Minimizing System Interruption During Configuration Using TransFR Technology

Technical Note

FPGA-TN-02198-4.4

October 2023

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
FPGA	Field Programmable Gate Array
GPIO	General Purpose Input/Output
JTAG	Joint Test Action Group
NVCM	Non-Volatile Configuration Memory
PLL	Phase-Locked Loop
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SVF	Serial Vector Format
TransFR	Transparent Field Reconfiguration

1. Introduction

One of the fundamental benefits of using an FPGA is the ability to reconfigure its functionality without removing the device from the system. A number of elaborate mechanisms to provide field updates have been implemented. Accessibility to the system FPGAs can be as simple as a direct cable connection or something as complex as remote access using wireless links or high-level communication protocols.

Current update methods generally require a significant disruption to the system during the configuration update. It is desirable to reduce or eliminate the downtime resulting from reconfiguration due to an update, especially for nonredundant and mission-critical equipment.

Lattice provides TransFR™ (Transparent Field Reconfiguration) Technology to help minimize system interruption. TransFR Technology support is provided in Lattice Diamond® Programmer and Radiant™ Programmer software. The devices supporting TransFR are listed in Table 1.1.

Table 1.1. Device Support for TransFR

Device Family	Non-Volatile Memory
CertusPro™-NX	External
CrossLink™-NX-33	External
CrossLinkU™-NX	External
ECP5™	External
ECP5-5G™	External
LatticeECP3™	External
LatticeECP2™ and LatticeECP2M™	External
LatticeXP2™	On-chip Flash
LatticeXP™	On-chip Flash
MachXO5™-NX	On-chip Flash
MachXO3LF™	On-chip Flash
MachXO2™	On-chip Flash
MachXO™	On-chip Flash

Note: Device-specific details are listed in the appendices at the end of this document.

2. Background Programming

Lattice non-volatile Flash FPGAs feature two sets of configuration storage. The SRAM contains the working configuration, and non-volatile Flash memory or NVCM retains the configuration for use as necessary. The contents of the Flash memory or NVCM can be loaded into SRAM automatically at power-up or at any desired time, replacing the need for external boot memory.

Devices without internal non-volatile storage configure their SRAM contents directly from an external device. Such a device might be an SPI serial Flash, a microprocessor, or an EEPROM.

Whether the non-volatile storage is on-chip or external, it can be programmed independently of the SRAM memory space and one can be modified while the other remains intact. One powerful use of this arrangement is programming of the non-volatile configuration memory, while the SRAM continues to operate uninterrupted. This is referred to as background programming.

3. Boundary Scan Control

Lattice FPGAs also feature a rich set of IEEE 1149.1 (Boundary Scan Test) capabilities, providing additional control when accessing the device through the ispJTAG™ port. Boundary scan cells can be sampled and preloaded, allowing controllable I/O behavior during programming or configuration.

4.   TransFR Technology

Minimizing system interruption is achieved through background programming capabilities and TransFR technology. The result is a process in which systems can be upgraded with very little disruption. Careful system design allows TransFR to be completely transparent to the application.

Lattice supports JTAG and non-JTAG mode TransFR operation, depending on device family. Refer to appendices at the end of the document for device specific implementation details. The JTAG port uses BSCAN cells to either capture the current state on the I/O, or to force the I/O to a known state. JTAG port controlled TransFR operations are the only TransFR sequences able to control the boundary scan cells. Non-JTAG mode TransFR operations are bitstream driven. Here are some advantages and disadvantages for each TransFR operation mode:

4.1.   JTAG Mode TransFR

Advantages:
- Allows you to release I/O at a user-defined time during TransFR
- Allows you to customize I/O value during TransFR

Disadvantages:
- JTAG port on board is required because the operation is JTAG command-based
- Operation is relatively complicated because it is command-based

The following is a detailed description of JTAG mode TransFR.
- Phase 1: Background Programming. The non-volatile memory (internal or external) is reprogrammed while the SRAM is running undisturbed, allowing the system to continue operating without any disruption.
- Phase 2: I/O states are captured and held or driven to a user-defined level using JTAG commands. The outputs retain these levels throughout the reconfiguration process. As far as the system is concerned, this effectively pauses the FPGA, keeping any critical control and status outputs in their desired states during the system update.
- Phase 3: While the I/O states remain under the control of boundary scan, JTAG commands are used to initiate the transfer of the new functionality from non-volatile memory to the SRAM configuration space. For device-specific implementation details, refer to Appendix A to Appendix I.

  After the SRAM is configured, the I/O settings are returned to those you specified. The GSR signal is asserted internally to place the device into a predictable state.

  After reconfiguration is completed and prior to the exit of boundary scan mode in Phase4, the internal device logic is actively interpreting input signals. This time period can be used for a number of purposes to allow for a custom reactivation. Common uses include:
  - Synchronization of PLLs to incoming clock sources
  - Manipulation of event counters and state machines into desired operational states
  - Ensuring status and error indicators are properly initialized
- Phase 4: I/O are released from boundary scan control and back to the new desired function. The internal logic seamlessly reassumes control of the I/O.

4.1.1. Design Considerations for JTAG Mode TransFR

The clock source controlling the JTAG state machine and boundary scan circuitry is often asynchronous to the system clocks. To maintain desired output levels, it is recommended that an *indicator* input be included in the design to force the internal logic to the desired state. This input can be asserted during the TransFR process to ensure that sequential design elements are initialized to their desired values.

An example of this, as shown in Figure 4.1, is a synchronous load function for a counter. Determining the number of clocks required to reach the target value and the synchronization with the JTAG clock is often not practical. Inclusion of a load function for the counter allows it to be placed in a desired state during the TransFR process for return to normal post-configuration device operation, as illustrated in Figure 4.2.

```
always@ (posedge clk or negedge rstn)
begin
  if (~rstn)
    cnt <= 32'h00000000;
  else begin
      if (loadn == 1'b0)
        cnt <= load_val;
      else
        if (cnten == 1'b1)
          cnt <= cnt + 1;
  end
end
```

Figure 4.1. Verilog Code Fragment for Synchronous Load

Note: loadn is the *indicator* signal.

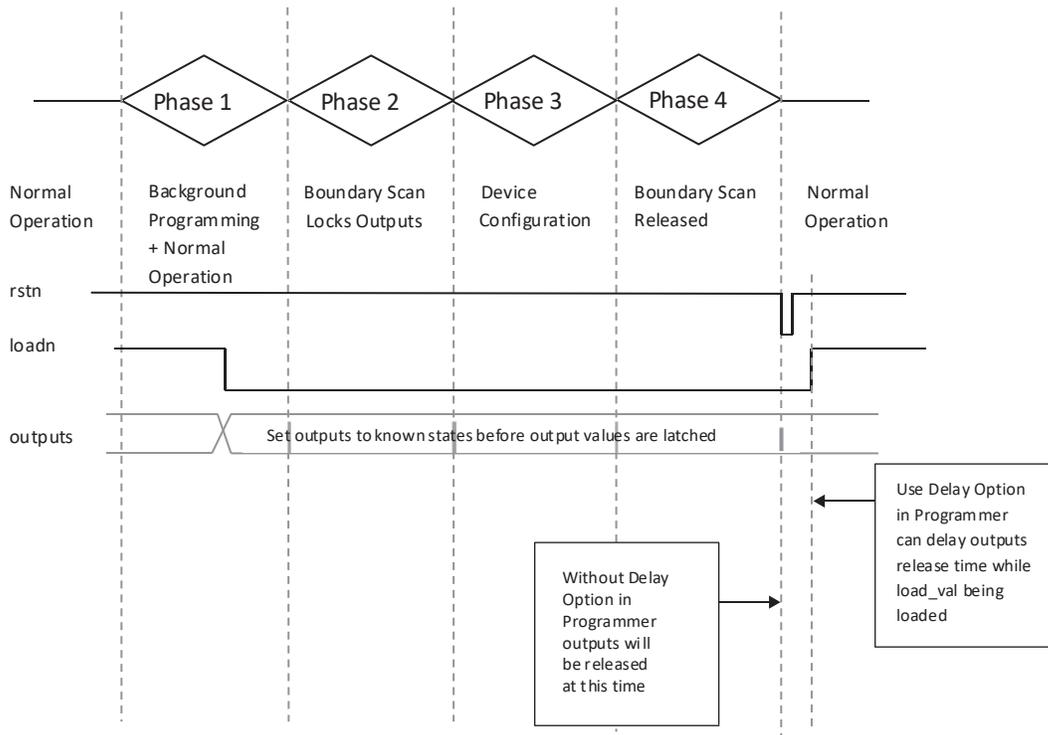**Figure 4.2. Example JTAG Mode TransFR Sequence**

4.2.   Non-JTAG Mode TransFR

Advantages:
- JTAG port is NOT required; any CONFIG port may be used
- Operation is relatively simple; most of these only need to refresh the device

Disadvantages:
- Does not allow you to release I/O at a specific time during TransFR
- Does not allow you to customize I/O value during TransFR. I/O values are always kept as is.

The following is a detailed description of non-JTAG mode TransFR:

- Phase 1: Background Programming. The non-volatile memory (internal or external) is reprogrammed while the SRAM is running undisturbed, allowing the system to continue operating without any disruption.
- Phase 2: Device starts to refresh. I/O states are captured into I/O latches and held. Outputs retain these levels throughout the reconfiguration process. As far as the system is concerned, this effectively pauses the FPGA, keeping any critical control and status outputs in their desired states during the system update.
- Phase 3: I/O remain under the control of I/O latches while the new functionality transfers from non-volatile memory to SRAM configuration space.
- Phase 4: I/O are released from I/O latch to user function only when GOE is released during wakeup stage.

4.2.1. Design Consideration for Non-JTAG Mode TransFR

In Non-JTAG Mode TransFR, I/O get released at the same time as Global Output Enable (GOE) signal get released during device wake up. To maintain all output states after device, wake up, resetting on the output registers needs to be avoided after wake up. In MachXO2 as an example, GOE signal get released before Global Set/Reset (GSR) signal. If design involves GSR, then asserting GSR could reset all the output registers.
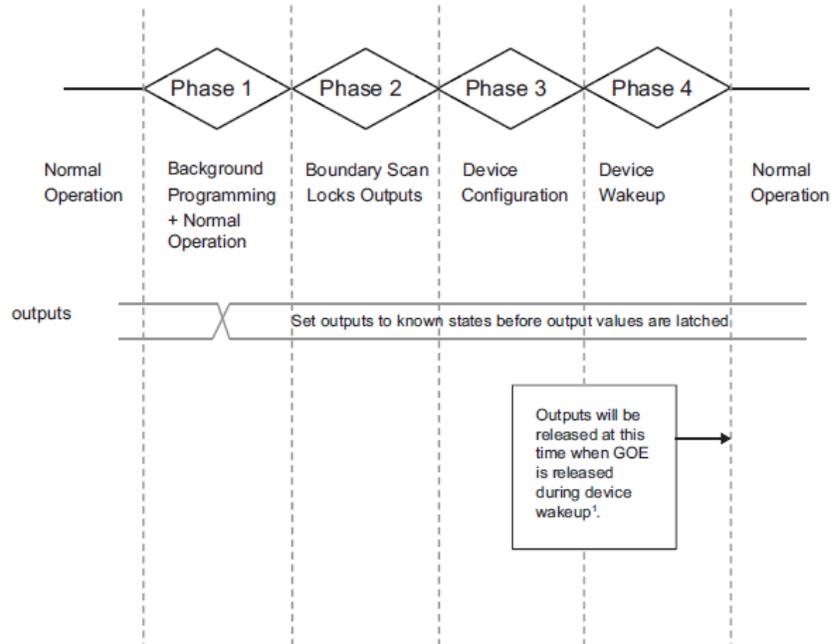

Figure 4.3. Example Non-JTAG Mode TransFR Sequence

For details on device wakeup, refer to the Wake-up section in MachXO2 Programming and Configuration Usage Guide (FPGA-TN-02155).

5. TransFR Using Diamond Programmer

The Diamond Programmer incorporates the TransFR flow for Lattice FPGAs in two operations. The first operation targets the background programming of the non-volatile memory, while the second controls the I/O states and initiates the configuration process.

Note: External configuration memory programming can also be accomplished through other means such as an onboard microprocessor. Provided that this does not disrupt the operation of the FPGA, the TransFR operation can be utilized.

For applicable devices, background programming is specified in Programmer from the *Operation* drop-down list in the Device Properties dialog box as shown in Figure 5.1.

Figure 5.1. Background Programming Selection in Programmer (MachXO2)

The Diamond Programmer also provides the capability to transparently program SPI serial Flash devices through the connected SRAM-based FPGA, as shown in Figure 5.2.
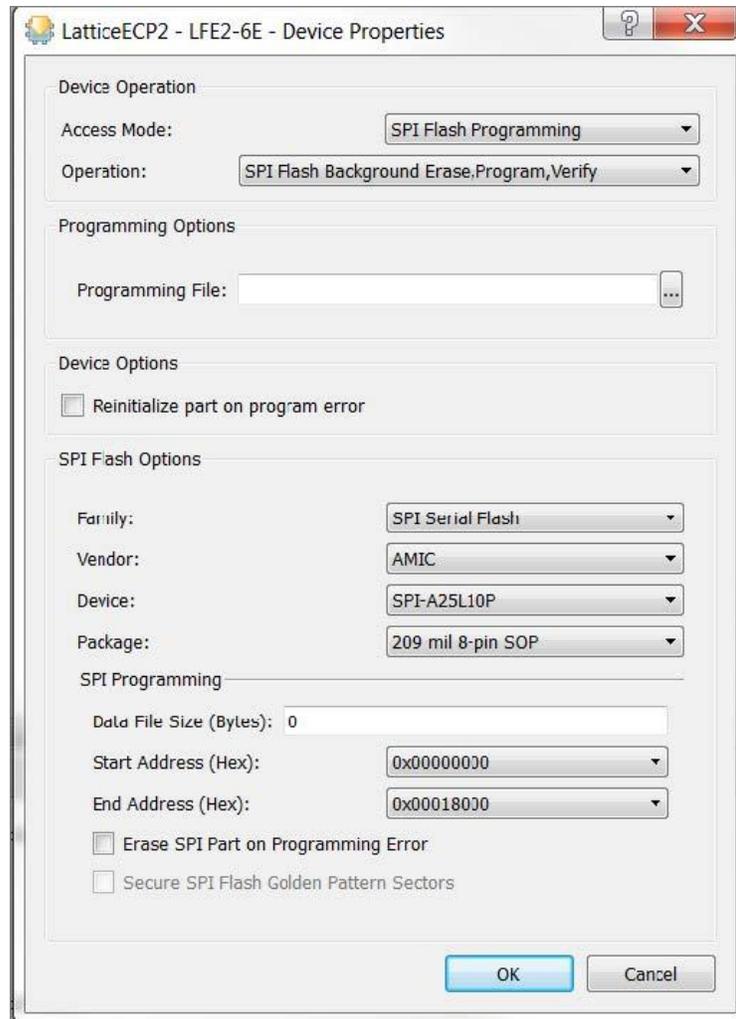
Figure 5.2. SPI Flash Programming Selection in Programmer (LatticeECP2)

The remaining portion of the TransFR procedure is selected by choosing a dedicated device operation, as shown in Figure 5.3. For device-specific details, refer to Appendix A to Appendix I.
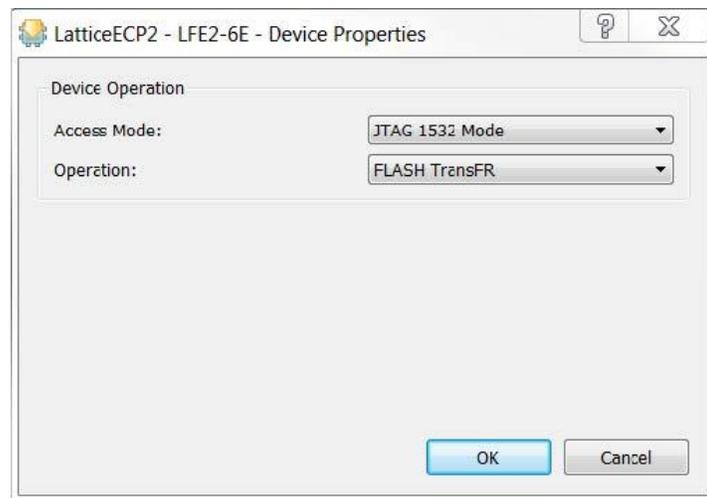

Figure 5.3. TransFR Operation Selection in Programmer (LatticeECP2)

6. TransFR Using Radiant Programmer

The Radiant Programmer provides the capability to transparently program SPI serial Flash devices through the connected SRAM-based FPGA. See specific device Appendix for Radiant Programmer configuration.

7. Embedded and Third-Party Support

The Diamond Embedded and Radiant Embedded source codes support TransFR Technology. The embedded instructions for TransFR can be exported from the Diamond embedded or Radiant embedded directory to a VME file for use with a microprocessor.

In addition to embedded routines, support is provided for the creation of standard SVF and ATE file formats for use outside of the Programmer environment from Deployment Tool.

Appendix A. LatticeXP

In the LatticeXP device family, the Flash to SRAM transfer occurs on the low-to-high transition of the PROGRAMN pin. To allow control of the PROGRAMN pin from the ispJTAG port, an external connection is required to a General Purpose I/O (GPIO) pin, as shown in Figure A.1. The operation is supported by the ispVM System software and is not supported by Programmer. LatticeXP devices are featured with JTAG mode TransFR.

Note: This GPIO pin should not be used for any other purpose than updating through the TransFR. During operational mode, a low on this pin results in undesired reconfiguration cycles.
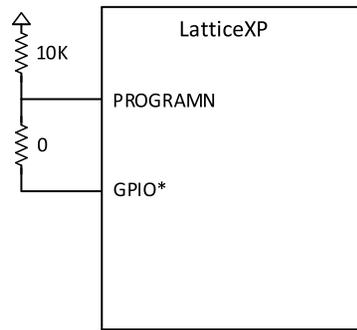
*This pin is any available user-specified I/O.

Figure A.1. PROGRAMN to GPIO Connection for LatticeXP

The location of this GPIO pin is user-selectable using ispVM System. Without specification, this pin is assigned to a default location, shown in Table A.1. Alternatively, ispVM System may also be configured to directly drive the PROGRAMN pin from the ispEN signal of the ispDOWNLOAD® cable. The PROGRAMN pin can also be asserted at the appropriate time by an external source, such as a microprocessor.

Table A.1. LatticeXP Default GPIO for PROGRAMN Control

LatticeXP Device	Pin Function	100 TQFP	144 TQFP	208 PQFP	256 fpBGA	388 fpBGA	484 fpBGA
XP3	PROGRAMN	3	1	3	—	—	—
	GPIO	8	9	14	—	—	—
XP6	PROGRAMN	—	1	3	C2	—	—
	GPIO	—	9	14	E2	—	—
XP10	PROGRAMN	—	—	—	C2	F4	—
	GPIO	—	—	—	D1	H4	—
XP15	PROGRAMN	—	—	—	C2	F4	F5
	GPIO	—	—	—	E2	H3	H3
XP20	PROGRAMN	—	—	—	C2	F4	F5
	GPIO	—	—	—	E2	H3	H3

To select the control source of the LatticeXP PROGRAMN pin, the device package must be specified when selecting the device. The *TransFR Options...* button in the *Device Information* dialog, then becomes available, as shown in Figure A.2.
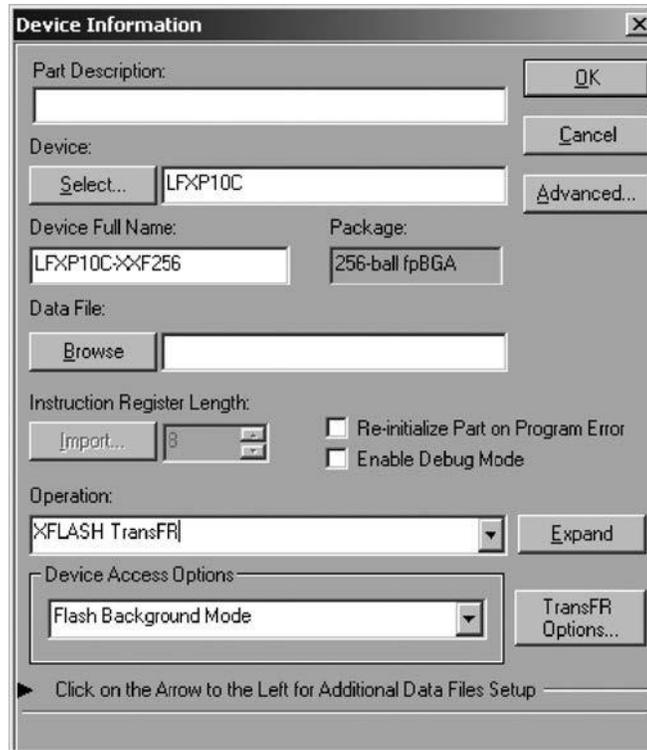
Figure A.2. Device Information Dialog with TransFR

The resulting dialog appears as shown in Figure A.3.

Figure A.3. TransFR Options Dialog
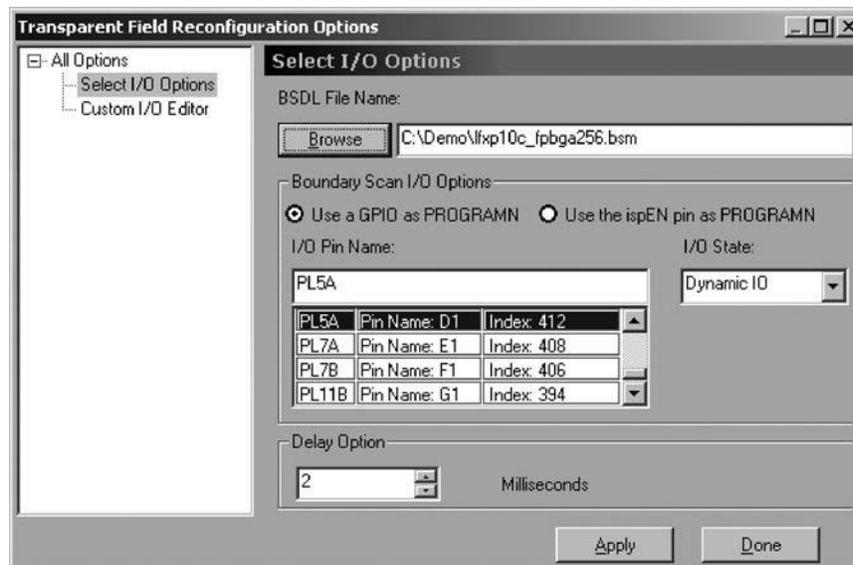
- BSDL File Name: A BSDL file must be supplied when specifying the state of I/O during the TransFR operation. BSDL files can be downloaded at www.latticesemi.com.
- I/O State: Provides control of the I/O behavior during the TransFR operation.
- Dynamic I/O: Specifies outputs are to be sampled and held to their last values (Leave-Alone), unless overridden using the Custom I/O Editor.

- Custom I/O: Outputs are tri-stated, unless overridden using the Custom I/O Editor.
- Delay Option: Specifies the delay (in milliseconds) from the start of reconfiguration of the SRAM from Flash to the release of the I/O.

To select a particular GPIO for connection with the PROGRAMN pin:

1. Select the Use a GPIO as PROGRAMN option.

2. Provide a BSDL file for the device. This can be downloaded at www.latticesemi.com.

3. Select the desired pin from the list.

To enable the ispEN output of the ispDOWNLOAD cable for control of PROGRAMN, choose the 'Use the ispEN pin as PROGRAMN' option. The 'PROG Pin Connected' option must also be enabled. This can be found from the menu as Options->Cable and I/O Port Setup.

The following device limitations apply from the time the boundary scan cells are locked in Phase2 until their release in Phase4:

- Open-drain outputs revert to standard, totem-pole drivers.
- Differential outputs are driven as independent single-ended drivers during boundary scan operations. When a differential output is captured and held, the positive (true) pin is driven to that logical value using an LVCMOS output driver. The negative (complement) pin of the differential pair is tri-stated. Additionally, during the brief configuration process the programmable I/O default to the following characteristics, while still under the control of boundary scan:
- Outputs are single-ended, totem-pole drivers with a maximum VOH of VCCIO and 8 mA drive strength.
- Internal weak pullup resistors are enabled.

Appendix B. MachXO

In the MachXO device family, Flash to SRAM configuration can be initiated by an instruction from the ispJTAG port. No additional external connections are required to facilitate TransFR capabilities. MachXO devices are featured with JTAG mode TransFR.

To specify TransFR in Programmer, choose the corresponding operation, as shown in Figure B.1.

Figure B.1. Selecting TransFR for MachXO

Edit the I/O state by right-clicking the device and selecting Edit I/O State as shown in Figure B.2 and Figure B.3.
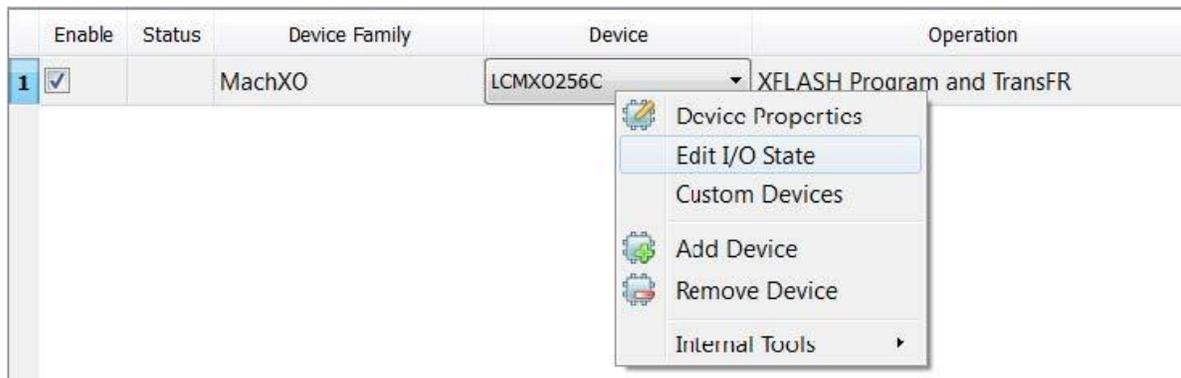

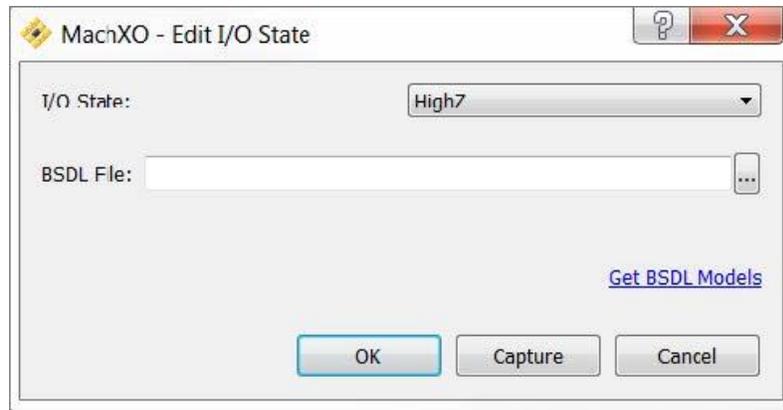**Figure B.2. Edit I/O State Menu Option**

Figure B.3. Edit I/O State Dialog Box

The I/O state provides control of the I/O behavior during the TransFR operation.
- HighZ – Specifies all outputs are tri-stated.
- All 1s – Specifies all outputs are high.
- All 0s – Specifies all outputs are low.
- Leave Alone – Specifies outputs are to be sampled and held to their last values.
- Dynamic I/O – Specifies outputs are to be sampled and held to their last values (Leave Alone), unless a value is explicitly specified using the Custom I/O Editor. A BSDL file is required for this option.
- Custom – Outputs are tri-stated, unless a value is explicitly specified using the Custom I/O Editor. A BSDL file is required for this option.
- TransFR Options – Brings up a Delay Option pop-up menu. Specifies the delay (in milliseconds) from the start of reconfiguration of the SRAM from Flash to the release of the I/O. (Temporarily not available in Programmer)

The following device limitations apply from the time the boundary scan cells are locked in Phase2 until their release in Phase4:
- Differential outputs are driven as independent single-ended drivers during boundary scan operations. When a differential output is captured and held, the positive (true) pin is driven to that logical value using an LVCMOS output driver. The negative (complement) pin of the differential pair is tri-stated.

Appendix C. MachXO2 and MachXO3LF

In the MachXO2 and MachXO3LF device families, Flash to SRAM configuration can be initiated by an instruction from any configuration port, not just from ispJTAG port. No additional external connections are required to facilitate TransFR capabilities. MachXO2 and MachXO3LF devices are featured with non-JTAG mode TransFR.

Note: The following examples show the steps for MachXO2 devices. The steps for MachXO3LF devices are essentially identical.

To enable TransFR, set ENABLE_TRANSFR to ENABLE through Global Preference, as shown in Figure C.1, in both patterns (the current one in the flash and the new one to be programmed into the flash).
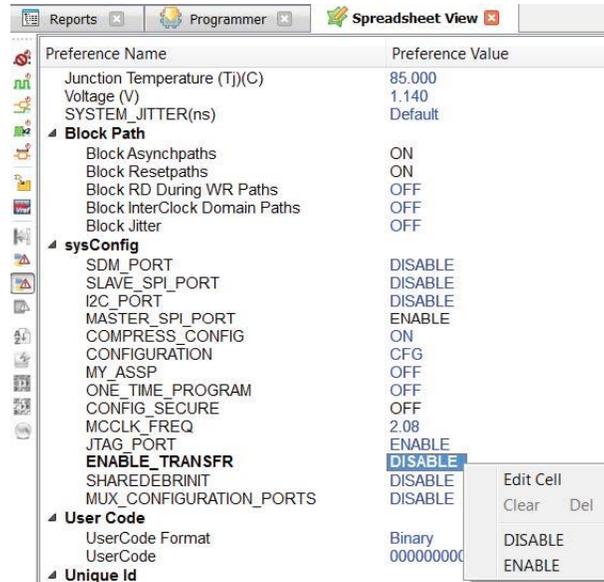

Figure C.1. Selecting TransFR for MachXO2

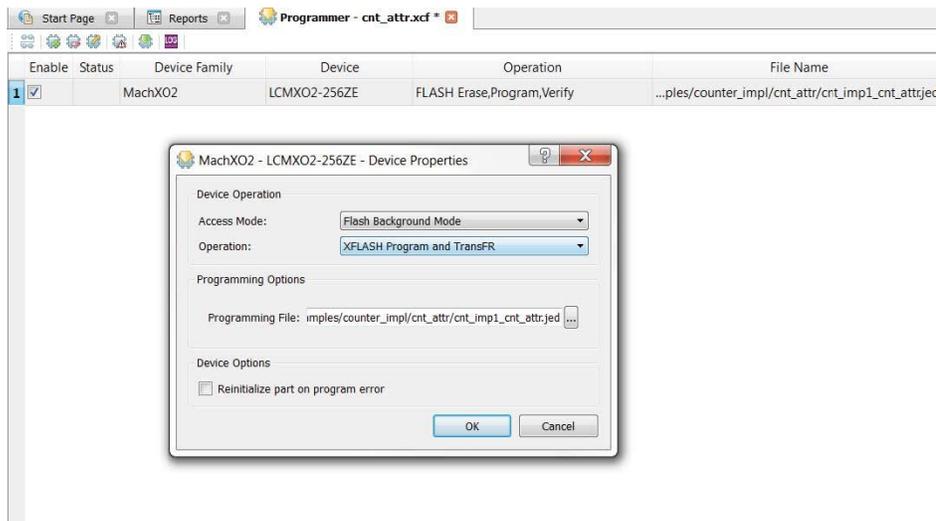To specify TransFR in Programmer, choose the corresponding operation in Figure C.2.

Figure C.2. Selecting TransFR for MachXO2

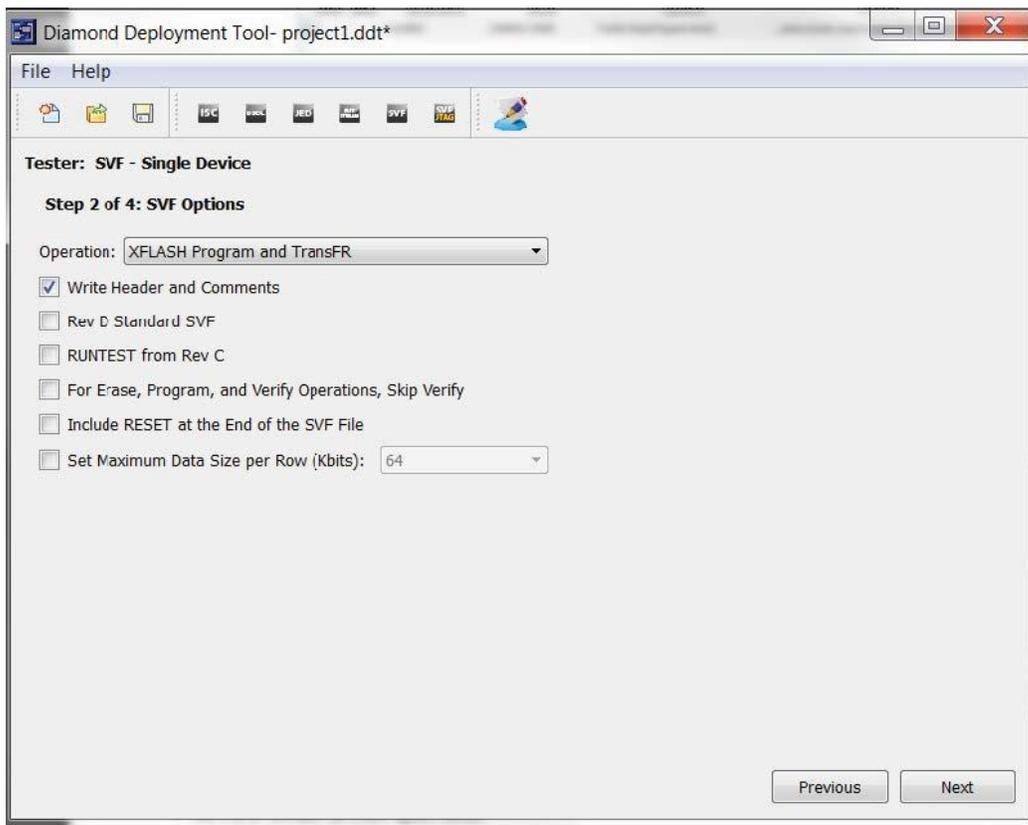To specify TransFR in Deployment Tool, choose the corresponding operation in Figure C.3.

Figure C.3. Selecting TransFR for MachXO2

The following are the key features of TransFR:
- I/O's are automatically sampled and held at their last state at the falling edge of INITN.
- All I/O's remain at their spec level.
- I/O refresh is automatic.
- If implemented, the hardened user SPI port outputs are disabled (tri-stated). (The configuration SPI port is not impacted.)

The following are the Limitations of TransFR:
- Whereas the MachXO allows you to release the TransFR I/O at a specific time through a BSCAN operation, the MachXO2 and MachXO3LF do not support this ability.
- Whereas the MachXO allows you to preload specific values into I/O through BSCAN operation during TransFR, the MachXO2 and MachXO3LF do not support this ability.

Appendix D. LatticeECP2/M

The LatticeECP2/M device family of SRAM-only FPGAs relies on non-volatile external memory to store configuration data. TransFR allows configuration through the sysCONFIG port while the I/O pins remain in a locked state for minimizing system disruption.

Note: The dual-purpose I/O pins in bank 8 should not be used for signals critical to the system during TransFR. These pins are not controlled by boundary scan in phases 2 through 4 to allow full operation of the sysCONFIG port. If background programming of the SPI Flash from the LatticeECP2/M is desired, these pins cannot be used as I/O pins since the PERSISTENT preference must be set to ON.

The LatticeECP2/M supports a number of configuration sources, such as SPI serial Flash and parallel modes. All sysCONFIG modes are supported with TransFR. Refer to LatticeECP2/M sysCONFIG Usage Guide (TN1108) for the details and connection requirements for each mode.

Below is an example of a JTAG mode TransFR procedure.

To enable TransFR for the LatticeECP2/M, a preference in Diamond must be set. This can be done using the Global tab in the Spreadsheet View, as shown in Figure D.1. To enable TransFR, the ENABLE_NDR option must be set to ON.
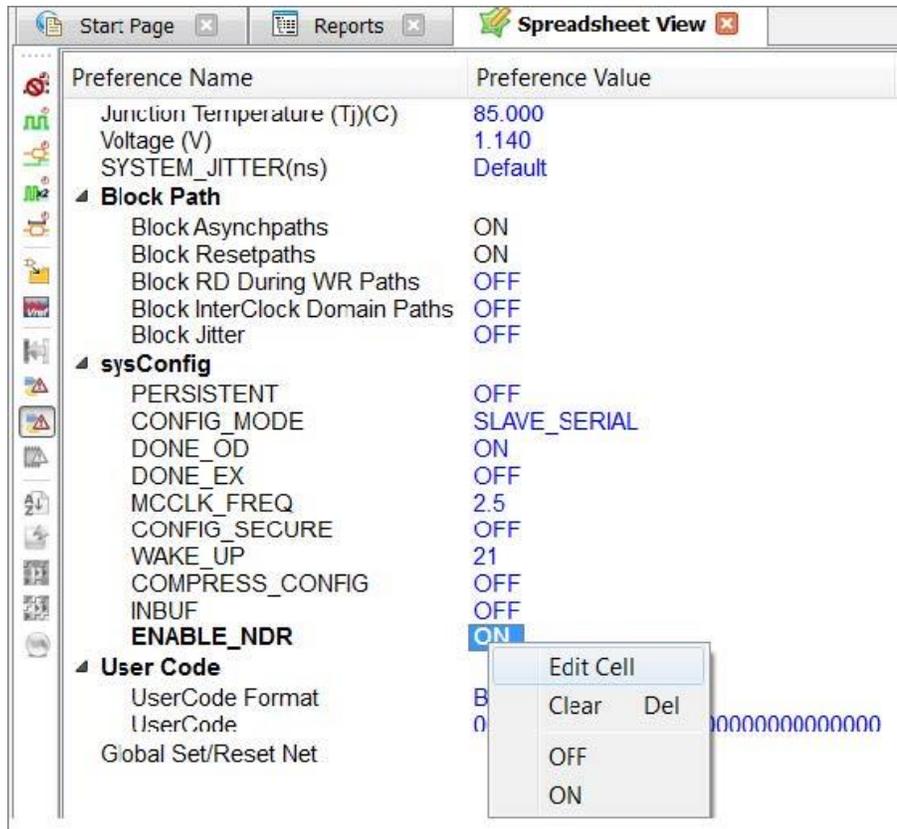

Figure D.1. Enabling TransFR in Diamond

The reconfiguration process for TransFR is initiated through the commands in the ispJTAG port. To specify TransFR in Programmer, choose the corresponding option, as shown in Figure D.2.
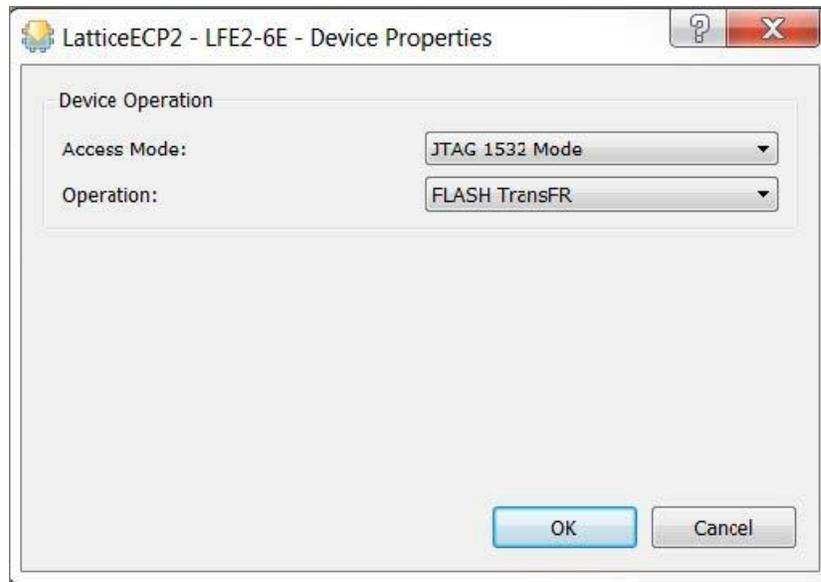
Figure D.2. Selecting TransFR for LatticeECP2

Edit the I/O state by right-clicking the device and selecting **Edit I/O** State as shown in Figure D.3 and Figure D.4.
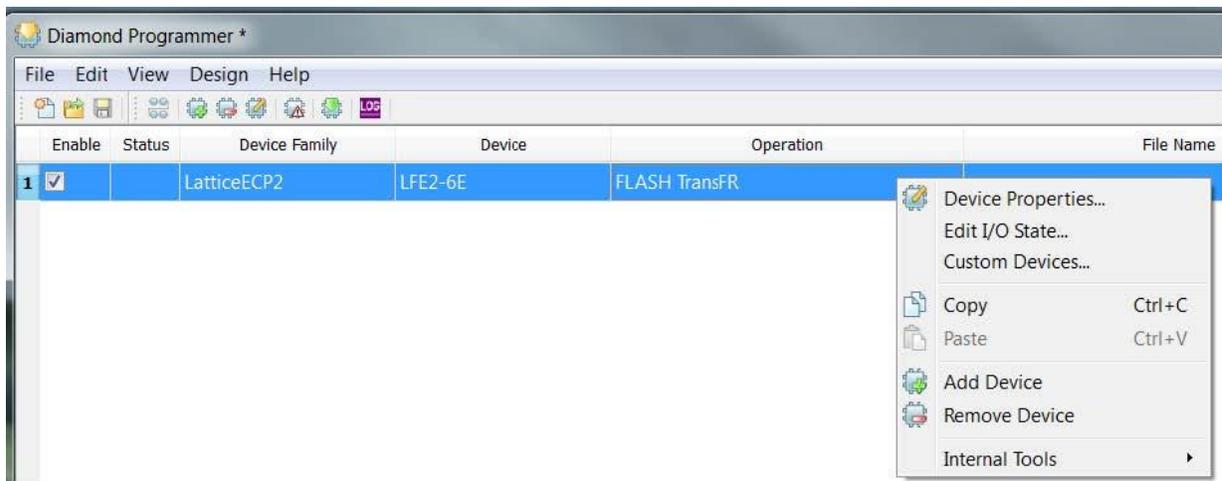

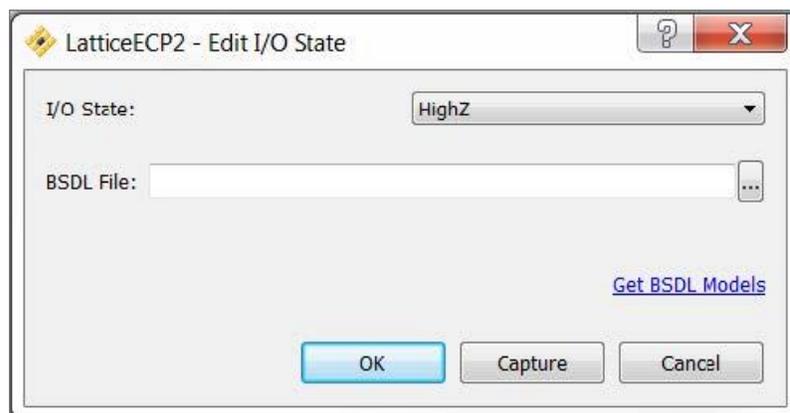**Figure D.3. Edit I/O State Menu Option**

Figure D.4. Edit I/O State Dialog Box

The I/O state provides control of the I/O behavior during the TransFR operation.

- HighZ – Specifies all outputs are tri-stated.
- All 1s – Specifies all outputs are high.
- All 0s – Specifies all outputs are low.
- Leave Alone – Specifies outputs are to be sampled and held to their last values.
- Dynamic I/O – Specifies outputs are to be sampled and held to their last values (Leave Alone), unless a value is explicitly specified using the Custom I/O Editor. A BSDL file is required for this option.
- Custom – Outputs are tri-stated, unless a value is explicitly specified using the Custom I/O Editor. A BSDL file is required for this option.
- TransFR Options – Brings up a Delay Option pop-up menu. Specifies the delay (in milliseconds) from the start of reconfiguration of the SRAM from Flash or NVCM to the release of the I/O.

Appendix E. LatticeXP2

The LatticeXP2 device family supports JTAG mode TransFR from the on-chip Flash memory.

The reconfiguration process for TransFR is initiated through the commands in the ispJTAG port. To specify TransFR in Programmer, choose the corresponding option, as shown in Figure E.1.
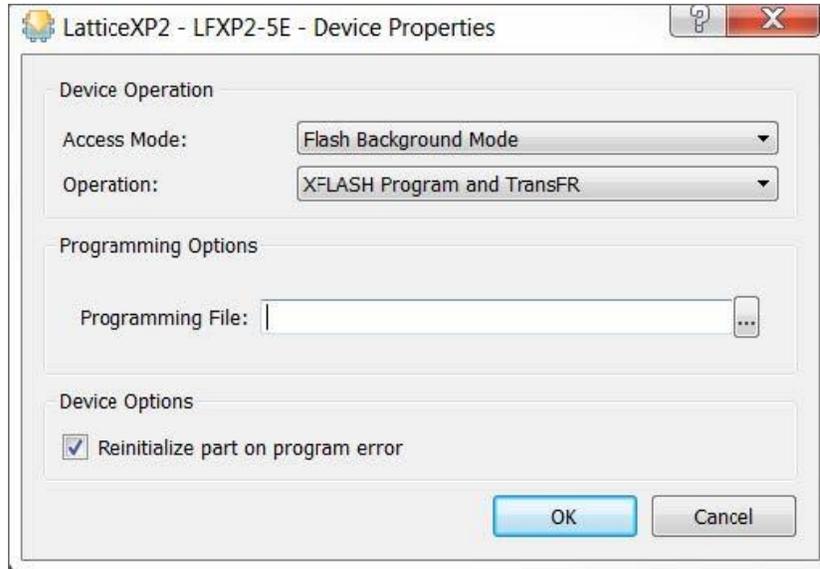

Figure E.1. Selecting TransFR for LatticeXP2

Edit the I/O state by right-clicking the device and selecting Edit I/O State as shown in Figure E.2 and Figure E.3.

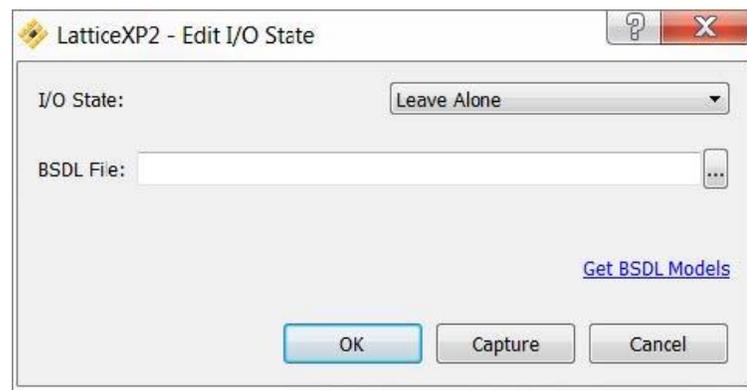**Figure E.2. Edit I/O State Menu Option**

Figure E.3. Edit I/O State Dialog Box

The I/O state provides control of the I/O behavior during the TransFR operation.

- HighZ – Specifies all outputs are tri-stated.
- All 1s – Specifies all outputs are high.
- All 0s – Specifies all outputs are low.
- Leave Alone – Specifies outputs are to be sampled and held to their last values.
- Dynamic I/O – Specifies outputs are to be sampled and held to their last values (Leave Alone), unless a value is explicitly specified using the Custom I/O Editor. A BSDL file is required for this option.
- Custom – Outputs are tri-stated, unless a value is explicitly specified using the Custom I/O Editor. A BSDL file is required for this option.
- TransFR Options – Brings up a Delay Option pop-up menu. Specifies the delay (in milliseconds) from the start of reconfiguration of the SRAM from Flash to the release of the I/O.

Appendix F. LatticeECP3

The LatticeECP3 device family of SRAM-only FPGAs relies on non-volatile external memory to store configuration data. TransFR allows configuration through the sysCONFIG port, while the I/O pins remain in a locked state for minimizing system disruption.

Note: The dual-purpose I/O pins in bank 8 should not be used for signals critical to the system during TransFR. These pins are not controlled by boundary scan in phases 2 through 4 to allow full operation of the sysCONFIG port. If background programming of the SPI Flash from the LatticeECP3 is desired, these pins cannot be used as I/O pins since the PERSISTENT preference must be set to ON.

The LatticeECP3 supports a number of configuration sources, such as SPI serial Flash and parallel modes. All sysCONFIG modes are supported with TransFR. Refer to LatticeECP3 sysCONFIG Usage Guide (FPGA-TN-02192) for the details and connection requirements for each mode.

Below is an example of a non-JTAG mode TransFR procedure.

To enable TransFR for the LatticeECP3, a preference in Diamond must be set. This can be done using the Global tab in the Spreadsheet View, as shown in Figure F.1. To enable TransFR, the ENABLE_NDR option must be set to ON.
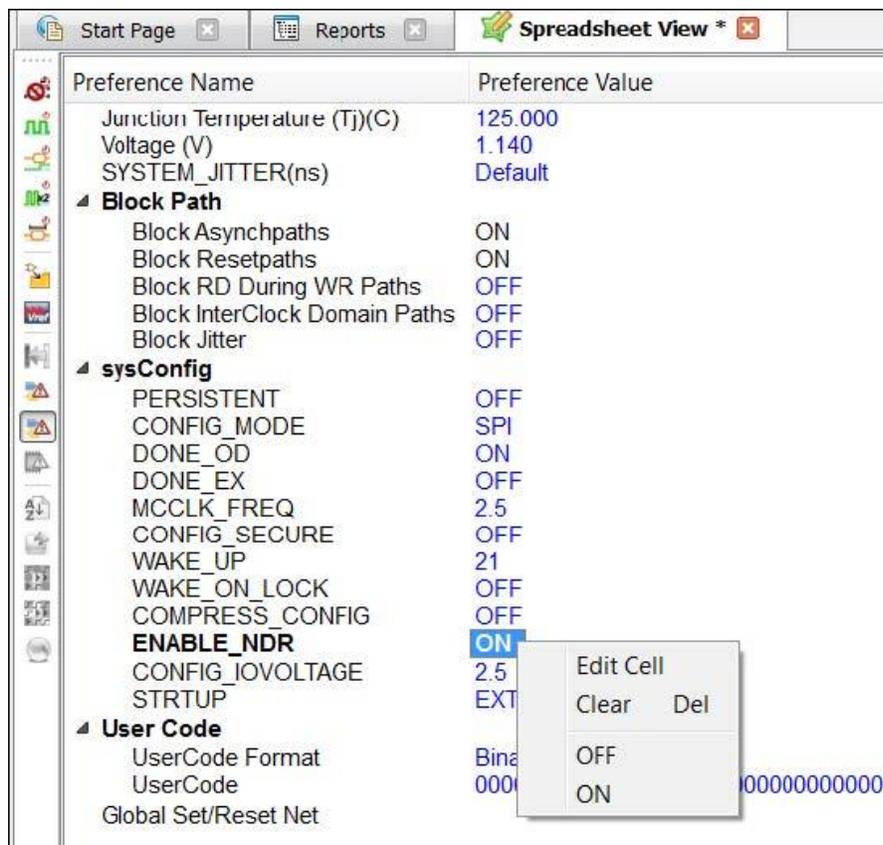

Figure F.1. Enabling TransFR in Diamond

The reconfiguration process for TransFR is initiated through the refresh command or by toggling PROGRAMN pins. To specify TransFR in Programmer, choose the corresponding option, as shown in Figure F.2.
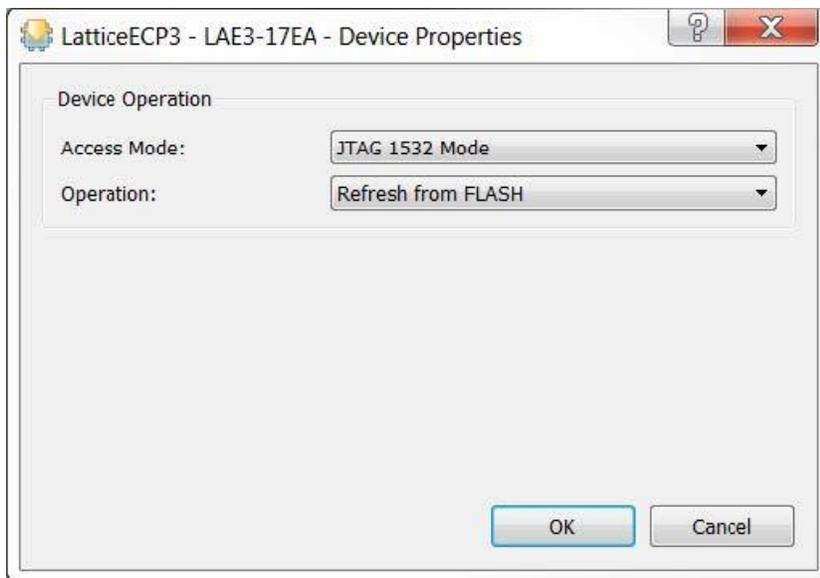
Figure F.2. Selecting TransFR for LatticeECP3

Appendix G. ECP5 and ECP5-5G

The ECP5 and ECP5-5G families of SRAM-only FPGAs rely on non-volatile external memory to store configuration data. TransFR allows configuration through the sysCONFIG port while the I/O pins remain in a locked state for minimizing system disruption.

Note: The dual-purpose I/O pins in bank 8 should not be used for signals critical to the system during TransFR. These pins are not controlled by boundary scan in phases 2 through 4 to allow full operation of the sysCONFIG port. If background programming of the SPI Flash from the ECP5 and ECP5-5G devices is desired, these pins cannot be used as I/O pins since the PERSISTENT preference must be set to ON.

The ECP5 and ECP5-5G devices support both JTAG mode and non-JTAG TransFR modes.

Below is an example of a non-JTAG mode TransFR procedure. To enable TransFR for the ECP5 and ECP5-5G devices, a preference in Diamond must be set. This can be done using the Global tab in the Spreadsheet View, as shown in Figure G.1. To enable TransFR, the TRANSFR option must be set to ON.
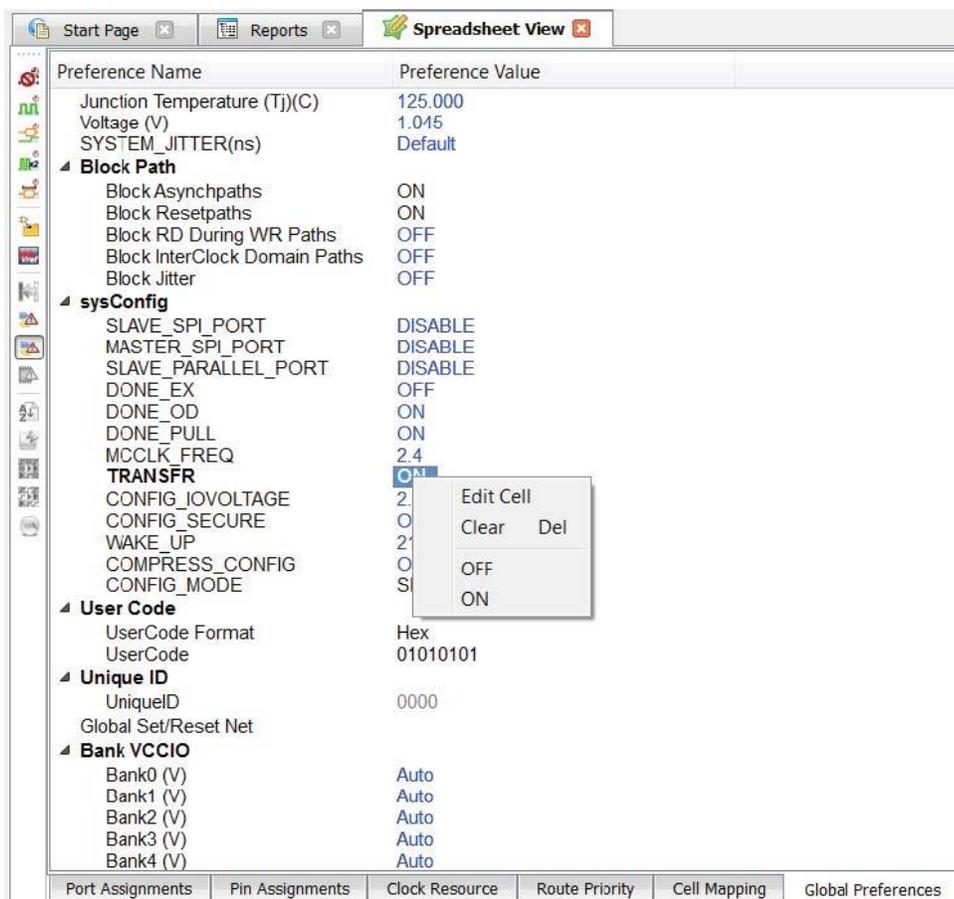

Figure G.1. Enabling TransFR in Diamond

The reconfiguration process for TransFR is initiated through the refresh command or by toggling PROGRAMN pins. To specify TransFR in Programmer, choose the corresponding option, as shown in Figure G.2.
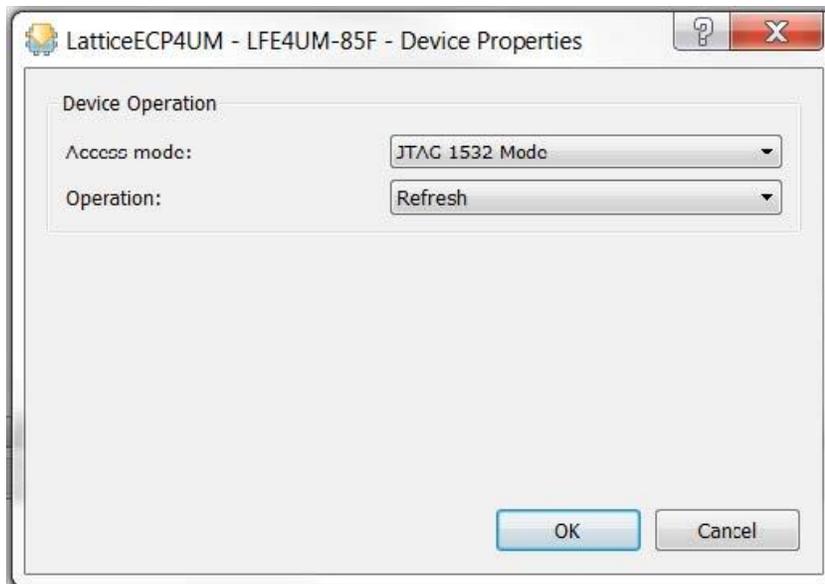
Figure G.2. Selecting TransFR for the ECP5 and ECP5-5G Devices

Appendix H. CertusPro-NX, CrossLink-NX-33, or CrossLinkU-NX

TransFR preference for CertusPro-NX, CrossLink-NX-33, or CrossLinkU-NX devices can be enabled/disabled through the Lattice Radiant software. The TRANSFR preference is listed in the Device Constraint Editor (Figure H.1) under the Global tab. This needs to be enabled in the new bitstream pattern that is going to be programmed into the external Flash.
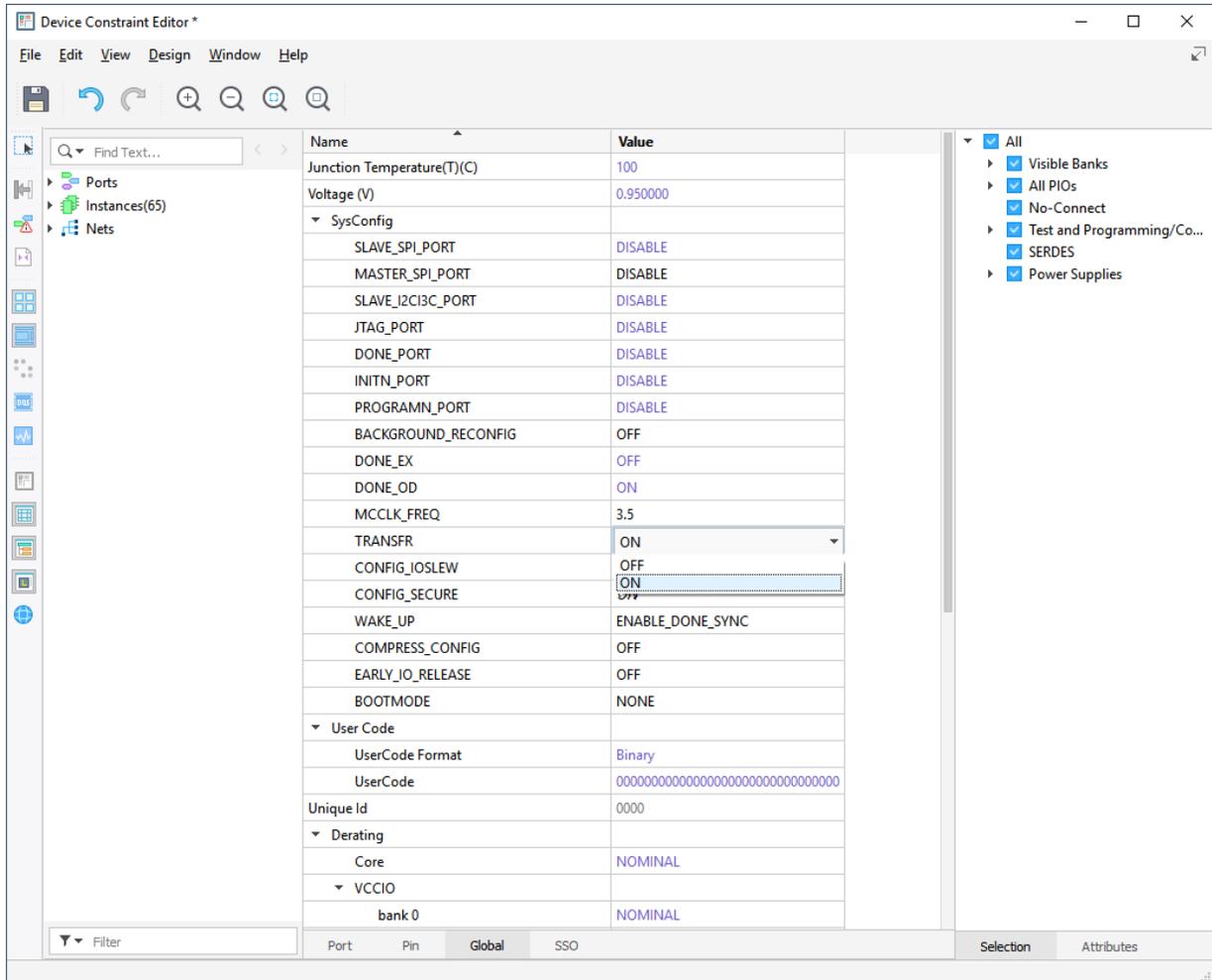

Figure H.1. Lattice Radiant Software: Device Constraint Editor – TRANSFR

Once the TransFR bitstream is generated, it needs to be programmed into the external Flash. This is done through the Radiant Programmer. Figure H.2 shows the configuration of an external Macronix SPI Flash using the JTAG port. Figure H.3 shows the configuration of an external Macronix SPI Flash using the SPI port. The reconfiguration process for TransFR is initiated through the refresh command or by toggling PROGRAMN pin.
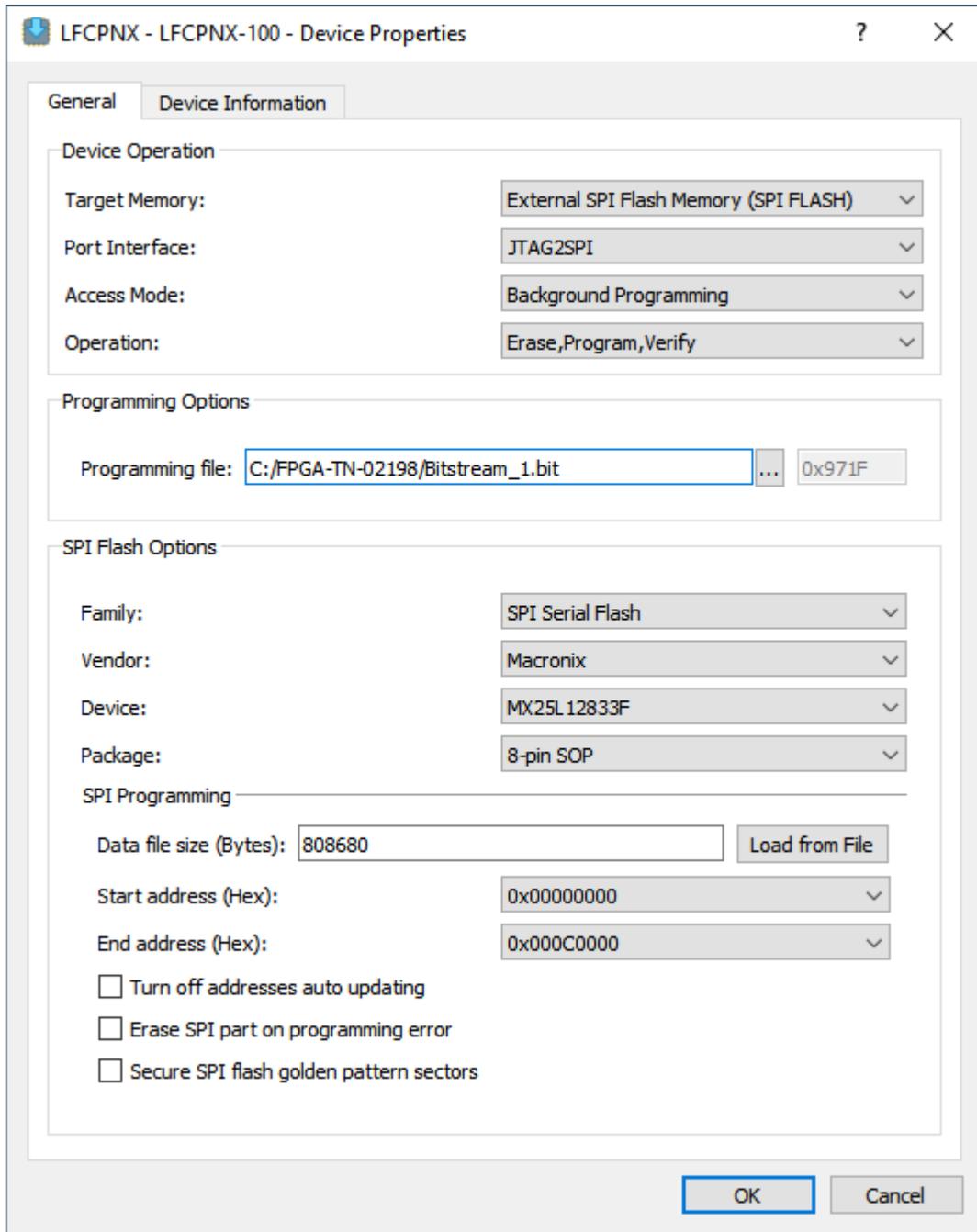
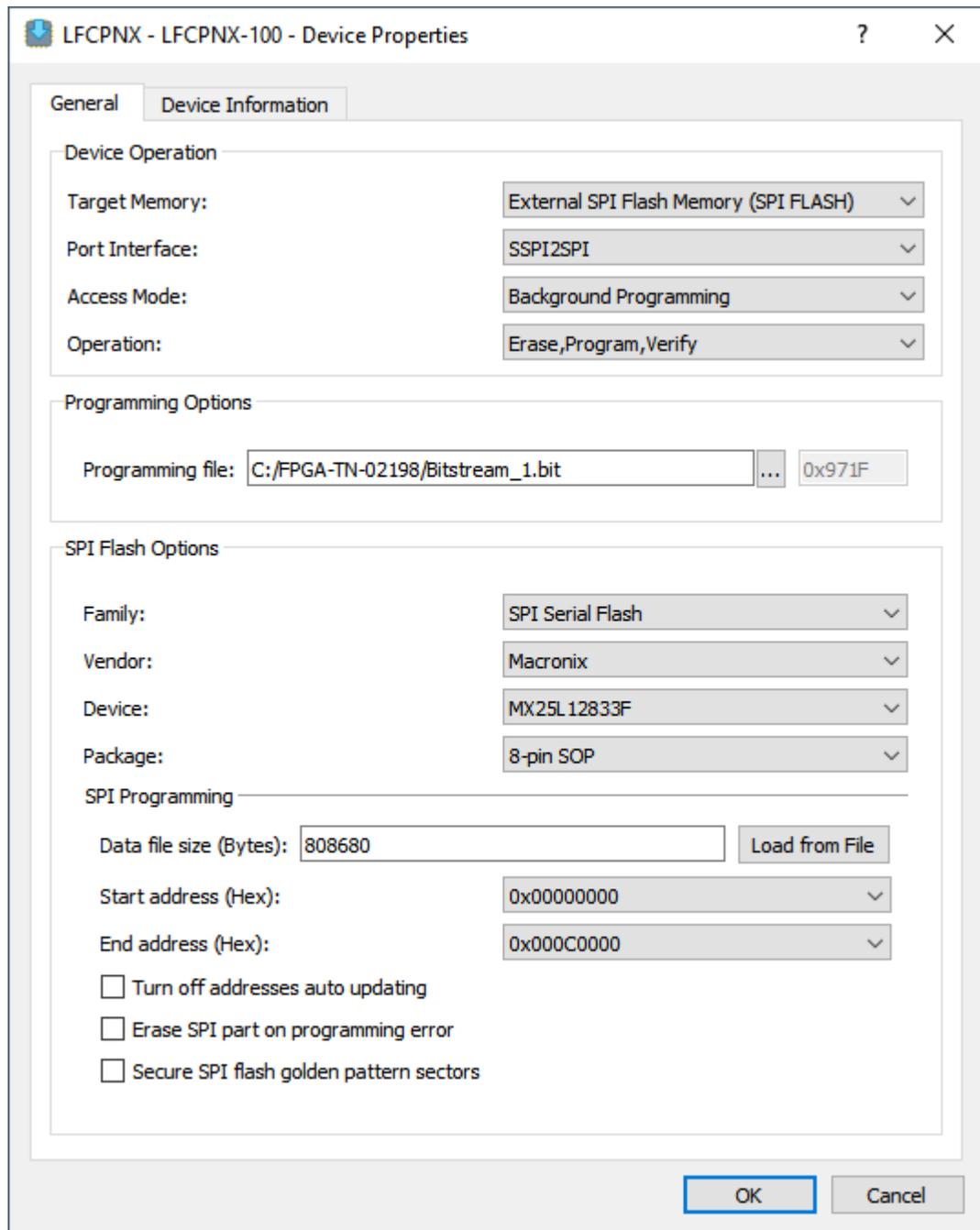**Figure H.2. Radiant Programmer – External SPI Flash Programming through JTAG**

Figure H.3. Radiant Programmer – External SPI Flash Programming through SPI

Appendix I. MachXO5-NX

TransFR preference for MachXO5-NX devices can be enabled/disabled through the Lattice Radiant software. The TRANSFR preference is listed in the Device Constraint Editor (Figure I.1) under the Global tab. This needs to be enabled in the new bitstream pattern that is going to be programmed into the on-chip Flash.
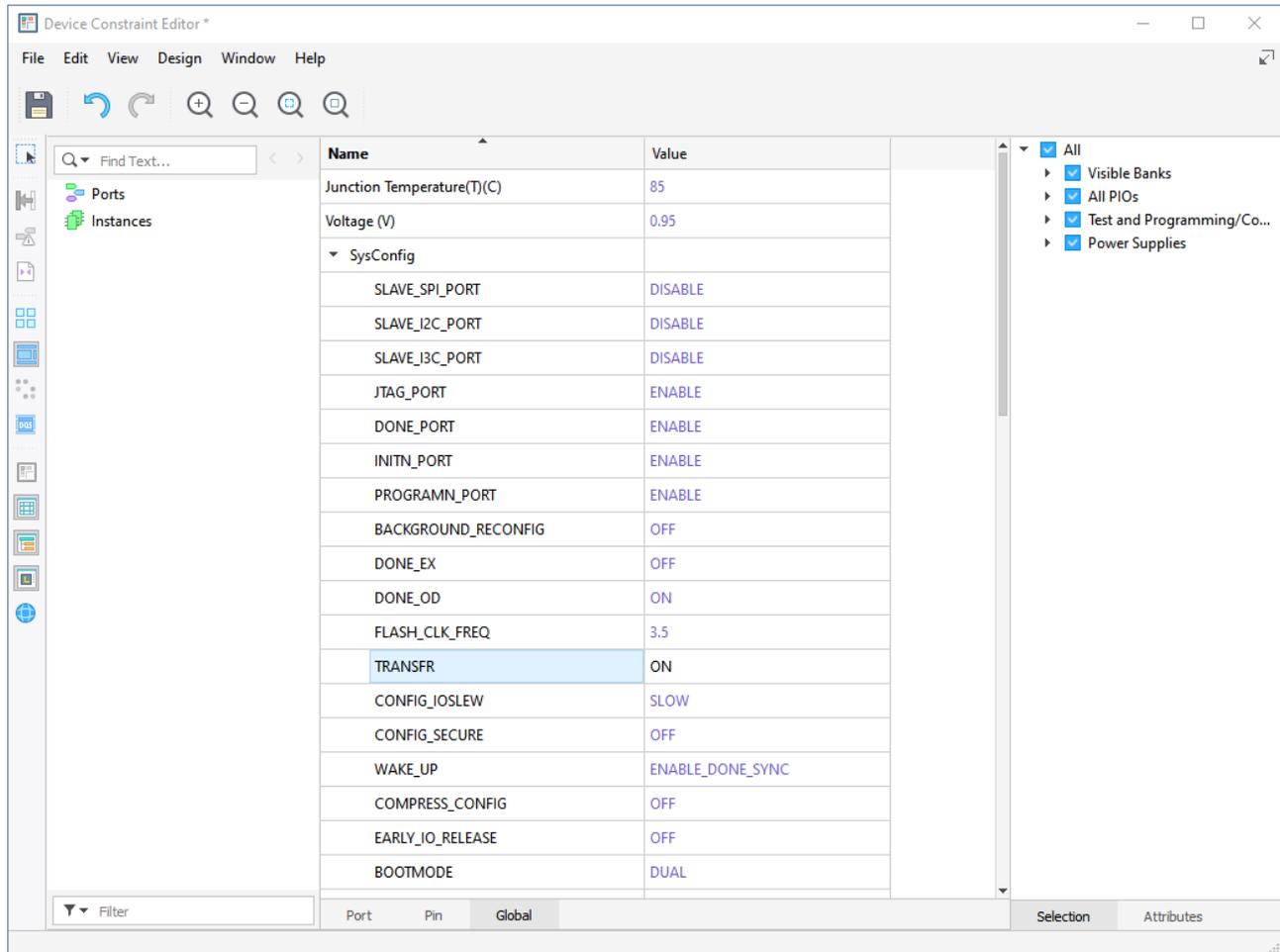
Figure I.1. Lattice Radiant Software: Device Constraint Editor – TRANSFR

Once the TransFR bitstream is generated, it needs to be programmed into the on-chip Flash. This is done through the Radiant Programmer. Figure I.2 shows the configuration of the Flash background programming mode using the JTAG port. The reconfiguration process for TransFR is initiated through the refresh command or by toggling PROGRAMN pin.
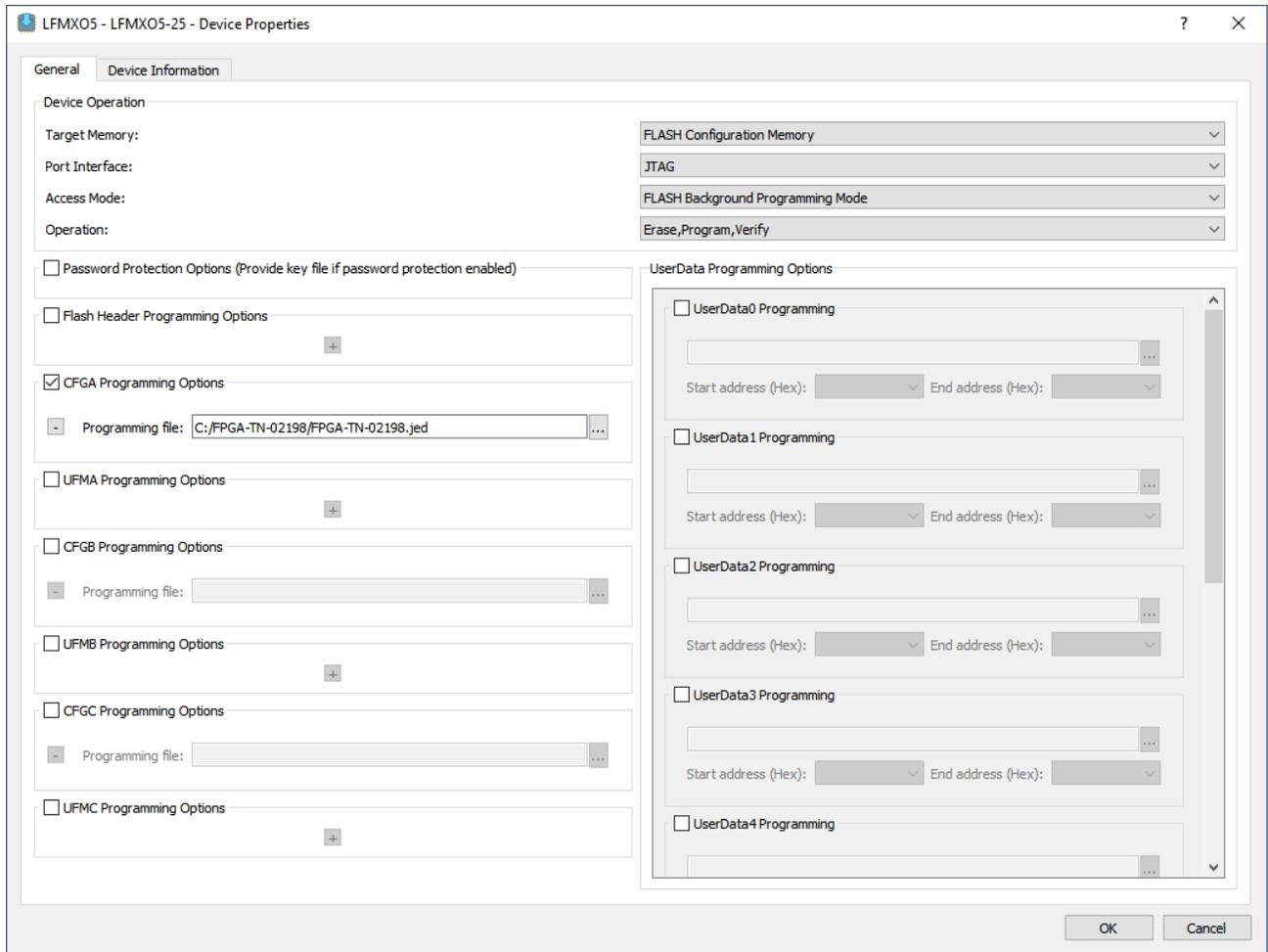
FPGA-TN-02198-4.4

Figure I.2. Radiant Programmer – Flash Background Programming through JTAG

References

For more information, refer to:

- MachXO2 Programming and Configuration Usage Guide (FPGA-TN-02155)
- CrossLink-NX FPGA website
- MachXO5-NX FPGA website
- MachXO3D FPGA website
- MachXO FPGA website
- ECP5/ECP5-5G FPGA website
- LatticeECP3 FPGA website
- LatticeECP2/M FPGA website
- LatticeXP2 FPGA website
- Lattice Diamond User Guide
- Lattice Radiant Software User Guide
- Lattice Radiant FPGA design software
- Lattice Diamond FPGA design software
- Lattice Insights for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

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

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 4.4, October 2023

Section	Change Summary
Disclaimers	Updated this section.
Introduction	Updated Table 1.1. Device Support for TransFR to change CrossLink-NX-33-U to *CrossLinkU-NX*.
Appendix H. CertusPro-NX, CrossLink-NX-33, or CrossLinkU-NX	Updated section name and content to change CrossLink-NX-33U to *CrossLinkU-NX*.
TransFR Technology	Corrected broken link for MachXO2 Programming and Configuration Usage Guide (FPGA-TN-02155) in Design Consideration for Non-JTAG Mode TransFR section.

Revision 4.3, September 2023

Section	Change Summary
Introduction	Updated Table 1.1. Device Support for TransFR to add the device family CrossLink-NX-33U.
Appendix H. CertusPro-NX, CrossLink-NX-33, or CrossLink-NX-33U	• Updated the title to add CrossLink-NX-33U. • Added CrossLink-NX-33U to the description.
References	Newly added section.

Revision 4.2, April 2023

Section	Change Summary
All	Minor adjustments in formatting in the document.
TransFR Technology	Updated wording in the opening sentence to: "Minimizing system interruption is achieved through background programming capabilities and TransFR technology".
Technical Support Assistance	Added reference link to Lattice Answer Database.

Revision 4.1, June 2022

Section	Change Summary
Introduction	CrossLink-NX-33 added to Device Family in Table 1.1.
Appendix H. CertusPro-NX or CrossLink-NX-33	Added CrossLink-NX-33 device.

Revision 4.0, March 2022

Section	Change Summary
All	Minor adjustments in formatting across the document.
Introduction	Added MachXO5-NX support in Table 1.1. Device Support for TransFR.
TransFR Technology	Updated JTAG Mode TransFR section to add Appendix I reference.
TransFR Using Diamond Programmer	Updated section content to add Appendix I. MachXO5-NX reference.
Appendix I. MachXO5-NX	Added this section.

Revision 3.9, November 2021

Section	Change Summary
All	Changed all references from 'MachXO3' to 'MachXO3LF'
Introduction	Removed MachXO3L support from Table 1.1.

Revision 3.8, August 2021

Section	Change Summary
All	• Changed document number from TN1087 to FPGA-TN-02198. • Updated document template.
Disclaimers	Added this section.
Acronyms in This Document	Added this section.
Introduction	Added CertusPro-NX support in Table 1.1.
TransFR Using Diamond Programmer	Changed TransFR Using Programmer heading name to TransFR Using Diamond Programmer.
Appendix H. CertusPro-NX	Added this section.
Glossary	Removed this section.

Revision 3.7, October 2015

Section	Change Summary
Introduction	Updated Introduction section. Revised Table 1.1, Device Support for TransFR. Added ECP5-5G.
Appendix G. ECP5 and ECP5-5G	Updated Appendix G. ECP5 and ECP5-5G section. Added ECP5-5G.
Technical Support Assistance	Updated Technical Support Assistance section.

Revision 3.6, March 2015

Section	Change Summary
All	• Product name/trademark adjustment. • Added NVCM information.
Introduction	Updated Introduction section. Revised Table 1.1, Device Support for TransFR. Added MachXO3LF.

Revision 3.5, March 2014

Section	Change Summary
Introduction	Updated Table 1.1, Device Support for TransFR. Changed LatticeECP4UM to ECP5.
Appendix G. ECP5 and ECP5-5G	Updated Appendix G. ECP5, Device Support for TransFR. Changed LatticeECP4UM to ECP5.

Revision 3.4, February 2014

Section	Change Summary
All	Added support for LatticeECP4UM.
Introduction	Updated Table 1.1, Device Support for TransFR
Appendix G. ECP5 and ECP5-5G	Added Appendix G. LatticeECP4UM.

Revision 3.3, September 2013

Section	Change Summary
TransFR Technology	• Introduced the Non-JTAG mode TransFR. • Separated device families based on TransFR mode
All	Updated example figures using Diamond Programmer and Deployment Tools screen shots.
Technical Support Assistance	Updated Technical Support Assistance information.

Revision 3.2, June 2013

Section	Change Summary
TransFR Technology	Added toggling PROGRAMN pins as an option for initiating the reconfiguration process for TransFR.
Appendix D. LatticeECP2/M	Updated the Selecting TransFR for LatticeECP3 screen shot.
All	Deleted the descriptions of I/O State options.

Revision 3.1, May 2013

Section	Change Summary
Appendix D. LatticeECP2/M	Updated the Enabling TransFR in the Design Planner screen shot for the LatticeECP2/M.

Revision 3.0, April 2013

Section	Change Summary
All	Updated document with new corporate logo.
Appendix C. MachXO2 and MachXO3	Created separate appendices for MachXO and MachXO2 and updated figures for selecting TransFR for MachXO2.
Appendix D. LatticeECP2/M	Updated the Enabling TransFR in the Design Planner screen shot.

Revision 2.9, August 2011

Section	Change Summary
All	Added User SPI during background programming caution.

Revision 2.8, May 2011

Section	Change Summary
All	Documented ispVM "TransFR Options..." for MachXO/XO2, LatticeXP2, LatticeECP2, and LatticeECP3 devices listed.

Revision 2.7, November 2010

Section	Change Summary
All	Updated for LatticeECP3 and MachXO2 device support.

Revision 2.6, October 2008

Section	Change Summary
Appendix D. LatticeECP2/M	Updated LatticeXP2 appendix.

Revision 2.5, May 2008

Section	Change Summary
All	• Updated screen shots. • Enhanced description of I/O state options. • Corrected MachXO behavior description.

Revision 2.4, May 2007

Section	Change Summary
Appendix D. LatticeECP2/M	Added LatticeXP2 appendix.

Revision 2.3, January 2007

Section	Change Summary
Appendix D. LatticeECP2/M	LatticeECP2/M appendix: added note about using dual-purpose pins and added ispLEVER software setting.

Revision 2.2, September 2006

Section	Change Summary
Appendix D. LatticeECP2/M	Added LatticeECP2M information.

Revision 2.1, September 2006

Section	Change Summary
Appendix D. LatticeECP2/M	Added LatticeECP2M information.
All	Updated screen shots
Introduction	Added Table 1.1 to list device family support.

Previous Lattice releases