

Proyecto básico con Libero

Creador: David Rubio G.

Entrada: <https://soceame.wordpress.com/2024/12/01/proyecto-basico-con-libero/>

Blog: <https://soceame.wordpress.com/>

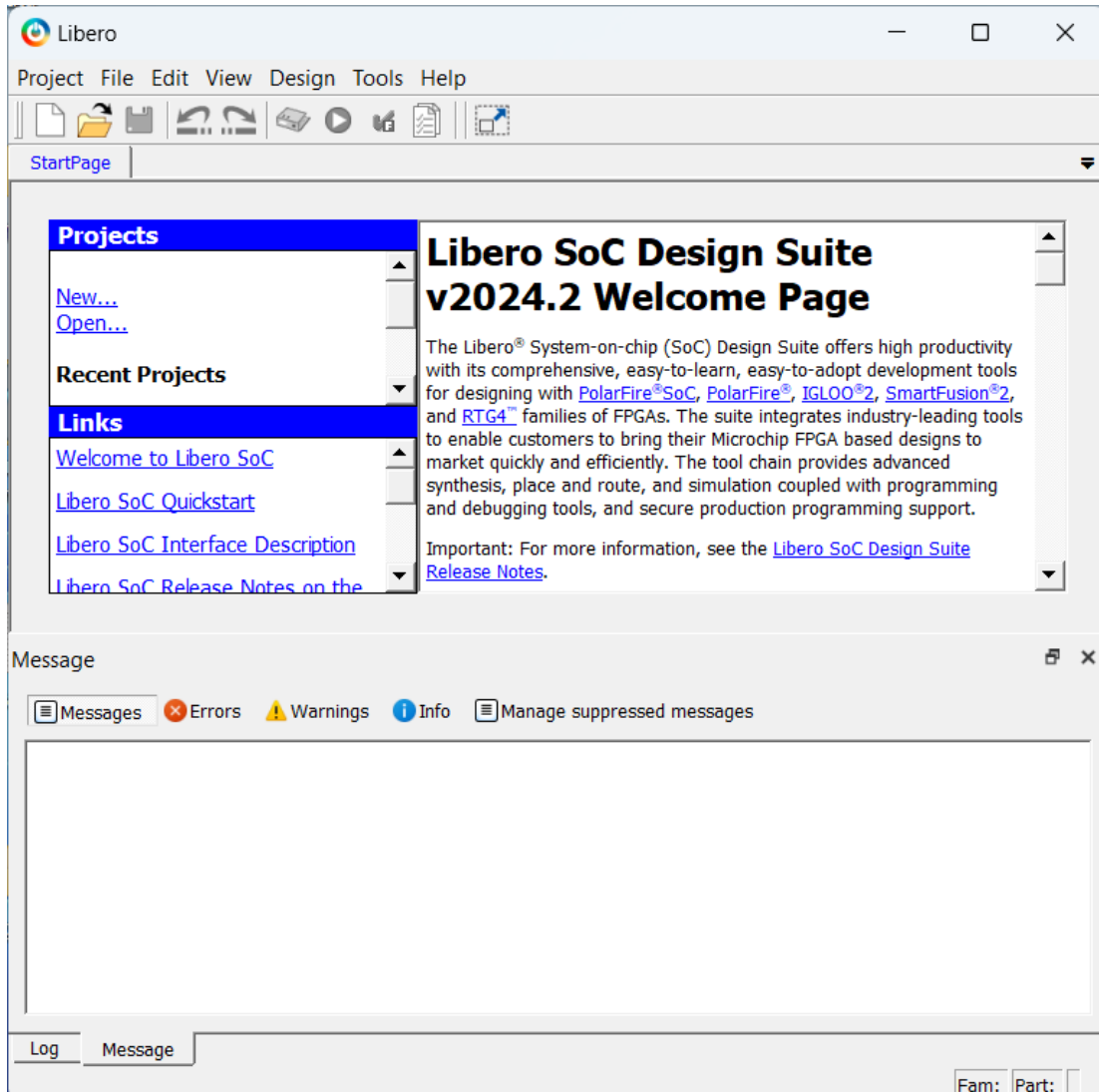
GitHub: <https://github.com/DRubioG>

Fecha última modificación: 23/02/2025

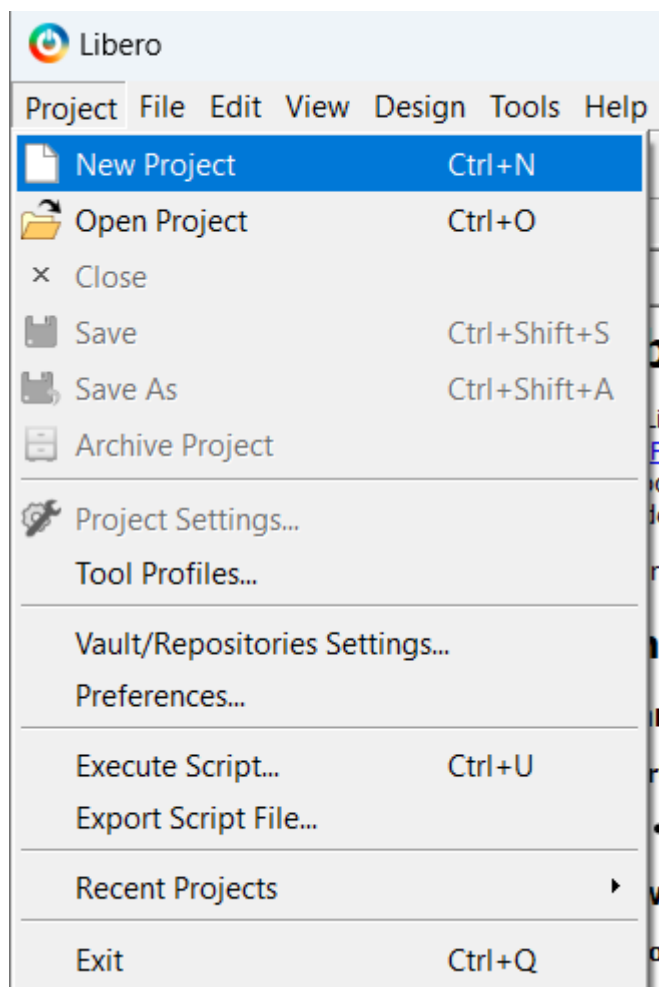
En esta entrada voy a comentar cómo hacer un proyecto básico con Libero.

Crear un proyecto

El primer paso para trabajar con Libero es abrir la herramienta.



Una vez abierta le decimos que queremos crear un proyecto nuevo.



Después se nos abre una pestaña para la construcción del proyecto. En la primera parte nos pregunta sobre cómo queremos llamar al proyecto, dónde lo vamos a crear y el lenguaje que vamos a utilizar.

NOTA: es conveniente recordar que la licencia gratuita de Libero no permite utilizar dos lenguajes en un mismo proyecto.

New project

Project details
Specify project details

Project name: test

Project location: C:/ Browse...

Description:

Preferred HDL type: VHDL

☐ Enable block creation

Block flow enables you to publish a reusable component that can be instantiated into another design. A block component may not contain I/O cells and cannot be programmed by itself. It could include timing constraints, physical constraints, placement or routing.

Libero
System-on-Chip

Help < Back Next > Finish Cancel

La siguiente pestaña nos pregunta por el dispositivo que vamos a programar.

New project

Device selection
Select a part for your project from the part number list

Selected part: MPFS095T-FCSG325E

Part filter

Family: PolarFireSoC Die: All Package: All

Speed: All Range: All

Reset filters

Search part:

Part Number	DFF	User I/Os	uSRAM	LSRAM	Math	H-Chip Globals	PLL
MPFS095T-1FCSG536E	93516	144	876	308	292	48	8
MPFS095T-1FCSG536I	93516	144	876	308	292	48	8
MPFS095T-1FCVG484E	93516	144	876	308	292	48	8
MPFS095T-1FCVG484I	93516	144	876	308	292	48	8
MPFS095T-1FCVG784E	93516	276	876	308	292	48	8
MPFS095T-1FCVG784I	93516	276	876	308	292	48	8
MPFS095T-FCSG325E	93516	80	876	308	292	48	8
MPFS095T-FCSG325I	93516	80	876	308	292	48	8

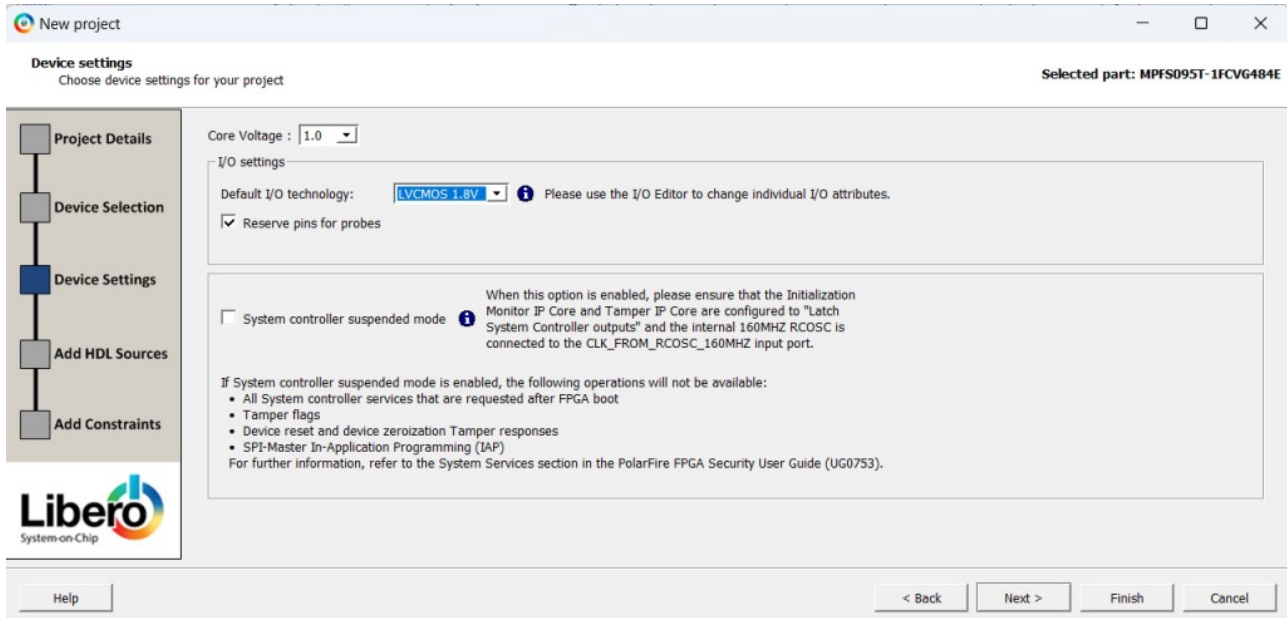
Libero
System-on-Chip

Help < Back Next > Finish Cancel

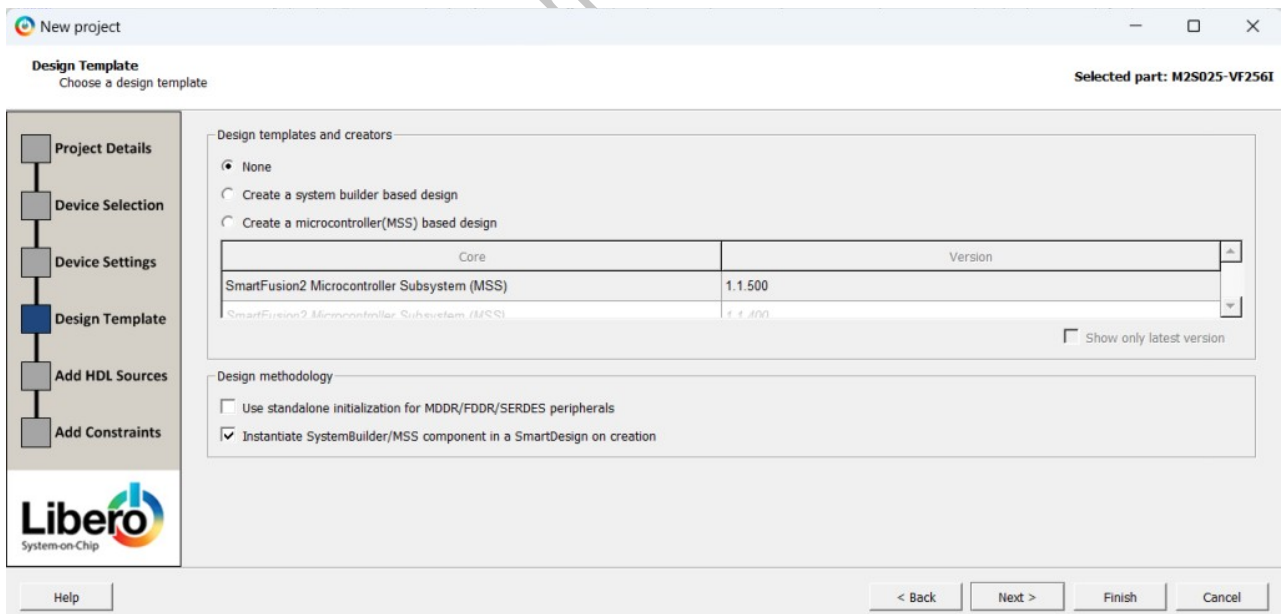
NOTA: si no te has dado cuenta entre las dos últimas imágenes ha desaparecido la opción Design Templates, esto es debido a que se ha seleccionado para programar un SoC que no tiene un Cortex-M3 (la familia que tiene el Cortex-M3 es la de la SmartFusion2), si se hubiera elegido una SmartFusion2 esa pestaña es la que permite programar el SoC (para las PolarFire SoC, se hace de otra manera).

La siguiente pestaña nos pregunta por la tensión del core y por la tensión por defecto de los pines, esto influye a la hora de declarar los pines. También nos pregunta si queremos reservar pines para depuración, *las formas de depurar con Libero las explicaré en otra entrada.*

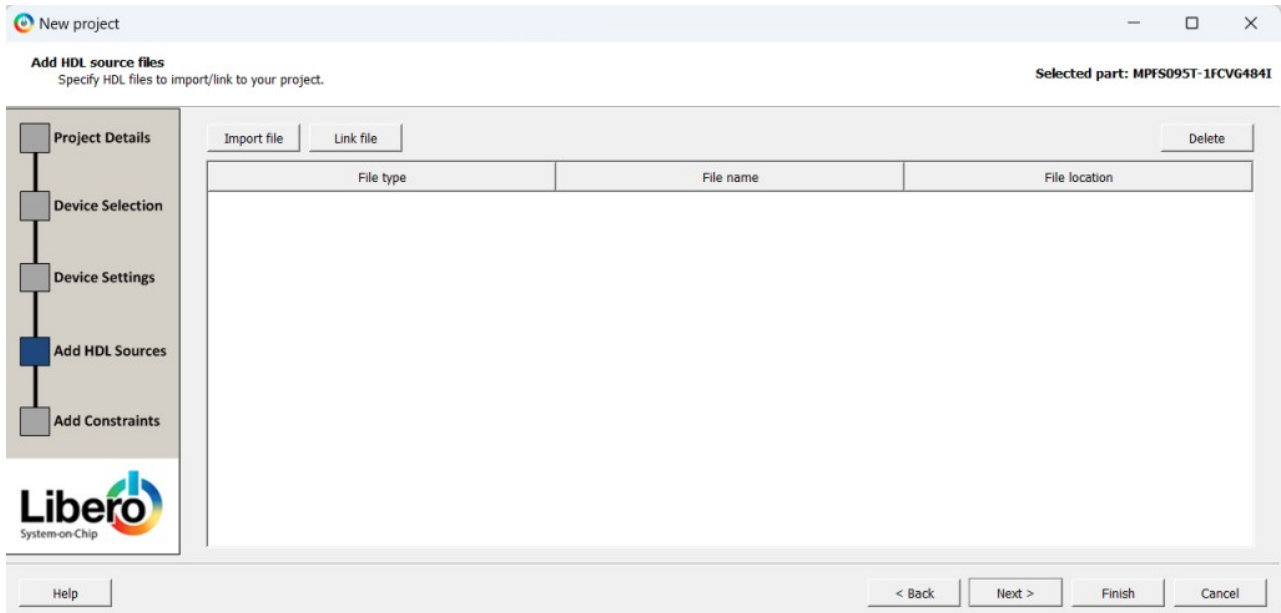
NOTA: en caso de no poder cambiar la tensión de los pines a la hora de declararlos, el problema está en la tensión definida por defecto, para acceder a esta opción para cambiarla una vez creado el proyecto, se tiene que ir a opción *Project*, y dentro buscar *Project Settings*, y después *Device settings*.



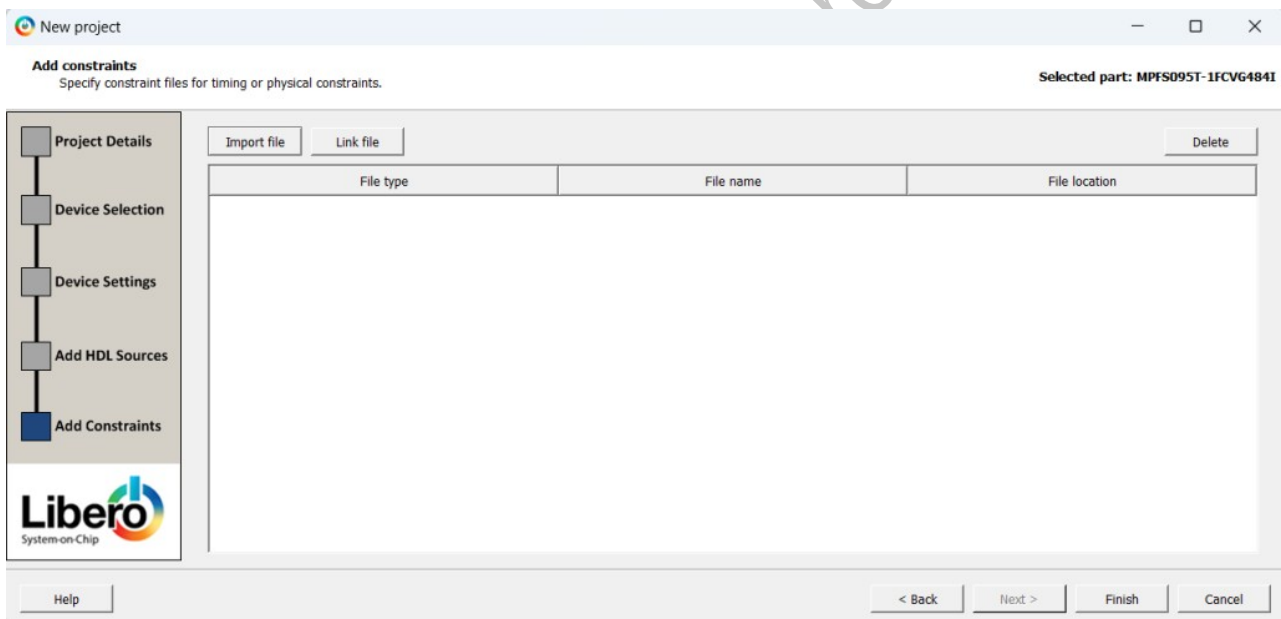
Si hubiéramos elegido programar un SoC de la familia de las SmartFusion2, nos aparecería esta pestaña, donde nos permite elegir programar la SmartFusion2 como FPGA (*None*) o como SoC, dándonos dos opciones, permitiéndonos preconfigurarlo o configurarlo más adelante.



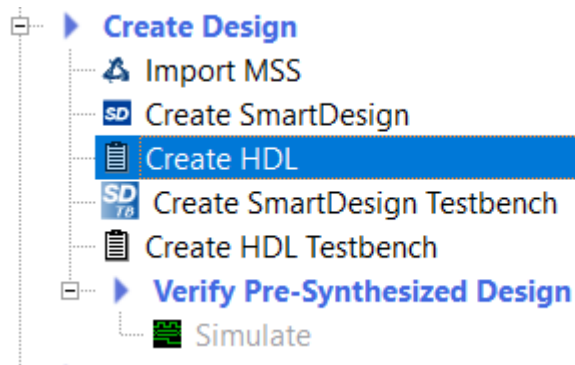
La penúltima pestaña nos pregunta si queremos añadir o enlazar una carpeta con código fuente.



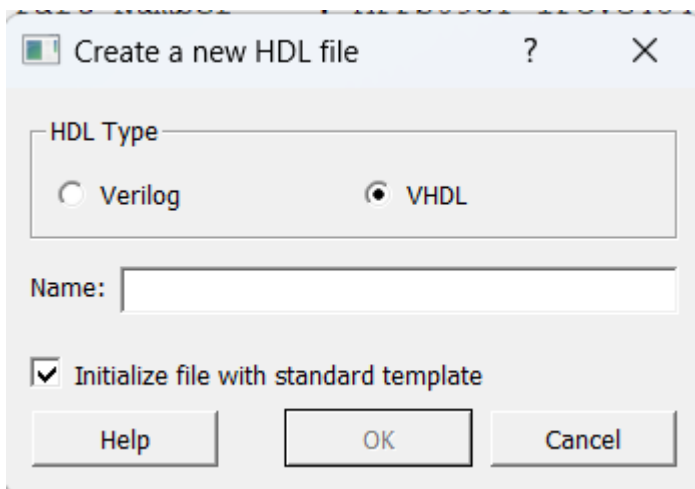
La última pestaña nos pregunta si queremos añadir o enlazar un fichero con restricciones.



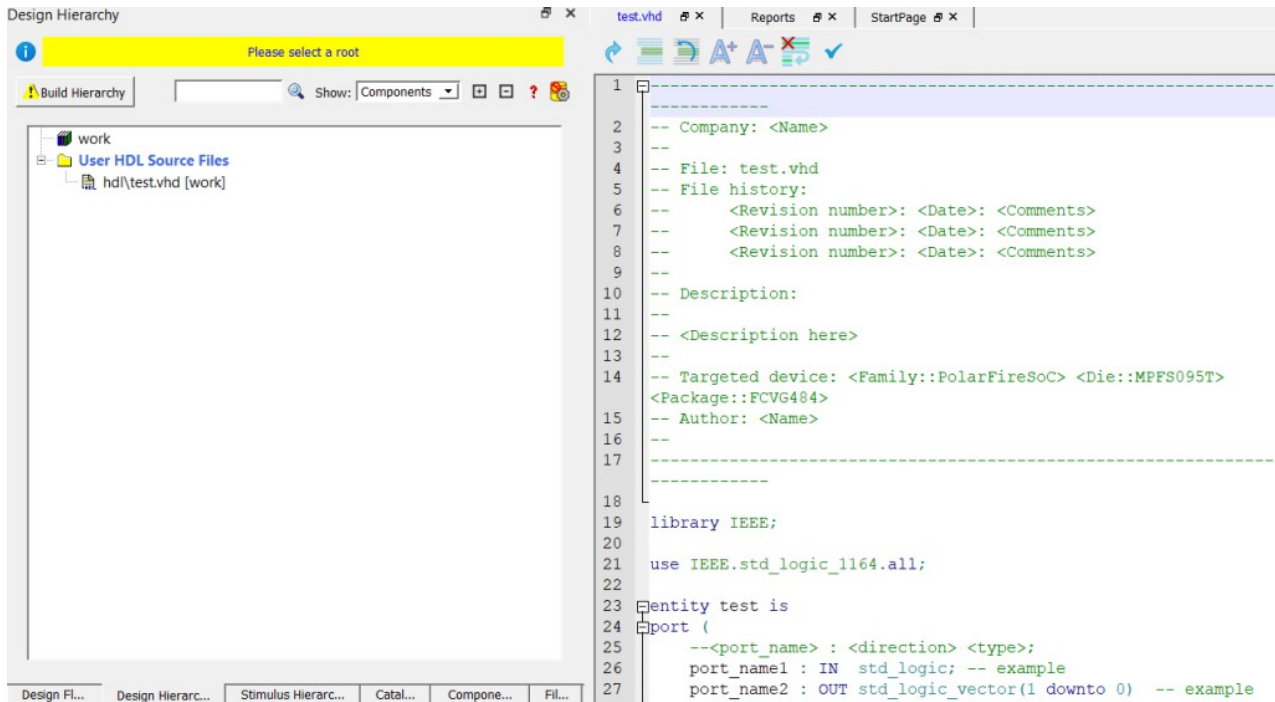
Una vez creado el proyecto, el primer paso es crear un fichero HDL, para ello pinchamos en la opción *Create HDL*.



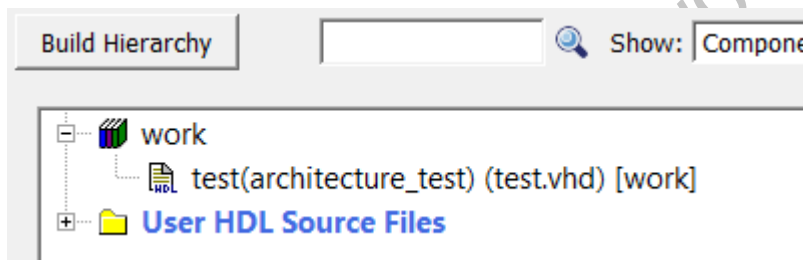
Lo siguiente que nos pregunta es por el nombre del fichero que se quiere crear y el lenguaje utilizado.



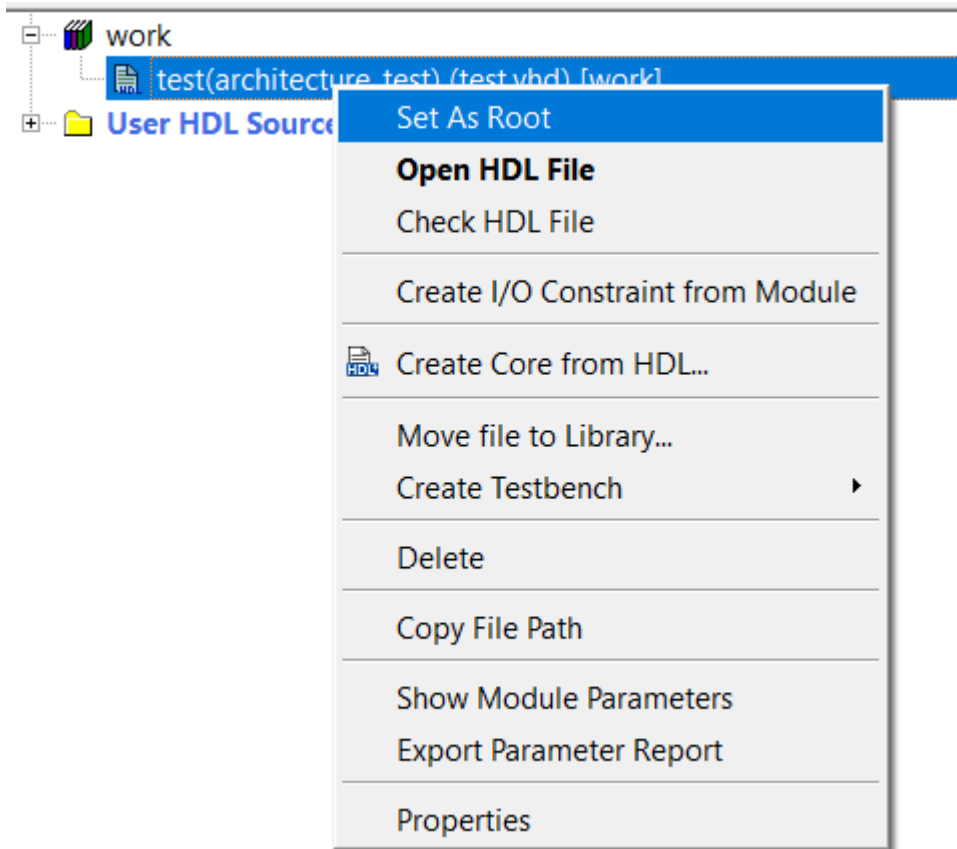
Una vez creado, en la pestaña *Design Hierarchy*, nos aparece el fichero creado, pero nos aparece un mensaje que pone *Please select a root*, esto es debido a que no hemos seleccionado ningún fichero como principal.



Desde el punto en el que estamos ahora mismo, primero tenemos que pulsar en el botón *Build Hierarchy*, al pulsarlo nos sube el fichero HDL que hemos creado a la librería *work*.



Después, le damos clic derecho *Set As Root*, y desaparece el mensaje.

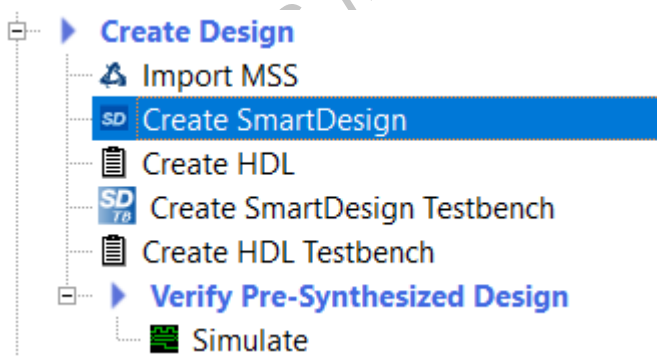


A partir de ahora ya podemos desarrollar el FW. Cada nuevo fichero que se añada, requiere de volver a pulsar el botón *Build Hierarchy*.

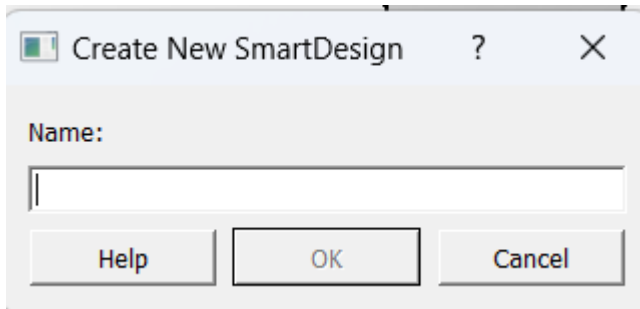
Trabajo con bloques

En Libero hay otra opción de trabajo cuando se trabaja con FW, y es el uso de bloques gráficos.

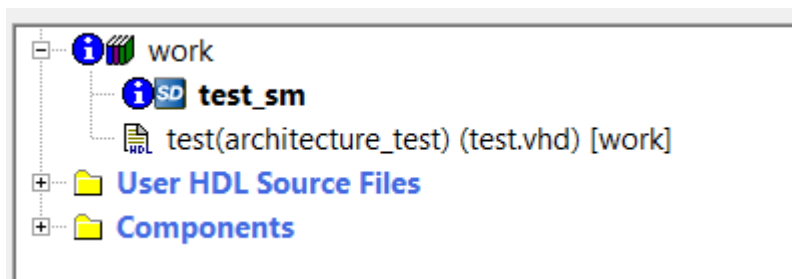
Para trabajar con ellos primero se tiene que crear un marco, Libero lo llama SmartDesign.



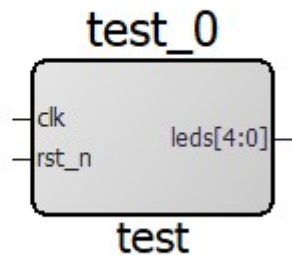
Al crear un SmartDesign nos pide un nombre.



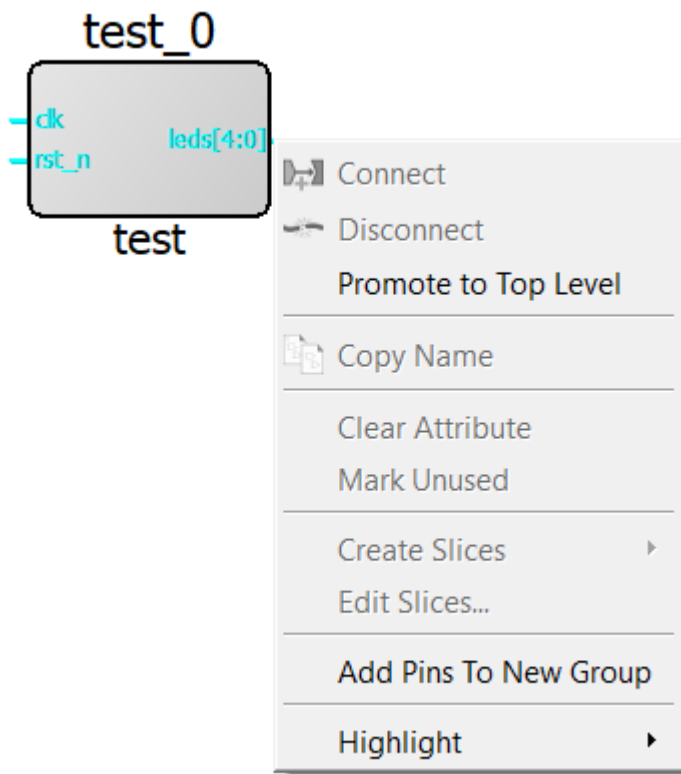
Al crear el SmartDesign, nos aparecen el SmartDesign al mismo nivel que el código HDL.



Primero hay que marcar el SmartDesign como *root* y después se coge el fichero FW y se arrastra al SmartDesign. Y nos aparece el bloque, pero sin conexiones al exterior.

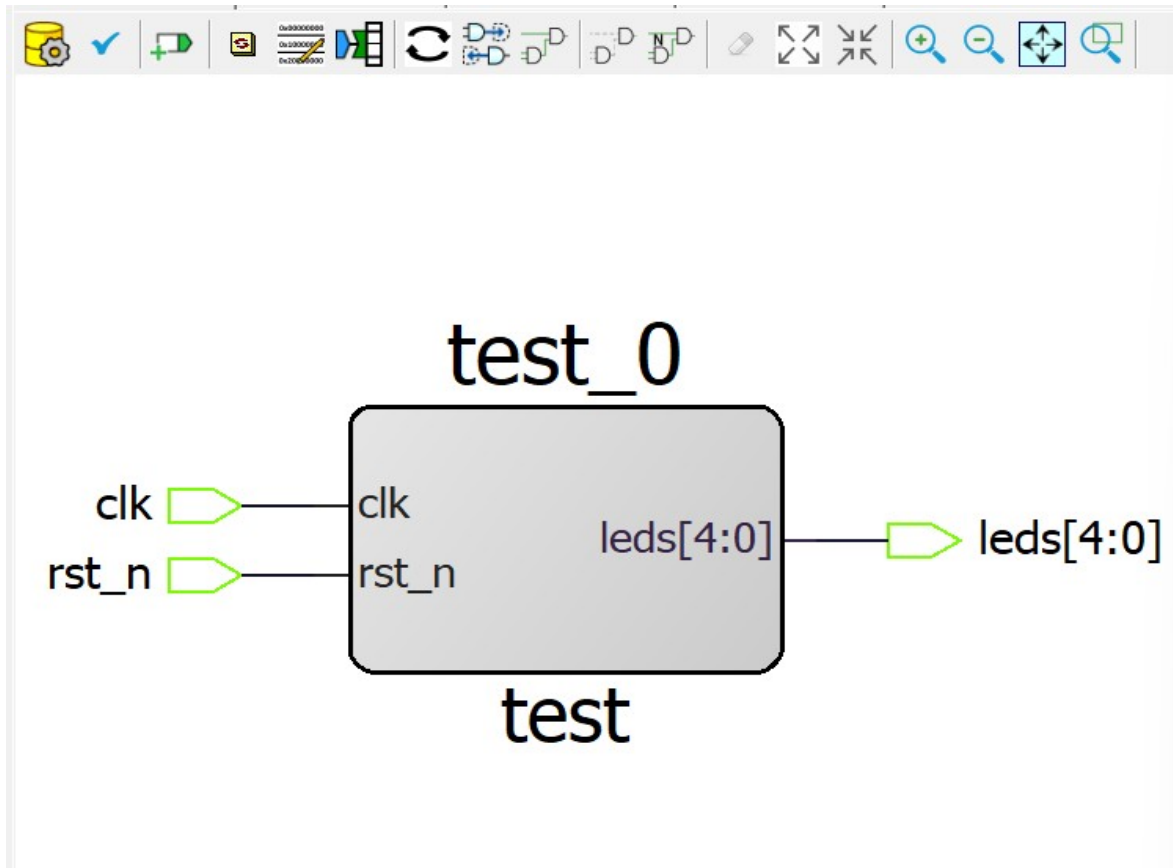


Pero pinchando en los pines nos aparece una opción llamada *Promote to Top Level*.

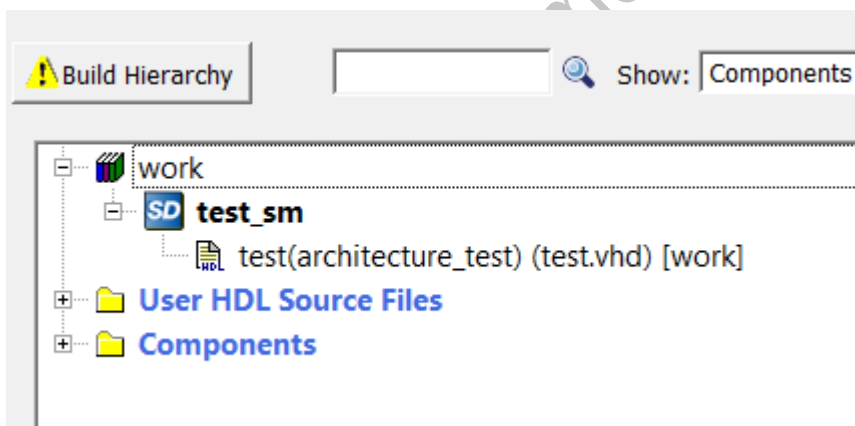


Una vez creados los pines, tenemos que darle al icono amarillo con el engranaje de arriba a la izquierda. Una vez creados los pines, tenemos que darle al icono amarillo con el engranaje de arriba a la izquierda.

Si se produce algún problema al darle nos informará abajo de los errores que se han producido, mayoritariamente, estos errores se deben a entradas no conectadas, entonces, hay que especificar el valor que van a tener.



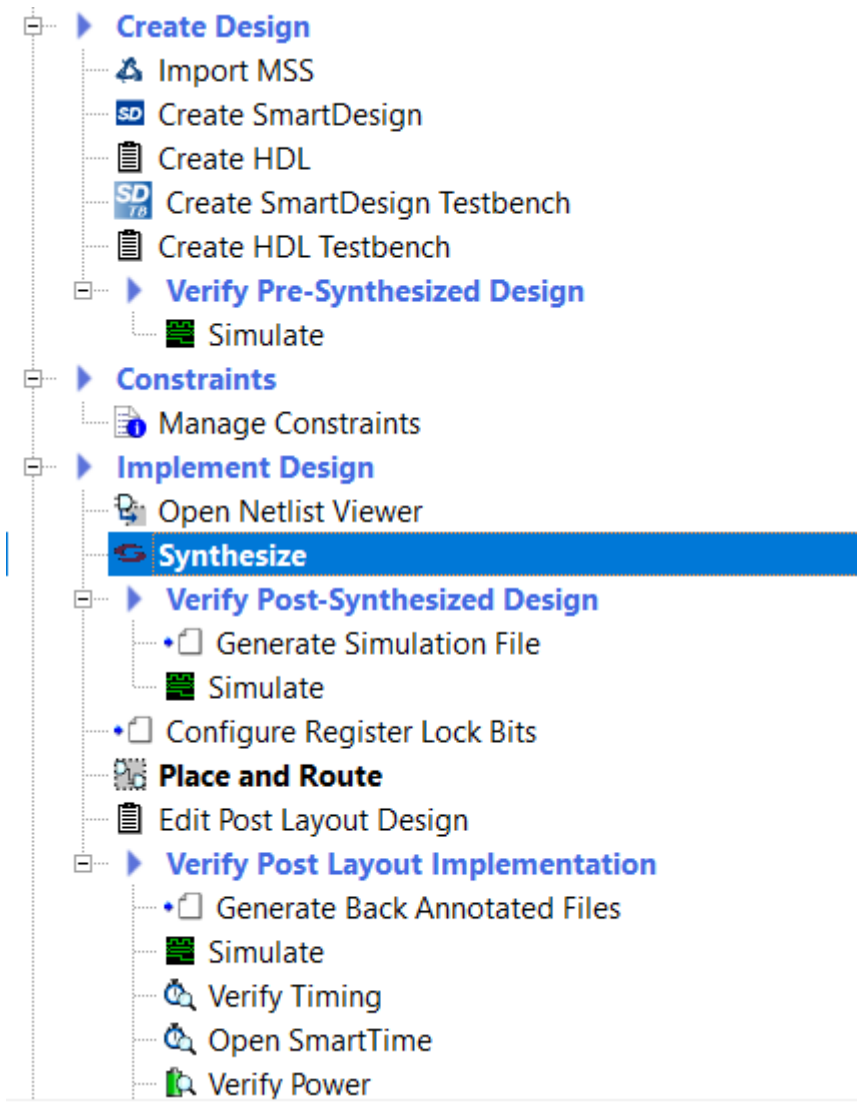
Ese icono hace que el proyecto se estructure.



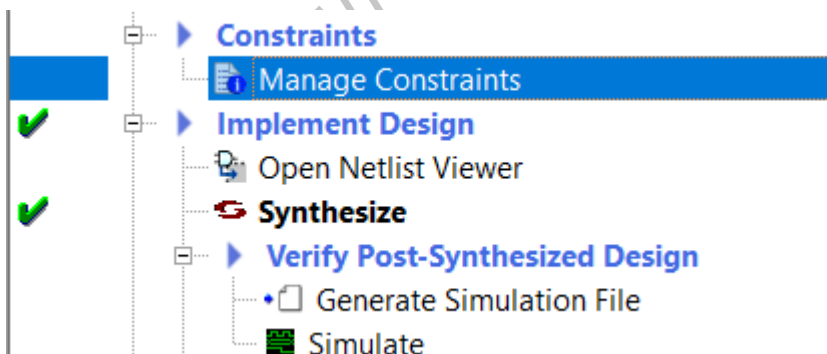
Ahora solo falta darle al botón *Build Hierarchy* para que podamos sintetizar el proyecto.

Generar el bitstream

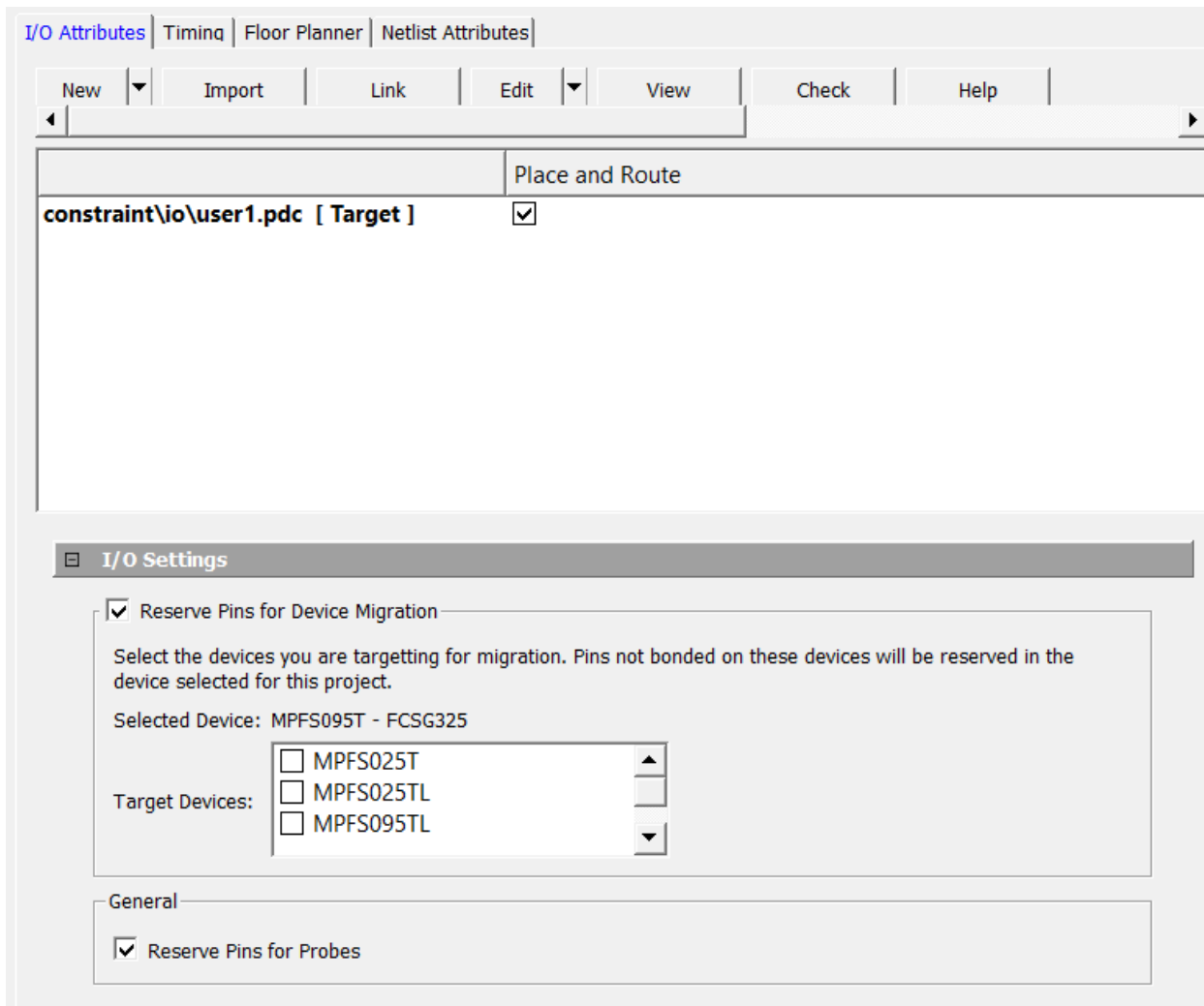
Una vez lo tengamos desarrollado, volviendo a *Design Flow*, ya podemos sintetizar el proyecto. Sintetizar el proyecto también nos permite desbloquear la selección de pines de la FPGA, porque los pines los recoge de la síntesis.



Una vez nos indique que las síntesis está correcta, abrimos la opción *Manage Constraints*.



Esta opción nos permite manejar las diferentes restricciones del proyecto, tanto de pines como de tiempos, etc. Las restricciones de pines generan un fichero *PDC*, mientras que las restricciones de tiempos generan un fichero *SDC*.

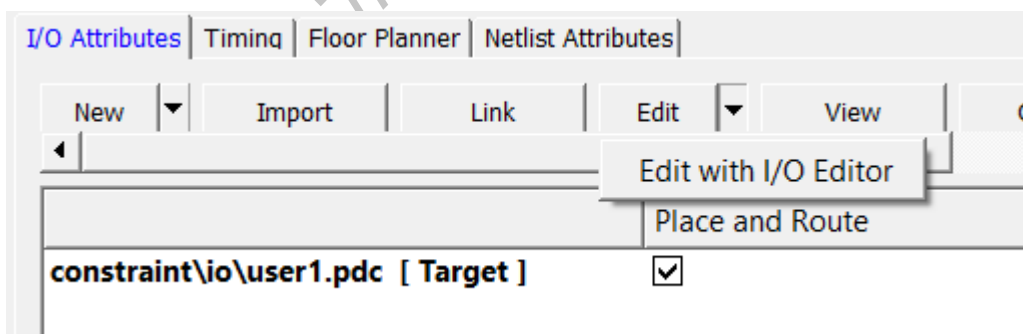


Para crear una restricción de pines tenemos dos opciones:

- Hacer las restricciones a mano, lo cuál puede ser bastante complejo si no se sabe cuál es la estructura del fichero PDC.

```
24
25 set_io -port_name clk \
26     -pin_name R18 \
27     -fixed true \
28     -io_std LVCMOS18 \
29     -DIRECTION INPUT
30
31
32 set_io -port_name {leds[0]} \
33     -pin_name T18 \
34     -fixed true \
35     -io_std LVCMOS18 \
36     -DIRECTION OUTPUT
37
38
39 set_io -port_name {leds[1]} \
40     -pin_name V17 \
41     -fixed true \
42     -io_std LVCMOS18 \
43     -DIRECTION OUTPUT
44
45
46 set_io -port_name {leds[2]} \
47     -pin_name U20 \
48     -fixed true \
```

- La otra opción es utilizar la herramienta interactiva que da Libero. Para acceder a ella, se abre el desplegable de *Edit* y se pincha en la opción *Edit with I/O Editor*.



Al abrir esta opción, se abre una herramienta interactiva en la que el usuario puede seleccionar pin que quiere para el puerto del FW.

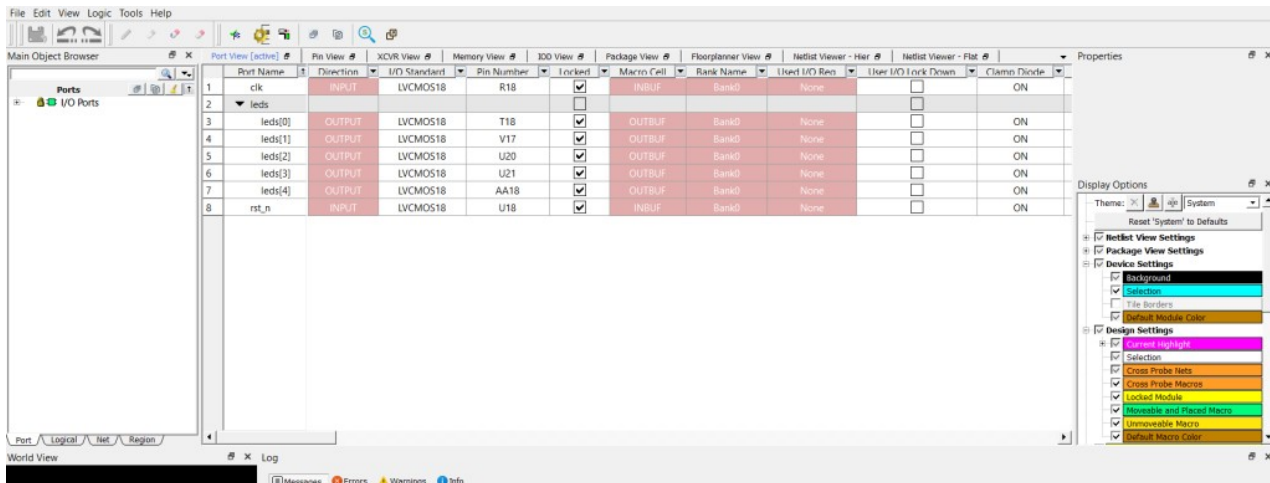
NOTA: esta herramienta funciona mediante bancos, de tal manera que si seleccionas un pin a una tensión específica, pone todo el banco a esa misma tensión. Además, si seleccionas mal un pin y se

<https://soceame.wordpress.com/2024/12/01/proyecto-basico-con-libero/>

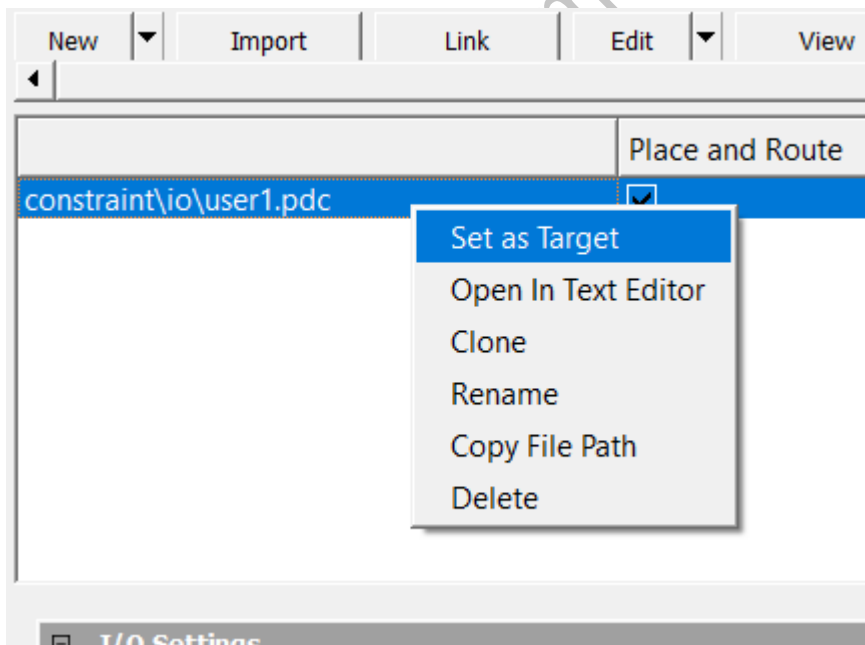
lo asignas a otro distinto puedes dejar todo el banco marcado a esa tensión, y el selector de pines solo te dejará utilizar pines de ese banco. Es conveniente tener cuidado.

NOTA 2: para solucionar el problema de preselección de los pines tienes que borrar de forma manual el pin y darle intro, para que deje de marcar ese pin. Muchos de los problemas de esta herramienta se solucionan borrando manualmente el pin.

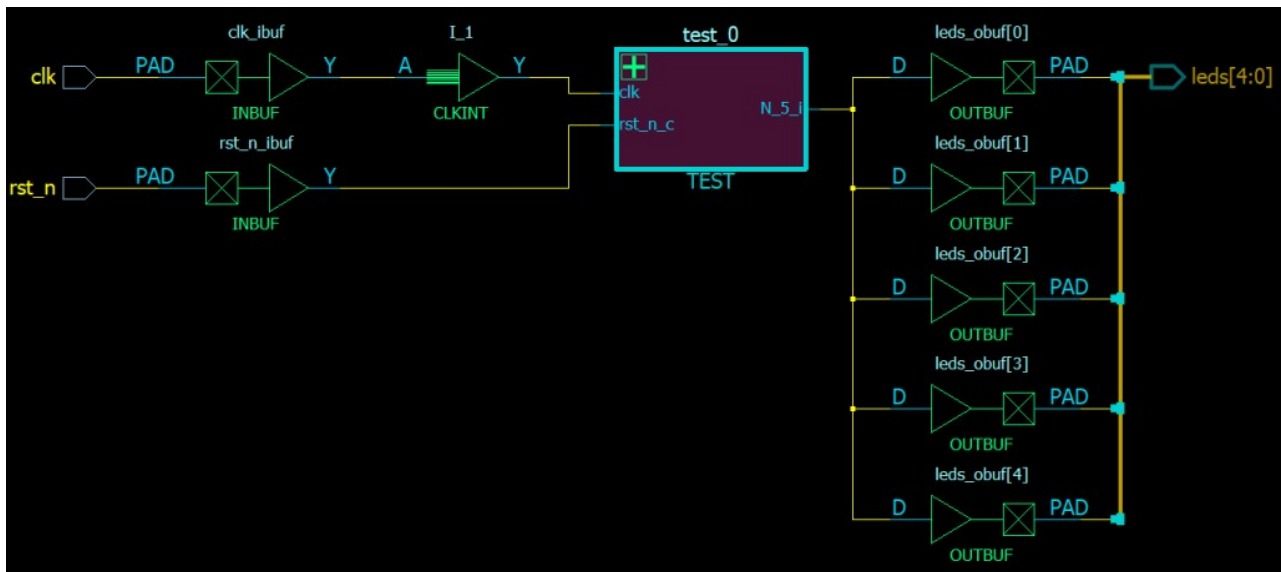
NOTA 3: en caso de no poder cambiar la tensión de los pines que figura por defecto, se tiene que ir a opción *Project*, y dentro buscar *Project Settings*, y después *Device settings*.



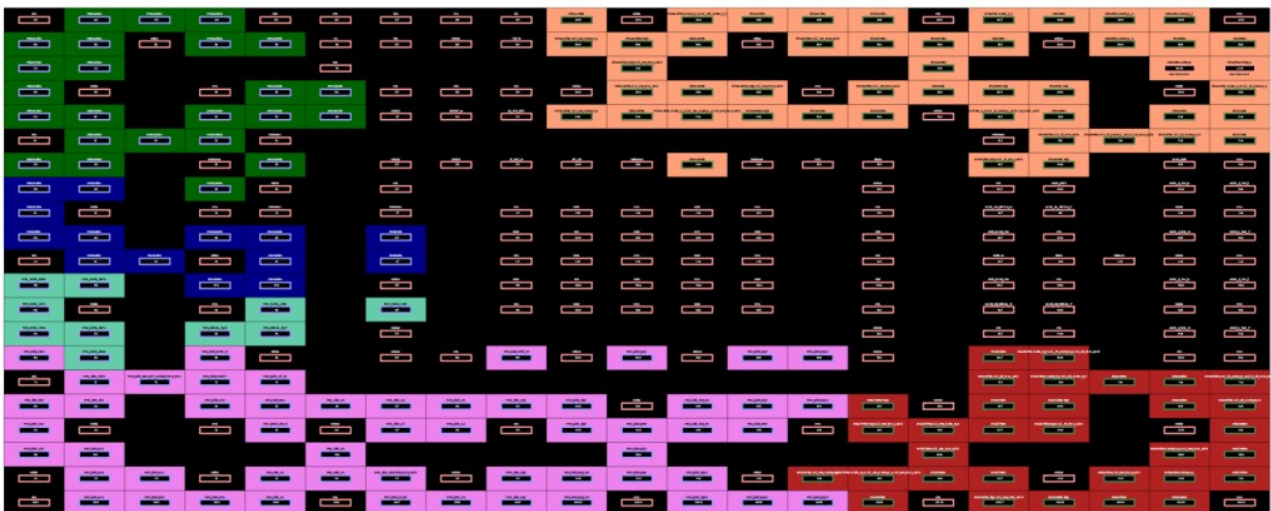
Una vez terminado, se le tiene que dar al icono de guardar, y entonces, crea el fichero de restricción de pines del proyecto. Lo siguiente que hay que hacer es marcar el fichero de pines como **Target**.



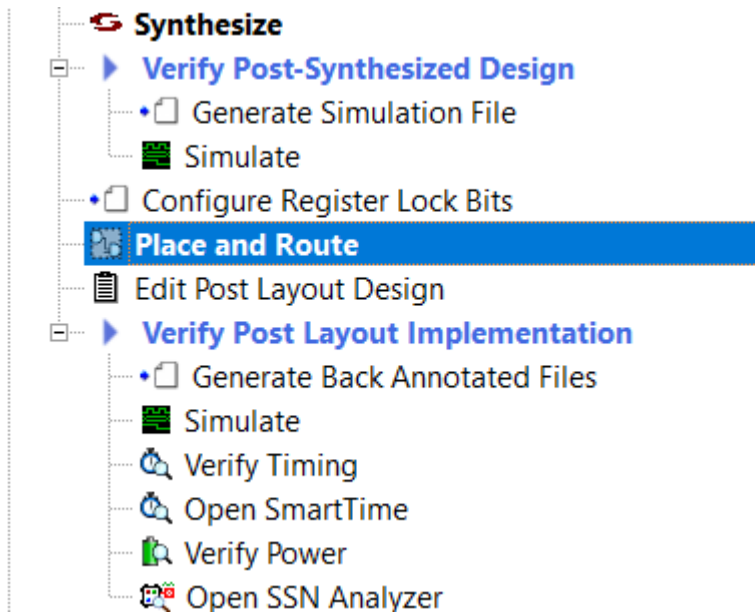
NOTA: Este editor también nos permite ver el modelo sintetizado que nos ha generado Libero, está en la pestaña *Netlist Viewer* (si aparece un mensaje en azul, solo hay que darle clic para que aparezca).



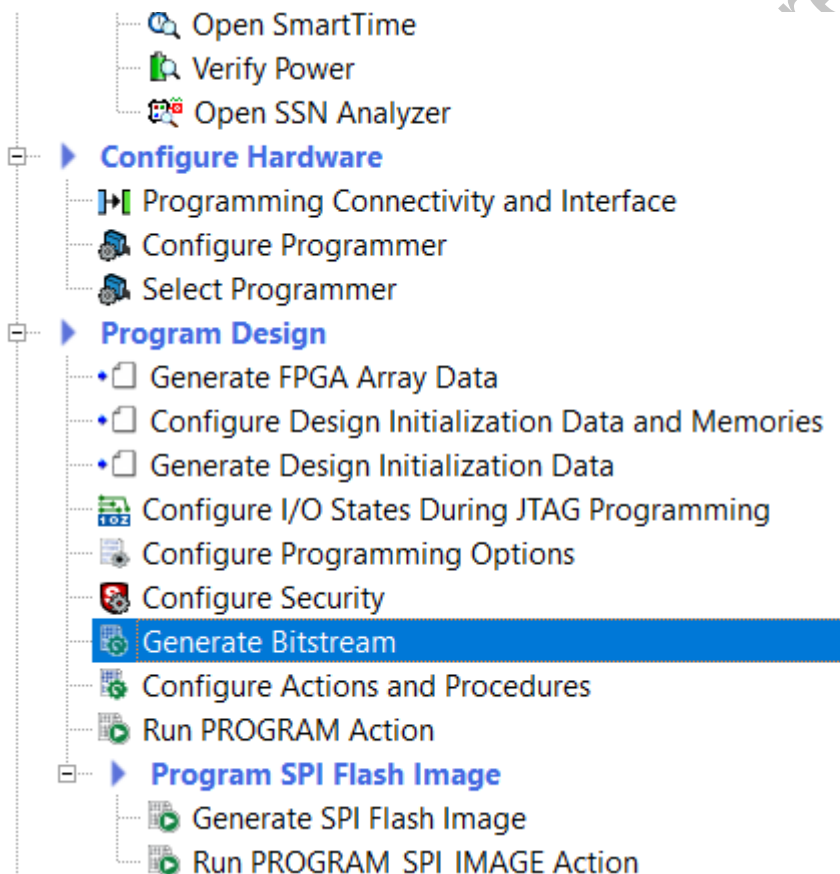
También permite ver otras cosas, como la ubicación de pines del chip o el tipo de pines seleccionables del chip.



Una vez seleccionados los pines, se realiza la implementación, para ello se pincha en la opción **Place and Route**.

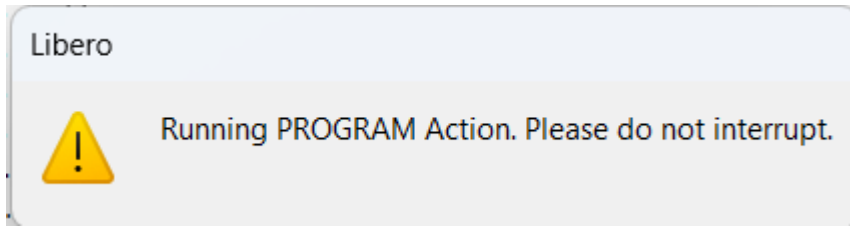


Una vez terminado se puede generar el bitstream. Para ello se marca la opción *Generate Bitstream*.

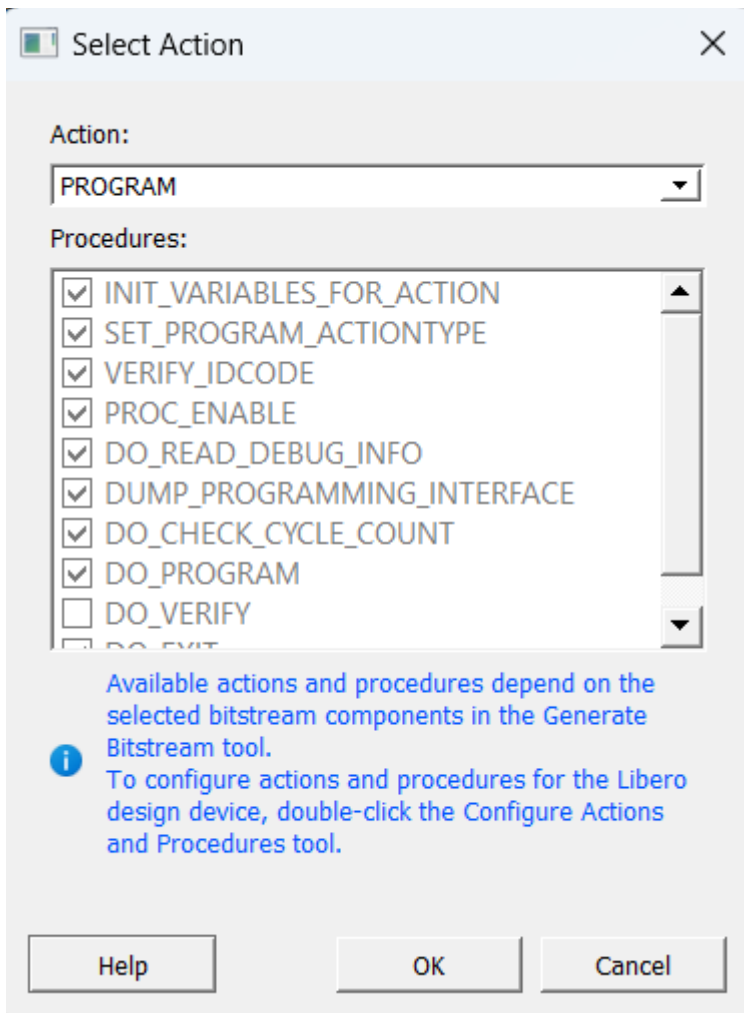


Una vez generado, ya tenemos el bitstream del proyecto generado, solo falta grabarlo en la FPGA o exportarlo.

- Para grabarlo en la FPGA se tiene que enganchar la FPGA al ordenador y pinchar en la opción *Run PROGRAM Action*. Mientras se está grabando, aparece esta pestaña y debajo en Libero a aparece una barra con el estado de carga del bitstream.



La opción de *Run PROGRAM Action* también tiene la opción de realizar más acciones, para ello, clic derecho en *Run PROGRAM Action* y darle a la opción *Select Action*. La acción de PROGRAM es la de grabar la FPGA.



- Para exportarlo se tiene que ir a las opciones de exportación de Libero. Aquí aparecen varias opciones, las principales son la de exportar el bitstream y la de exportar para generar un proyecto en *FlashPro Express*.



Las dos opciones de exportación están descritas al final de esta entrada anterior.

<https://soceame.wordpress.com/2024/11/24/tutorial-sobre-el-flashpro-express-instalacion-y-ejecucion/>