JTAG SPI Flash Loader Core User Guide

UG-FLASHCTRL-v3.6
September 2023
www.elitestek.com

Contents

Introduction

The JTAG SPI Flash Loader core provides access to the SPI flash memory using a computer connected to your board via a JTAG header. With JTAG SPI Flash Loader core you can use the JTAG interface to program configuration bitstream into the SPI flash memory. After programming the SPI Flash, The FPGA can be reconfigured with SPI Active mode. This method is useful for reducing the complexity of the hardware design and supports design updates in the SPI Flash device using only the JTAG header.

Use the IP Manager to select IP, customize it, and generate files. The JTAG SPI Flash Loader core has an interactive wizard to help you set parameters. The wizard also has options to create a testbench and/or example design targeting an 易灵思® development board.

Features

- Supports up to 100 256-bytes of write burst
- Supports up to 100 256-bytes of read burst
- Verilog RTL and simulation testbench
- Includes example designs targeting the:
  - Trion® T20 BGA256 Development Board
  - 钛金系列 Ti60 F225 Development Board
  - 钛金系列 Ti60 F100S3F2 System-in-package

Device Support

Table 1: JTAG SPI Flash Loader Core Device Support

FPGA Family	Supported Device
Trion	All Not including BGA49 packages
钛金系列	All

Resource Utilization and Performance

Note: The resources and performance values provided are based on some of the supported FPGAs. These values are just guidance and change depending on the device resource utilization, design congestion, and user design.

Table 2: 钛金系列 Resource Utilization and Performance

FPGA	Logic and Adders	Flip-flops	Memory Blocks	DSP48 Blocks	f_{MAX} (MHz)(1)	Efinity® Version(2)
Ti60 F225 C4	2271	843	32	0	179	2023.1

Table 3: Trion Resource Utilization and Performance

FPGA	Logic Utilization (LUTs)	Registers	Memory Blocks	Multipliers	f_{MAX} (MHz)(1)	Efinity® Version(2)
T20 BGA256 C4	2161	847	64	0	78	2023.1

Release Notes

You can refer to the IP Core Release Notes for more information about the IP core changes. The IP Core Release Notes is available in the Efinity Downloads page under each Efinity software release version.

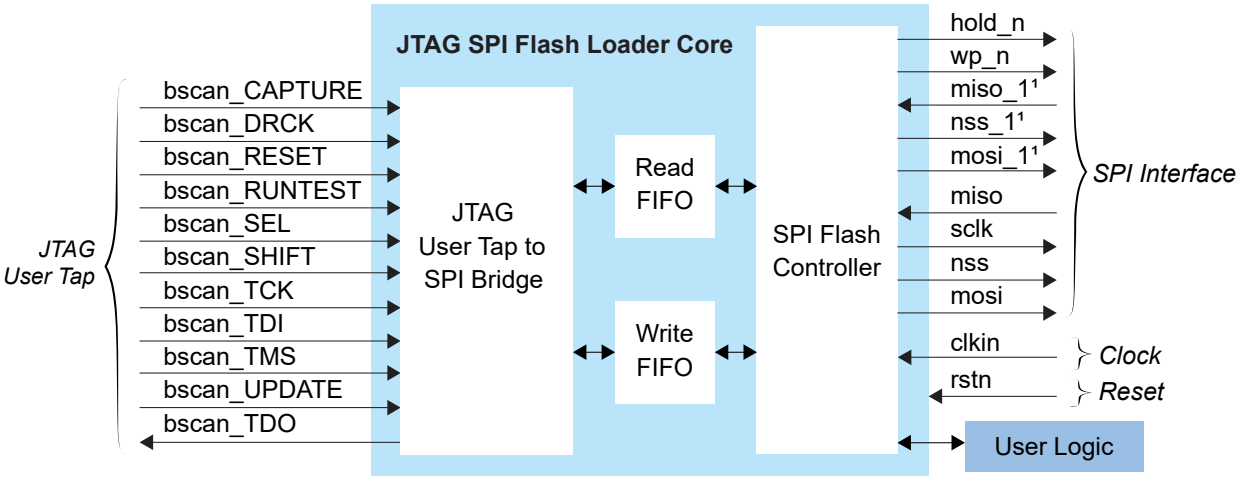
Note: You must be logged in to the Support Portal to view the IP Core Release Notes.

(1) Using default parameter settings.
(2) Using Verilog HDL.

Functional Description

The JTAG SPI Flash Loader core has a JTAG User Tap block interface, and clock, reset, and SPI signals on dedicated SPI interface pins.

Figure 1: JTAG SPI Flash Loader Core Block Diagram



[1] Applicable in dual flash mode.

Ports

Table 4: JTAG SPI Flash Loader Core Ports

Port	Direction	Description
miso	Input	Data input to FPGA CDI1 from SPI flash memory.
miso_1	Input	Data input to FPGA CDI5 from second SPI flash memory in dual flash mode.
sclk	Output	Clock output from FPGA CCK pin to SPI flash memory. The frequency of sclk is 1/12 of the clkin frequency.
nss	Output	Chip select output from FPGA SSL_N pin to SPI flash memory.
nss_1	Output	Chip select output from FPGA SSU_N pin to second SPI flash memory in dual flash mode.
mosi	Output	Data output from FPGA CDI0 to SPI flash memory.
mosi_1	Output	Data output from FPGA CDI4 to second SPI flash memory in dual flash mode.
hold_n	Output	Used for internal flash memory in F100S3F2 and Q100F3 packages. This output is fixed to 1'b1.
wp_n	Output	Used for internal flash memory in F100S3F2 and Q100F3 packages. This output is fixed to 1'b1.
clkin	Input	User system clock. This clock can be from a PLL or user clock pin. The minimum value depends on the JTAG clock.
rstn	Input	Recommended, but optional, reset pin.
bscan_CAPTURE	Input	JTAG. Capture output from the TAP controller.
bscan_DRCK	Input	JTAG. Gated TCK output.
bscan_RESET	Input	JTAG. Reset output for the TAP controller.
bscan_RUNTEST	Input	JTAG. Output asserted when the TAP controller is in the Run Test / Idle state.
bscan_SEL	Input	JTAG. USER instruction active output.
bscan_SHIFT	Input	JTAG. SHIFT output from TAP controller.

Port	Direction	Description
bscan_TCK	Input	JTAG test clock input (TCK).
bscan_TDI	Input	JTAG test data input (TDI).
bscan_TMS	Input	JTAG test mode select input (TMS).
bscan_UPDATE	Input	JTAG. UPDATE output from TAP controller.
bscan_TDO	Output	JTAG test data output (TDO).

IP Manager

The Efinity® IP Manager is an interactive wizard that helps you customize and generate 易灵思® IP cores. The IP Manager performs validation checks on the parameters you set to ensure that your selections are valid. When you generate the IP core, you can optionally generate an example design targeting an 易灵思 development board and/or a testbench. This wizard is helpful in situations in which you use several IP cores, multiple instances of an IP core with different parameters, or the same IP core for different projects.

Generating the JTAG SPI Flash Loader Core with the IP Manager

The following steps explain how to customize an IP core with the IP Configuration wizard.

1. Open the IP Catalog.
2. Choose **Memory Controllers** > **JTAG SPI Flash Loader** core and click **Next**. The **IP Configuration** wizard opens.
3. Enter the module name in the **Module Name** box.

   Note: You cannot generate the core without a module name.
4. Customize the IP core using the options shown in the wizard. For detailed information on the options, refer to the **Customizing the JTAG SPI Flash Loader** on page 8 section.
5. In the **Deliverables** tab, specify whether to generate an IP core example design targeting an 易灵思® development board and/or testbench. These options are turned on by default.

   Note: You can migrate the provided example designs to quickly create a JTAG SPI Flash Loader for your own board too. See **Using the JTAG SPI Flash Loader Core with Your Board** on page 10 for more information.
6. (Optional) In the **Summary** tab, review your selections.
7. Click **Generate** to generate the IP core and other selected deliverables.
8. In the **Review configuration generation** dialog box, click **Generate**. The Console in the **Summary** tab shows the generation status.

   Note: You can disable the **Review configuration generation** dialog box by turning off the **Show Confirmation Box** option in the wizard.
9. When generation finishes, the wizard displays the **Generation Success** dialog box. Click **OK** to close the wizard.

The wizard adds the IP to your project and displays it under **IP** in the Project pane.

Generated Files

The IP Manager generates these files and directories in the **<project>/ip/<module name>** directory:

- **<module name>_define.vh**—Contains the customized parameters.
- **<module name>_tmpl.v**—Verilog HDL instantiation template.
- **<module name>_tmpl.vhd**—VHDL instantiation template.
- **<module name>.v**—IP source code.
- **settings.json**—Configuration file.
- **<kit name>_devkit**—Has generated RTL, example design, and Efinity® project targeting a specific development board.
- **Testbench**—Contains generated RTL and testbench files.

Customizing the JTAG SPI Flash Loader

The core has parameters so you can customize its function. You set the parameters in the General tab of the core's IP Configuration window.

Table 5: JTAG SPI Flash Loader Core Parameters

Parameter	Options	Description
FPGA Version Number	1 - 255	Specifies the FPGA version number. Default: 1
Write Dual Clock FIFO Depth	1 - 100	Defines the write dual clock FIFO depth in the multiple of 256. Indicates the maximum number of write burst per 256 bytes of data to the flash. Higher value indicates faster overall write speed. Default: 60[3] Maximum settings for Trion T4 = 8 Maximum settings for Trion T8 = 16
Read Dual Clock FIFO Depth	1 - 100	Defines the read dual clock FIFO depth in the multiple of 256. Indicates the maximum number of read burst per 256 bytes of data from the flash. Higher value indicates faster overall read speed. Default: 60[3] Maximum settings for Trion T4 = 8 Maximum settings for Trion T8 = 16
Enable Dual Flash Programming	Yes, No	Enables the dual flash mode. JTAG SPI Flash Loader core programs a bitstream file into two SPI flash devices. Select the **SPI Active x8 using JTAG Bridge** mode in the Efinity Programmer software if you enable this option. Applicable in 钛金系列 FPGAs that support dual flash mode (FPGAs with GPIOx_x_xx_SSU_N pin) only. Default: No

[3] You must change the default value to be within the supported range when using a T4 or T8 Trion® FPGA.

JTAG SPI Flash Loader Example Design

You can choose to generate the example design when generating the core in the IP Manager Configuration window. Compile the example design project and download the **.bit** file to your board.

Important: 易灵思 tested the example design generated with the default parameter options only.

The example designs target the:

- Trion® T20 BGA256 Development Board (**T20F256_devkit**)
- 钛金系列 Ti60 F225 Development Board (**Ti60F225_devkit**)
- 钛金系列 Ti60 F100 system-in-package FPGA (**Ti60F100_devkit**)

Note: You can migrate the provided example designs to quickly create a JTAG SPI Flash Loader for your own board. However, some settings must be modified before you generate or compile the JTAG SPI Flash Loader core example design. See **Using the JTAG SPI Flash Loader Core with Your Board** on page 10 for more information.

Using the JTAG SPI Flash Loader Core with Your Board

Use the Trion® T20 BGA256 Development Board example design as a starting point if you have a Trion FPGA on your board and the 钛金系列 Ti60 F225 Development Board example design if you have a 钛金系列 FPGA on your board. Then migrate the design to your targeted FPGA.

Learn more: Refer to the Trion Hardware Design Checklist and Guidelines or 钛金系列 Hardware Design Checklist and Guidelines for the SPI Active configuration pin circuitries.

Check the following settings before compiling the JTAG SPI Flash Loader core example design:

Note: In addition to these settings, you need to verify that 易灵思 supports the flash device you want to use. See **Supported Flash Devices** on page 12.

FIFO Depth Setting (Trion T4 and T8 FPGAs only)

Verify that the **Write Dual Clock FIFO Depth** and **Read Dual Clock FIFO Depth** settings are based on the supported value for your FPGA. See **Customizing the JTAG SPI Flash Loader** on page 8 for the supported values.

Clock Source Assignment

Verify and assign the PLL `clkin` resources based on your board.

Dual Flash Support

If you have dual flash on your board, you can enable the **Enable Dual Flash Programming** before you generate the JTAG SPI Flash Loader example design. Additionally you need to select the **SPI Active x8 using JTAG Bridge** option when programming your board using the Efinity Programmer.

SPI Flash Resource Assignments

Verify that the SPI flash resource assignments are as follows:

Table 6: SPI Flash Resource Assignments

SPI Flash Pin	Trion Resource	钛金系列 Resource
MOSI	CDI0	CDI0
MISO	CDI1	CDI1
NSS	SS_N	SSL_N
SCK	CCK	CCK
MOSI_1	–	CDI4[4]
MISO_1	–	CDI5[4]
NSS_1	–	SSU_N[4]

Learn more: Refer to the device Pinouts for the actual pin names.

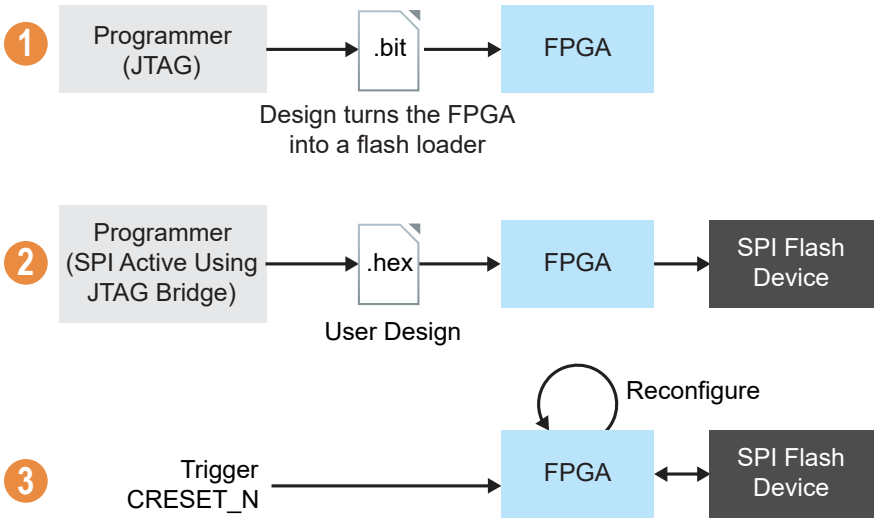
[4] Required only when using the SPI Active x8 using JTAG Bridge programming mode.

Programming the Board

Before programming your board with the example design:

- Open the generated example design project file and compile the project to generate the flash loader file (**jtag_spi_flash_loader.bit**).
- Prepare a configuration bitstream, also refered to as user design bitstream file (**.hex**) to be programmed to the board.

Figure 2: SPI Active Using JTAG Bridge Programming Flow



Note: Although the JTAG SPI Flash Loader writes data to the flash serially (x1), the FPGA can be reconfigured using x1, x2, x4, or x8 depending on the **Project Editor** > **Bitstream Generation** > **Programming Mode** settings. Refer to AN 006: Configuring Trion FPGAs or AN 033: Configuring 钛金系列 FPGAs for more information.

1. Configure the FPGA into a flash loader:
   a. In the Efinity Programmer, select **JTAG** as the **Programing Mode**.
   b. Under **Image**, click the **Select Image File** button and select **jtag_spi_flash_loader.bit**
   c. Click the **Start Program** button.
2. Programming the SPI flash device through the flash loader:
   a. In the Efinity Programmer, select the **SPI Active using JTAG Bridge** Programming Mode.
   b. Under **Image**, click the **Select Image File** button and select your design file (**.hex**).
   c. Disable the **Auto configure JTAG Bridge Image** option.
   d. Ensure that the **Starting Flash Address** is set to **0x000000**.
   e. Click the **Start Program** button.
3. Toggle `CRESET_N` to reconfigure the FPGA with the new user design programmed in the SPI flash.

Note: You can also combine steps **1** and **2** by using the **SPI Active using JTAG Bridge** with the **Auto configure JTAG Bridge Image** option enabled.

JTAG SPI Flash Loader Testbench

You can choose to generate the testbench when generating the core in the IP Manager Configuration window.

Note: You must include all **.v** files generated in the **/testbench** directory in your simulation.

易灵思 provides a simulation script for you to run the testbench quickly using the Modelsim software. To run the Modelsim testbench script, run `vsim -do modelsim.do` in a terminal application. You must have Modelsim installed on your computer to use this script.

The testbench performs ten iterations of write and read tests. The written data to the flash loader is then compared with the data read from the flash memory. Additionally, it indicates an overall pass/fail for the entire test.

To enable the dual flash simulation, you need to add `` `define DUAL_FLASH `` at the top of the testbench module. Example:

```
`timescale 1ns/1ps
`include "test_defines.v"
`include "dbg_defines.v"
`define DUAL_FLASH

module tb_efx_spi_loader_top
  ();
```

After running the simulation, the test prints the following message:

```
Total Test Error(s) :           0
PASS
```

Supported Flash Devices

Table 7: Supported Flash Devices

Manufacturer	Family Part Number
GigaDevice	GD25Q, GD25WQ, and GD25LQ
Macronix	MX25L, MX25U, MX25V, MX75L, and MX75U
Puya Semiconductor	P25Q
Winbond	W25Q
Micron	M25P and MT25Q
XTX	XT25F
Atmel (Adesto Technologies)	AT25SF
ISSI	IS25LP128

Note: 易灵思 recommends using SPI NOR flash memories.

Revision History

Table 8: Revision History

Date	Version	Description
September 2023	3.6	Corrected typo SPI Flash Resource Assignments and updated the resource names. (DOC-1449)
August 2023	3.5	Updated Using the JTAG SPI Flash Loader Core with Your Board and Example Design topics. (DOC-1142) Added Device Support and release notes sections. (DOC-1234)
February 2022	3.4	Updated supported flash devices. (DOC-896) Added note about the The resorce and performance values in the resource and utilization table are for guidance only.
January 2022	3.3	Updated resource utilization table. (DOC-700)
December 2021	3.2	Added Macronix MX75L and MX75U to supported flash devices. (DOC-573) Added support for Ti60 F100 FPGA.
October 2021	3.1	Added note to state that the f_{MAX} in Resource Utilization and Performance, and Example Design Implementation tables were based on default parameter settings. Updated design example target board to production 钛金系列 Ti60 F225 Development Board and updated Resource Utilization and Performance, and Example Design Implementation tables. (DOC-553) Added XT25F family to list of supported flash devices. (DOC-529)
June 2021	3.0	Added note about including all **.v** generated in testbench folder is required for simulation. Added dual flash programming mode support; ports and parameters. Updated for Efinity v2021.1.
December 2020	2.0	Updated core name to JTAG SPI Flash Loader. Updated user guide for 易灵思® IP Manager which includes added IP Manager topics, updated parameters, and user guide structure. Removed Flash Loader Utility content. The content are moved to a separate application note.
September 2020	1.6	Added supported values for T4 and T8 FPGAs to WFIFO_DEPTH_256 and RFIFO_DEPTH_256 parameters.
July 2020	1.5	Updated Flash Controller Utility system directory. Updated LUTs and f_{MAX} in resource utilization and performance. Added customizing the core topic with parameters. Updated sclk and clkin description.
April 2020	1.4	Corrected GigaDevice supported family (GD25 not GD32).
March 2020	1.3	Updated logic utilization and f_{MAX} in resource utilization and performance table. Added Micron to the table of supported flash devices. Added version 1.2 of the core.
February 2020	1.2	Added GigaDevice to the table of supported flash devices.
February 2020	1.1	The -s option now supports a range from 1 to 100. Updated supported flash device listing. Added table describing which core version works with which Efinity® version. Version 1.1 of the core has improved read and write times.
December 2019	1.0	Initial release.