

Tutorial Xilinx-ISE



Manejo básico



Xilinx-ISE

La herramienta **Xilinx-ISE** (Integrated Software Environment) es una herramienta de diseño de circuitos profesional que nos va a permitir, entre otras funciones, la realización de esquemáticos y su posterior simulación

En este tutorial se va a utilizar concretamente:

-  **Entorno ISE**: donde se realizará el diseño del circuito, bien mediante un esquemático o utilizando un lenguaje específico de diseño
-  **ModelSim**: donde podrá realizarse la simulación del funcionamiento del circuito y de este modo comprobar si cumple con las especificaciones establecidas

Se describirá a continuación el funcionamiento del programa, y para ello se realizará el diseño de un circuito lógico que cumpla la siguiente tabla de verdad:

Num	A	B	C	D	Z
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	1
15	1	1	1	1	0

Haciendo la tabla de Karnaugh y las simplificaciones habituales, se obtiene:

$$Z = \overline{B}\overline{D} + \overline{A}BCD + AC\overline{D}$$

Para implementar este circuito, por tanto, necesitaremos:

- 1 puerta OR de 3 entradas
- 1 puerta AND de 2 entradas
- 1 puerta AND de 3 entradas
- 1 puerta AND de 4 entradas
- 3 inversores para generar las líneas negadas de A, B y D.

Este programa de diseño de circuitos trabaja bajo entorno WINDOWS, por lo que primero habrá que entrar en este sistema. Una vez iniciado Windows buscamos el icono correspondiente al programa **Xilinx ISE** (Figura 1) y lo ejecutamos.



Fig. 1

1. Creación de un nuevo proyecto

Un proyecto es un conjunto de ficheros de diseño, tales como esquemáticos, líneas de código (si se ha realizado el diseño utilizando un lenguaje de descripción y diseño de circuitos HDL), listas de conexiones, bibliotecas de componentes, vectores de test para la simulación, etc., seleccionados para un diseño específico

Nada más acceder al programa, aparecerá por pantalla una ventana como la que se muestra en la figura 2, y que da acceso al programa de diseño

Si hubiera ya algún proyecto cargado, se puede cerrar seleccionando la opción de menú [File → Close Project](#)

1. Creación de un nuevo proyecto

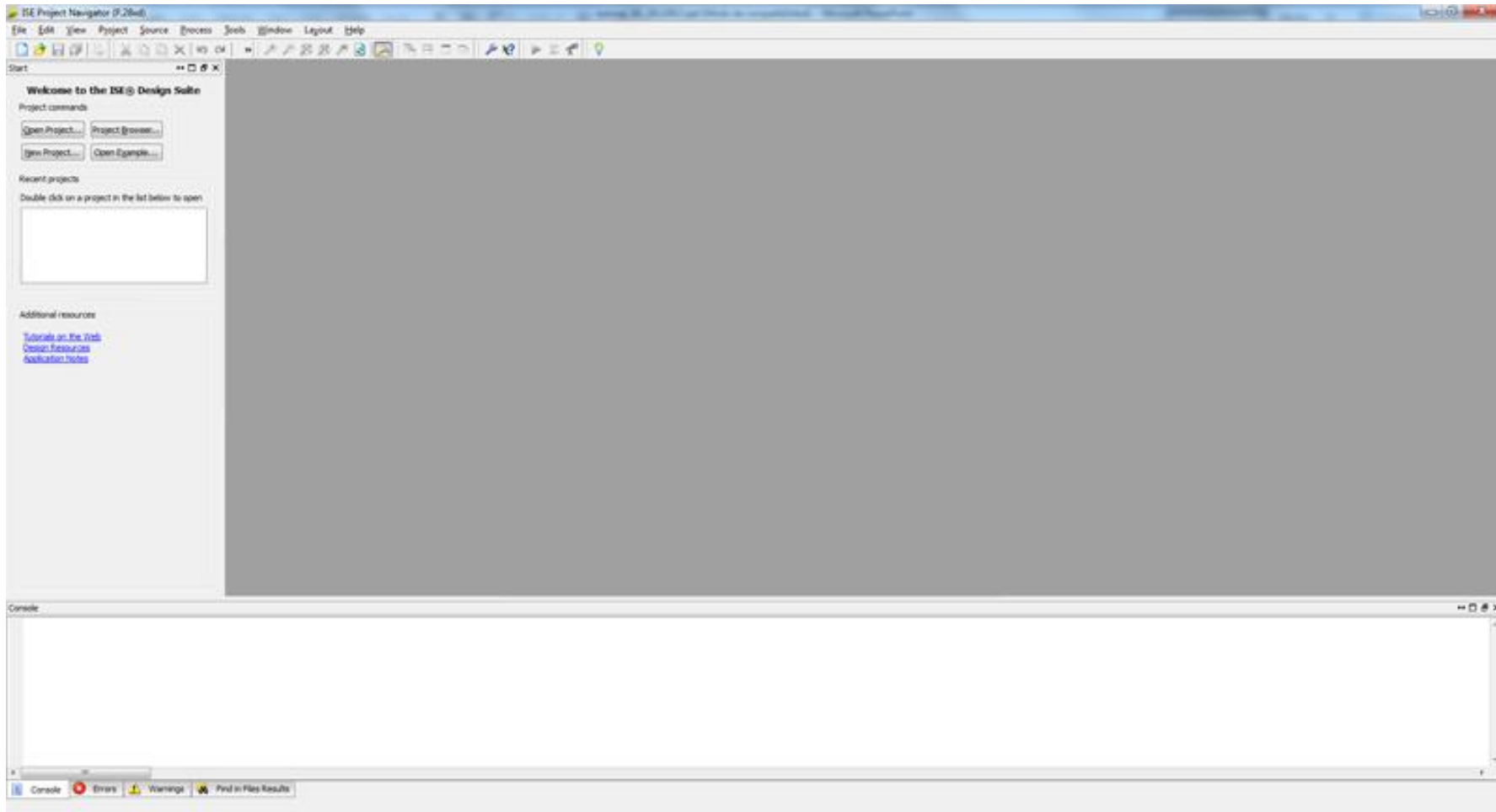


Fig. 2

1. Creación de un nuevo proyecto

Para crear un nuevo proyecto:

- Seleccionar **File → New Project**
- En la ventana de diálogo de New Project indicar el directorio de ubicación del proyecto en **Location**
- Añadir el nombre “Tutorial” en **Name**.
- En la opción **Top-level source type** se elige “Schematic” ya que el diseño se va a realizar con esquemáticos.
- Automáticamente se crea un subdirectorio en la ruta indicada en Location con el nombre del proyecto (p. ej. “Tutorial”), y donde se almacenará todo lo relacionado con este proyecto

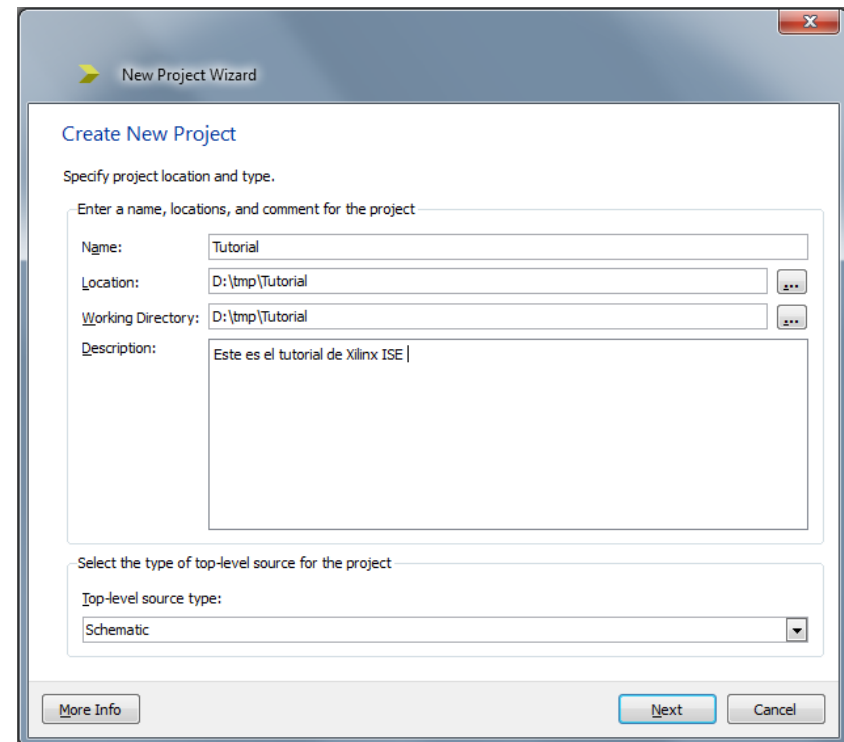


Fig. 3

1. Creación de un nuevo proyecto

En la segunda pantalla hay que seleccionar la FPGA sobre la que se va a sintetizar el diseño. En principio esta pantalla no es importante, pero para que todos tengáis las mismas opciones se seleccionará:

- **Family:** Spartan3 (familia)
- **Device:** xc3s200 (dispositivo dentro de la familia)
- **Package:** FT256 (encapsulado)
- **Speed:** -4 (indica la velocidad máxima del dispositivo)
- **Synthesis Tool:** XST (VHDL/Verilog)
- **Simulator:** Modelsim SE VHDL (herramienta de simulación)
- **Generated Language:** VHDL (lenguaje asociado para los ficheros que crea la herramienta)

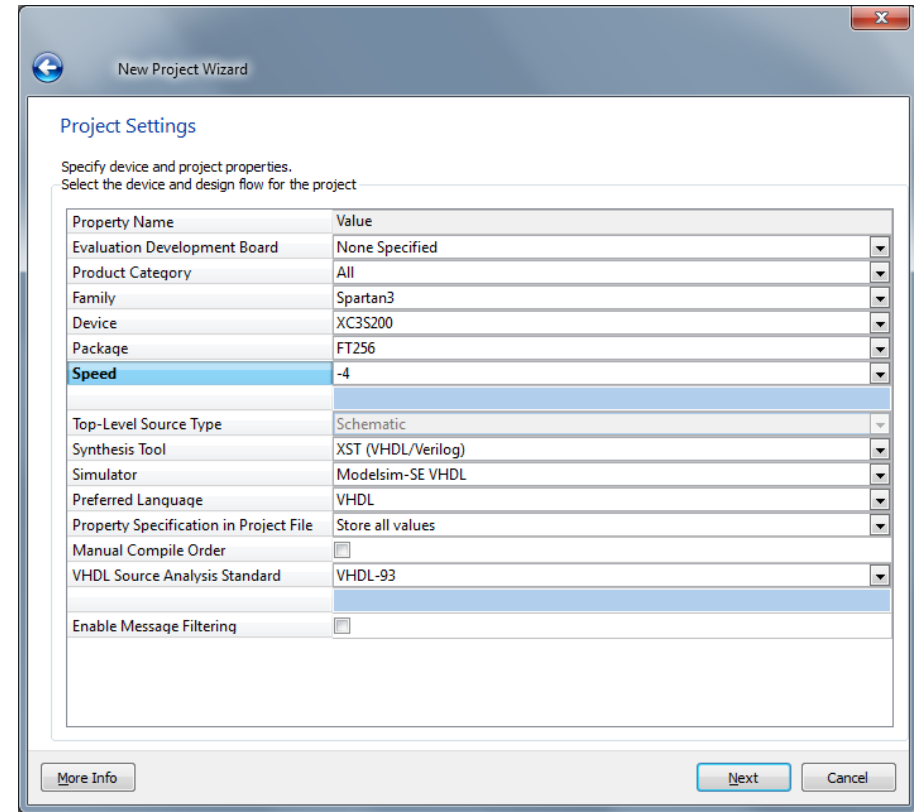


Fig. 4

1. Creación de un nuevo proyecto

Después se pasan las dos siguientes pantallas eligiendo **Next** y al dar a **Finish**, ISE crea y muestra el nuevo proyecto.

Se observarán cambios con respecto al aspecto inicial de la ventana en la parte izquierda, en **Design**, donde aparece el proyecto creado, como se ve en la figura 5

El paso siguiente será el diseño del circuito que responda a la tabla de verdad indicada al comienzo del tutorial.

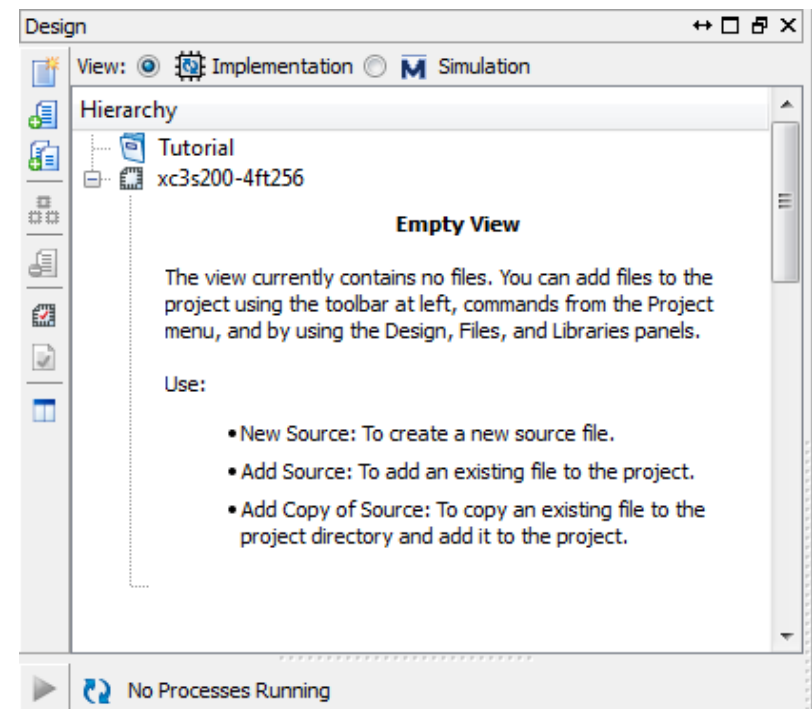


Fig. 5

2. Realización del esquemático del circuito

- Seleccionar **Project** → **New Source**
- Seleccionar **Schematic** como tipo de fuente
- En **File Name** le damos el nombre al diseño, por ejemplo “circuito” (Figura 6)
- Se presiona **Next**
- Se presiona **Finish** para concluir

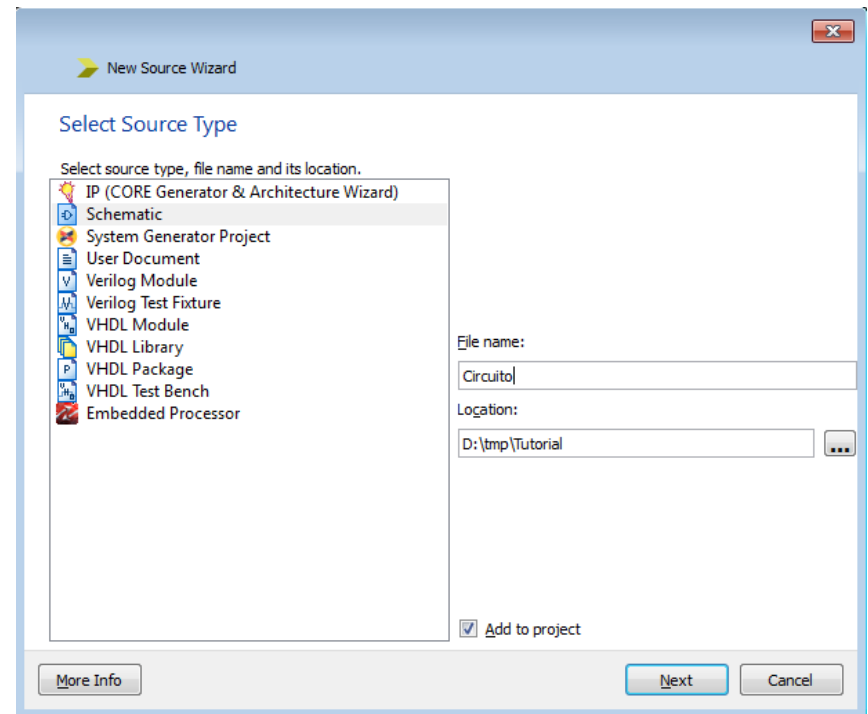


Fig. 6

2. Realización del esquemático del circuito

Se abre entonces la ventana derecha correspondiente al editor de esquemáticos (Figura 7), y los menús izquierdos cambian.

En el menú de la izquierda hay una nueva pestaña llamada Symbols que contiene la biblioteca de diseño de Xilinx.

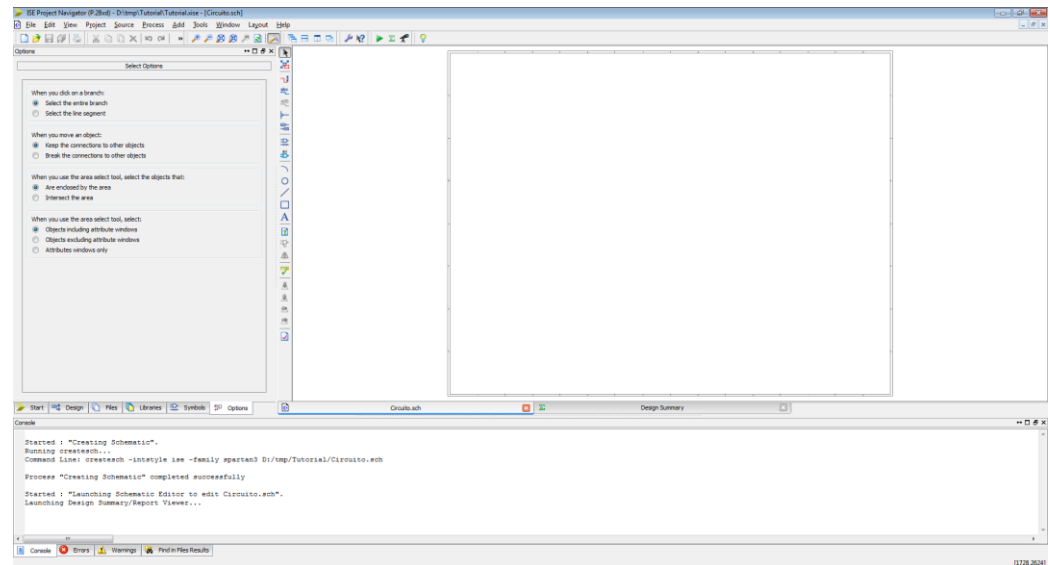


Fig. 7

2. Realización del esquemático del circuito

2.1. Componentes

Lo primero será fijar las puertas necesarias para la realización del circuito. En el menú de la parte izquierda de la herramienta (figura 8) puede verse una parte llamada **Categories**, donde se muestra una clasificación de las distintas categorías de componentes. Seleccionando cualquiera de ellas puede verse en la ventana inmediatamente inferior, **Symbols**, las puertas y componentes que forman la categoría seleccionada. Si se selecciona **<All Symbols>** podrá tenerse un listado de todos los componentes pertenecientes a la biblioteca de componentes

Para poder obtener las puertas lógicas necesarias, habrá que seleccionar **<All Symbols>** o Logic en la ventana de Categories, y una vez en Symbols, coger:

- **and2**: puerta AND de 2 entradas
- **and3**: puerta AND de 3 entradas
- **and4**: puerta AND de 4 entradas
- **or3**: puerta OR de 3 entradas
- **inv**: inversor (se necesitarán 3)

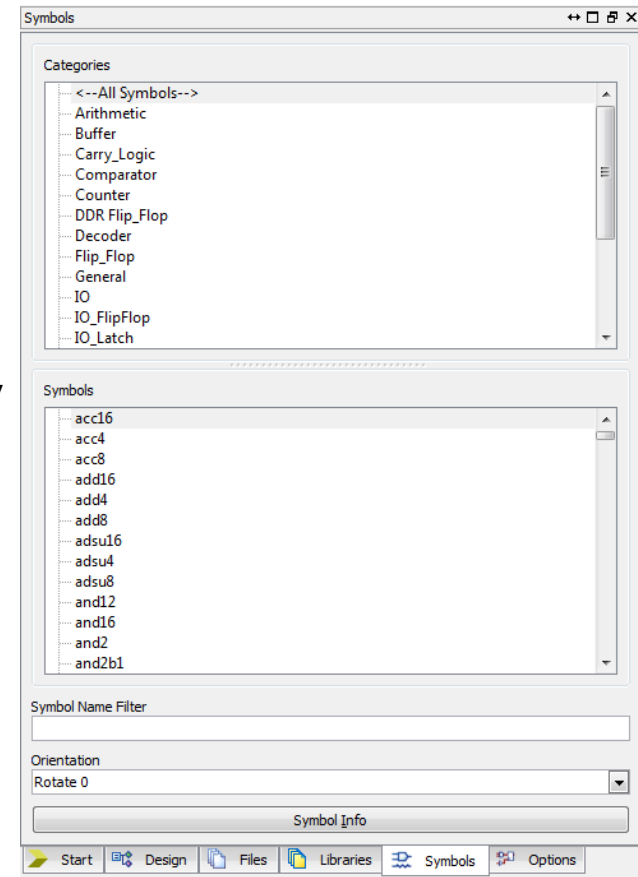


Fig. 8


2. Realización del esquemático del circuito

2.1. Componentes

Si se desea rotar un componente puede hacerse empleando la opción inferior del mismo menú, **Orientation**, y utilizando el menú desplegable elegir la orientación deseada

Para eliminar un componente del editor de esquemáticos simplemente hay que seleccionarlo y pulsar la tecla Supr. Asimismo, para mover un componente bastará con seleccionarlo y arrastrarlo hasta el lugar deseado mientras se mantiene pulsado el botón izquierdo del ratón

Tras la elección de las puertas necesarias, el editor de esquemáticos muestra la apariencia de la figura 9

Se recomienda salvar el esquemático a menudo, para evitar posibles problemas. Ello puede hacerse desde **File → Save As...** o pulsando el icono 

2. Realización del esquemático del circuito

2.1. Componentes

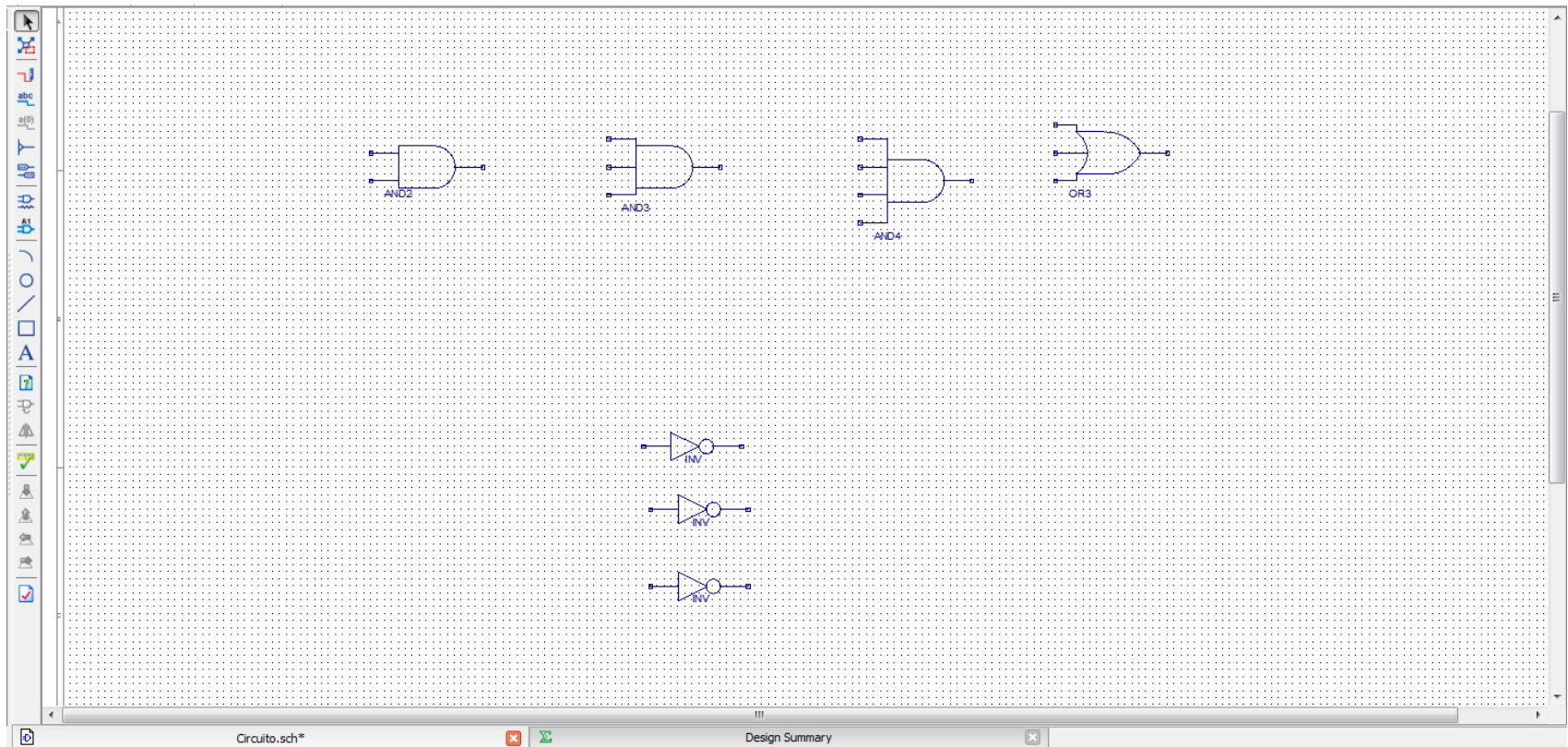



Fig. 9

2. Realización del esquemático del circuito

2.2. Conexiones

El siguiente paso será realizar las uniones entre las distintas puertas. Ello puede hacerse desde el menú principal del editor de esquemáticos: **Add → Wire (Ctrl+W)**, o pulsando el símbolo . El cursor se transformará en una cruz. Tendremos que pulsar con el botón izquierdo del ratón los dos extremos para que queden unidos por un cable

Si queremos unir una entrada o salida de un componente y dejar el otro lado del cable “al aire” tendremos que hacer un doble click con el botón izquierdo del ratón para indicar que el extremo final va suelto

El aspecto que ofrece el circuito en este momento, y tras hacer un zoom para tener una mejor visión, es el mostrado en la figura 10. Se añade, para mejor comprensión, y utilizando flechas auxiliares (que no forman parte del diseño) a qué entrada o salida corresponde cada cable

2. Realización del esquemático del circuito

2.2. Conexiones

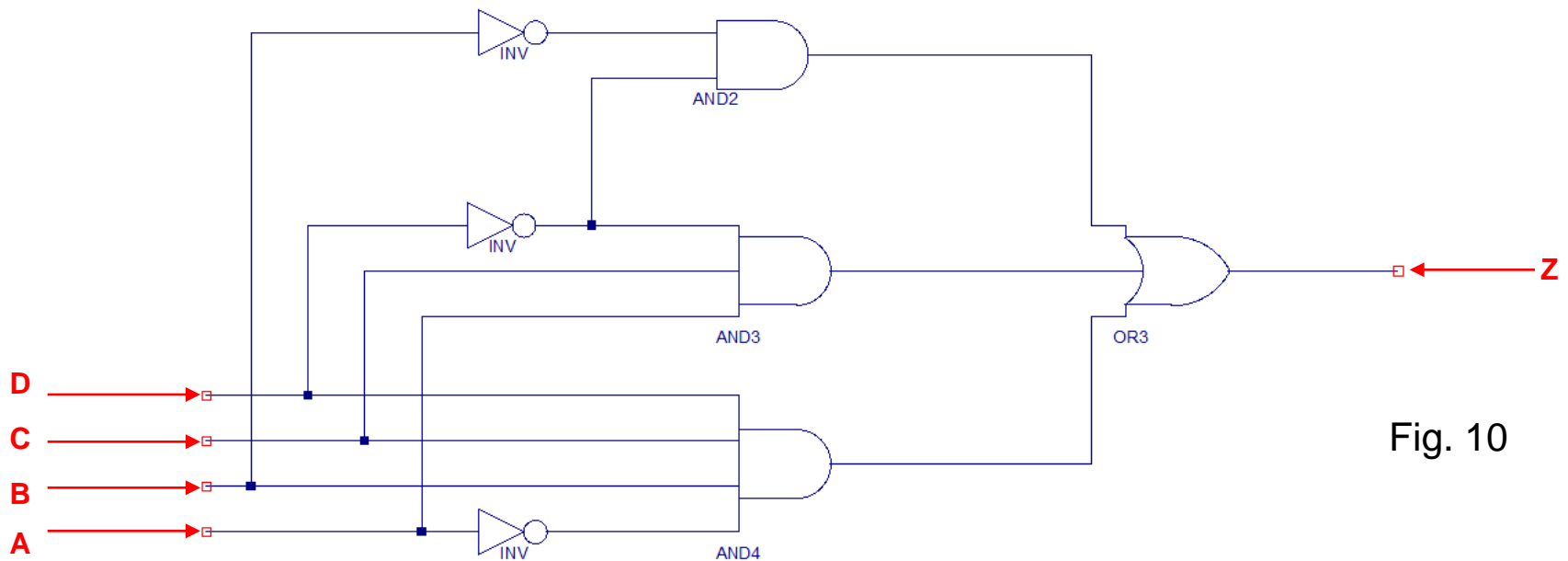



Fig. 10

2. Realización del esquemático del circuito

2.3. Añadir nombres a los cables

Llega el momento de indicar el nombre de cada uno de los cables, para saber así qué señal están llevando u obteniendo de las puertas

Esto puede realizarse desde el menú principal del editor de esquemáticos: **Add → Net Name**, o pulsando el símbolo . El cursor se transformará en una cruz

En el menú izquierdo habrá que añadir el nombre del cable en el campo de **Name**. Comenzaremos por “A” (ver figura 11a), y se seguirá con “B”, “C”, “D” y “Z”

En el cursor, que se había transformado en una cruz, se añade ahora la letra A. Habrá que arrastrar el cursor hasta el punto final del cable correspondiente a la señal A y que habíamos dejado libre y pinchar con el botón izquierdo en ese punto para añadir el nombre

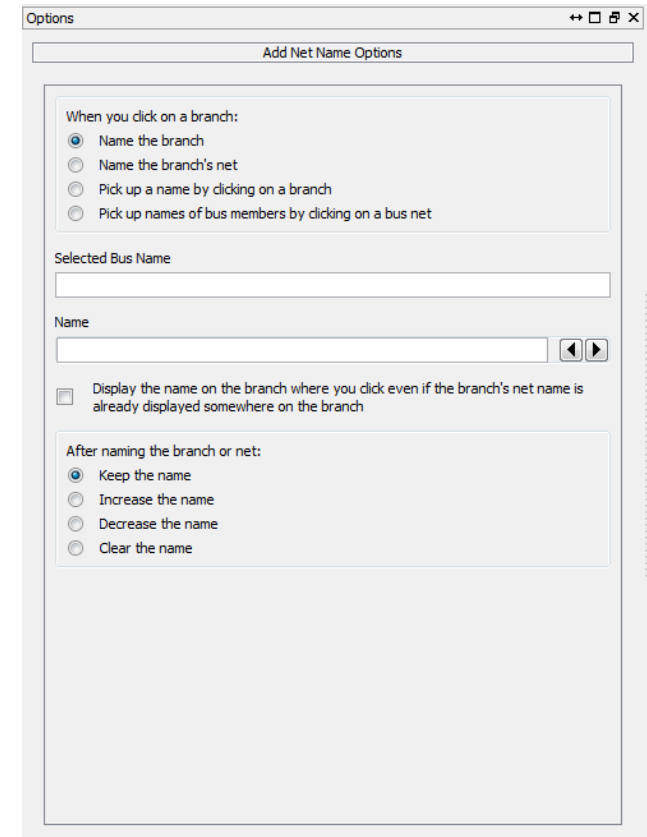



Fig. 11 a

2. Realización del esquemático del circuito

2.4. Terminales de entrada/salida

El siguiente paso será identificar la dirección de cada señal, esto es, si corresponde a una entrada o a una salida

Para añadir terminales de entrada/salida: **Add → I/O Marker** o bien pulsar el icono 

En el menú izquierdo (fig 11b) hay que indicar si es un terminal de entrada, de salida o bidireccional. Comenzaremos con las entradas, con lo que seleccionaremos **input**.

El cursor se transforma en un símbolo de terminal de entrada que habrá que arrastrar hasta el punto final de alguna de las entradas, que hemos nombrado previamente, y pinchar con el botón izquierdo del ratón. Veremos que el símbolo de entrada engloba a la letra correspondiente, fijándola como terminal de entrada:

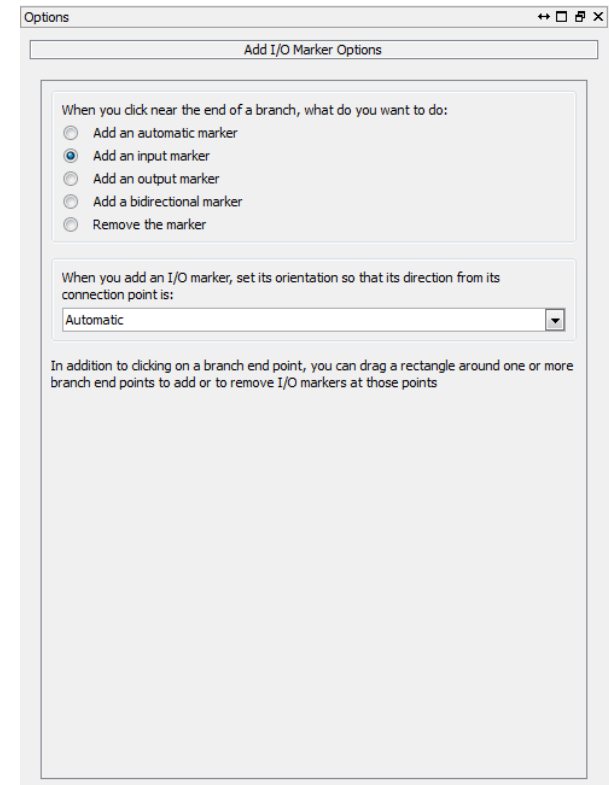


Fig. 11 b

2. Realización del esquemático del circuito

2.4. Terminales de entrada/salida

Tras repetir la operación para el resto de entradas y para la salida (seleccionando previamente **Output** en el mismo menú) el aspecto del esquemático del circuito se muestra en la figura 12

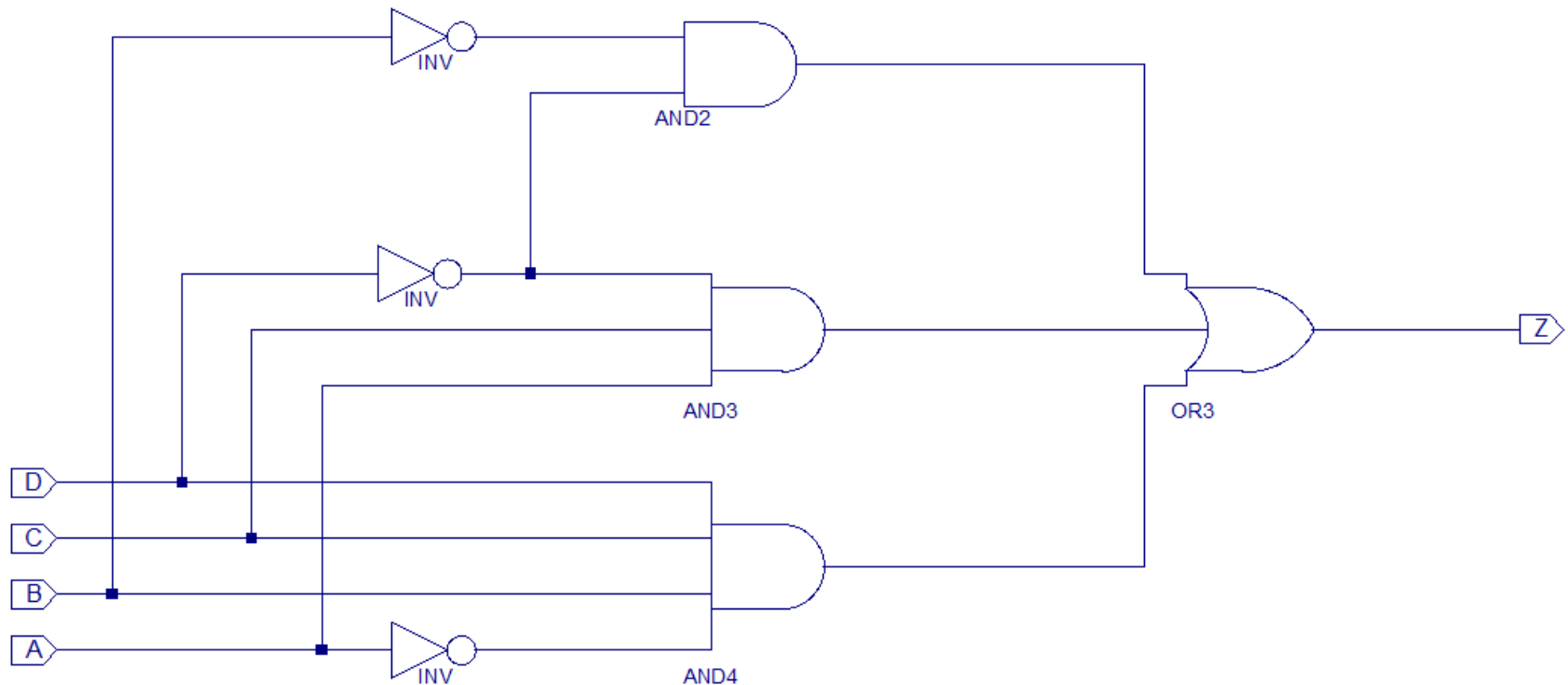


Fig. 12

2. Realización del esquemático del circuito

2.5. Comprobación del circuito

Una vez terminado el diseño del circuito puede hacerse una simple comprobación sobre el mismo, para asegurarnos de que todas las conexiones están hechas correctamente y que no quedan cables sueltos o puertas sin conectar.

Para ello, desde el menú principal: [Tools](#)→[Check Schematic](#).

En la ventana inferior se informa si hay algún error en el circuito. Si todo está correcto la ventana será la mostrada en la figura 13

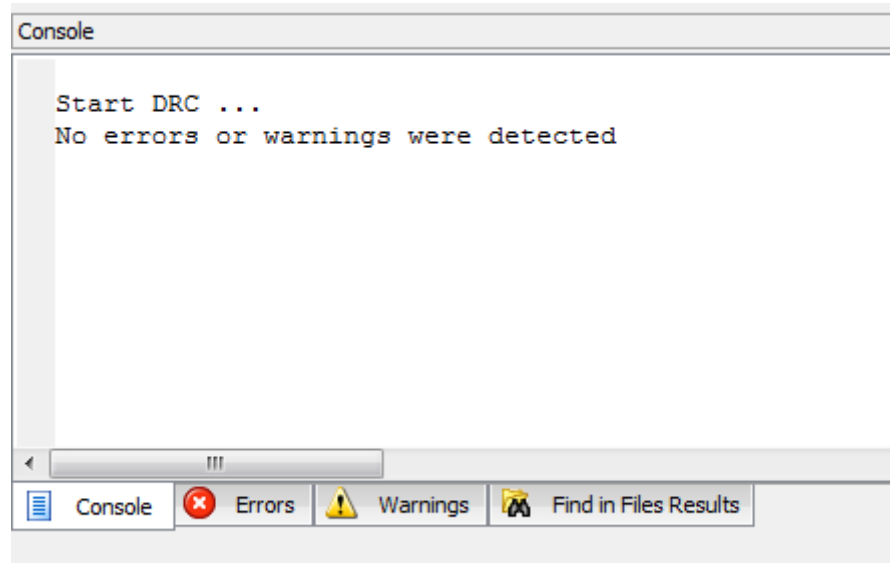


Fig. 13

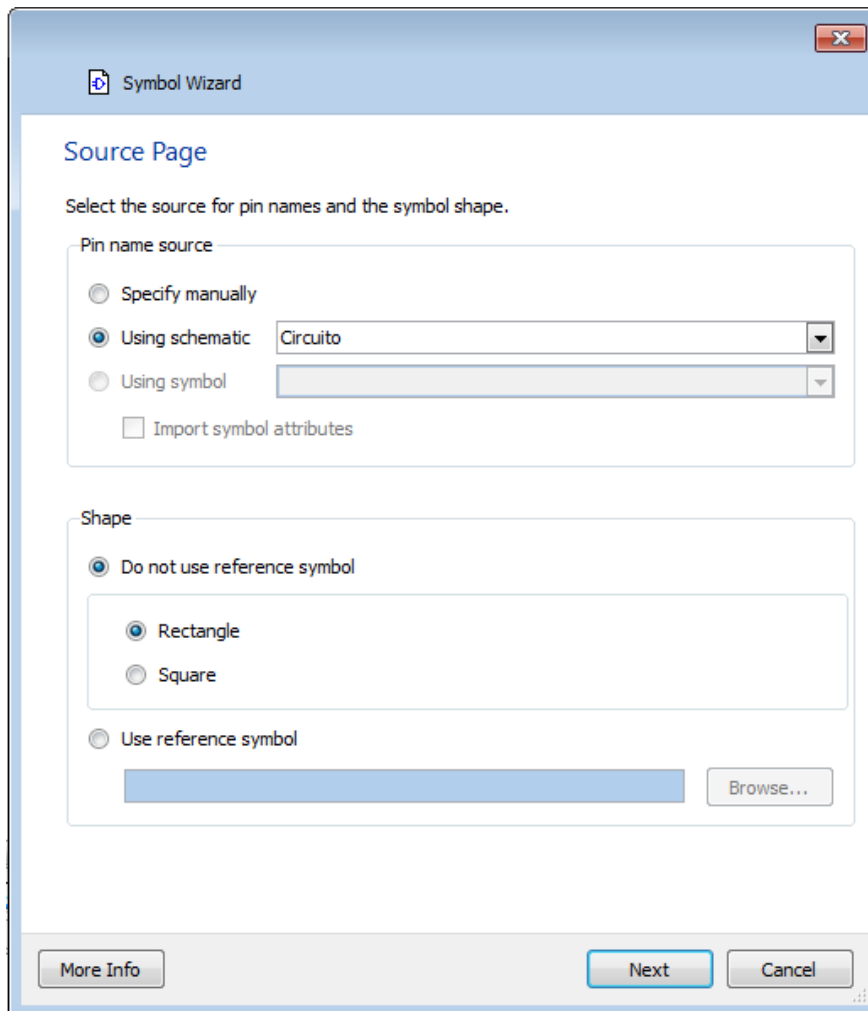
3. Creación de un símbolo

Con el esquema realizado se puede hacer un nuevo componente o símbolo consistente en una especie de “caja negra”, donde el usuario sólo verá un bloque con el mismo número de entradas y de salidas que el esquema original, que englobará a éste y realizará su misma función

Este símbolo que se crea pasa a formar parte de la biblioteca de componentes del proyecto, y podrá utilizarse como un componente más

Para crearlo: **Tools→Symbol Wizard** desde el menú principal. Aparece entonces la ventana de la figura 14, donde hay que indicar que el símbolo se crea a partir del esquemático ya realizado (opción **Using Schematic** con el esquemático “Circuito”)

3. Creación de un símbolo



Pulsando Next tres veces y después Finish (aceptando las opciones por defecto a menos que se quiera variar el aspecto final del símbolo) el programa genera el símbolo y lo muestra (figura 15)

Fig. 14

3. Creación de un símbolo

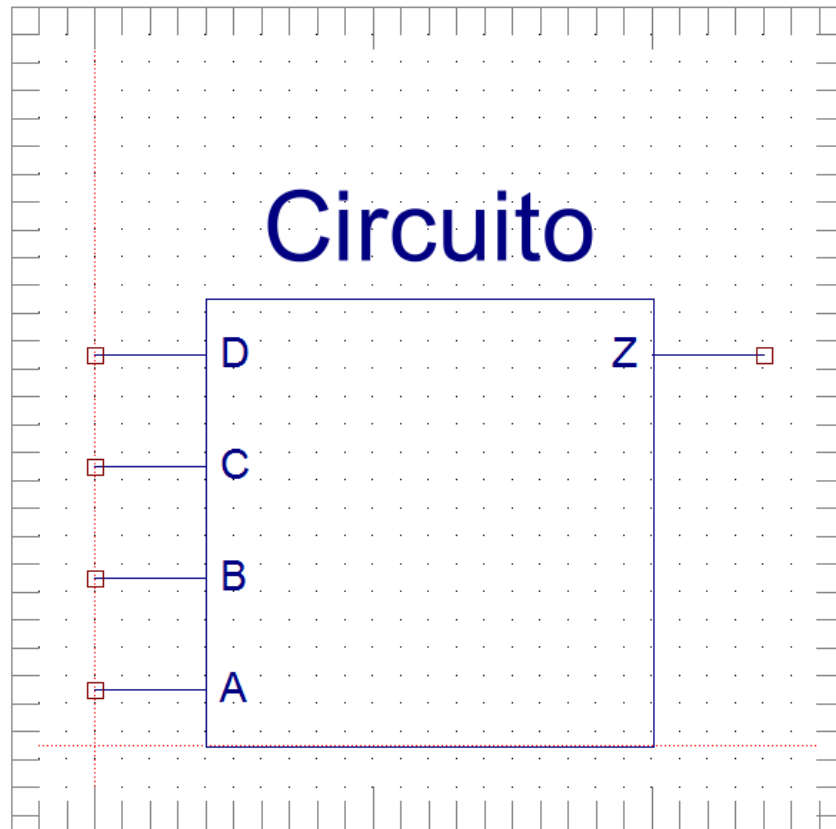


Fig. 15

3. Creación de un símbolo

Si nos fijamos en la parte inferior izquierda de la ventana del editor de esquemáticos, podemos ver que hay dos pestañas, [Circuito.sch](#) y [Circuito.sym](#), correspondientes a las dos hojas que tenemos abiertas en el editor de esquemáticos. Pulsando sobre [Circuito.sch](#) se accede al esquema original y sobre [Circuito.sym](#) al símbolo que hemos creado del circuito

En el esquema del circuito original, si ahora nos fijamos en la ventana de la parte izquierda, en la pestaña [Symbols](#) y en la parte de [Categories](#), vemos que hay una nueva categoría, [<D:\tmp\tutorial>](#), que se corresponde a la biblioteca del proyecto que hemos abierto, y donde ha almacenado el componente creado. De hecho, si seleccionamos esta categoría, podemos ver en la ventana inferior, [Symbols](#), que ahí está el componente circuito creado, y que ahora puede utilizarse como un componente más a la hora de diseñar

4. Simulación

Una vez que se tiene el circuito listo, revisado y libre de errores, será necesario comprobar su correcto funcionamiento, y para ello habrá que realizar la simulación del circuito.

Para ello, se debe realizar un banco de pruebas (o *testbench* en inglés), el cual genere estímulos en las entradas del circuito, y así poder comprobar si su salida es correcta.

En versiones anteriores de Xilinx ISE se podían generar bancos de pruebas de forma intuitiva y gráfica. Sin embargo, en las versiones modernas, los bancos de pruebas deben realizarse mediante VHDL. VHDL es un lenguaje de descripción de circuitos muy potente pero su aprendizaje no entra en los objetivos del presente curso. Por tanto, se usará un banco de pruebas en VHDL, pero sin entrar en detalles del código.

4. Simulación

4.1. Añadir el banco de pruebas

Lo primero será añadir un banco de pruebas suministrado junto al enunciado de la práctica.

Los pasos a seguir son los siguientes:

- Seleccionar **Project→Add Source...**
- En la ventana de diálogo que aparece (figura 16), seleccionar el fichero **CircuitoTb.vhd**
- En la siguiente ventana, pulsar **OK**

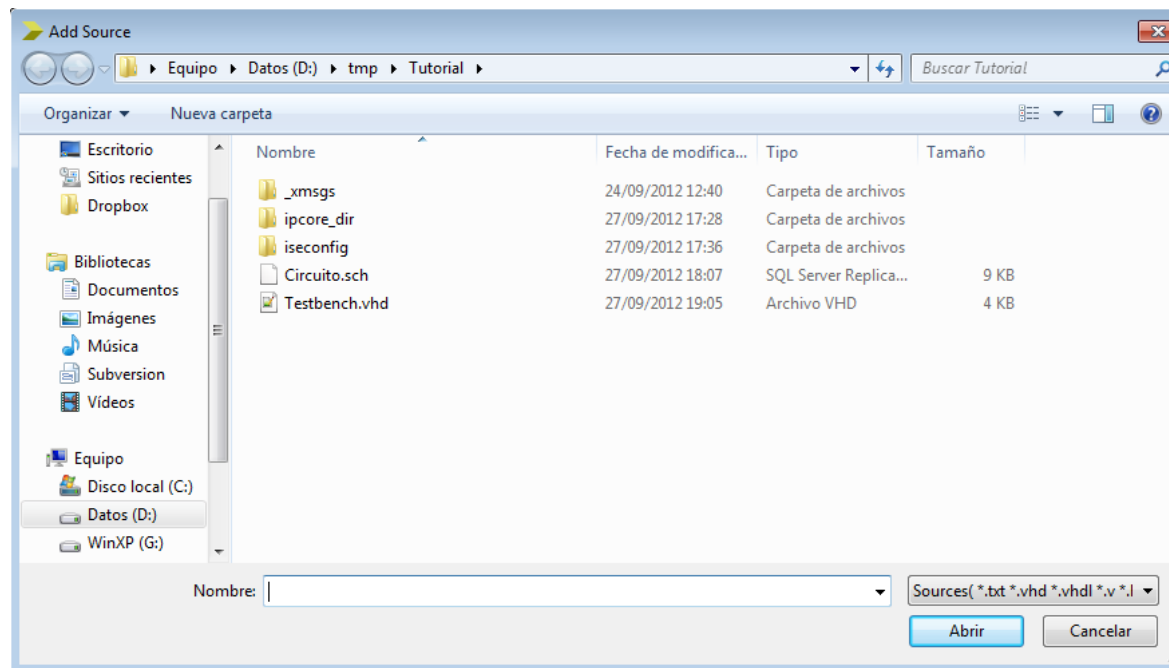
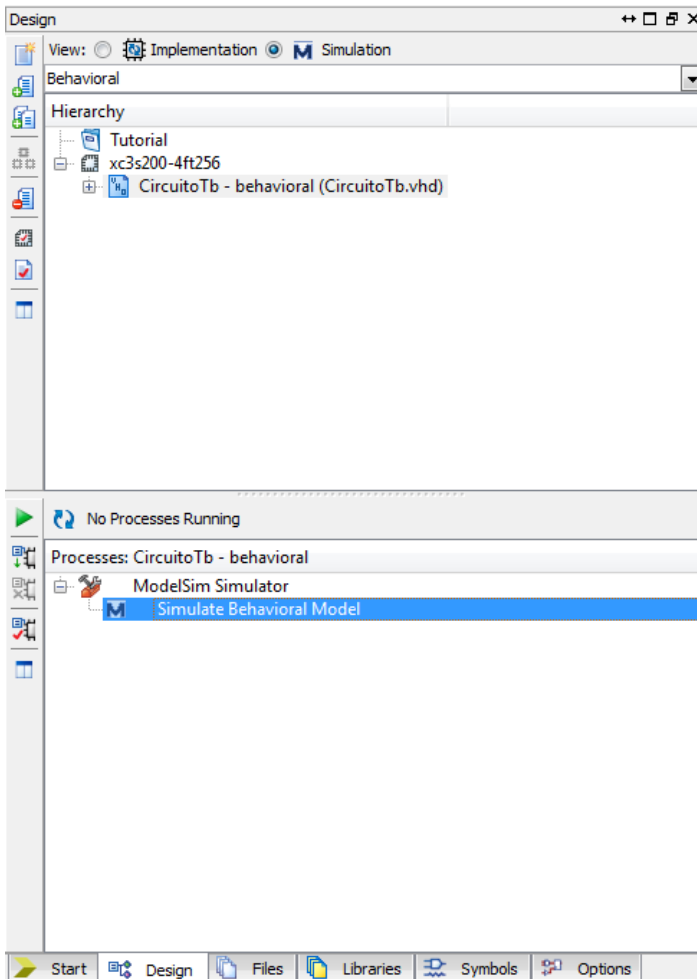


Fig. 16

4. Simulación

4.2. Simulación del comportamiento del circuito



Para realizar una **simulación del comportamiento** del circuito (behavioral), se utiliza la herramienta **ModelSim**

Pasos a seguir:

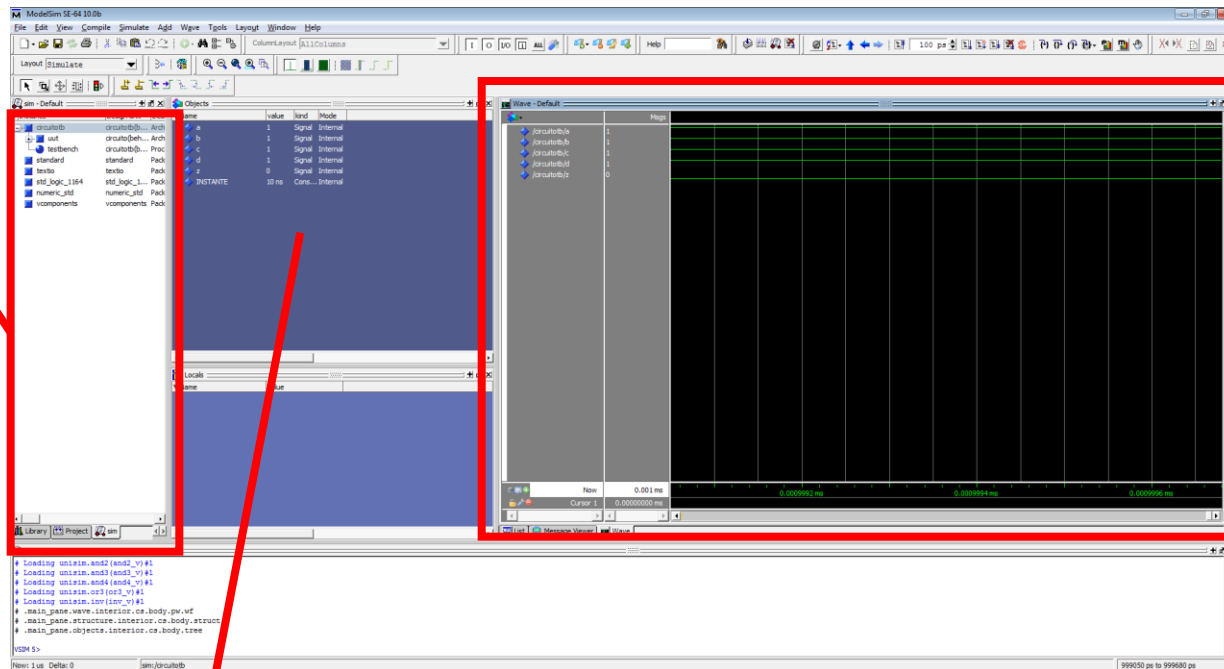
- En el menú izquierdo, pulsar sobre la pestaña **Design**, y arriba marcar la opción **Simulation**
- Seleccionar el fichero **CircuitoTb.vhd**
- En la ventana **Processes**, desplegar el menú de jerarquía con **+** y hacer doble click en **Simulate Behavioral Model** (figura 17). En ese momento se carga el programa **ModelSim**

Fig. 17

4. Simulación

Al arrancar el programa ModelSim aparece la ventana ModelSim SE, se ejecuta la simulación y aparecen una serie de ventanas embebidas en la principal (ver figura 18)

Estructura
del diseño
simulado
(workspace)



Formas de
onda
simuladas
(wave)

Señales en el nivel
seleccionado de la
estructura (objects)

Fig. 18

4. Simulación

Los resultados de la simulación del comportamiento del circuito se mostrarán en la ventana de ondas ([wave](#))

Nota: ISE automatiza el proceso de simulación creando y ejecutando un fichero macro de simulación (fichero [.fdo](#)), que no es visible para el usuario pero que realiza las siguientes funciones:

- Crea las bibliotecas de diseño
- Compila el diseño y los ficheros del banco de pruebas
- Llama al simulador
- Abre todas las ventanas relacionadas con la simulación
- Añade todas las señales a la ventana de ondas ([wave window](#))
- Corre un tiempo igual a la duración del Test Bench Waveform generado previamente

4. Simulación

En la ventana **Wave-Default**, habrá que hacer un zoom para poder visualizar la salida del circuito. Presionar el menú **Zoom → Zoom Full**. El resultado de la simulación debería ser como el mostrado en la figura 19

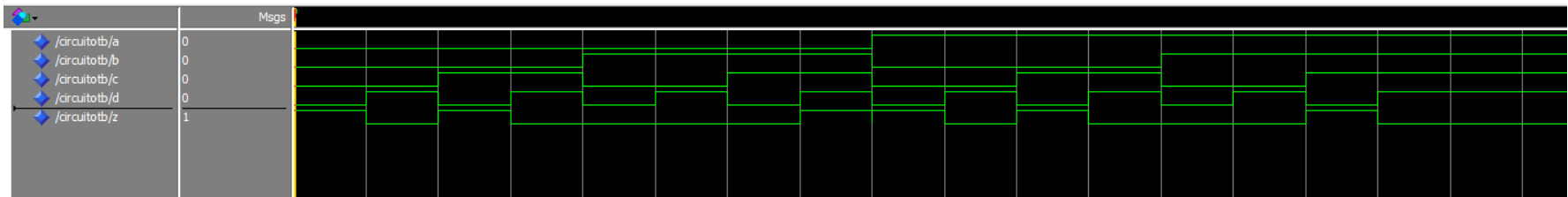
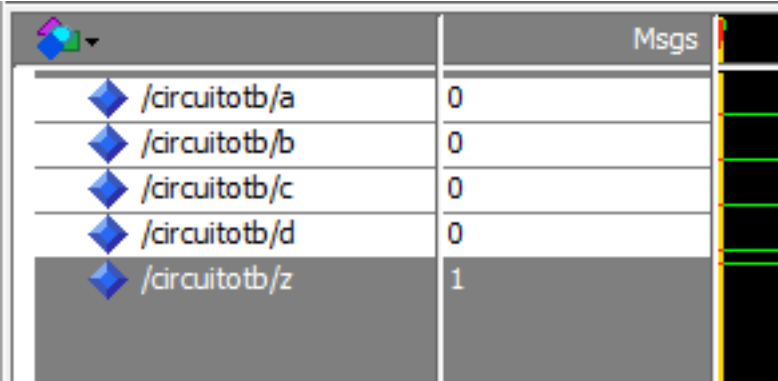


Fig. 19

4. Simulación

Para una comprobación más fácil del correcto funcionamiento del circuito y puesto que las entradas recorren todos los posibles estados (0, 1, 2... 15) sería bueno poder reunir las entradas individuales en una, bajo la forma de un vector, y que así nos mostrara los estados por los que van pasando las combinaciones de las señales a las que representa

Para ello, seleccionamos las entradas del circuito en la ventana **Wave-Default** pulsando el botón izquierdo del ratón y las teclas Ctrl, con lo que aparecerán en el interior de un recuadro blanco:



	Msgs
◆ /circuitotb/a	0
◆ /circuitotb/b	0
◆ /circuitotb/c	0
◆ /circuitotb/d	0
◆ /circuitotb/z	1

Fig. 20

4. Simulación

Con las señales que se vayan a combinar ya seleccionadas, se utiliza la opción de menú **Tools** (o **Wave**, según versión) → **Combine Signals...**:

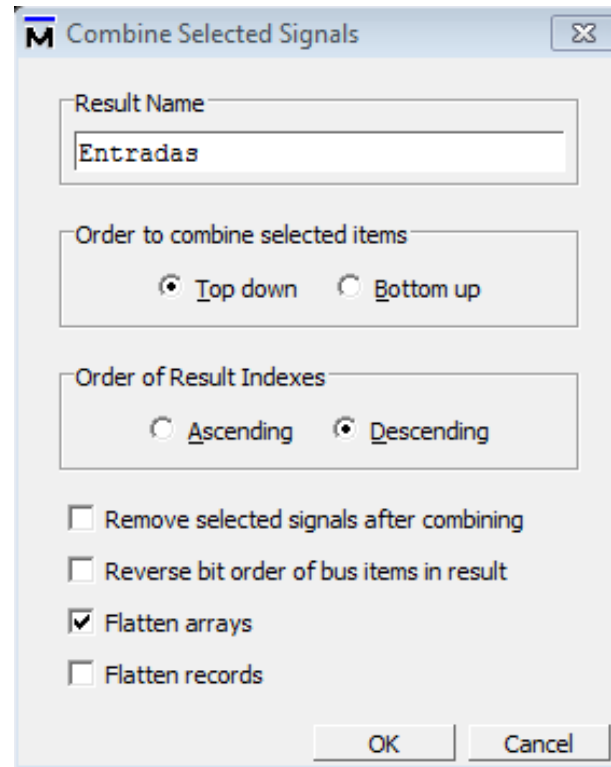


Fig. 21

En el cuadro de diálogo que aparece hay que dar un nombre a la nueva señal combinada, por ejemplo “**Entradas**”

4. Simulación

Pulsamos OK y vemos que en la ventana [Wave-Default](#) aparece una nueva señal [/CircuitoTb/Entradas](#). Si pulsamos en el símbolo **+** podemos ver la jerarquía, esto es, las señales que tiene por debajo y a las que representa

Para visualizar los valores de esta señal combinada como número en vez de en binario, habrá que seleccionar la entrada combinada, pulsar el botón derecho del ratón y en el menú de opciones elegir [Radix→Unsigned](#). Para mayor facilidad, podemos desplazar la señal de salida Z inmediatamente debajo de la señal de entrada combinada, pinchando en la señal de salida y arrastrando hacia arriba la señal con el botón izquierdo del ratón pulsado. El aspecto de la simulación se muestra en la figura 22

4. Simulación

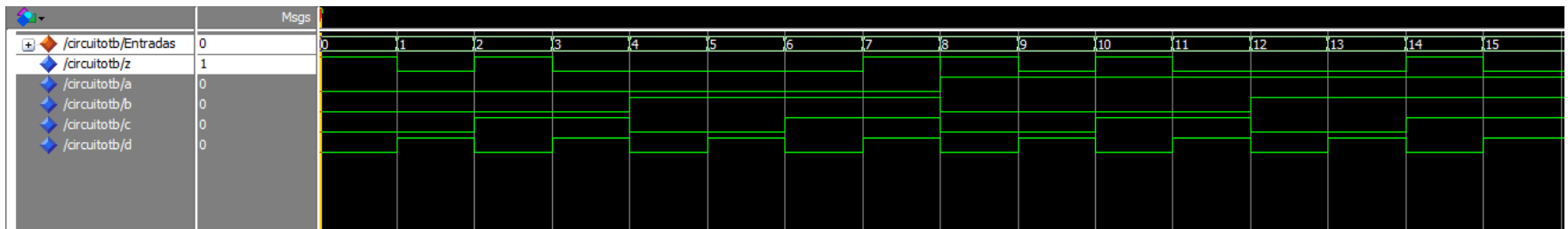


Fig. 22

Se sale de ModelSim cerrando la ventana [ModelSim SE](#)

5. Instalar ISE versión estudiante

- ✓ Xilinx facilita de forma gratuita la aplicación ISE, en un versión reducida pero suficiente para la realización de las prácticas.
 1. Accediendo a la página web de Xilinx: <http://www.xilinx.com/>
 2. Avanzando por los enlaces siguientes: **Support** >> [Downloads](#)
 3. Seleccionar y acceder a la versión gratuita de ISE_2012.2 - 14.1, que es la versión utilizada en los laboratorios.
 4. Seleccionada la versión y dependiendo del SO instalado, Xilinx permite el acceso en la sección “**ISE Design Suites** – 14.1 Full Product Installation” entre otras a la aplicación “ISE WebPACK (Free)” que debe ser nuestro objetivo.
 5. A partir de este punto, la web solicita abrir una cuenta antes de proceder a la descarga. Este afiliación no supone ningún gasto ni compromiso.
 6. Si se encuentran dificultades en este sencillo proceso, consultar con cualquiera de los profesores de la asignatura.
- ✓ A diferencia de la aplicación para simulación utilizada en el laboratorio (ModelSim), ISE utiliza su propio simulador (ISim) integrado en la propia aplicación ISE. Esto no debe suponer un impedimento para trabajar en casa puesto que ISim tiene similares características a ModelSim .