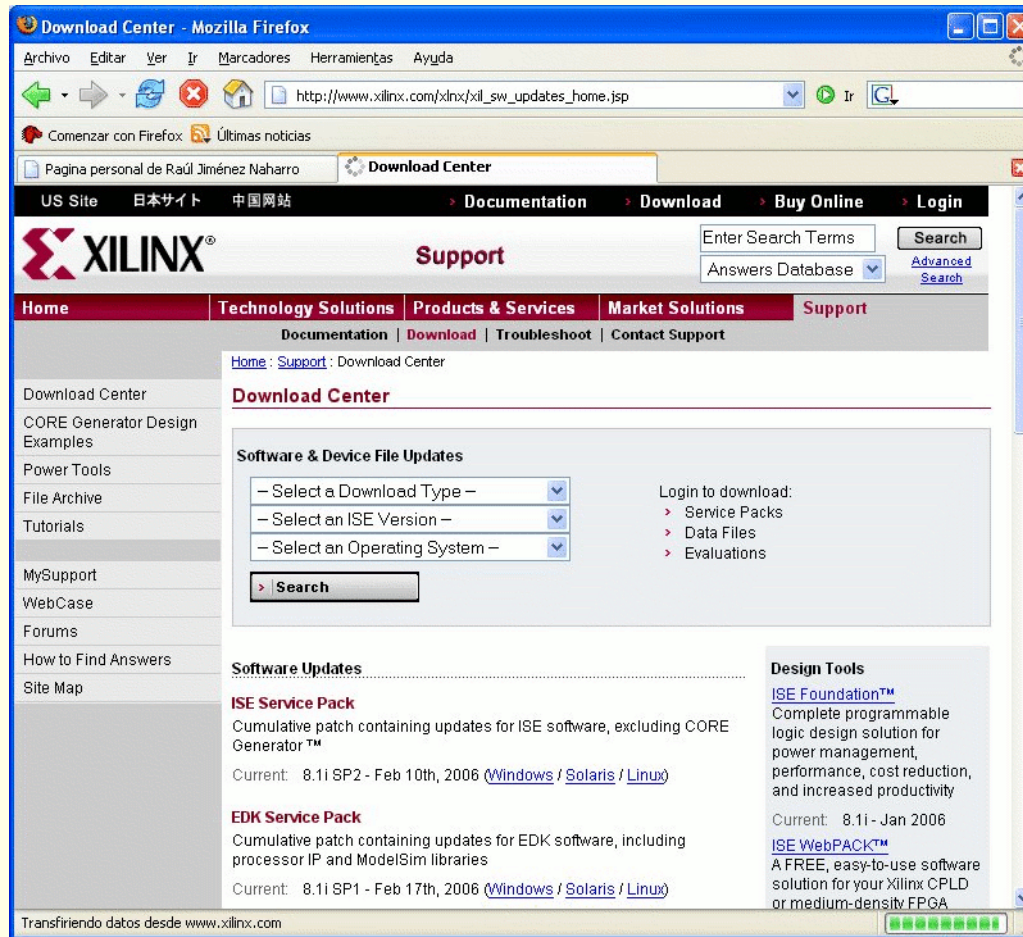




PRÁCTICA:

MANUAL DE XILINX

Introducción



Página web XILINX

<http://www.xilinx.com/>

Ir a la sección Download

Entrar en ISE WebPACK

Registrarse

Descargar WebPACK

Instalar

Tutoriales

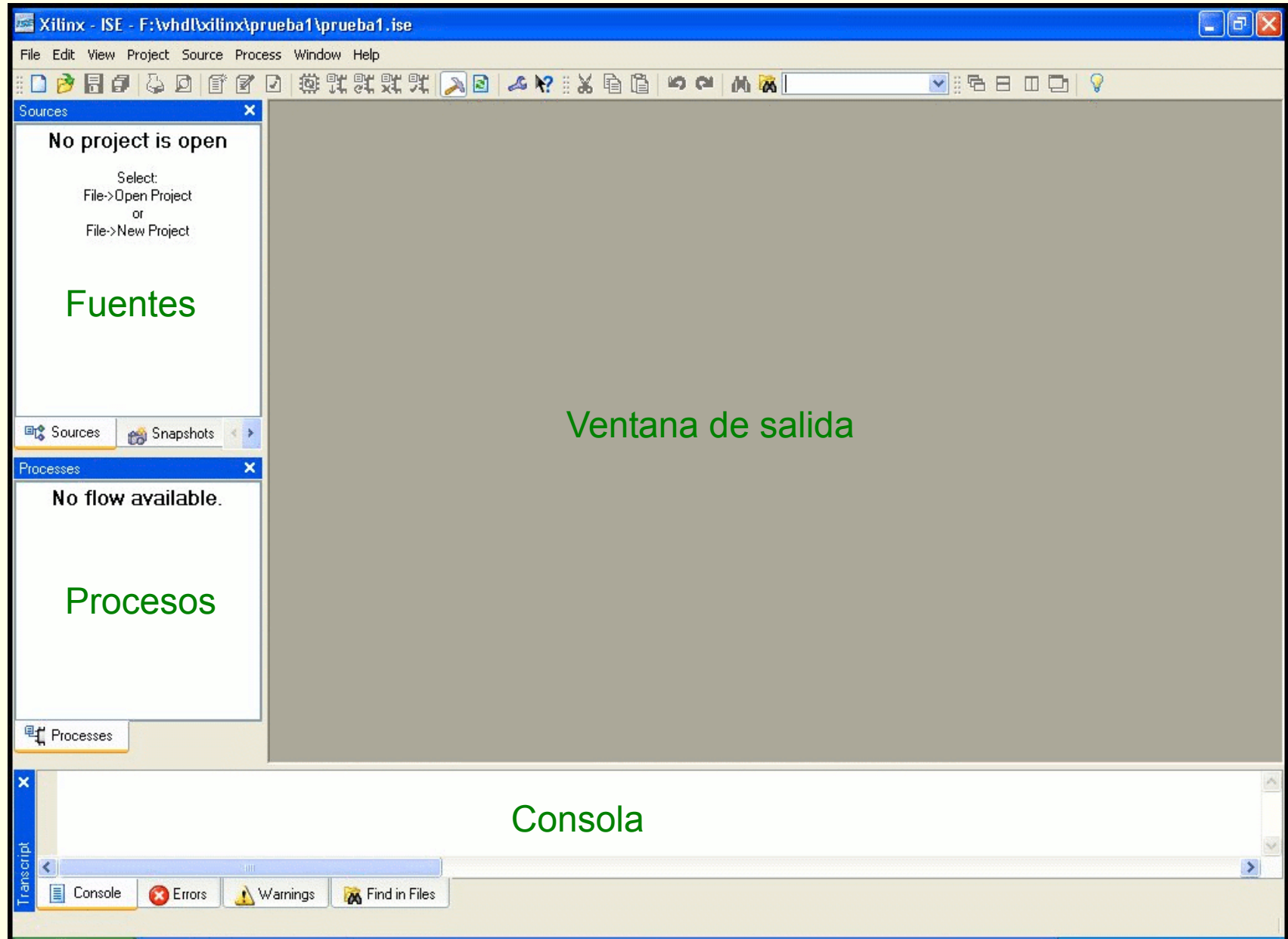
[http://support.xilinx.com/
support/techsup/tutorials/
index.htm](http://support.xilinx.com/support/techsup/tutorials/index.htm)

**Flujo de
diseño**



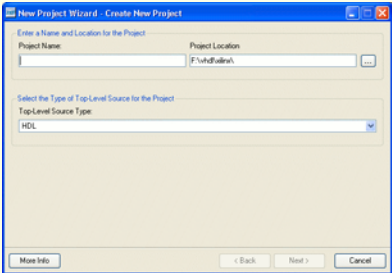
**Herramienta
XILINX**

Introducción

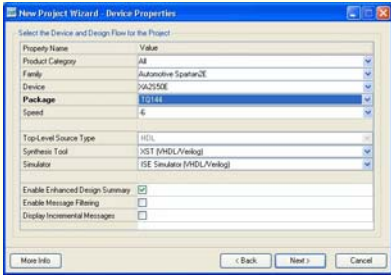


Nuevo proyecto

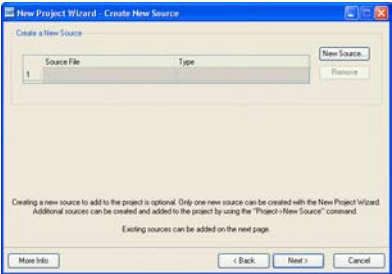
Nuevo Proyecto



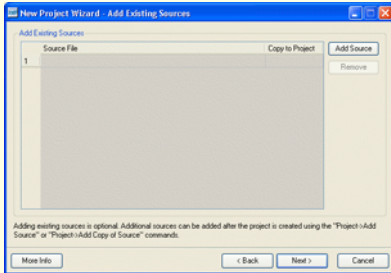
Nombre del proyecto



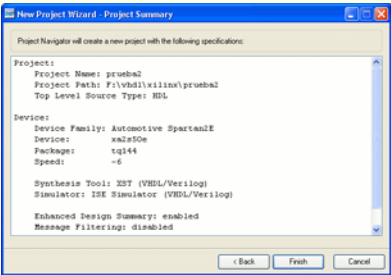
Datos del proyecto



Añadir nueva fuente



Añadir fuentes existentes



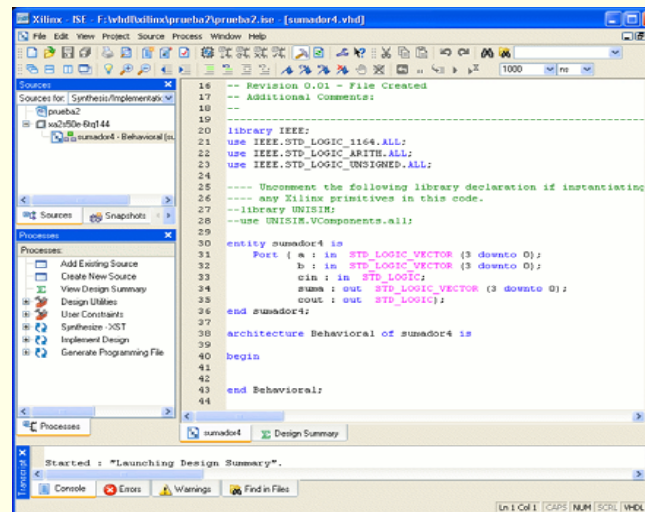
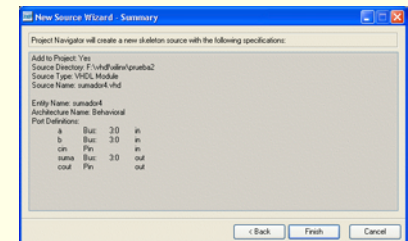
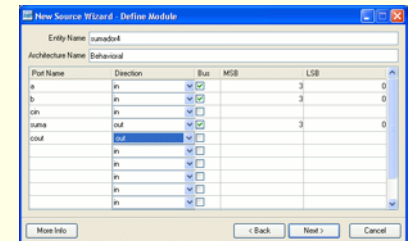
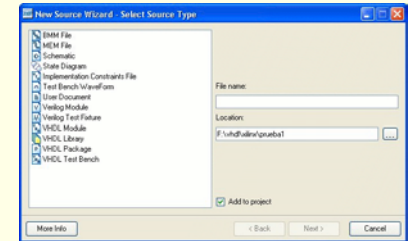
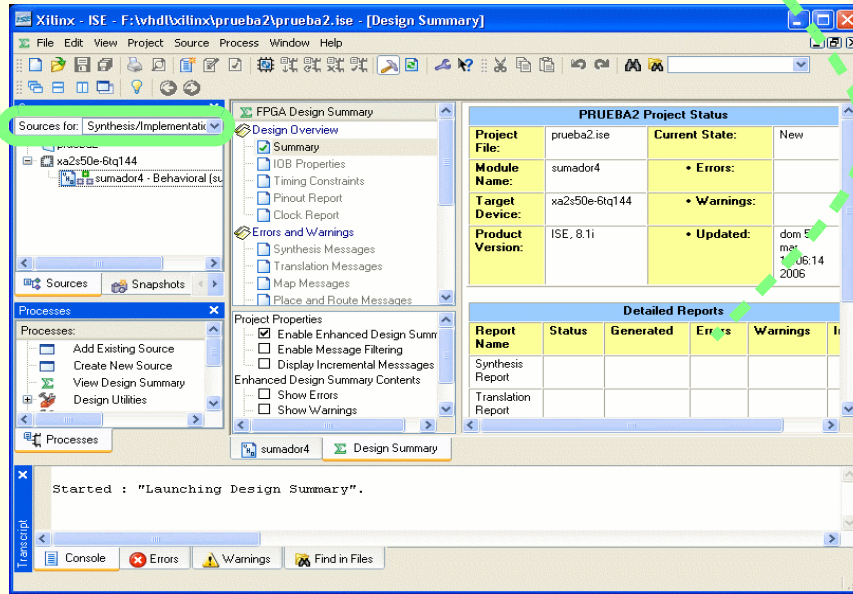
Resumen del proyecto

Nueva fuente

Tipos de procesos

Datos del proyecto

Nueva fuente



Herramienta XILINX

Código VHDL

Nueva fuente

Sources

Sources for: Behavioral Simulation

prueba2

xa2s50e-6tq144

sumador4 - Behavioral (suma)

Processes

Processes:

- Add Existing Source
- Create New Source
- Xilinx ISE Simulator
- Check Syntax
- Simulate Behavioral Mode

```
21 use IEEE.STD_LOGIC_1164.ALL;
22 use IEEE.STD_LOGIC_ARITH.ALL;
23 use IEEE.STD_LOGIC_UNSIGNED.ALL;
24
25 ---- Uncomment the following library declaration if instantiating
26 ---- any Xilinx primitives in this code.
27 --library UNISIM;
28 --use UNISIM.VComponents.all;
29
30 entity sumador4 is
31     Port ( a : in  STD_LOGIC_VECTOR (3 downto 0);
32           b : in  STD_LOGIC_VECTOR (3 downto 0);
33           cin : in  STD_LOGIC;
34           suma : out  STD_LOGIC_VECTOR (3 downto 0);
35           cout : out  STD_LOGIC);
36 end sumador4;
37
38 architecture Behavioral of sumador4 is
39     signal carry :std_logic_vector(4 downto 0);
40 begin
41     carry(0) <= cin;
42     F1:for i in a'range generate
43     begin
44         suma(i) <= a(i) xor b(i) xor carry(i);
45         carry(i+1) <= (carry(i) and (a(i) or b(i))) or (a(i) and b(i));
46     end generate;
47     cout <= carry(4);
48 end Behavioral;
49
```

sumador4 Design Summary

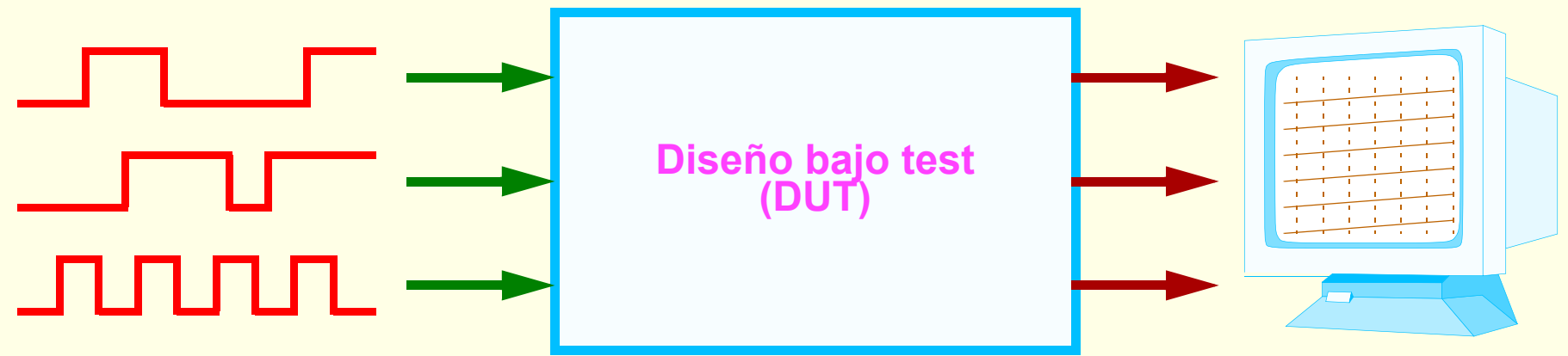
Transcript

Console Errors Warnings Find in Files

Ln 48 Col 1 CAPS NUM SCRL VHDL

Herramienta XILINX

Fichero de patrones (testbench)



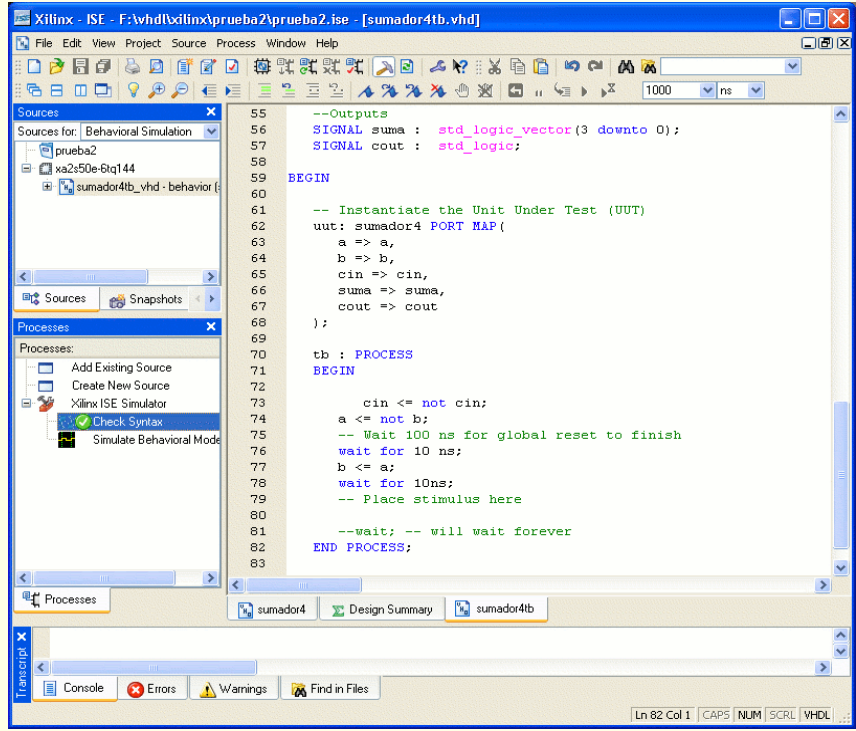
Nueva fuente de patrones (testbench)

This block shows the step-by-step process of creating a new testbench source in the Xilinx IDE:

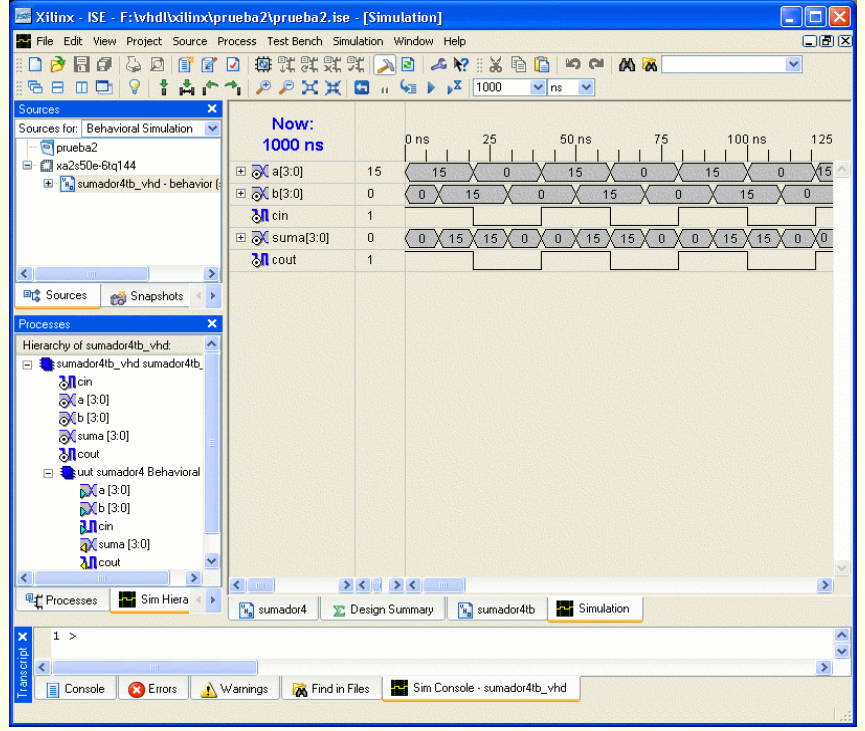
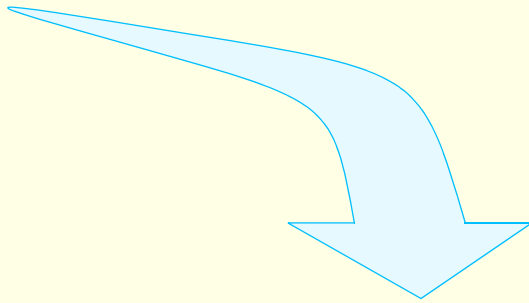
- New Source Wizard - Select Source Type:** The user selects 'VHDL Test Bench' from the list of source types. The file name is 'sumador415' and the location is 'F:\vhdl\win\prueba2'.
- New Source Wizard - Associate Source:** The user selects 'sumador4' as the source to associate with the new testbench.
- New Source Wizard - Summary:** The wizard summarizes the specifications: 'Add to Project: Yes', 'Source Directory: F:\vhdl\win\prueba2', 'Source Type: VHDL Test Bench', 'Source Name: sumador415.vhd', and 'Association: sumador4'.
- Xilinx - IS: F:\vhdl\win\prueba2\prueba2_1st [sumador415.vhd]:** The final screenshot shows the testbench code in the editor. The code includes signal declarations for 'a', 'b', 'sum', and 'cout', and a test process that instantiates the 'sumador' DUT, sets initial values, and waits for the simulation to finish.

Herramienta XILINX

Fichero de patrones (testbench)



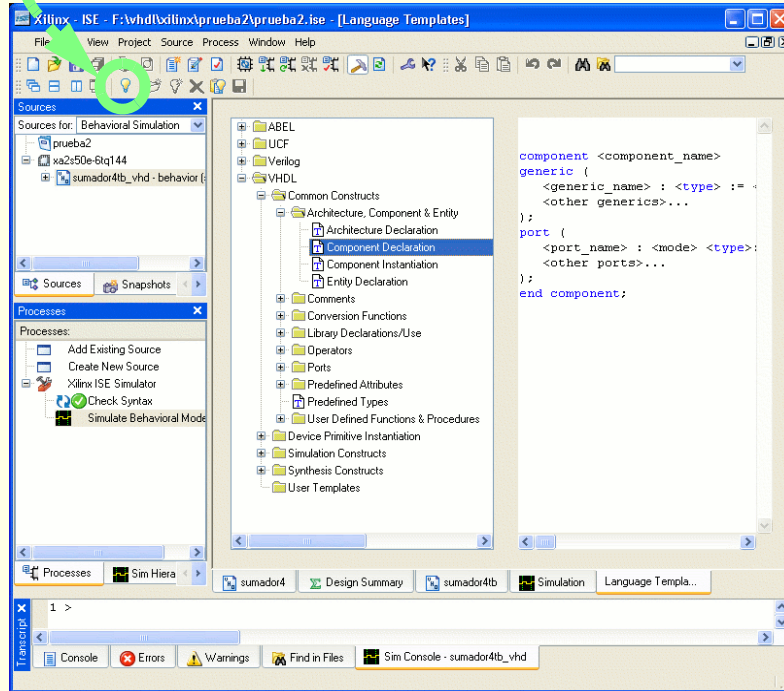
Simulación



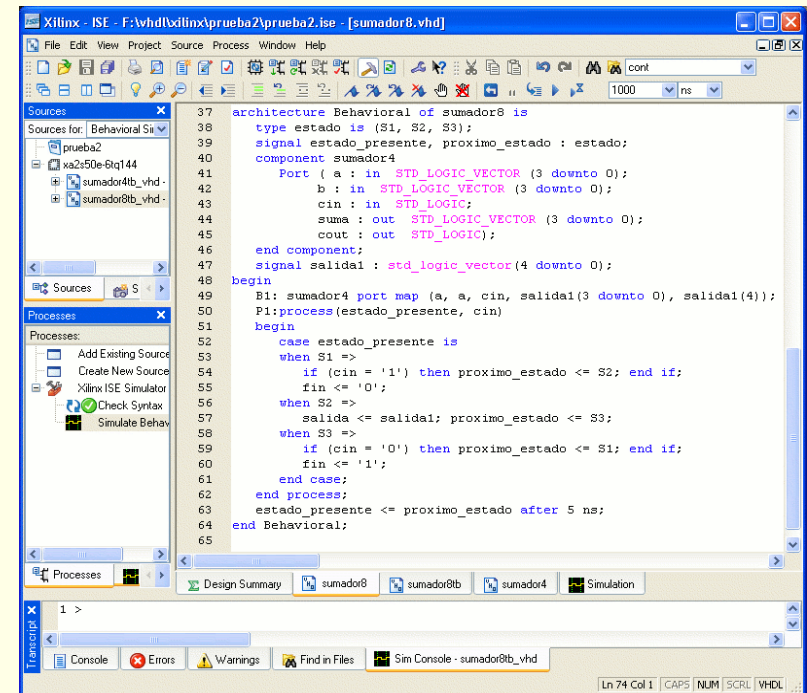
Herramienta XILINX

Creación de una nueva fuente

Plantillas



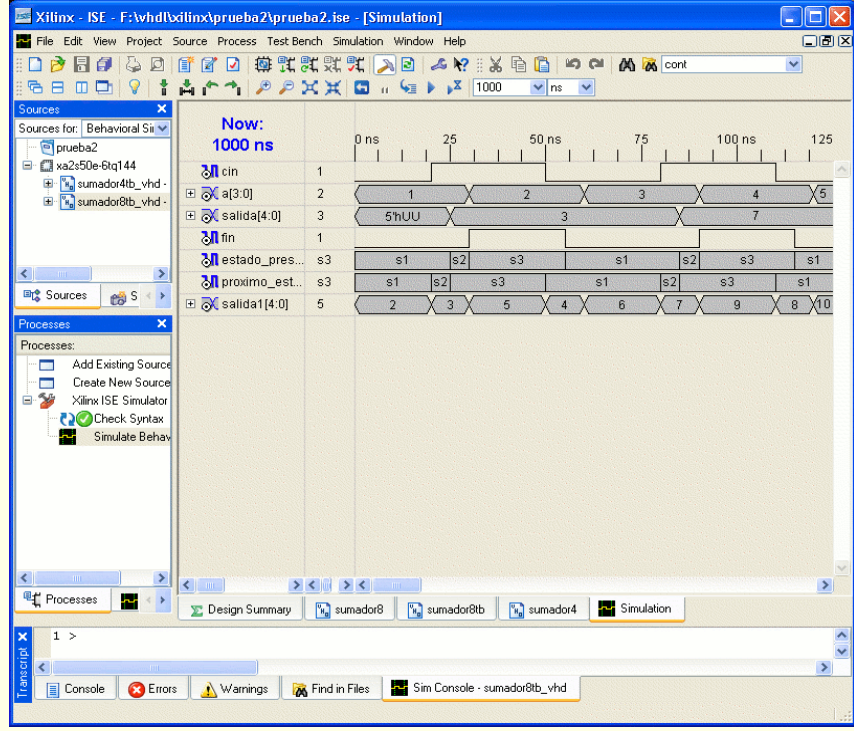
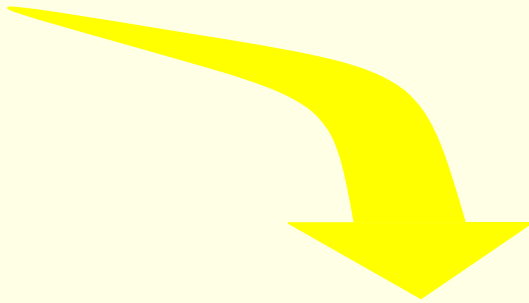
Copiar, pegar y cambiar



Simulación

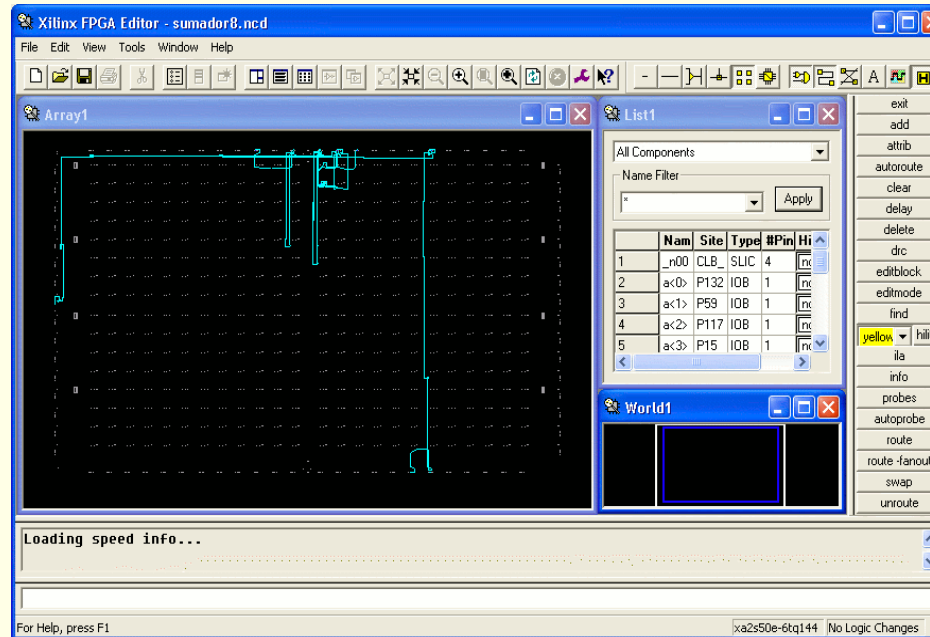
```
50 SIGNAL cin : std_logic := '0';  
51 SIGNAL a : std_logic_vector(3 downto 0) := (others=>'0');  
52  
53 --Outputs  
54 SIGNAL salida : std_logic_vector(4 downto 0);  
55 SIGNAL fin : std_logic;  
56  
57 BEGIN  
58  
59 -- Instantiate the Unit Under Test (UUT)  
60 uut: sumador8 PORT MAP(  
61 a => a,  
62 cin => cin,  
63 salida => salida,  
64 fin => fin  
65 );  
66  
67 tb : PROCESS  
68 BEGIN  
69 a <= a + 1;  
70 -- Wait 100 ns for global reset to finish  
71 wait for 20 ns;  
72 cin <= not cin;  
73 wait for 10 ns;  
74  
75 END PROCESS;  
76  
77 END;  
78
```

Simulación



Implementar

Implementar

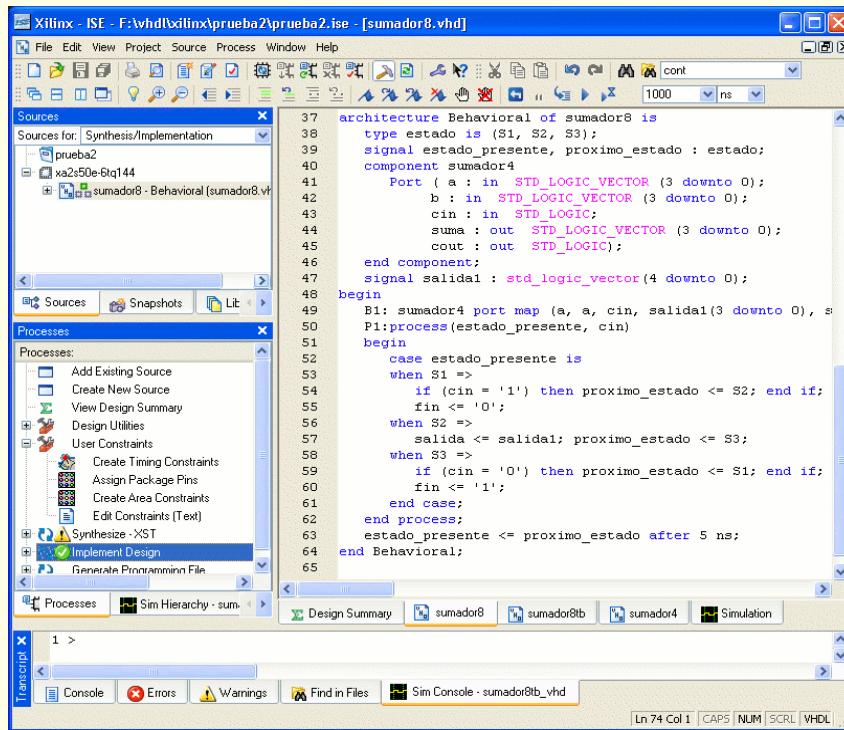


Cableado ineficiente
Colocación ineficiente

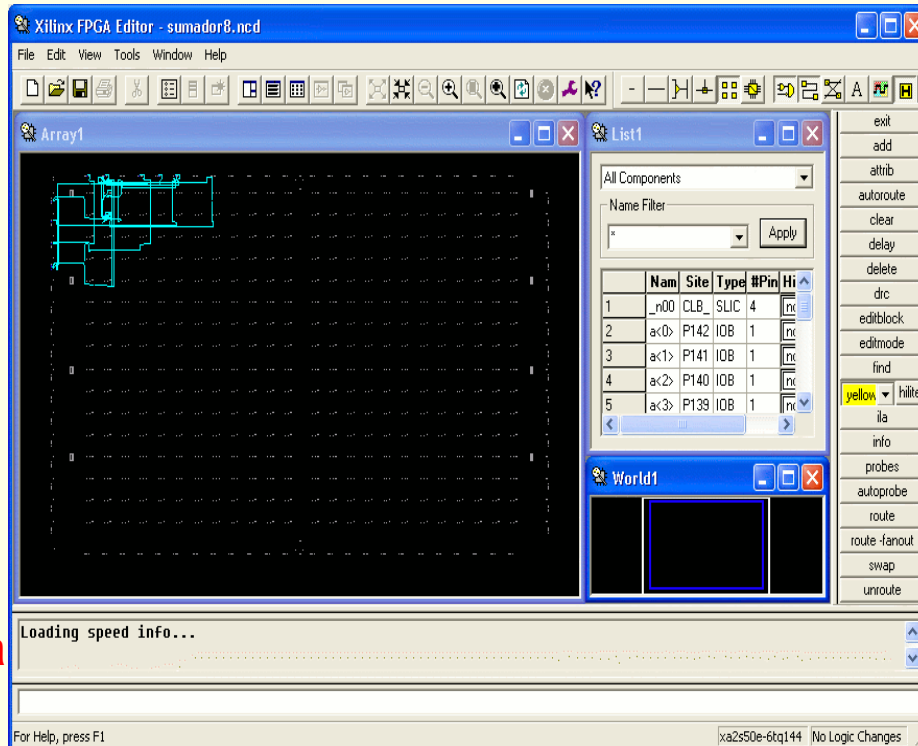
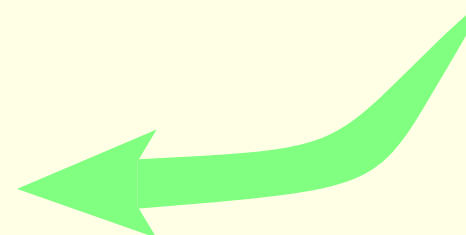
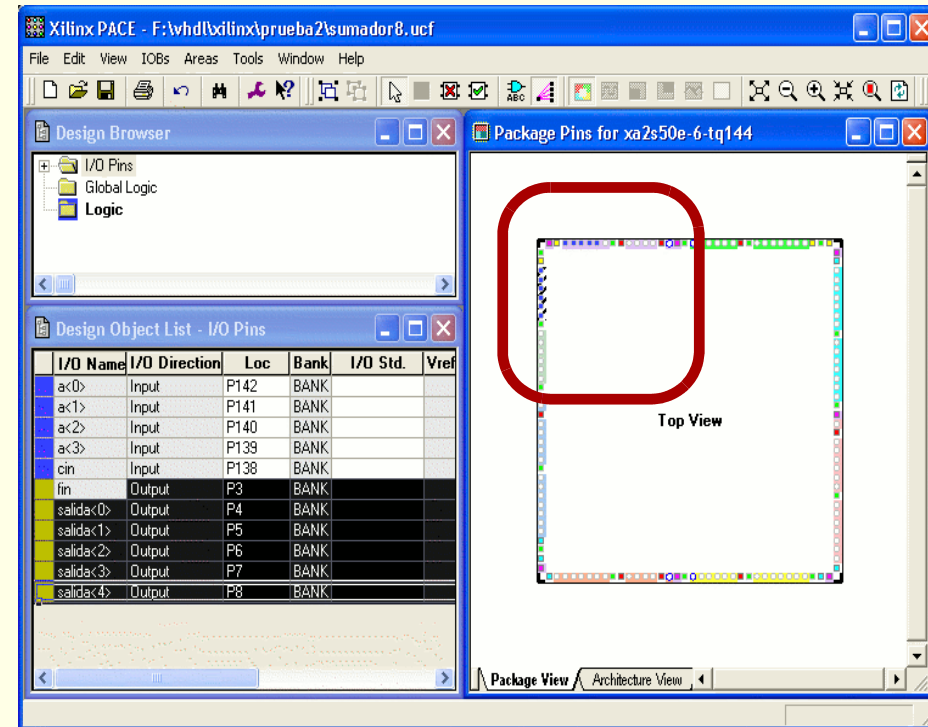
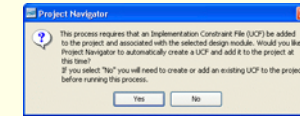
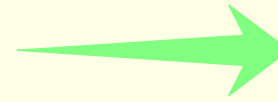
Muy útil incluir restricciones:

- Indicar la colocación de los puertos
- Indicar una estimación de área ocupada
- Indicar una estimación de retraso

Implementar

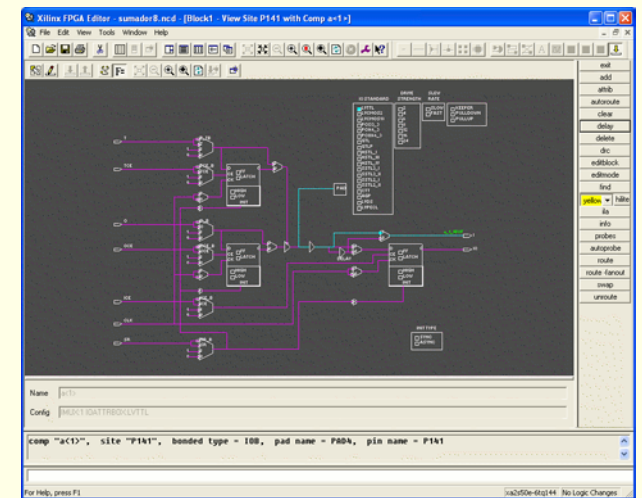
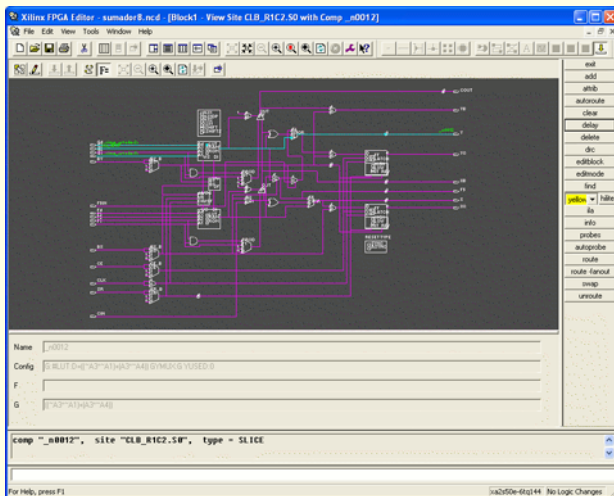
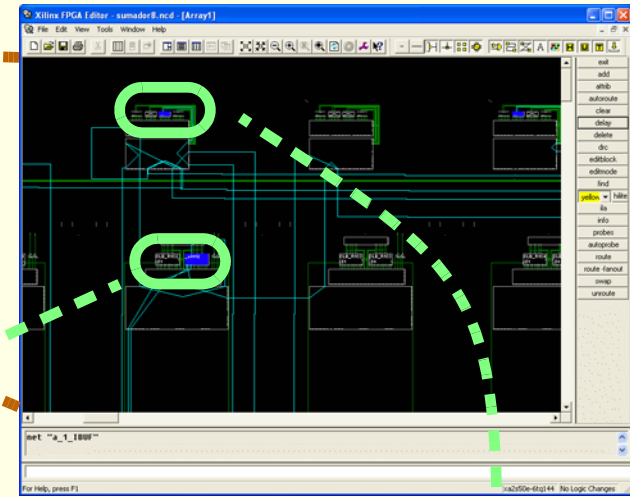
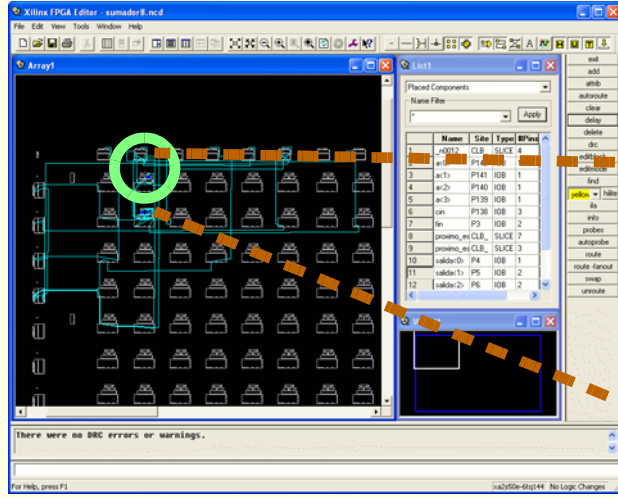


Fichero de restricciones



Herramienta XILINX

Implementar



Herramienta
XILINX