

Cómo grabar la flash de una FPGA en Quartus

Creador: David Rubio G.

Entrada: <https://soceame.wordpress.com/2025/01/13/como-grabar-la-flash-de-una-fpga-en-quartus/>

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

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

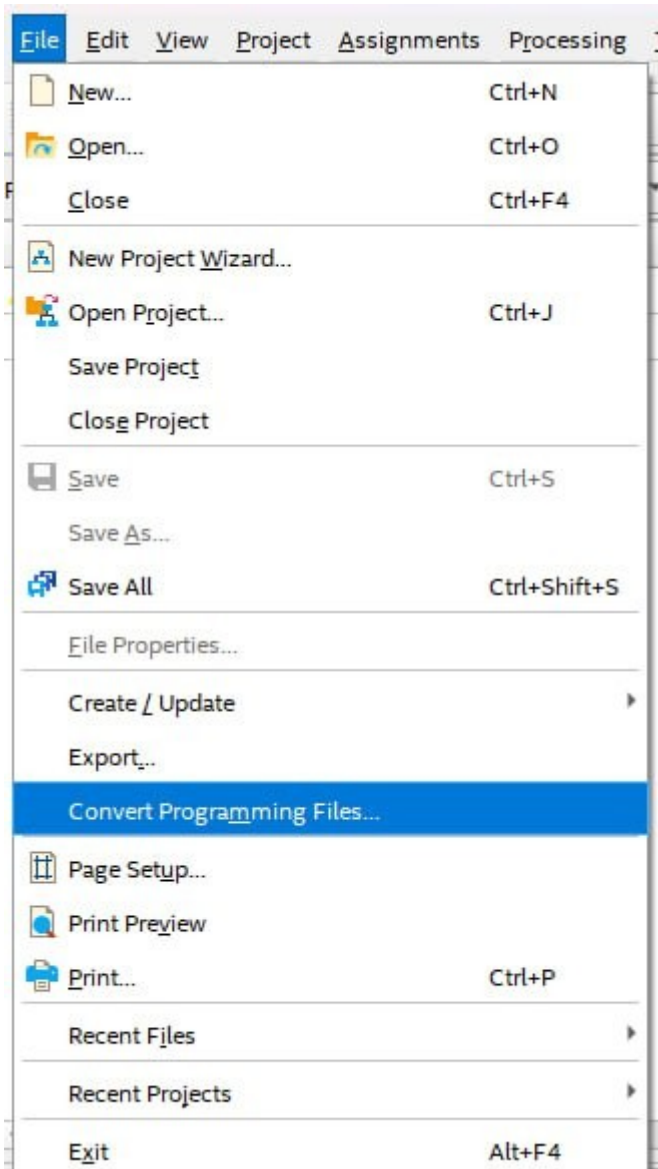
Fecha última modificación: 24/02/2025

Esta entrada parte desde esta última entrada.

<https://soceame.wordpress.com/2025/01/13/proyecto-basico-en-quartus/>

Para poder grabar la memoria flash de una FPGA de Quartus necesitamos un fichero de tipo JIC (*JTAG Indirect Configuration File*).

Para generarlo vamos a File y nos vamos a la opción de *Convert Programming Files...*



Una vez se ha abierto la pestaña, le decimos que queremos generar un fichero .jic.

The screenshot shows the 'JTAG Indirect Configuration File (.jic)' generation window in Intel Quartus Prime. The window has a menu bar with 'File', 'Tools', and 'Window'. A search bar labeled 'Search Intel FPGA' is in the top right. The main area contains instructions: 'Specify the input files to convert and the type of programming file to generate. You can also import input file information from other files and save the conversion setup information created here for future use.'

Conversion setup files

Buttons: 'Open Conversion Setup Data...', 'Save Conversion Setup...'

Output programming file

Programming file type: JTAG Indirect Configuration File (.jic) (dropdown)

Options/Boot info... (button)

Configuration device: EPCQL1024 (dropdown) ... Mode: Active Serial (dropdown)

File name: output_files/output_file.jic (text field) ... (button)

Advanced... (button)

Remote/Local update difference file: NONE (dropdown)

☒ Create Memory Map File (Generate output_file.map)

☐ Create CvP files (Generate output_file.periph.jic and output_file.core.rbf)

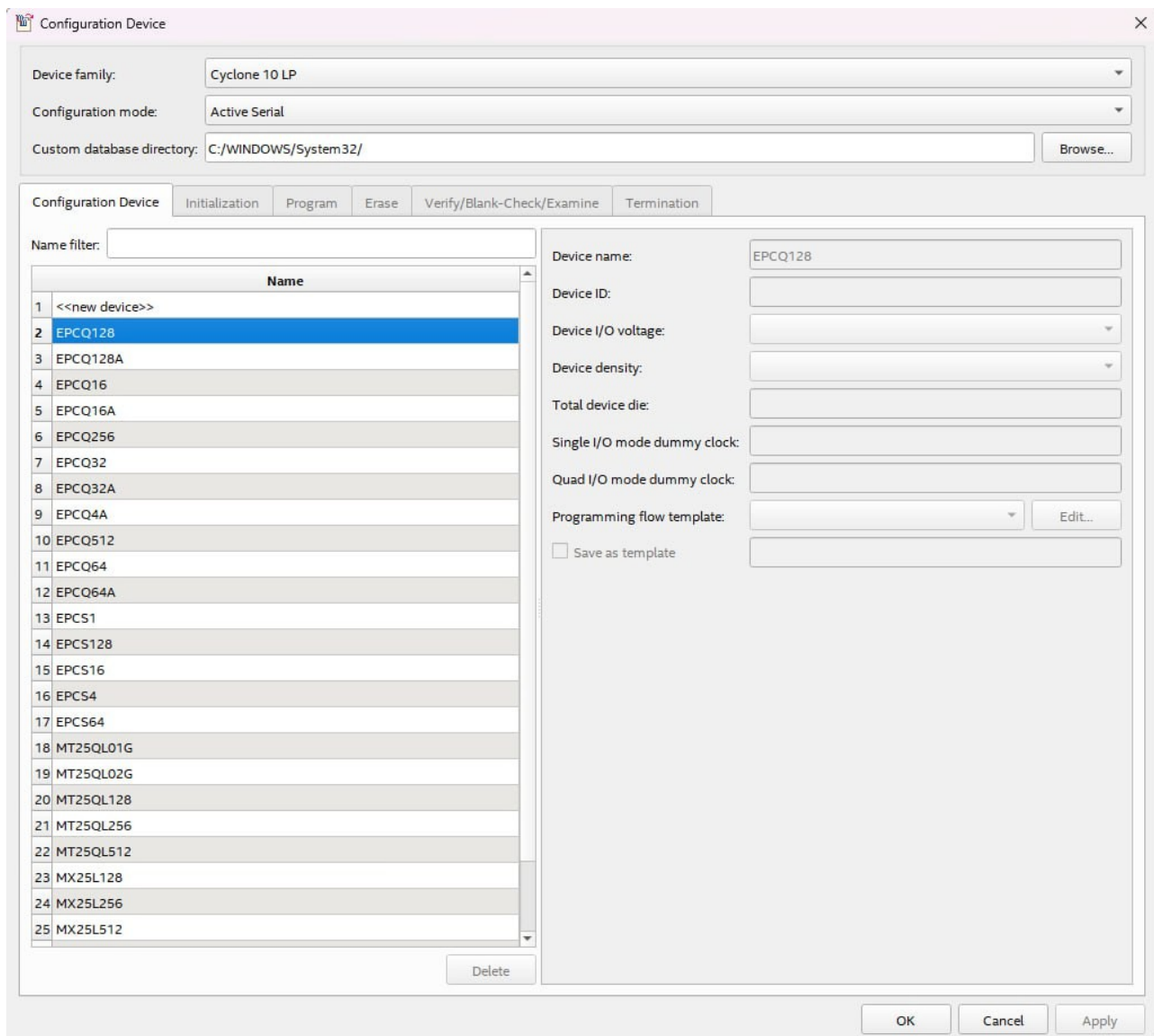
☐ Create config data RPD (Generate output_file_auto.rpd)

Input files to convert

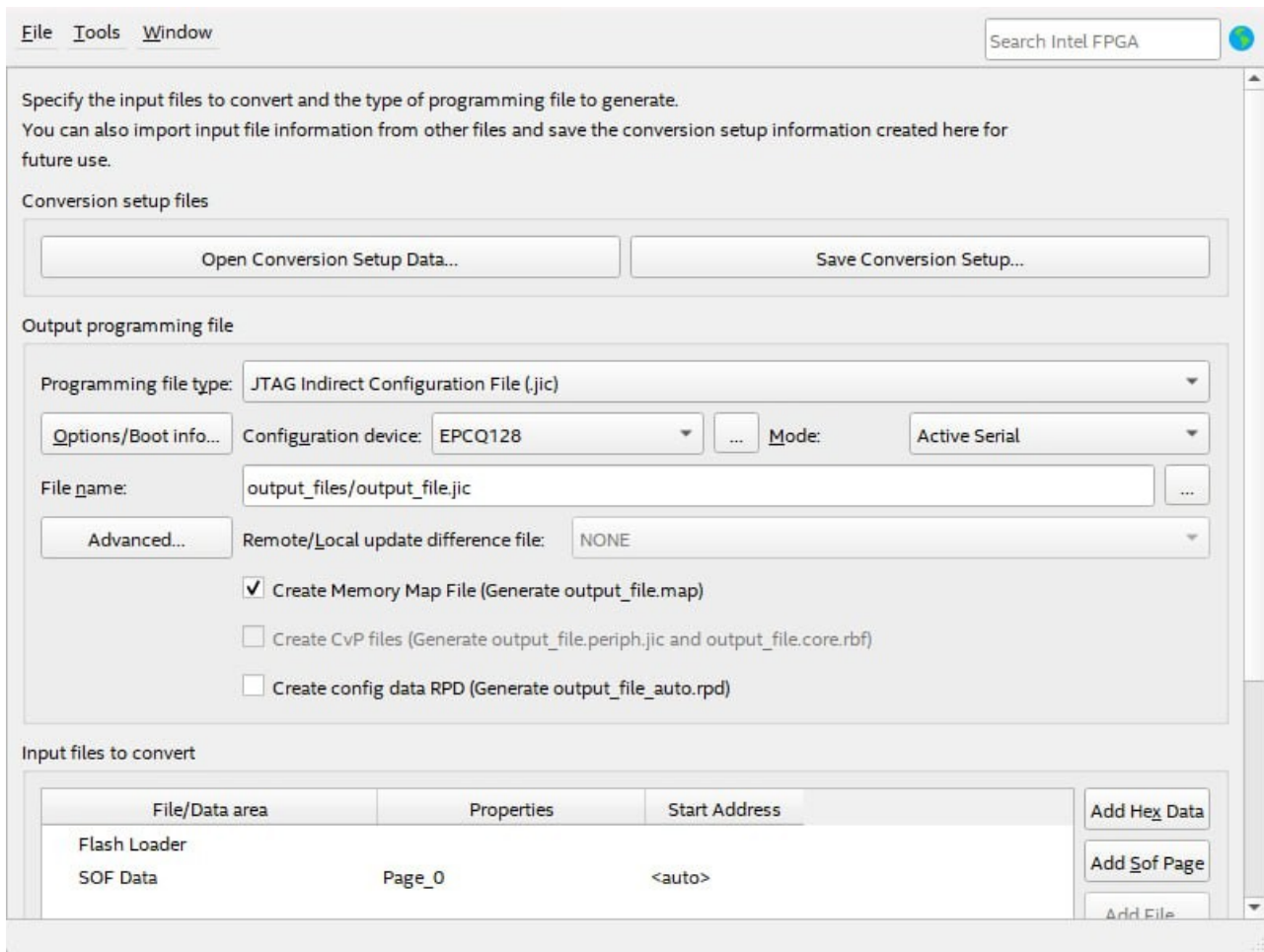
File/Data area	Properties	Start Address
Flash Loader		
SOF Data	Page_0	<auto>

Buttons on the right: 'Add Hex Data', 'Add Sof Page', 'Add File'.

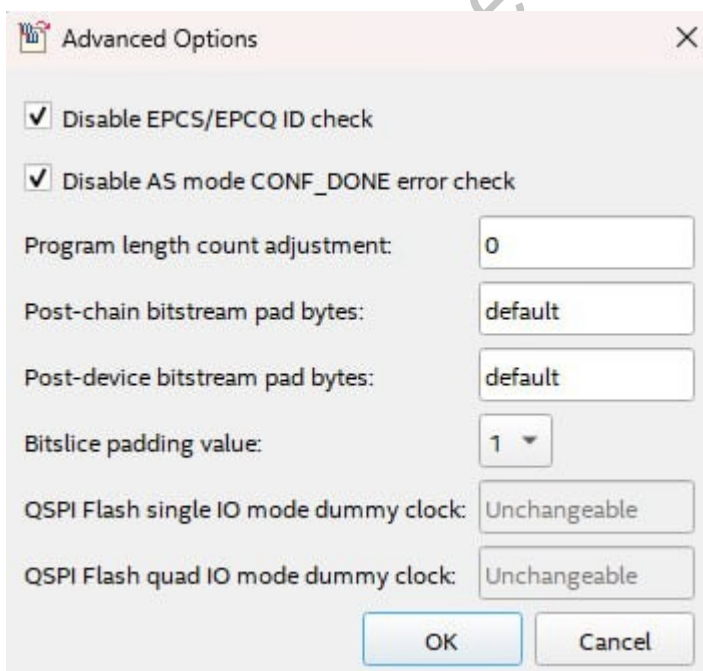
Ahora tenemos que cambiar el dispositivo que queremos configurar, para ello en *Configuration device* le damos a los tres puntos y se nos abre una pestaña donde buscar el dispositivo.



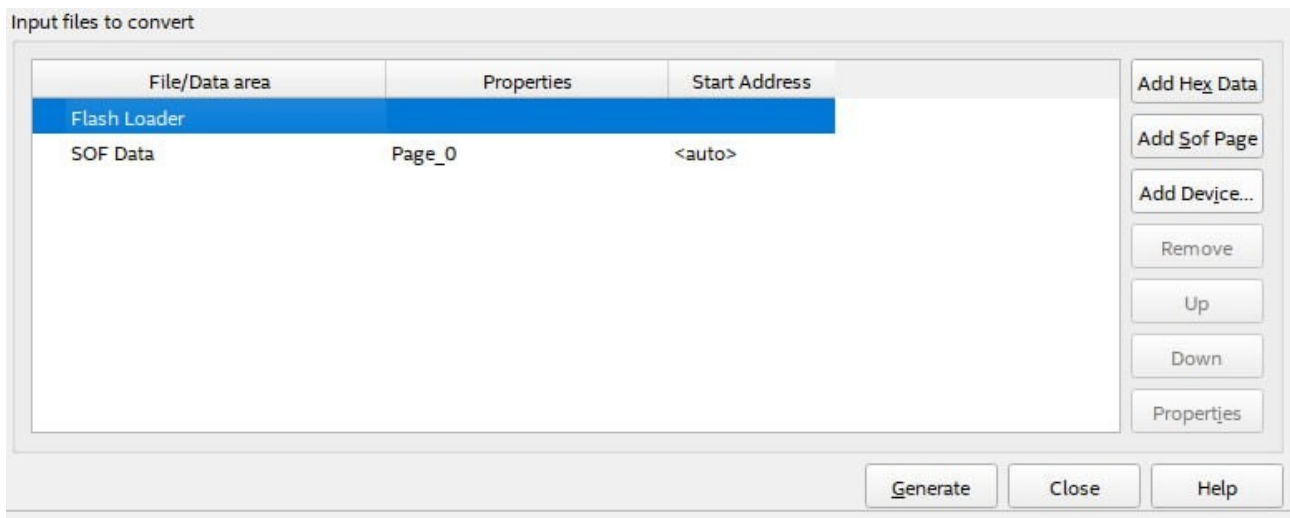
Una vez lo tenemos seleccionado podemos continuar con la configuración.



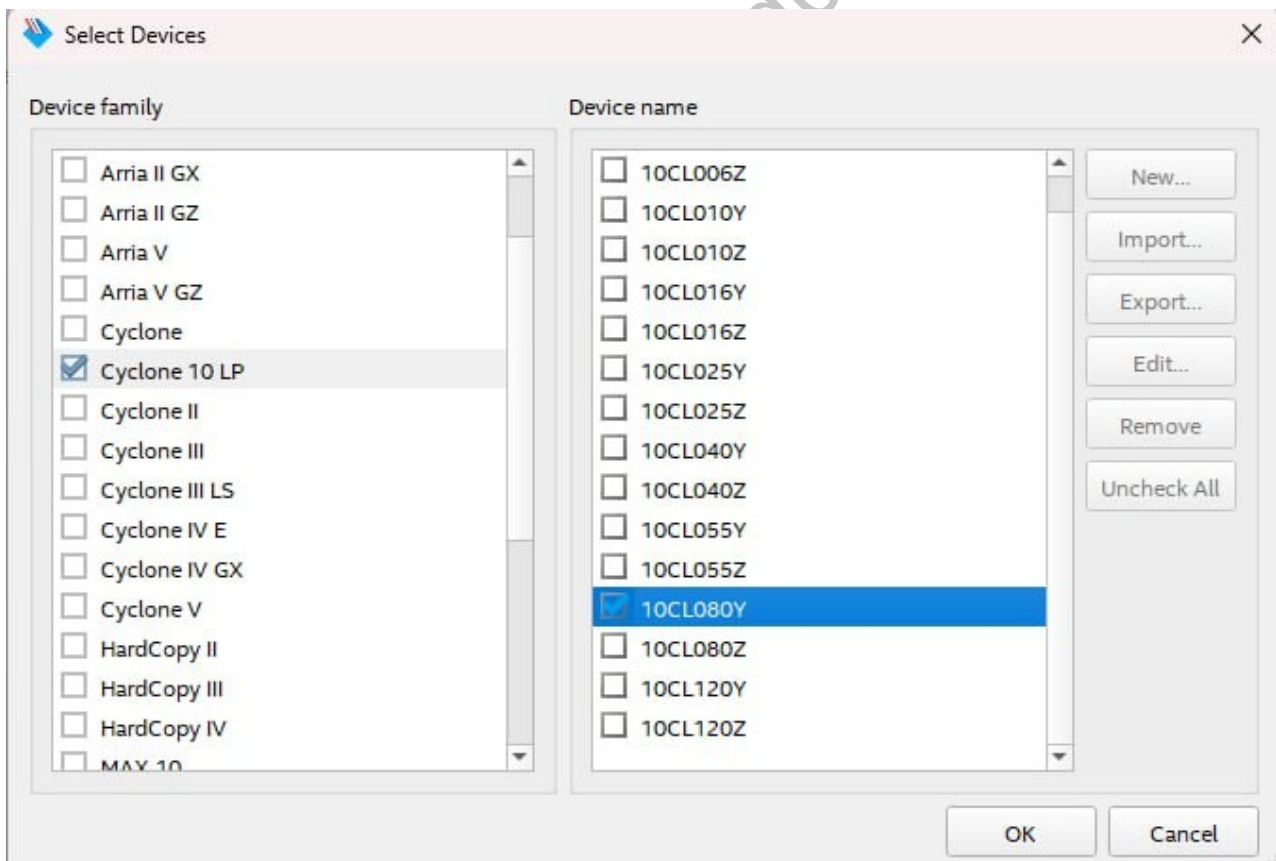
Ahora le damos a la opción *Advanced...* Y marcamos las dos casillas que aparecen arriba.



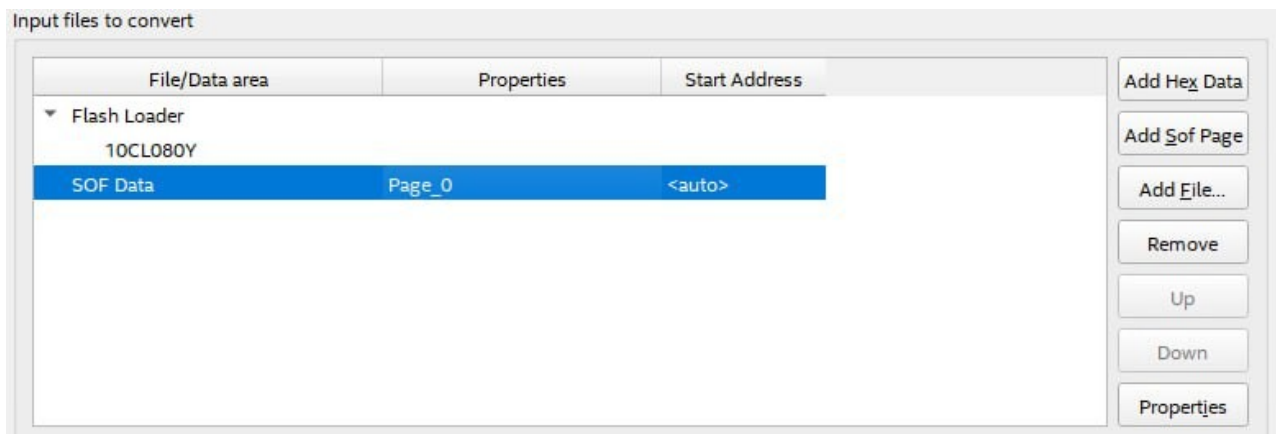
Ahora pasamos al recuadro inferior, en este recuadro tiene que estar el dispositivo a programar (*Flash Loader*) y el bitstream (*SOF Data*).



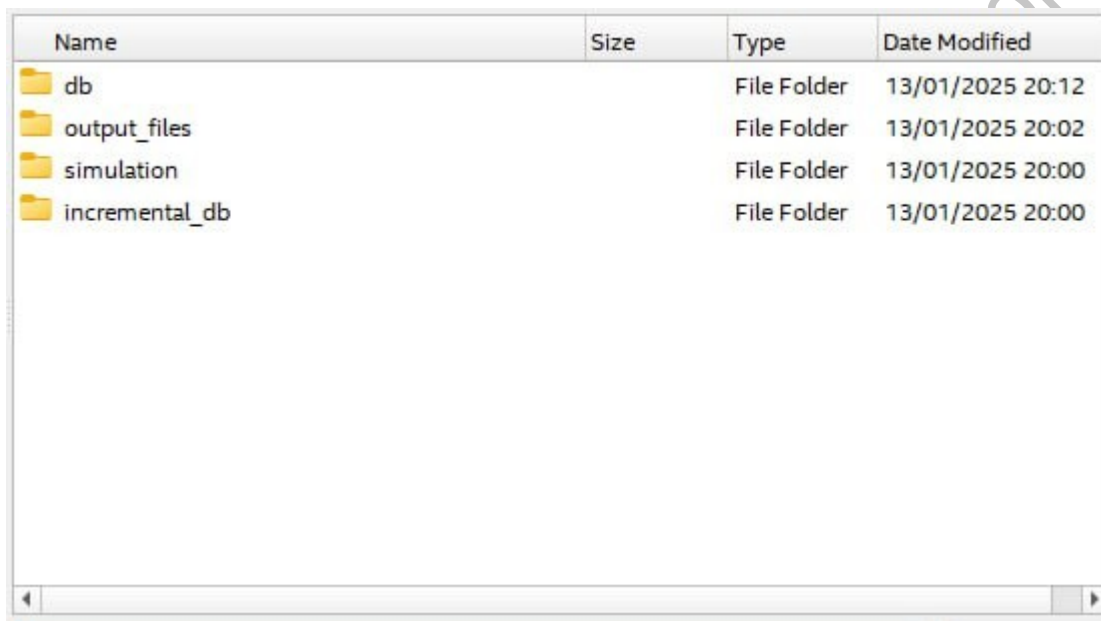
Ahora le damos a *Add Device...* y en la pestaña que se nos abre elegimos la FPGA que queremos programar.



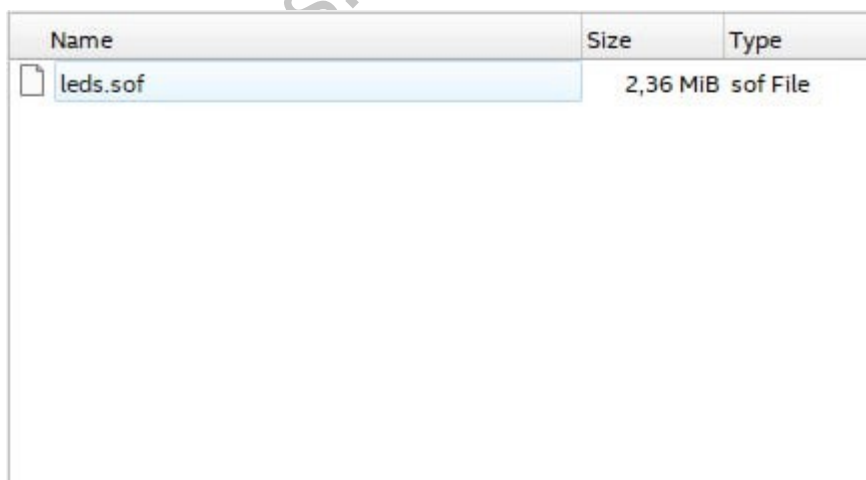
Una vez lo tenemos seleccionado pasamos a incluir el bitstream. Marcamos *SOF Data* y le damos a *Add File...* (si le das a *Add Sof Page*, lo que hace es añadir otro *SOF Data* en la interfaz, con uno solo vale).



Ahora nos pregunta donde está el bitstream que queremos grabar.



Ahora se selecciona el bitstream deseado.



Al terminar la creación del bitstream, quedaría una pestaña tal que así.

Specify the input files to convert and the type of programming file to generate.
You can also import input file information from other files and save the conversion setup information created here for future use.

Conversion setup files

Open Conversion Setup Data... Save Conversion Setup...

Output programming file

Programming file type: JTAG Indirect Configuration File (.jic)

Options/Boot info... Configuration device: EPCQ128 ... Mode: Active Serial

File name: output_files/output_file.jic

Advanced... Remote/Local update difference file: NONE

☒ Create Memory Map File (Generate output_file.map)

☐ Create CvP files (Generate output_file.periph.jic and output_file.core.rbf)

☐ Create config data RPD (Generate output_file_auto.rpd)

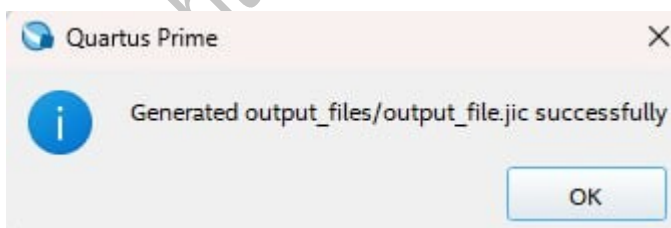
Input files to convert

File/Data area	Properties	Start Address
Flash Loader		
10CL080Y		
SOF Data	Page_0	<auto>
leds.sof	10CL080YU484	

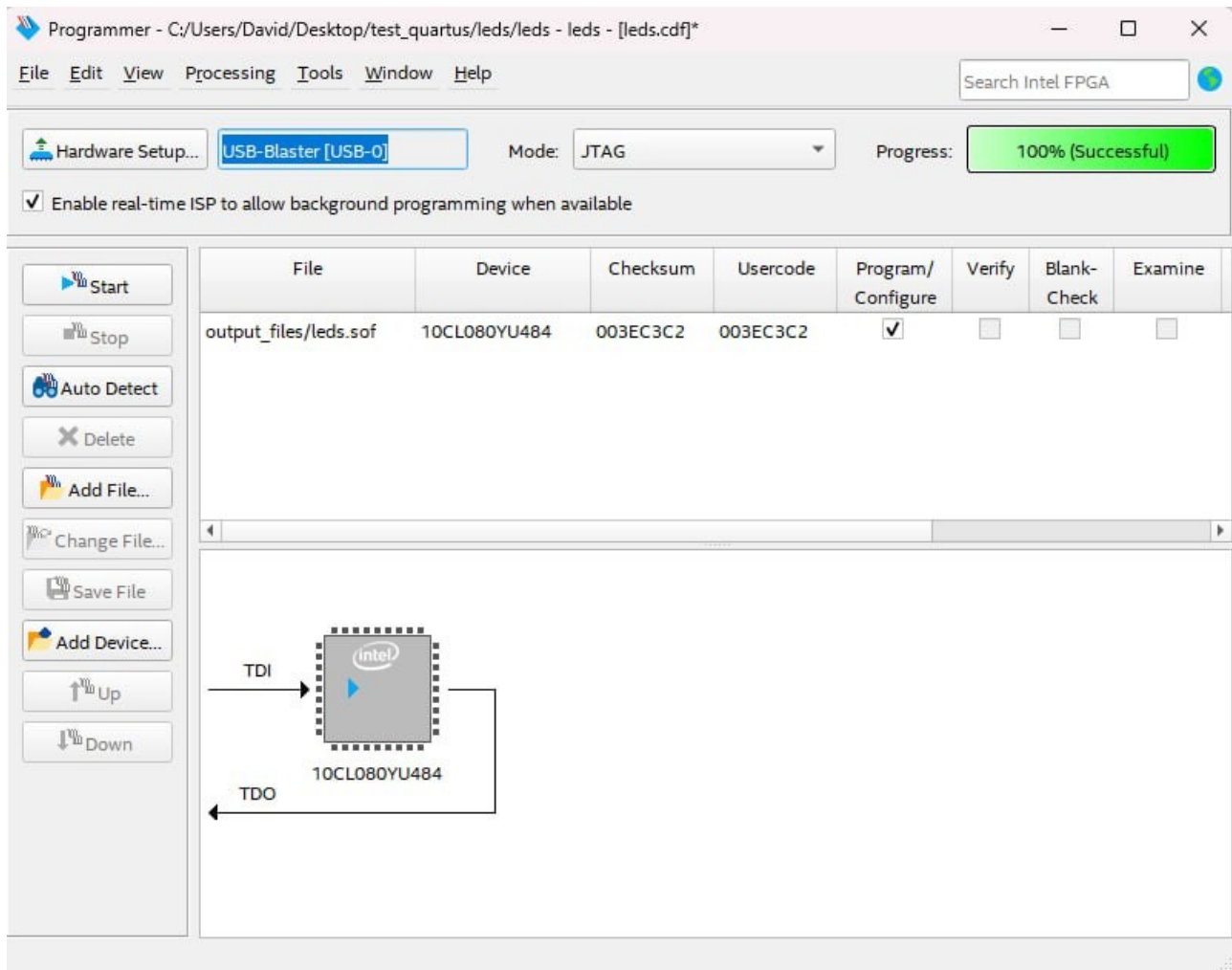
Add Hex Data
Add Sof Page
Add File...
Remove
Up
Down
Properties

Generate Close Help

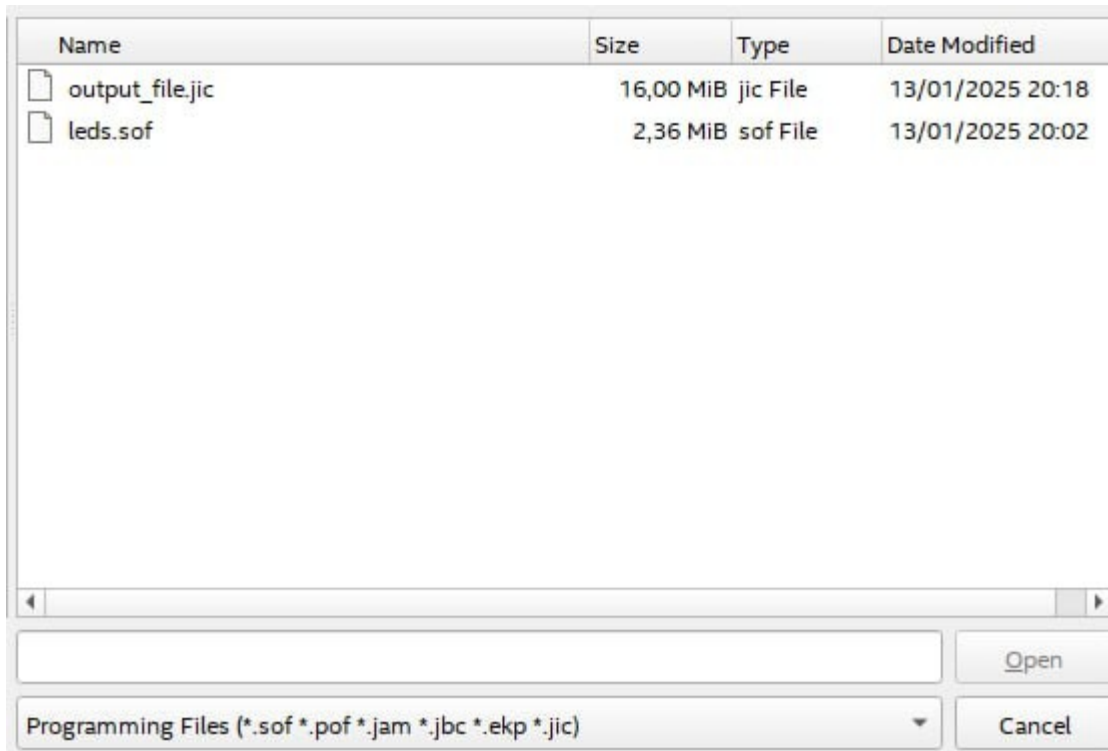
Ahora le damos a *Generate*. Si está todo correcto sale una pestaña como esta.



Ahora volvemos al *Programmer*.



Y le damos a *Add File*. Ahora no seleccionamos el SOF, si no que seleccionamos el JIC.

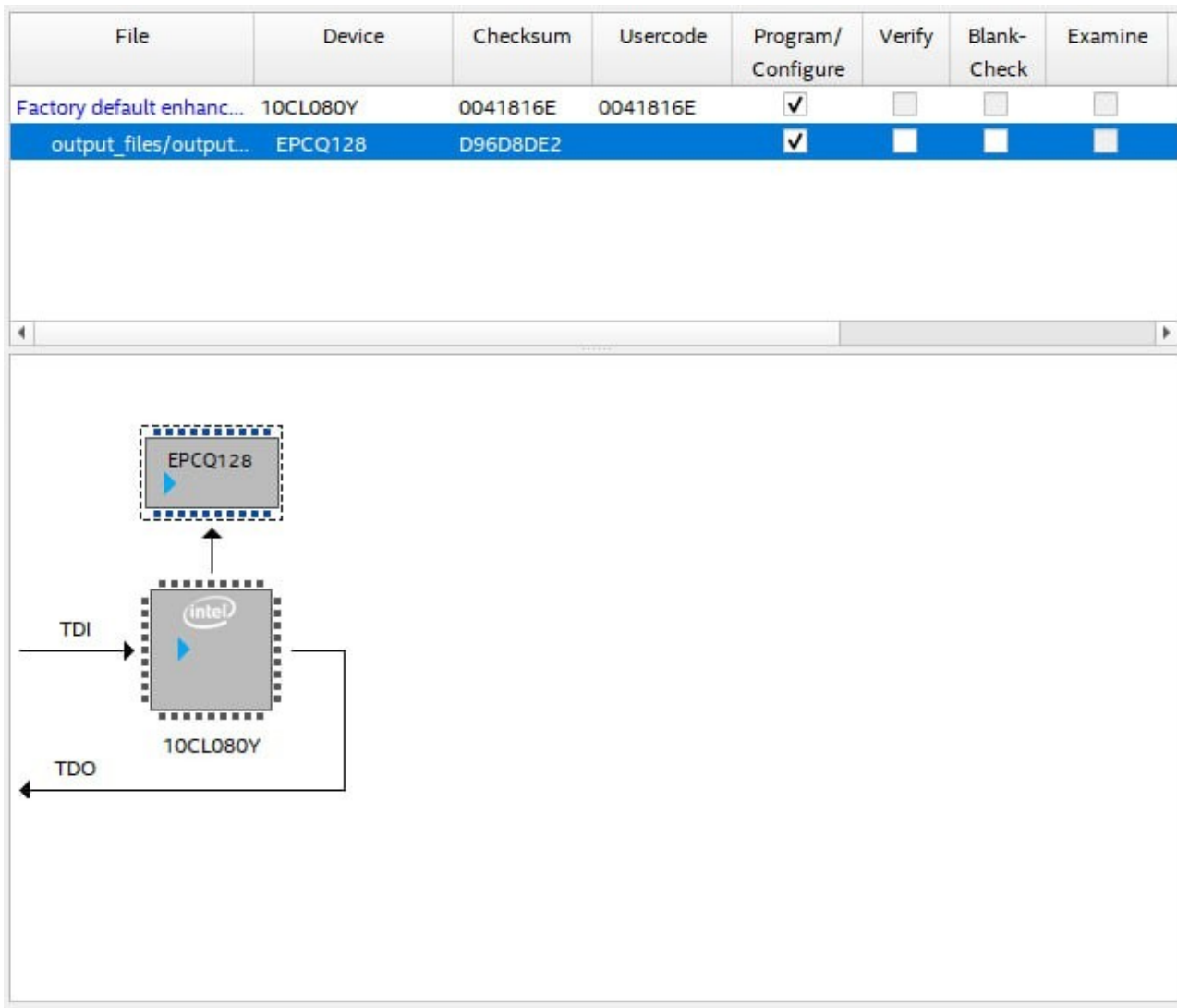


Esto nos mete la memoria flash interna junto con el bitstream anterior.

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine
output_files/leds.sof	10CL080YU484	003EC3C2	003EC3C2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<none>	10CL080Y	00000000	0041816E	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
output_files/output...	EPCQ128	D96D8DE2		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The diagram illustrates the JTAG chain configuration. It shows three devices connected in a chain: a 10CL080YU484 (Intel), a 10CL080Y (Intel), and an EPCQ128 (Intel). The 10CL080YU484 is connected to the 10CL080Y via a TDI/TDO signal. The 10CL080Y is connected to the EPCQ128 via a TDI/TDO signal. The 10CL080YU484 is also connected to the 10CL080Y via a TDI/TDO signal.

Borramos el bitstream anterior y marcamos la casilla *Program/Configure*.



Ahora lo único que hay que hacer es darle al botón de *Start* del *Programmer*, y desde entonces, la FPGA quedará programada.

Es muy posible que la primera vez que se programa la FPGA no empiece a funcionar después de que se grabe, eso es porque hay que resetear la FPGA, puedes o apagar y encender la FPGA o resetear utilizando el pin *nCONFIG* de la FPGA de Intel.