



## About this document

#### Scope and purpose

This application note discusses how to connect Infineon radiation-tolerant SPI flash devices to the UltraScale family of devices from Xilinx. It describes how to use various FPGA configuration options when using Infineon radiation-tolerant SPI NOR flash devices.

FPGAs for aerospace and defense applications from Xilinx are used for basic logic functions, chip-to-chip connectivity, signal processing, and embedded processing. These devices are programmed and configured using an array of SRAM cells that need to be programmed on every power-up by means such as a microprocessor, JTAG port, or directly through a serial PROM or flash.

UltraScale and radiation-tolerant Kintex UltraScale FPGAs from Xilinx support Single/Quad/Octal SPI configuration mode. Infineon serial peripheral interface (SPI) NOR flash is radiation-tolerant and can operate in aerospace and defense applications, thus can be used as a configuration solution for these FPGAs at power-up.

#### **Intended audience**

Hardware and platform developers using Infineon's radiation-tolerant NOR flash devices in systems based on devices from Xilinx.

#### Associated part family

CYRS16B256/CYRS16B512

## **Table of contents**

| Abou  | It this document                               | 1  |
|-------|--|----|
| Table | e of contents                                  | 1  |
| 1     | Introduction                                   |    |
| 2     | SPI connection basics                          |    |
| 3     | SPI NOR flash connections                      | 6  |
| 4     | Programming options                            |    |
| 4.1   | Programming based on the toolchain from Xilinx |    |
| 4.1.1 | Bit file generation                            |    |
| 4.1.2 | Programming the flash device                   |    |
| 4.2   | Direct programming the SPI NOR flash           | 14 |
| 4.3   | Standalone SPI NOR flash programmer            |    |
| 5     | Other considerations                           | 19 |
| 5.1   | Applying voltages at power-on                  |    |
| 6     | Summary  | 21 |
| Refer | rences   |    |

| Radiation-tolerant SPI NOR flash configuration options for |  |
|--|--|
| JltraScale FPGAs from Xilinx                               |  |
|  |  |



**Table of contents** 



Introduction

## 1 Introduction

Infineon's SPI flash programming solutions using the CYRS16Bxxx family of devices are now available for use with aerospace- and defense-rated FPGAs from Xilinx. These radiation-tolerant SPI flash devices can be used in single SPI (x1), quad SPI (x4), or dual quad SPI (2x4) configuration options. These devices are available in 256-Mbit and 512-Mbit densities. These radiation-tolerant SPI NOR flash devices support the same commands and frequencies of the commercial-grade Infineon SPI NOR flash devices. Currently, there are two Infineon hardware solutions to connect and program SPI NOR flash devices:

- Infineon SPI NOR flash configuration solution development kit (CYDK-NOR) is a mezzanine card which is compatible with the alpha data development platform for space-grade FPGAs (ADA-SDEV-KIT2). This mezzanine card is set up to use the programming approach based on the toolchain from Xilinx.
- Standalone SPI NOR flash programming daughter card solution that uses a standalone NOR flash programmer from EmbeddedComputers: (https://www.embeddedcomputers.net/products/FlashcatUSB\_Mach1/)



SPI connection basics

## 2 SPI connection basics

Serial peripheral interface (SPI) is a simple 4-wire synchronous interface protocol which enables a master device and one or more slave devices to intercommunicate. The SPI bus consists of four signal wires:

- The master out slave in (MOSI) signal generated by the master (data out to the slave)
- The master in slave out (MISO) signal generated by the slave (data in to the master)
- The serial clock (SCK) signal generated by the master to synchronize data transfers
- The slave select (SS#) signal generated by the master to select individual slave devices; also known as chip select (CS#) or chip enable (CE#)

The quad serial peripheral interface (QSPI) expands the data bus to four bits and adds QSPI commands. Two QSPI devices can be used in parallel to provide an 8-bit data bus for the SPI dual quad SPI (DQSPI) configuration interface; an additional slave select is also used for this connection.

Following the SPI/QSPI/DQSPI protocol, the master is assigned to the FPGA device and the slave to the SPI NOR flash device, as shown in **Figure 1**. Per these connections, the SPI NOR flash is available to configure the FPGA at power-up.

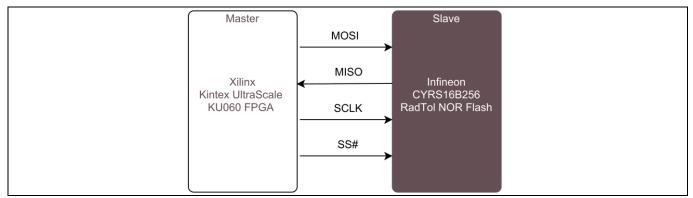


Figure 1 Direct configuring FPGA to interface with SPI NOR flash

Figure 2 shows the SPI NOR flash device connection when used in Quad SPI connection mode.

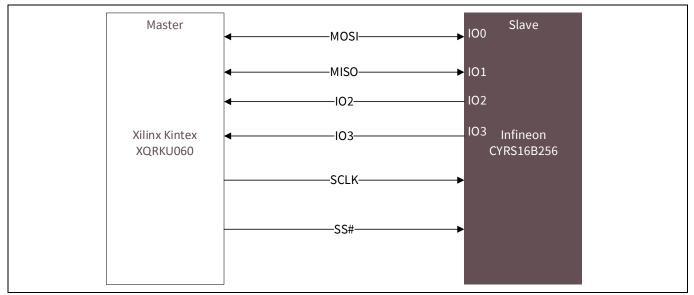
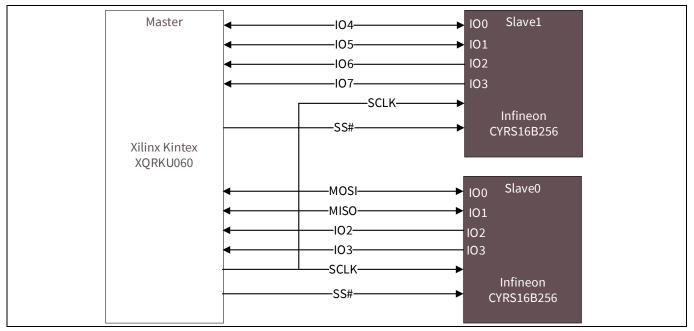


Figure 2 QSPI configuration connection with SPI NOR flash

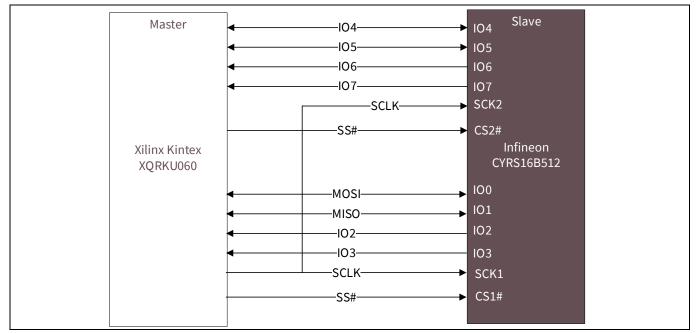


#### SPI connection basics

**Figure 3** shows the dual quad SPI device connection using two quad SPI NOR flash devices in parallel. In the case of 512-Mb devices, two separate quad SPI busses are available on the same device as shown in **Figure 4**.









DQSPI configuration connection with single 512-Mb SPI NOR flash



SPI NOR flash connections

## **3** SPI NOR flash connections

A connection table for SPI NOR flash SPI pins to FPGA configuration pins is shown below. Some FPGA pins will not have a corresponding SPI NOR flash pin, but are essential in configuring the FPGA to the correct mode and I/O voltage level. The CYRS16B256 and CYRS16B512 devices only support 3.3V I/O CMOS voltage levels.

The pin connection table is based on the Infineon CYRS16B256/CYRS16B512 parts.

| Table 1            | Pin connec         | tions              |                  |  |
|--------------------|--------------------|--------------------|------------------|--|
| SPI x1 pin<br>name | SPI x4 pin<br>name | SPI x8 pin<br>name | FPGA pin<br>name | Description  |
| SI/IO0             | SI/IO0             | SI/IO0             | D00_MOSI         | Master out slave in, when in 1-bit SPI mode. D[0] of D[3:0] when in quad SPI mode.   |
| SO/I01             | SO/IO1             | SO/IO1             | D01_DIN          | Slave out master in when in 1-bit SPI mode. D[1] of D[3:0] when in quad SPI mode. D[1] of D[7:0] when in dual quad SPI mode.   |
| NA                 | 102                | 103                | D02              | D[2] of D[3:0] when in quad SPI mode. D[2] of D[7:0] when in dual quad SPI mode.   |
| NA                 | 103                | 104                | D03              | D[3] of D[3:0] when in quad SPI mode. D[3] of D[7:0] when in dual quad SPI mode.   |
| NA                 | NA                 | 104                | D04              | D[4] of D[7:0] when in dual quad SPI mode  |
| NA                 | NA                 | 105                | D05              | D[5] of D[7:0] when in dual quad SPI mode  |
| NA                 | NA                 | 106                | D06              | D[6] of D[7:0] when in dual quad SPI mode  |
| NA                 | NA                 | 107                | D07              | D[7] of D[7:0] when in dual quad SPI mode  |
| CS#                | CS#                | CS1#               | FCS_B            | Active LOW chip select for D[3:0], connect with a pullup to VCCO_0 $\leq$ 4.7 k $\Omega$   |
| NA                 | NA                 | CS2#               | FCS2_B           | Active LOW chip select for D[7:4], connect with a pullup to VCCO_0 $\leq$ 4.7 k $\Omega$   |
| SCK                | SCK                | SCK1#              | CCLK             | FPGA-sourced configuration clock   |
| NA                 | NA                 | SCK2#              | -                | In DQSPI mode, connect this clock pin on the SPI flash to CCLK.  |
| -                  | -                  | -                  | CFGBVS           | This pin's value determines the I/O operating voltage<br>range of Bank0. Tie this pin to HIGH or VCCO_0. The<br>VCCO_0 voltage will be 3.3 V because Infineon SPI<br>flash devices support only 3.3 V I/O.   |
| -                  | -                  | -                  | M[2:0]           | Mode configuration bits set to M[2:0] =001, master SPI   |
| -                  | -                  | -                  | PROGRAM_B        | Active LOW input. When pulsed LOW, the FPGA configuration is cleared, and new configuration is initiated.  |
| -                  | _                  | -                  | INIT_B           | Active LOW FPGA initialization pin. After initialization<br>is completed, the FPGA device will tristate this pin<br>and begin the FPGA configuration. External masters<br>can force this pin LOW to delay the configuration by<br>driving it LOW actively. This pin is connected with a<br>pull-up to VCCO_0 $\leq$ 4.7 k $\Omega$ . |



## SPI NOR flash connections

| SPI x1 pin | SPI x4 pin | SPI x8 pin | FPGA pin | Description  |
|------------|------------|------------|----------|--|
| name       | name       | name       | name     |  |
| -          | -          | -          | DONE     | When DONE is HIGH, it indicates that the FPGA programming was completed. This pin is connected |
|            |            |            |          | with a pull-up to VCCO_0 $\leq$ 4.7 k $\Omega$ .   |



**Programming options** 

## 4 **Programming options**

There are three ways to program the SPI NOR flash; they are discussed in detail in the following sections:

- **Programming based on the toolchain from Xilinx:** This is achieved by communicating to the FPGA through the JTAG interface and programming the flash indirectly through the FPGA.
- Direct programming the SPI NOR flash: An additional master is connected to the SPI bus.
- **Standalone SPI NOR flash programmer:** The SPI NOR flash is programmed using a third-party standalone NOR flash programmer.

## 4.1 Programming based on the toolchain from Xilinx

Vivado IDE from Xilinx can be used to generate a bit file to program through the JTAG interface of the FPGA using the programming cable from Xilinx. The Vivado (2019.2 and later versions) toolchain from Xilinx supports the generation of the bit file and programming in this mode for the Kintex XCKU060 FPGA from Xilinx. In this mode, the tools will program the bit file through the configuration cable from Xilinx connected to the JTAG interface of the FPGA; the FPGA SPI configuration interface is then used to program the incoming bit file on the JTAG interface to the SPI NOR flash.

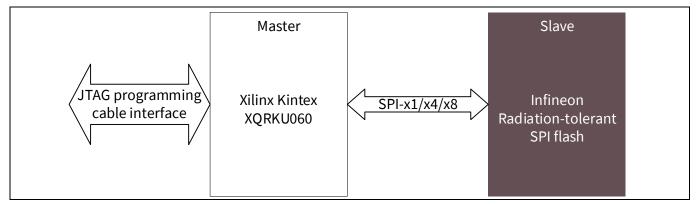


Figure 5 Programming based on the toolchain from Xilinx

## 4.1.1 Bit file generation

In the Vivado IDE from Xilinx, after the implementation has been completed, before generating the bit file, choose the SPI configuration method (i.e., SPI, QSP, DQSPI, master serial mode, etc.).

- 1. Open the implemented design in the Vivado IDE from Xilinx.
- 2. Right-click on the Generate bitstream option and select Bitstream Settings.



#### Programming options

| ile <u>E</u> dit Flow <u>T</u> ools Rep     | orts Window Layout View Help Q. Quick Acces     | 5                                     |  | write_bitstream Complete                          |
|---|---|---------------------------------------|--|---|
| 9 K A 🖪 🐘 🗙 🕨                               | μ 🕸 Σ 🗶 🖉 🕺                                     |                                       |  | 😀 Default Layout 🔷                                |
| w Navigator 😤 🗢 ? _<br>Kun synulesis        | PROJECT MANAGER - qdr2plus v7k7                 |                                       |  | ?   |
| > Open Synthesized Design                   | Sources   | ? _ 🗆 🖆 ×                             | Project Summary                                    | ? 🗆 🗅 ×   |
|   | Q X + 0   | ٥                                     | Overview   Dashboard                               |   |
| IMPLEMENTATION                              | D. M. role2alue x7k2 recetratote.vdc/taraat)    | <b>^</b>                              |  |   |
| Run Implementation                          | Hierarchy Libraries Compile Order               |                                       | Settings Edit                                      |   |
| <ul> <li>Open Implemented Design</li> </ul> |   |                                       | Project name: qdr2plus                             | v7k7  |
| Constraints Wizard                          | Properties                                      | ? _ 🗆 🗆 ×                             |  | DRIIPLUS/xilinx/kintex7-ULTRASCALE/QDRIIPLUS_ECC6 |
| Edit Timing Constraints                     | Q1_qdr2plus_x7k7_constraints.xdc                | $\leftarrow   \Rightarrow   \diamond$ | Product family: Kintex Ult                         |   |
| ů.  | <   | > ^                                   | Project part: xcku060-1<br>Top module name: QDR_II | ffva1517-1-i                                      |
| Report Timing Summary                       | General Properties                              |                                       | <  |   |
| Report Clock Networks                       |   |                                       |  |   |
| Report Clock Interaction                    | Tcl Console Messages Log Reports × Design       | Runs                                  |  | ? _ 🗆 🗅   |
| Report Methodology                          | Q   ≚   ♦   +   −   ∅   ▶                       |                                       |  |   |
| Report DRC                                  | Repart  | Type                                  | Options  | Ma  |
| Report Utilization                          | ✓ Synthesis                                     |                                       |  |   |
|   | <ul> <li>Synth Design (synth_design)</li> </ul> |                                       |  |   |
| 🗯 Report Power                              | synth_1_synth_report_utilization_0              | report_utilization                    |  | 5/1   |
| 🕄 Schematic                                 | synth_1_synth_synthesis_report_0                |                                       |  | 5/1   |
|   | Implementation                                  |                                       |  |   |
| PROGRAM AND DEBUG                           | ✓ impl_1  |                                       |  |   |
| Generate Bitstream                          | Design Initialization (init_design)             |                                       |  |   |

#### Figure 6 Bitstream configuration settings selection in Vivado IDE from Xilinx

3. In the pop-up window, select the *.bin* file options and click the **Configure additional bitstream settings** link.



#### Programming options

| Ĵ.                         | Bitstream   |    |
|----------------------------|---|----|
| Project Settings           | Specify various settings related to writing Bitstream | P~ |
| General                    |   |    |
| Simulation                 | ① Configure additional bitstream settings.            |    |
| Elaboration                | ✓ Write Bitstream (write_bitstream)                   |    |
| Synthesis                  | tcl.pre   |    |
| Implementation             | td.post   |    |
| Bitstream                  | -raw_bitfile  |    |
| > IP                       | -mask_file  |    |
| Tool Settings              | -no_binary_bitfile                                    |    |
| Project                    | -bin_file*  |    |
| IP Defaults                | -readback_file  |    |
| Board Repository           | -logic_location_file                                  |    |
| Example Project Repository | -verbose  |    |
| Source File                | More Options  |    |
| Display                    |   |    |
| WebTalk                    |   |    |
| Help                       |   |    |
| > Text Editor              | Select an option above to see a description of it     |    |
| 3rd Party Simulators       |   |    |
| > Colors                   |   |    |
| Selection Rules            |   |    |
| Shortcuts                  |   |    |
| > Strategies               |   | >  |

Figure 7 Configure additional bitstream settings in Vivado IDE from Xilinx

4. In the next widow, select the **Configuration** option on the left pane. All configuration options should now be seen on the right pane of the window.



**Programming options** 

|                                | Configuration            |                 |                     |         |     |     |   |
|--------------------------------|--------------------------|-----------------|---------------------|---------|-----|-----|---|
| General                        |                          |                 |                     |         |     |     | ^ |
| Configuration                  | Configuration Setup      |                 |                     |         |     |     |   |
| Configuration Modes<br>Startup | Tristate CCLK pin during | g reconfigurati | on                  | FALSE   | ♥ C |     |   |
| Encryption                     | Configuration Rate (MF   | Hz)             |                     | 33      | ~ C |     |   |
| Readback                       | Enable external configu  | ration clock ar | nd set divide value | DISABLE | ♥ C |     |   |
| Authentication                 | Configuration Voltage    |                 |                     | 3.3     | ~ C |     |   |
|                                | Configuration Bank Vol   | Itage Selection |                     | VCCO    | ~ C |     |   |
|                                | BPI Configuration        |                 |                     |         |     |     |   |
|                                | 1st Read cycle 1         | ۲ ×             | C                   |         |     |     |   |
|                                | Page Size (bytes)        | I ~             | C                   |         |     |     |   |
|                                | Synchronous Mode         | DISABLE 🗸       | C                   |         |     |     |   |
|                                | SPI Configuration        |                 |                     |         |     |     |   |
|                                | Enable SPI 32-bit addre  | ess style       |                     | YES     | 5   | ~ C |   |
|                                |                          |                 |                     |         |     |     |   |

#### Figure 8 Configuration of clock and voltage for configuration memory in the bitstream in Vivado IDE from Xilinx

- 5. In the configuration setup, change the following for default values:
  - Configuration Rate (MHz): 33 MHz (The clock frequency will vary depending on the board design and other factors that may limit the maximum frequency of operation for the configuration interface.)
  - Configuration Voltage: 3.3 V
  - Configuration bank voltage Selection: VCCO



**Programming options** 

| Q.                                       | Configuration   |               |  |  |  |  |  |
|--|---|---------------|--|--|--|--|--|
| General<br>Configuration                 | Synchronous Mode DISABLE V C  | ^             |  |  |  |  |  |
| Configuration Modes<br>Startup           | SPI Configuration   |               |  |  |  |  |  |
| Encryption<br>Readback<br>Authentication | Enable SPI 32-bit address styleYESCBus width4CEnable the FPGA to use a falling edge clock for SPI data captureYESC                                |               |  |  |  |  |  |
|  | MultiBoot Settings  |               |  |  |  |  |  |
|  | Load a fallback bitstream when a configuration attempt fails  | ENA           |  |  |  |  |  |
|  | Starting address for the next configuration in a MultiBoot setup  | 0X000         |  |  |  |  |  |
|  | Enable the IPROG command in Bitstream   |               |  |  |  |  |  |
|  | Specifiy the internal value of the RS[1:0] settings in the Warm Boot Start Address  | 00            |  |  |  |  |  |
|  | Enable whether the RS[1:0] tristate is enabled by setting the option in the Warm Boot Start Address<br>Watchdog Timer value in Configuration mode | DISA<br>0X000 |  |  |  |  |  |
|  |   |               |  |  |  |  |  |

Figure 9 Configuration memory SPI interface type for bitstream transmission in Vivado IDE from Xilinx

- 6. In the SPI configuration section, change the following:
  - Enable SPI 32-bit address style: YES
  - Bus width: 4 (Set for QSPI; if DQSPI, set to 8.)
  - Enable the FPGA to use a falling edge clock for the SPI data capture: YES
- 7. Select **Configuration modes** in the left panel of the window and select the corresponding SPI mode (in this case, Master SPI x4).
- 8. Click **OK** to accept the settings and then click **OK** to close the setting window.



#### **Programming options**

| Edit Device Properties<br>e this dialog to edit the pro | gramming and configuration properties for your current design; default values are set automatically. $\redset$  |
|---|---|
| Q,-   | Configuration Modes   |
| General<br>Configuration<br>Configuration Modes         | Select up to two device configuration modes. JTAG is always selected. Selecting a check box assigns a mode and clicking an image displays details.  |
| Startup   | $\Box$ Prohibit usage of the configuration pins as user I/O and persist after configuration $\mathbb C$   |
| Encryption  | Reset configuration mode ${f C}$  |
| Readback  |   |
| Authentication  | <complex-block>Image: state state state state</complex-block>   |
|   | V023F<br>Te to todary unply ofter method<br>all works if the VCOULD = V000<br>all works if the VCOULD = V000<br>we V6VT method<br>We VCOULD = V000, if VCOU, |
| <u>H</u> elp  | Reset All OK Cancel   |

Figure 10 SPI configuration mode selection for the FPGA in Vivado IDE from Xilinx

9. Generate the bitstream. The *.bit* and *.bin* files will be generated with the QSPI programming options.

10. Follow the same steps to generate the DQSPI programming file.

## 4.1.2 Programming the flash device

- 1. Generate the bit file using the flow in the Vivado IDE flow from Xilinx. Note that version 2019.2 and later support the Rad-Hard SPI NOR flash natively.
- 2. Open the Hardware Manager in the **Program and Debug** section of the IDE.
- 3. Connect to the hardware and set up the FPGA device from Xilinx. Select the XCKU060 Kintex radiationtolerant device from Xilinx.
- 4. Select **Spansion S25fl256l-spi-x1\_x4\_x8** to add the appropriate configuration memory.



#### **Programming options**

| Choose a configuration memory part.   Device: <ul> <li>xcku060_0</li> </ul> Manufacturer All   Manufacturer All   Density (Mb) 256     Beset All Filters     Elect Configuration Memory Part     Search: <ul> <li>c</li> </ul> Name   Part Manufact   Alias   Family Type   Density (   \$25fl256l-spi-x1_x2_x4_x8   \$25fl256scocccc0   \$pansion   \$25fl256scocccc0   \$pansion   \$25fl256scocccc0   \$pansion   \$25fl256scocccc0   \$pansion   \$25fl256scocccc0   \$pansion   \$25fl256scoccccc0   \$pansion   \$25fl256scocccc0   | Add Configurati       | on Memory Device     |                 |                   |         |       |     |           |      | >         |
|---|-----------------------|----------------------|-----------------|-------------------|---------|-------|-----|-----------|------|-----------|
| Manufacturer       All       Type       All       Image: Constitution of the second secon | Choose a co           | nfiguration memory p | part.           |                   |         |       |     |           |      |           |
| Manufacturer All   Density (Mb) 256     Reset All Filters     Reset All Filters     elect Configuration Memory Part     Search:     Q-     Name   Part   Manufact   Alias   Family   Type   Density (   \$25fl256l-spi-x1_x2_x4_x8   \$25fl256l-spi-x1_x2_x4   \$25fl256sxxxxxx0   Spansion     \$25fl256sxxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxx0     \$25fl256sxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx   | Device: 👜 xcku06      | 50_0                 |                 |                   |         |       |     |           |      |           |
| Density (Mb) 256   Width All   Reset All Filters   elect Configuration Memory Part   Search: Q:   Name   Part Manufact   Alias   Family Type   Density (   \$25fl256l-spi-x1_x2_x4_x8   \$25fl256l-spi-x1_x2_x4   \$25fl256sxxxxx0   \$pi   256 \$25fl256sxxxxx0  | lter                  |                      |                 |                   |         |       |     |           |      |           |
| Reset All Filters         elect Configuration Memory Part         Search:       Q:         Name       Part       Manufact       Alias         Family       Type       Density (.         \$ s25fl256l-spi-x1_x2_x4_x8       s25fl256l       Spansion         \$ s25fl256sxxxxx0-spi-x1_x2_x4       s25fl256sxxxxx0       Spansion   | <u>M</u> anufacturer  | All                  | ~               |                   |         | Type  | All |           |      | ~         |
| elect Configuration Memory Part          Search:       Q-         Name       Part       Manufact       Alias       Family       Type       Density (.         Image: s25fl256l-spi-x1_x2_x4_x8       s25fl256l       Spansion       s25floxxl       spi       256         Image: s25fl256sxxxxxx0-spi-x1_x2_x4       s25fl256sxxxxxx0       Spansion       s25floxxs       spi       256  | Density ( <u>M</u> b) | 256                  | ~               |                   | V       | Vidth | All |           |      | ~         |
| Search:       Q.         Name       Part       Manufact       Alias       Family       Type       Density (.         Image: s25fl256l-spi-x1_x2_x4_x8       s25fl256l       Spansion       s25flxxxl       spi       256         Image: s25fl256sxxxxx0-spi-x1_x2_x4       s25fl256sxxxxx0       Spansion       s25flxxxl       spi       256   |                       |                      |                 | <u>R</u> eset All | Filters |       |     |           |      |           |
| Name       Part       Manufact       Alias       Family       Type       Density (.   | elect Configuration   | n Memory Part        |                 |                   |         |       |     |           |      |           |
|   | Search: Q-            |                      |                 |                   |         |       |     |           |      |           |
| \$\$25fl256l-spi-x1_x2_x4_x8       \$\$25fl256l       \$\$pansion       \$\$25fl256l       \$\$25fl256l <t< td=""><td>Name</td><td></td><td>Part</td><td>Manufact</td><td>Alias</td><td></td><td></td><td>Family</td><td>Туре</td><td>Density (</td></t<>   | Name                  |                      | Part            | Manufact          | Alias   |       |     | Family    | Туре | Density ( |
|   | 🞐 s25fl256l-spi       | -x1_x2_x4_x8         | s25fl256l       | Spansion          |         |       |     | s25flxxxl | spi  | 256       |
|   | 😻 s25fl256sxxx        | xxx0-spi-x1_x2_x4    | s25fl256sxxxxx0 | Spansion          |         |       |     | s25flxxxs | spi  | ¥         |
| OK     Cancel   | <                     |                      |                 |                   |         |       |     | _         |      | >         |
|   | (?)                   |                      |                 |                   |         |       |     | OI        | ۲    | Cancel    |

Figure 11 Infineon radiation-tolerant SPI NOR flash selection in Vivado IDE from Xilinx

5. Follow the instructions in the IDE to select the *.bin* file and program the SPI NOR flash device.

## 4.2 Direct programming the SPI NOR flash

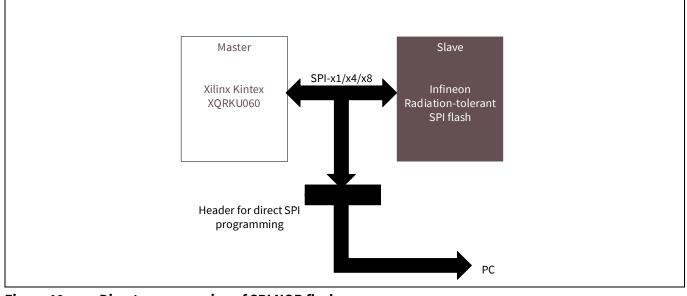


Figure 12 Direct programming of SPI NOR flash

In this mode, the SPI NOR flash can be programmed directly by another SPI master such as a microcontroller or PC.



#### Programming options

The FPGA supplies the CCLK output from its internal oscillator to drive the clock input of the SPI NOR flash. The in-system programming control (ISP) signal is used to tristate the FPGA SPI interface signals during in-system programming to re-program the configuration data in the SPI NOR flash.

To read the configuration data from the SPI NOR flash at power-up, the FPGA from Xilinx must issue a read command to the SPI NOR flash. This is done automatically based on the MODE setting. The following figure shows the connection for this programming option.

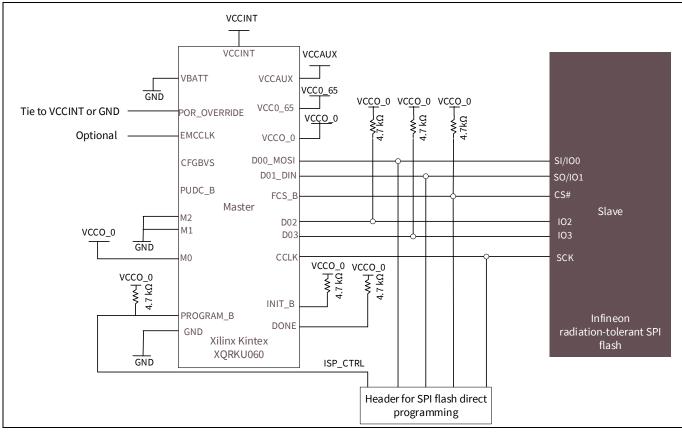


Figure 13 Connections for SPI direct programming

## 4.3 Standalone SPI NOR flash programmer

A standalone SPI NOR flash programmer can also be used to program the SPI NOR flash device. Generic off-theshelf programmers are widely available such as the RAD-HARD SPI NOR flash device socket board from EmbeddedComputers that will plug into their FlashcatUSB-Mach programmer (https://www.embeddedcomputers.net/products/FlashcatUSB\_Mach1/).

Using the FlashcatUSB software (https://www.embeddedcomputers.net/software/), the SPI NOR flash device can be programmed from a computer with a USB port. Install the USB driver and run the software as instructed in the user manual. When the FlashcatUSB-Mach programmer from EmbeddedComputers is connected to the USB port, it will detect the flash device in the socket. With the flash device detected, the software will create a Flash tab. All flash read and write operations can be done through this tab.

The following image shows the FlashcatUSB-MACH programmer board and socket board from EmbeddedComputers:



Programming options

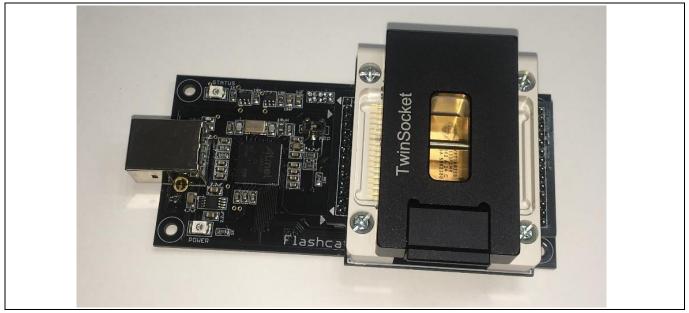


Figure 14 FlashcatUSB-Mach programmer + socket board from EmbeddedComputers

In the Vivado IDE from Xilinx, generate a *.bin* file that can be used by the FlashcatUSB software from EmbeddedComputers. Do the following to program the SPI NOR flash:

- 1. Run the *FlashcatUSB.exe* program.
- 2. Connect the FlashcatUSB programmer from EmbeddedComputers with the SPI flash plugged into the programmer board.

The **Status** tab shows the software connected to the FlashcatUSB programmer from EmbeddedComputers.

3. Click on the **SPI Flash** tab. Ensure that the following is displayed:

| FlashcatUSB (Build 592) - Personal Use Only         | - 🗆 X                                 |
|---|---------------------------------------|
| Main Mode Script Tools Language                     |                                       |
| Status Console SPI Flash                            |                                       |
| Cypress S25FL256L                                   |                                       |
| 1 🔣 🚬 🛱 🖉   | 0×0                                   |
| Write data to memory                                |                                       |
| 00000000000000000000000000000000000000              | ••••••••••••••••••                    |
| 00000010: FF    | =                                     |
| 00000020: FF    |                                       |
| 00000030: FF    |                                       |
| 00000040: FF    |                                       |
| 00000060: FF    |                                       |
| 00000070: FF    |                                       |
| 00000080: 00 00 00 00 00 00 00 00 00 00 00 00 0     |                                       |
| 00000090: 00 00 00 00 00 00 00 00 00 00 00 00 0     |                                       |
| 000000A0: 00 00 00 00 00 00 00 00 00 00 00 00 0     |                                       |
| 000000B0: 00 00 00 00 00 00 00 00 00 00 00 00 0     |                                       |
| 000000C0: 00 00 00 00 00 00 00 00 00 00 00 00 0     |                                       |
| 00000000: 00 00 00 00 00 00 00 00 00 00             |                                       |
| 000000E0: 00 00 00 00 00 00 00 00 00 00 00 00 0     |                                       |
| 000000F0: 00 00 00 00 00 00 00 00 00 00 00 00 0     | · · · · · · · · · · · · · · · · · · · |
|   |                                       |
|   |                                       |
| User canceled. No data written to Cypress S25FL256L |                                       |

Figure 15 Selection of the memory contents window for SPI flash in FlashcatUSB IDE from EmbeddedComputers



Programming options

- 4. Click Write Data to memory.
- 5. On the window, select the *.bin* file.

| Choose file to write interest | o Cypress S25FL256L         |                       | ×              |
|-------------------------------|-----------------------------|-----------------------|----------------|
|                               | soft > FCUSB_BUILD592 > v さ | Search FCUSB_BUILD592 | 0220 <b>,0</b> |
| Organize 🔻 New fo             | lder                        |                       | •              |
| This PC                       | Name                        | Date modified         | Туре           |
| 3D Objects                    | Driver                      | 1/27/2020 2:20 PM     | File folder    |
| Desktop                       |                             | 1/19/2020 12:39 PM    | File folder    |
| Documents                     | JTAG-SVF                    | 2/17/2019 5:16 PM     | File folder    |
| Downloads                     | Manual                      | 2/10/2020 2:22 AM     | File folder    |
| incoming on ftp               | Scripts                     | 1/27/2020 2:28 PM     | File folder    |
| incoming on ftp               |                             |                       |                |
| Music                         |                             |                       |                |
| Pictures                      |                             |                       |                |
| Videos                        |                             |                       |                |
| Windows (C:)                  |                             |                       |                |
| 🚔 SDHC (D:)                   | , «                         |                       | 2              |
| File                          | name: 🗸 🗸                   | Binary Files (*.bin)  | ~              |
|                               |                             | Open (                | Cancel         |

Figure 16 Procedure to select the bin file for the SPI flash device

6. Verify that the size displayed in the pop-up window is correct, and then click **OK**.

| FlashcatUSB (Build 592) - Personal Use Only  | – 🗆 X                                 |  |
|--|---------------------------------------|--|
| Main Mode Script Tools Language  |                                       |  |
| Status Console SPI Flash   |                                       |  |
| Cypress S25FL256L  |                                       |  |
|  | 0x0                                   |  |
| 000000000: FF  | · · · · · · · · · · · · · · · · · · · |  |
| 00000020: FF FF FF FF FF FF FF Base address Length<br>00000030: FF FF FF FF FF FF FF FF 00x0 24124908<br>000000040: FF |                                       |  |
| 00000050: FF FF FF FF FF FF FF FF Cancel OK  |                                       |  |
|  |                                       |  |
|  |                                       |  |
|  |                                       |  |
| 000000E0: 00 00 00 00 00 00 00 00 00 00 00 00 0  | ····· ·                               |  |
| Select range to write data to Cupress \$25E1 2561  |                                       |  |
| Select range to write data to Cypress S25FL256L  |                                       |  |





#### **Programming options**

7. The software begins programming the SPI flash. Wait for the "Write operation complete" status on the bottom of the window.

| Main       Mode       Script       Tools       Language         Status       Console       SPI Flash         Cypress       S25FL256L       Image: S25FL256L         Image:  |     |
|---|-----|
| Cypress S25FL256L   |     |
| Image: |     |
| 00000000: FF  |     |
| 00000010: FF  | -   |
| 00000020: FF  |     |
| 00000030: FF  |     |
|   |     |
|   |     |
| 00000050: AA 99 55 66 20 00 00 00 30 03 E0 01 00 00 00 B Uf 0   |     |
| 00000060: 30 00 80 01 00 00 01 2 20 00 00 00 20 00 00 00 0  |     |
| 00000070: <b>30 02 20 01 00 00 00 30 02 00 01 00 00 00</b> 00   |     |
| 00000080: 30 00 80 01 00 00 00 00 20 00 00 30 00 80 01 0  |     |
| 00000090: 00 00 00 07 20 00 00 00 20 00 00 30 00 20 01 0.   |     |
| 000000000 00 00 00 00 00 00 00 00 00 00   |     |
| 00000080: 38 00 35 E5 30 01 C0 01 00 40 00 00 30 01 80 01 8.5.0@0.<br>000000C0: 03 91 90 93 30 00 80 01 00 00 00 92 00 00 000   |     |
|   |     |
| 000000E0: 30 00 C0 01 00 00 00 00 30 03 00 01 00 00 00 00 0   |     |
| 000000F0: 20 00 00 00 20 00 00 20 00 00 20 00 00  | · · |



SPI flash programming completed in FlashcatUSB IDE from EmbeddedComputers



Other considerations

## 5 Other considerations

Because the power-up timing and voltage thresholds are different for both FPGAs and SPI NOR flash devices, it is important to carefully review the differences to ensure compatibility between the devices on power-up.

## 5.1 Applying voltages at power-on

A race condition can occur between the SPI NOR flash and the FPGA devices from Xilinx at power-on. After completing its power-on reset sequence, the FPGA sends a read command to the SPI NOR flash to acquire the configuration data bit stream. If the SPI NOR flash has not yet completed its own POR sequence, it is not ready to respond to the FPGA read command, which is issued only once. Under this scenario, the FPGA is not configured.

The FPGA waits for its three power supplies (VCCINT, VCCAUX, and VCCO\_0) to reach their individual power-on thresholds before starting the configuration process. In many applications, VCCO\_0, which supplies +3.3 V to both the FPGA and SPI NOR flash, is valid before the FPGA's other two power supply inputs (VCCINT and VCCAUX) are valid; consequently, there may be no issue. However, because the SPI NOR flash minimum voltage threshold is much higher than the VCCO\_0 threshold, and the SPI NOR flash has an additional delay after it reaches its minimum voltage before SPI NOR flash is available for read operations, a careful analysis must be completed to ensure timing compatibility between the FPGA and SPI NOR flash.

After the three FPGA voltages reach their POR thresholds, the FPGA starts its configuration process:

- Clears its internal configuration
- De-asserts INIT\_B, and selects SPI NOR Flash
- Sends the appropriate read command to start configuration bitstream from SPI NOR Flash

After the Infineon SPI NOR flash device reaches its minimum voltage, a power-up time delay ( $t_{PU}$ ) must be added to the SPI NOR flash power-on before it is available for read operations. For the CYRS16B256 and CRYS16B512,  $t_{pu}$  is ~300 µs.

The power-on time difference is considerable between the FPGA and SPI NOR flash devices; although the FPGA power-up time is in the range of milliseconds compared to microseconds for the SPI NOR flash, this should still be considered because these devices could be connected to different power supplies. This necessitates a circuit solution to guarantee power-on compatibility between the FPGA and SPI NOR flash.

To protect against accessing the SPI NOR flash before it has completed t<sub>pu</sub>, use external control to hold the INIT\_B or PROG\_B pin LOW until the SPI NOR flash has powered up reliably and is ready to accept commands.

For this solution, use an open-drain or open-collector output when driving INIT\_B or PROG\_B pins. One example of external control is using a power monitor supervisor device as shown in **Figure 19**.



Other considerations

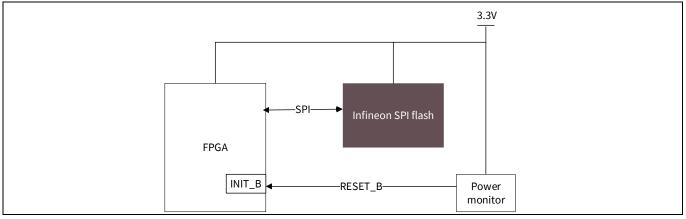


Figure 19 POR using power monitor supervisor

An example of the power monitor supervisor is ADM6384x27D2 from analog devices. The RESET# signal from this device is held LOW until its power supply voltage reaches  $2.7 V + 20 ms (t_{RP} time)$ , as shown in **Figure 20**.

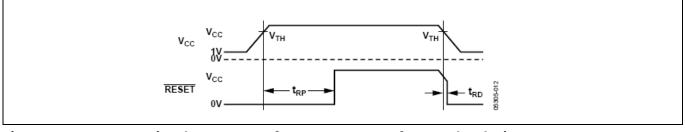


Figure 20 RESET# signal at power-on for ADM6384x27D2 from analog devices



# Summary

## 6 Summary

This application note provided the various options available for connecting and configuring the Infineon radiation-tolerant SPI NOR flash. More resources and FPGA-specific application notes related to programming and configuration can be found on the Xilinx website. Visit the Infineon website for more information on Infineon radiation-tolerant SPI NOR flash devices.

References



## References

- [1] UltraScale architecture configuration user guide UG570 from Xilinx
- [2] FlashcatUSB-Mach user guide from EmbeddedComputers
- [3] SPI configuration and flash programming in UltraScale FPGAs XAPP1233 from Xilinx

**Revision history** 



## **Revision history**

| Document<br>version | Date of release | Description of changes |
|---------------------|-----------------|------------------------|
| **                  | 2021-09-10      | Initial release        |

#### Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-09-10 Published by Infineon Technologies AG 81726 Munich, Germany

© 2021 Infineon Technologies AG. All Rights Reserved.

Do you have a question about this document? Go to www.cypress.com/support

Document reference 002-33563 Rev. \*\*

#### **IMPORTANT NOTICE**

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of noninfringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application. For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

#### WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.