

# Depuración de una FPGA de Intel en Quartus

Creador: David Rubio G.

Entrada: <https://soceame.wordpress.com/2025/01/14/depuracion-de-una-fpga-de-intel-en-quartus/>

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

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

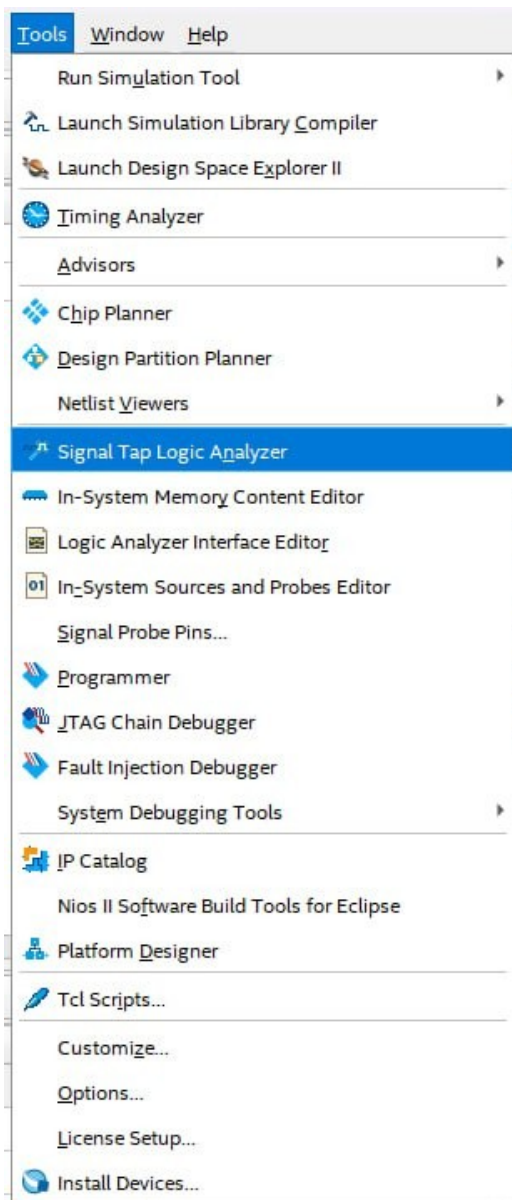
Fecha última modificación: 24/02/2025

Para hacer la depuración de FPGAs, Quartus dispone de varias herramientas, en esta entrada voy a hablar de dos de ellas, que son las que tienen otros fabricantes.

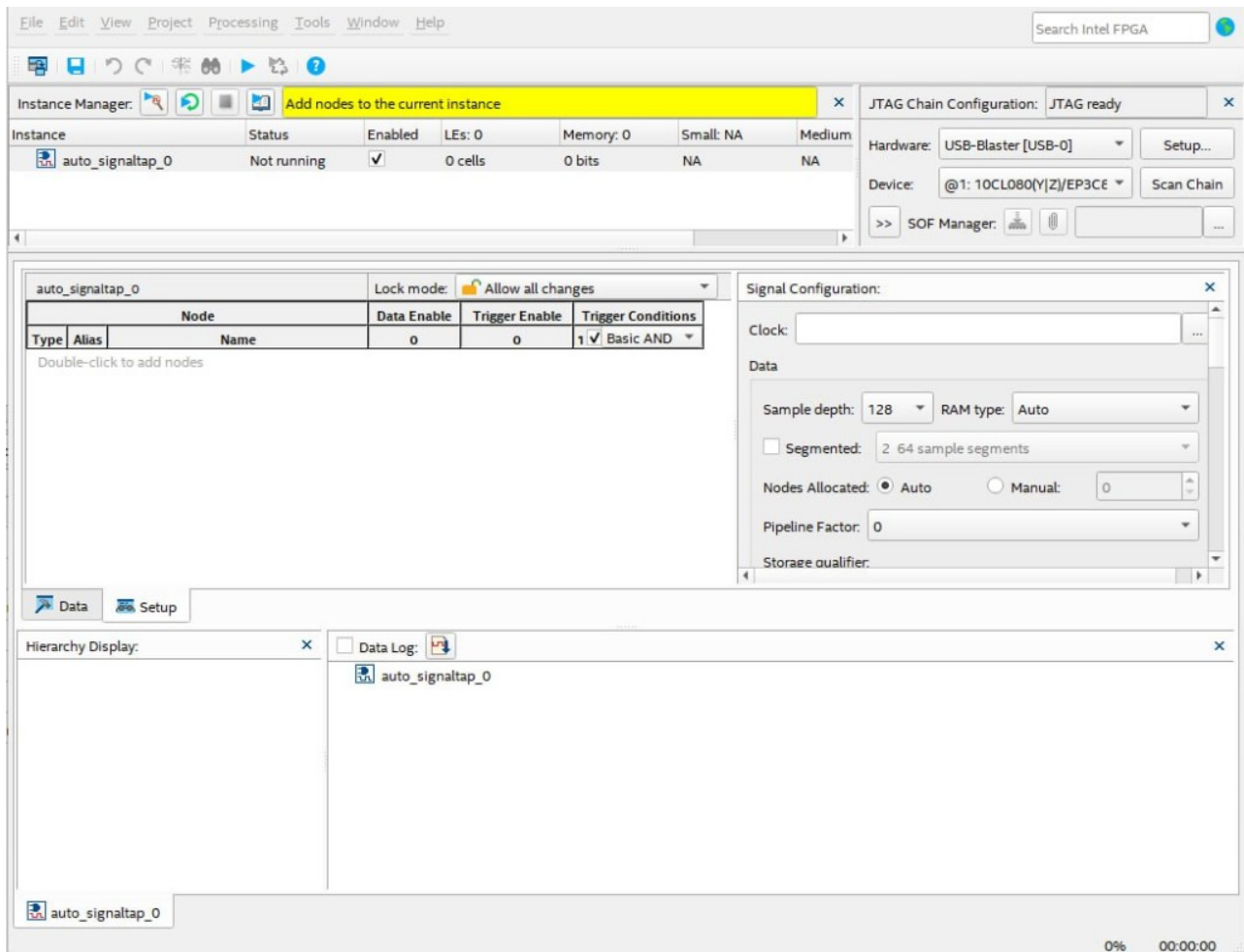
## Depuración de señales mediante interfaz

Quartus dispone de un herramienta para poder hacer análisis de señales internas de síntesis.

Para acceder a esta herramienta solo hay que darle a la opción de *Tools* y a *Signal Tap Logic Analyzer*.

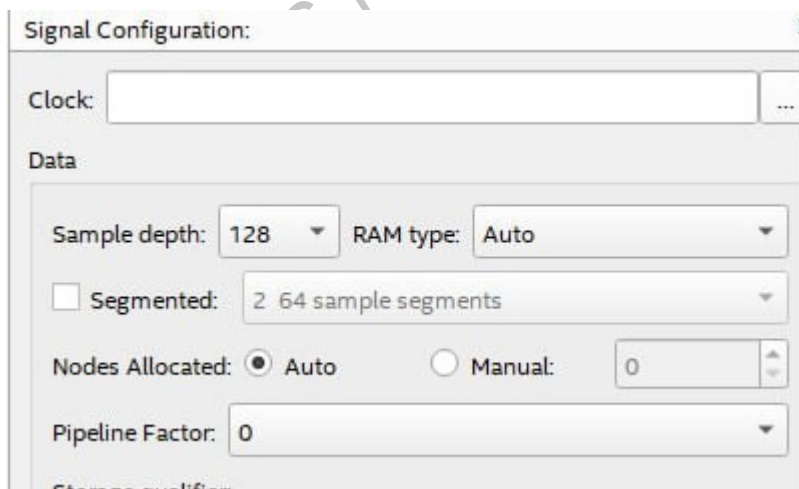


Esto nos abre una herramienta como la siguiente.

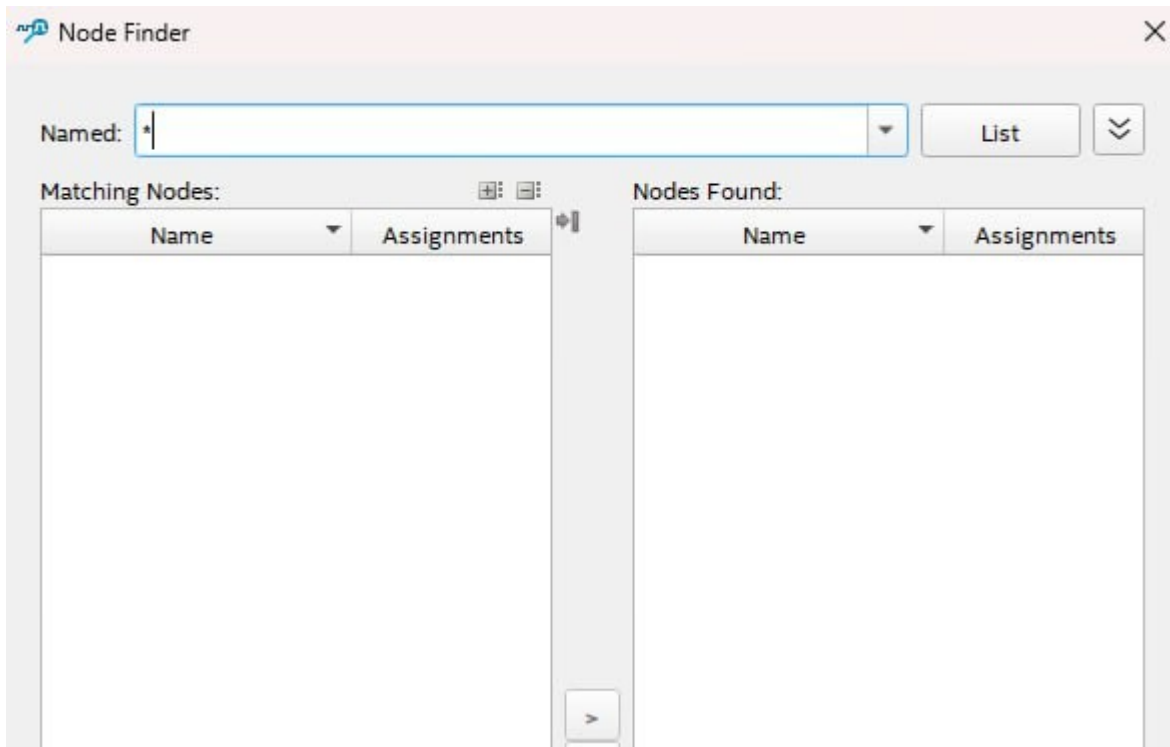


Esta herramienta cuenta con dos partes, una de configuración, Setup, y otra de depuración, Data. Para que Data funcione primero que configurar las señales que se quieren depurar.

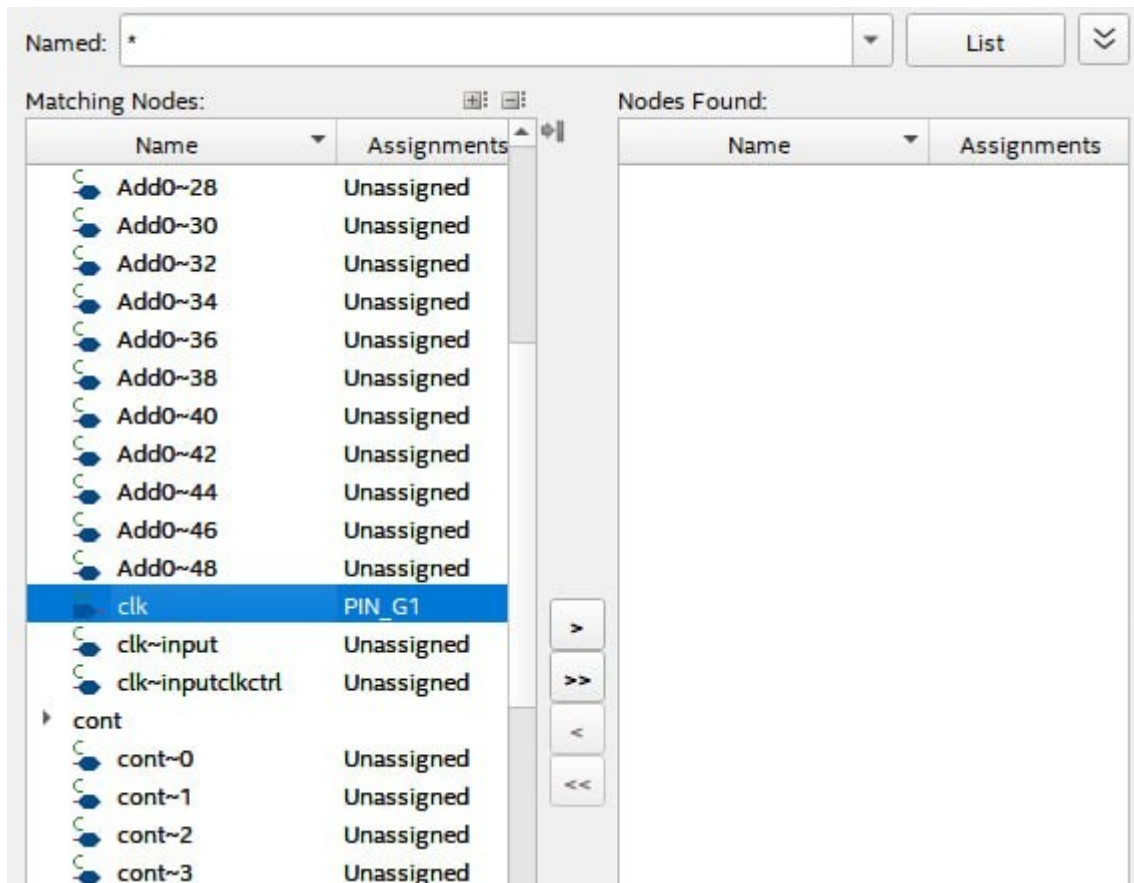
Lo primero que se tiene que configurar es la señal de reloj que se va a utilizar para depurar, y el número de muestras que se va a utilizar.



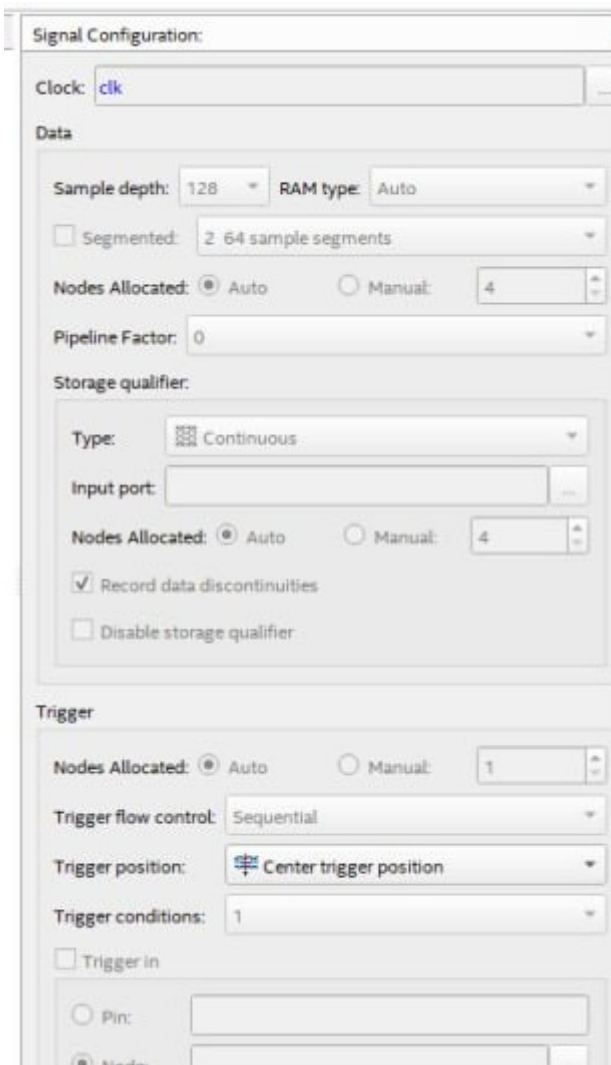
Para seleccionar la señal de reloj, primero hay que haber depurado el proyecto, después se le da a los tres puntos, y nos abre una pestaña como la siguiente. Aquí le damos a List.



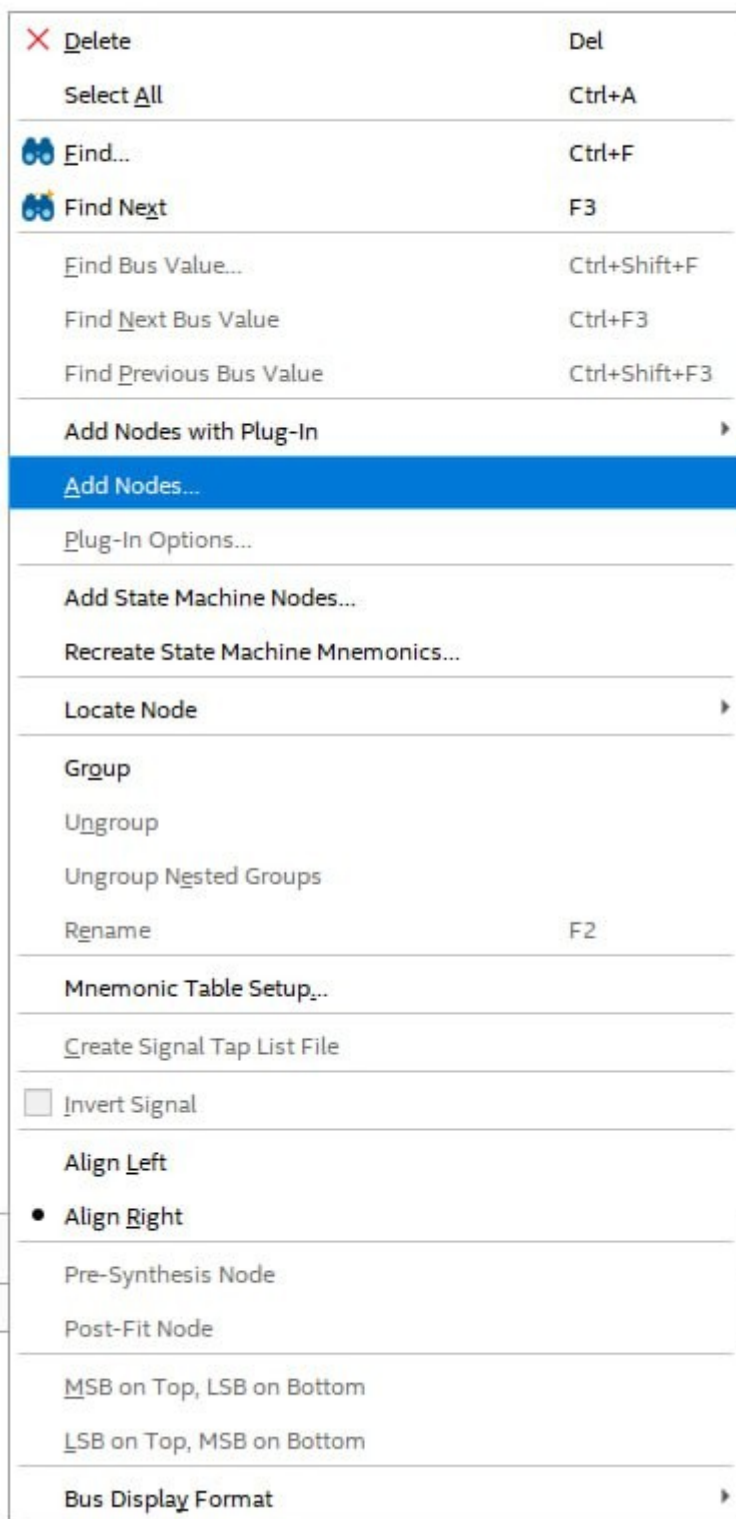
Esto nos abre un listado con todas las señales de síntesis que ha creado Quartus.



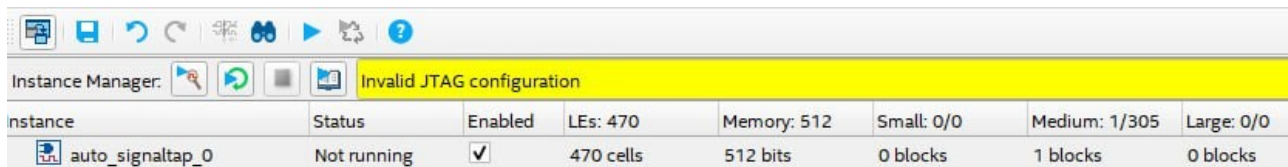
Entonces aparecerá el reloj en la pestaña derecha.



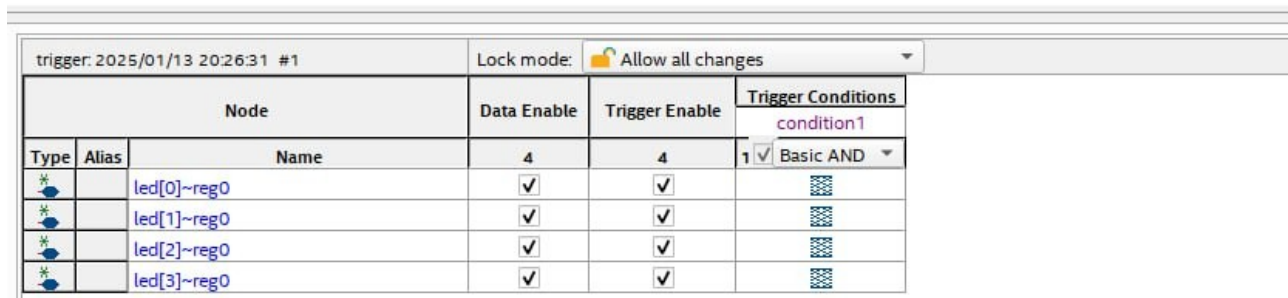
Después, lo que hay que hacer es añadir la señales a depurar. Para ello se le da clic derecho en la interfaz, y se busca la opción de *Add Nodes*.



El funcionamiento es igual que la señal de reloj, elegir las señales a depurar, llamadas Nodes (nodos).



Instance	Status	Enabled	LEs: 470	Memory: 512	Small: 0/0	Medium: 1/305	Large: 0/0
auto_signaltap_0	Not running	<input checked="" type="checkbox"/>	470 cells	512 bits	0 blocks	1 blocks	0 blocks

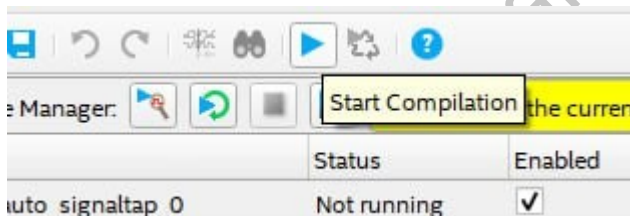


Node		Data Enable	Trigger Enable	Trigger Conditions
Type	Alias			
	Name	4	4	1 <input checked="" type="checkbox"/> Basic AND
	led[0]~reg0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	led[1]~reg0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	led[2]~reg0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	led[3]~reg0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Ahora al seleccionar las señales de depuración nos aparecerá este mensaje arriba. Este significa que hay que re-sintetizar el diseño para que la FPGA pueda sacar las señales de depuración.

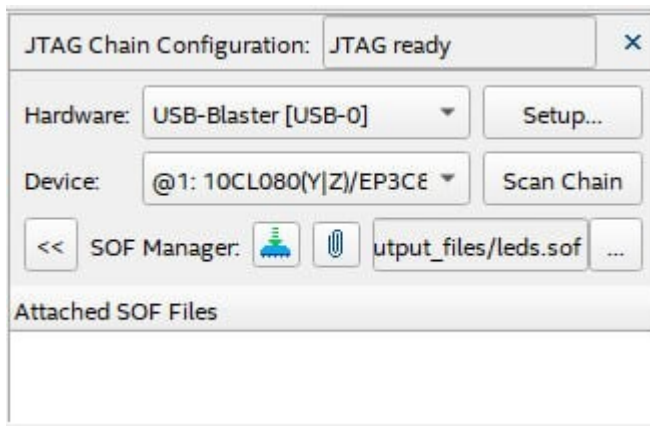


Para ello se le da *Start Compilation*, que te lleva a Quartus y allí se vuelve a ejecutar la síntesis, cuando termina volvemos al programa depurador.



Ahora lo que tenemos que hacer es grabar el bitstream dándole al icono de guardar, lo que hace que la barra amarilla desaparezca, y ahora podemos ir a Data.





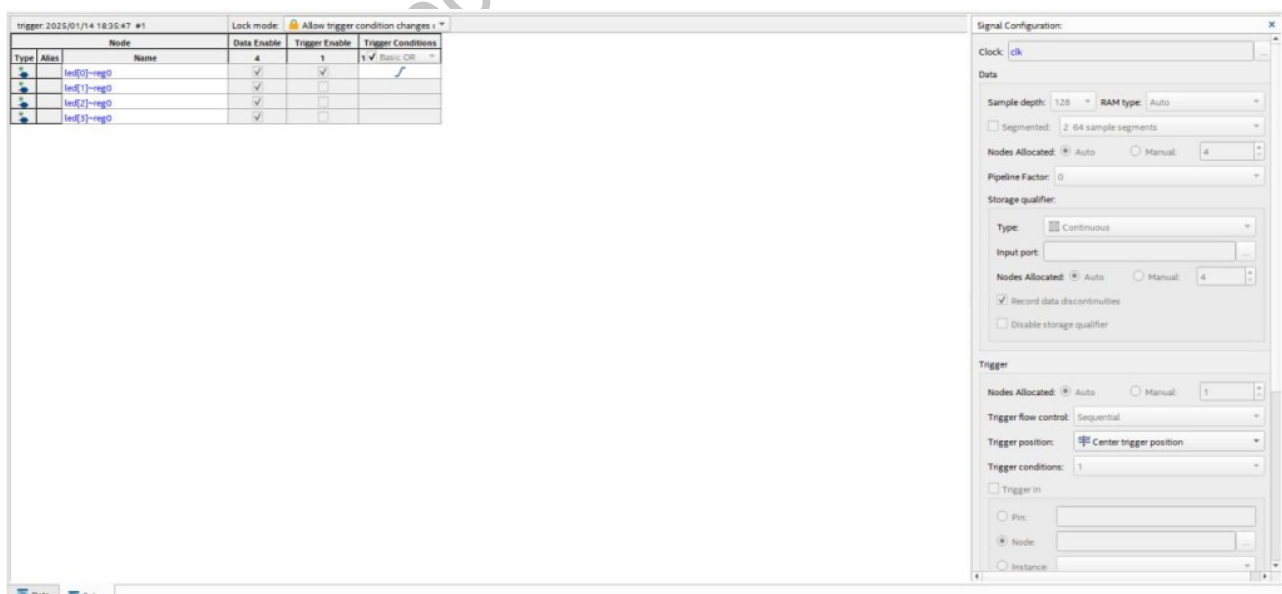
En Data aparecen las señales a depurar. Ahora solo tenemos que darle a estas opciones. Una es para ejecutar la depuración normal, otra es para depurar de forma continua, de tal forma que se pueden ver los datos según cambian con el tiempo. Después viene un *Stop* que es el que para la depuración lazoada, y por último el lector de valores.



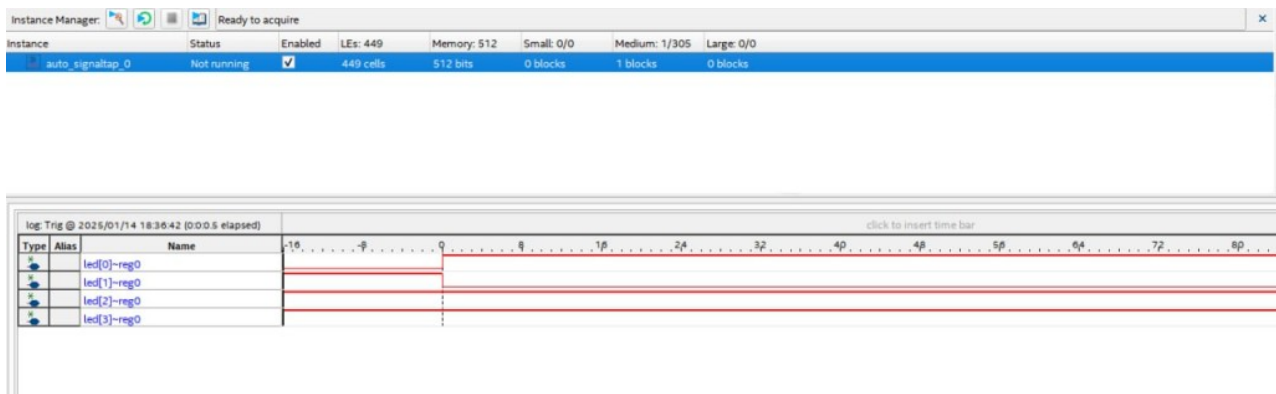
Ahora para poder conseguir que la depuración funcione como queremos, tenemos que cambiar las opciones en *Setup*, como que sea flanco de subida/bajada, '0'/'1', etc.

**NOTA:** a la derecha la única opción que nos permite hacer la depuración sin tener que regenerar el bitstream es la de *Trigger Position*.

**NOTA 2:** para que la depuración funcione, hay que cambiar el *Trigger Condition*.



Ahora solo tenemos que darla a ejecutar la depuración y cuando se pare es que se ha cumplido la condición de muestreo de la depuración.



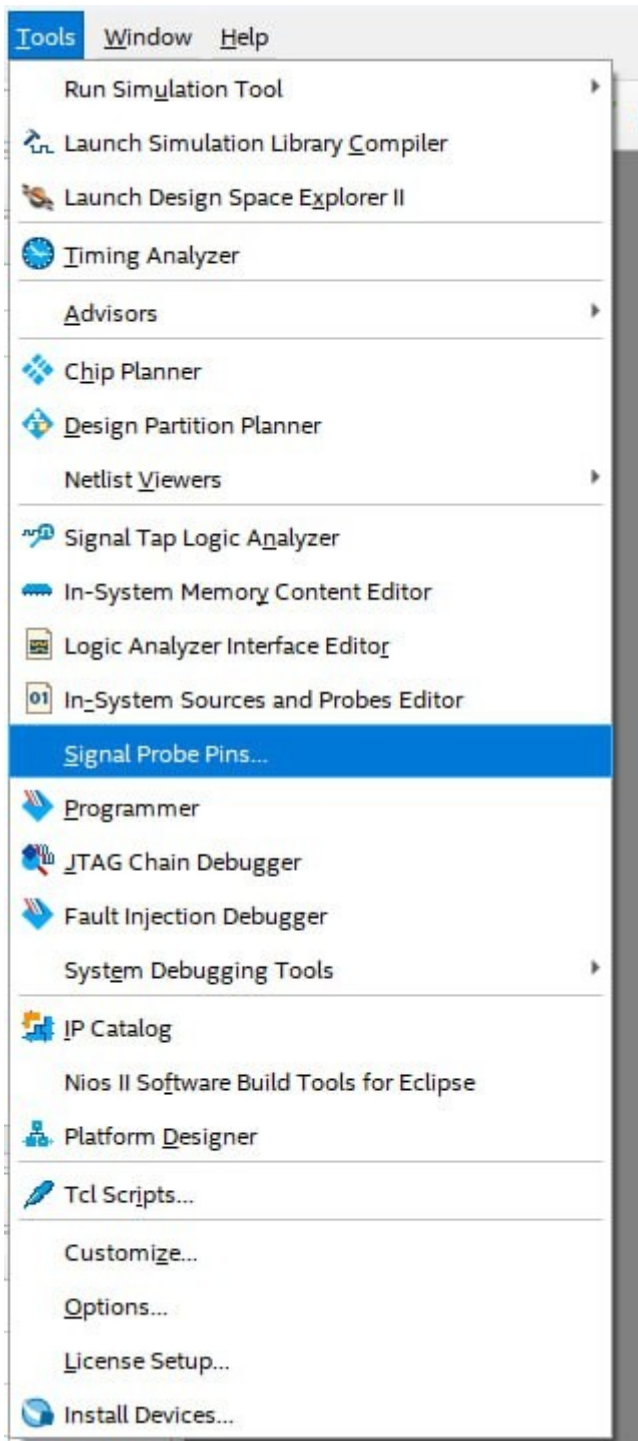
Por último comentar que se pueden tener diferentes perfiles de depuración dentro de este analizador, lo que permite poder comprobar diferentes parámetros sin tener que regenerar continuamente los bitstreams. Lo único es que tienen que tener su propio bitstream independiente, y para cambiar de depuración hay que grabar el bitstream que se ha creado para cada perfil.



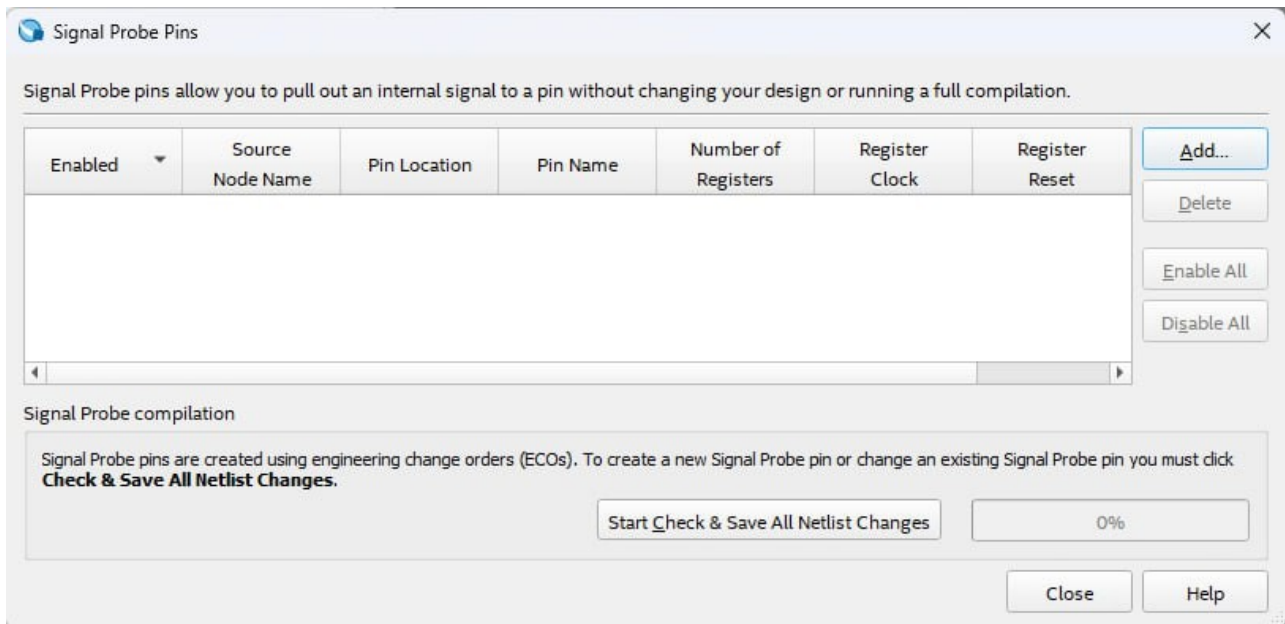
## Depuración de señales mediante Probes

La depuración mediante Probes lo que hace es sacar por un pin de la FPGA una señal interna de la síntesis, para poder verla en un osciloscopio u otra herramienta.

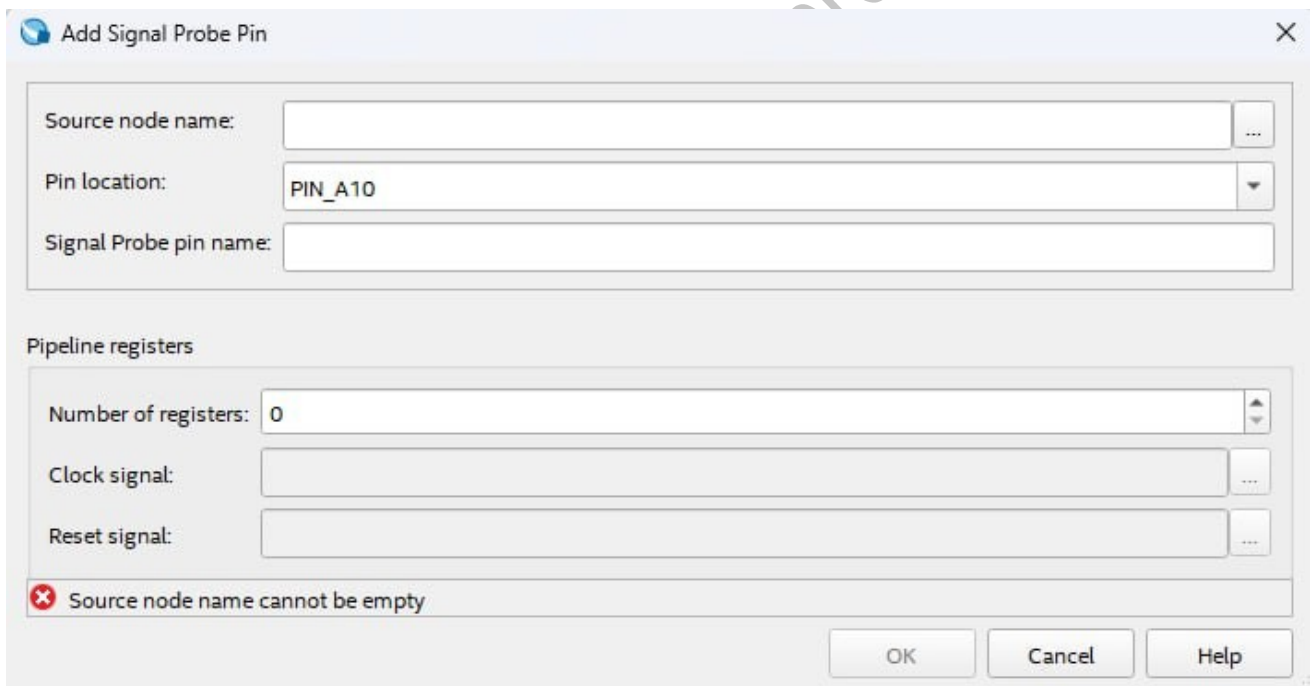
Para ello hay que ir a *Tools* y después a *Signal Probe Pins*.



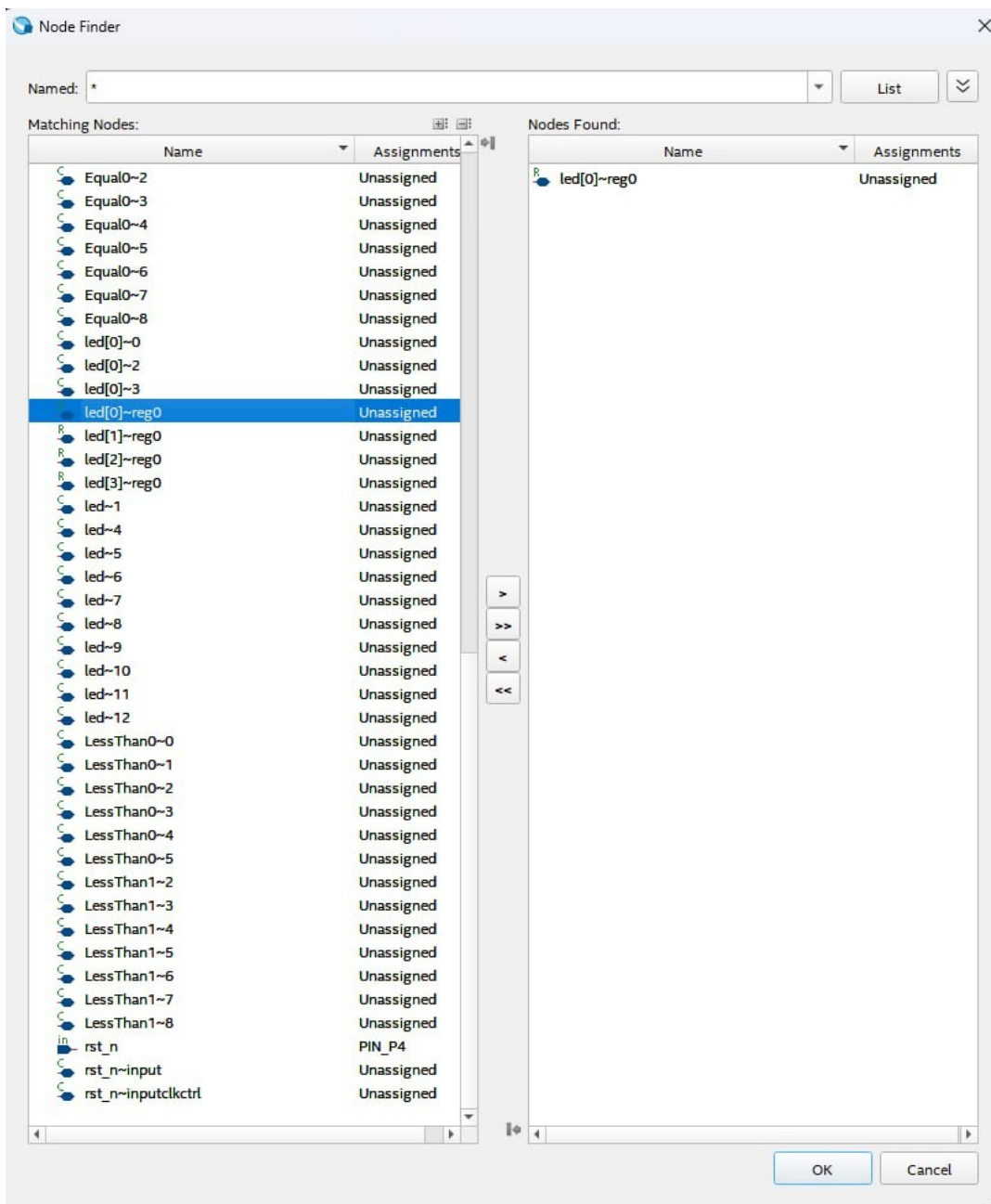
Entonces, se nos abre una interfaz que es donde configuraremos los pines para sacarlos al exterior. Ahora le damos a *Add...*



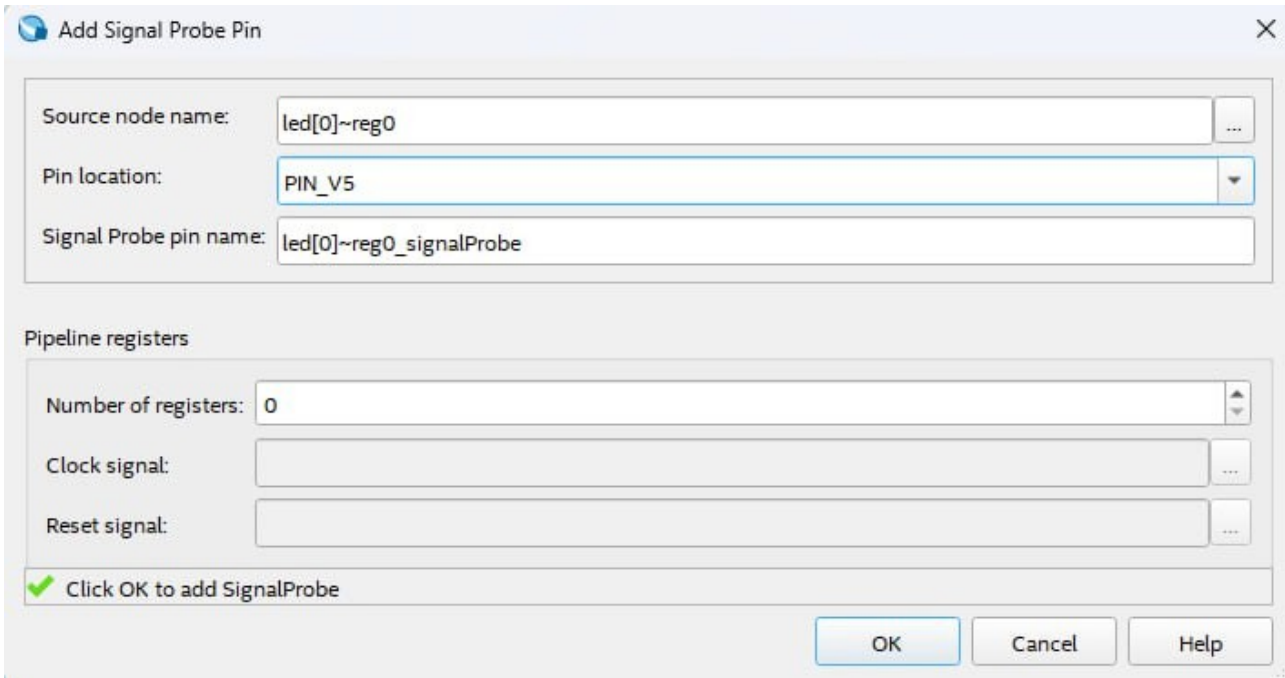
Y se nos abre una pestaña que es donde seleccionaremos el pin y la señal.



Le damos a los tres puntos del nombre y buscamos la señal que queremos depurar.



Cuando la tengamos, podemos elegir el pin por el que queremos que salga la señal.



Source node name: led[0]~reg0

Pin location: PIN\_V5

Signal Probe pin name: led[0]~reg0\_signalProbe

Pipeline registers

Number of registers: 0

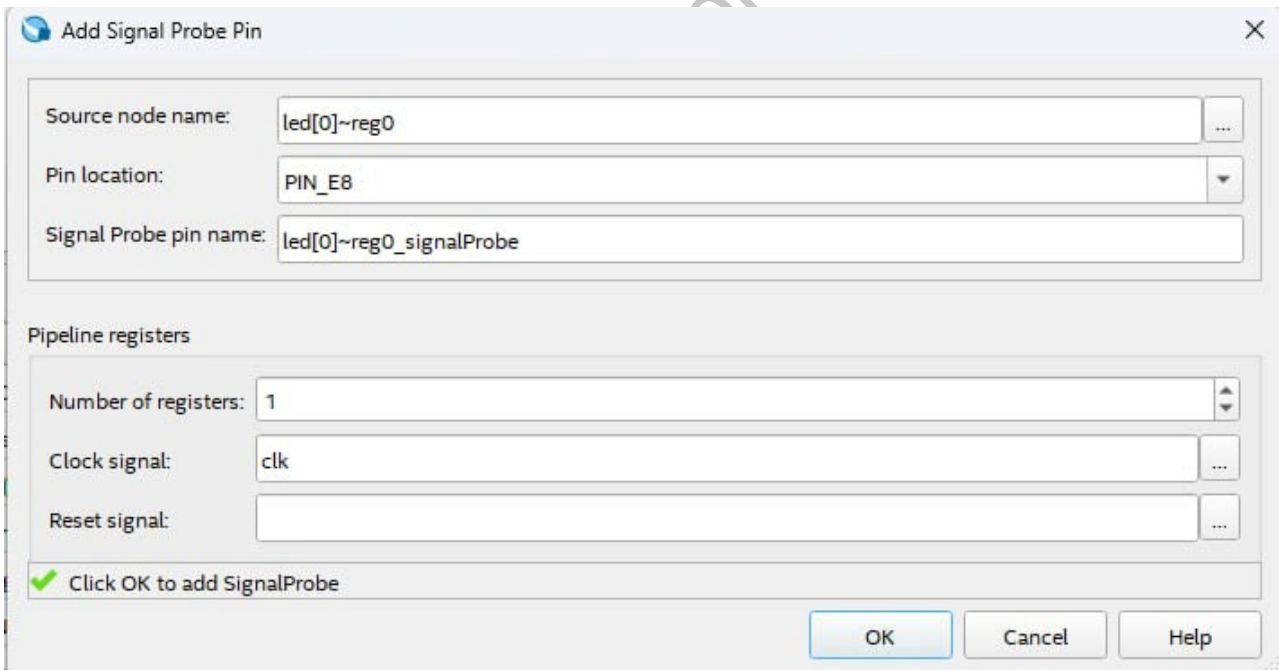
Clock signal:

Reset signal:

✓ Click OK to add SignalProbe

OK Cancel Help

Es posible que esta configuración no llegue a funcionar, por ello se añade 1 registro y se le da la señal de reloj que se quiere para el registro.



Source node name: led[0]~reg0

Pin location: PIN\_E8

Signal Probe pin name: led[0]~reg0\_signalProbe

Pipeline registers

Number of registers: 1

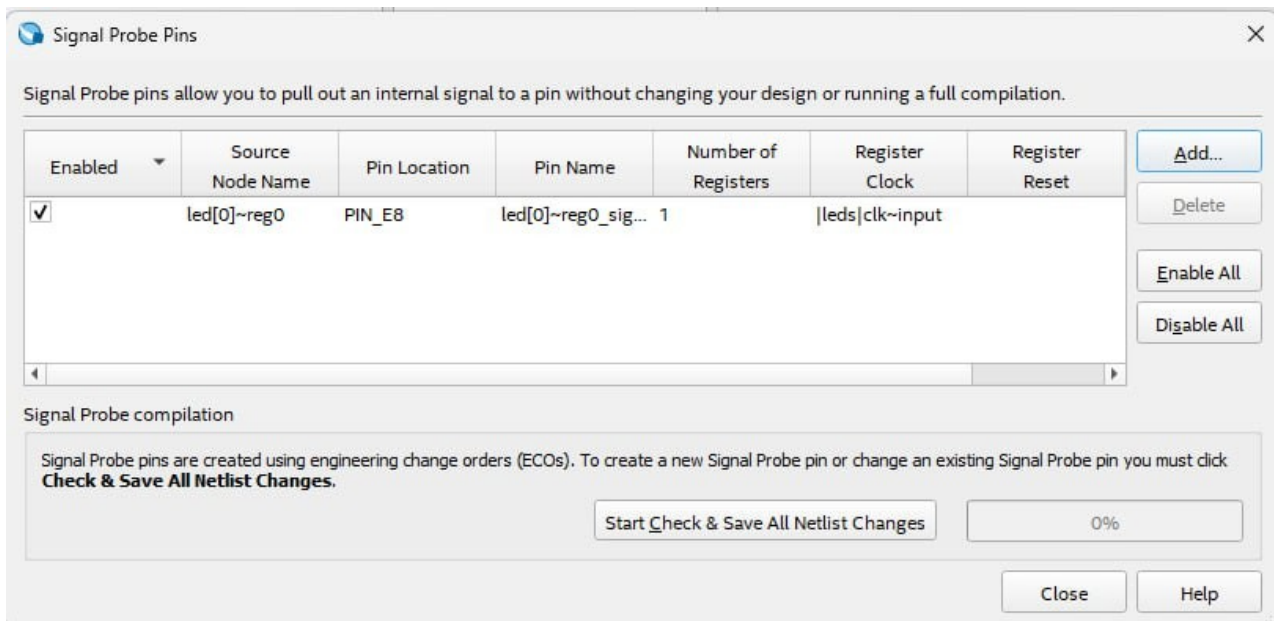
Clock signal: clk

Reset signal:

✓ Click OK to add SignalProbe

OK Cancel Help

Al terminar la configuración nos aparece esta pestaña rellena con todos los pines externos de depuración (a los que se le puede cambiar el nombre para ser más amigables).

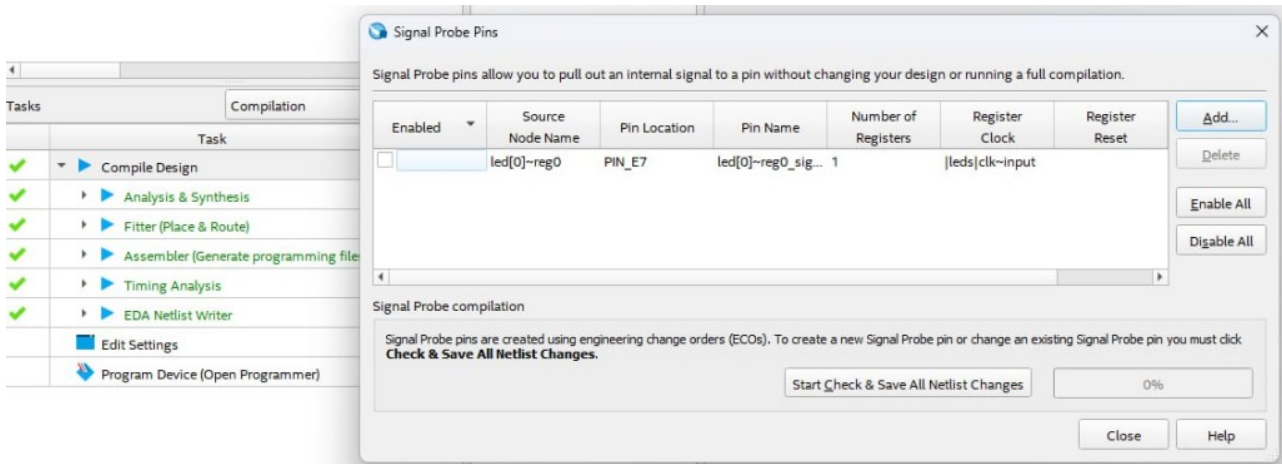


Ahora generamos el bitstream o lo que le falte, pero **NUNCA** dándole a *Compile Design*, solo dándole uno a uno a lo que falte para generar el bitstream.

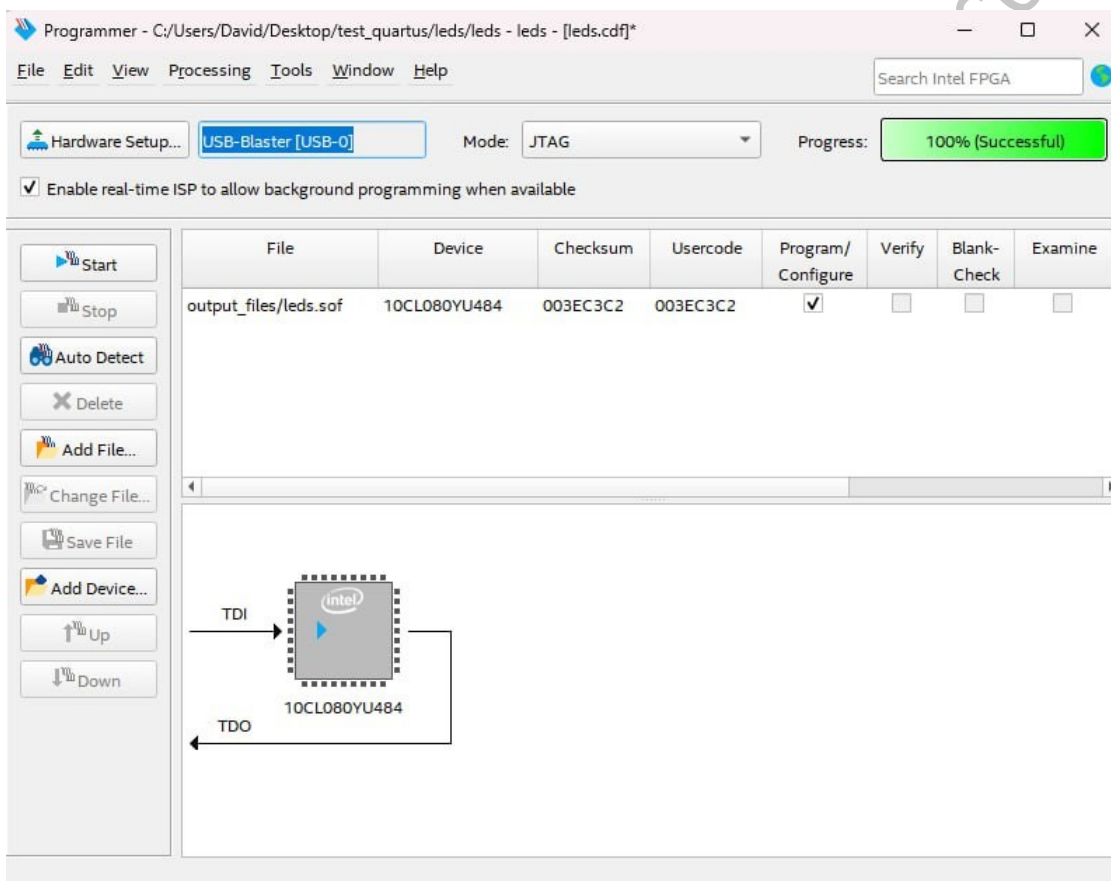
	Task	Time
	▶ Compile Design	
✓	▶ ▶ Analysis & Synthesis	00:00:22
✓	▶ ▶ Fitter (Place & Route)	00:00:47
✓	▶ ▶ Assembler (Generate programming files)	00:00:05
✓	▶ ▶ Timing Analysis	00:00:02
✓	▶ ▶ EDA Netlist Writer	00:00:03
	■ Edit Settings	
	🔧 Program Device (Open Programmer)	

Porque si le das a *Compile Design* desactiva los pines externos de depuración.





Ahora lo único que tienes que hacer es grabar el bitstream.



Si analizamos la señal de salida por el pin, podemos ver que en el pin que hemos seleccionado saca la señal de encendido/apagado del led.



