

Cómo funciona el sistema de arranque en la Flash de una FPGA de Xilinx

Creador: David Rubio G.

Entrada: <https://soceame.wordpress.com/2025/01/10/como-funciona-el-sistema-de-arranque-en-la-flash-de-una-fpga-de-xilinx/>

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

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

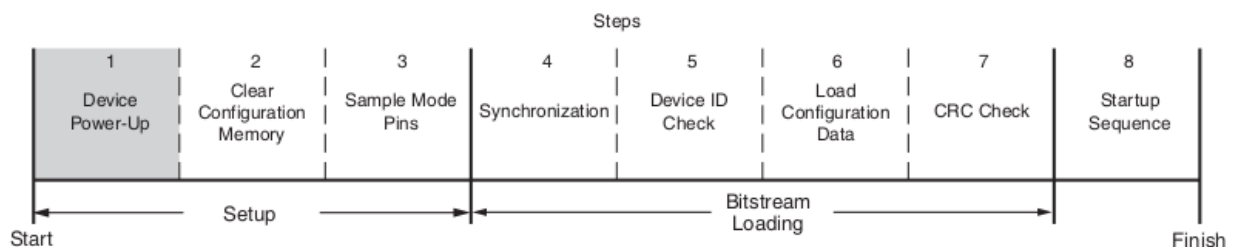
Fecha última modificación: 24/02/2025

En una entrada anterior y explique cómo se graba una FPGA de Xilinx utilizando Vivado. Ahora voy a explicar cuál es el diseño que tiene que tener la PCB de la FPGA para que esto ocurra.

<https://soceame.wordpress.com/2024/06/09/como-grabar-una-fpga-de-xilinx/>

Arranque de la FPGA

El sistema de arranque de una FPGA de Xilinx sigue el siguiente esquema. En este esquema se puede ver que hay un arranque de la FPGA donde se realiza una limpieza interna, seguido de una lectura del bitstream externo que viene de la Flash con una carga del bitstream y una comprobación de que el bitstream está correctamente grabado.



Además, las FPGAs de Xilinx tienen varios modos de arranque, esto define desde qué lugar se obtiene el bitstream al arrancar la FPGA.

El método más normal para el arranque de una FPGA es desde una memoria Flash que esté en la misma placa que la FPGA (la forma de conexión se explica en el siguiente apartado).

Configuration Mode	M[2:0]	Bus Width	CCLK Direction
Master Serial	000	x1	Output
Master SPI	001	x1, x2, x4	Output
Master BPI	010	x8, x16	Output
Master SelectMAP	100	x8, x16	Output
JTAG	101	x1	Not Applicable
Slave SelectMAP	110	x8, x16, x32 ⁽¹⁾	Input
Slave Serial ⁽²⁾	111	x1	Input

Nota: esta tabla se aplica tanto a la serie 7 como a la UltraScale.

Para que estos modos funcionen, es necesario que los pines de la FPGA que tienen características especiales de arranque estén conectados según las tablas que da fabricante, para así garantizar el arranque (la siguiente tabla es para la Serie 7).

Table 2-2: Configuration Mode Pins (Table 1 of 2)

Pin Name	Bank	JTAG (Only)	Slave Serial	Master Serial	Master SPI		
					x1	x2	x4
CFGBVS	0	CFGBVS	CFGBVS	CFGBVS	CFGBVS	CFGBVS	CFGBVS
M[2:0]	0	M[2:0]=101	M[2:0]=111	M[2:0]=000	M[2:0]=001	M[2:0]=001	M[2:0]=001
TCK	0	TCK	TCK	TCK	TCK	TCK	TCK
TMS	0	TMS	TMS	TMS	TMS	TMS	TMS
TDI	0	TDI	TDI	TDI	TDI	TDI	TDI
TDO	0	TDO	TDO	TDO	TDO	TDO	TDO
PROGRAM_B	0	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B
INIT_B	0	INIT_B	INIT_B	INIT_B	INIT_B	INIT_B	INIT_B
DONE	0	DONE	DONE	DONE	DONE	DONE	DONE
CCLK	0	CCLK	CCLK	CCLK	CCLK	CCLK	CCLK
PUDC_B ⁽¹⁾	14	PUDC_B ⁽¹⁾	PUDC_B ⁽¹⁾	PUDC_B ⁽¹⁾	PUDC_B ⁽¹⁾	PUDC_B ⁽¹⁾	PUDC_B ⁽¹⁾
EMCCLK ⁽²⁾	14	–	–	EMCCLK ⁽²⁾	EMCCLK ⁽²⁾	EMCCLK ⁽²⁾	EMCCLK ⁽²⁾
CSL_B	14	–	–	–	–	–	–
DOUT_CSO_B ⁽³⁾⁽⁴⁾	14	–	[DOUT] ⁽³⁾	[DOUT] ⁽³⁾	[DOUT] ⁽³⁾	–	–
RDWR_B	14	–	–	–	–	–	–
FCS_B	14	–	–	–	FCS_B	FCS_B	FCS_B
D00_MOSI	14	–	–	–	MOSI	MOSI/D00	MOSI/D00
D01_DIN	14	–	DIN	DIN	DIN	DIN/D01	DIN/D01
D02	14	–	–	–	–	–	D02
D03	14	–	–	–	–	–	D03
D[04-07]	14	–	–	–	–	–	–
D[08-15]	14	–	–	–	–	–	–
A[00-15]_D[16-31]	14	–	–	–	–	–	–
A[16-28]	15	–	–	–	–	–	–
FOE_B	15	–	–	–	–	–	–
FWE_B	15	–	–	–	–	–	–
ADV_B	15	–	–	–	–	–	–

Por ejemplo, si lo que se quieres es que la FPGA arranque desde una memoria Flash utilizando un *master SPI x4* (es importante recordar que la memoria flash tiene que ser compatible), solo sería necesario conectar los pines que establece la columna correspondiente. Aunque de esta configuración no sería necesario conectar el EMCCLK, debido a que la FPGA actúa como maestra de la memoria Flash.

Para el caso de una UltraScale también hay una tabla propia.

Table 1-7: Configuration Pins - Serial Modes

Pin Name	Bank	JTAG (Only)	Master SPI				Slave Serial	Master Serial
			x1	x2	x4	x8 (dual x4)		
POR_OVERRIDE	N/A	POR_OVERRIDE	POR_OVERRIDE	POR_OVERRIDE	POR_OVERRIDE	POR_OVERRIDE	POR_OVERRIDE	POR_OVERRIDE
VBATT	N/A	VBATT	VBATT	VBATT	VBATT	VBATT	VBATT	VBATT
CFGBVS ⁽¹⁾	0	CFGBVS	CFGBVS	CFGBVS	CFGBVS	CFGBVS	CFGBVS	CFGBVS
M[2:0]	0	M[2:0]=101	M[2:0]=001	M[2:0]=001	M[2:0]=001	M[2:0]=001	M[2:0]=111	M[2:0]=000
TCK	0	TCK	TCK	TCK	TCK	TCK	TCK	TCK
TMS	0	TMS	TMS	TMS	TMS	TMS	TMS	TMS
TDI	0	TDI	TDI	TDI	TDI	TDI	TDI	TDI
TDO	0	TDO	TDO	TDO	TDO	TDO	TDO	TDO
PROGRAM_B	0	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B	PROGRAM_B
INIT_B	0	INIT_B	INIT_B	INIT_B	INIT_B	INIT_B	INIT_B	INIT_B
DONE	0	DONE	DONE	DONE	DONE	DONE	DONE	DONE
CCLK	0	CCLK	CCLK	CCLK	CCLK	CCLK	CCLK	CCLK
PUDC_B ⁽²⁾	0	PUDC_B ⁽²⁾	PUDC_B ⁽²⁾	PUDC_B ⁽²⁾	PUDC_B ⁽²⁾	PUDC_B ⁽²⁾	PUDC_B ⁽²⁾	PUDC_B ⁽²⁾
RDWR_FCS_B	0	-	FCS_B	FCS_B	FCS_B	FCS_B	-	-
D00_MOSI	0	-	MOSI	D00_MOSI	D00_MOSI	D00_MOSI	-	-
D01_DIN	0	-	DIN	D01_DIN	D01_DIN	D01_DIN	DIN	DIN
D02	0	-	-	-	D02	D02	-	-
D03	0	-	-	-	D03	D03	-	-
D[07:04]	65	-	-	-	-	D[07:04]	-	-
D[15:08]	65	-	-	-	-	-	-	-
A[15:00]_D[31:16]	65	-	-	-	-	-	-	-
A[28:16]	65	-	-	-	-	-	-	-
EMCCLK ⁽³⁾	65	-	EMCCLK ⁽³⁾	EMCCLK ⁽³⁾	EMCCLK ⁽³⁾	EMCCLK ⁽³⁾	-	EMCCLK ⁽³⁾
CS1_ADV_B	65	-	-	-	-	-	-	-
DOUT_CSO_B ⁽⁴⁾⁽⁵⁾	65	-	DOUT ⁽⁴⁾	-	-	-	DOUT ⁽⁴⁾	DOUT ⁽⁴⁾
FOE_B	65	-	-	-	-	-	-	-
FWE_FCS2_B	65	-	-	-	-	FCS2_B	-	-
RS0, RS1 ⁽⁶⁾	65	-	-	-	-	-	-	-

Pines de la FPGA/SoC

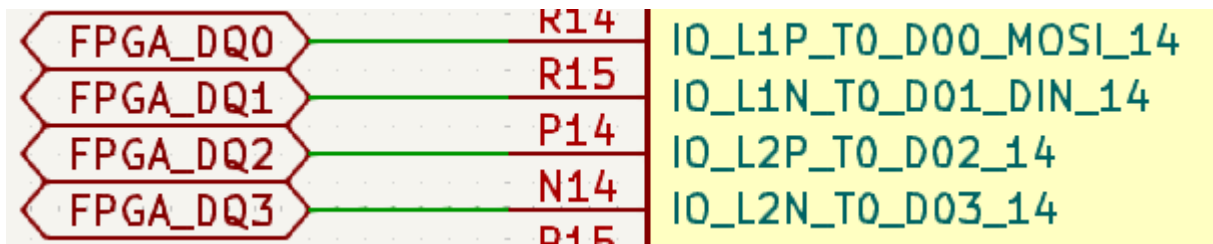
Para programar de la Flash

Para programar la FPGA se utilizan memorias Flash por SPI, para esta tarea se requieren de los diferentes pines. Estos pines suelen estar en los bancos 14 de las FPGAs de la serie 7.

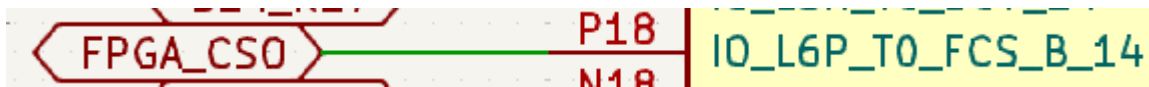
- **CCLK:** este es un pin de salida que hace de reloj con la memoria Flash por SPI, este pin es específico de la comunicación.



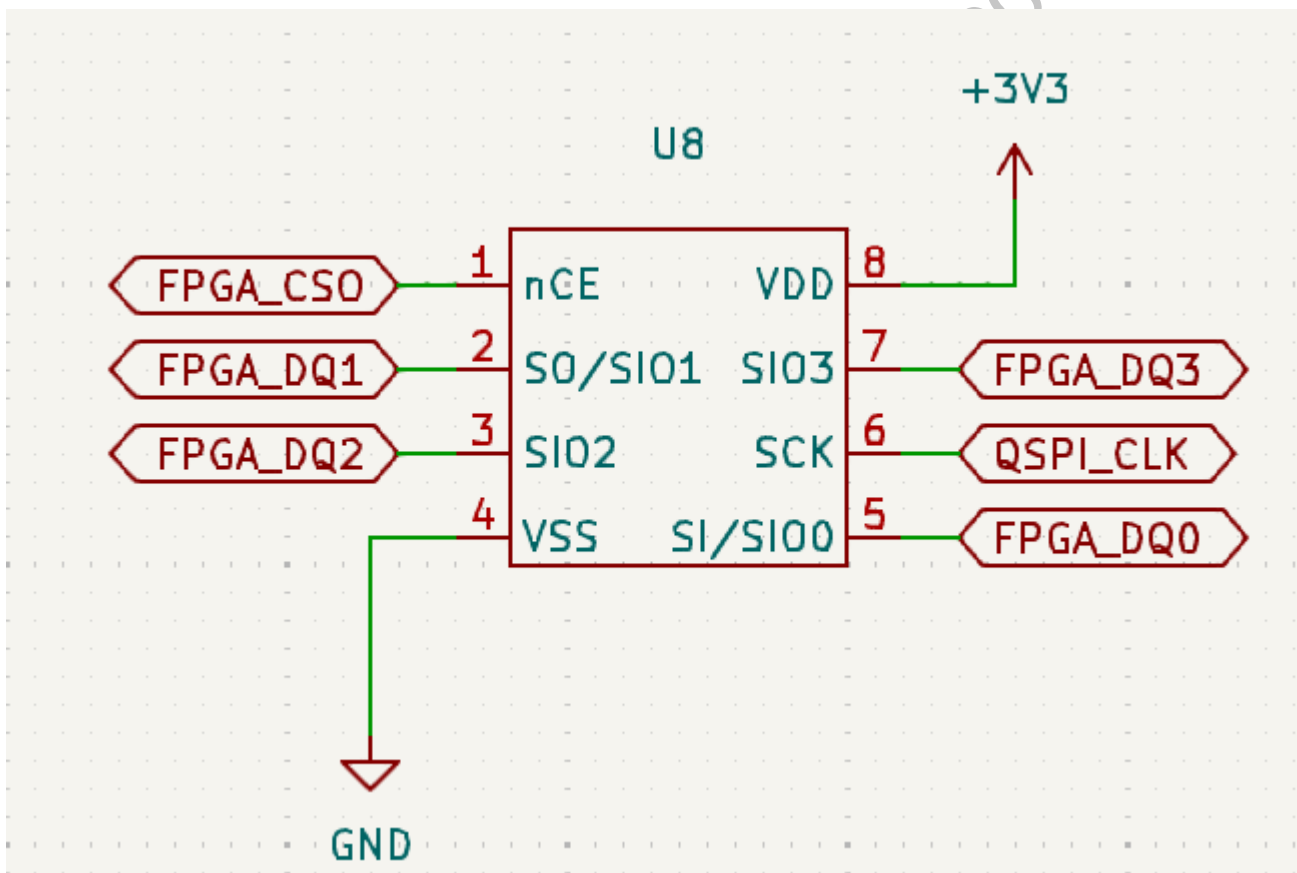
- **D0x:** estos pines se utilizan para la memoria QSPI. Estos pines son configurables, por los que el usuario los pueden usar con la FPGA y también para el arranque por QSPI.



- **FCS_B/FWE_FCS2_B:** este es pin CS para SPI que se utiliza para grabar la FPGA. Este pin es programable, por lo que el usuario lo puede programar.

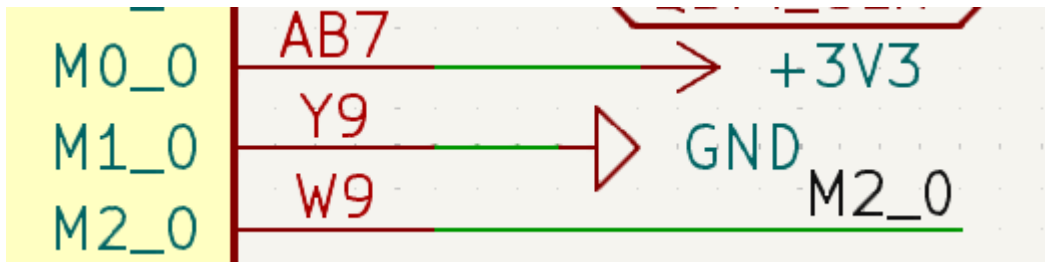


La memoria quedaría configurada de la siguiente manera.

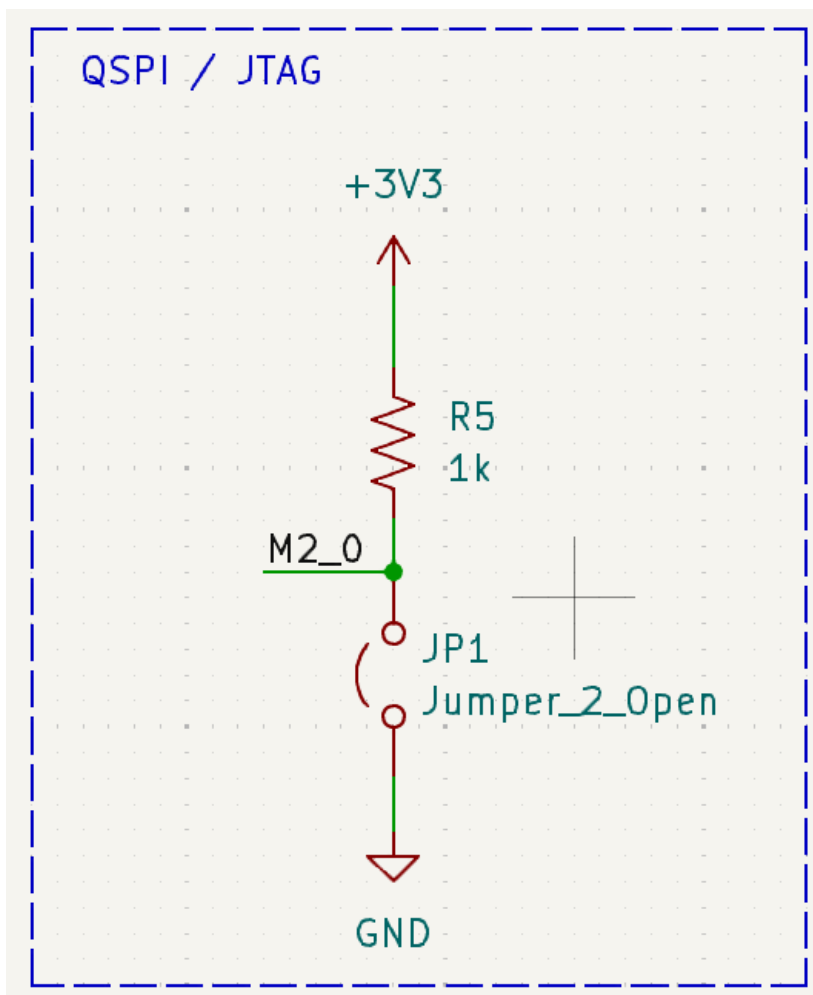


Pines de modo

Estos son los pines que eligen el modo de arranque de la FPGA, se puede ver en la primera tabla la forma de programarse.



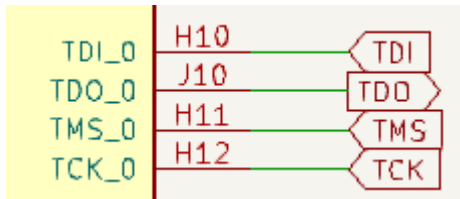
Si por ejemplo se quisiese seleccionar el tipo de arranque entre JTAG y QSPI, se puede utilizar un esquema como el siguiente. Donde por defecto las entradas quedan fijadas con un «101», que es un arranque por JTAG, y con un jumper se puede dejar en el modo «001» que es el modo de arranque por Master SPI.



NOTA: el arranque por JTAG no influye en la grabación de la FPGA para hacer pruebas. Por lo que puedes dejar la FPGA en arranque por JTAG y seguir reprogramando la FPGA sin ningún problema, al igual que ocurre con los otros modos de arranque.

Pines de programación

Entre los otros pines que quedan reflejados en las FPGAs de Xilinx son los pines del JTAG, que son *TDI*, *TDO*, *TMS* y *TCK*.



Estos pines van directos al conector JTAG, o en el caso de FPGAs de nueva generación a un FT2232 o FT4232. Este último chip hace la conversión de JTAG a RS232 para poder conectar la FPGA por USB sin cable depurador.

El FT2232/4232 también permite tener otro tipo de interfaces con el mismo chip, como por ejemplo, otras UARTs u otros protocolos.

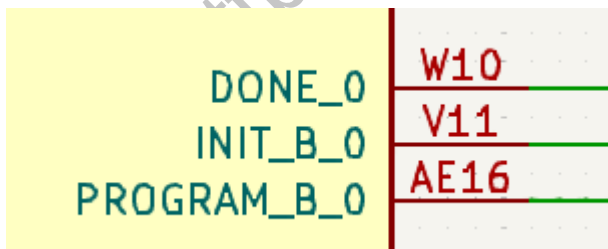
Pero este último sistema de carga de la FPGA tiene un problema y es que el FT2232 tiene una memoria acoplada a unos pines, que lo que hace es grabar una configuración interna dentro antes de empezar a funcionar, y esa configuración es la que utiliza Vivado para reconocer el FT2232 como un cable depurador, pero eso puede significar que si el código interno que tiene esa memoria no se corresponde con el chip que tienes, este sistema del FT2232 puede no funcionar.

NOTA: tanto los clones de los cables depuradores de Xilinx como los cables depuradores de Xilinx llevan un FT2232, lo que pasa es que los clones lo que han hecho es sacar el código del FT2232 para grabárselo al clon. Y si ese clon tiene el código de una familia distinta a la del chip que quieres utilizar, puede no funcionar.

Pines indicadores

Las FPGAs de Xilinx tienen pines indicadores del estado de la carga de la FPGA.

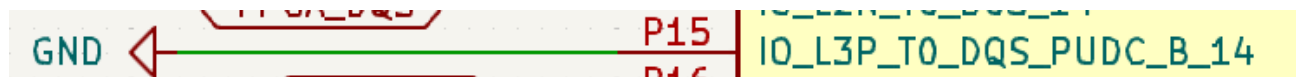
- pin **DONE**: es un de salida, que indica que el bitstream de la FPGA ha sido grabado. Se puede conectar a un led.
- pin **PROGRAM_B**: es un pin de entrada que permite resetear el bitstream interno de la FPGA. Se puede conectar a un botón.
- pin **INIT_B**: este es un pin de inicialización de la FPGA. Este pin se suele poner a pull-up con una resistencia $\leq 4,7k\Omega$ a la alimentación del banco 0. En caso de no utilizarse se deja en pull-up.



Pin PUDC_B

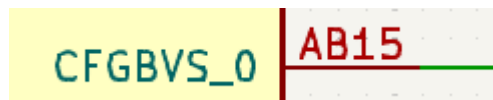
Este pin es único, y solo tiene la tarea de configurar los pines de la FPGA como pull-up o pull-down mientras se graba la FPGA. Este pin es también programable, por lo que mientras que se graba la FPGA tiene una función específica, y cuando termina pasa a ser un pin configurable por el usuario.

Este pin se suele dejar fijo a un valor, por lo que no se suele utilizar.



Pin CFGBVS

Este pin es el selector de tensión del banco 0 y de los pines de configuración. Este pin tiene que estar a nivel alto si la tensión de alimentación del banco 0 es 2,5V o 3,3V. Si la tensión del banco 0 es de 1,8V o menor, este pin tiene que estar a masas (GND).



En la siguiente tabla se puede ver a que valor de tensión se tiene que poner el pin CFGBVS. (*Vcco_0*: es la tensión del banco 0; *Vcco_14*: es la tensión del banco 14; *Vcco_15*: es la tensión del banco 15).

Configuration Mode	Banks Used	Configuration Interface I/O Voltage	HR Bank 0 V _{CC0_0}	HR Bank 14 V _{CC0_14}	HR Bank 15 V _{CC0_15}	CFGBVS
JTAG (only)	0	3.3V	3.3V	Any	Any	VCCO_0
		2.5V	2.5V	Any	Any	VCCO_0
		1.8V	1.8V	Any	Any	GND
		1.5V	1.5V	Any	Any	GND
Serial, SPI, or SelectMAP	0, 14 ⁽¹⁾	3.3V	3.3V	3.3V	Any	VCCO_0
		2.5V	2.5V	2.5V	Any	VCCO_0
		1.8V	1.8V	1.8V	Any	GND
		1.5V	1.5V	1.5V	Any	GND
BPI	0, 14, 15	3.3V	3.3V	3.3V	3.3V	VCCO_0
		2.5V	2.5V	2.5V	2.5V	VCCO_0
		1.8V	1.8V	1.8V	1.8V	GND
		1.5V	1.5V	1.5V	1.5V	GND

Otros pines

Otros pines que faltaría de comentar son los siguientes, cabe recordar que muchos de estos pines son pines que tienen doble funcionalidad, una para el arranque y otra para el usuario, en caso de no utilizarse se pueden dejar como pines de usuario.

- **EMCCLK**: este pin es el esclavo de reloj, este es solo necesario cuando la FPGA sea esclava de la memoria Flash, cosa que no suele ser lo frecuente.
- **CS1_B**: este pin es el pin CS de entrada de una memoria Flash, este pin se utiliza en la configuración SelectMAP, este pin solo sería necesario si la FPGA es esclava de una memoria Flash.

- **RDWR_B**: este pin es un pin de entrada de la configuración SelectMAP, este pin selecciona la dirección del bus.
- **DOUT/DOUT_CSO_B**: este pin es solo para la configuración por Daisy Chain (*FPGAs comunicadas en cadena*).
- **Ax**: estos pines (son hasta 29) reflejan el bus de direcciones de los modos de configuración BPI y SelectMAP.
- **FOE_B**: este es el pin de habilitación de la salida de la flash para el modo de configuración BPI.
- **FWE_B**: este es el pin de habilitación de la escritura de la flash para el modo de configuración BPI.
- **ADV_B**: este es el pin de salida para direcciones válidas para el modo BPI.
- **RS0, RS1**: este es el pin de revisión de los pines de la Flash. No se aplica en el modo de configuración Master SPI.
- **VCCBATT**: este pin sirve para almacenar la clave de cifrado AES en la memoria volátil. En caso de que el bitstream esté cifrado, este pin tiene que estar conectado a un sistema de batería para conservar la clave.
- **POR_OVERRIDE** (solo para UltraScale): este pin es el Power On Reset Override, que lo que hace es seleccionar el retardo previo a que entre la configuración en funcionamiento. Este pin se puede dejar a masa.

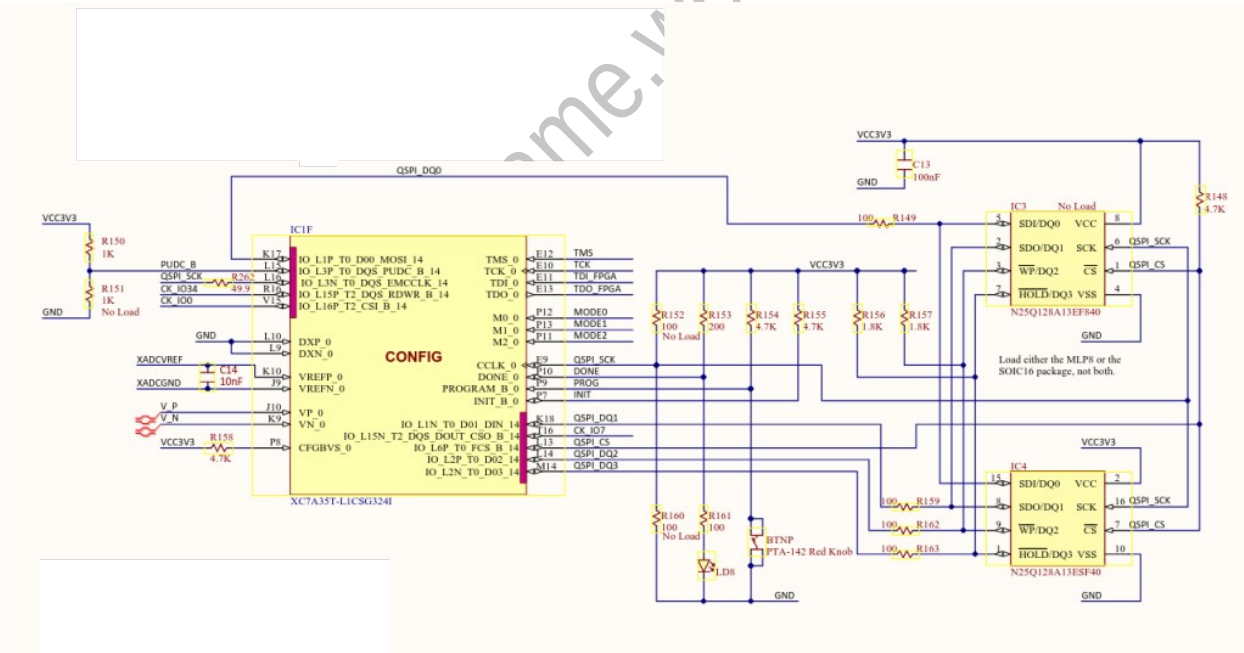
Para facilitar la localización de los distintos pines aquí dejo una tabla para saber dónde localizarlos.

Pin	Banco (serie 7)	Banco (UltraScale)
CFGBVS	0	0
M[2:0]	0	0
TCK	0	0
TMS	0	0
TDI	0	0
TDO	0	0
PROGRAM_B	0	0
INIT_B	0	0
DONE	0	0
CCLK	0	0
PUDC_B	14	0
EMCCLK	14	65
CSI_B	14	65
CSO_B	14	65
DOUT	14	65

RDWR_B	14	0
D00_MOSI	14	0
D01_DIN	14	0
D[00-31]	14	65
A[00-28]	14	65
FCS_B	14	65
FOE_B	15	65
FWE_B	15	65
ADV_B	15	65
RS0, RS1	15	65
VCCBATT/VBAT	N/A	N/A
POR_OVERRIDE		N/A

NOTA FINAL

Aquí deajo un pequeño ejemplo real de una Arty A100 para que se pueda ver como se podrían conectar dos memorias Flash a una FPGA para el arranque.



Y en caso de que querer conectar una UltraScale con dos memorias Xilinx da este ejemplo en la documentación.

