

Formas de depuración en Libero

Creador: David Rubio G.

Entrada: <https://soceame.wordpress.com/2024/12/07/formas-de-depuracion-en-libero/>

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

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

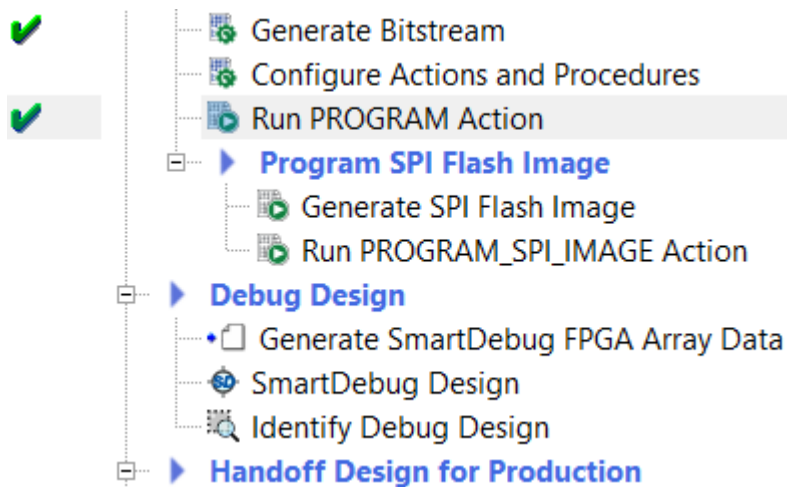
Fecha última modificación: 23/02/2025

En esta entrada voy a comentar las diferentes formas de depurar una FPGA/SoC de Microchip utilizando Libero. Algunas de estas formas son bastante interesantes.

La diferencia entre depurar una FPGA de Microchip y una de Xilinx, es que Xilinx te exige que le digas que señal quieres depurar, mientras que Microchip te permite depurar cualquier señal interna de la FPGA, sin tener que definir cuál quieres.

Acceder a la depuración

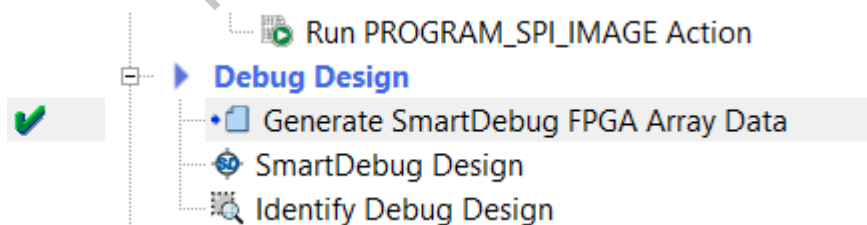
Para acceder a la depuración con Libero tenemos que utilizar las opciones de *Debug Design*.



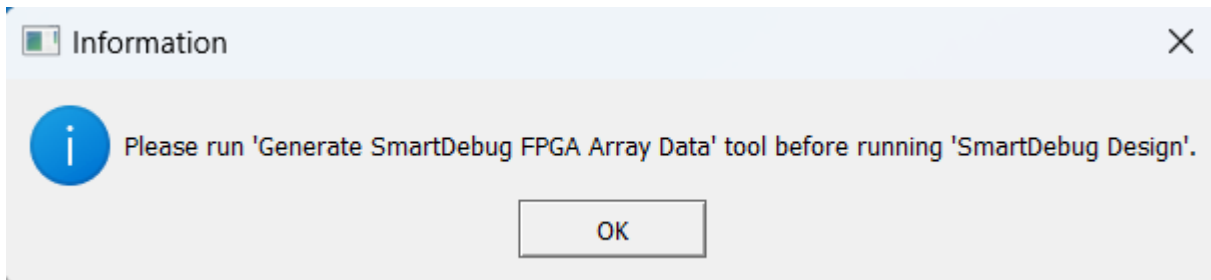
Aquí aparecen tres opciones:

- *Generate SmartDebug FPGA Array Data*: esta opción es la inicialización del sistema de depuración de Libero.
- *SmartDebug Design*: esta opción es la que nos lleva a las herramientas de depuración.
- *Identify Debug Design*: esta opción abre el programa *Identify* de Synopsys para depuración. Este se intentará explicar en otra entrada de forma exclusiva.

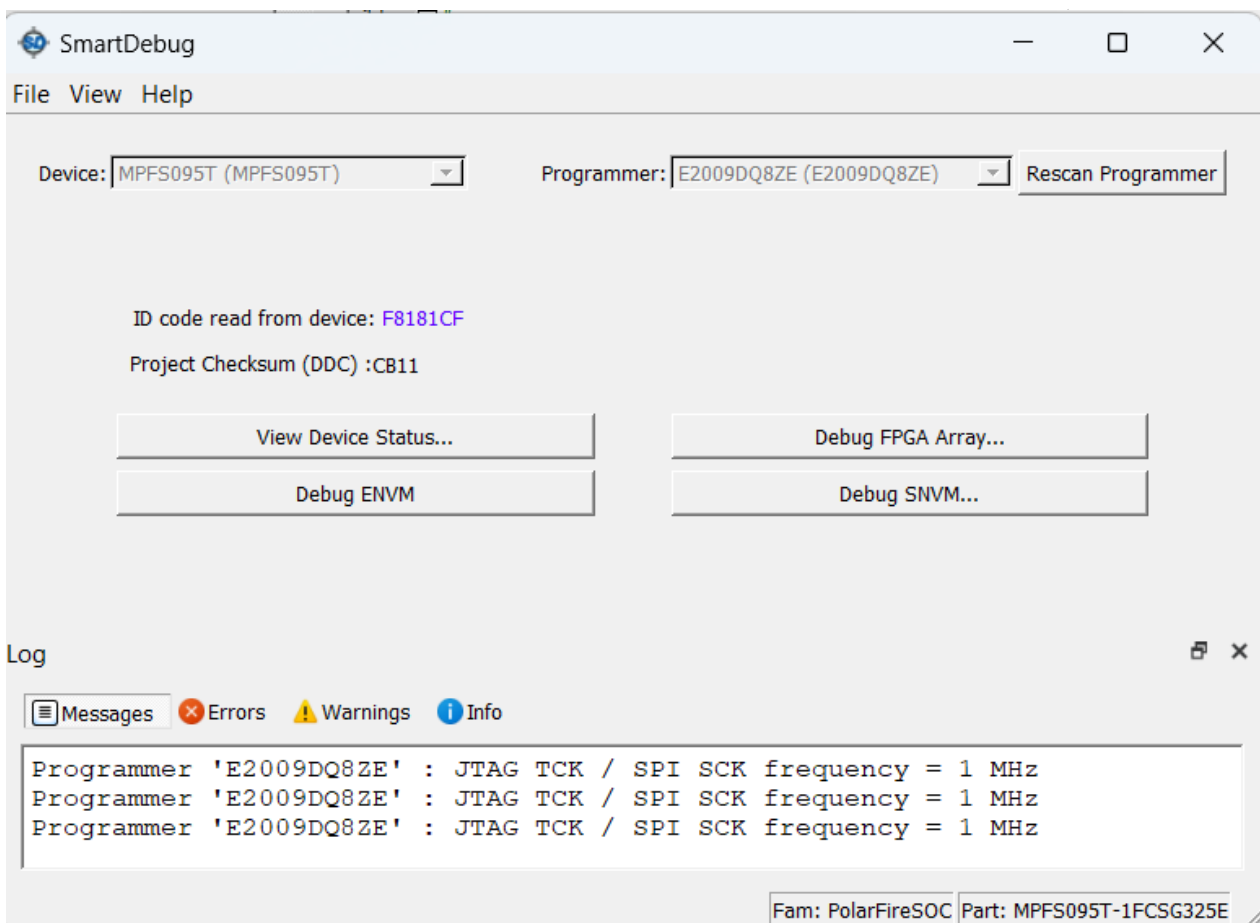
Para empezar primero tenemos un bitstream grabado en la FPGA/SoC y después hay que darle a *Generate SmartDebug FPGA Array*, lo cuál nos genera el perfil de depuración.



NOTA: si no le damos primero nos salta esta pestaña



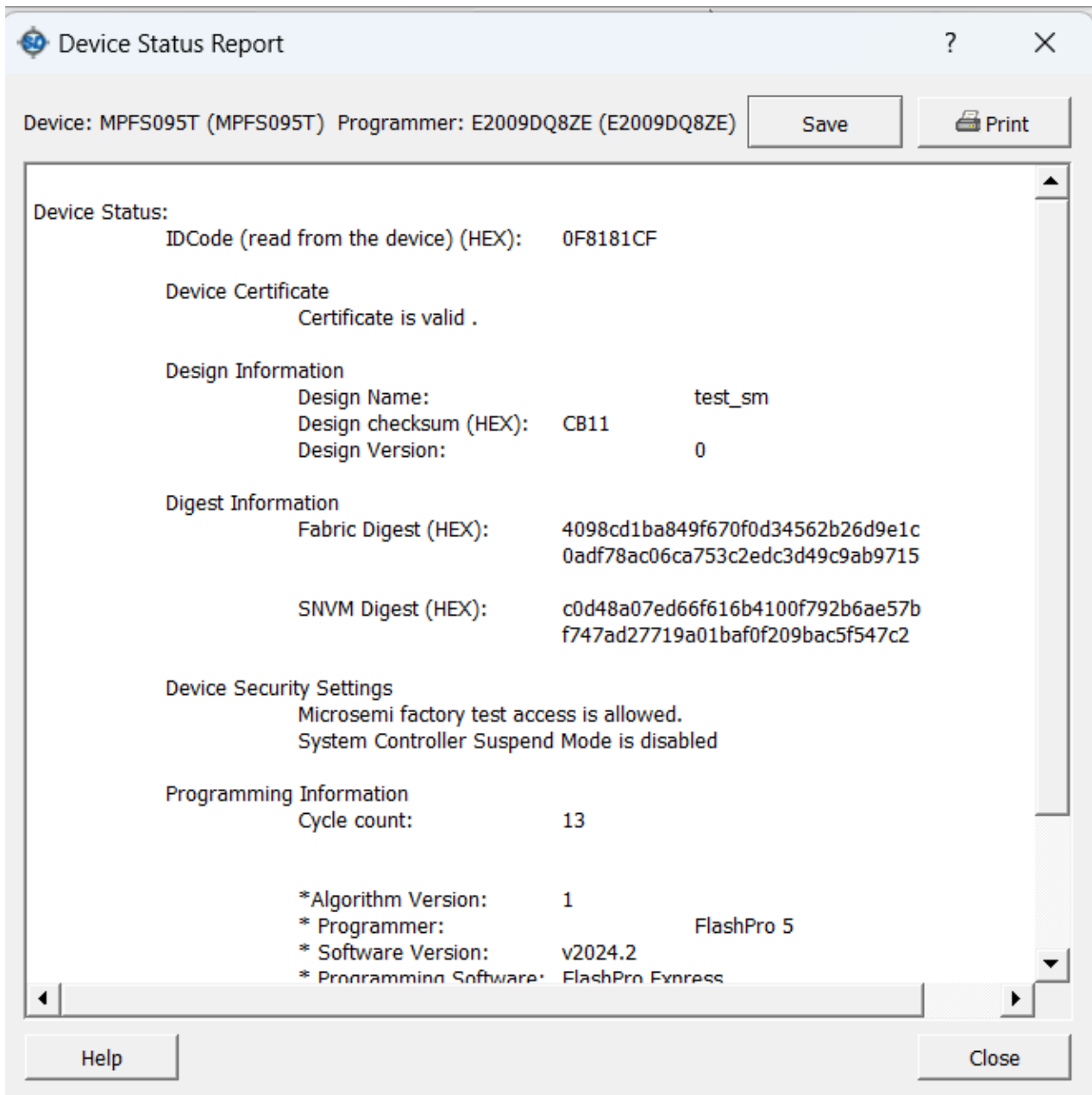
Después le podemos dar a la opción *SmartDebug Design*, lo cuál nos abre una pestaña para realizar la depuración.



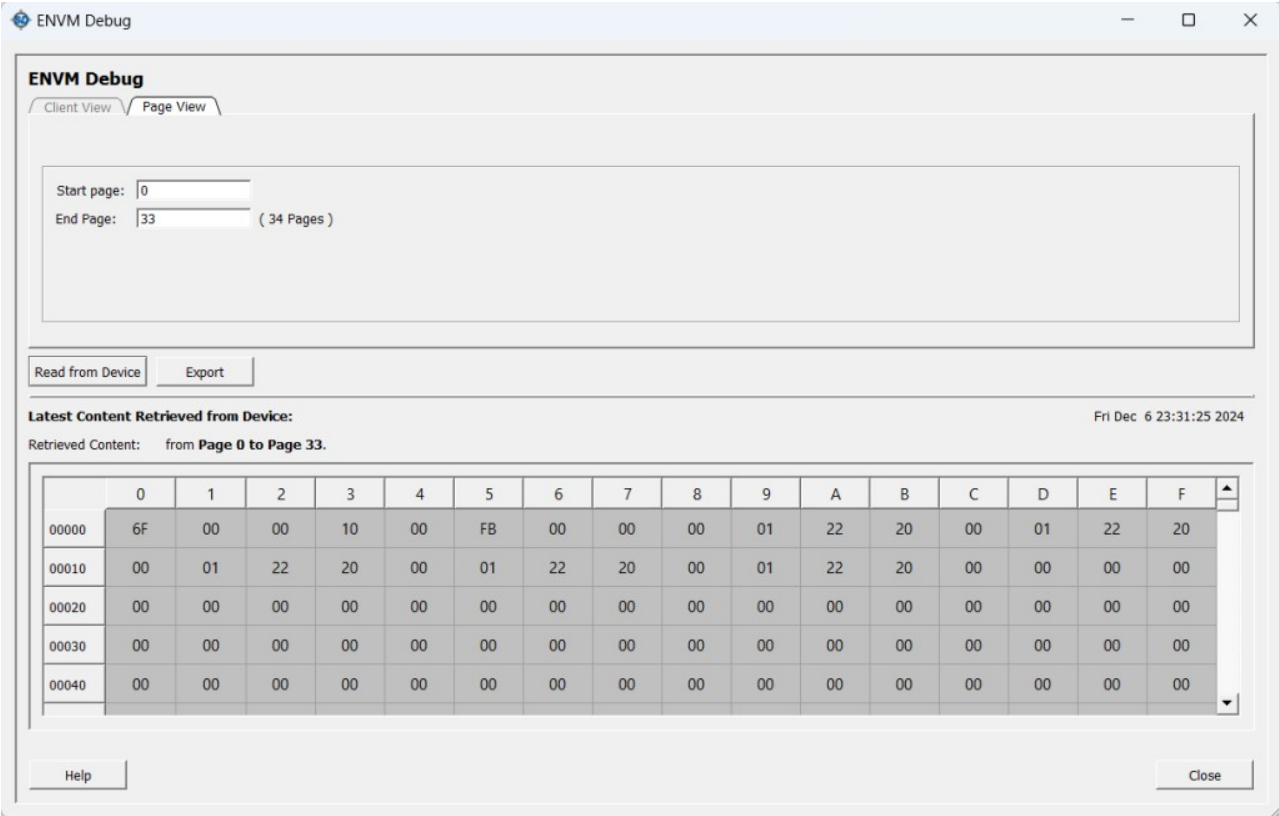
Herramientas de depuración

La herramienta que se nos abre en el paso anterior tiene diferentes herramientas dentro.

- **View Device Status:** esta herramienta nos da información sobre el dispositivo sobre el que vamos a depurar.



- **Debug ENVM:** esta herramienta nos permite depurar la memoria no-volátil interna de las FPGAs/SoCs de Microchip. Solo hace falta indicarle una dirección de memoria de inicio y una dirección de memoria de finalización. También cuenta con un botón que permite leer la memoria en el momento exacto de pulsarlo (*Read from Device*).



- **Debug SNVM:** esta herramienta nos permite depurar la memoria SRAM del SoCs, siempre que haya un perfil de memoria grabado en ella.

sNVM Debug

Client View

Page View

Refresh Client Details

Client List	Start Page	End Page	Number of Bytes	Write cycles	Page Type	Used as ROM	USK s
INIT_STAGE_1_SNVN_CLIENT	202	219				Yes	
Page 202	202		252	12	Plain Text	Yes	N/A
Page 203	203		252	12	Plain Text	Yes	N/A
Page 204	204		252	12	Plain Text	Yes	N/A
Page 205	205		252	12	Plain Text	Yes	N/A
Page 206	206		252	12	Plain Text	Yes	N/A

Read from Device

Latest Content Retrieved from Device:

Fri Dec 6 23:31:59 2024

Retrieved Content: Client "INIT_STAGE_1_SNVN_CLIENT".

View All Page Status

Export

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	01	00	10	C0	32	00	00	00	00	0F	11	C0	B8	0B	00	00
0010	FF	3F	0C	12	00	00	00	20	00	00	00	00	00	00	00	00
0020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

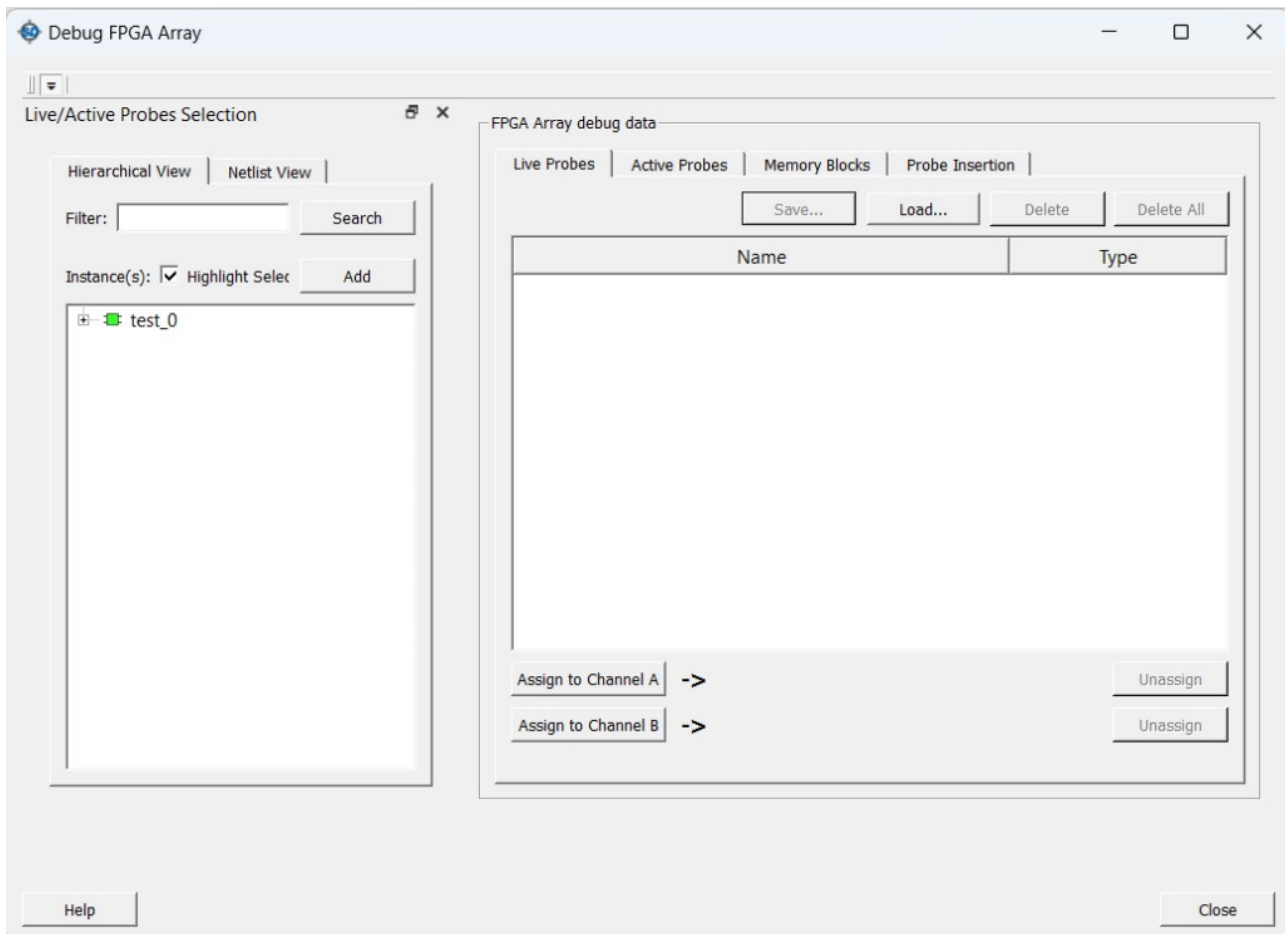
Help

Close

- **Debug FPGA Array:** esta herramienta es la que nos permite depurar las FPGAs con diferentes opciones de depuración.

Creado por David Rubio G.

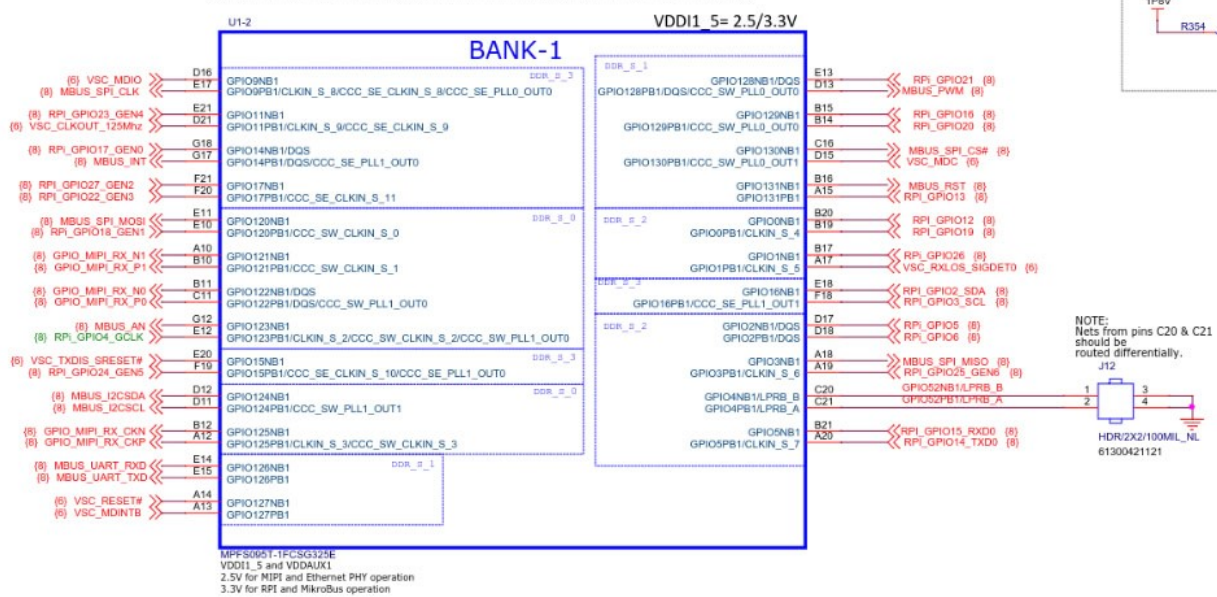
6/23



Debug FPGA Array

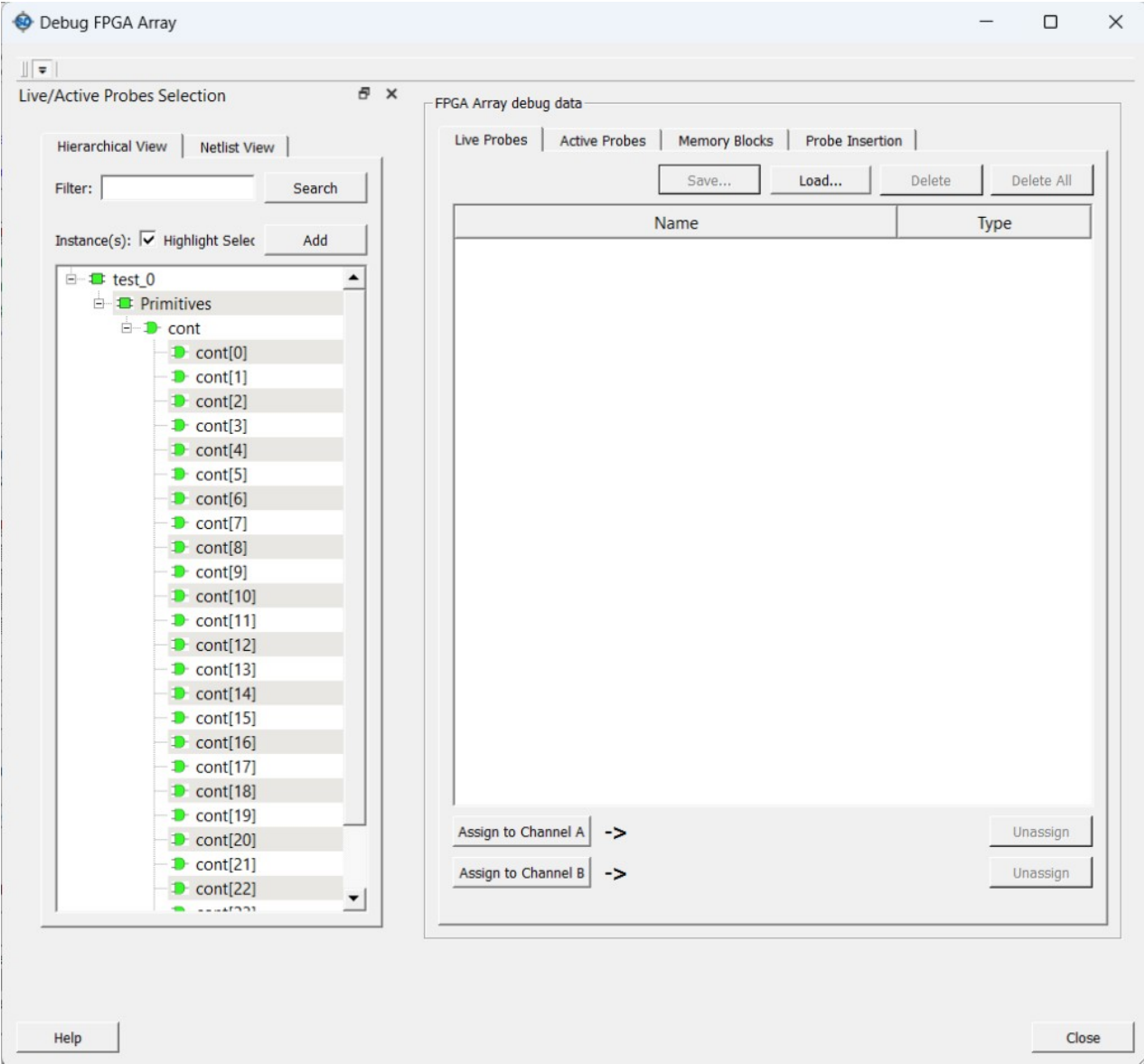
Esta herramienta tiene opciones muy interesantes para depurar un FPGA.

- **Live Probes:** esta opción permite sacar por uno de los dos pines específicos de depuración que tiene la FPGA una de las señales internas de la FPGA (**LPRB_A** y **LPRB_B**).

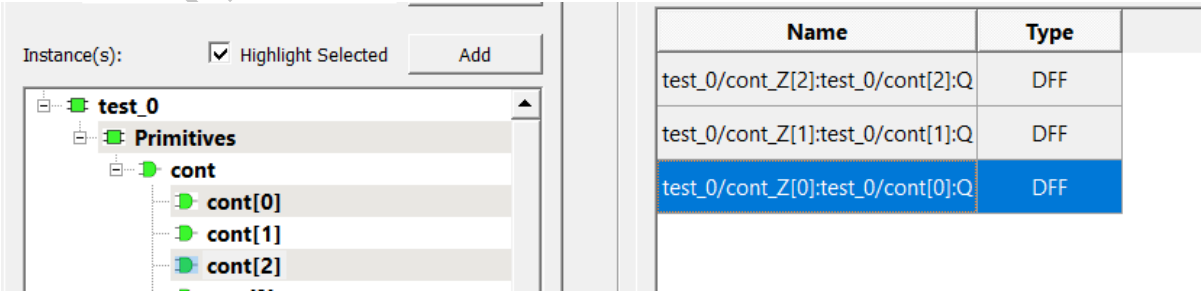


De tal forma que si quiero saber cuál es el estado de una señal interna en tiempo real, puedo sacar por un pin esa señal y visualizarla en un osciloscopio. Esta herramienta está más enfocada a señales que varíen en el tiempo (*para señales discretas hay otra herramienta*)

NOTA: esta herramienta es automática, por lo que al seleccionar una señal sale de forma automática por los pines.



Para ello tengo dos opciones, elegir señal a señal cuáles quiero depurar (doble clic en cada señal que se quiera) o seleccionar todas las señales de un golpe (para ello dos clics en el contenedor de las señales). Pero solo se pueden depurar dos al mismo tiempo.



Para depurar señales, se seleccionan en la interfaz y se le da a los botones de abajo (*Assign to Channel ...*).

Save...Load...DeleteDelete All

Name	Type
test_0/cont_Z[25]:test_0/cont[25]:Q	DFF
test_0/cont_Z[24]:test_0/cont[24]:Q	DFF
test_0/cont_Z[23]:test_0/cont[23]:Q	DFF
test_0/cont_Z[22]:test_0/cont[22]:Q	DFF
test_0/cont_Z[21]:test_0/cont[21]:Q	DFF
test_0/cont_Z[20]:test_0/cont[20]:Q	DFF
test_0/cont_Z[19]:test_0/cont[19]:Q	DFF
test_0/cont_Z[18]:test_0/cont[18]:Q	DFF
test_0/cont_Z[17]:test_0/cont[17]:Q	DFF
test_0/cont_Z[16]:test_0/cont[16]:Q	DFF
test_0/cont_Z[15]:test_0/cont[15]:Q	DFF
test_0/cont_Z[14]:test_0/cont[14]:Q	DFF
test_0/cont_Z[13]:test_0/cont[13]:Q	DFF
test_0/cont_Z[12]:test_0/cont[12]:Q	DFF

Assign to Channel A

-> test_0/cont_Z[18]:test_0/cont[18]:Q

Unassign

Assign to Channel B

->

Unassign

Se pueden seleccionar hasta dos señales, que se pueden depurar al instante en el que se pulsa el botón.

Name	Type
test_0/cont_Z[25]:test_0/cont[25]:Q	DFF
test_0/cont_Z[24]:test_0/cont[24]:Q	DFF
test_0/cont_Z[23]:test_0/cont[23]:Q	DFF
test_0/cont_Z[22]:test_0/cont[22]:Q	DFF
test_0/cont_Z[21]:test_0/cont[21]:Q	DFF
test_0/cont_Z[20]:test_0/cont[20]:Q	DFF
test_0/cont_Z[19]:test_0/cont[19]:Q	DFF
test_0/cont_Z[18]:test_0/cont[18]:Q	DFF
test_0/cont_Z[17]:test_0/cont[17]:Q	DFF
test_0/cont_Z[16]:test_0/cont[16]:Q	DFF
test_0/cont_Z[15]:test_0/cont[15]:Q	DFF
test_0/cont_Z[14]:test_0/cont[14]:Q	DFF
test_0/cont_Z[13]:test_0/cont[13]:Q	DFF
test_0/cont_Z[12]:test_0/cont[12]:Q	DFF

Assign to Channel A

-> test_0/cont_Z[24]:test_0/cont[24]:Q

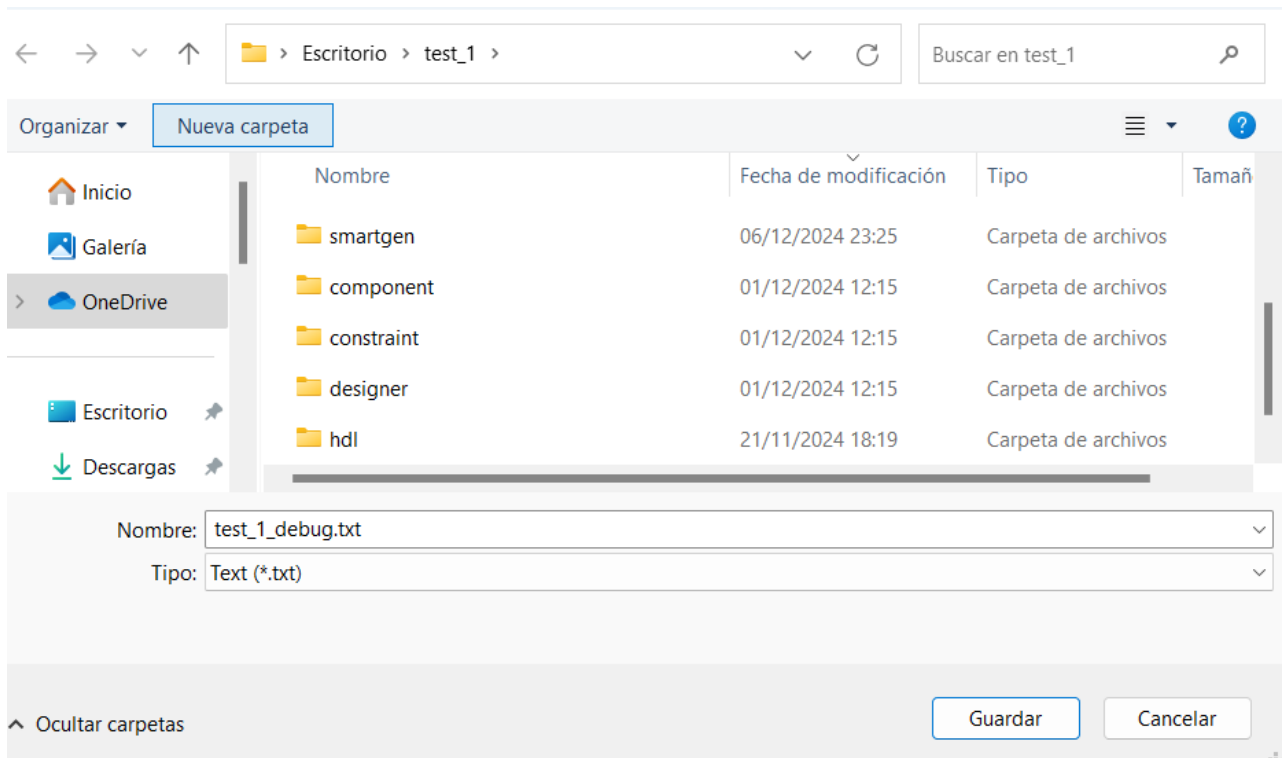
Unassign

Assign to Channel B

-> test_0/cont_Z[25]:test_0/cont[25]:Q

Unassign

Para facilitar depuraciones posteriores, existe la opción de guardar el perfil de depuración, para ello se le da a Save...



Y crea un fichero como el siguiente.

```
Bus test_0/cont_Z[25:0] display_format hex
test_0/cont_Z[25]:test_0/cont[25]:Q
test_0/cont_Z[24]:test_0/cont[24]:Q
test_0/cont_Z[23]:test_0/cont[23]:Q
test_0/cont_Z[22]:test_0/cont[22]:Q
test_0/cont_Z[21]:test_0/cont[21]:Q
test_0/cont_Z[20]:test_0/cont[20]:Q
test_0/cont_Z[19]:test_0/cont[19]:Q
test_0/cont_Z[18]:test_0/cont[18]:Q
test_0/cont_Z[17]:test_0/cont[17]:Q
test_0/cont_Z[16]:test_0/cont[16]:Q
test_0/cont_Z[15]:test_0/cont[15]:Q
test_0/cont_Z[14]:test_0/cont[14]:Q
test_0/cont_Z[13]:test_0/cont[13]:Q
test_0/cont_Z[12]:test_0/cont[12]:Q
test_0/cont_Z[11]:test_0/cont[11]:Q
test_0/cont_Z[10]:test_0/cont[10]:Q
test_0/cont_Z[9]:test_0/cont[9]:Q
test_0/cont_Z[8]:test_0/cont[8]:Q
test_0/cont_Z[7]:test_0/cont[7]:Q
test_0/cont_Z[6]:test_0/cont[6]:Q
test_0/cont_Z[5]:test_0/cont[5]:Q
test_0/cont_Z[4]:test_0/cont[4]:Q
test_0/cont_Z[3]:test_0/cont[3]:Q
test_0/cont_Z[2]:test_0/cont[2]:Q
test_0/cont_Z[1]:test_0/cont[1]:Q
test_0/cont_Z[0]:test_0/cont[0]:Q
Bus end
```

- **Active Probes:** esta opción está hecha para depurar todos los valores discretos internos de la FPGA. Esta herramienta puede leer todos los valores de las señales internas y también puede escribir sobre ellos el valor deseado.

The screenshot displays a software interface with two main panels. The left panel, titled 'Hierarchical View', shows a tree structure under 'test_0' with a sub-entry 'Primitives' containing a list of 'cont' elements from 'cont[0]' to 'cont[25]'. The right panel, titled 'Live Probes', contains a table with columns 'Name', 'Type', 'Read Value', and 'Write Value'. The table lists 26 probes, all of type 'DFF', with 'Read Value' set to 'Unread' and 'Write Value' set to '26'h'. Below the table are buttons for 'Read Active Probes', 'Save Active Probes' Data...', and 'Write Active Probes'.

Name	Type	Read Value	Write Value
test_0/cont_Z[25:0]	DFF	Unread	26'h
test_0/cont.../cont[25]:Q	DFF	Unread	
test_0/cont.../cont[24]:Q	DFF	Unread	
test_0/cont.../cont[23]:Q	DFF	Unread	
test_0/cont.../cont[22]:Q	DFF	Unread	
test_0/cont.../cont[21]:Q	DFF	Unread	
test_0/cont.../cont[20]:Q	DFF	Unread	
test_0/cont.../cont[19]:Q	DFF	Unread	
test_0/cont.../cont[18]:Q	DFF	Unread	
test_0/cont.../cont[17]:Q	DFF	Unread	
test_0/cont.../cont[16]:Q	DFF	Unread	
test_0/cont.../cont[15]:Q	DFF	Unread	
test_0/cont.../cont[14]:Q	DFF	Unread	
test_0/cont.../cont[13]:Q	DFF	Unread	
test_0/cont.../cont[12]:Q	DFF	Unread	
test_0/cont.../cont[11]:Q	DFF	Unread	
test_0/cont.../cont[10]:Q	DFF	Unread	
test_0/cont.../cont[9]:Q	DFF	Unread	
test_0/cont.../cont[8]:Q	DFF	Unread	
test_0/cont.../cont[7]:Q	DFF	Unread	
test_0/cont.../cont[6]:Q	DFF	Unread	
test_0/cont.../cont[5]:Q	DFF	Unread	
test_0/cont.../cont[4]:Q	DFF	Unread	
test_0/cont.../cont[3]:Q	DFF	Unread	
test_0/cont.../cont[2]:Q	DFF	Unread	
test_0/cont.../cont[1]:Q	DFF	Unread	

Para ello se le da al botón *Read Active Probes*, entonces, se pueden ver todos los valores internos de las señales que se han seleccionado.

FPGA Array debug data

Live Probes | Active Probes | Memory Blocks | Probe Insertion

Name	Type	Read Value	Write Value
test_0/cont_Z[25:0]	DFF	26'h0B9594E	26'h
test_0/cont.../cont[25]:Q	DFF	0	▼
test_0/cont.../cont[24]:Q	DFF	0	▼
test_0/cont.../cont[23]:Q	DFF	1	▼
test_0/cont.../cont[22]:Q	DFF	0	▼
test_0/cont.../cont[21]:Q	DFF	1	▼
test_0/cont.../cont[20]:Q	DFF	1	▼
test_0/cont.../cont[19]:Q	DFF	1	▼
test_0/cont.../cont[18]:Q	DFF	0	▼
test_0/cont.../cont[17]:Q	DFF	0	▼
test_0/cont.../cont[16]:Q	DFF	1	▼
test_0/cont.../cont[15]:Q	DFF	0	▼
test_0/cont.../cont[14]:Q	DFF	1	▼
test_0/cont.../cont[13]:Q	DFF	0	▼
test_0/cont.../cont[12]:Q	DFF	1	▼
test_0/cont.../cont[11]:Q	DFF	1	▼
test_0/cont.../cont[10]:Q	DFF	0	▼
test_0/cont...0/cont[9]:Q	DFF	0	▼
test_0/cont...0/cont[8]:Q	DFF	1	▼
test_0/cont...0/cont[7]:Q	DFF	0	▼
test_0/cont...0/cont[6]:Q	DFF	1	▼
test_0/cont...0/cont[5]:Q	DFF	0	▼
test_0/cont...0/cont[4]:Q	DFF	0	▼
test_0/cont...0/cont[3]:Q	DFF	1	▼
test_0/cont...0/cont[2]:Q	DFF	1	▼
test_0/cont...0/cont[1]:Q	DFF	1	▼

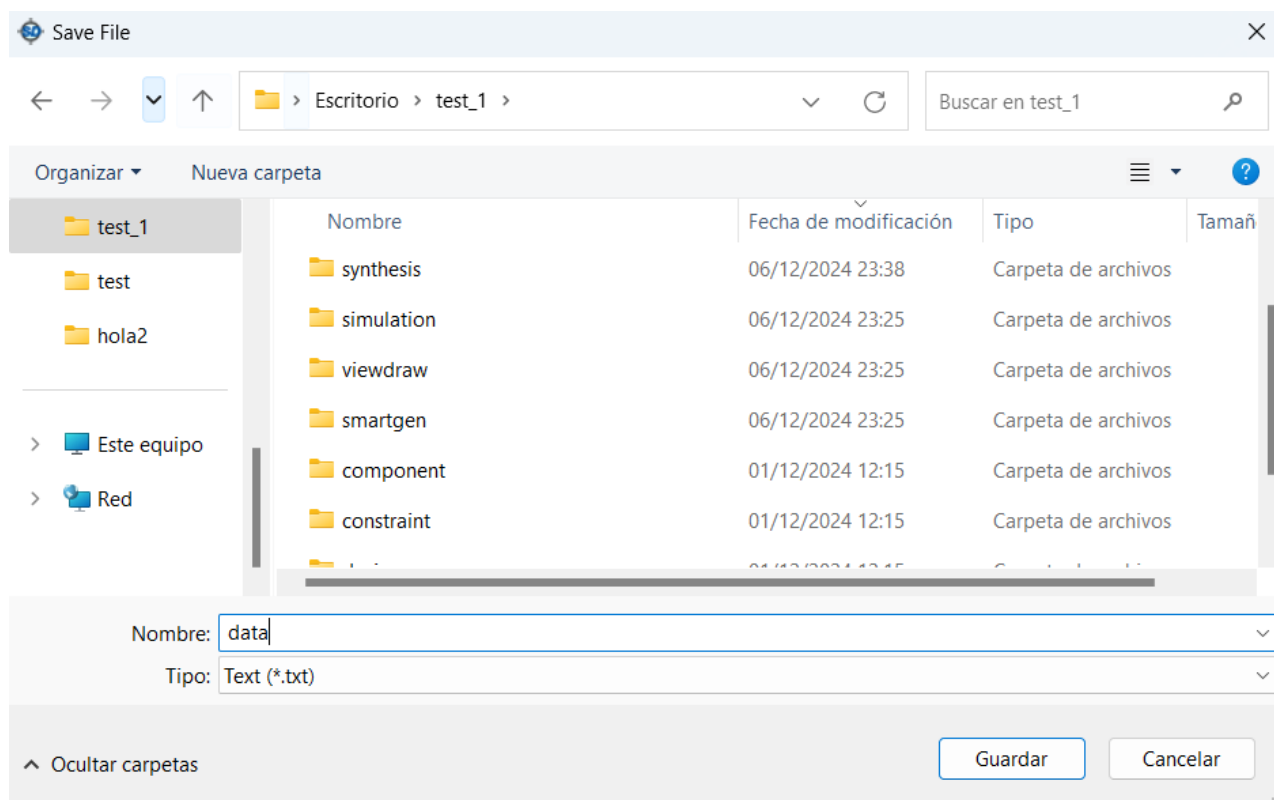
Para escribir un valor interno se tiene que cambiar el valor de la señal deseado y después se le da al botón *Write Active Probes*.

FPGA Array debug data

Live Probes | Active Probes | Memory Blocks | Probe Insertion

Name	Type	Read Value	Write Value
test_0/cont_Z[25:0]	DFF	26'h0B9594E	26'h
test_0/cont.../cont[25]:Q	DFF	0	1
test_0/cont.../cont[24]:Q	DFF	0	1
test_0/cont.../cont[23]:Q	DFF	1	0
test_0/cont.../cont[22]:Q	DFF	0	1
test_0/cont.../cont[21]:Q	DFF	1	
test_0/cont.../cont[20]:Q	DFF	1	
test_0/cont.../cont[19]:Q	DFF	1	
test_0/cont.../cont[18]:Q	DFF	0	
test_0/cont.../cont[17]:Q	DFF	0	
test_0/cont.../cont[16]:Q	DFF	1	
test_0/cont.../cont[15]:Q	DFF	0	
test_0/cont.../cont[14]:Q	DFF	1	
test_0/cont.../cont[13]:Q	DFF	0	
test_0/cont.../cont[12]:Q	DFF	1	
test_0/cont.../cont[11]:Q	DFF	1	
test_0/cont.../cont[10]:Q	DFF	0	
test_0/cont...0/cont[9]:Q	DFF	0	
test_0/cont...0/cont[8]:Q	DFF	1	
test_0/cont...0/cont[7]:Q	DFF	0	
test_0/cont...0/cont[6]:Q	DFF	1	
test_0/cont...0/cont[5]:Q	DFF	0	
test_0/cont...0/cont[4]:Q	DFF	0	
test_0/cont...0/cont[3]:Q	DFF	1	
test_0/cont...0/cont[2]:Q	DFF	1	
test_0/cont...0/cont[1]:Q	DFF	1	

Si se quieren grabar los valores leídos en un fichero, solo hace falta darle a la opción *Save Active Probes' Data*.



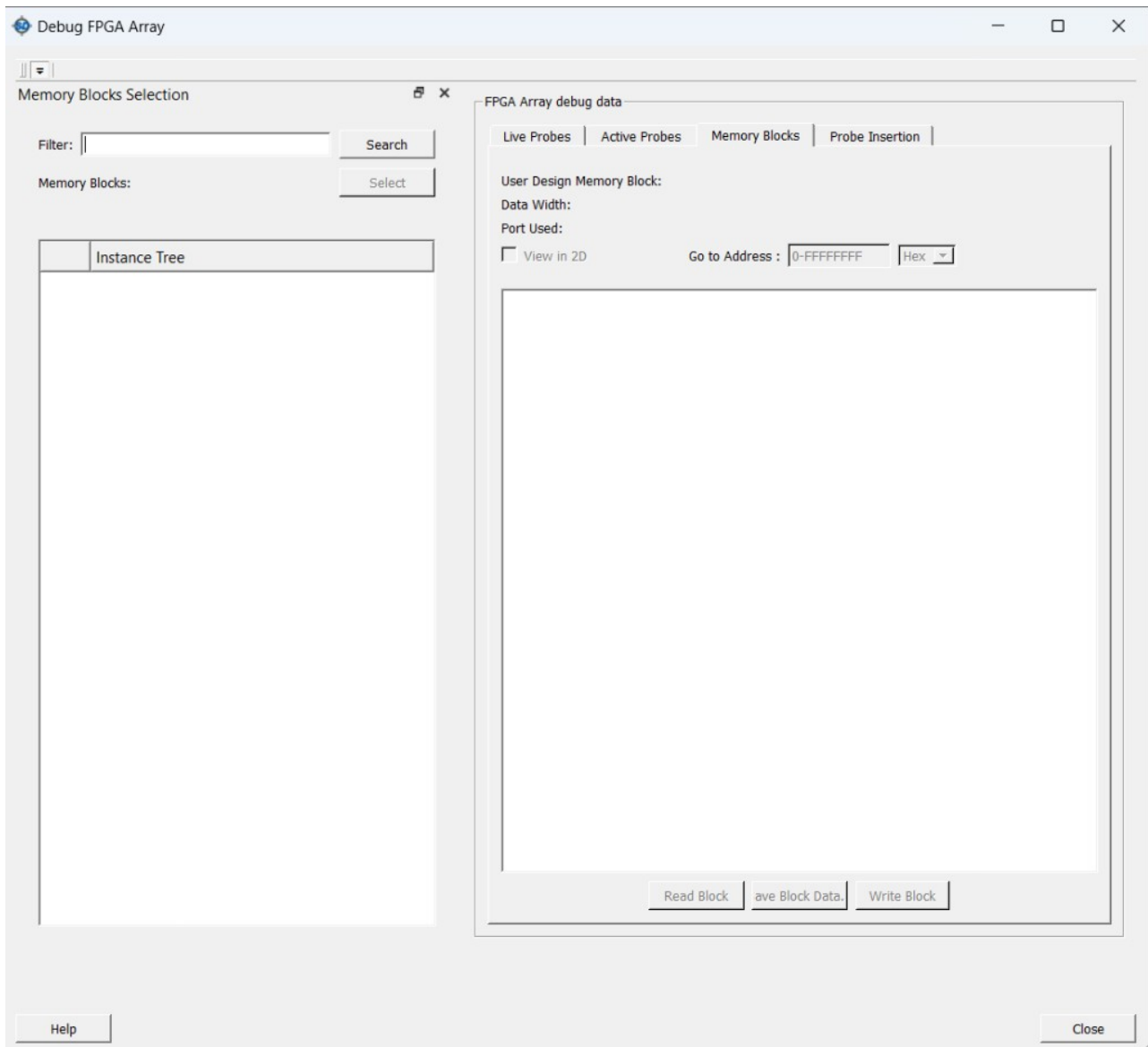
Esto genera un fichero de texto con la siguiente forma.

Active probes and values for .dprj

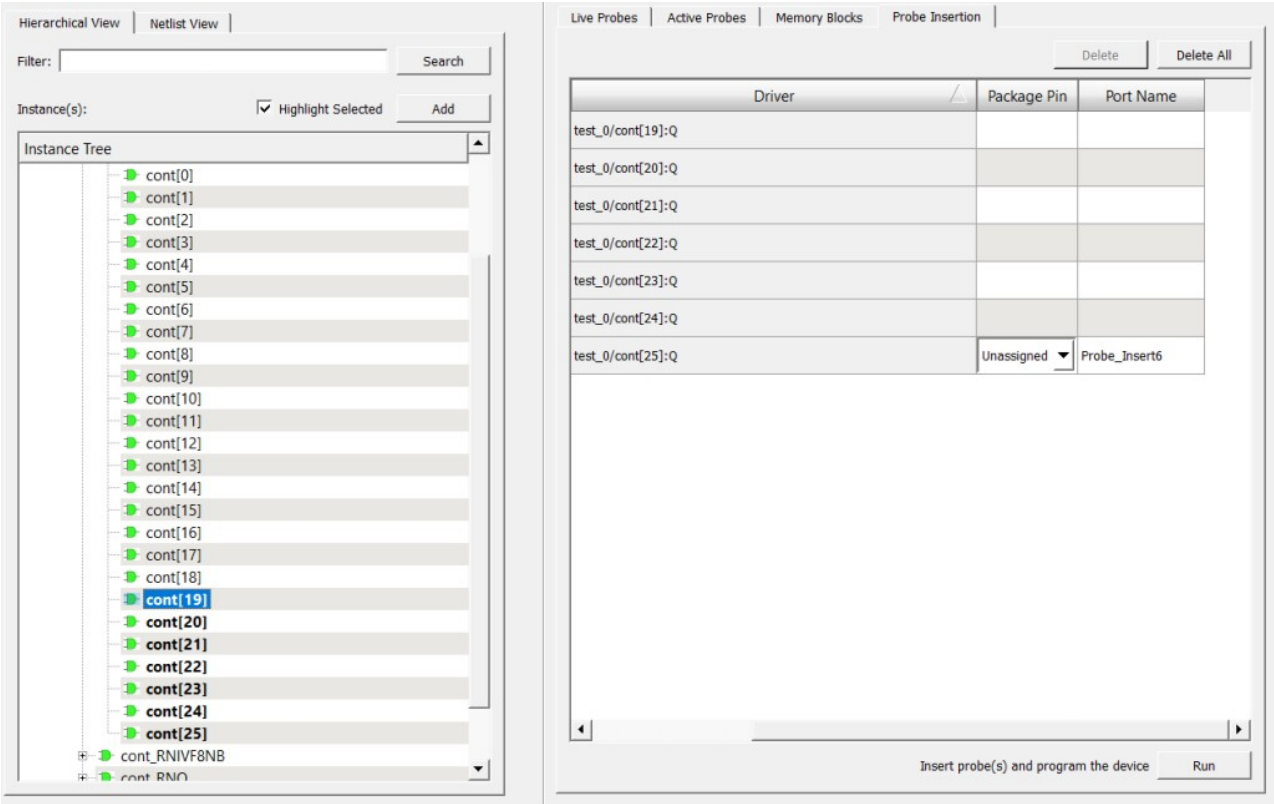
Saved at: vi. dic. 6 23:48:27 2024

Name	Type	Read Value
test_0/cont_Z[25:0]	DFF	26'h31FDA74
test_0/cont_Z[25]:test_0/cont[25]:Q	DFF	1
test_0/cont_Z[24]:test_0/cont[24]:Q	DFF	1
test_0/cont_Z[23]:test_0/cont[23]:Q	DFF	0
test_0/cont_Z[22]:test_0/cont[22]:Q	DFF	0
test_0/cont_Z[21]:test_0/cont[21]:Q	DFF	0
test_0/cont_Z[20]:test_0/cont[20]:Q	DFF	1
test_0/cont_Z[19]:test_0/cont[19]:Q	DFF	1
test_0/cont_Z[18]:test_0/cont[18]:Q	DFF	1
test_0/cont_Z[17]:test_0/cont[17]:Q	DFF	1
test_0/cont_Z[16]:test_0/cont[16]:Q	DFF	1
test_0/cont_Z[15]:test_0/cont[15]:Q	DFF	1
test_0/cont_Z[14]:test_0/cont[14]:Q	DFF	1
test_0/cont_Z[13]:test_0/cont[13]:Q	DFF	0
test_0/cont_Z[12]:test_0/cont[12]:Q	DFF	1
test_0/cont_Z[11]:test_0/cont[11]:Q	DFF	1
test_0/cont_Z[10]:test_0/cont[10]:Q	DFF	0
test_0/cont_Z[9]:test_0/cont[9]:Q	DFF	1
test_0/cont_Z[8]:test_0/cont[8]:Q	DFF	0
test_0/cont_Z[7]:test_0/cont[7]:Q	DFF	0
test_0/cont_Z[6]:test_0/cont[6]:Q	DFF	1
test_0/cont_Z[5]:test_0/cont[5]:Q	DFF	1
test_0/cont_Z[4]:test_0/cont[4]:Q	DFF	1
test_0/cont_Z[3]:test_0/cont[3]:Q	DFF	0
test_0/cont_Z[2]:test_0/cont[2]:Q	DFF	1
test_0/cont_Z[1]:test_0/cont[1]:Q	DFF	0
test_0/cont_Z[0]:test_0/cont[0]:Q	DFF	0

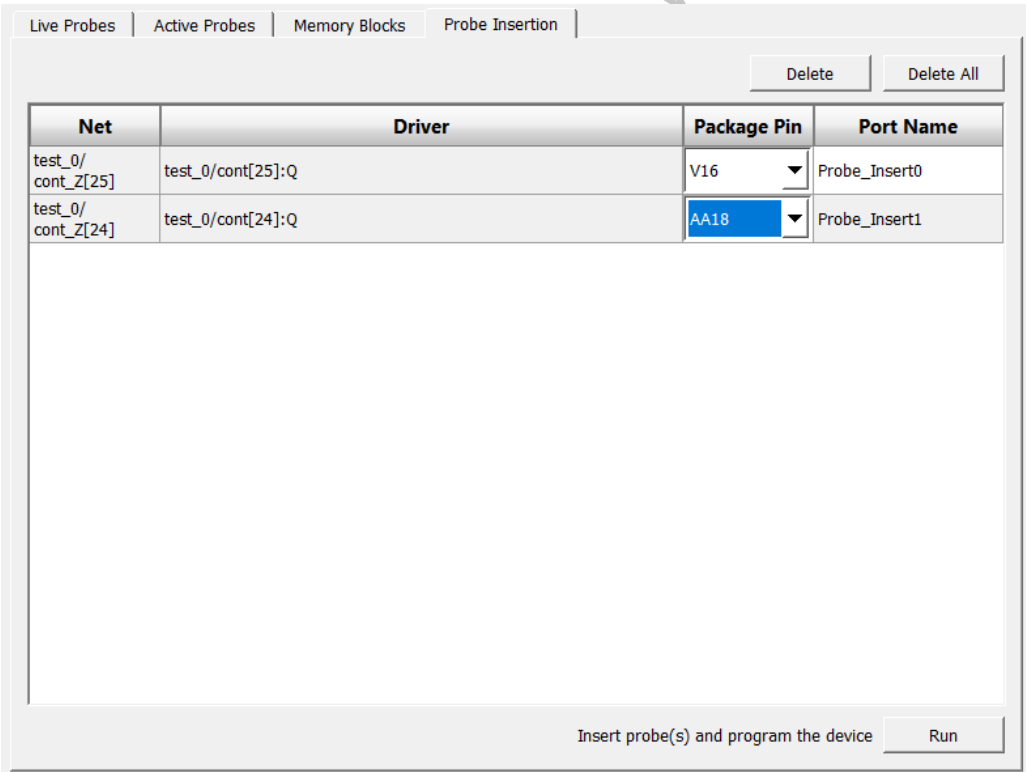
- **Memory Block:** esta es una opción para leer la memoria interna bloque a bloque. Esta opción solo está habilitada si existe un bloque de memoria predefinido para una FPGA/SoC.



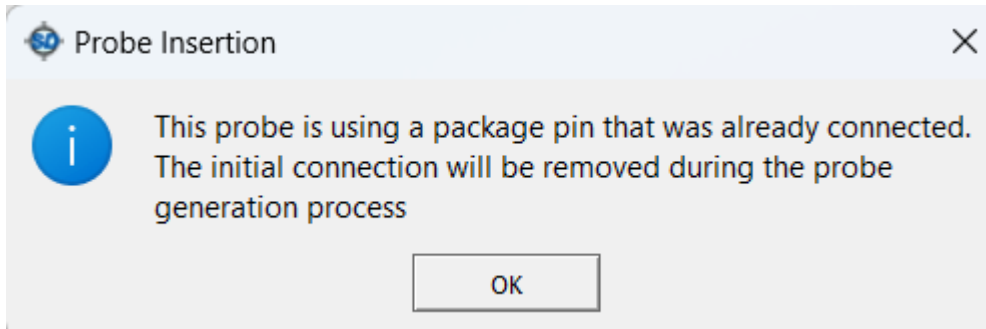
- **Probe Insertion:** Esta opción hace lo mismo que el *Live Probes*, pero en vez de sacar la señal por dos pines específicos de la FPGA, puedes elegir cualquier pin del encapsulado para sacar la señal. Esta opción está creada para sacar señales por pines libres de la FPGA. Para ello, primero se selecciona las señales que se quieren sacar.



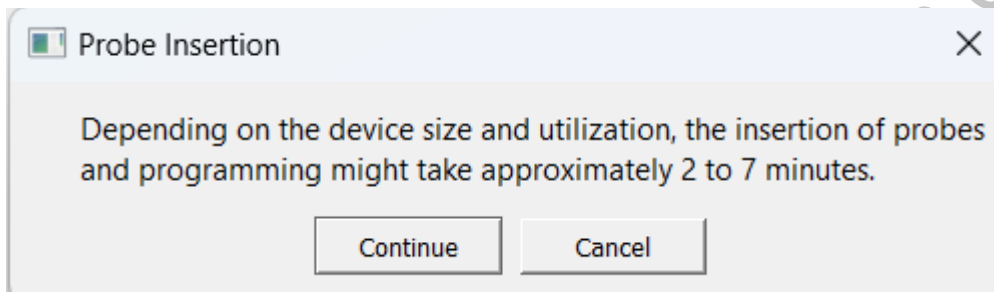
Y después se selecciona el pin por el que se quiere sacar la señal.



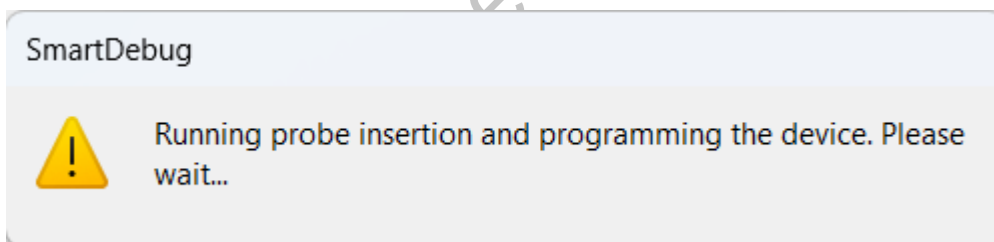
NOTA: se puede sacar por cualquier pin, incluso aquellos que han sido seleccionados para generar el bitstream. Si lo haces salta esta pestaña.



Una vez tienes elegidas las señales que quieres sacar junto con los pines, se le da a *Run*. Entonces, salta una pestaña como la siguiente. En ella se dice que se puede tardar entre 2 a 7 minutos en que haga efecto, eso es porque lo que hace internamente es generar un nuevo bitstream para la depuración, para ello se aprovecha de la síntesis ya hecha y hace un nuevo *Place and Route*, y después carga un nuevo bitstream.



Cuando hace la nueva carga del bitstream la FPGA se resetea y vuelve a empezar de cero, por lo que no carga el bitstream de forma dinámica. Es importante tener esto en cuenta. **Esto no ocurre con las Live Probes.**

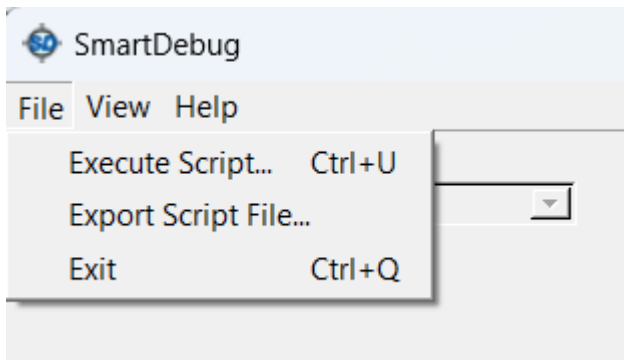


Una vez terminado, saca las nuevas señales por los pines deseados.

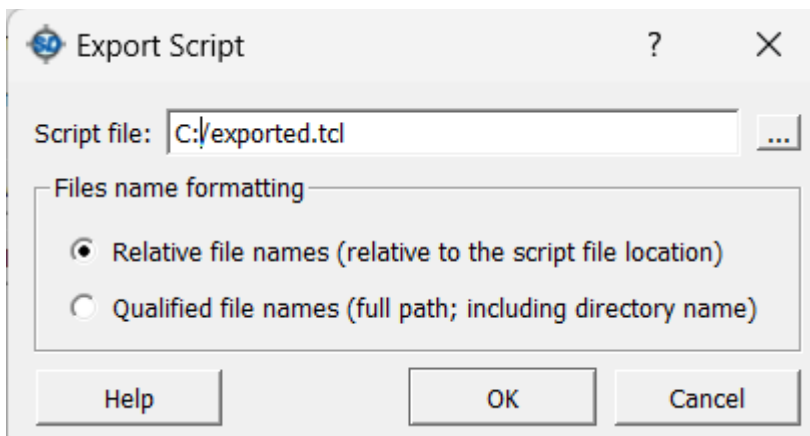
Nota final

Si se quiere guardar los perfiles de depuración para facilitar depuraciones posteriores, lo que se puede hacer es exportar un fichero TCL con las configuraciones utilizadas.

Para ello en File, se da a *Export Script File...*



Al darle nos pide que le digamos la ruta en la que guardar el fichero.



Al guardarlo nos genera un fichero como el siguiente.

```
# Microsemi Tcl Script
# sdbg
# Date: Sat Dec 7 00:11:33 2024
# Directory C:\Users\David\Desktop\test_1
# File C:\..\exported.tcl

export_script -file {./exported.tcl} -relative_path 1
select_active_probe \
    -name {test_0/cont_Z[0]:test_0/cont[0]:Q} \
    -name {test_0/cont_Z[10]:test_0/cont[10]:Q} \
    -name {test_0/cont_Z[11]:test_0/cont[11]:Q} \
    -name {test_0/cont_Z[12]:test_0/cont[12]:Q} \
    -name {test_0/cont_Z[13]:test_0/cont[13]:Q} \
    -name {test_0/cont_Z[14]:test_0/cont[14]:Q} \
    -name {test_0/cont_Z[15]:test_0/cont[15]:Q} \
    -name {test_0/cont_Z[16]:test_0/cont[16]:Q} \
    -name {test_0/cont_Z[17]:test_0/cont[17]:Q} \
    -name {test_0/cont_Z[18]:test_0/cont[18]:Q} \
    -name {test_0/cont_Z[19]:test_0/cont[19]:Q} \
    -name {test_0/cont_Z[1]:test_0/cont[1]:Q} \
    -name {test_0/cont_Z[20]:test_0/cont[20]:Q} \
    -name {test_0/cont_Z[21]:test_0/cont[21]:Q} \
    -name {test_0/cont_Z[22]:test_0/cont[22]:Q} \
    -name {test_0/cont_Z[23]:test_0/cont[23]:Q} \
    -name {test_0/cont_Z[24]:test_0/cont[24]:Q} \
    -name {test_0/cont_Z[25]:test_0/cont[25]:Q} \
    -name {test_0/cont_Z[2]:test_0/cont[2]:Q} \
    -name {test_0/cont_Z[3]:test_0/cont[3]:Q} \
    -name {test_0/cont_Z[4]:test_0/cont[4]:Q} \
    -name {test_0/cont_Z[5]:test_0/cont[5]:Q} \
    -name {test_0/cont_Z[6]:test_0/cont[6]:Q} \
    -name {test_0/cont_Z[7]:test_0/cont[7]:Q} \
    -name {test_0/cont_Z[8]:test_0/cont[8]:Q} \
    -name {test_0/cont_Z[9]:test_0/cont[9]:Q} \
    -reset 0
read_active_probe
export_script -file {./exported.tcl} -relative_path 1
remove_probe_insertion_point -net {test_0/cont_Z[25]} -driver {test_0/cont[25]:Q}
program_probe_insertion -all 0
```

Para importarlo le damos a *Execute Script...*

Referencias

Para más info tenéis el siguiente video.

<https://youtu.be/7-hmSy-TdPI>