



SEU Demo for MachXO3LF Starter Kit

User Guide

UG115 Version 1.0

April 2016

Contents

Acronyms in This Document	3
1. Introduction	4
2. Demo Package	4
3. Hardware Requirements	4
4. Software Requirements	4
5. Demo Design Overview	4
6. Port Assignments and Descriptions	6
7. Demo Package Directory Structure	6
8. Running the Demo	7
8.1. Loading and Running SEU Demo	7
8.2. Loading SEI and Detecting the Inserted Error	9
8.3. Refreshing the Image and Confirming that Error has been Corrected (SEC)	12
9. Diamond SEI Generation Tool	15
10. Developer Notes	18
References	19
Technical Support	19
Revision History	20

Figures

Figure 5.1. MachXO3LF Starter Kit (Top View)	5
Figure 5.2. SEU Demo Design Block Diagram	5
Figure 7.1. SEU Demo Package Directory Structure	6
Figure 8.1. Getting Started	7
Figure 8.2. Selecting the Device	7
Figure 8.3. Device Properties	8
Figure 8.4. Access Mode – Flash Programming Mode	8
Figure 8.5. Launching the Program Step	9
Figure 8.6. Program Operation Successful	9
Figure 8.7. Device Properties Window	10
Figure 8.8. Access Mode – Static RAM Cell Background Mode	10
Figure 8.9. Launching the Program Step	11
Figure 8.10. SEI Operation Successful	11
Figure 8.11. Device Properties Selection	12
Figure 8.12. Background Programming File Loading	13
Figure 8.13. Background Refresh Command	13
Figure 8.14. Program Option	14
Figure 8.15. Original Bitstream Successful Programming	14
Figure 9.1. Diamond Software – SEI Editor Utility	15
Figure 9.2. SEI Editor Utility in Diamond Project Window	16
Figure 9.3. SEI Editor Tool	16
Figure 9.4. SEI Editor showing Error Bit in Frame	17

Tables

Table 6.1. FPGA Demo Design Ports	6
Table 6.2. DIP Switch Definitions	6
Table 6.3. LED Definitions	6

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
ESD	Electrostatic Discharge
FPGA	Field-Programmable Gate Array
I ² C	Inter-Integrated Circuit
SEC	Soft Error Correction
SED	Soft Error Detection
SEI	Soft Error Injection
SEU	Single Event Upset
SPI	Serial Peripheral Interface
SRAM	Static Random-Access Memory
USB	Universal Serial Bus

1. Introduction

This document provides technical information and instructions for using the Single Event Upset (SEU) demo design from Lattice Semiconductor. This demo demonstrates the functionality of Soft Error Detection (SED) and Soft Error Correction (SEC) using the MachXO3LF starter kit. The MachXO3LF development kit includes the MachXO3LF starter kit board, a USB cable, power supply, and demo files. This document provides a description of the demo design as well as instructions for running the demo.

The demo design makes use of Lattice Diamond® software feature of Soft Error Injection (SEI) to induce an SRAM bit-flip error. This error is then detected by the FPGA SED feature running on the board. Finally, the demonstration shows the MachXO3LF device's ability to correct the inserted error (SEC feature) without interruption of the user function.

2. Demo Package

The demo package includes the following:

- Verilog source code for the demo logic design
- Lattice Diamond® Project file and preference file for the demo project
- SEU demo bitstream
- SEI bitstream for soft error injection

3. Hardware Requirements

Beside the MachXO3LF starter kit, a USB cable is required for programming the MachXO3LF device.

Note: Static electricity can severely shorten the lifespan of electronic components. Be careful when handling the MachXO3LF starter kit as to not damage it from electrostatic discharge (ESD).

4. Software Requirements

The following software programs are available at www.latticesemi.com/en/Products/DesignSoftwareAndIP

The software programs are available for download only if you log in at www.latticesemi.com

- Lattice Diamond Design Software, version 3.7 or later
- MachXO3LF specific SEI enabled license feature
- Lattice Programmer® Software for bitstream downloading

5. Demo Design Overview

This demo design consists of two major parts:

- SED module – Performs read and error detection of SRAM content.
- User logic – Includes soft error indication and a function block that rotates the LEDs on board.

The status of the demo is indicated with onboard LEDs. A SEI bitstream is used to induce a single SRAM error into the SEU demo. This SEI bitstream is generated with SEI editor under Tools tab in Lattice Diamond software.

[Figure 5.1](#) on the next page shows the top side of the MachXO3LF starter kit.

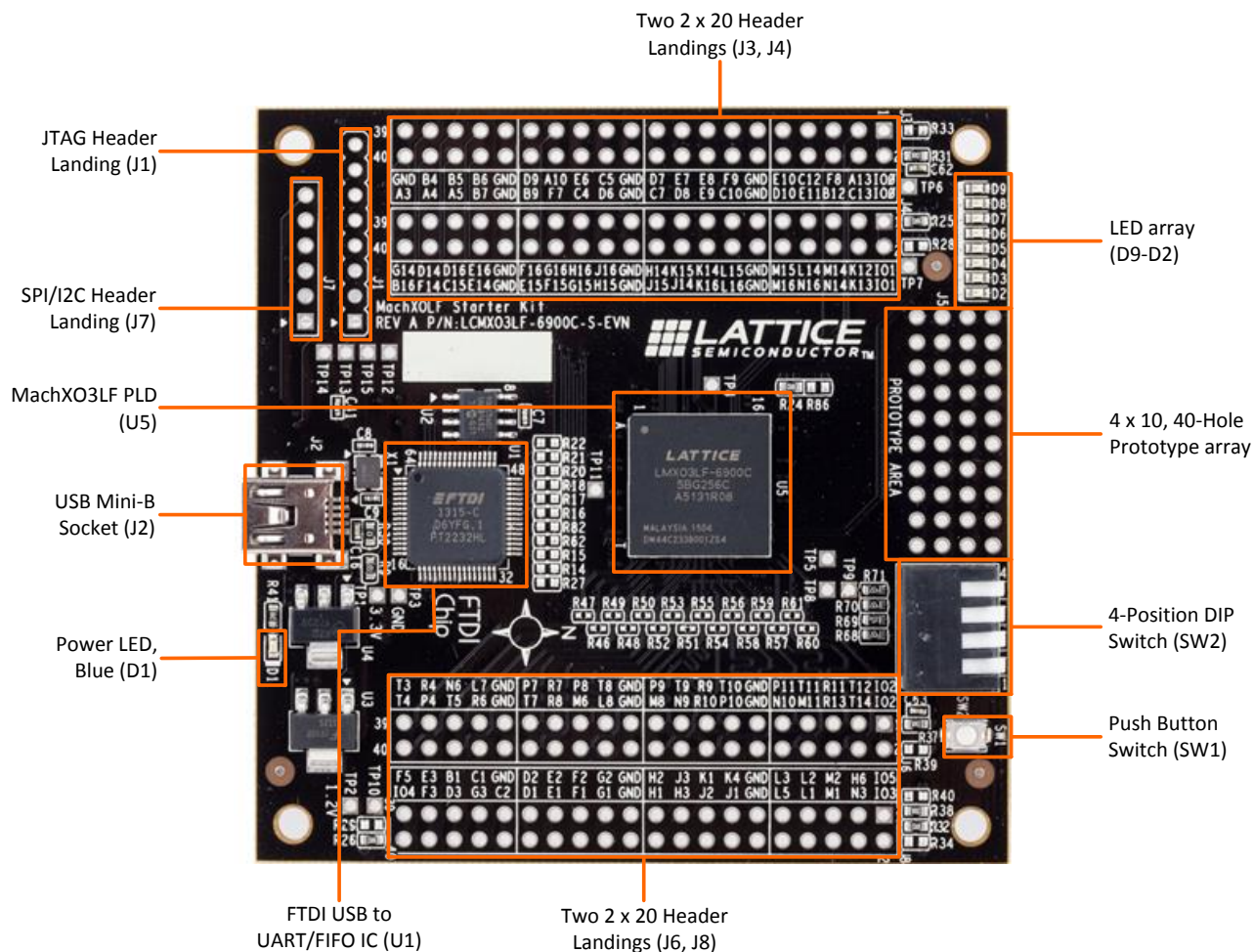


Figure 5.1. MachXO3LF Starter Kit (Top View)

Figure 5.2 shows the block diagram of SEU Demo design.

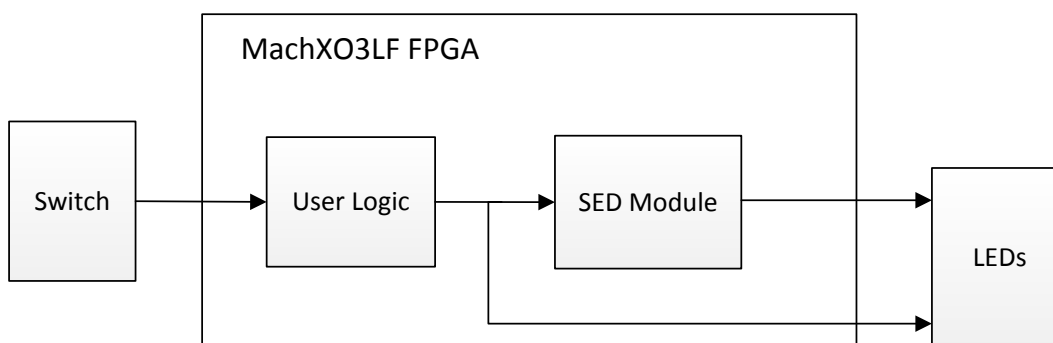


Figure 5.2. SEU Demo Design Block Diagram

6. Port Assignments and Descriptions

Table 6.1. FPGA Demo Design Ports

Port Name	Direction	Description
clk	Input	Clock for user logic
reset_n	Input	GSRN button on board. Asynchronous reset for FPGA
sw[1:0]	Input	DIP switch on board
led[6:0]	Output	LEDs on board
sed_err	Output	SED error indication

Table 6.2. DIP Switch Definitions

SW2 on Board	Position Definition	Description
#1	1 st switch next to SW1	SED Enable
#2	2 nd switch next to 1 st switch	SED Start

Table 6.3. LED Definitions

LED on Board	Description
D9, D8, D7, D6, D5, D4, D3	Rotating from D9 to D3. Indicate user logic is running
D2	Indicate SED error

7. Demo Package Directory Structure

The demo bitstreams are located at the following path:

SEU_XO3LF_Demo\bistream

The Diamond project is located at the following path:

SEU_XO3LF_Demo\SED\impl

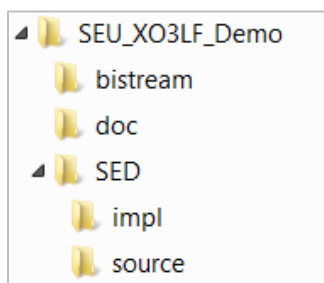


Figure 7.1. SEU Demo Package Directory Structure

8. Running the Demo

There are three processes involved in running the demo:

1. Loading and running SEU demo
2. Loading SEI and detecting the Inserted Error
3. Refreshing the image and confirming that the error has been corrected (SEC)

8.1. Loading and Running SEU Demo

To load and run the SEU demo:

1. Before the board is powered on, set all positions of DIP Switch SW2 to DOWN (all OFF).
2. Power on the board by connecting it to the PC via the USB cable.
3. Launch the Diamond Programmer Software.

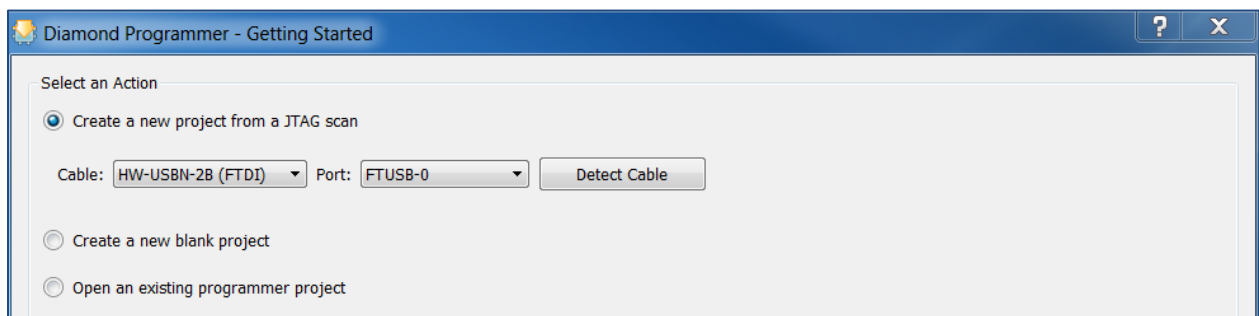


Figure 8.1. Getting Started

4. When the scanning of the device is completed, the device present on the board is detected as shown in Figure 8.2.

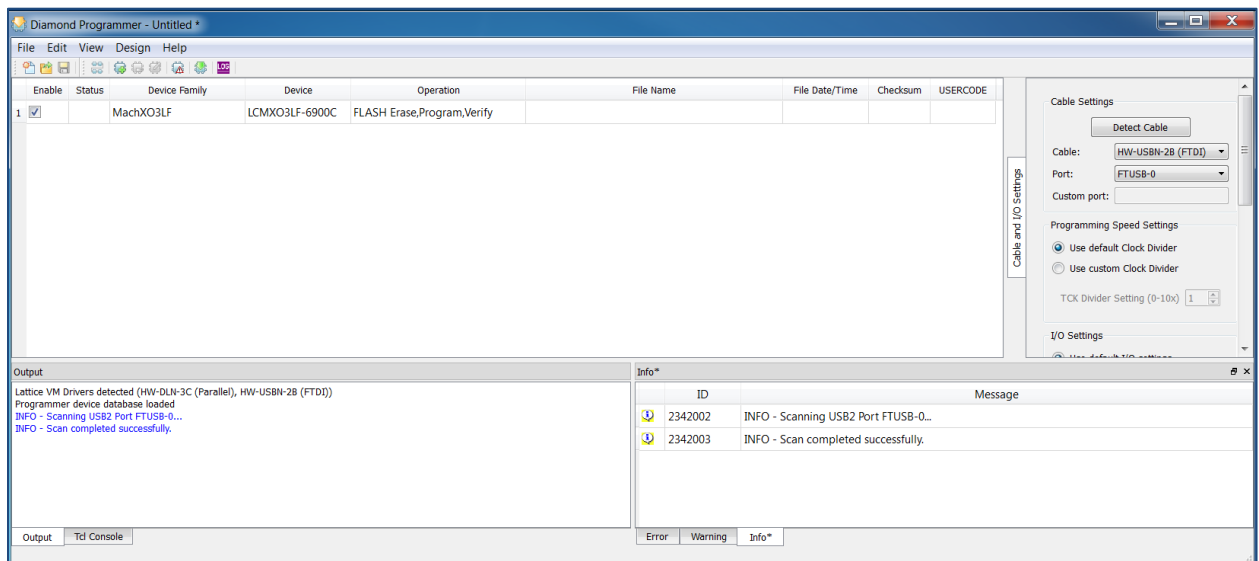


Figure 8.2. Selecting the Device

5. The device selected appears as LCMXO3LF-6900C. Click on the Device line so its gets highlighted in blue as shown in Figure 8.3.
6. Click **Edit**, and then **Device Properties**.

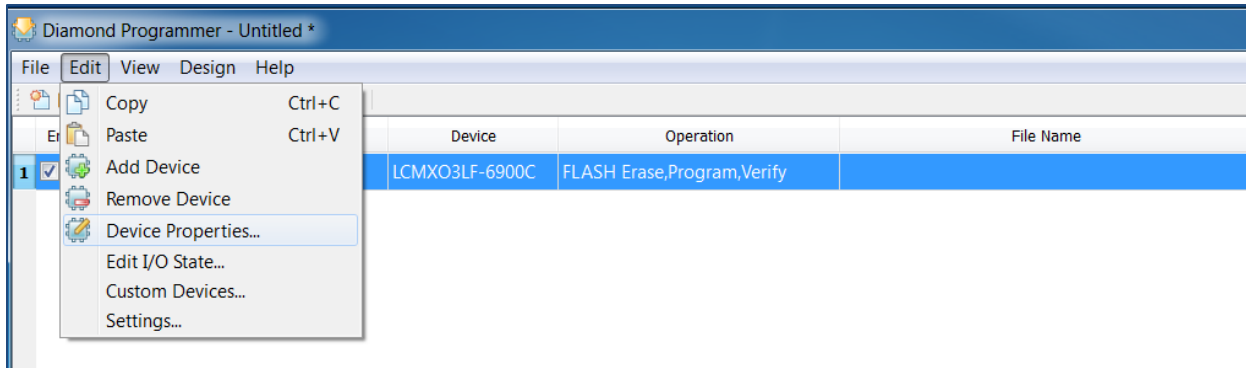


Figure 8.3. Device Properties

7. Select *Flash Programming Mode* from the **Access mode** dropdown list, and *FLASH Erase, Program, Verify* from the **Operation** dropdown list as shown in [Figure 8.4](#).
8. In **Programming Options** section select the original bitstream file
SEU_XO3LF_Demo\bistream\SEU_XO3LF_demo_SEU_XO3LF_demo.jed to be programmed.
9. Click **OK**.

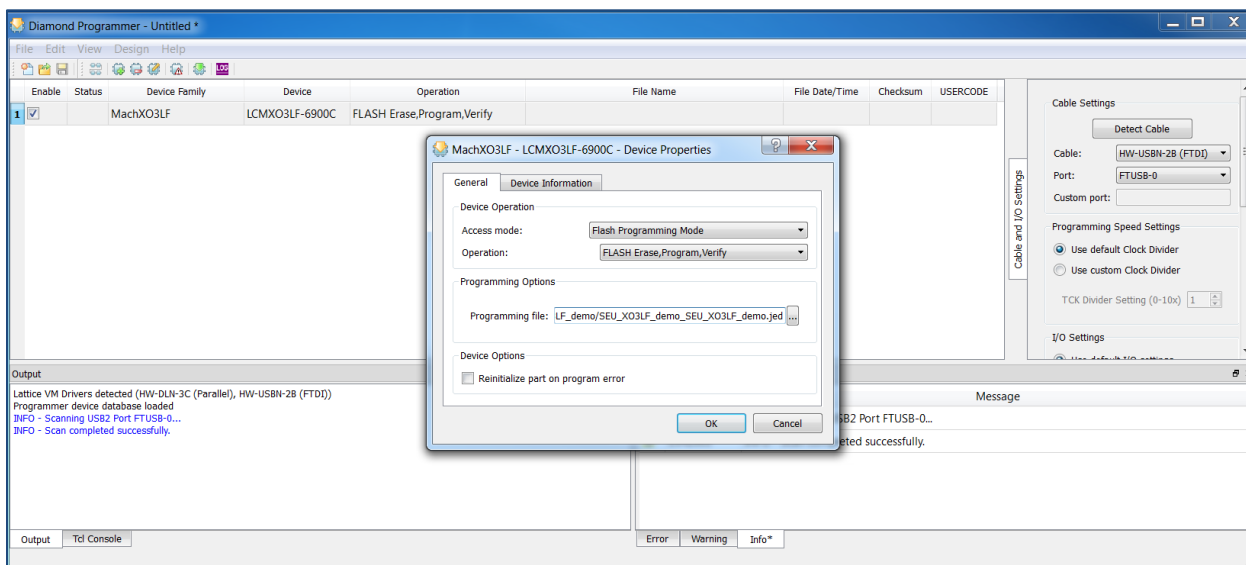


Figure 8.4. Access Mode – Flash Programming Mode

10. On the menu bar, click **Design**, and on the dropdown menu click **Program** to initiate programming of FPGA as shown in [Figure 8.5](#).

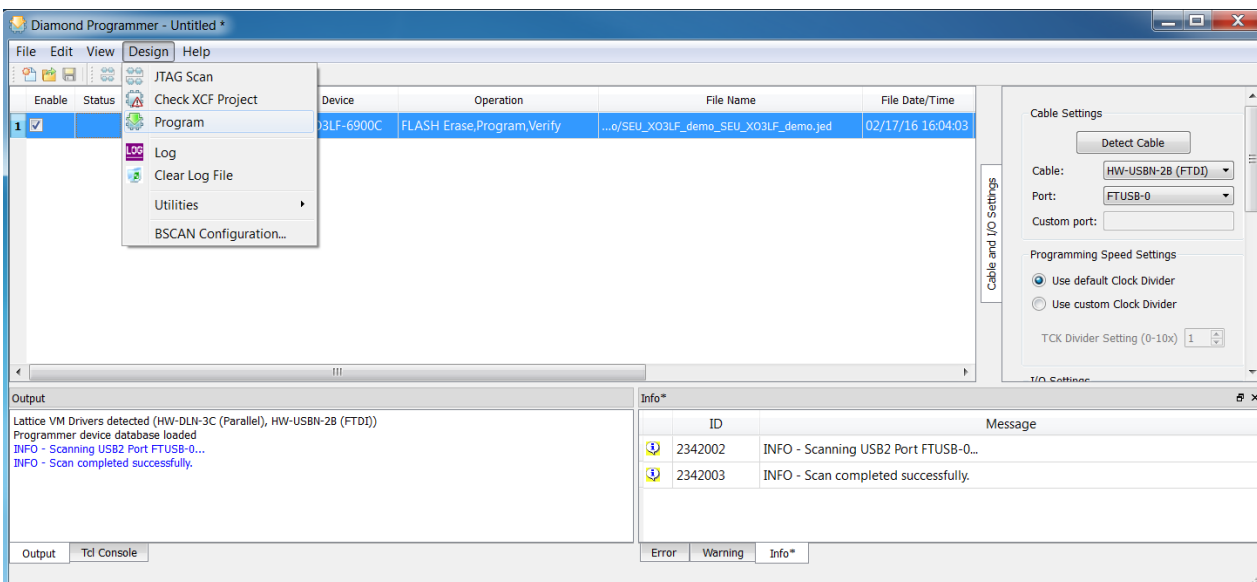


Figure 8.5. Launching the Program Step

When programming is done, rotating LEDs from D9 to D3 are seen on the board as well as *Operation: Successful* message is displayed in the Output console window as shown in Figure 8.6.

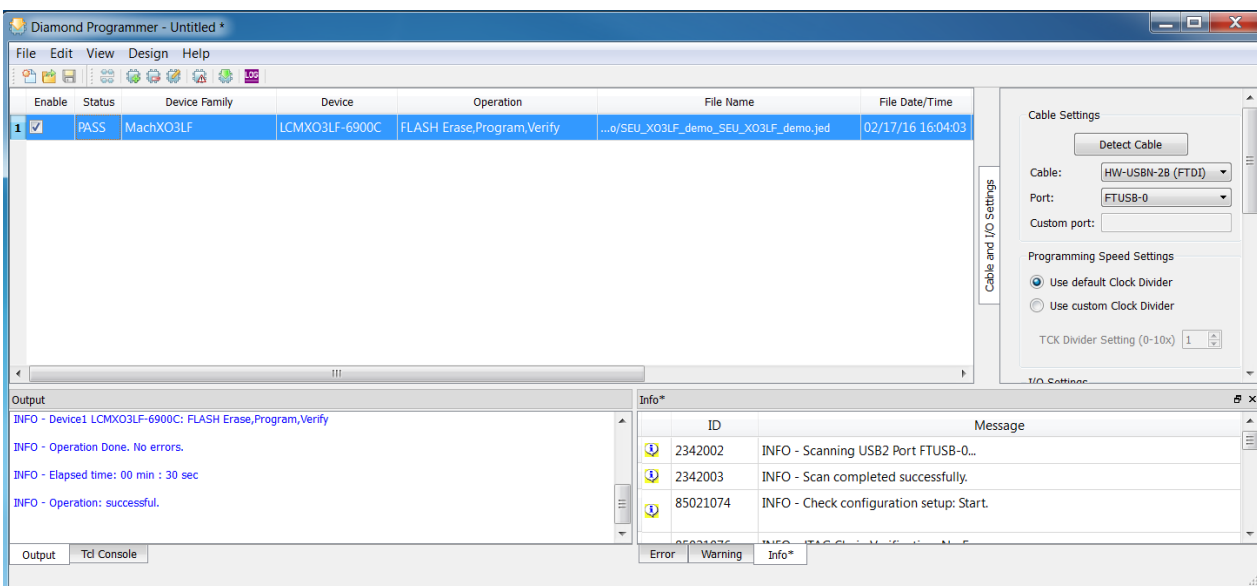


Figure 8.6. Program Operation Successful

11. For SW2, first set switch #1 position UP (ON) and then switch #2 position UP (ON). There should be no SED error indication (LED D2 is not lit).
12. Set SW2, switch #1 and switch #2 positions DOWN (OFF). The LED (D2) will remain off.

8.2. Loading SEI and Detecting the Inserted Error

To insert an error in the SRAM array (load the SEI bit file):

1. Click on the Device line so its gets highlighted in blue as shown in Figure 8.7.
2. Click **Edit**, and then **Device Properties**.

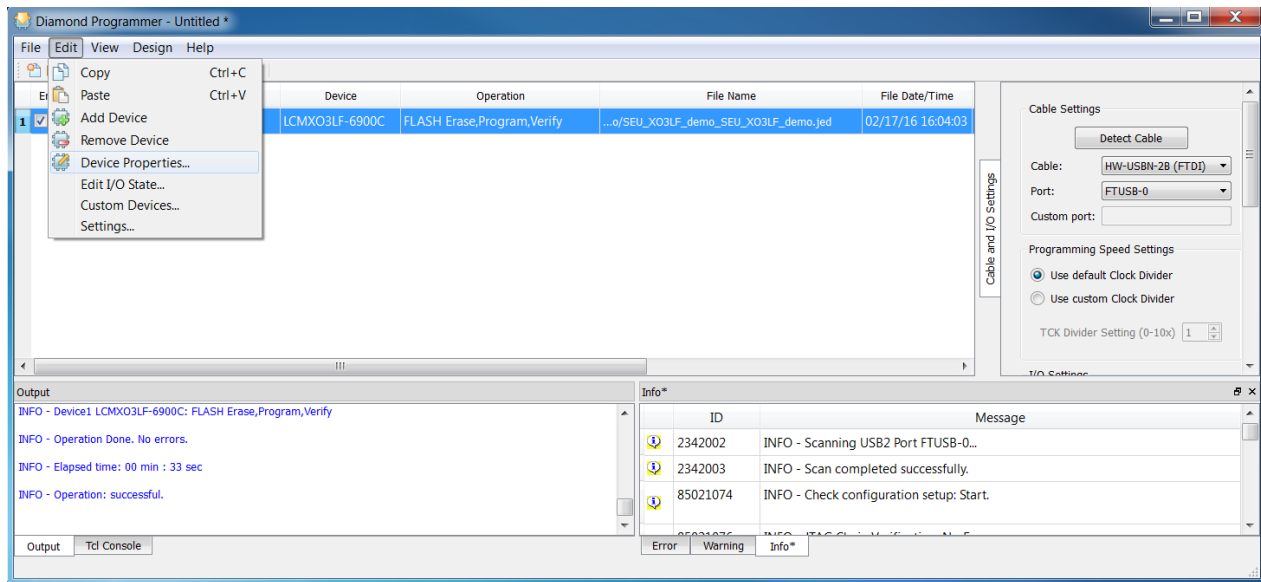


Figure 8.7. Device Properties Window

3. Select *Static RAM Cell Background Mode* in the **Access mode** option and *XSRAM SEI Fast Program* in **Operation** as shown in Figure 8.8.
4. In **Programming Options** section select the SEI bitstream file SEU_XO3LF_Demo\bistream\SEU_XO3LF_demo_sei_0.bit to be programmed.
5. Click **OK**.

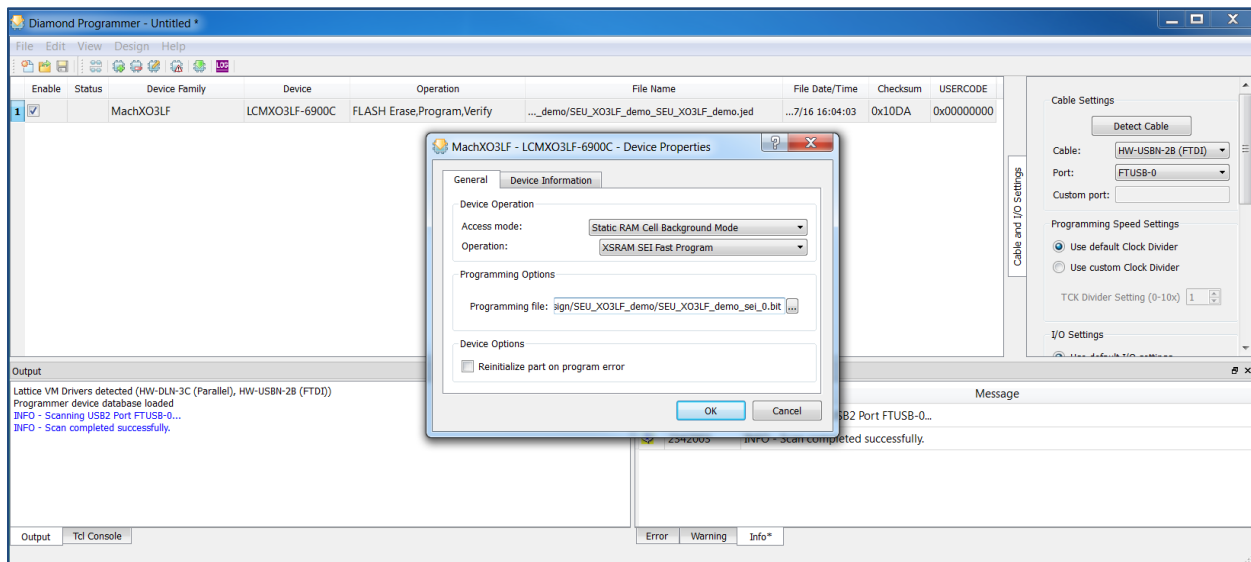


Figure 8.8. Access Mode – Static RAM Cell Background Mode

6. To insert the error into the SRAM array, on the menu bar, click **Design**, and on the dropdown menu click **Program** to insert the error bit into the SRAM array through background reconfiguration.

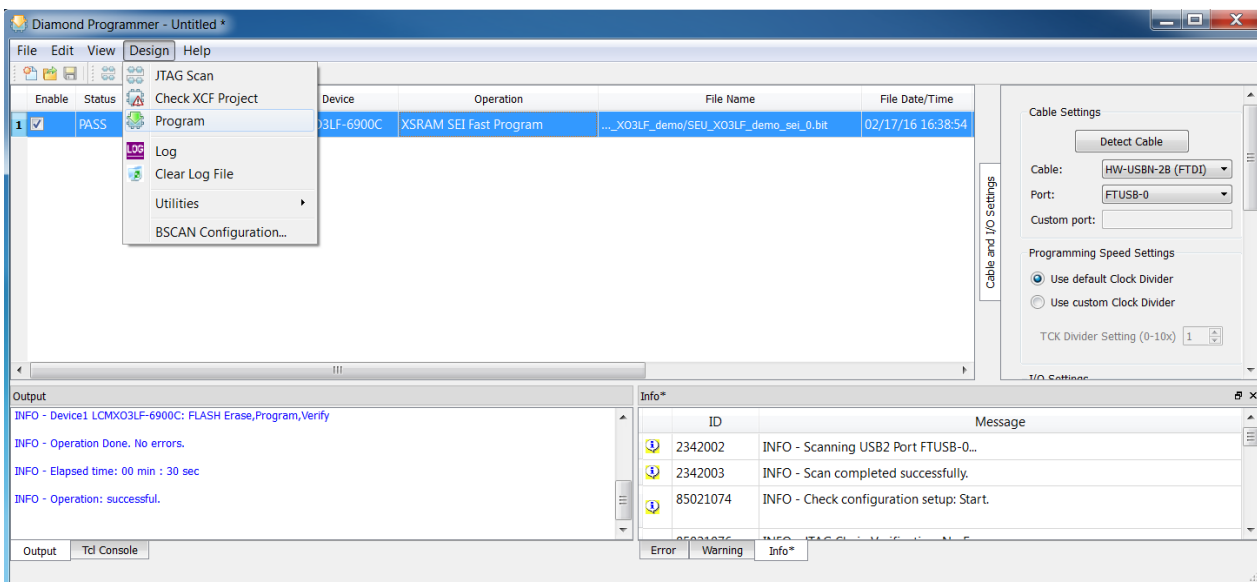


Figure 8.9. Launching the Program Step

When error insertion is completed, *Operation: Successful* message is displayed in the Output console window as shown in Figure 8.10.

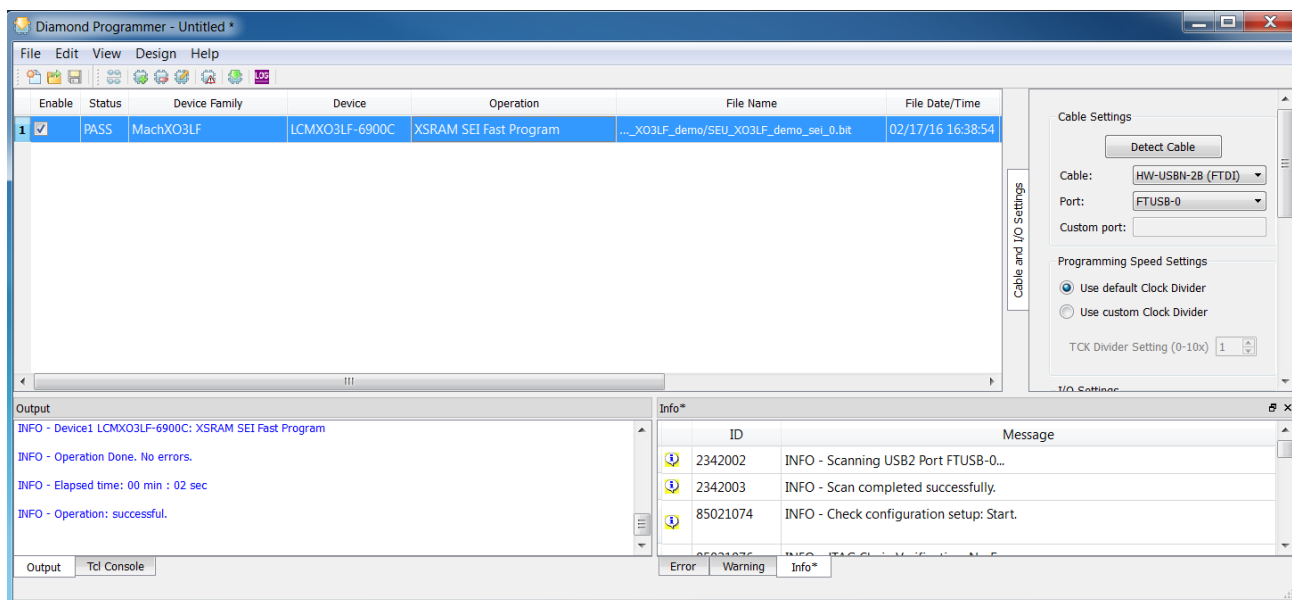


Figure 8.10. SEI Operation Successful

LEDs from D9 to D3 should keep rotating when the SEI bitstream is being programmed. This indicates that there is no interruption to the currently running bitstream on the device. For this demo the DONE pin is enabled. Probing DONE pin may be helpful to observe the FPGA device configuration status.

7. For SW2, first set switch #1 position UP (ON) and then set switch #2 position UP (ON). The LED D2 will light in response to the SED.
8. Set SW2, switch #1 and switch #2 positions DOWN (OFF). The LED D2 will remain on.

8.3. Refreshing the Image and Confirming that Error has been Corrected (SEC)

The Lattice MachXO3 device supports multiple methods to refresh the SRAM array and correct any detected soft error:

- By providing the original bitstream in the background via an external sysConfig slave port (for example JTAG or I²C).
- By transferring the original bitstream from internal or external Flash memory, initiated via sysConfig REFRESH command.
- By transferring the original bitstream from internal or external Flash memory, initiated via external PROGRAMN pin assertion.

This demo design supports the first two methods as described below. To refresh the SRAM array with the original bitstream via the JTAG port:

1. Click on the Device line so it gets highlighted as shown in Figure 8.11.
2. Click **Edit**, and then **Device Properties**.

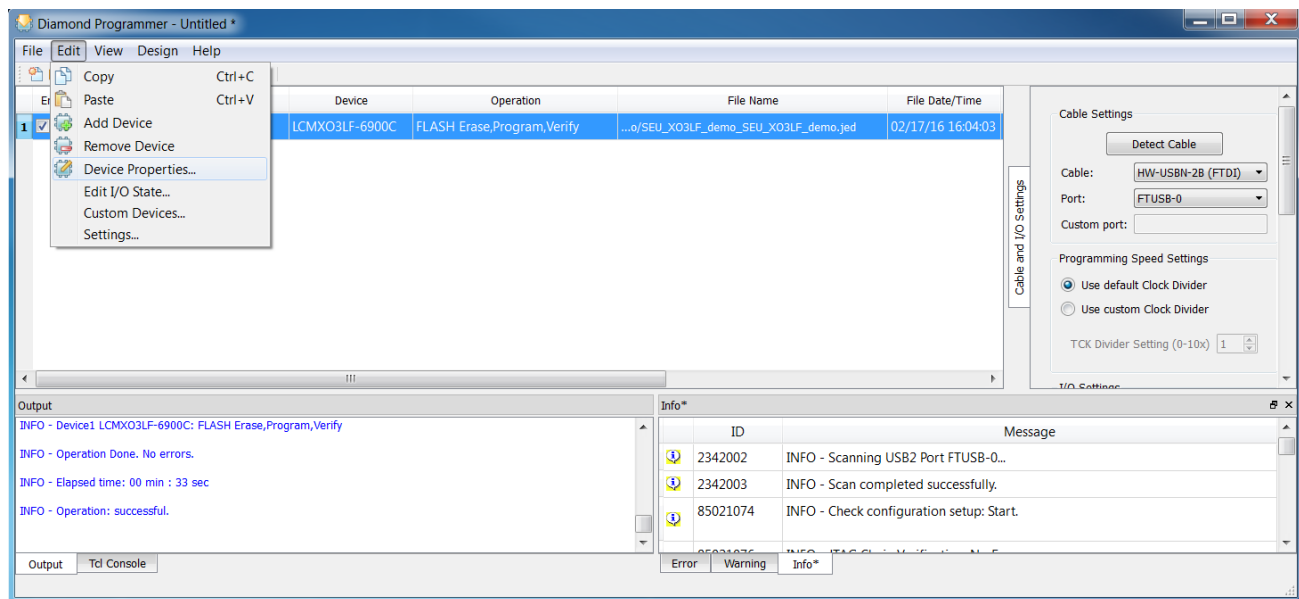


Figure 8.11. Device Properties Selection

3. Select *Static RAM Cell Background Mode* from the **Access mode** dropdown list, and *XSRAM SEI Fast Program* from the **Operation** dropdown list as shown in Figure 8.12.
4. In **Programming Options** section select the original bitstream file SEU_XO3LF_Demo\bistream\SEU_XO3LF_demo_SEU_XO3LF_demo.bit to be programmed.
5. Click **OK**.

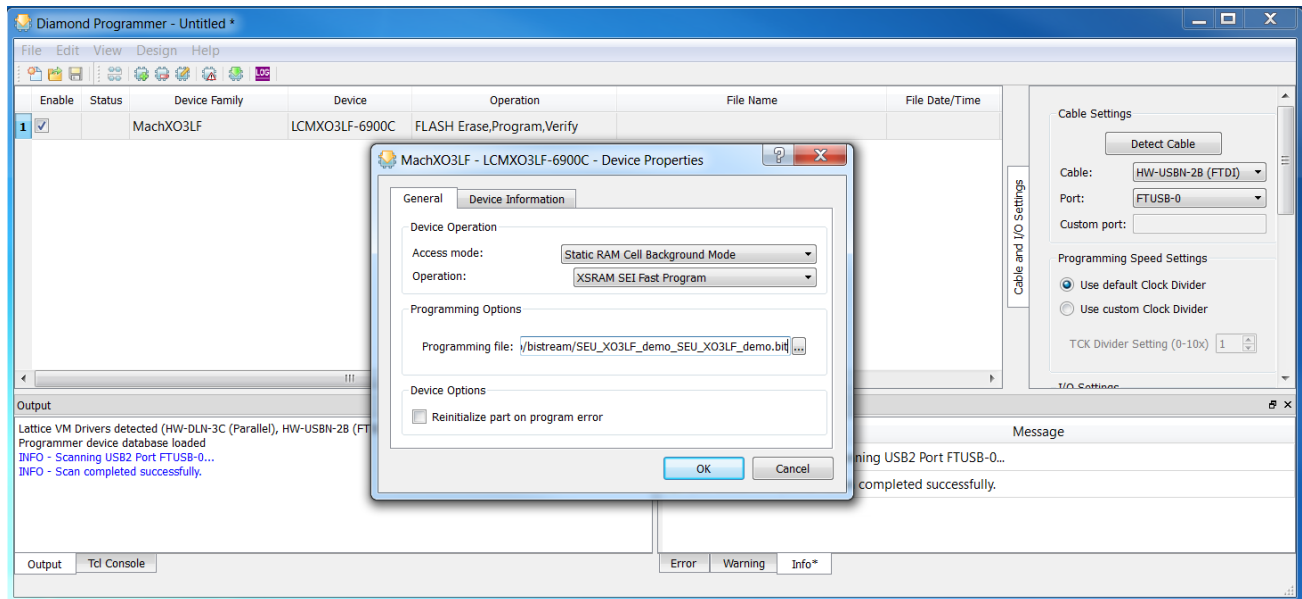


Figure 8.12. Background Programming File Loading

Alternatively, refresh the SRAM array from the internal flash using the REFRESH command:

1. Click on the Device line so it gets highlighted.
2. Click **Edit**, and then **Device Properties**.
3. Select *Static RAM Cell Background Mode* from the **Access Mode** dropdown list, and *XSRAM Refresh* from the **Operation** dropdown list as shown in Figure 8.13.
4. Click **OK**.

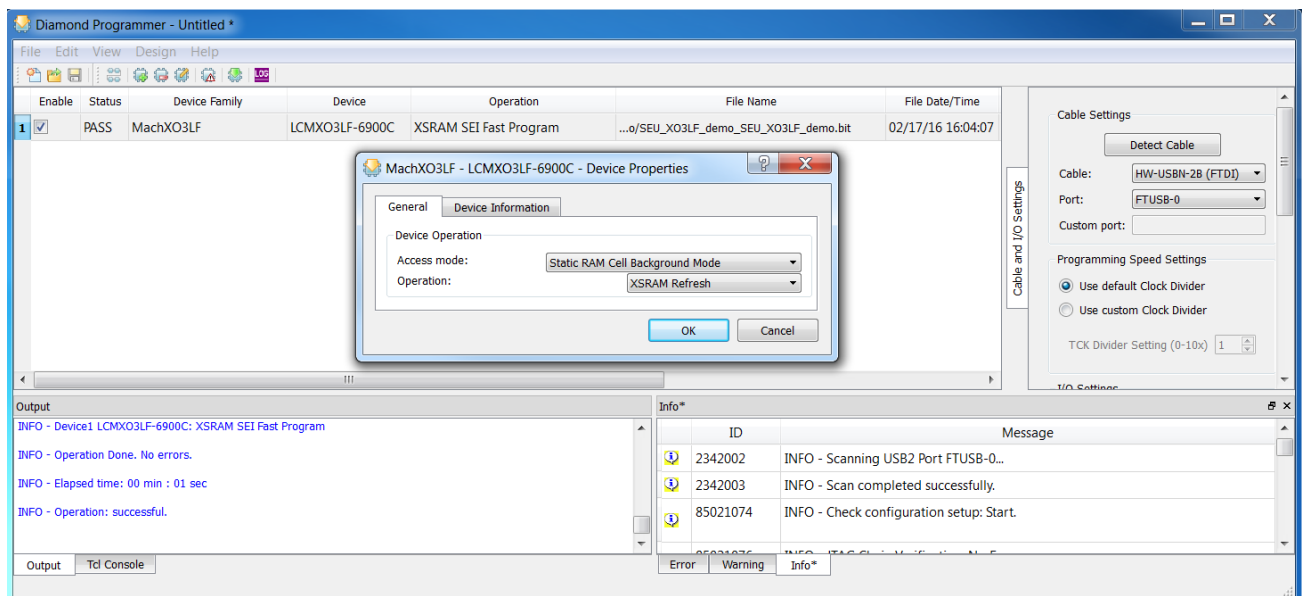


Figure 8.13. Background Refresh Command

To refresh the SRAM array by the method chosen above:

1. On the menu bar, click **Design** and on the dropdown menu click **Program** to configure the FPGA with the original bitstream as shown in Figure 8.14.

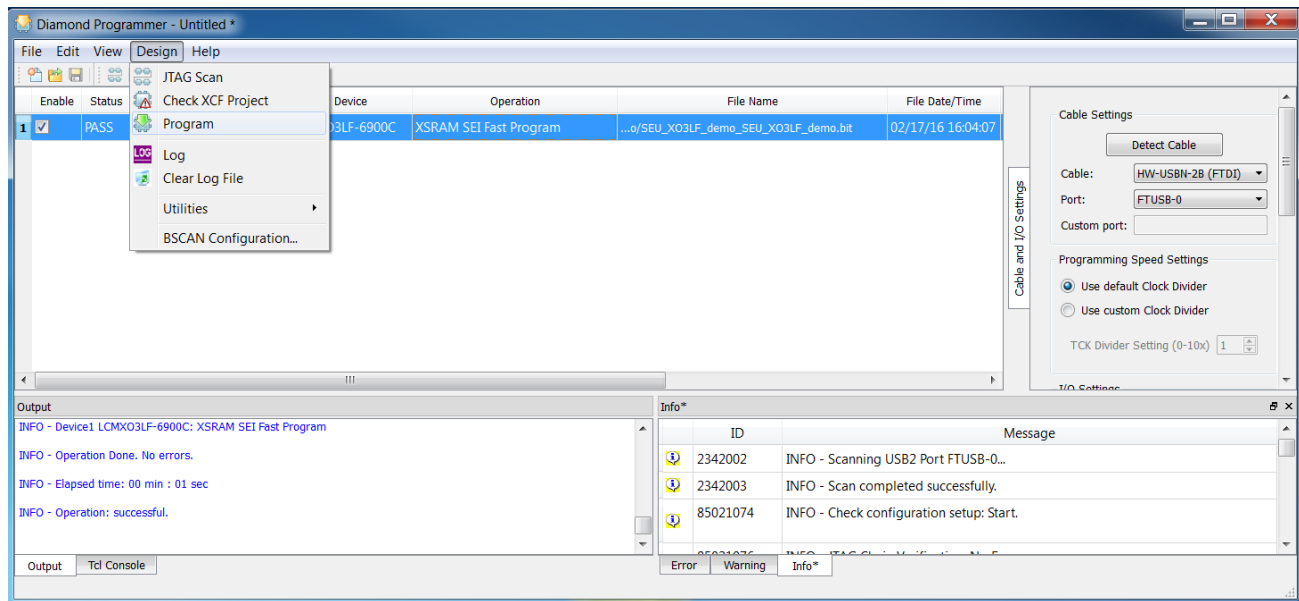


Figure 8.14. Program Option

When the FPGA is configured correctly, the *Operation: Successful* message is displayed in the Output console window as shown in Figure 8.15.

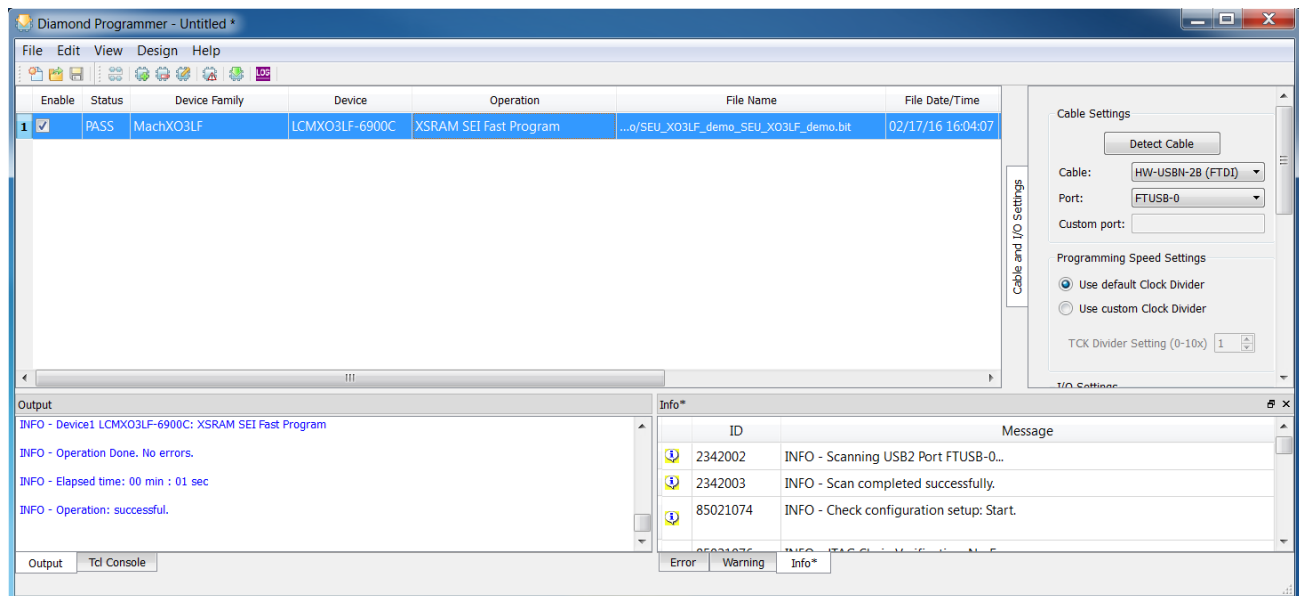


Figure 8.15. Original Bitstream Successful Programming

LEDs from D9 to D3 should keep rotating when the original bitstream is being programmed.

- For SW2, first set the switch #1 position UP (ON) and set switch #2 position UP (ON). The LED D2 will extinguish to confirm Soft Error is corrected (SEC).
- Set SW2, switch #1 and switch #2 positions DOWN (OFF).

9. Diamond SEI Generation Tool

Diamond Software has SEI editor utility which is used to generate an SRAM error bit for this demo. To use this SEI editor utility in Lattice Diamond Design Software, version 3.7 a special license is required. To get the license for this utility please contact www.latticesemi.com/techsupport. No license is required for Lattice Diamond Design Software, version 3.8 or later.

After installation, the tool can be accessed through the **Tools** menu option of Lattice Diamond Programmer as shown in Figure 9.1.

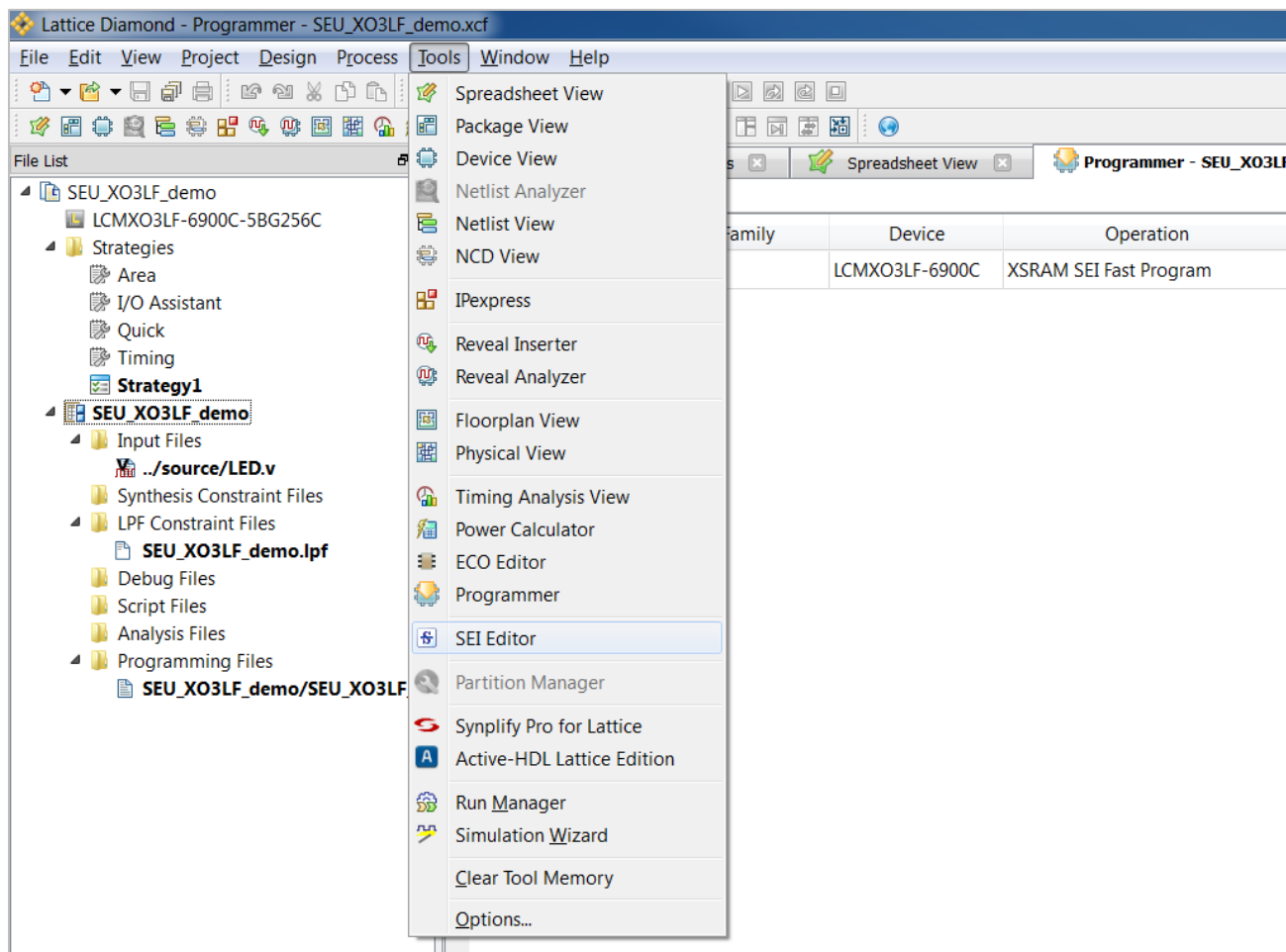


Figure 9.1. Diamond Software – SEI Editor Utility

Figure 9.2 on the next page shows the SEI Editor utility in Lattice Diamond project window.

In the SEI Editor tool, there are options to Run, Add and Remove the SEI error bit as shown in Figure 9.3.

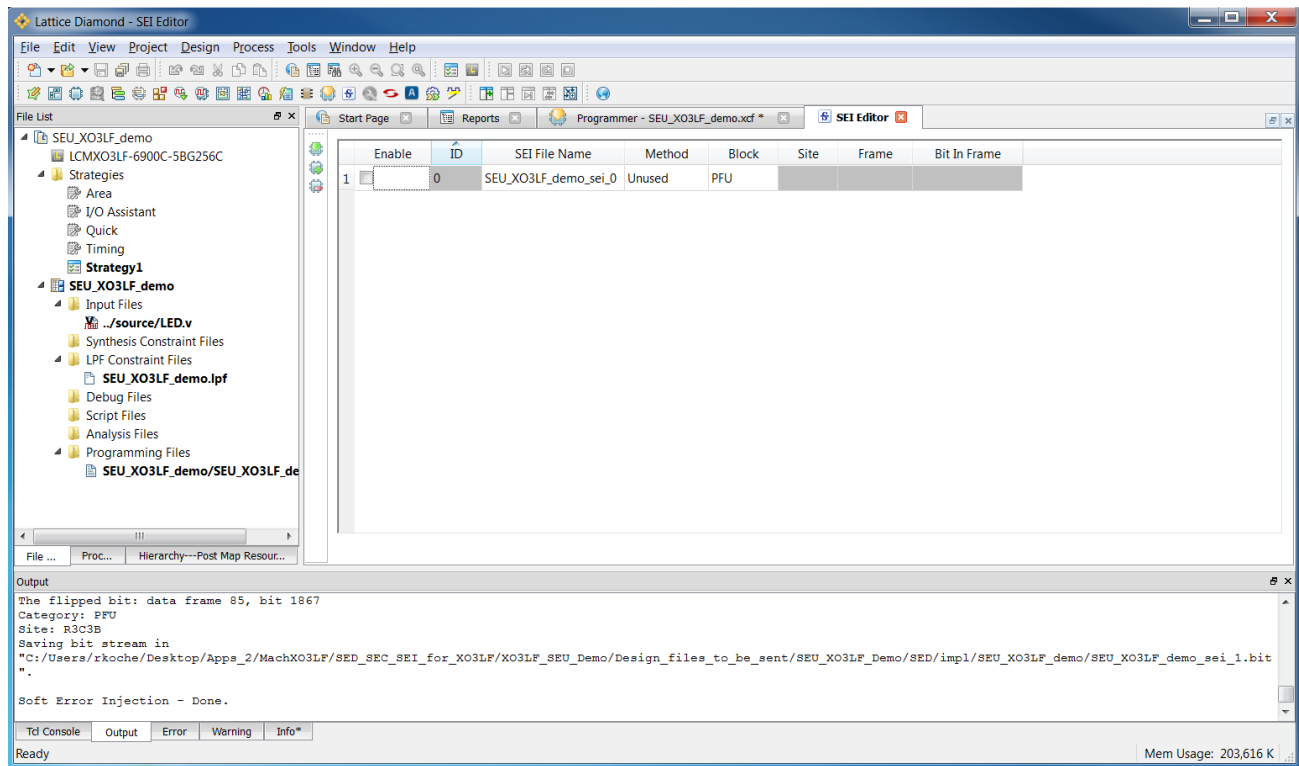


Figure 9.2. SEI Editor Utility in Diamond Project Window

To generate the error bit:

1. In the **Enable** tab, select the checkbox to select the row. See Figure 9.3.
2. Click the Run icon  to generate the error bit.

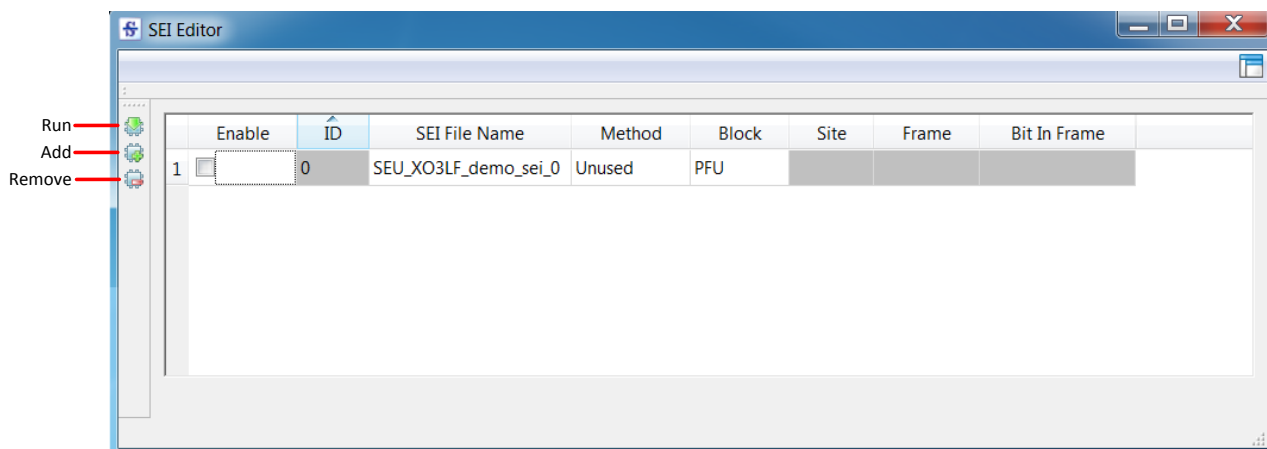


Figure 9.3. SEI Editor Tool

When the error bit is generated the tool shows which bit has been changed as shown in Figure 9.4.

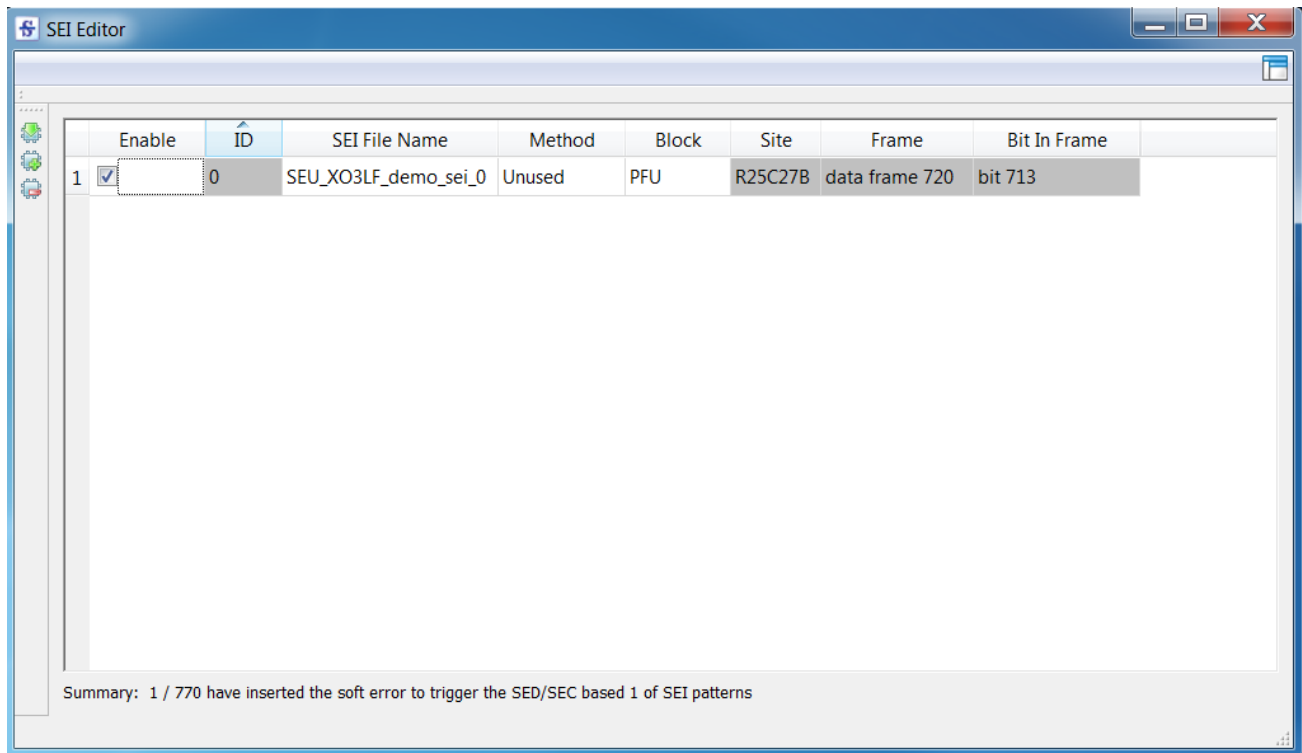


Figure 9.4. SEI Editor showing Error Bit in Frame

When the generation of this SEI bit is successful, a *.bit file is generated which has the error bit that is used to inject error in the design.

10. Developer Notes

To develop your own FPGA design to make use of the SEI flow:

1. The SED module primitive needs to be instantiated in the FPGA design. This is a hard IP module, and does not consume any user logic.
2. When a bitstream is generated, BACKGROUND_RECONFIG should be set to ON in Diamond Programmer to support the SEI and SEC features.
3. The SEI bitstream generation feature in Diamond Programmer software is enabled with an XO3 SEI specific software license feature. Visit www.latticesemi.com/techsupport to request this feature from the Lattice license administrator.

References

For more information, refer to:

- DS1047, [MachXO3 Family Data Sheet](#)
- TN1292, [MachXO3 SED Usage Guide](#)
- TN1279, [MachXO3 Programming and Configuration Usage Guide](#)

Technical Support

For assistance, submit a technical support case at www.latticesemi.com/techsupport.

Revision History

Date	Version	Change Summary
April 2016	1.0	Initial release.



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