



CrossLink Programming and Configuration Usage Guide

Preliminary Technical Note

FPGA-TN-02014 Version 1.1

February 2017

Contents

Acronyms in This Document	4
1. Overview	5
2. CrossLink Features	5
3. Definition of Terms	6
4. Configuration Process and Flow	7
4.1. Power-up Sequence	8
4.2. Initialization	8
4.3. Configuration Ports Default Behavior and Arbitration	8
4.4. Configuration	8
4.5. Wake-up	9
4.6. User Mode	9
4.7. Clearing the Configuration Memory and Re-initialization	9
4.8. Bitstream/PROM Sizes	10
4.9. Configuration Modes of CrossLink	10
4.10. sysCONFIG Pins	10
4.10.1. Self-Download Port Pins	11
4.10.2. Master and Slave SPI Configuration Port Pins	12
4.10.3. I ² C Configuration Port Pins	15
5. Configuration Modes	16
5.1. SDM Mode	16
5.2. Master SPI Configuration Mode	16
5.3. Dual Boot Configuration Mode	17
5.4. Slave SPI Mode	18
5.5. I ² C Configuration Mode	19
5.6. TransFR Operation	21
6. Software Selectable Options	23
6.1. Configuration Mode and Port Options	24
6.2. Bitstream Generation Options	25
6.3. Security Options	26
7. Device Wake-up Sequence	27
7.1. Wake-up Signals	27
References	28
Technical Support Assistance	28
Revision History	28

Figures

Figure 5.1. Configuration Flow	7
Figure 5.2. Period CRESETB is always Observed	11
Figure 5.3. Configuration from CRESETB Timing.....	11
Figure 5.4. Configuration Error Notification	12
Figure 6.1. Slave SPI Configuration Mode.....	19
Figure 6.2. I ² C Configuration Logic.....	20
Figure 6.3. Bitstream Update Using TransFR	21
Figure 6.4. Example Process Flow.....	22
Figure 7.1. sysCONFIG Preferences in Global Preferences Tab, Diamond Spreadsheet View	23

Tables

Table 5.1. Slave Configuration Port Activation Key	8
Table 5.2. Maximum Configuration Bits	10
Table 5.3. Default State of sysCONFIG Pins	11
Table 5.4. Default State in Diamond for each Port	11
Table 5.5. Master SPI Configuration Port Pins.....	12
Table 5.6. Slave SPI Configuration Port Pins	13
Table 6.1. Master SPI Configuration Port Pins.....	16
Table 6.2. Master SPI Configuration Software Settings	17
Table 6.3. Dual-Boot Configuration Settings	18
Table 6.4. Slave SPI Port Pins	18
Table 6.5. I ² C Port Pins.....	19
Table 6.6. Slave Addresses for I ² C Ports	20
Table 7.1. Configuration Mode/Port Options	24
Table 7.2. Configuration Mode/Port Options	26

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
CRC	Cyclic Redundancy Check
EBR	Embedded Block RAM
GOE	Global Output Enable
GSR	Global Set Reset
GWDIS	Global Write Disable
I ² C	Inter-Integrated Circuit
LUT	Look Up Table
MSPI	Master Serial Peripheral Interface
NVCM	Non Volatile Configuration Memory
POR	Power On Reset
SDM	Self Download Mode
SSPI	Slave Serial Peripheral Interface

1. Overview

The Lattice Semiconductor CrossLink™ is an SRAM-based Programmable Logic device that includes an internal Non-Volatile Configuration Memory (NVCM), as well as flexible SPI and I²C configuration modes. CrossLink provides a rich set of features for the programming and configuration of the FPGA. Many options are available for building the programming solution that fits your needs. The following sections describe each of the options in detail.

2. CrossLink Features

The key programming and configuration features of CrossLink devices are:

- Instant-on configuration from internal NVCM – powers up in milliseconds
- Single-chip, secure solution
- Multiple programming and configuration interfaces:
 - Self download
 - Slave SPI
 - Master SPI
 - Dual Boot
 - I²C
- Optional dual boot with external SPI memory
- Optional security bits for design protection

3. Definition of Terms

This document uses the following terms to describe common functions:

- **BIT** – The BIT file is the configuration data for CrossLink that is stored in an external SPI Flash. It is a binary file and is programmed unmodified into the SPI Flash.
- **Configuration** – Configuration refers to a change in the state of the CrossLink SRAM memory cells.
- **Configuration Data** – This is the data read from the non-volatile memory and loaded into the FPGA's SRAM configuration memory. This is also referred to as a bitstream, or device bitstream.
- **Configuration Mode** – The configuration mode defines the method CrossLink uses to acquire the configuration data from the non-volatile memory.
- **Dummy Byte** – A dummy byte is any data in which the numeric value is considered to be invalid. In some cases, external devices controlling the resident FPGA scan in dummy bytes as a requirement of the protocol.
- **Internal NVCM** – The BIT file can be programmed directly into the internal NVCM. The user does not need to know where an actual page of the configuration data starts. The CrossLink configuration engine handles the parsing in the NVCM to SRAM transfer.
- **Number Formats** – The following nomenclature is used to denote the radix of numbers
 - 0x: Numbers preceded by '0x' are hexadecimal
 - b (suffix): Numbers suffixed with 'b' are binary
 - All other numbers are decimal
- **NVCM** – Internal, lowest-cost, secure, one-time programmable Nonvolatile Configuration Memory.
- **Port** – A port refers to the physical connection used to perform programming and some configuration operations. Ports on CrossLink include SPI and I²C physical connections.
- **Programming** – Programming refers to the process used to alter the contents of the internal or external non-volatile configuration memory.
- **User Mode** – CrossLink is in User Mode when configuration is complete, and the FPGA is performing the logic functions it was programmed to perform.

4. Configuration Process and Flow

Prior to becoming operational, the FPGA goes through a sequence of states, including initialization, configuration and wake-up. Figure 4.1 shows the configuration flow.

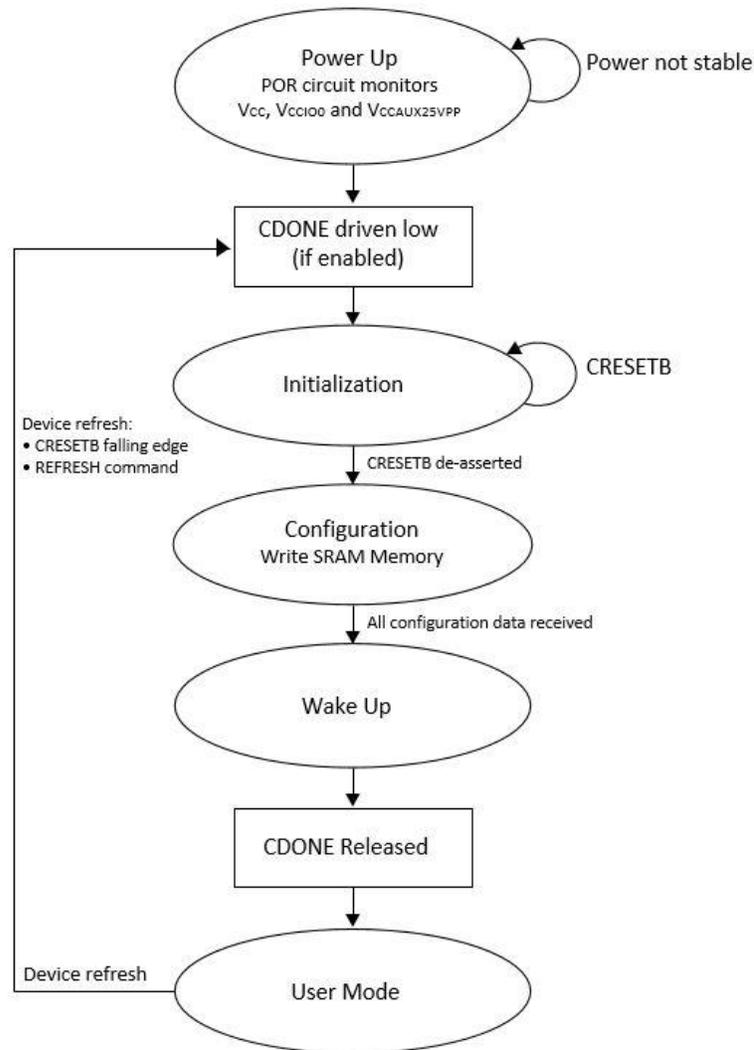


Figure 4.1. Configuration Flow

The CrossLink sysCONFIG ports provide industry standard communication protocols for programming and configuring the FPGA. Each protocol listed in Table 4.1 provides a way to access the CrossLink device’s internal NVCM, or to load its configuration SRAM.

The sysCONFIG ports capable of accessing the NVCM have a priority order. Table 4.1 lists each of the sysCONFIG ports in their priority order. The MSPI configuration port does not have the ability to alter the NVCM space, and as a result is not a factor in the sysCONFIG port priority scheme.

4.1. Power-up Sequence

In order for CrossLink to operate, power must be applied to the device. For a short period of time, as the voltages applied to the system rise, the FPGA will have an indeterminate state.

As power continues to ramp, a Power On Reset (POR) circuit inside the FPGA becomes active. The POR circuit, once active, makes sure the external I/O pins are in a high-impedance state. It also monitors the V_{CC} , V_{CC100} and $V_{CCAUX25VPP}$ input rails. Refer to FPGA-DS-02007, [CrossLink Family Data Sheet](#) for exact Power On Voltage levels.

When POR conditions are met, the POR circuit releases an internal reset strobe, allowing the device to begin its initialization process. CrossLink drives CDONE low.

4.2. Initialization

CrossLink enters the memory initialization phase immediately after the Power On Reset circuit drives the CDONE status pin low. The purpose of the initialization state is to clear all of the SRAM memory inside the FPGA.

The FPGA remains in the initialization state until the CRESETB pin is deasserted.

4.3. Configuration Ports Default Behavior and Arbitration

At Power Up or CRESETB pin toggle LOW or REFRESH command execution, the Configuration Logic puts the device into master auto booting mode. The device boots either from internal NVCM or external SPI boot PROM, if the CRESETB pin is “HIGH”, after a brief internal initialization time.

The blank CrossLink device employs the default BOOT_UP_SEQUENCE for Dual Boot configuration mode with the NVCM-EXT. The configuration engine first attempts to boot from the NVCM. If it fails due to blank NVCM, it tries to boot from the external SPI Flash using the MSPI configuration mode as a default behavior.

Holding the CRESETB LOW postpones the master auto booting event, and allows the slave configuration ports (Slave SPI or Slave I²C) to detect a ‘Slave Active’ condition. This means that the slave port is addressed, the Slave Configuration Port Activation Key (as listed in [Table 4.1](#)) is sent in, and the Activation Key matches the pre-defined key code. If any slave port declares active before CRESETB is released HIGH, the device is activated for slave configuration. If no slave port is declared active before the CRESETB pin is released HIGH, the device performs master auto booting sequence.

Table 4.1. Slave Configuration Port Activation Key

Active Key	Header	Data
Slave SPI Port	Dummy Bytes ¹	32’HA4C6F48A
I ² C Port	Slave I ² C Port Address Write ²	32’HA4C6F48A

Notes:

1. The number of dummy bytes should be at least 1.
2. The slave I²C address could be either 7 bits or 10 bits address.

The I²C and SPI pins are intentionally shared (MCK/SPI_SCK/SDA and CSN/SPI_SS/SCL) in such manner to prevent unintentional activation of either port. For example, a valid I²C interface can never inadvertently activate the SPI port and vice versa.

4.4. Configuration

The FPGA is able to accept the configuration bitstream created by the Lattice Diamond® development tools.

CrossLink begins fetching configuration data from non-volatile memory. The memory used to configure CrossLink is either the internal NVCM, or an external SPI Flash.

During configuration, the external SPI Flash is accessed in MSPI mode in the following two cases:

- Case 1: At HW default state, with NVCM-EXT boot up sequence.
- Case 2: If the BOOT_UP_SEQUENCE configuration option has been set as EXT, NVCM-EXT, EXT-NVCM and EXT-EXT in the device feature row, the configuration engine uses the MSPI mode. The only setting for BOOT_UP_SEQUENCE that does not use the MSPI mode is NVCM.

Note: The MSPI persistence has no effect with the availability of MSPI mode to the configuration engine during device configuration.

CrossLink does not leave the Configuration state if there are no memories with valid configuration data. In this case, only the SSPI and I²C mode may be used to program the device when it is in a blank/erased state.

4.5. Wake-up

Wake-up is the transition from configuration mode to User Mode. CrossLink's fixed four-phase Wake-up sequence starts when the device has correctly received all of its configuration data. When all configuration data is received, the FPGA asserts an internal DONE status bit. The assertion of the internal DONE causes a Wake-up state machine to run that sequences four controls. The four control strobes are:

- External CDONE
- Global Write Disable (GWDIS)
- Global Output Enable (GOE)
- Global Set/Reset (GSR)

In the first phase of the Wake-up process at default software settings, CrossLink releases the Global Output Enable and asserts the Global Write Disable.

When Global Output Enable is asserted, it permits the FPGA's I/O to exit a high-impedance state and take on their programmed output function. The FPGA inputs are always active. The input signals are prevented from performing any action on the FPGA flip-flops by the assertion of the Global Set/Reset (GSR).

The Global Write Disable is a control that overrides the write enable strobe for all RAM logic inside the FPGA. The inputs on the FPGA are always active, as mentioned in the Global Output Enable section. Keeping GWDIS asserted prevents accidental corruption of the instantiated RAM resources inside the FPGA.

The second phase of the Wake-up process releases the Global Set/Reset and the Global Write Disable controls.

The Global Set/Reset is an internal strobe that, when asserted, causes all I/O flip-flops, Look Up Table (LUT) flip-flops, distributed RAM output flip-flops, and Embedded Block RAM output flip-flops that have the **GSR enabled** attribute to be set/cleared per their hardware description language definition.

The last phase of the Wake-up process is to assert the external CDONE pin. The CDONE pin may also be held low externally to delay the User Mode entry in order to synchronize with other devices. This behavior is configurable, see the [sysCONFIG Pins](#) section on page 10 for details on the CDONE pin.

When the final Wake-up phase is complete, the FPGA enters User Mode.

4.6. User Mode

CrossLink enters User Mode immediately when the Wake-up sequence has completed. User Mode is the point in time when CrossLink begins performing the logic operations you designed. CrossLink remains in this state until the configuration memory is cleared or power is lost.

4.7. Clearing the Configuration Memory and Re-initialization

The current User Mode configuration of CrossLink remains in operation until it is actively cleared, or power is lost. Several methods are available to clear the internal configuration memory of CrossLink:

- Remove power and reapply power.
- Execute an Erase command while in programming mode
- Toggle the CRESETB pin.
- Reinitialize the memory through a Refresh command. Any active configuration port can be used to send a Refresh command.

Invoking one of these methods causes CrossLink to drive CDONE low. CrossLink enters the initialization state as described earlier.

4.8. Bitstream/PROM Sizes

CrossLink is an SRAM based FPGA. The SRAM configuration memory must be loaded from a non-volatile memory that can store all of the configuration data. The size of the configuration data is variable. It is based on the amount of logic available in the FPGA, and the number of pre-initialized Embedded Block RAM (EBR) components. A CrossLink design using the largest device, with every EBR pre-initialized with unique data values, and generated without compression turned on requires the largest amount of storage.

Table 4.2. Maximum Configuration Bits

Device	Bitstream Size Without Pre-Initialized EBR	Bitstream Size With Maximum Number of Pre-Initialized EBR	Units
LIF-MD6000	1.24	1.59	Mb

4.9. Configuration Modes of CrossLink

The CrossLink configuration SRAM memory must be loaded with valid configuration data before the FPGA operates. CrossLink provides four modes of loading the configuration data into the SRAM memory. The four modes available are Self-Download (NVCM) mode, Master SPI mode, Slave SPI mode and I²C Configuration mode. Dual-boot operation is supported as a combination of the Self-Download mode and Master SPI mode.

4.10. sysCONFIG Pins

CrossLink provides a set of sysCONFIG I/O pins to program and configure the FPGA. The sysCONFIG pins are grouped together to create ports (that is SSPI, I²C, MSPI) that are used to interact with the FPGA for programming, configuration, and access of resources inside the FPGA. The sysCONFIG pins in a configuration port group may be active, and used for programming the FPGA, or they can be reconfigured to act as general purpose I/O.

Recovering the configuration port pins for use as general purpose I/O requires you to adhere to the following guidelines:

- You must DISABLE the unused port. You can accomplish this by using the Diamond Spreadsheet View’s Global Preferences tab. Each configuration port is listed in the sysCONFIG options tree.
- You must prevent external logic from interfering with device programming. Make sure that recovered sysCONFIG pins are not asserted when CrossLink is in Feature Row HW Default Mode state. One example is driving CRESETB with an active low signal after CrossLink is in Feature Row HW Default Mode state. Failure to reprogram the Feature Row with CRESETB disabled prevents the FPGA from configuring and entering User Mode.

[Table 4.3](#) lists the default state of the shared sysCONFIG pins. A device with an erased/HW default Feature Row has the SPI Slave and I²C ports enabled. Upon entry to User Mode, the state of the SSPI and I²C ports is determined by the sysCONFIG port settings. The SW default sysCONFIG port setting is SSPI is enabled while I²C is disabled. This means you lose the ability to program CrossLink using I²C when using the default sysCONFIG port settings. You must assert CRESETB to program over I²C in that case. To retain the I²C sysCONFIG pins in User Mode, be sure to ENABLE them using the Diamond Spreadsheet View editor.

The sysCONFIG pins are powered by the VCCIO0 voltage. It is crucial that you take this into consideration when provisioning other logic attached to Bank 0.

The function of each sysCONFIG pin is described in detail in [Table 4.3](#).

Table 4.3. Default State of sysCONFIG Pins

Pin Name	Associated sysCONFIG Port	Pin Function in Feature Row Blank Mode (Configuration Mode)	Pin Direction (Configuration Mode)	Default Function in User Mode (Software Default State)
CRESETB	SDM	CRESETB	Input with weak pull up	CRESETB
CDONE	SDM	I/O	I/O with weak pull up	User-defined I/O
SPI_SCK/MCK/SDA	SSPI/MSPI/I ² C	SSPI/I ² C	Input with weak pull up	SSPI
SPI_SS/CSN/SCL	SSPI/MSPI/IC	SSPI/I ² C	Input with weak pull up	SSPI
MOSI	SSPI/MSPI	SSPI	Input	SSPI
MISO	SSPI/MSPI	SSPI	Output	SSPI

Note: All pins are in Configuration Mode until the device is configured and enters User Mode.

Table 4.4. Default State in Diamond for each Port

sysConfig Port	Diamond Default ¹
CDONE_PORT	CDONE_USER_IO
SLAVE_SPI_PORT	Enable
I2C_PORT	Disable
MASTER_SPI_PORT	Disable ²

Note:

1. This default setting can be modified in the Diamond Spreadsheet View, Global Preferences tab.
2. The MASTER_SPI_PORT setting does not influence the behavior during configuration. For details, see the [Configuration](#) section.

4.10.1. Self-Download Port Pins

CRESETB: The CRESETB is an input used to configure the FPGA. The CRESETB pin, when enabled, is sensitive to a high-to-low transition, and has an internal weak pull-up. When CRESETB is asserted low, the FPGA exits User Mode and starts a device configuration sequence at the Initialization phase, as described earlier. Holding the CRESETB pin low prevents CrossLink from leaving the Initialization phase. The CRESETB has a minimum pulse width assertion period in order for it to be recognized by the FPGA. You can find this minimum time in FPGA-DS-02007, [CrossLink Family Data Sheet](#) in the AC timing section.

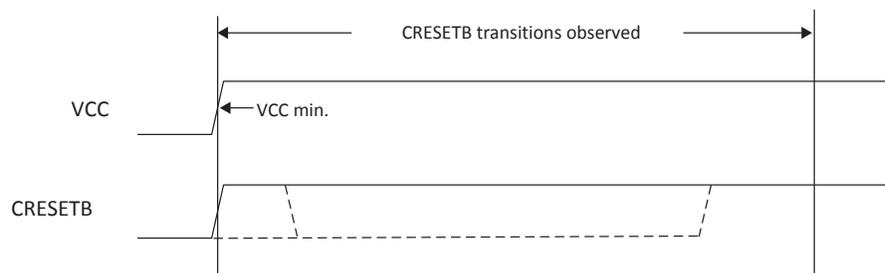


Figure 4.2. Period CRESETB is always Observed

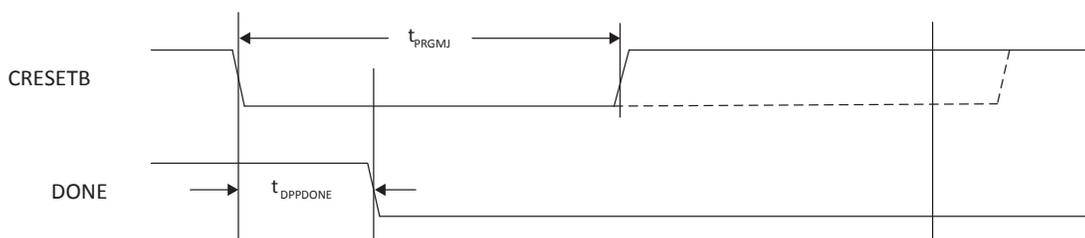


Figure 4.3. Configuration from CRESETB Timing

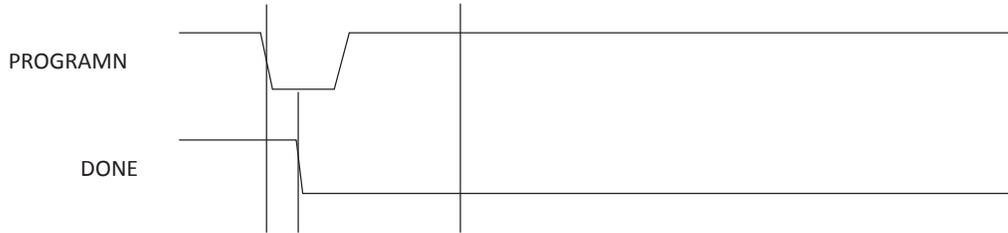


Figure 4.4. Configuration Error Notification

If an error is detected when reading the bitstream, the internal DONE bit is not set, the CDONE pin stays low, and the device does not wake up. The device configuration fails when the following occurs:

- The bitstream CRC error is detected.
- The invalid command error is detected.
- A timeout error is encountered when loading from the on-chip Flash.
- The program DONE command is not received when the end of on-chip SRAM configuration or on-chip NVCM is reached.
- Device ID code mismatch.

CDONE: The CDONE pin is a bi-directional open drain with a weak pull-up that signals the FPGA is in User mode. CDONE is first able to indicate entry into User mode only after an internal DONE bit is asserted. The internal DONE bit defines the beginning of the FPGA Wake-up state.

The CDONE output pin is controlled by the CDONE_PORT and DONE_EX configuration parameter that is modified in the Diamond Spreadsheet View. By default, the CDONE pin is a general purpose I/O when CrossLink is in the Feature Row HW Default Mode state. The default mode causes CrossLink to automatically pass through the Wake-up sequence after the internal DONE bit is asserted. The FPGA does not stall waking up waiting for the CDONE pin to be asserted high.

The FPGA can be held from entering User Mode indefinitely by having an external agent keep the CDONE pin asserted low. In order to use CDONE to stall entering User Mode, the CDONE_PORT must be set to CDONE_ONLY and the DONE_EX set to ON, these settings are to be changed from Diamond Spread Sheet View. If DONE_EX = ON, the device waits for CDONE to be driven high by an external signal to wake up. A common reason for keeping CDONE driven low is to allow multiple FPGAs to be completely configured. As each FPGA reaches the DONE state, it is ready to begin operation. The last FPGA to configure can cause all FPGAs to start in unison.

The CDONE pin drives low when the FPGA enters Initialization mode. As described earlier, this condition happens when power is applied, CRESETB is asserted, or a Refresh command is received via an active configuration port.

4.10.2. Master and Slave SPI Configuration Port Pins

Table 4.5. Master SPI Configuration Port Pins

Pin Name	Function	Direction	Description
MCK	MCK	Output with weak pullup	Master clock used to time data transmission/reception from the CrossLink Configuration Logic to a slave SPI PROM.
CRESET_B	CRESET_B	Input	CRESETB is used to initiate programming / configuration
MOSI	MO	Output	This is the Master output which carries configuration commands to the external SPI PROM.
MISO	MI	Input	This is the input to the Master which carries output data from the slave SPI PROM to the CrossLink Configuration Logic.
CSN	CSN	Output	CrossLink Master SPI chip select output pin for external SPI Flash.

Table 4.6. Slave SPI Configuration Port Pins

Pin Name	Function	Direction	Description
SPI_SCK	SPI_SCK	Input with weak pullup	Clock used to time data transmission/reception from an external SPI master device to the CrossLink Configuration Logic.
CRESET_B	CRESET_B	Input	CRESETB is used to initiate programming / configuration
MOSI	SI	Input	This is the input to the slave which receives data from the external SPI master to the CrossLink Configuration Logic.
MISO	SO	Output	This is the output from the slave which carries output data from the CrossLink Configuration Logic to the external SPI master.
SPI_SS	SPI_SS	Input with weak pullup	CrossLink Configuration Logic slave SPI chip select input.

MCK/SPI_SCK: The MCK/SPI_SCK, when active, are clocks used to sequentially load the configuration data for the FPGA. The pin functions as:

The MCK/SPI_SCK pin's default state for a CrossLink in the Feature Row HW Default Mode state is to act as the configuration clock (that is MCK or SPI_SCK). This allows an external Slave SPI master controller to program CrossLink. The maximum SPI_SCK frequency and the data setup/hold parameters can be found in the AC timing section of FPGA-DS-02007, [CrossLink Family Data Sheet](#). The Feature Row must be configured to ENABLE the Slave SPI Port if you want to use the port to reprogram CrossLink after it enters User Mode.

The MCK/SPI_SCK pin functions as a Master Clock (MCK) when CrossLink is configured in Dual Boot or External Boot modes. The MCK becomes an output and provides a reference clock for a SPI Flash attached to the CrossLink's Master SPI Configuration port. MCK actively drives until all of the configuration data has been received. When CrossLink enters User Mode, the MCK output tri-states. This allows the MCK to become a general purpose I/O. The MCK is reserved for use, in most post-configuration applications, as the reference clock for performing memory transactions with the external SPI PROM.

CrossLink generates MCK from an internal oscillator. The initial frequency of the MCK is nominally 2 MHz. The MCK frequency can be altered using the MCCLK_FREQ parameter. You can select the MCCLK_FREQ using the Diamond Spreadsheet View. Following is a complete list of supported MCCLK frequencies:

- 2 MHz
- 3 MHz
- 6 MHz
- 12 MHz
- 24 MHz
- 48 MHz

During the initial stages of device configuration, the frequency value specified using MCCLK_FREQ is loaded into the FPGA. When CrossLink accepts the new MCCLK_FREQ value, the MCK output begins driving the selected frequency.

Make sure when selecting the MCCLK_FREQ that you do not exceed the frequency specification of your configuration memory, or of your PCB. Refer to the CrossLink AC specifications in FPGA-DS-02007, [CrossLink Family Data Sheet](#) when making MCCLK_FREQ decisions.

SPI_SS: The SPI_SS pin is the Slave SPI ports chip select. An external SPI bus master asserts the SPI_SS pin active low in order to perform actions using CrossLink's programming and Configuration Logic. The SPI_SS pin is available when CrossLink is in the Feature Row HW Default Mode state, and in User Mode when the Slave SPI port is set to the ENABLE setting. The SPI_SS pin is a general purpose I/O in User Mode when the Slave SPI port is set to the DISABLE setting.

Proper operation of the CrossLink device depends upon maintaining the SPI_SS pin in the correct state:

- SPI_SS must be deasserted (that is, held High) when configuring using Master SPI mode. SPI_SS signal needs to be clean during power up. Noise on SPI_SS pins may cause device failing to download from flash. SPI_SS must be asserted when configuring using Slave SPI mode.
- SPI_SS must be deasserted when CrossLink is in User Mode, and SPI memory transactions are initiated using the internal WISHBONE bus.
 - The Master SPI port and the Slave SPI port share three common pins, MOSI, MISO, and MCK/SPI_SS. They are not permitted to be accessed at the same time. In Diamond, if both the ports are enabled at the same time, the flow fails.
- SPI_SS must be deasserted (even if recovered for GPIO) whenever the Feature Row is erased via I²C sysConfig port (for example embedded reconfiguration). If asserted, configuration may not complete successfully.

Lattice recommends the SPI_SS pin to be pulled high externally to augment the weak internal pull-up.

Note: In case of CrossLink, the SPI_SS pin is shared with I²C SCL line. The startup sequence for CrossLink decides Slave SPI and Slave I²C pins while the device is in configuration mode. The Startup state machine determines if either the I²C or the SPI slave mode is activated, and if so, identifies which one is activated. When one is activated, the other is locked out until the next refresh event or power cycle and so are its concerned pins.

CSN: The CSN pin is an active low chip select used by the Master SPI configuration mode to enable an external SPI Flash. When CrossLink is programmed to configure in either External or Dual Boot mode, the CSN pin is asserted to the attached SPI Flash. CrossLink asserts CSN until all configuration data bytes have been loaded, at which time the MCK enters a high impedance state.

When CrossLink is in the Feature Row HW Default Mode state, the CSN is SPI_SS with a weak pullup. It must have an external pullup resistor when the External and Dual Boot configuration modes are used. MCK must ramp in tandem with the SPI PROM VCC input. It remains SPI_SS when the FPGA enters User Mode in software default state. You must ENABLE the Master SPI port to reserve CSN for use by the internal SPI Master logic.

When configuring from an external SPI Flash, ensure that the SPI Flash VCC and the CrossLink VCCIO0 are at the same level. Ensure that the SPI Flash VCC is at the recommended operating level.

Some SPI PROM manufacturers require the chip select input of the PROM ramp in unison to the PROMs VCC rail. The CSN pin, by default, has a weak pull-up resistor internally. Adding a 4.7 kΩ to 10 kΩ pull-up resistor to the CSSPIN pin on CrossLink is recommended.

MOSI: The MOSI is a dual function bi-directional pin. The direction depends upon whether a Master or Slave mode is active. The SI/SISPI is an input data pin when using the Slave SPI mode and is an output data pin when using the Master SPI mode. In Master SPI mode, CrossLink drives MOSI until all configuration data bytes have been loaded, at which time the MOSI enters a high impedance state.

At least one of the sysCONFIG preferences, MASTER_SPI_PORT or SLAVE_SPI_PORT, must be set to ENABLE in order to preserve this pin as MOSI and allow access to the SPI interface.

MISO: The MISO pin is a dual function bi-directional pin. The direction depends upon whether a Master or Slave mode is active. The MISO is an input data pin when using the Master SPI mode and is an output data pin when using the Slave SPI mode.

At least one of the sysCONFIG preferences, MASTER_SPI_PORT or SLAVE_SPI_PORT, must be set to ENABLE in order to preserve this pin as SO/SPISO and allow access to the SPI interface.

CRESET_B: CRESET_B is a configuration reset pin which is pulled. When CRESETB is asserted low, the FPGA exits User Mode and starts a device configuration sequence at the Initialization phase, as described earlier. Holding the CRESETB pin low prevents CrossLink from leaving the Initialization phase.

4.10.3. I²C Configuration Port Pins

SCL: CrossLink provides an I²C configuration port. The SCL is the bi-directional I²C Serial Clock pin, and is used to initiate and time transactions on the I²C bus. SCL requires an external pull-up resistor in order to operate.

The SCL pin is available as a user I/O when CrossLink is in the Feature Row HW Default Mode state. You must ENABLE the I2C_PORT for the configuration access to continue to be available in User Mode (see the [I2C Configuration Mode](#) section on page 19 for details.) The SCL pin becomes a general purpose I/O if you do not ENABLE the I2C_PORT. The configuration SCL pin is not shared with the I2C0 USER_SCL pin. The I2C0 and I2C1 User Mode I2C blocks operate independently of the configuration I2C block.

SDA: The SDA pin is the I²C serial data input/output pin. It is bi-directional, open-drain, and requires an external pull-up resistor in order to operate. The pin changes direction dynamically during data transactions on the I²C bus. The current state depends on the current bus master and the operation being performed by that master.

The SDA pin is available as a user I/O when CrossLink is in the Feature Row HW Default Mode state. You must ENABLE the I2C_PORT for the configuration access to be available in User Mode (see the [I2C Configuration Mode](#) section for details.) The SDA pin becomes a general purpose I/O if you do not ENABLE the I2C_PORT. The configuration SDA pin is not shared with the I2C0 USER_SDA pin.

5. Configuration Modes

CrossLink provides multiple options for loading the configuration SRAM from a non-volatile memory. The previous section describes the physical interface necessary to interact with the CrossLink Configuration Logic. This section describes the functionality of each of the different configuration modes. Descriptions of important settings required in the Diamond Spreadsheet View are also discussed. See the [Configuration](#) section on page 8 for details on the default configuration behavior of the device.

5.1. SDM Mode

Self-Download Mode is the primary configuration method for CrossLink. The advantages of Self Download Configuration Mode include:

- **Speed:** CrossLink is ready to run in a few milliseconds depending on the density of the device.
- **Security:** The configuration data is never seen outside the device during the load to SRAM. You can prevent the internal memory from being read.
- **Reduced cost:** There is no need to purchase a PROM specifically reserved for programming CrossLink.
- **Reduced board space:** Elimination of an external PROM allows your board to be smaller.

CrossLink retrieves the configuration data from the internal NVCM when it is using Self Download Mode. SDM is triggered when power is applied, a REFRESH command is received, or by asserting the CRESETB pin.

5.2. Master SPI Configuration Mode

Master SPI (MSPI) configuration mode is the only other self-controlled configuration mode available to CrossLink. Lattice recommends having a secondary configuration port available that is active when CrossLink is in Feature Row HW Default Mode state (that is, blank). The secondary port allows you to recover CrossLink in the event of a programming error.

For CrossLink to operate correctly using the MSPI configuration mode, ensure that:

- The POR of the SPI Flash device is lower than the POR of CrossLink or the SPI Flash is powered first.
- SPI Flash Fmax is greater than CrossLink MCK Fmax.
- Board routing requirements are checked to ensure CrossLink setup and hold time parameters are met. Refer to the FPGA-DS-02007, [CrossLink Family Data Sheet](#) for detailed setup and hold time information.

Table 5.1. Master SPI Configuration Port Pins

Pin Name	Function
MCK	Clock output from the CrossLink Configuration Logic and Master SPI controller. Connect MLK to the SCLK input of the Slave SPI device.
MOSI	Serial Data output from CrossLink to the slave SPI SI input.
MISO	Serial Data input to the CrossLink Configuration Logic from the slave SPI SO output.
CSN	Chip select output from the CrossLink Configuration Logic to the slave SPI Flash holding configuration data for CrossLink.

[Table 4.2](#) provides information about the amount of memory needed for CrossLink configuration data by device density. Select an SPI Flash that accepts 03 hex Read Opcodes. CrossLink is only able to use the 03 hex Read Opcode.

CrossLink begins retrieving configuration data from the SPI Flash when power is applied, a REFRESH command is received, or the CRESETB pin is asserted and released. The MCK/SPI_SCK I/O takes on the Master Clock (MCK) function, and begins driving a nominal 2 MHz clock to the SPI Flash's SCLK input. CSSPIN is asserted low, commands are transmitted to the PROM over the MOSI output, and data is read from the PROM on the MISO input pin. When all of the configuration data is retrieved from the PROM, the CSSPIN pin is deasserted and the MSPI output pins are tri-stated.

The MCK frequency always starts downloading the configuration data at the nominal 2 MHz frequency. The MCLK_FREQ parameter, accessed using Spreadsheet View, can be used to increase the configuration frequency.

The configuration data in the PROM has some padding bits, and then the data altering the MCK base frequency is read. CrossLink reads the remaining configuration data bytes using the new MCK frequency.

After CrossLink enters User Mode, the Master SPI configuration port pins tri-state. This permits background programming of or access to the SPI Flash. To set CrossLink for operation using the MSPI configuration mode:

- Store the entire configuration data in an external SPI Flash
- The data must start at offset 0x000000 within the PROM
- Set the preferences as listed in [Table 5.2](#)
- Enable Bitstream File creation in the Diamond Process Pane
- Run the Export Files process to build your design

Table 5.2. Master SPI Configuration Software Settings

Preference	Setting
MASTER_SPI_PORT	ENABLE
BOOT_UP_SEQUENCE	EXT

The Export Files process generates both a PROM file and a BIT file. The BIT file must be programmed into the external SPI Flash. There are several ways to get the data into the SPI Flash:

- Diamond Programmer can transmit the SPI Flash data using a download cable
- An on-board SOC can program the SPI Flash
- Automatic test equipment can program the SPI Flash
- Pre-programmed SPI Flash memories can be pre-assembled onto your printed-circuit board

When CrossLink Feature Row is programmed and the SPI Flash contains your configuration data, you can test the configuration. Assert the CRESETB, transmit a REFRESH command, or cycle power to the board, and CrossLink is configured from the external SPI Flash.

5.3. Dual Boot Configuration Mode

Dual Boot Configuration Mode is a combination of Self Download Mode and Master SPI Configuration Mode. When set up in Dual Boot Mode, CrossLink tries to configure first from a primary image stored in external SPI Flash or NVCM. If the primary image configuration fails, CrossLink attempts to configure itself using a failsafe golden image stored in either external SPI Flash or NVCM. The load order can be changed by setting the BOOT_UP_ORDER preference.

The primary image can fail in one of several ways:

- A bitstream CRC error is detected during configuration
- A time-out error is encountered when loading the configuration SRAM
- A Device ID mismatch occurs during configuration
- An illegal command is asserted which can cause failure

A CRC error is caused by incorrect data being written into the internal NVCM or external SPI Flash. The configuration data is read out in rows. As each row enters the Configuration Engine the data is checked for CRC consistency. Before the data enters the Configuration SRAM the CRC must be correct. Any incorrect CRC causes the device to erase the Configuration SRAM and retrieve configuration data from the failsafe image.

There is a corner case wherein it is possible for the data to be correct from a CRC calculation perspective, but not functionally correct. In this instance, the internal DONE bit never becomes active. CrossLink counts the number of master clock pulses it provided after the Power On Reset signal is released. When the count expires without DONE becoming active, the FPGA attempts to get its configuration data from the failsafe image.

The external SPI Flash must have a lower Power-On-Reset voltage supply level than the CrossLink POR to ensure proper configuration.

Dual boot configuration mode typically requires two configuration data files. One of the two configuration data files is a failsafe image that is rarely, if ever, updated. The second configuration data file is a working image (also called primary image) that is routinely updated. The failsafe image can be stored in the internal NVCM, with the working image stored in external SPI Flash. Alternatively, both images can be stored in the external SPI Flash. One Diamond project (or implementation) can be used to create both the working and the failsafe configuration data files.

Refer to the Diamond Online Help for more information about using Diamond implementations.

Use the following preferences to build a dual-boot design:

Table 5.3. Dual-Boot Configuration Settings

Preference	Dual-Boot Setting
MASTER_SPI_PORT	ENABLED
BOOT_UP_SEQUENCE	NVCM-EXT, EXT-NVCM, EXT-EXT

The failsafe configuration image generated by Diamond can be stored in either the NVCM or the external SPI Flash. In dual boot scenarios where only one image is stored in the external SPI Flash (BOOT_UP_SEQUENCE = NVCM-EXT or EXT-NVCM), the external image is stored in the SPI Flash starting at address 0x000000. This differs from a single image Master SPI Configuration Mode (BOOT_UP_SEQUENCE = EXT-EXT), which requires the primary configuration data to be stored at offset 0x000000 and the secondary configuration data at offset 0xFFFF00. The external SPI Flash memory file for dual boot can be generated using the Diamond Deployment Tool. Use the External Memory: Dual Boot option in Deployment Tool to generate the dual boot image.

The following processes are recommended for programming internal NVCM and external flash to use Dual Boot Mode:

Option A — Using background mode to program external flash:

1. Program CrossLink internal NVCM (using NVCM Programming Mode). Make sure SPI port enabled and persistent is on.
2. Program the external SPI Flash.
3. Refresh or power cycle.

Option B — Using offline mode to program external SPI Flash:

1. Program the external SPI Flash first (may be none-background mode).
2. Program CrossLink internal NVCM (using NVCM Programming Mode).
3. Refresh or power cycle.

5.4. Slave SPI Mode

CrossLink provides a Slave SPI (SSPI) configuration port that allows you to access features provided by the Configuration Logic. You can program the NVCM and access status/control registers within the Configuration Logic block. It is necessary to send a REFRESH command to load a new NVCM image into the SRAM.

Table 5.4. Slave SPI Port Pins

Pin Name	Description
SPI_SCK	Configuration clock input that is driven by a SPI master controller.
MOSI	Serial Data Input to the CrossLink Configuration Logic for command and data.
MISO	Serial Data Output from the CrossLink Configuration Logic.
SPI_SS	Chip select to enable the CrossLink Configuration Logic.

In the Slave SPI mode, the MLK/SPI_SCK pin becomes SPI_SCK (that is Configuration clock). Input data is read into the CrossLink device on the MOSI pin at the rising edge of SPI_SCK. Output data is valid on the MISO pin at the falling edge of SPI_SCK. The SPI_SS acts as the chip select signal. When SPI_SS is high, the SSPI interface is deselected and the MISO pin is tri-stated.

Commands can be written into and data read from CrossLink when SPI_SS is asserted. The CrossLink SSPI port only accepts Mode 0 bus transactions to the Configuration Logic, Where the Mode 0 bus transaction is the Master SPI setting of configuration master configured at CPHA = 0 and CPOL = 0.

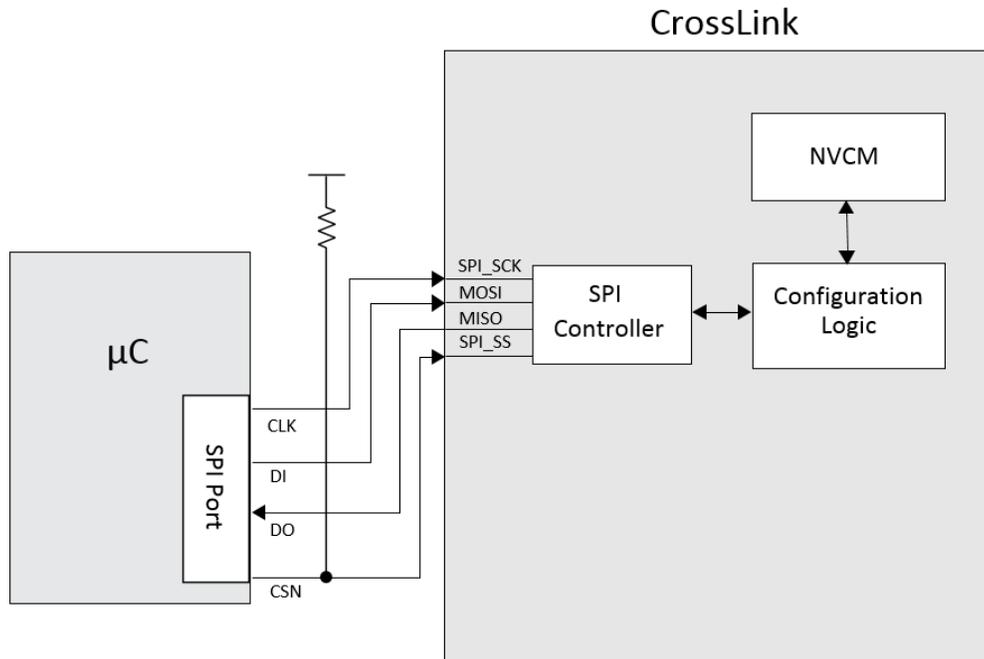


Figure 5.1. Slave SPI Configuration Mode

The SSPI port is active when CrossLink is in Feature Row HW Default Mode state (that is, blank/erased). Diamond's default preference for the SLAVE_SPI_PORT is to ENABLE the port. Use the Spreadsheet View to ENABLE the SPI_PORT preference in your design to keep the SSPI port active in User Mode.

The SSPI port is used to program and verify the NVCM or to configure the SRAM. The SSPI port can issue a REFRESH command to make a newly programmed image active. Programming CrossLink using the SSPI port is complex. Lattice provides 'C' source code called SSPIEmbedded to circumvent the complexity of programming CrossLink. Use SSPIEmbedded to reprogram the CrossLink NVCM. To modify the SSPIEmbedded code as per the user-specific environment programming master, refer to the [Programming Tools User Guide](#) document.

5.5. I²C Configuration Mode

CrossLink has an I²C Configuration port for use in accessing the Configuration Logic. An I²C master can communicate to the Configuration Logic using 10-bit or 7-bit addressing modes. You can reprogram the NVCM or configure the SRAM and access status/control registers within the Configuration Logic block. Note that only one of the SPI or I²C interfaces can be operated at a time. When the I²C interface is activated, all subsequent communication to the SPI port is ignored, even if in the middle of an active communication.

Table 5.5. I²C Port Pins

Pin Name	Description
SCL	I ² C bus clock
SDA	I ² C bus data line

The I²C Configuration port is available when CrossLink is in Feature Row HW Default Mode state (that is, blank/erased). The default state set for the I2C_PORT in the Diamond design software is to place the I2C_PORT in the DISABLE state. You must make sure the I2C_PORT is set to the ENABLE state to leave the I²C interface active in User Mode.

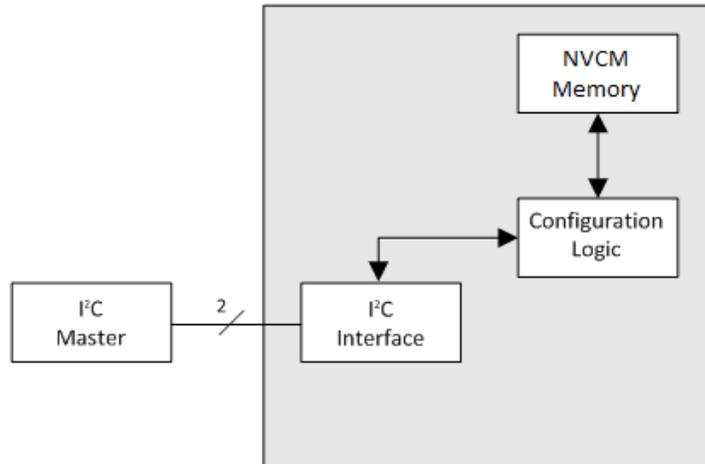


Figure 5.2. I²C Configuration Logic

An external I²C master accesses the Configuration Logic using address 1000000 (7-bit mode) or 1111000000 (10-bit mode) unless the I²C base address has been modified.

Table 5.6 lists the address decoding used to access the I²C resources in CrossLink.

Table 5.6. Slave Addresses for I²C Ports

Slave Address	I ² C Function
yyyxxxx00	Primary I ² C Controller Configuration Logic address. Always responds to 7-bit or 10-bit addresses.
yyyxxxx11	Primary I ² C Configuration Logic Reset. Always responds to 7-bit or 10-bit addresses.

Note: Although there are four possible combinations of the reserved address bits 1000 0XX, only the two combinations listed above are used. The remaining two addresses are reserved for future I²C bus enhancements.

The CrossLink I²C controller supports two separate slave addresses as listed in Table 5.6. These are determined by the two least significant bits in the slave address – 00 corresponds to the Configuration Logic, while 11 corresponds to a reset port. In some instances, an I²C memory transaction to the Configuration Logic may be interrupted or abandoned. It is possible for a command to be accepted by the Configuration Logic that causes the Configuration Logic to respond with data. In the event that the I²C memory transaction is interrupted or abandoned, the Configuration Logic continues to return the queued data.

New incoming I²C commands may be considered padding bytes or may be misinterpreted. Clear this condition by writing any value to the address with least significant bits 11. The Configuration Logic command interpreter is reset, any queued data is flushed, and subsequent I²C memory transactions to the Configuration Logic operates correctly.

5.6. TransFR Operation

CrossLink, like other Lattice FPGAs, provides for the TransFR™ capability. TransFR is described in TN1087, [Minimizing System Interruption During Configuration Using TransFR Technology](#).

Figure 5.3 is an example of how you can update bitstream in CrossLink by using the TransFR feature.

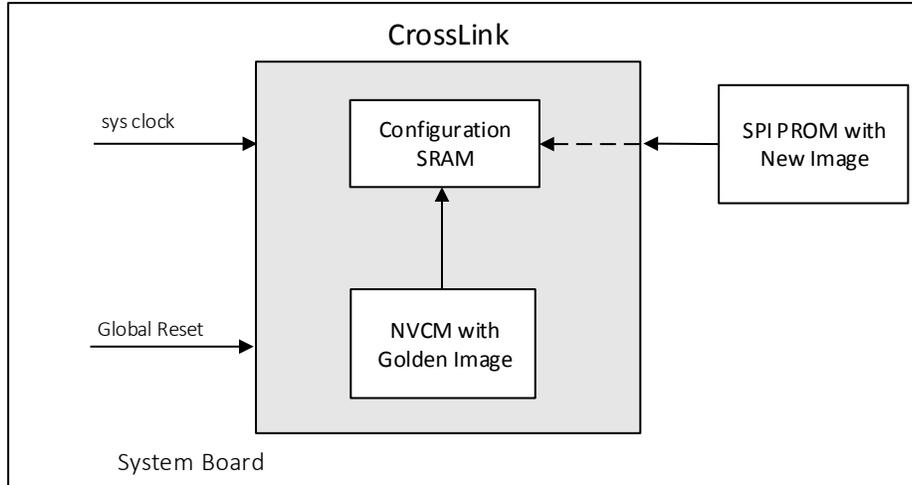
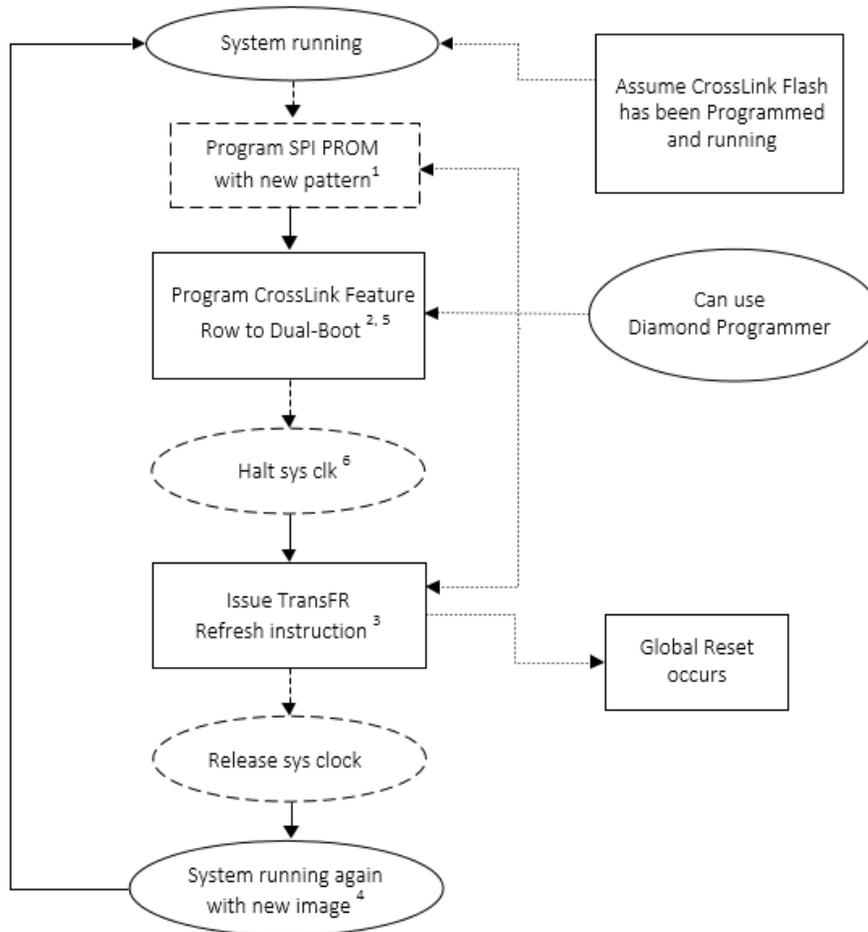


Figure 5.3. Bitstream Update Using TransFR

The example assumes the golden image is stored in NVCM to initiate the system, and then uses SPI PROM as a resource for image updates without disturbing the system. Figure 5.4 on the next page shows the process flow for performing this task.



Notes:

1. User can use operations such as “SPI Flash Background Erase, Program, Verify” for this.
2. User can use operations such as “XNVCN Program Feature Rows” for this.
3. User can use operations like “XNVCN TransFR” for this.
4. If new image failed to config CrossLink, the golden image in NVCM will still config CrossLink, so system will still be running with original image.
5. Feature Row only needs to be programmed if changes need to be made, for instance, disable or enable Slave Port. If no changes need to be made, please skip this step.
6. This step is optional.

Figure 5.4. Example Process Flow

Caution when using the above process flow:

As a Global Reset is triggered during device wake-up after Refresh instruction is issued, attention needs to be given in designing I/O with following conditions:

- Register output pins
- Impact on the system board level when value changes (may shut off the board, for instance)
- Register is set/reset by global reset

For the I/O in the example above, the state of the I/O is not changed during the TransFR refresh, but may change when the device gets into User Mode right after the TransFR refresh. Following are design tips to avoid this:

- For critical I/Os, do not use global reset
- For critical I/Os, if you have to use global reset, use the set/reset option so that when GSR occurs, the state of the I/O pin does not trigger a system crash.

6. Software Selectable Options

The operation of the CrossLink Configuration Logic is managed by options selected in the Diamond design software. Other FPGAs provide dedicated I/O pins to select the configuration mode. CrossLink uses the non-volatile Feature Row to select how it will configure. The Feature Row's default state needs to be modified in almost every design. Use the Diamond Spreadsheet View to make the changes to the operation of the CrossLink Feature Row which alters the operation of the Configuration Logic.

The Configuration Logic preferences are accessed using Spreadsheet View. Click the Global Preferences tab and look for the sysCONFIG tree. Figure 6.1 shows the sysCONFIG section. The sysCONFIG preferences are divided into three categories:

- Configuration mode and port related
- Bitstream generation related
- Security related

Preference Name	Preference Value
Junction Temperature (Tj)(C)	100
Voltage (V)	1.140
SYSTEM_JITTER(ns)	Default
▲ Block Path	
Block Asynchpaths	OFF
Block Resetpaths	OFF
Block RD During WR Paths	OFF
Block InterClock Domain P...	OFF
Block Jitter	OFF
▲ sysConfig	
SLAVE_SPI_PORT	DISABLE
MASTER_SPI_PORT	DISABLE
I2C_PORT	DISABLE
MCCLK_FREQ	2
MY_ASSP	OFF
ONE_TIME_PROGRAM_SRAM	DISABLE
ONE_TIME_PROGRAM_NVCM	DISABLE
SECURITY_SRAM	DISABLE
SECURITY_NVCM	DISABLE
TRANSFR	ON
BOOT_UP_SEQUENCE	NVCM
CDONE_PORT	CDONE_ONLY
DONE_EX	OFF
▲ User Code	
UserCode Format	Binary
UserCode	00000000000000000000000000000000
▲ I2C Address	
I2C Address Size	7bit
I2C Address	0000000
▲ TRACEID	
Trace ID	00000000
▲ CUSTOM_IDCODE	
Custom IDCode Format	Binary
Custom IDCode	00000000000000000000000000000000
▲ Derating	
VCCADPHY0	NOMINAL
VCCADPHY1	NOMINAL

Figure 6.1. sysCONFIG Preferences in Global Preferences Tab, Diamond Spreadsheet View

6.1. Configuration Mode and Port Options

The configuration and port options allow you to set which configuration ports continue to operate after the CrossLink device enters User Mode. You can also control the availability of status pins, as well as the speed at which configuration data is read from an external PROM. The selections made here are saved in the Feature Row and remain in effect until the Feature Row is erased. The only exception is the MCCLK_FREQ parameter, which is stored in the configuration data. The configuration and port options can be used in any combination.

Table 6.1. Configuration Mode/Port Options

Option Name	Default Setting	All Settings
SLAVE_SPI_PORT	ENABLE	DISABLE, ENABLE
MASTER_SPI_PORT	DISABLE	DISABLE, ENABLE
I2C_PORT	DISABLE	DISABLE, ENABLE
MCCLK_FREQ	2	See description below
TRANSFER	OFF	OFF, ON
CDONE_PORT	CDONE_USER_IO	CDONE_ONLY, CDONE_USER_IO

Slave SPI Port

The SLAVE_SPI_PORT allows you to preserve the Slave SPI configuration port after the CrossLink device enters User Mode. The SLAVE_SPI_PORT preference can be set in two states:

- **ENABLE** — This setting preserves the Slave SPI port I/O when the CrossLink device is in User Mode. When the pins are preserved, an external SPI master controller can interact with the Configuration Logic. Choosing ENABLE also prevents you from over-assigning I/O to the port pins.
- **DISABLE** — This setting disconnects the SPI port pins from the Configuration Logic. By itself it does not make the port pins general purpose I/O. Both SLAVE_SPI_PORT and MASTER_SPI_PORT must be set to disable to make the port pins general purpose I/O.

Master SPI Port

The MASTER_SPI_PORT allows you to preserve the Master SPI configuration port after the CrossLink device enters User Mode. The MASTER_SPI_PORT preference can be set in two states:

- **ENABLE** — This setting preserves the Master SPI port I/O when the CrossLink device is in User Mode. Choosing ENABLE also prevents you from over-assigning I/O to the port pins.
- **DISABLE** — This setting disconnects the SPI port pins from the Configuration Logic. By itself it does not make the port pins general purpose I/O. Both SLAVE_SPI_PORT and MASTER_SPI_PORT must be set to disable to make the port pins general purpose I/O.

I²C Port

The I2C_PORT allows you to preserve the I²C configuration port after the CrossLink device enters User Mode. The I2C_PORT preference can be set in two states:

- **ENABLE** — This setting preserves the I²C port I/O when the CrossLink device is in User Mode. Choosing ENABLE also prevents you from over-assigning I/O to the port pins.
- **DISABLE** — This setting disconnects the I²C port pins from the Configuration Logic. The port pins become general purpose I/O.

MCCLK Frequency

The MCCLK_FREQ preference allows you to alter the MCCLK frequency used to retrieve data from an external SPI Flash when using EXTERNAL or Dual Boot configuration modes. The MCCLK_FREQ value is stored in the incoming configuration data. It is not stored in the Feature Row. The CrossLink device reads a series of padding bits, a “start of data” word (0xBDB3) and a control register value. The control register contains the new MCCLK_FREQ value. CrossLink switches to the new clock frequency shortly after receiving the MCCLK_FREQ value.

The MCCLK_FREQ has a range of possible frequencies available from 2 MHz up to 48 MHz (see the list on page 13). Do not exceed the maximum clock rate of your SPI Flash, or of your printed circuit board.

Lattice recommends having a back-up configuration port available in case you specify a clock frequency that is out of specification.

ENABLE_TRANSFR

The TransFR function used by CrossLink requires the configuration data loaded into the configuration SRAM and any future configuration data file loaded into the external SPI Flash to have the ENABLE_TRANSFR set to the ENABLE state. See the [TransFR Operation](#) section and refer to TN1087, [Minimizing System Interruption During Configuration Using TransFR Technology](#) for more information about using TransFR with CrossLink.

6.2. Bitstream Generation Options

The Bitstream Generation options allow you to decide how the Diamond development tools create the configuration data for the CrossLink device. The CONFIGURATION, USERCODE, and CUSTOM_IDCODE settings are saved in the Feature Row and remain in effect until the Feature Row is erased. The other options allow you to control the BIT files that are generated by Diamond.

CONFIGURATION

The BOOT_UP_SEQUENCE preference allows you to control the boot up sequence. The BOOT_UP_SEQUENCE preference has five possible settings:

- **NVCM** — The NVCM setting is the SW default mode for building configuration data. The configuration bitstream is stored in the Configuration NVCM.
- **NVCM-EXT** — This setting boots up the system using the NVCM first. If an error occurs, the system boots up with the golden image in the External SPI Flash. This is the HW default mode for a blank device.
- **EXT-NVCM** — This setting boots up the system using the External SPI Flash first. If an error occurs, the system boots up with the golden image in the NVCM.
- **EXT-EXT** — This setting enables the user to use the external SPI Flash itself to store two configuration images. This setting boots up the system with the image kept at the first sector of the SPI Flash. If the system fails to boot with the first image, the system boots up with the second golden image at the second sector.
- **EXTERNAL** — This preference generates configuration data that is stored in an external memory.

USERCODE

The CrossLink Configuration NVCM sector contains a 32-bit register for storing a user-defined value. The default value stored in the register is 0x00000000. The USERCODE preference allows you to assign any value to a register. Suggested uses include the configuration data version number, a manufacturing ID code, and date of assembly among others.

The format of the USERCODE field is controlled using the USERCODE_FORMAT preference. Data entry can be performed in either Binary, Hex, or ASCII formats.

USERCODE_FORMAT

The USERCODE_FORMAT preference sets the format for the data field used to assign a value in the USERCODE preference. The USERCODE_FORMAT has three options:

- **Binary** — USERCODE is set using 32 '1' or '0' characters.
- **Hex** — USERCODE is set using eight hexadecimal digits (that is 0-9A-F)
- **ASCII** — USERCODE is set using up to four ASCII characters

CDONE_PORT

CDONE and a GPIO are bonded to a shared package ball. This option enables the feature in the software.

If CDONE_PORT = CDONE_ONLY, the GPIO becomes unavailable to the user. If CDONE_PORT = CDONE_USER_IO, the GPIO becomes available to the user.

DONE_EX

If ON, the device waits for CDONE to be driven high by an external signal to wake up. A common reason for keeping CDONE driven low is to allow multiple FPGAs to be completely configured. As each FPGA reaches the DONE state, it is ready to begin operation. The last FPGA to configure can cause all FPGAs to start in unison.

If the default setting is OFF, the CDONE pin is asserted by the device when the internal DONE bit is set. When set to OFF, the device wakes up regardless of the external signal state on CDONE.

6.3. Security Options

Security Options allows you to select from a range of options for tracking or securing the CrossLink device. [Table 6.2](#) provides a summary of these options.

Table 6.2. Configuration Mode/Port Options

Option Name	Default Setting	All Settings
TRACEID	<all zero>	8-bit arbitrary
MY_ASSP	OFF	OFF, ON
SECURITY_NVCM	DISABLE	DISABLE, ENABLE
SECURITY_SRAM	DISABLE	DISABLE, ENABLE
ONE_TIME_PROGRAM_SRAM	DISABLE	DISABLE, ENABLE
ONE_TIME_PROGRAM_NVCM	DISABLE	DISABLE, ENABLE

TRACEID

TraceID stamps each CrossLink with a unique 64-bit ID. No two CrossLink devices have the same TraceID value even when they are loaded with the same configuration data. This differs from a USERCODE which is present in the configuration data. Every device that receives the configuration data using a USERCODE receives the same USERCODE value.

The TraceID is 64 bits long with the least significant 56 bits being immutable data. The 56 bits are a combination of the wafer lot, the wafer number and the X/Y coordinates locating the die on the wafer. The most significant eight bits are provided by you and are stored in the Feature Row. The TraceID is changed using the Diamond Spreadsheet View. You enter a unique 8-bit binary value in the TraceID field and generate configuration data.

MY_ASSP

Every Lattice device has its own identification code identifying the device family, device density, and other parameters (for example, voltage, device stepping, and so on). The code is accessible from any CrossLink configuration port. The value stored in the IDCODE register allows you to uniquely identify a Lattice device.

The MY_ASSP preference permits you to change the value returned when the IDCODE is read from the FPGA. Turning the MY_ASSP ON enables the CUSTOM_IDCODE preference.

CUSTOM_IDCODE

The CUSTOM_IDCODE is the value you assign to override the default IDCODE in the CrossLink device. You are only allowed to enter a 32-bit hexadecimal or binary value when the MY_ASSP preference is ON.

Overriding the IDCODE prevents the Lattice programming software from being able to identify the CrossLink device, and as a result, prevents Programmer from being able to directly program the CrossLink device. It is necessary to migrate to generating Serial Vector Format (SVF) files in order to program MY_ASSP enabled CrossLink devices.

CUSTOM_IDCODE_FORMAT

The CUSTOM_IDCODE_FORMAT preference selects the format for the data field used to assign a value in the CUSTOM_IDCODE preference. The CUSTOM_IDCODE_FORMAT has two options:

- Binary — CUSTOM_IDCODE is set using 32 '1' or '0' characters.
- Hex — CUSTOM_IDCODE is set using eight hexadecimal digits (that is 0-9A-F)

CONFIG_SECURE

When this preference is set to ON, the read-back of the SRAM memory or the NVCM are blocked. The CrossLink device cannot be read back. The CONFIG_SECURE fuse and the NVCM are erased in tandem. When the security fuses are reset, the device can be reprogrammed. This depends on the target block (SRAM or NVCM). CONFIG_SECURE only prevents readback for the targeted block.

ONE_TIME_PROGRAM

CrossLink has One Time Programmable (OTP) fuses that can be used to prevent the SRAM from being erased or programmed.

7. Device Wake-up Sequence

When configuration is completed (the SRAM is loaded), the device wakes up in a predictable fashion. If the CrossLink device is the only or the last device in the chain, the Wake-up process begins when configuration is completed and the internal DONE bit is set.

7.1. Wake-up Signals

Three internal signals, GSR, GWDIS, and GOE, determine the Wake-up sequence.

- GSR is used to set and reset the core of the device. GSR is asserted (low) during configuration and de-asserted (high) in the Wake-up sequence.
- When the GWDIS signal is low, it safeguards the integrity of the RAM Blocks and LUTs in the device. This signal is low before the device wakes up. This control signal does not control the primary input pin to the device but controls specific control ports of EBR and LUTs.
- When low, GOE prevents the device's I/O buffers from driving the pins. The GOE only controls **output** pins. When the internal DONE is asserted CrossLink responds to input data.
- When high, the CDONE pin indicates that configuration is complete and that no errors are detected.

References

For more information, refer to the following documents:

- FPGA-DS-02007, [CrossLink Family Data Sheet](#)
- FPGA-TN-02012, [CrossLink High-Speed I/O Interface](#)
- FPGA-TN-02013, [CrossLink Hardware Checklist](#)
- FPGA-TN-02015, [CrossLink sysCLOCK PLL/DLL Design and Usage Guide](#)
- FPGA-TN-02016, [CrossLink sysI/O Usage Guide](#)
- FPGA-TN-02017, [CrossLink Memory Usage Guide](#)
- FPGA-TN-02018, [Power Management and Calculation for CrossLink Devices](#)
- FPGA-TN-02019, [CrossLink I2C Hardened IP Usage Guide](#)
- FPGA-TN-02020, [Advanced CrossLink I2C Hardened IP Reference Guide](#)
- TN1087, [Minimizing System Interruption During Configuration Using TransFR Technology](#)
- [Programming Tools User Guide](#)

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Revision History

Date	Version	Change Summary
February 2017	1.1	<ol style="list-style-type: none">1. Updated Configuration Ports Default Behavior and Arbitration section with default behavior2. Updated Configuration section with two cases3. Added Note 2 to Table 4.4. Default State in Diamond for each Port4. Changed CCLK to "MCK or SPI_SCK", MCK to CSN, and general purpose I/O to SPI_SS in Master and Slave SPI Configuration Port Pins section5. Changed MCLK to MCK in Master SPI Configuration Mode section6. Added reference to Programming Tools User Guide
August 2016	1.0	Updated document numbers, the previous document number was TN1303.
May 2016	1.0	First preliminary release.



7th Floor, 111 SW 5th Avenue
Portland, OR 97204, USA
T 503.268.8000
www.latticesemi.com