



Avant Multi-Boot User Guide

Preliminary

Technical Note

FPGA-TN-02314-0.80

November 2022

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 associated risk the responsibility entirely of the Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. 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. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. 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

Glossary	4
1. Introduction	5
2. Resources	6
3. Avant Dual Boot Mode	7
3.1. Description of the Avant Dual Boot Flow Diagram	7
4. Creating a PROM File	9
4.1. Using Radiant Deployment Tool to Create a Dual Boot PROM Hex File	9
5. Programming the Dual Boot Pattern into the SPI Flash Device	14
Technical Support Assistance	16
Revision History	17

Figures

Figure 3.1. Avant Family Dual Boot Flow Diagram	7
Figure 4.1. Creating New Deployment for Dual Boot PROM Hex File	9
Figure 4.2. Select Input Files Window	10
Figure 4.3. Dual Boot Options Window	11
Figure 4.4. Select Output File Window	12
Figure 4.5. Generate Deployment Window	13
Figure 5.1. Radiant Programmer – Getting Started Window	15

Tables

Table 2.1. Maximum Configuration Bitstream Size – Single Bitstream Boot Mode	6
Table 2.2. Maximum Configuration Bitstream Size – Dual Boot Mode	6
Table 3.1. Control Register 1 [3:2] – Master Preamble Timer Retry Value	8

Glossary

A list of terms used in this document.

Acronym	Definition
Alternative Boot	After the FPGA device has been configured, this pattern is loaded when the PROGRAMN pin is toggled or the Refresh instruction is issued. Up to four Alternative Boot patterns are possible.
Binary Hex Data File (.bin File)	The data image of the Hex data file in binary format. All Hex data files are converted into this format prior to consumption. This type of file is not printable.
Bitstream Data File (.bit File)	The configuration data file, for a single FPGA device, in the format that can be loaded directly into the FPGA device to configure the SRAM cells. The file is expressed in binary Hex format. The file is not printable.
Configure	Write the pattern into the SRAM fuses of the FPGA device and wake up.
Dual Boot	The device has two patterns, a Primary pattern and a Golden pattern, to choose to load.
EBR	Embedded Block RAM
FD-SOI (Fully Depleted Silicon On Insulator)	A process that uses an ultra-thin buried oxide layer.
Flash Lock	The feature provides protection to the Flash fuses against accidental erase or corruption. Most of the SPI Flash devices support Soft Lock. Lock choices include: <ul style="list-style-type: none"> • Whole device • Bottom half • Bottom quarter • Last sector Details can be found in the SPI Flash device data sheet.
Golden Boot	The guaranteed good pattern loaded into the FPGA device when booting failure occurs. It is also known as the root boot. Only one Golden Boot pattern is allowed.
Hex Data File (.exo, .mcs, .xtek Files)	The data record files that are in the format commonly known as Intel Hex, Motorola Hex or Extended Tektronix Hex. They are also known as addressed record files. The advantages include its small size and it is printable, and thus good for record keeping. This type of file is not directly consumable by the utilities supporting it.
Multi-Boot Multiple Boot	The device has more than two patterns, a Primary pattern, a Golden pattern and some Alternative patterns, to choose to load.
Primary Boot	Upon power cycling, the FPGA device loads this pattern in first. Only one Primary pattern is allowed.
Program	Writes into the selected Flash cells state a logical zero (0) (close fuse).
Refresh	The action loads the pattern from a non-volatile source to configure the FPGA device.
SPI	Stands for the Serial Peripheral Interface defined originally by Motorola.
Sector (Block)	The smallest number of bytes of Flash fuses can be erased at the same time by the erase command.

1. Introduction

Lattice Avant™ is a low-power mid-range density FPGA platform optimized for a wide range of applications across multiple markets — optimized for edge computing workloads requiring large memories and DSP resources and delivering a variety of high bandwidth interfaces ideal for video processing, communications, and machine learning inferencing. With Lattice low-power FPGA architecture, Avant devices deliver best-in-class power efficiency while meeting performance requirements for a wide range of applications.

The Avant platform supports various booting options for loading the configuration SRAM from a non-volatile memory for configuration flexibility and fail-safe configuration. Avant devices require an external memory for storage of the configuration bitstreams.

Multiple configuration boot modes mitigate risk during the field upgrade process and allow flexibility of executing different patterns. Field upgrade disruptions may occur due to power disruption, communication interruption or bitstream pattern corruption. The Avant platform supports the following boot modes:

- Single Bitstream Boot mode.
- Dual Boot Mode — Switches to load from the second known good (Golden) pattern when the first pattern becomes corrupted.
- Ping-Pong Boot Mode — Switches between two bitstream patterns based on your choice. If the system fails to boot from one of the bitstreams, it automatically boots from the second bitstream.
- Multi-Boot Mode — Enables the device to dynamically switch from more than two bitstream patterns, inclusive of the Primary and Golden patterns.

The Avant platform combines multiple bitstream patterns into a single boot image stored in a single external SPI Flash device. This solution decreases cost, reduces board space, and simplifies field upgrades.

2. Resources

The Avant family is an SRAM-based FPGA. The volatile SRAM configuration memory must be loaded from an external non-volatile memory that can store all the configuration data. The size of the configuration data is based on the amount of logic available in the FPGA and number of pre-initialized Embedded Block RAM (EBR) components. An Avant design using the largest device, with every EBR pre-initialized with unique data values and generated without compression requires the largest amount of storage. The minimum SPI Flash densities required to support the different configuration boot modes are listed in [Table 2.1](#) and [Table 2.2](#).

Table 2.1. Maximum Configuration Bitstream Size – Single Bitstream Boot Mode

Device	Configuration	Uncompressed ¹	
		Unencrypted/Encrypted Single Bitstream Size (Mb)	SPI Mode Minimum SPI Flash Size (Mb)
LAV-AT-500E	No EBR	35.947	64
	MAX EBR	132.448	256

Notes:

1. The Avant family of devices support bitstream compression. Compression ratio depends on the bitstream. Therefore, Table 2.1 only provides uncompressed bitstream data.
2. Both unencrypted and encrypted bitstream are the same size.

Table 2.2. Maximum Configuration Bitstream Size – Dual Boot Mode

Device	Configuration	Uncompressed ¹		SPI Mode
		Unencrypted/Encrypted Single Bitstream Size (Mb)	Unencrypted/Encrypted Two Bitstreams Size (Mb)	Minimum SPI Flash Size (Mb)
LAV-AT-500E	No EBR	35.947	71.894	128
	MAX EBR	132.448	264.896	512

Notes:

1. The Avant family of devices support bitstream compression. Compression ratio depends on the bitstream. Therefore, Table 2.2 only provides uncompressed bitstream data.
2. Both unencrypted and encrypted bitstream are the same size.

3. Avant Dual Boot Mode

Avant Dual Boot mode supports booting from two configuration patterns that reside in an external SPI Flash device. One pattern is designated as the Primary pattern, and the second pattern is designated as the Golden pattern. When Avant device boots up, it attempts to boot from the Primary pattern. If loading of the Primary pattern fails, the Avant device boots from the Golden pattern.

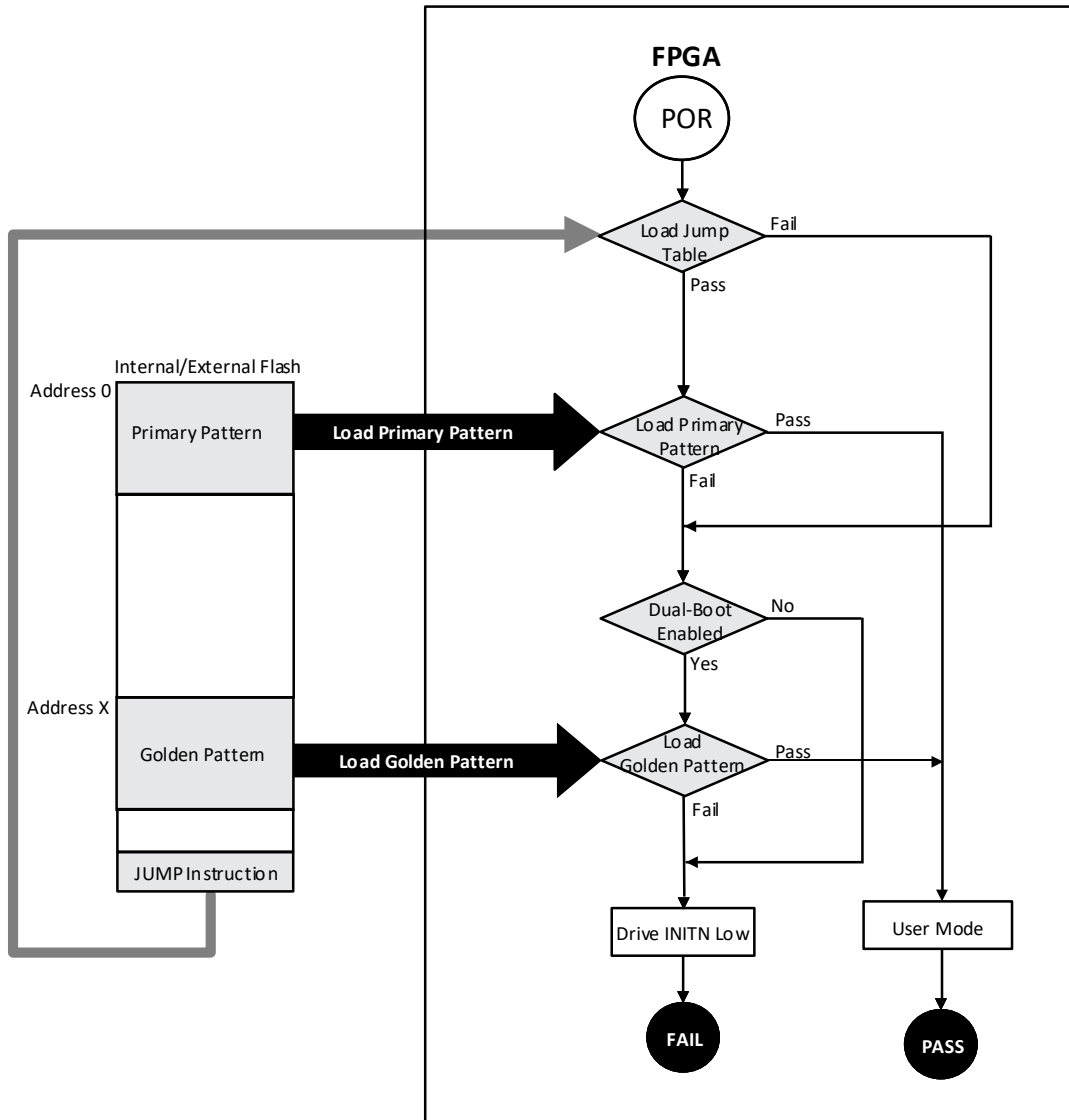


Figure 3.1. Avant Family Dual Boot Flow Diagram

3.1. Description of the Avant Dual Boot Flow Diagram

This flow is triggered either by power cycle, the PROGRAMN pin being toggled, or by the REFRESH instruction being received.

When the Dual Boot mode is selected, in addition to the standard CRC check, a time-out check is performed while reading the Primary pattern, the Golden pattern, and the JUMP command.

- Time-Out Check — Avant device searches for the preamble code 0xBDCD (0xBDB3 when Byte Wide Bit Mirror is enabled) from the Primary Pattern as part of the configuration protocol. The number of retries the Avant device attempts is defined in Control Register 1 [3:2] (Table 3.1).

- Data Corruption Check — After the detection of the preamble code, the CRC engine is turned on to detect whether or not the bitstream is corrupted. This determines whether the Flash device has a corrupted Primary pattern or Golden pattern due to Flash program disruption or data loss.

Table 3.1. Control Register 1 [3:2] – Master Preamble Timer Retry Value

	Bit 3	Bit 2
No retry	0	0
Retry 1 time	0	1
Retry 3 times	1	0
Reserved	1	1

If the Primary pattern fails one of the two checks above, the Avant device knows that the Primary pattern is not valid. It drives the INITN pin LOW briefly to indicate an error and resets the configuration engine. After clearing all the SRAM fuses, it drives the INITN pin HIGH, and reads the JUMP command that directs it to the location of the Golden pattern in the Flash.

If the JUMP command is corrupted, it also causes a configuration failure. It is important to note that a corrupted Golden pattern is not the only possible cause for Dual Boot configuration failure.

If the JUMP command is valid, the Avant device stops the SPI clock, drives the INITN pin LOW, resets the configuration engine, and performs a Clear All operation. The device then drives the INITN pin HIGH after the completion of the Clear All action, restarts the SPI clock, and reads the Golden pattern from the Flash address designated in the JUMP command.

Avant device performs the same time-out check and CRC check when searching for the preamble code from the Golden pattern. If the Golden pattern is also corrupted, configuration fails. The Avant device stops driving the SPI clock, and the INITN pin is driven LOW.

4. Creating a PROM File

The dual boot feature on the Avant family of devices is simple, requiring only one external SPI Flash device, and flexible, due to the intelligent use of the JUMP command. The Lattice Radiant software provides a turn-key solution to implement this feature. The Lattice Deployment Tool, part of Lattice Radiant Software, merges the different patterns and the JUMP command into one PROM hex file. The PROM hex file can later be programmed into an external Flash device using Radiant Programmer or a third-party programmer.

4.1. Using Radiant Deployment Tool to Create a Dual Boot PROM Hex File

The following steps provide the procedure of generating a Dual Boot PROM hex file using the Radiant Deployment Tool.

- Generate the Golden and Primary bitstream files in Lattice Radiant Software.
 - Golden bitstream file MCCLK_FREQ (SPI Master Clock Frequency) setting should not exceed the external Flash device normal/standard read speed.
 - MCCLK_FREQ can be configured using the Global tab of the Device Constraint Editor in Lattice Radiant software.
- Invoke Lattice Radiant Deployment Tool by choosing **Tools > Deployment Tool** from Lattice Radiant Programmer.
- In the Radiant Deployment Tool window, select External Memory as the Function Type and select Dual Boot as the Output File Type (Figure 4.1).
- Select OK.

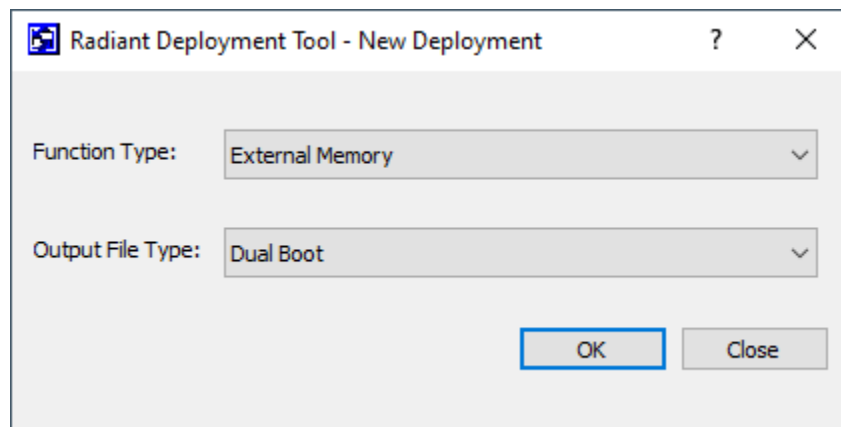


Figure 4.1. Creating New Deployment for Dual Boot PROM Hex File

Step 2 of 4: Dual Boot Options window (Figure 4.3)

- Select the Output Format (Intel Hex, Motorola Hex or Extended Tektronix Hex).
- Select the SPI Flash Size (4, 8, 16, 32, 64, 128, 256, 512, or 1024 Mb).
- Select SPI Flash Read Mode (Standard Read, Fast Read, Dual I/O SPI Flash Read or Quad I/O SPI Flash Read).
- The Radiant Deployment Tool automatically assigns the bitstream files selected in Step 1 to be used for the Golden pattern and Primary pattern.
 - Change the pattern options by clicking on the drop down menu of the respective fields.
 - The Starting Address of the Golden pattern is automatically assigned.
 - Change the Starting Address of the Golden pattern by clicking on the drop down menu.
- Select the following options as required.
 - Byte Wide Bit Mirror – Flips each byte in Intel, Extended Tektronix, or Motorola hexadecimal data files.
For example, 0xCD (b1100 1101) becomes 0xB3 (b1011 0011) when this is selected.
 - Retain Bitstream Header – By default, Radiant Deployment Tool replaces the bitstream header information (name, version number and date of the file) with 0xFF values.
Selecting this option retains the header information that was generated as the header.
 - Optimize Memory Space – When checked, it uses the bitstream file size instead of the worst case bitstream size.
- Select Next.

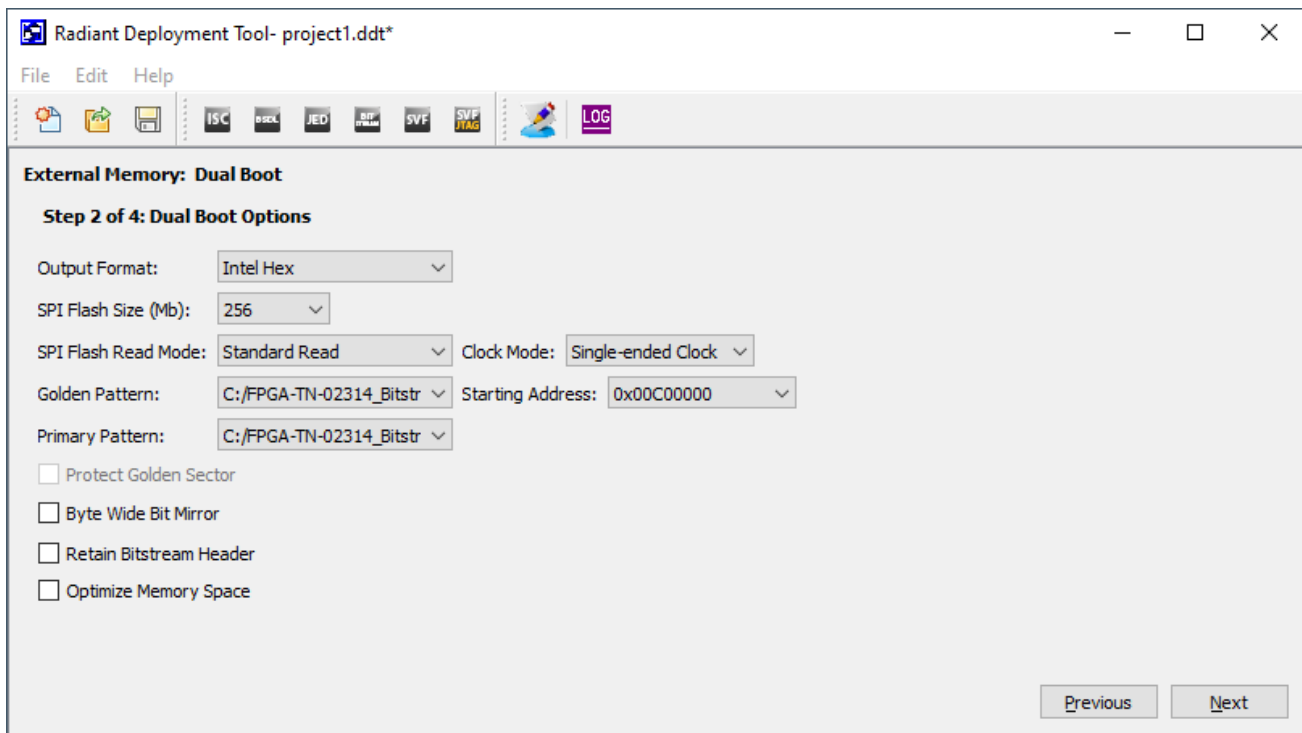


Figure 4.3. Dual Boot Options Window

Step 3 of 4: Select Output File(s) window (Figure 4.4)

- Specify the name of the output PROM hex file in the Output File 1 field.
- Select Next.

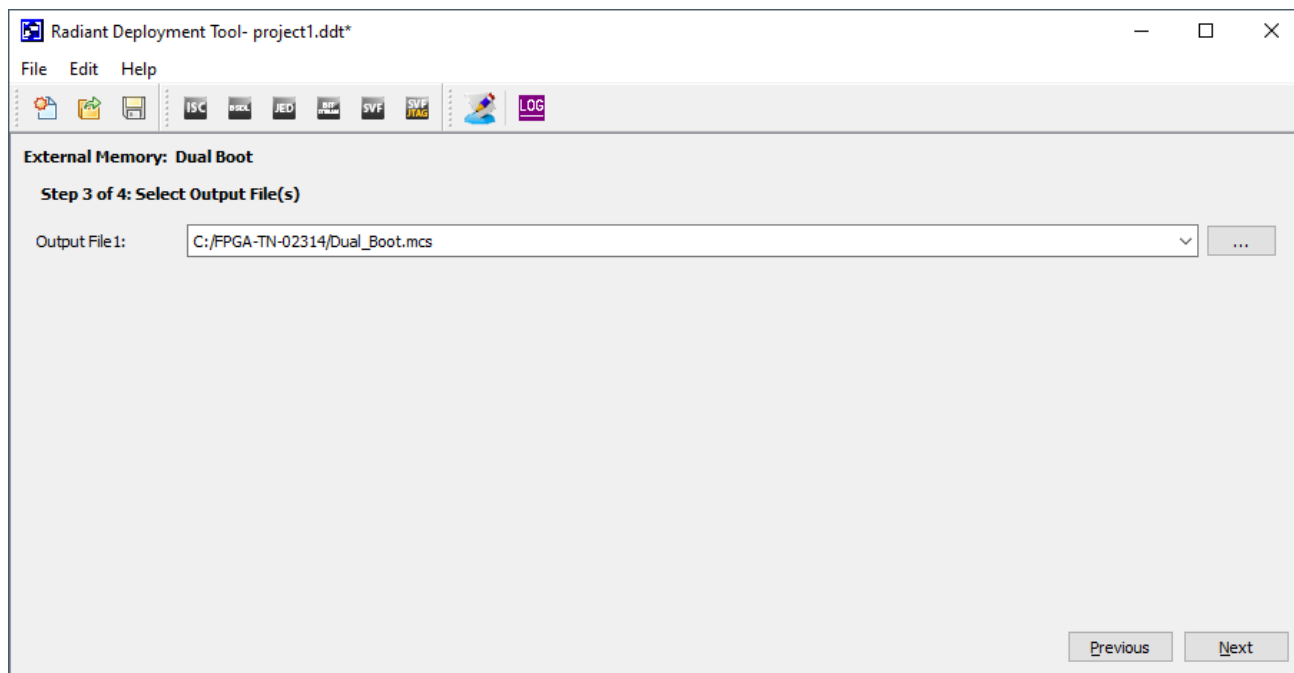


Figure 4.4. Select Output File Window

Step 4 of 4: Generate Deployment window (Figure 4.5)

- Review the summary information.
- If everything is correct, click the Generate button.
- The Generate Deployment pane should indicate that the PROM file was generated successfully.
- Save the deployment settings by selecting **File > Save**.
- To exit, select **File > Exit**.

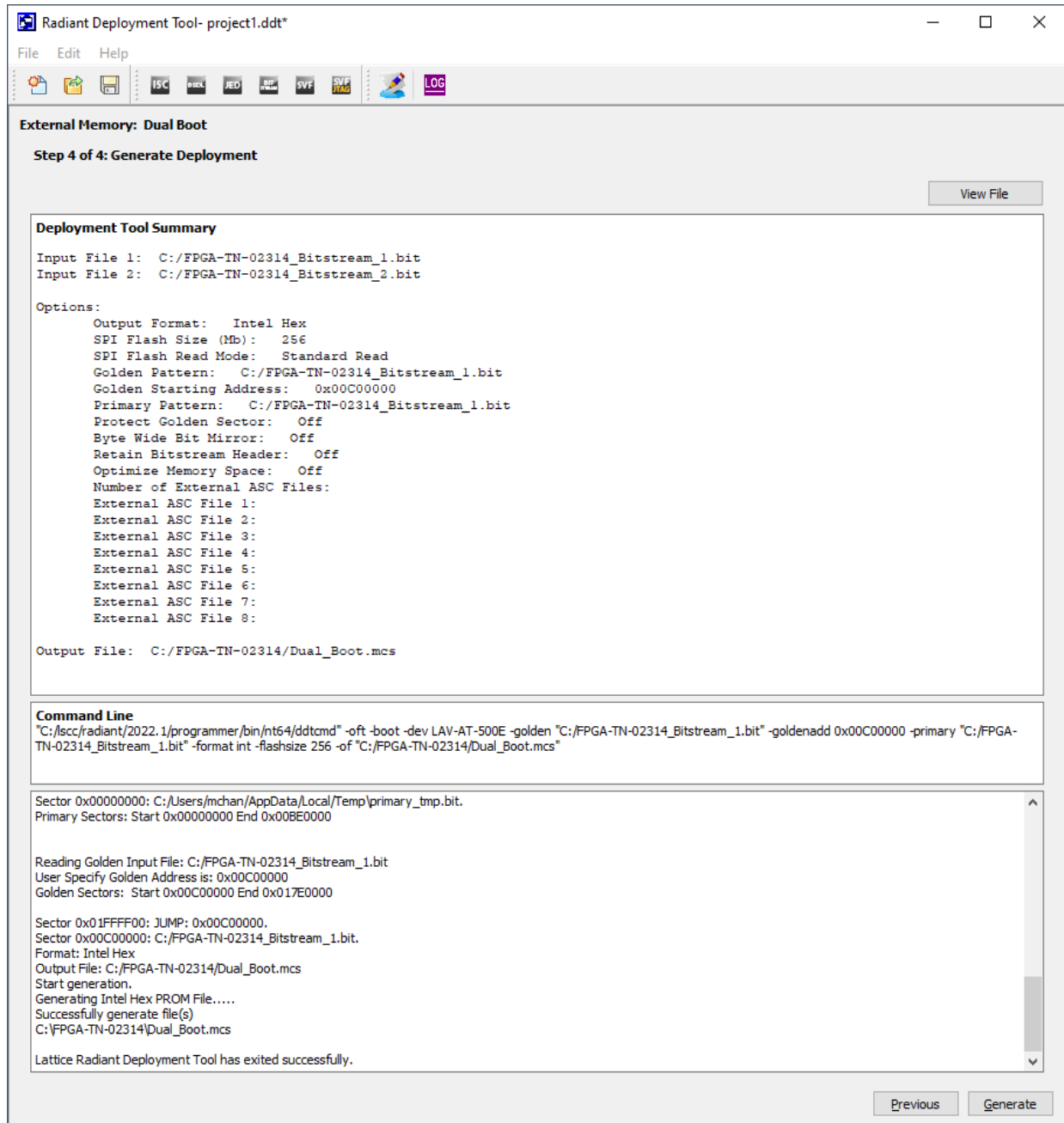




Figure 4.5. Generate Deployment Window

5. Programming the Dual Boot Pattern into the SPI Flash Device

The following procedure is for Programming a Dual Boot Pattern into the SPI Flash Device using Radiant Programmer:

1. Connect power to the board and connect a download cable from the board to the PC.
2. Invoke Radiant Programmer using one of the following methods:
 - In Radiant Software window, select **Tools > Programmer**;
 - In Radiant Software window, select the Programmer icon () in the Radiant toolbar;
 - In the Windows Start menu, select Start > Lattice Radiant Software 2022.1 > Radiant Programmer;
 - In the Windows Start menu, select Start > Lattice Radiant Software 2022.1 > Radiant Software > Radiant Programmer.
3. Radiant Programmer – Getting Started window opens ([Figure 5.1](#)).
 - Select Create a New Project from a Scan, or Create a new blank project, or Select Open an existing Programmer project.
 - Select Detect Cable to scan the PC to determine what cable is connected. Or manually select the type of Cable and Port.
 - Select OK.
4. Select the Operation field by moving the cursor over it and double clicking the left mouse button.
5. The Device Properties window opens.
 - For Target Memory, select External SPI Flash Memory (SPI Flash).
 - For Port Interface, select JTAG2SPI.
 - For Access Mode, select Direct Programming.
 - For Operation, select Erase, Program, Verify.
 - For Programming File, browse to select the .mcs file.
 - In the SPI Flash Options field, specify the Family, Vendor, Device, and Package of the Flash device used on the board.
 - For Data File Size (Bytes), click on the Load from File button.
 - Select the OK button.
6. Program the external Flash device with one of the following methods:
 - In the Radiant Programmer window, select Run > Program Device.
 - In the Radiant Programmer window, select the Program Device icon () in the toolbar.

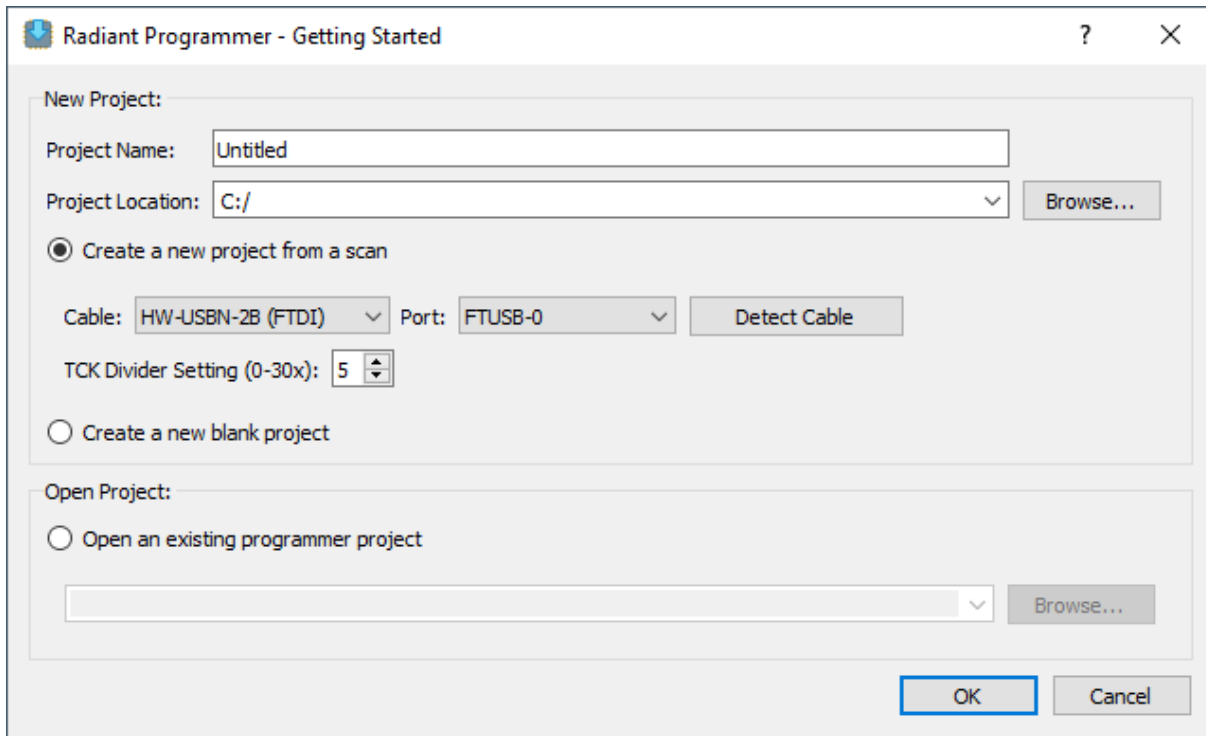


Figure 5.1. Radiant Programmer – Getting Started Window

Technical Support Assistance

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

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

Revision History

Revision 0.80, November 2022

Section	Change Summary
All	Initial Preliminary release.



www.latticesemi.com