
4 FLUJO DE DISEÑO XILINX

4.1 DISEÑO ESTÁNDAR

A continuación se detalla el proceso que sigue el diseño desde el fichero ‘.vhd’, hasta obtener el fichero ‘.bit’ que configura la FPGA mediante el uso de las herramientas proporcionadas por Xilinx.

Estos programas están destinados a realizar un diseño de la forma más fácil y transparente al usuario. Aunque incorporan ciertas funcionalidades que serán explotadas para el desarrollo del control del rutado automático.

Una vez se tiene el proyecto en ficheros que contienen el correspondiente código VHDL se obtiene una *netlist* que contiene ya cierta información sobre las características de la tecnología objetivo. La *netlist* que se obtiene usando el programa *FPGA Express*, tiene el formato común ‘.edf’.

Mediante el programa *Ngdbuild* se convierte a un formato de *netlist* interno de Xilinx denominado ‘.ngd’. En este punto no se encuentran las restricciones físicas que imponen los pins de entrada/salida, para ello es necesario utilizar el fichero ‘.ngd’ para poder construir el fichero de restricciones ‘.ucf’.

Se puede realizar mediante la edición del fichero UCF o mediante el programa *Constrain Editor*. Una vez establecidas las restricciones se debe que construir de nuevo el fichero NGD con las nuevas restricciones contenidas en el formato UCF.

Mediante *map* se obtiene el formato NCD a partir del fichero ‘.ngd’. Se realiza un mapeo de las funciones lógicas a CLBs e IOBs, obteniendo en NCD una implementación física del diseño.

Las siguientes herramientas utilizan como formato de entrada y salida el mismo, el formato NCD. Por tanto existe una versión inicial del mismo que no tiene realizado el *placement & routing* y otra posterior que sí.

La herramienta *par* realiza el *P&R*, y posteriormente se aplica la herramienta *trce* que realiza un análisis temporal. Resultando siempre un fichero ‘.ncd’ como salida de ambas herramientas.

En los últimos pasos se tiene un menor control, y los formatos utilizados son transparentes para el diseño de una implementación. Al final de ellos se obtiene el correspondiente flujo de bits para configurar la FPGA en el fichero ‘.bit’.

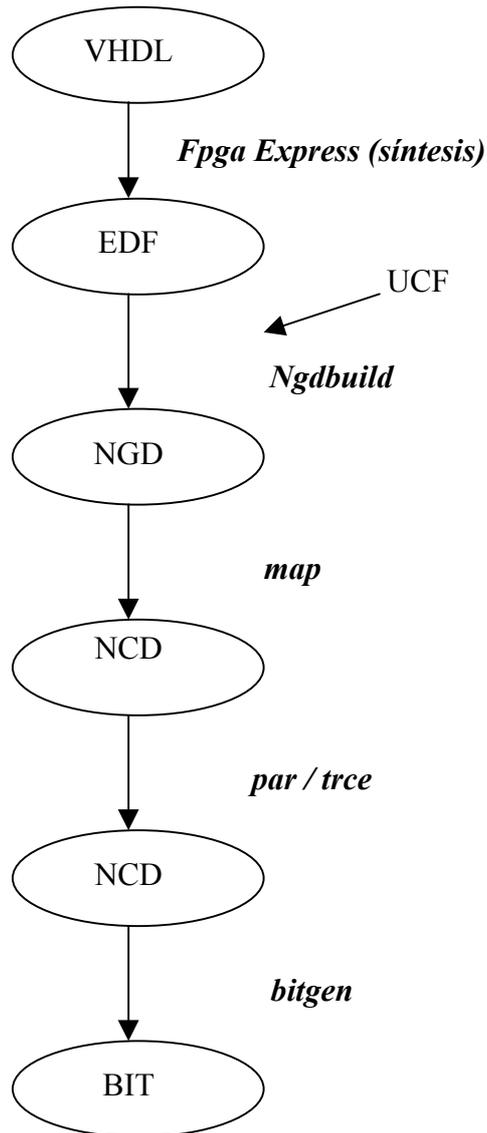


Figura 4-1: Flujo de diseño en XILINX

4.1.1 PROGRAMACIÓN

Del flujo de diseño normal se obtendrá el fichero “.bit” que contiene los valores necesarios para realizar la configuración de la FPGA Virtex. La configuración de la FPGA se realiza cargando previamente el diseño en la memoria FLASH que posee la placa de desarrollo.

Este tipo de configuración permite que la FPGA se configure automáticamente cada vez que se realiza la conexión de alimentación, y sin necesidad de recurrir a una estación de trabajo conectada mediante puerto paralelo.

Para ello, se debe configurar la CPLD integrada en la placa XSV con un programa que permite la carga en la FLASH y posteriormente la configuración de la FPGA obteniendo el flujo de datos de la memoria FLASH. Estos pasos los realiza automáticamente las herramientas proporcionadas por la placa de desarrollo.

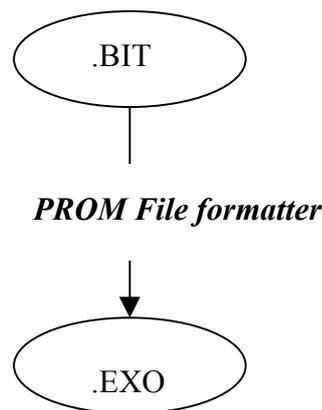


Figura 4-2: Flujo de diseño para la obtención del fichero “.exo”

Sólo hay que obtener un fichero de configuración apto para la memoria *flash* a partir del fichero “.bit”. El formato de fichero que se utiliza para programar la memoria FLASH es “.exo”. Mediante la utilidad proporcionada por Xilinx, *PROM File Formatter*, se obtiene el fichero “.exo” a partir del “.bit”.

Para programar la memoria FLASH a través de la CPLD es necesario puentear los pins p2 y p3 en el puente J31. Una vez programada con la configuración que permite configurar la memoria FLASH cada vez que se conecta la alimentación, es conveniente eliminar el puente en J23 para evitar posibles programaciones fantasmas.

4.2 MODO FLOORPLAN

Si se incorporan restricciones de área u otro tipo mediante la herramienta *Floorplanner*, el flujo de diseño comentado anteriormente cambia, para poder incorporar las nuevas restricciones al fichero de configuración final.

Al incorporar al flujo de diseño un *floorplan*, las restricciones que se han impuesto en el fichero *.ucf* quedan deshabilitadas. Es importante por tanto incorporar estas restricciones al *floorplan*.

El resultado de planificar la colocación del diseño en distintas áreas sobre la FPGA produce dos ficheros con formato texto: *.fnf*, que almacena la información del *floorplan* sobre elementos lógicos que componen el diseño (BELs), grupos de lógica y áreas que se han delimitado. El otro fichero, *.mfp.*, contiene la información y restricciones del *floorplan* en el formato de entrada necesario para la herramienta *map*.

Siguiendo este modo de diseño, se deben de realizar los siguientes pasos, que se pueden ver reflejados en la figura 4-3:

- Obtención del fichero NCD, resultado de la herramienta *map*, sin restricciones.
- Uso de *Floorplanner* para asignar las restricciones, áreas y grupos del diseño, obteniendo el fichero *.mfp*.
- Nueva ejecución de *map*, pero con la opción de *floorplan* habilitada.

```
map -p xcv800-4-hq240 -o map.ncd -fp <file.mfp> <file.ngd>
    <file.pcf>
```

- Aplicación de la herramienta *par* para obtener el *placement* del diseño (opción de rutado deshabilitada)

```
par -w -ol 2 -r map.ncd file.ncd file.pcf
```

- Aplicación de la herramienta *par* para obtener el rutado del diseño (opción *placement* deshabilitada)

```
par -w -ol 4 -p file.ncd filer.ncd file.pcf
```

- Uso de *bitgen* para obtener el flujo de bits de configuración (*.bit*).

Realizar el P&R en dos fases, permite manipular si se desea la versión del diseño sin rutar, como se hace en la técnica de control del rutado automático.

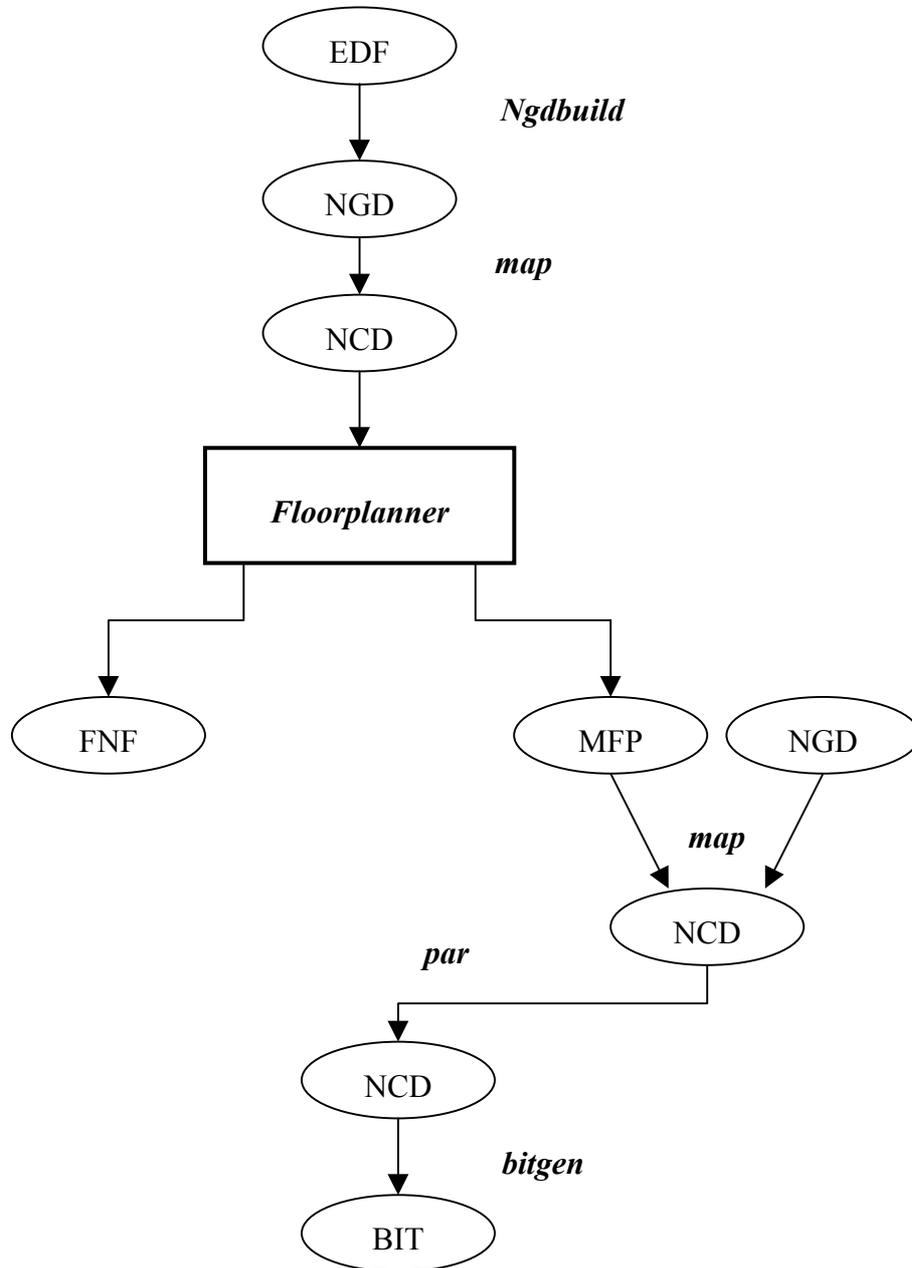


Figura 4-3: Flujo de diseño en Xilinx usando *Floorplanner*