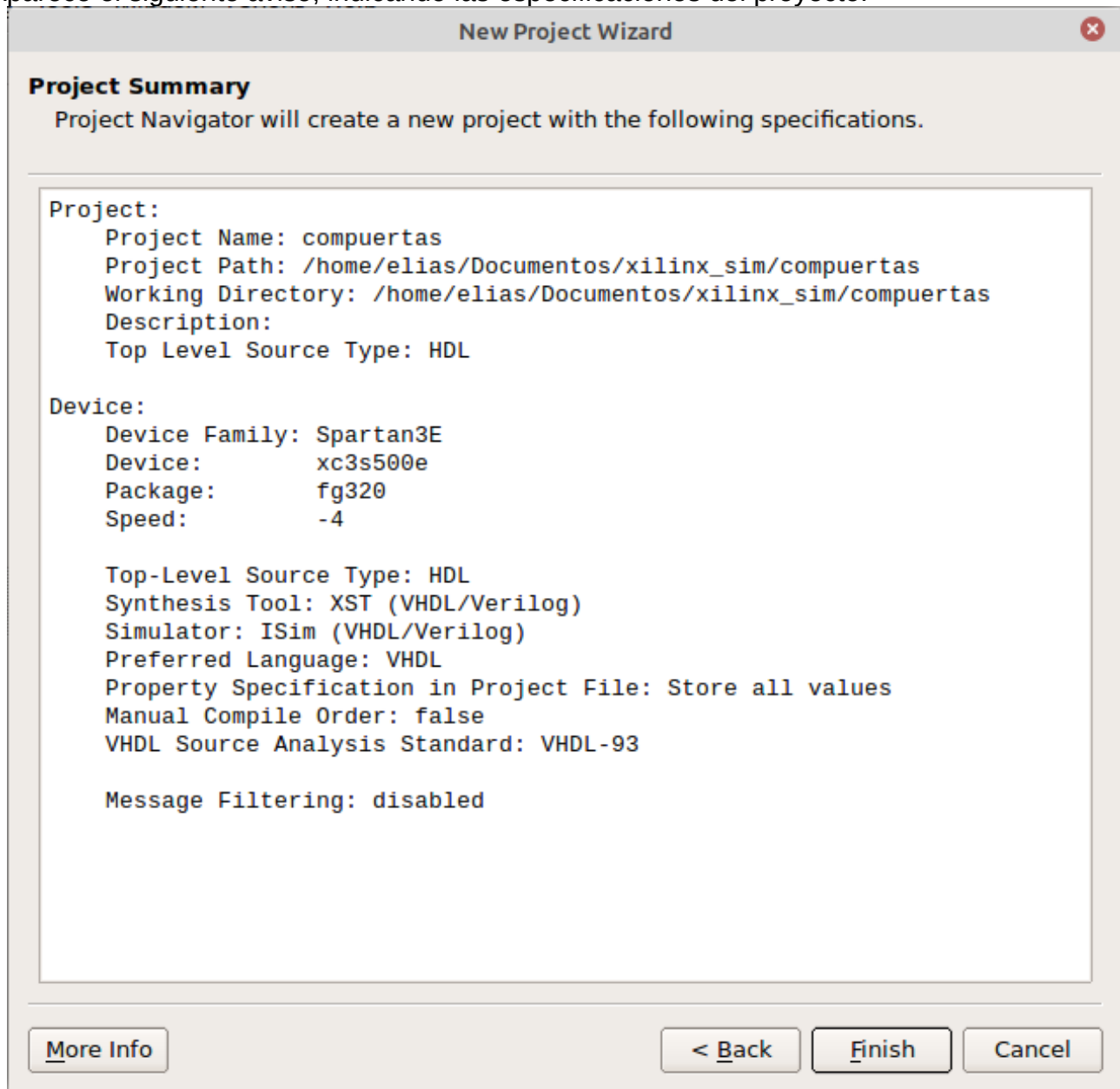
Select the device and design flow for the project

Property Name	Value
Evaluation Development Board	None Specified
Product Category	All
Family	Spartan3E
Device	XC3S500E
Package	FG320
Speed	-4
Top-Level Source Type	HDL
Synthesis Tool	XST (VHDL/Verilog)
Simulator	ISim (VHDL/Verilog)
Preferred Language	VHDL
Property Specification in Project File	Store all values
Manual Compile Order	<input type="checkbox"/>
VHDL Source Analysis Standard	VHDL-93
Enable Message Filtering	<input type="checkbox"/>

[More Info](#) [< Back](#) [Next >](#) [Cancel](#)

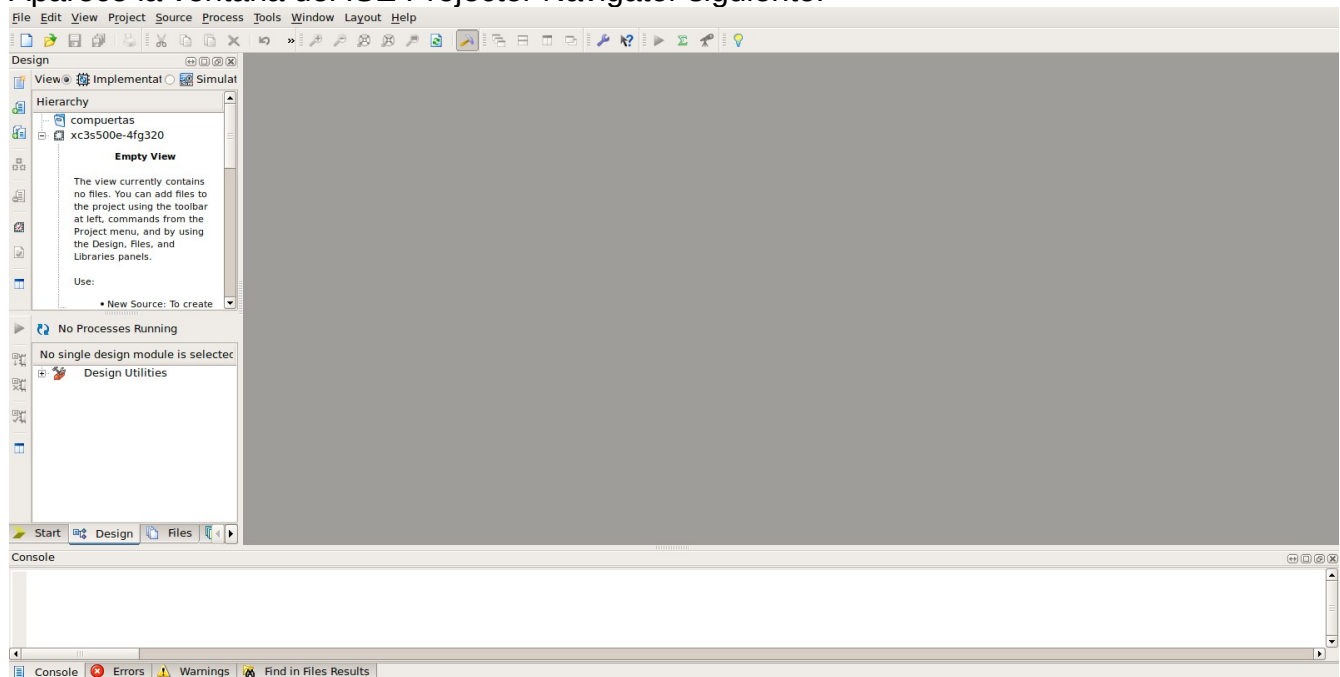
Click en Next >.

Aparece el siguiente aviso, indicando las especificaciones del proyecto.



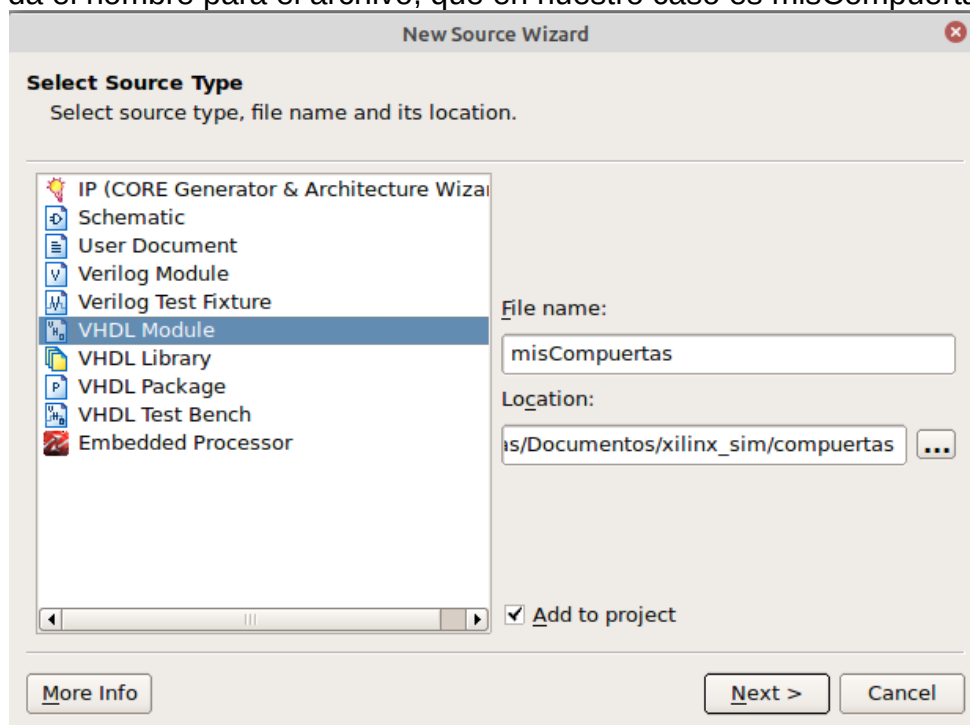
Click en el botón Finish.

Aparece la ventana del ISE Projector Navigator siguiente:



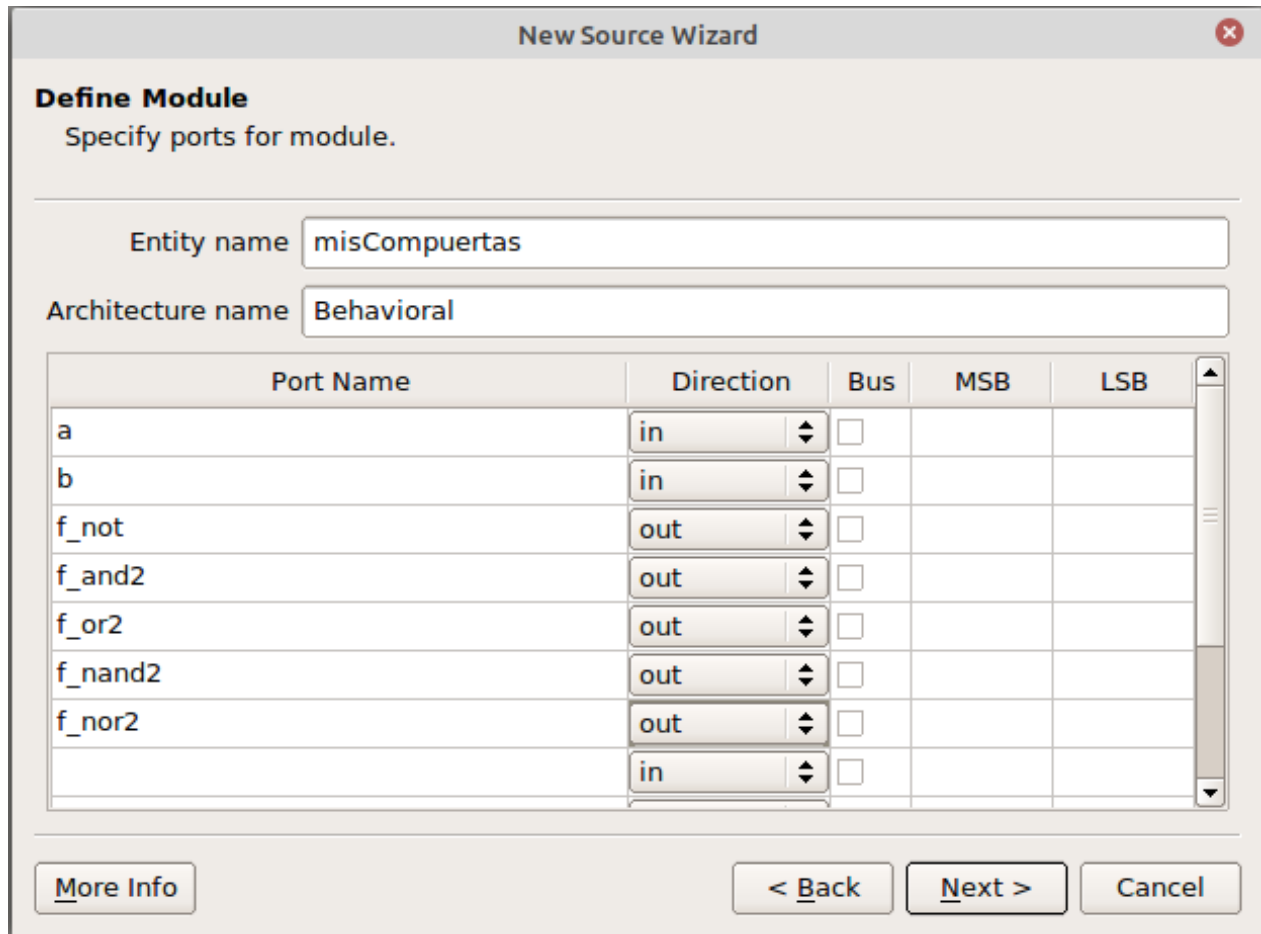
Para describir el diseño de nuestro circuito digital, agregaremos al proyecto un archivo VHDL nuevo; para lo cual, en el panel Hierarchy, seleccionamos xc3s500e-4fg320, damos sobre él click derecho y elegimos New Source...

y aparece el diálogo siguiente, donde seleccionamos el tipo de fuente: VHDL Module. También se da el nombre para el archivo, que en nuestro caso es misCompuertas.



Damos click en Next >.

Aparece el diálogo para definir las entradas y salidas del diseño que queremos hacer. Ingrese la información como se muestra a continuación:

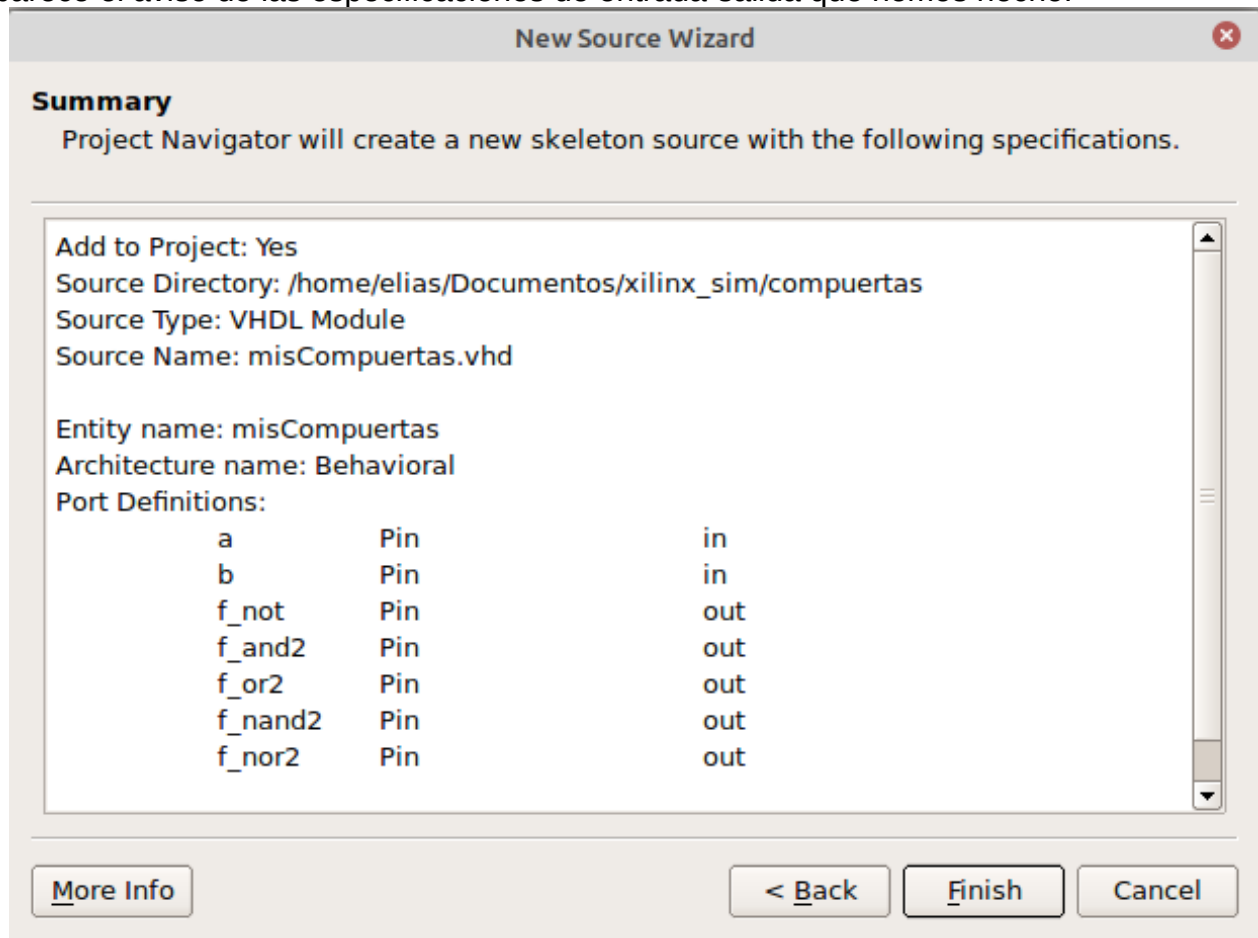


The image shows a 'New Source Wizard' dialog box with the title 'Define Module' and the instruction 'Specify ports for module.' The 'Entity name' field contains 'misCompuertas' and the 'Architecture name' field contains 'Behavioral'. Below these fields is a table for defining ports. The table has five columns: 'Port Name', 'Direction', 'Bus', 'MSB', and 'LSB'. There are eight rows in the table. The first seven rows have port names 'a', 'b', 'f_not', 'f_and2', 'f_or2', 'f_nand2', and 'f_nor2' respectively. The 'Direction' column for 'a' and 'b' is 'in', and for the others it is 'out'. The 'Bus' column has checkboxes that are all unchecked. The 'MSB' and 'LSB' columns are empty. The eighth row is empty and has 'in' in the 'Direction' column. At the bottom of the dialog are four buttons: 'More Info', '< Back', 'Next >', and 'Cancel'.

Port Name	Direction	Bus	MSB	LSB
a	in	<input type="checkbox"/>		
b	in	<input type="checkbox"/>		
f_not	out	<input type="checkbox"/>		
f_and2	out	<input type="checkbox"/>		
f_or2	out	<input type="checkbox"/>		
f_nand2	out	<input type="checkbox"/>		
f_nor2	out	<input type="checkbox"/>		
	in	<input type="checkbox"/>		

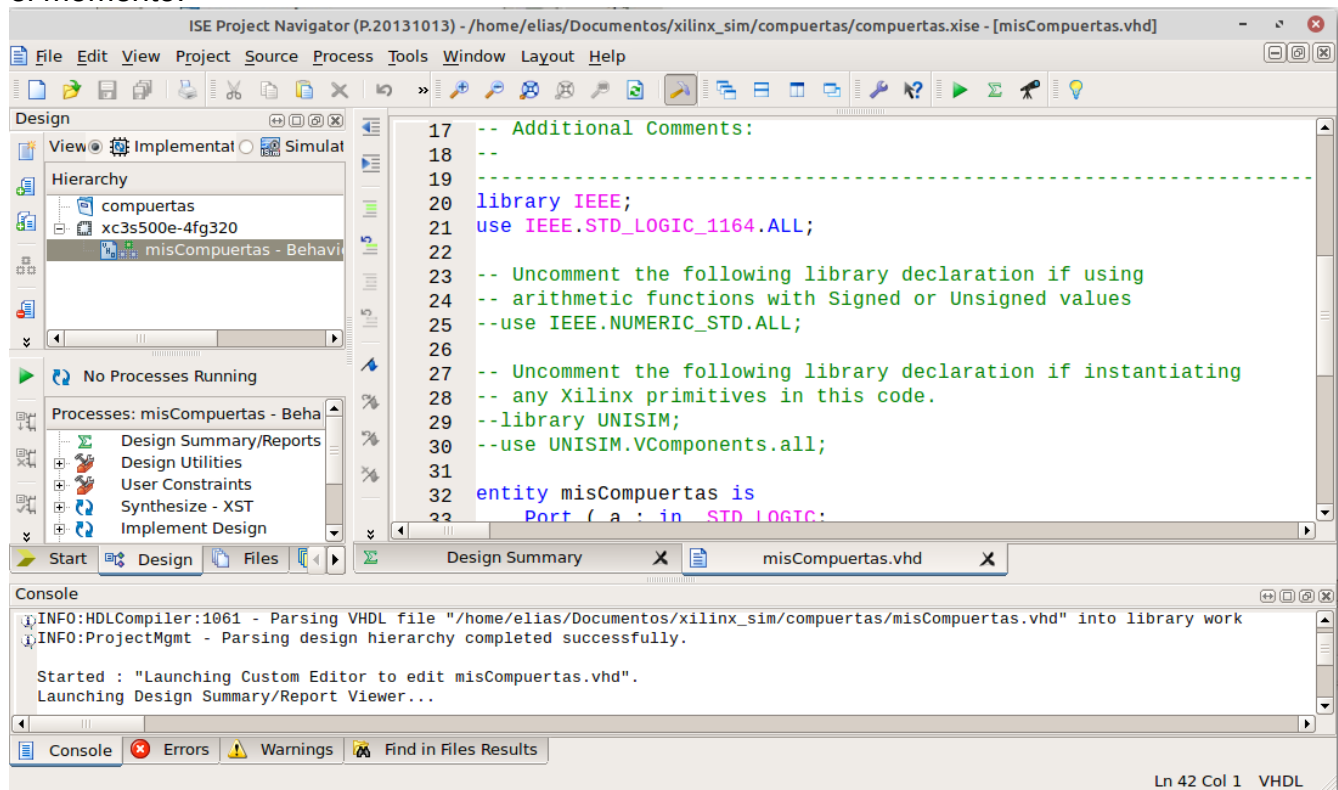
Dar click en Next >.

Aparece el aviso de las especificaciones de entrada-salida que hemos hecho.



Click en Finish.

En el ISE Project Navigator se muestra el archivo VHDL con la información necesaria hasta el momento.



Lo que falta introducir es la parte de la arquitectura del diseño:

architecture Behavioral of misCompuertas is

begin

f_not <= not a;

f_and2 <= (a and b);

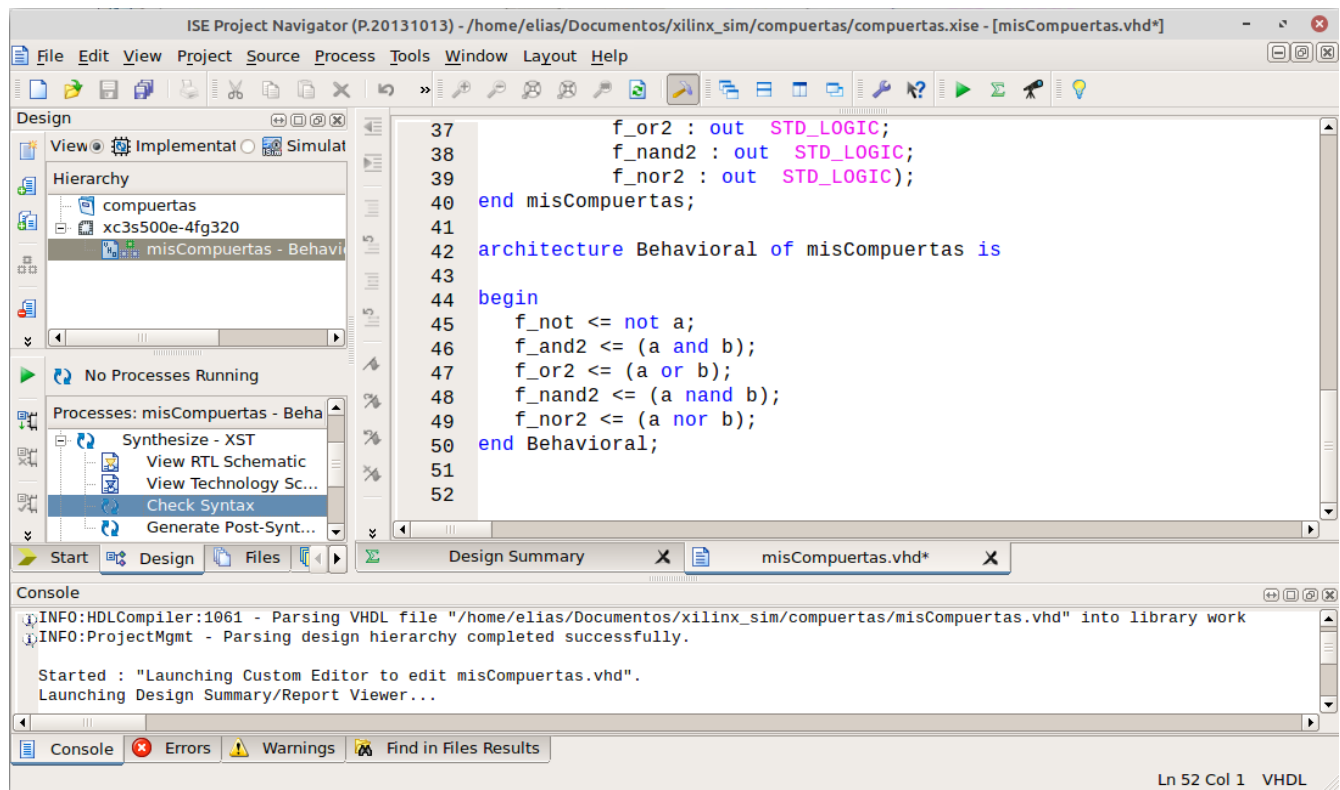
f_or2 <= (a or b);

f_nand2 <= (a nand b);

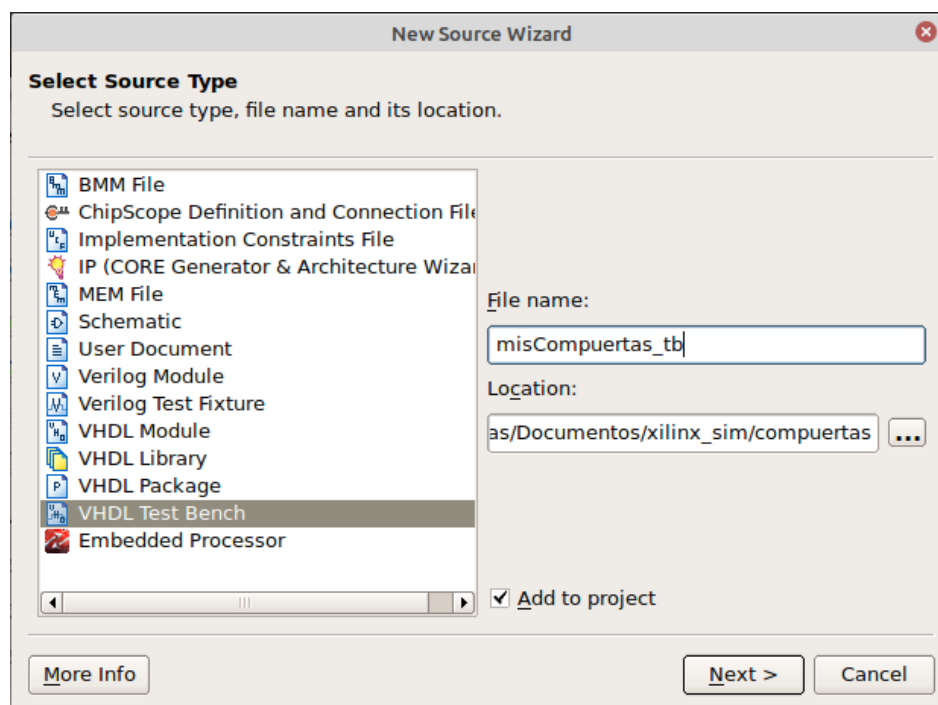
f_nor2 <= (a nor b);

end Behavioral;

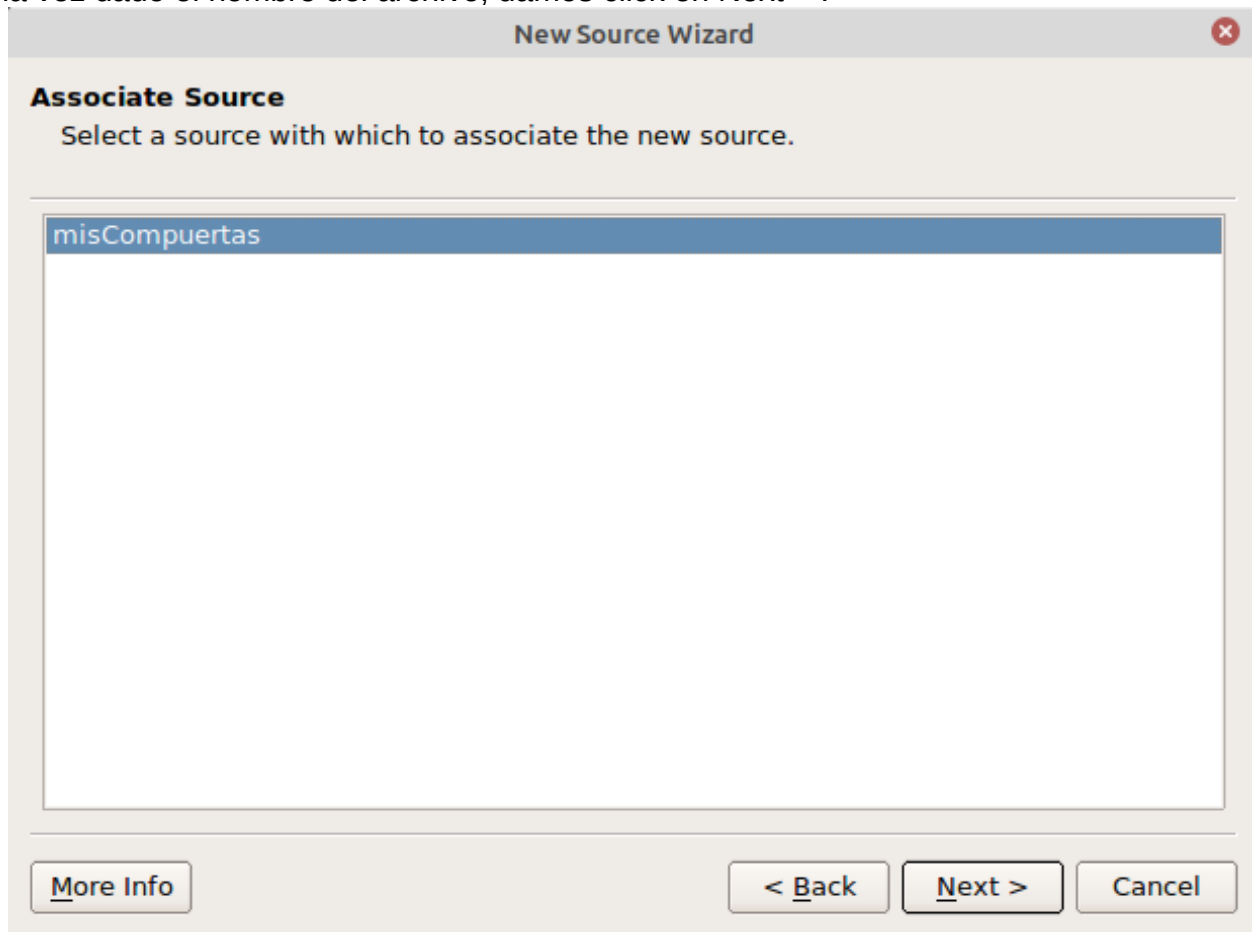
Salvar el archivo y checar la sintaxis seleccionando Synthesize – XST del panel Processes y dando doble click en Check Syntax.



Para hacer la simulación es necesario agregar un archivo más al proyecto, procedemos de manera similar a como agregamos el archivo VHDL; seleccionamos xc3s500e-4fg320, damos sobre él click derecho y elegimos New Source... y aparece el diálogo siguiente, donde seleccionamos el tipo de fuente: VHDL Test Bench.

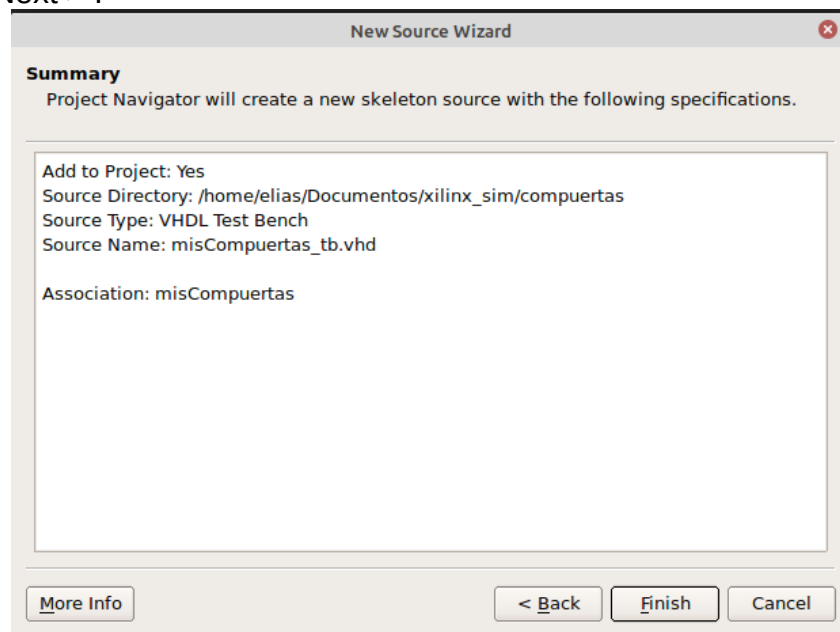


Una vez dado el nombre del archivo, damos click en Next > .

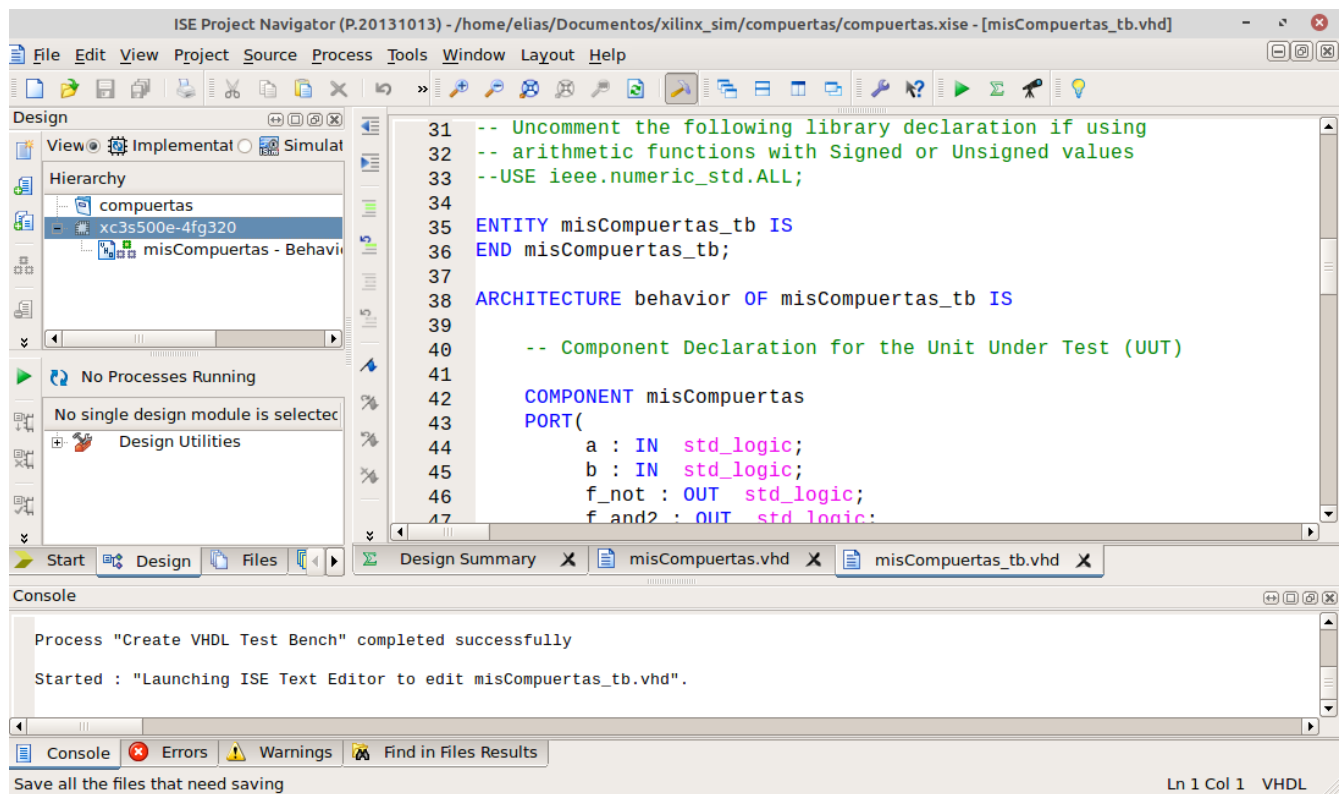


y aparece ésta ventana donse asociamos el archivo Test Bench con el archivo principal.

Damos click en Next > .



Damos click en Finish.



Ahora podemos proceder a editar directamente el archivo para simulación.

Como el sistema es combinacional, se deberá eliminar o poner en comentario todo lo concerniente a la señal de reloj.

```
66 -- appropriate port name
67
68 -- constant <clock>_period : time := 10 ns;
69
```

```
83 -- Clock process definitions
84 - <clock>_process :process
85 - begin
86 -     <clock> <= '0';
87 -     wait for <clock>_period/2;
88 -     <clock> <= '1';
89 -     wait for <clock>_period/2;
90 - end process;
```

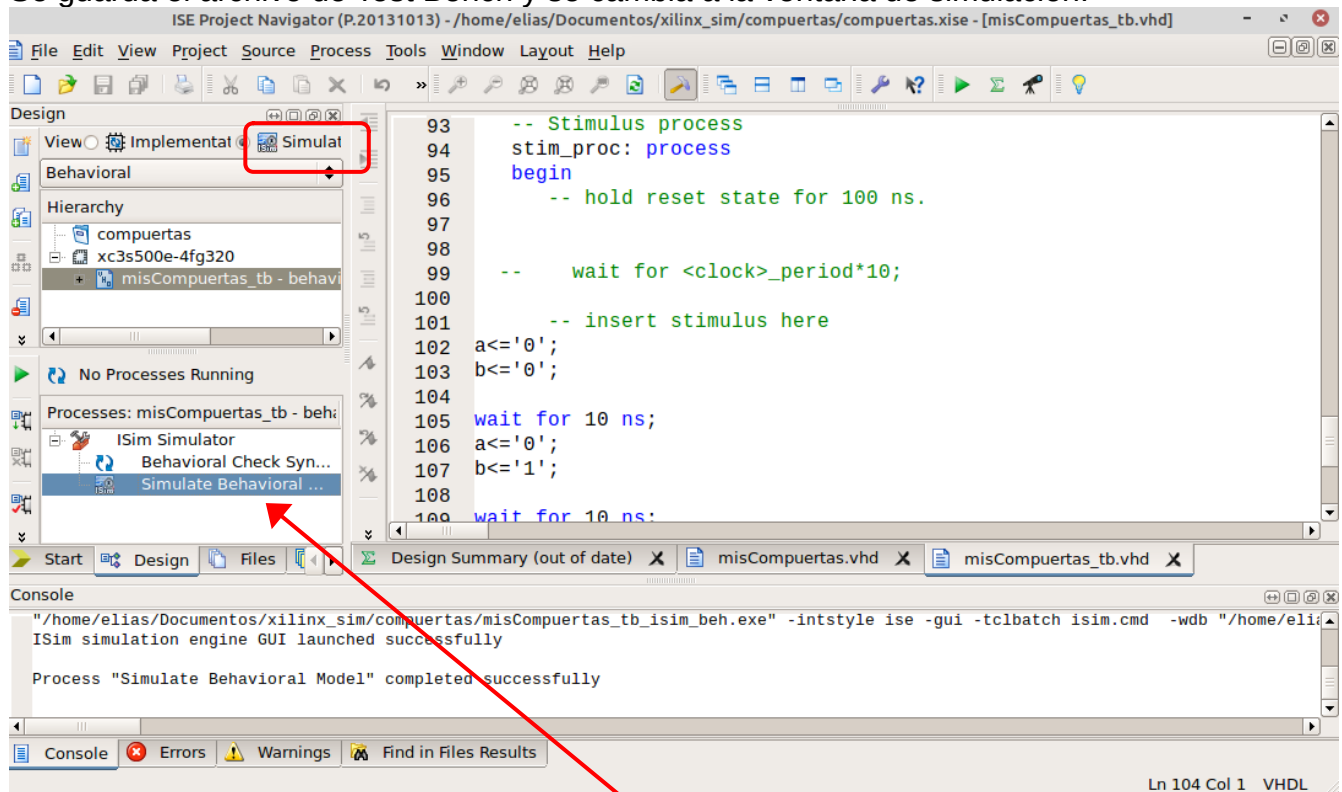
```
93  -- Stimulus process
94  stim_proc: process
95  begin
96      -- hold reset state for 100 ns.
97      wait for 100 ns;
98  |
99      -- wait for <clock>_period*10;
100
101      -- insert stimulus here
102
103      wait;
104  end process;
```

Se escribe los valores de las entradas tomando en cuenta todas las combinaciones:

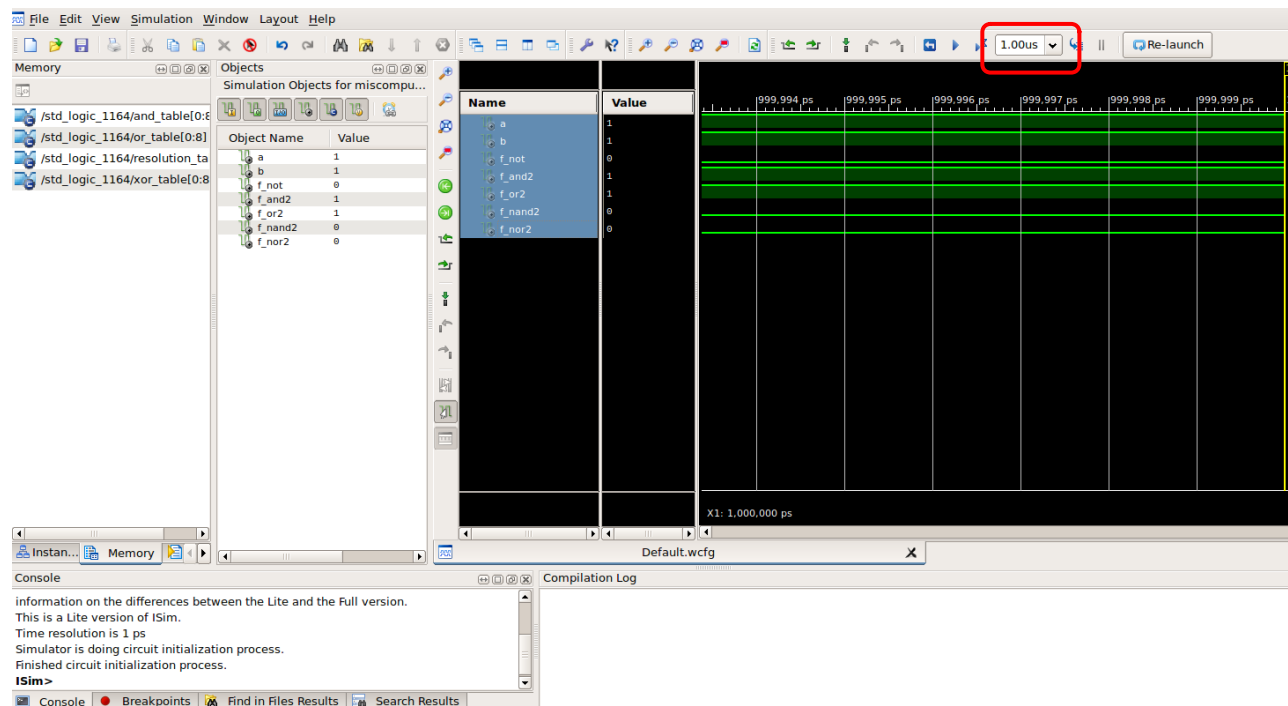
```
93  -- Stimulus process
94  stim_proc: process
95  begin
96      -- hold reset state for 100 ns.
97
98
99      -- wait for <clock>_period*10;
100
101      -- insert stimulus here
102      a<='0';
103      b<='0';
104
105      wait for 10 ns;
106      a<='0';
107      b<='1';
108
109      wait for 10 ns;
110      a<='1';
111      b<='0';
112
113      wait for 10 ns;
114      a<='1';
115      b<='1';
116      wait;
117  end process;
```

El tiempo de espera puede ser de cualquier valor.

Se guarda el archivo de Test Bench y se cambia a la ventana de simulación:

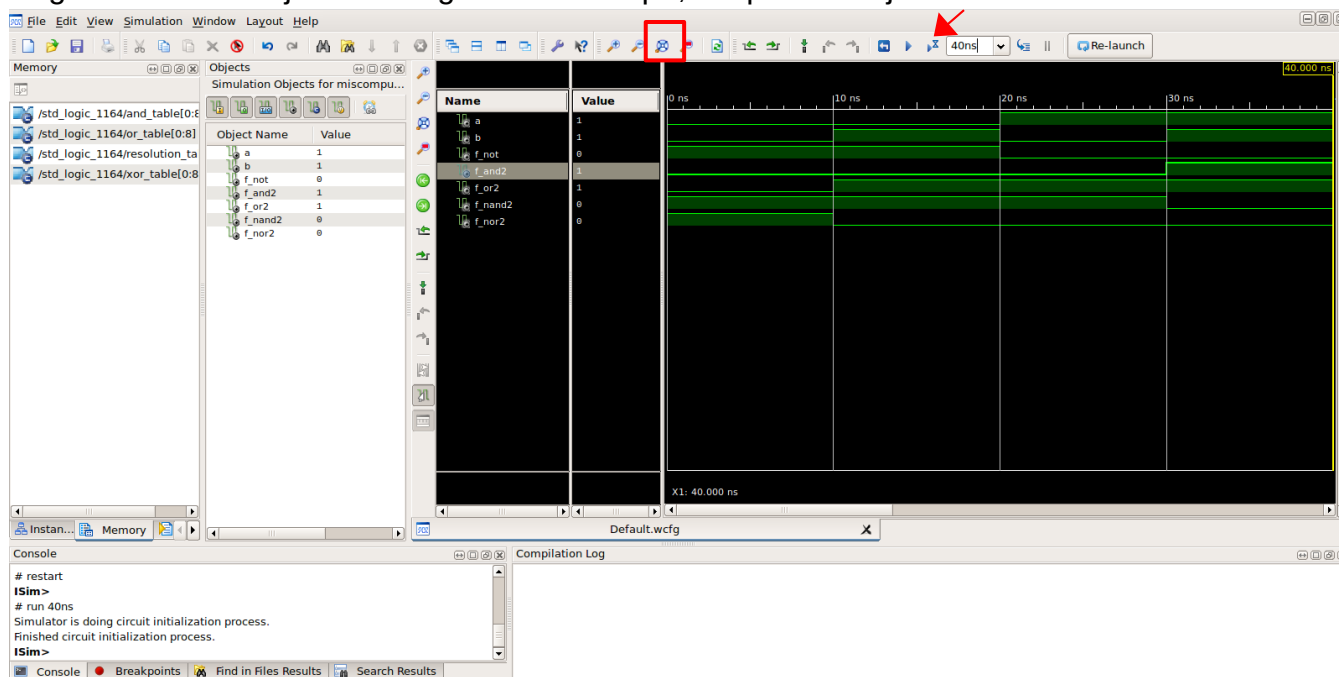


y se selecciona la simulación comportamental. Se hace doble clic.



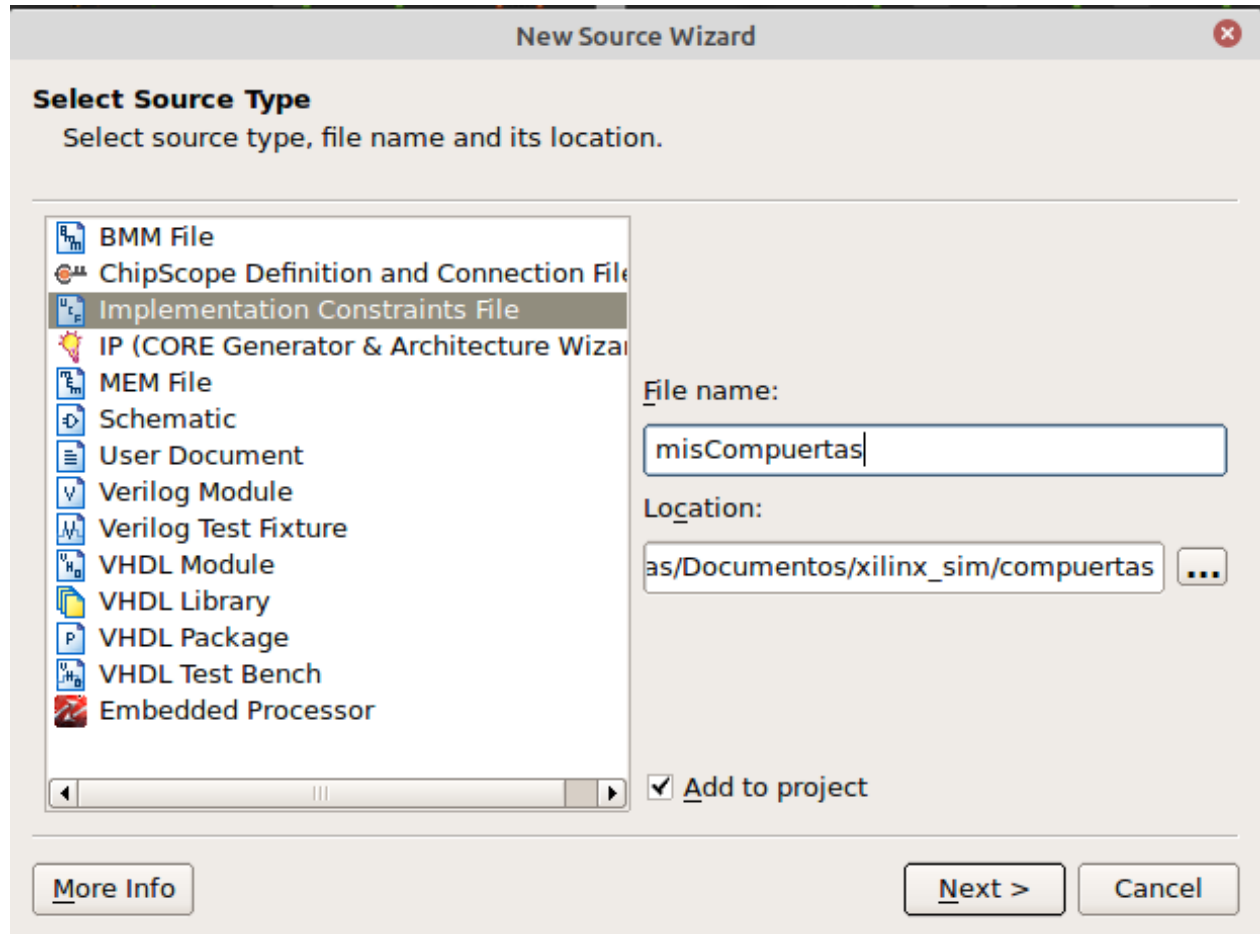
Nos vamos al tiempo de simulación y se cambia al valor máximo de simulación la cual fue de 40 ns.

Seguidamente se ajusta el diagrama de tiempo, después de ejecutar

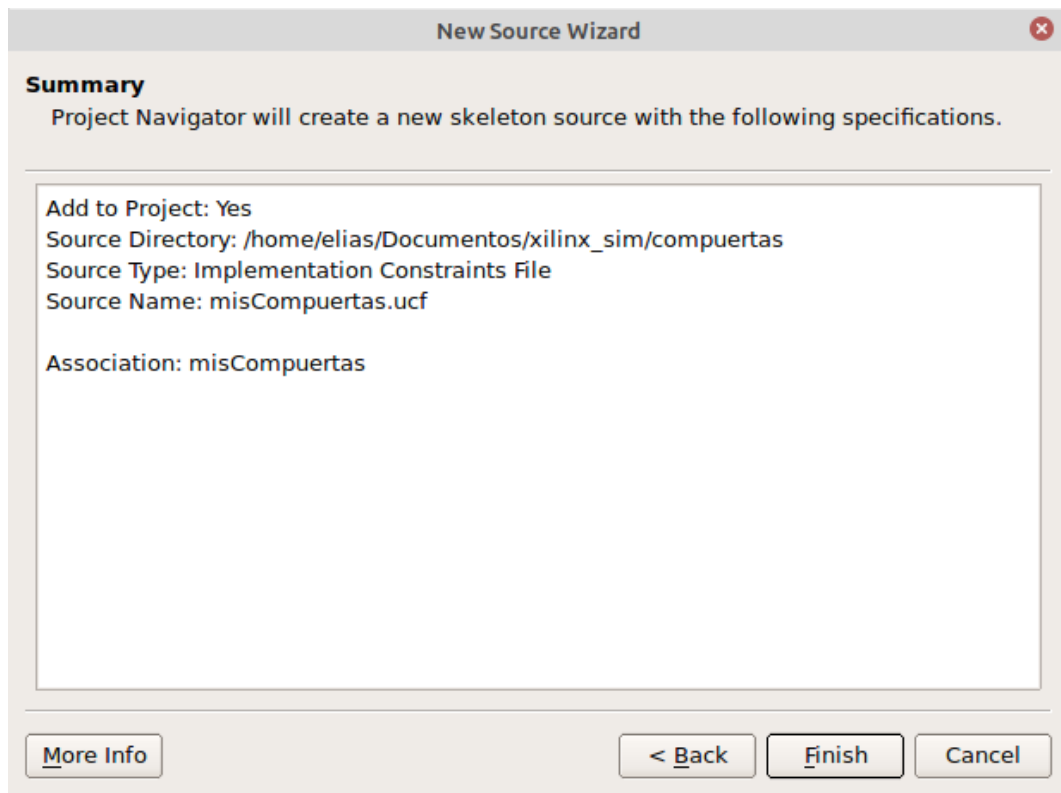


Para implementarlo en la tarjeta Nexys2. Para hacer esto es necesario agregar un archivo más al proyecto, procedemos de manera similar a como agregamos el archivo VHDL; seleccionamos xc3s500e-4fg320, damos sobre él click derecho y elegimos New Source... y aparece el diálogo siguiente, donde seleccionamos el tipo de fuente: Implementation Constraints File, que es un archivo "Universal Constraints File" (.ucf).

Aparece el diálogo siguiente, donde seleccionamos el tipo de fuente: Implementation Constraints File, que es un archivo “Universal Constraints File” (.ucf).

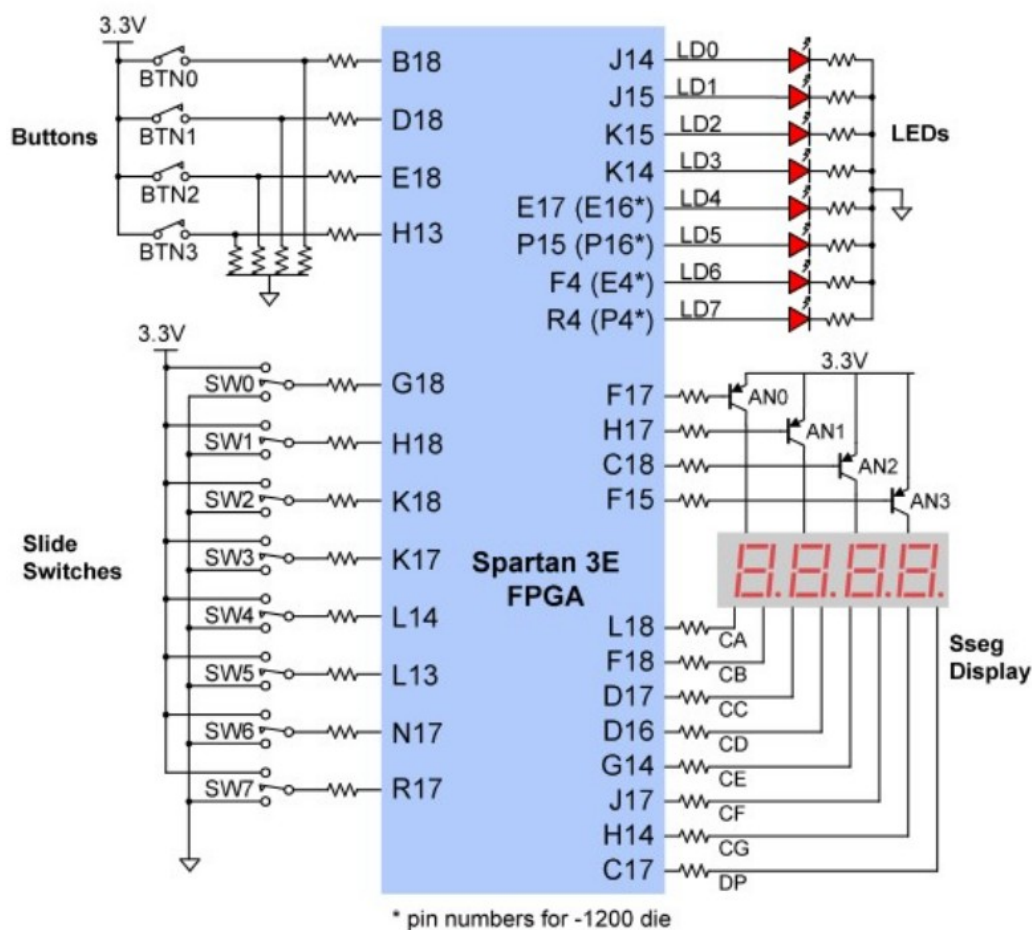


Una vez dado el nombre del archivo, damos click en Next > . Aparece el siguiente aviso:



Damos click en Finish.

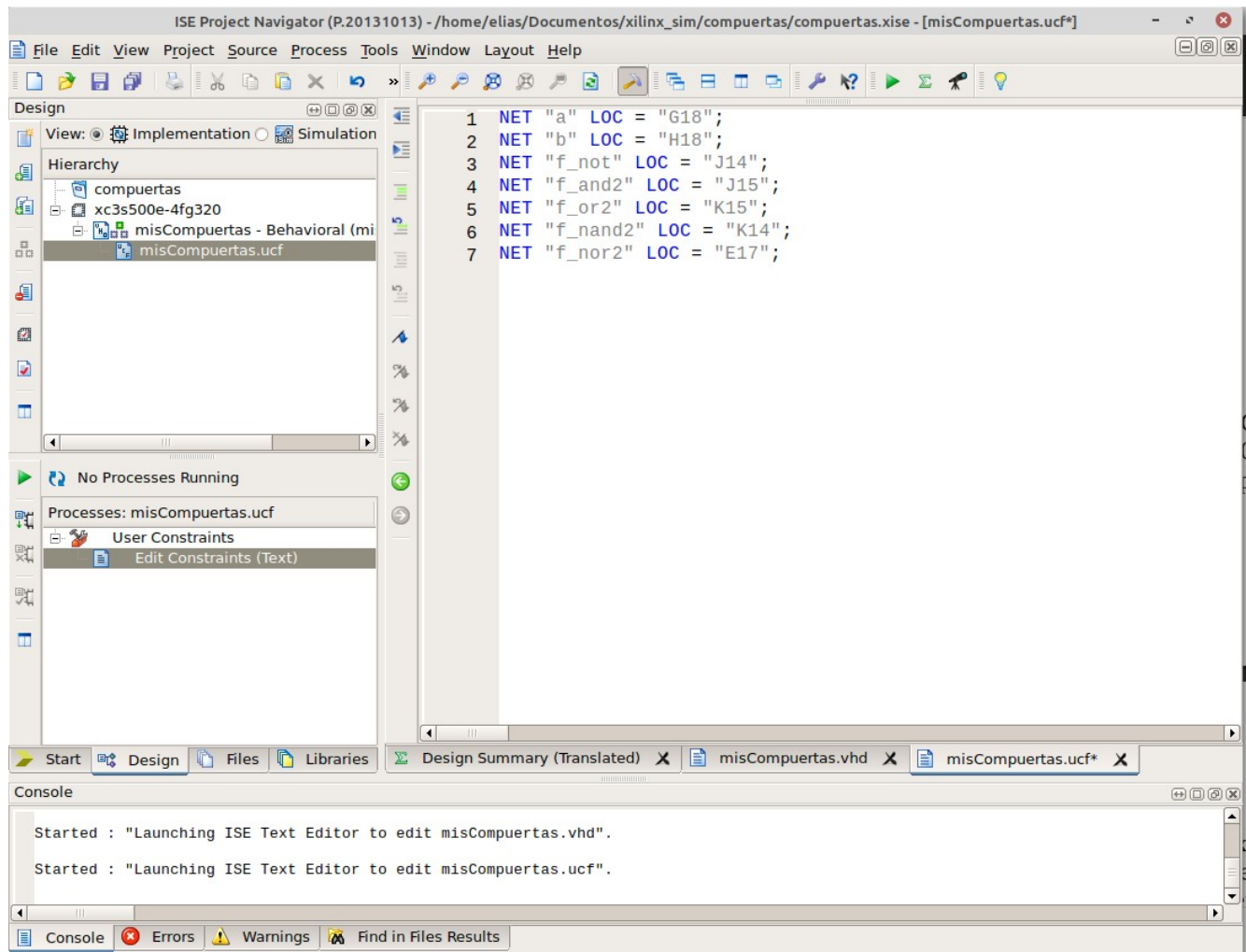
Ahora podemos proceder a editar directamente el archivo .ucf, teniendo en cuenta las conexiones de E/S que tenemos disponibles en la tarjeta Nexys2, las cuales pueden verse en el siguiente diagrama. También podemos “add copy” del archivo nexys2.ucf en donde están definidos todos los pines de E/S.



```

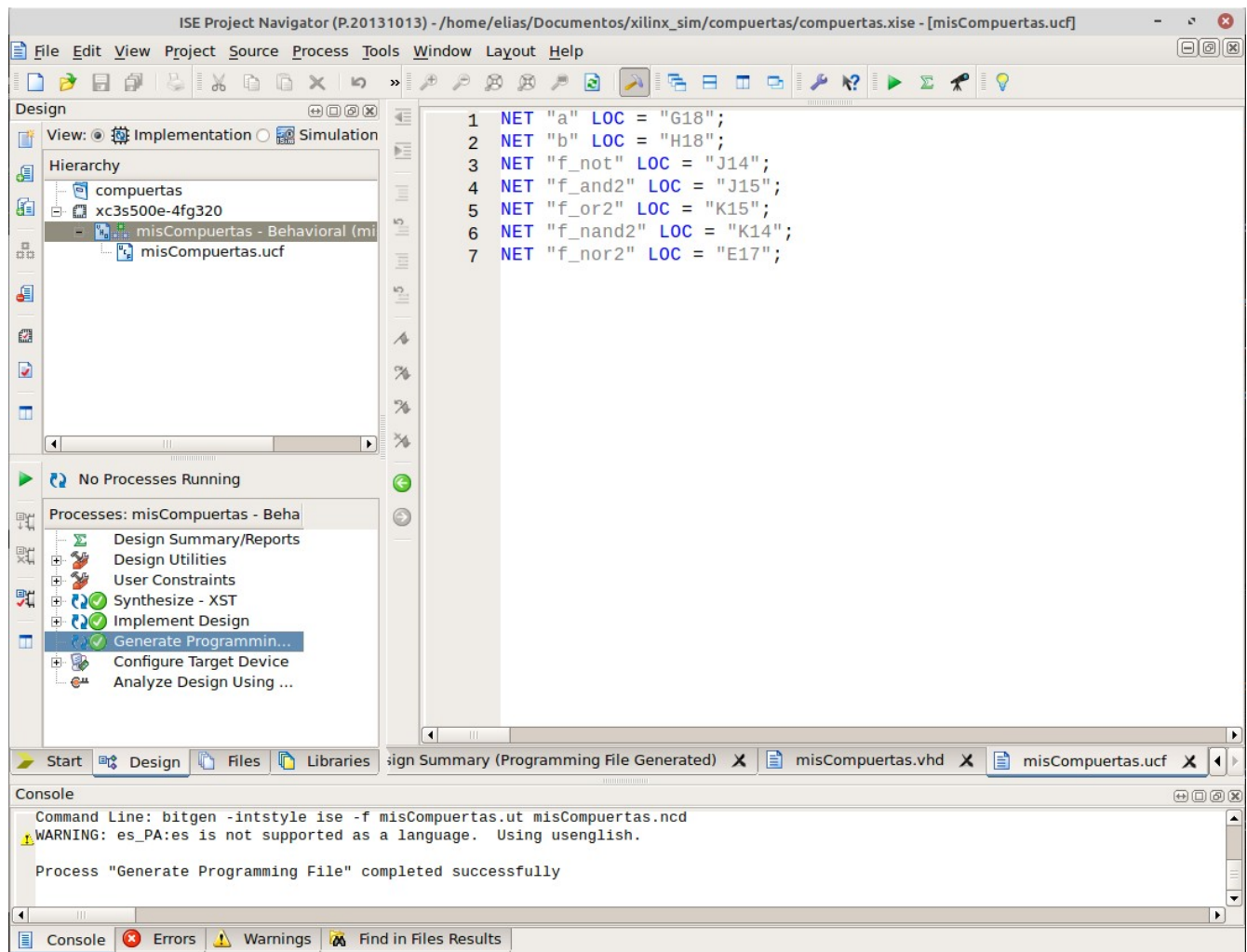
NET "a" LOC = "G18";
NET "b" LOC = "H18";
NET "f_not" LOC = "J14";
NET "f_and2" LOC = "J15";
NET "f_or2" LOC = "K15";
NET "f_nand2" LOC = "K14";
NET "f_nor2" LOC = "E17";

```

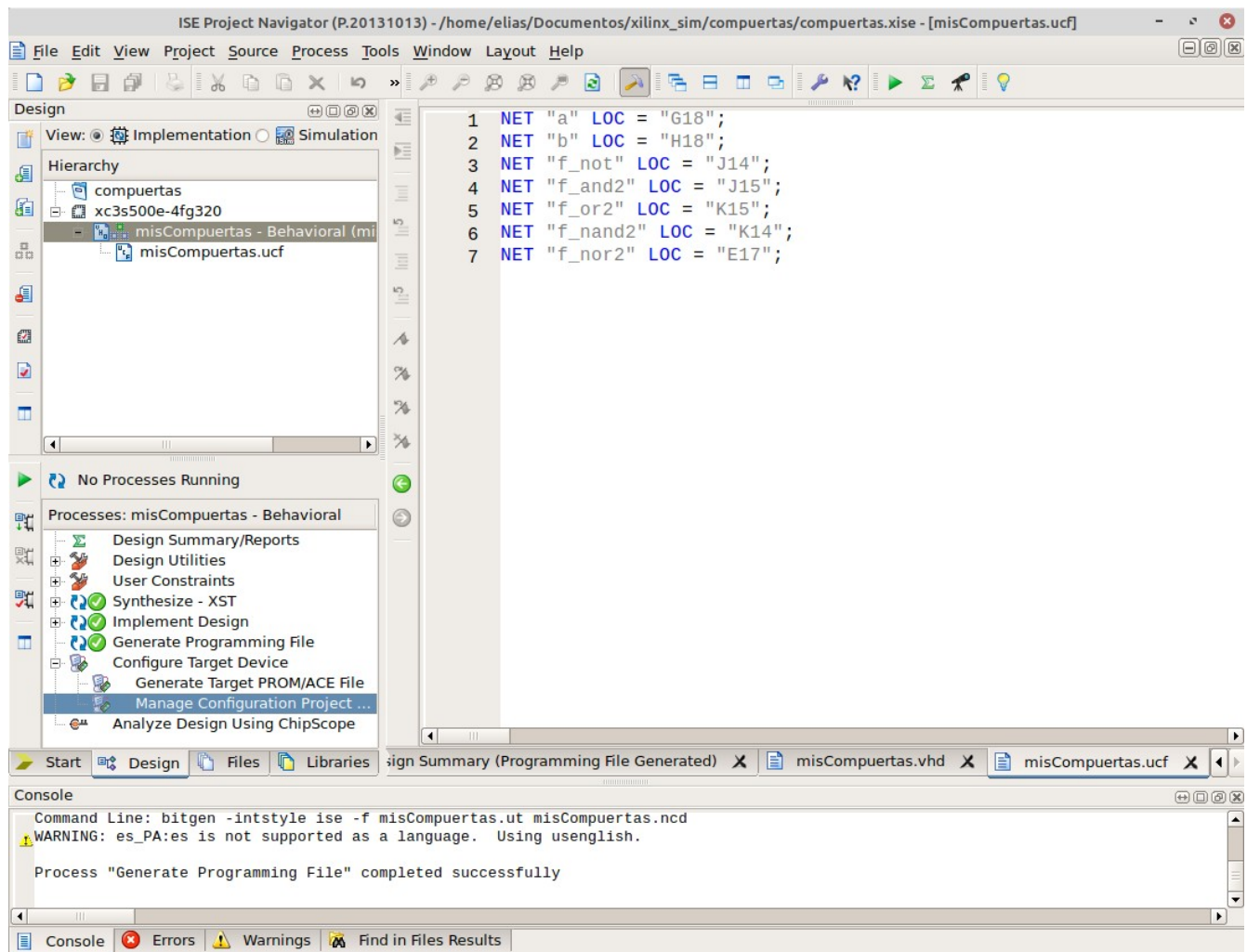


Guardamos el archivo de restricciones. Estamos listos para intentar generar el archivo para el FPGA, para ello seleccionamos el archivo VHDL misCompuertas, que esta justo abajo de xc3s500e-4fg320 en el panel Hierarchy, después en el panel Processes damos doble click en Generate Programming para generar el archivo de diseño que se debe enviar a la tarjeta Nexys2.

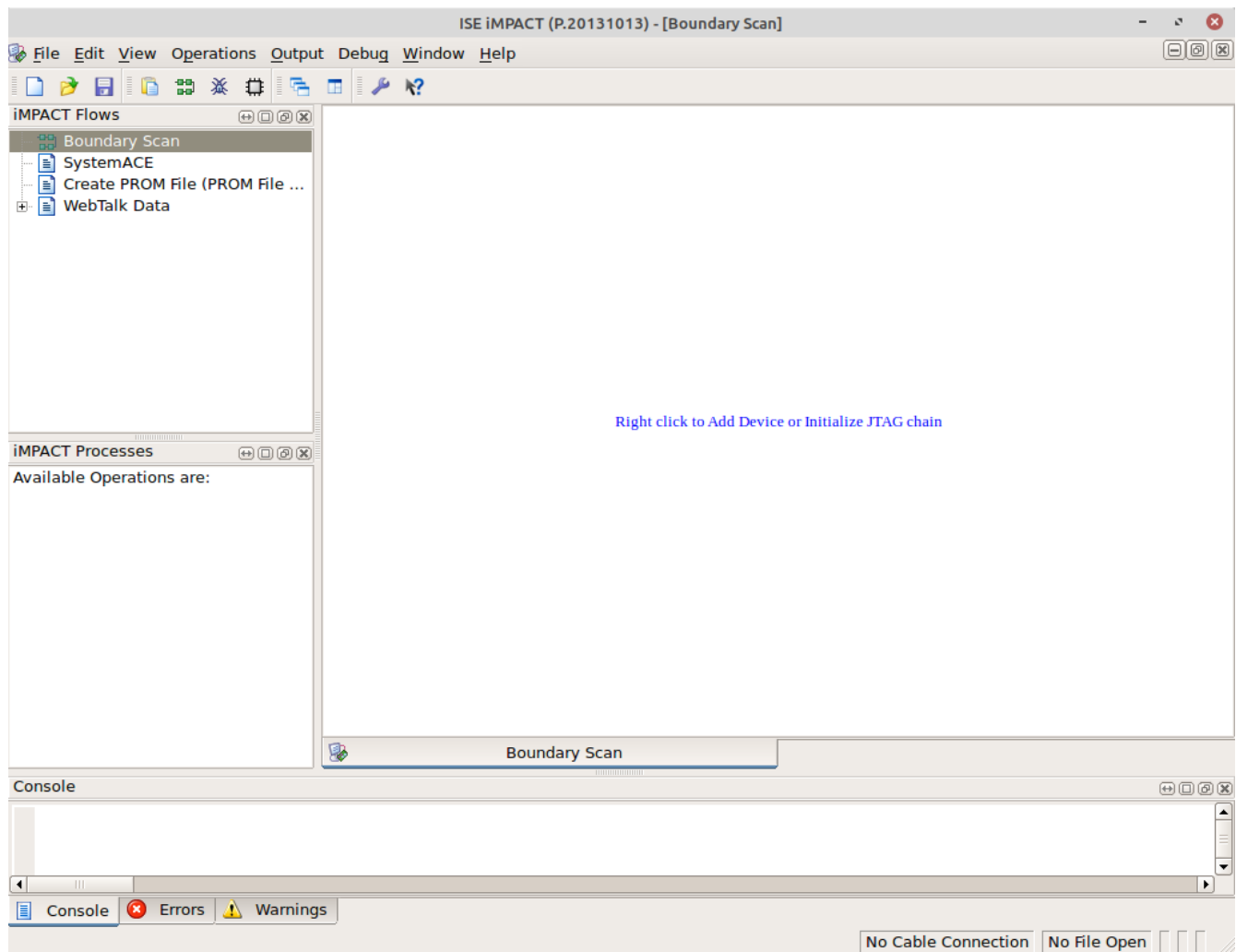
Opcionalmente, se puede avanzar por pasos, en lugar de generar el archivo para la Nexys2 directamente; primero se da doble click en Synthesize, después en Implement Design y finalmente en Generate Programming. Si no hay ningun error obtenemos en la consola el mensaje: Process "Generate Programming File" completed successfully.



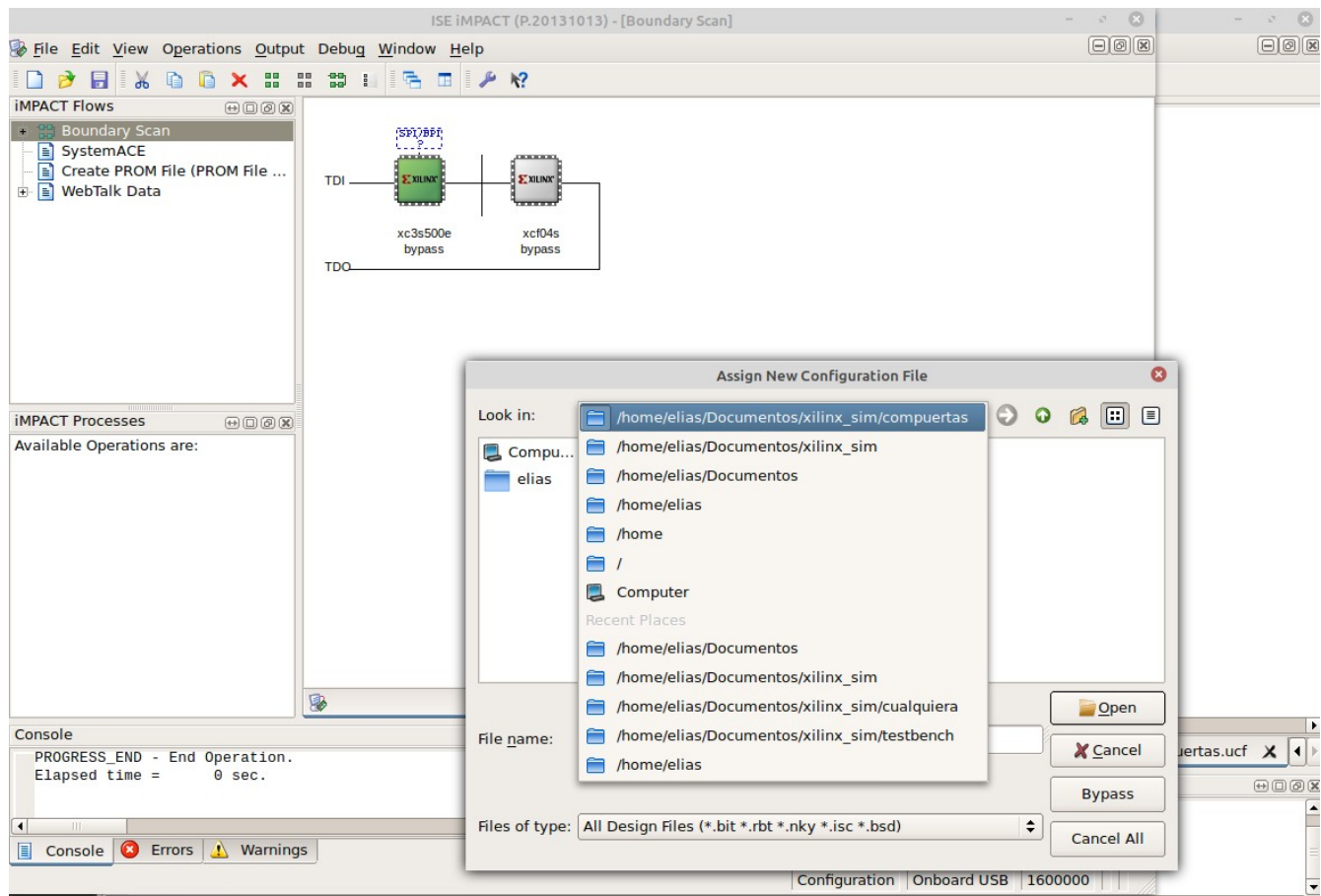
Seguidamente se selecciona el "Manage Configuration Project (iMPACT)":



Se abre la siguiente ventana:

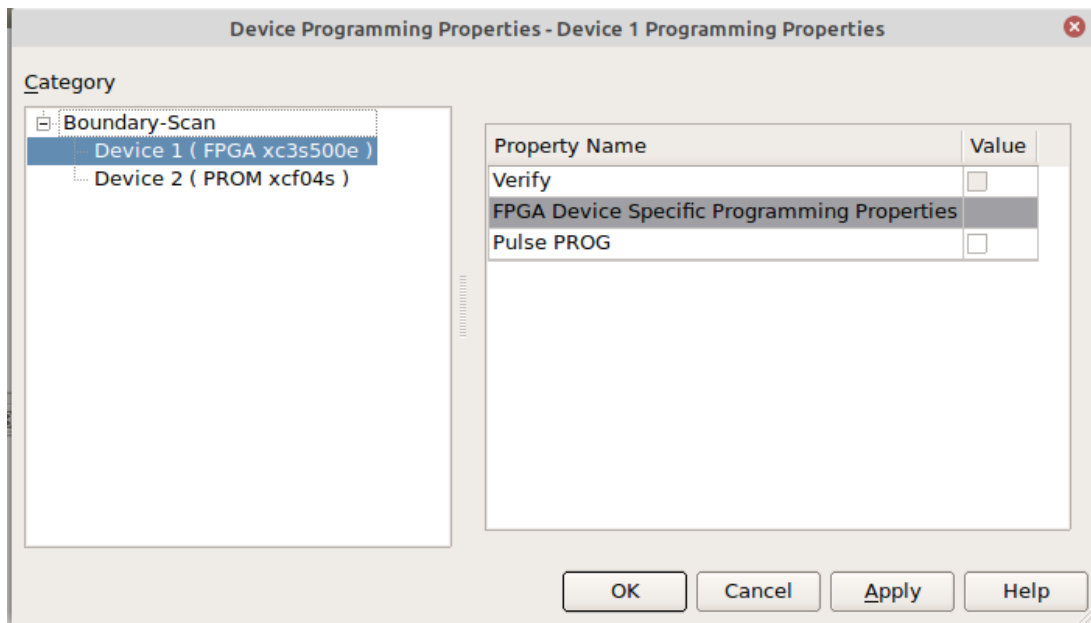


Se da doble clic en "Boundary Scan" y se sigue las instrucciones o se teclea CTRL+I.



Se selecciona el archivo .bit y se da "open" y se abrirá un cuadro de dialogo explicando que se puede guardar en la PROM, se selecciona NO. y luego Bypass en el siguiente cuadro.

Por último se selecciona OK.



Colocarse sobre el primer IC y dar botón derecho y seleccionar "Program"

