SPI Flash Memory Controller IP

IP Version: v2.0.0

User Guide

FPGA-IPUG-02134-2.1

July 2025

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 REDUNDANCIES, 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.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Figures

Tables

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
ACK	Acknowledgement
AHB	Advanced High-performance Bus
AHB-L	Advanced High-performance Bus Lite
APB	Advanced Peripheral Bus
CFG	Configure
CPU	Central Processing Unit
CMD	Command
DRC	Design Rule Checking
EBR	Embedded Block RAM
FPGA	Field Programmable Gate Array
GPIO	General Purpose I/O
GUI	Graphical User Interface
ID	Identification
LUT	Look Up Table
PDC	Physical Design Constraint
PIC	Programmable Interrupt Controller
PLL	Phase-Locked Loop
RTL	Register Transfer Level
SDK	Software Development Kit
SoC	System-on-a-Chip
SPI	Serial Peripheral Interface
SRAM	Static Random-Access Memory
UART	Universal Asynchronous Receiver Transmitter

1. Introduction

1.1. Overview of the IP

The Serial Peripheral Interface (SPI) Flash Memory Controller IP Core provides an industry-standard interface between a central processing unit (CPU) and an off-chip SPI flash memory device. The controller has two separate ports: Data Port Advanced High-performance Bus (AHB-Lite) interface and Control Port Advanced Peripheral Bus (APB) interface. Data Port can be used by the CPU to directly read from or write to any memory location within the SPI flash. Control Port can be used to define and set how SPI flash transactions are executed by the controller. The Control Port can also be used to perform commonly used flash operations for example, erase, page program, and read.

1.2. Quick Facts

Table 1.1. Summary of the SPI Flash Memory Controller IP

IP Requirements	Supported FPGA Family	Lattice Avant™, MachXO5™-NX, CrossLink™-NX, Certus™-NX, CertusPro™-NX, Certus-NX-RT, CertusPro-NX-RT
	IP Changes	Refer to the SPI Flash Memory Controller IP Release Notes (FPGA-RN-02039).
Resource Utilization	Targeted Devices	LAV-AT-E70, LAV-AT-G70, LAV-AT-X70, LFMXO5-25, LFMXO5-55T, LFMXO5-100T, LIFCL-17, LIFCL-33, LIFCL-40, LFD2NX-9, LFD2NX-15, LFD2NX-17, LFD2NX-25, LFD2NX-28, LFD2NX-40, LFCPNX-50, LFCPNX-100, UT24C40, UT24CP100
	Supported User Interface	AHB-Lite, APB
	Resources	Refer to Appendix A. Resource Utilization.
Design Tool Support	Lattice Implementation[1]	IP Core v2.0.0 – Lattice Radiant™ Software 2024.2 and Lattice Propel™ Software 2024.2
	Synthesis	Synopsys® Synplify Pro
	Simulation	Refer to the Lattice Radiant Software User Guide for the list of supported simulators.

Note:

1. Lattice Implementation indicates the IP version release coinciding with the software version release. Check the software for IP version compatibility with earlier or later software versions.

1.3. IP Support Summary

Table 1.2. SPI Flash Memory Controller IP Support Readiness

Device Family	IP Configuration and Settings	On-board Flash	Clock Frequency	Flash Settings	Flash Commands	Radiant Timing Model	Hardware Validated
CrossLink-NX	Page Buffer used on APB Interface, Page Program and Read Buffer are enabled, Supported SPI Flash Commands	Macronix 512 Mbit SPI Flash (MX25L51245G)	APB Interface: 30 MHz, SPI Clock: 30 MHz (SCLK_RATE = 0), and 15 MHz (SCLK_RATE = 1)	Standard SPI Clock mode 0 and 3, 3-byte and 4-byte address mode	Block Erase Type 1, 2, and 3, Page Program, Read Data (Normal Read), Fast Read	Final	Yes
	Page Buffer used on AHBL Interface, Page Program and Read Buffer are enabled, Direct Flash Memory Write and Read Access		APB Interface: 30 MHz, AHB-L Interface: 100 MHz, SPI Clock: 30 MHz (SCLK_RATE = 0), and 15 MHz (SCLK_RATE = 1)	Standard SPI Clock mode 0 and 3, 4-byte address mode	Block Erase Type 1, 2, and 3, Page Program, Read Data (Normal Read), Fast Read	Final	Yes
	Page Program and Read Buffer are disabled, Direct Flash Memory Write and Read Access		APB Interface: 30 MHz, AHB-L Interface: 100 MHz, SPI Clock: 30 MHz (SCLK_RATE = 0), and 15 MHz (SCLK_RATE = 1)	Standard SPI Clock mode 0 and 3, 3-byte address mode	Block Erase Type 1, Page Program, Read Data (Normal Read)	Final	Yes
	Page Buffer used on AHBL Interface, Page Program and Read Buffer are enabled, User-Defined Commands			Standard SPI Clock mode 0 and 3, 4-byte address mode	Enable 4-byte Address Mode	Final	Yes
	Page Buffer used on AHBL Interface, Page Program and Read Buffer are enabled, User-Defined Commands		APB Interface: 25 MHz, SPI Clock: 25 MHz (SCLK_RATE = 0)	Standard SPI Clock mode 0 and 3	Read Identification	Final	Yes

1.4. Features

Key features of the SPI Flash Memory Controller IP include:

- Two bus interfaces: Data Port AHB-Lite Subordinate interface and Control Port APB Completer interface
- Option to individually configure the data bus widths of Data Port and Control Port to 8 or 32 bits
- Option to configure controller with Page Program Buffer to speed up Page Program
- Option to configure controller with Page Read Buffer to speed up Page Read
- Configurable serial clock (sclk_o) frequency
- Supports all SPI clocking modes, combing clock polarity and clock phase
- Configurable SPI flash page sizes
- Configurable SPI flash memory map size
- Configurable SPI flash command opcode

1.5. Licensing and Ordering Information

The SPI Flash Memory Controller IP is provided at no additional cost with the Lattice Radiant software.

1.6. Hardware Support

Refer to the Example Design section for more information on the boards used.

1.7. Naming Conventions

1.7.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL.

1.7.2. Signal Names

Signal Names that end with:

- *_n* are active low (asserted when value is logic 0)
- *_i* are input signals
- *_o* are output signals
- *_io* are bi-directional input/output signals

1.7.3. Controller

The logic unit inside the FPGA interacts with the SPI Flash Memory Controller IP Core through either APB or AHB-Lite.

1.7.4. Attribute

The names of attributes in this document are formatted in title case and italicized (Attribute Name).

2. Functional Description

2.1. IP Architecture Overview

The Serial Peripheral Interface flash memory controller provides an industry-standard interface between a CPU and an off-chip SPI flash memory device. AHB-Lite interface can be used to perform direct write and read access to any memory location within the SPI flash memory map size configured through the IP Graphical User Interface (GUI). APB interface, on the other hand, can be used to issue any flash operations from the command opcode set or using the user-defined registers.

This IP receives and processes the data or control port signals, checks the flash operation to be executed, automatically creates the flash command sequence of the supported flash command opcodes, then translates each flash command into serial form to be communicated to the SPI flash.

Figure 2.1 shows the IP functional diagram.

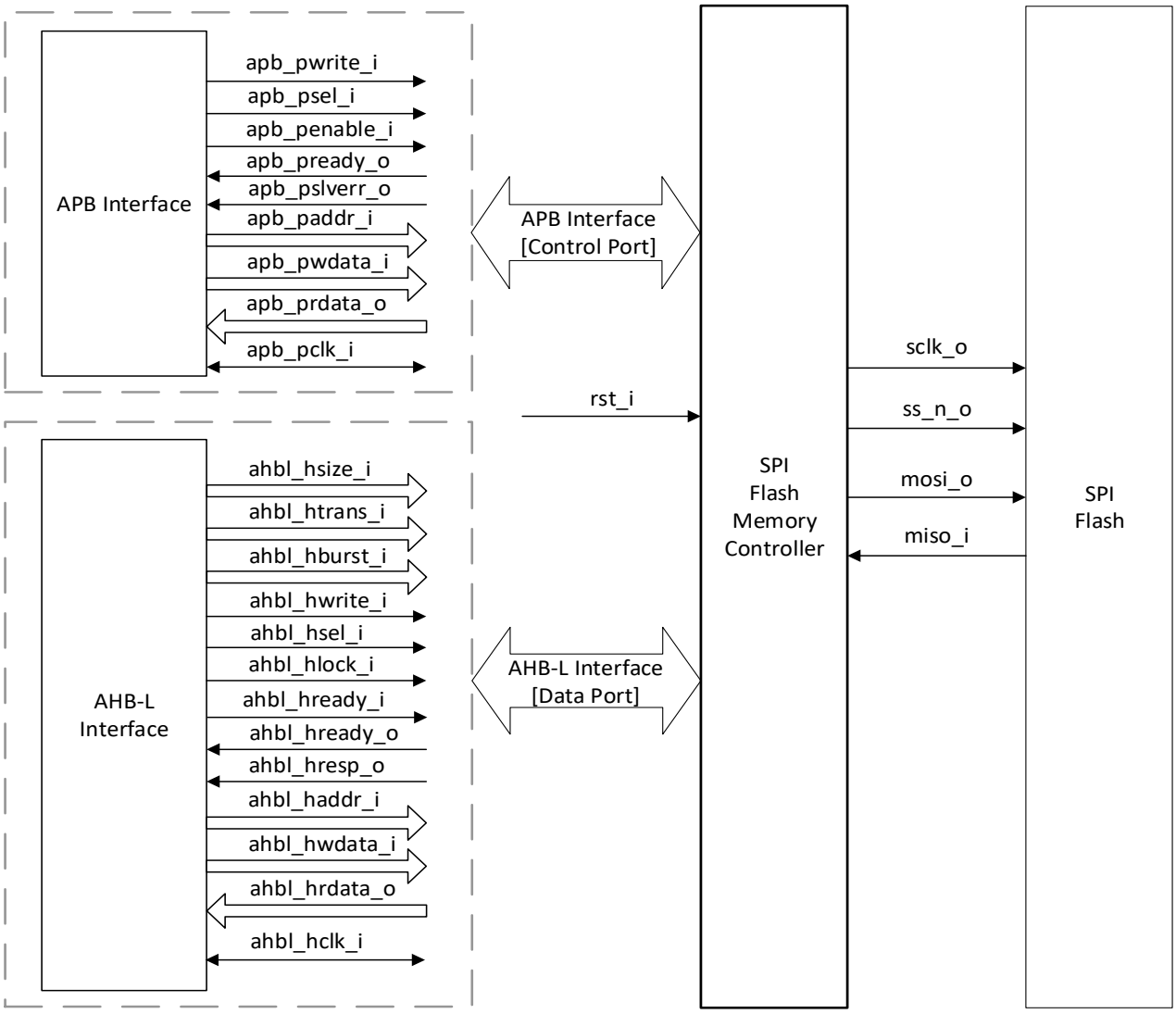

Figure 2.1. SPI Flash Memory Controller IP Core Functional Diagram

2.2. Clocking

There are three clocks for SPI Flash Memory Controller IP.

2.2.1. Clocking Overview

- ahb_hclk_i, AHB-Lite input clock
- apb_pclk_i, APB input clock
- sclk_o, Serial clock to SPI flash

2.3. Reset

There is one reset for SPI Flash Memory Controller IP.

2.3.1. Reset Overview

rst_i is an asynchronous active high reset. When asserted, output ports and registers are forced to their reset values.

2.4. User Interfaces

Table 2.1. User Interfaces and Supported Protocols

User Interface	Supported Protocols	Description
Advanced High-performance Bus Interface	AHB-Lite	The Advanced High-performance Bus Interface is used as the data port for SPI Flash Memory Controller IP. The interface can be used by the CPU to directly read from or write to any memory location within the SPI flash.
Advanced Peripheral Bus Interface	APB	The Advanced Peripheral Bus Interface is used as the control port for SPI Flash Memory Controller IP. The interface can be used to define and set how SPI flash transactions are executed by the controller. The control port can also be used to perform commonly used flash operations for example, erase, page program, and read.
Serial Peripheral Interface	SPI	The Serial Peripheral Interface can complete the communication between SPI Flash Memory Controller IP and SPI flash.

2.5. Other IP Specific Blocks/Layers/Interfaces

The following sections further describes the IP features.

2.5.1. Byte/Halfword/Word Direct Read or Write

This direct read from or write to SPI flash memory can only be initiated on the AHB-Lite interface. The IP only initiates a new read from or write to SPI flash memory when it determines that the SPI flash is not busy processing another command. The SPI flash memory address is obtained from the lower 24 bits or the full 32 bits of the AHB-Lite transaction address. The AHB-L write transaction performs direct write to the SPI flash memory. The IP obtains the write data from the AHB-Lite input data bus. The AHB-L read transaction performs direct read from the SPI flash memory. Valid read data is available on the output data bus of the AHB-Lite data port after read completion. All settings for AHB-L write and read transactions should be set to APB-related control port registers.

The SPI flash is a non-volatile memory and therefore, it is not possible to overwrite a memory location that is previously written without erasing it. To write on the memory, you must issue an erase command through the APB interface.

2.5.2. Erase, Write Enable/Disable, Status Read/Write, Power Up/Down, Read Manufacturer ID

These SPI flash commands can only be performed using the APB interface. An APB read or write transaction can be executed using the corresponding APB offset addresses shown in Table 5.1. If the flash operation does not involve any return data from the SPI flash, the controller transaction status automatically return to idle state as soon as the command is issued to the SPI flash.

Note: For erase and status write supported flash operations, the controller automatically appends the flash write enable command on the sequence before sending the flash command of the operation to be executed. When command and address are already sent for flash erase operations, it also automatically sends the read status register command continuously until the flash device achieves the idle state.

2.5.3. IP Configuration and Control Settings

These commands can only be initiated on the APB interface. They are used by the IP to define the serial data to be sent to the flash. They can be used to perform the following:

- Set the flash addressing mode, 3-byte or 4-byte, and flash address value.
- Control whether the SPI flash read is performed as fast read or slow read.
- Control the SPI flash page program and read size.
- Set the write protect bit.
- Configure the APB clock divider to derive the serial clock frequency.
- Control the shifting order of the SPI bits for flash operations.
- Enable page buffer when AHB-Lite interface is used.
- Set the SPI clock mode.
- Configure the SPI flash command opcode set so that the core can be used with a variety of SPI flash vendors.

2.5.4. SPI Flash Page Read/Program

The process of reading or writing an entire page through the AHB-Lite interface can be slow because each page read/write needs to be split up into multiple AHB-Lite single read/writes. To speed up, buffer can be enabled. Once enabled, multiple AHB-Lite write accesses can be performed specifying the flash address and write data on the bus. After writing all data, they are stored on the buffer and write to flash does not start until the page program is executed through the APB offset address. Likewise, when the buffer is enabled for read and page read is executed through the APB offset address, read data can be continuously read from the AHB-Lite interface using the same flash address specified for write.

For the APB interface, page program buffer offset address should be used to store the write data. Starting flash address should be set on corresponding APB offset address. Once done, page program can already be executed. Page read buffer offset address is used to read all the return data from flash after page read is executed.

Refer to Table 5.1 for the corresponding offset addresses.

Note: For the flash page program, the controller automatically appends the flash write enable command on the sequence before sending the flash command, address and write data. After all the write data are sent by the controller, it also automatically sends the read status register command continuously until the flash device achieves the idle state.

2.5.5. User-Defined Flash Operations

There are some flash commands that the IP does not directly support via APB offset addresses. For these commands, the user-defined command registers can be used. All the data and transaction definitions are set on these registers before starting the transaction. Refer to Table 5.1 for the corresponding offset addresses and sections 5.17 to 5.25 for the detailed descriptions.

2.5.6. IP Transaction Status

Once flash transaction is started on applicable interface, it is essential to track its status. This can be done by reading the available IP transaction status registers right after the flash transaction is started. This can be read multiple times until the IP status is indicated as idle.

After page program is started, the controller sends all the available write data to the flash via SPI. It also automatically polls the flash memory status on programming the data to selected memory location. While the flash memory is still busy, IP transaction status also remains busy until the flash memory status flags a not busy status. The same behavior is observed when an erase transaction is executed. IP transaction status stays busy while erase operation is still on-going on the flash memory.

For flash page read transaction or other flash command with return data, IP transaction status registers available can be used to track when read on flash memory location is still on-going or done and whether there is already flash return data that can be read on the interface. This allows you to either read all return data all at once after transaction is completed or execute a read of data on interface while page read is happening on the background.

After a flash transaction that does not have any write or read data is started, reading an idle status means that the command is already sent by the controller on SPI.

3. IP Parameter Description

3.1. Attributes Summary

The configurable attributes are shown in Table 3.1 and are described in Table 3.2. The attributes can be configured through the IP Catalog Module/IP wizard of the Lattice Radiant software.

Table 3.1. Attributes Summary

Attribute	Selectable Values	Default	Dependency on Other Attributes
Control Port Settings			
Enable Page Program Buffer	Checked, Unchecked	Unchecked	—
Enable Page Read Buffer	Checked, Unchecked	Unchecked	—
Page Program Buffer Memory Type	Distributed Memory, EBR	Distributed Memory	*Enable Page Program Buffer* == Checked
Page Read Buffer Memory Type	Distributed Memory, EBR	Distributed Memory	*Enable Page Read Buffer* == Checked
Page Buffer Interface	APB, AHB	APB	*Enable Page Program Buffer* == Checked or *Enable Page Read Buffer* == Checked
Control Base Address	0x80000000 – 0xFFFFF800	0x80000000	—
SPI Flash Settings			
Page Size	256, 512	256	—
Flash Memory Map Size (KB)	1, 2, 4, 8, …, 2097152	4	—
SCLK Rate	0 – 15	0	—
SCLK Polarity	0, 1	0	
SCLK Phase	0, 1	0	
First Transmitted Bit	MSB, LSB	MSB	—
Data and Control Bus			
Data Port AHB Address Width	24,32	32	—
Data Port AHB Data Width	8, 32	32	—
Data Port AHB Data Byte Endianness	little-endian, big-endian	little-endian	*Data Port AHB Data width = 32*
Control Port APB Address Width	11	11	*Display information only.*
Control Port APB Data Width	8, 32	32	—
SPI Flash Command Opcodes Specify the one-byte instruction Opcodes for each command within the SPI flash instruction set. Refer to the SPI flash data sheet to obtain these values.			
Slow Read	—	8'h03	—
Fast Read	—	8'h0b	—
Page Program	—	8'h02	—
Block Erase Type 1	—	8'h20	—
Block Erase Type 2	—	8'h52	—
Block Erase Type 3	—	8'hd8	—
Chip Erase	—	8'h60	—
Write Enable	—	8'h06	—
Write Disable	—	8'h04	—
Read Status Register	—	8'h05	—
Write Status Register	—	8'h01	—
Deep Power Down	—	8'hb9	—
Resume from Power Down	—	8'hab	—
Read Manufacturer ID	—	8'h9f	—

Table 3.2. Attributes Descriptions

Attribute	Description
Control Port Settings	
Enable Page Program Buffer	This can be enabled, set to 1, to speed up page program. Setting this to 1 indicates that page write data are locally stored before writing the entire page to SPI flash. The size of the buffer is equal to the page size.
Enable Page Read Buffer	Enable Page Read Buffer to speed up page reads by reading an entire page from SPI flash and locally storing it. The size of the buffer is equal to the page size.
Page Program Buffer Memory Type	Indicates whether EBR or Distributed RAM is used as Page Program buffer.
Page Read Buffer Memory Type	Indicates whether EBR or Distributed RAM is used as Page Read buffer.
Page Buffer Interface	Indicates the interface to be used for data buffer setting.
Control Base Address	Specifies the base address for the Control Port APB interface. The minimum boundary alignment is 0x800.
SPI Flash Settings	
Page Size	Specifies the size of each page in the SPI flash. Refer to the SPI flash data sheet to obtain this value.
Flash Memory Map Size (KB)	Indicates the size in Kilobytes (KB) of the flash memory to be mapped. Available options are values on the power of 2. For example, 1, 2, 4, 8 up to 2097152.
SCLK Rate	Specifies the factor for deriving sclk_o from the 0 – 15 component input clock (apb_pclk_i). sclk_o is derived from the following equation: sclk_o = apb_pclk_i/(2 x SCLK_Rate) when SCLK_RATE is greater than 0. If SCLK_RATE is equal to 0 sclk_o = apb_pclk_i. For example: For SCLK Rate = 1, sclk_o = apb_pclk_i/2
SCLK Polarity	Sets the polarity of the sclk_o signal during idle state.
SCLK Phase	Indicates whether the rising or falling clock edge is used to sample and shift the data.
First Transmitted Bit	Specifies the direction of shifting of bits into and out of the SPI interface. See the SPI First Transmitted Bit Register 0x114 section for the complete details.
Data and Control Bus	
Data Port AHB Address Width	Size of AHB-Lite address bus
Data Port AHB Data Width	Size of AHB-Lite data bus
Data Port AHB Data Byte Endianness	Endianness of AHB-Lite data bus. See the AHB-Lite Byte Endianness section for the complete details.
Control Port APB Address Width	Size of APB address bus
Control Port APB Data Width	Size of APB data bus
SPI Flash Command Opcodes	
Generic Flash Command Opcode	Specifies the equivalent opcode of a flash command of the off-chip SPI flash memory device.

3.2. IP Parameter Settings for Example Use Cases

In SPI, the transaction is always initiated by the controller. During the time that ss_n_o is active (LOW), the clock signal (sclk_o) is toggled while command information is first transferred on the data signals from the controller to the SPI flash. The sclk_o signal toggles during any period required for information access in the SPI Flash. The clock continues to toggle during the transfer of read data from the SPI Flash to controller or write data from the controller to the SPI flash. When controller has transferred the desired amount of data, it drives the ss_n_o inactive (HIGH). This section describes more in detail the IP features used on operations.

3.2.1. Clocking Modes (SCLK Polarity and SCLK Phase)

Through the IP GUI and register, SCLK polarity and SCLK phase can be set. The SCLK polarity sets the polarity of the sclk_o signal during the idle state — ss_n_o bit is in inactive. Transitioning of ss_n_o signal from inactive logic to active logic marks the start of the transmission and transitioning from active logic to inactive logic marks the end of the transmission. The SCLK phase selects the sclk_o phase. It sets whether the rising or falling clock edge is used to sample and shift the data. The IP must be set to the corresponding SCLK polarity and phase as per the requirement of the flash. There are four SPI clocking modes available, which are shown in Table 3.3.

Table 3.3. Clocking Modes

Clocking Mode	SCLK Polarity	SCLK Phase	Description
0	0	0	The ss_n_o signal transition to active logic shifts out a data bit. The first clock transition samples the data. Data is sampled on the sclk_o rising edge and is shifted out on the falling edge.
1	0	1	The first clock transition shifts out a data bit and the second clock transition samples the data. Data is sampled on the sclk_o falling edge and is shifted out on the rising edge.
2	1	0	The ss_n_o signal transition to active logic shifts out a data bit. The first clock transition samples the data. Data is sampled on the sclk_o falling edge and is shifted out on the rising edge.
3	1	1	The first clock transition shifts out the data and the second clock transition captures the data. Data is sampled on the sclk_o rising edge and is shifted out on the falling edge.

The sample waveforms for the SPI clocking modes are shown in Figure 3.1, Figure 3.2, Figure 3.3, and Figure 3.4.

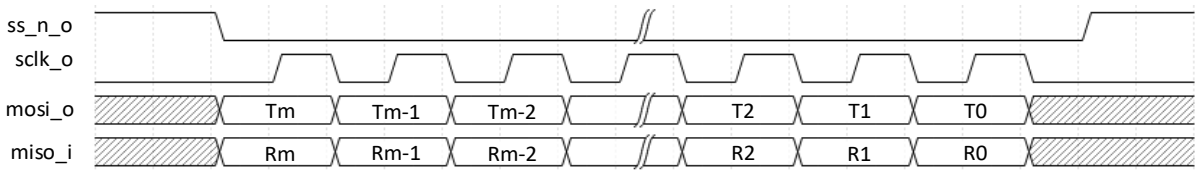

Figure 3.1. Clocking Mode 0 (SCLK Polarity=0, SCLK Phase=0)

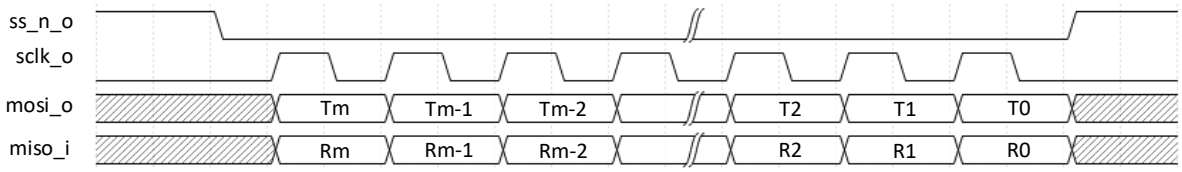

Figure 3.2. Clocking Mode 1 (SCLK Polarity=0, SCLK Phase=1)

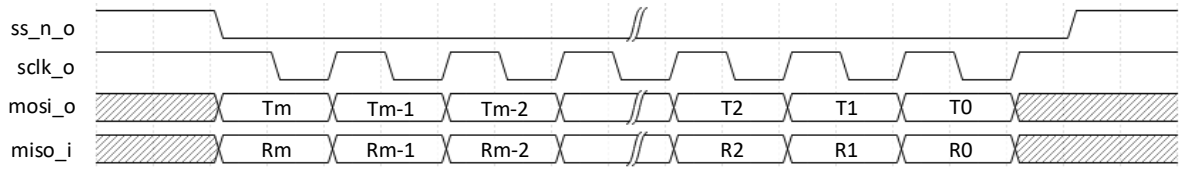

Figure 3.3. Clocking Mode 2 (SCLK Polarity=1, SCLK Phase=0)

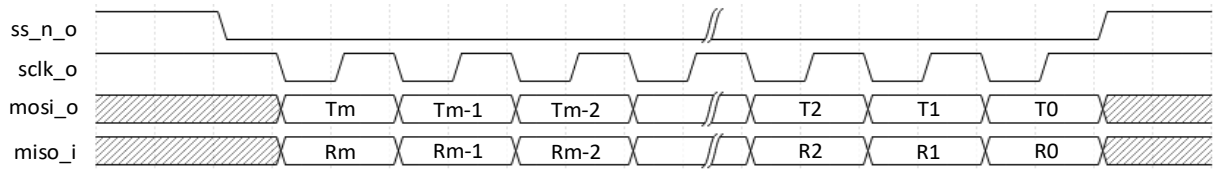

Figure 3.4. Clocking Mode 3 (SCLK Polarity=1, SCLK Phase=1)

3.2.2. SPI First Transmitted Bit

The IP has four transfer phases: command, address, dummy and data. For write operation, only command, address, and data phases are used. For read operation, all phases can be used, and dummy phase depends on the target flash read operation. Table 3.4 shows the supported byte length during each phase of the SPI transaction.

Table 3.4. SPI Transaction Phase and Byte Length

SPI Transaction Phase	Byte Length
Command	1
Address	3 or 4
Dummy	1 to 16
Data	1 to 256 or 512

The endianness of each phase depends on First Transmitted Bit GUI attribute or its corresponding register setting. For data phase, endianness is in byte level. There is no endianness consideration for dummy phase.

Figure 3.5, Figure 3.6, Figure 3.7, and Figure 3.8 show when MSB or LSB is the first transmitted bit during write and read SPI transaction. All sample waveforms have clocking mode 0.

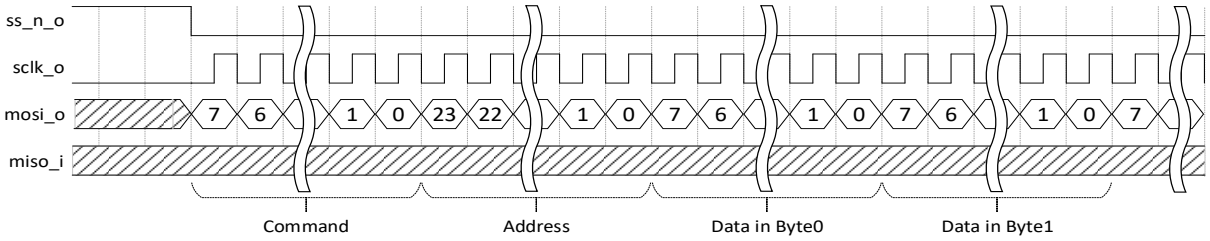

Figure 3.5. SPI Flash Write Transaction with MSB as First Transmitted Bit

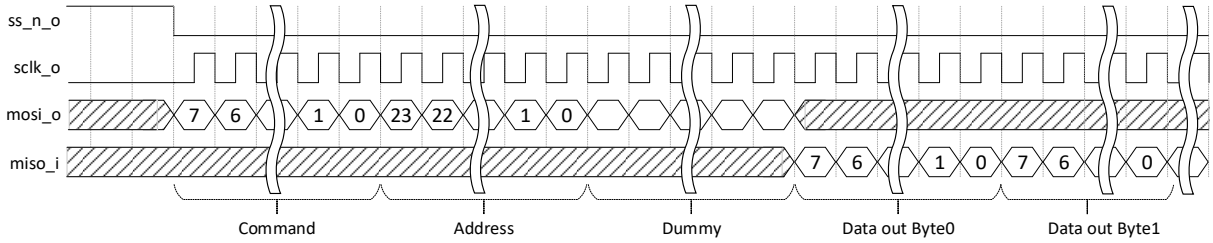

Figure 3.6. SPI Flash Read Transaction with MSB as First Transmitted Bit

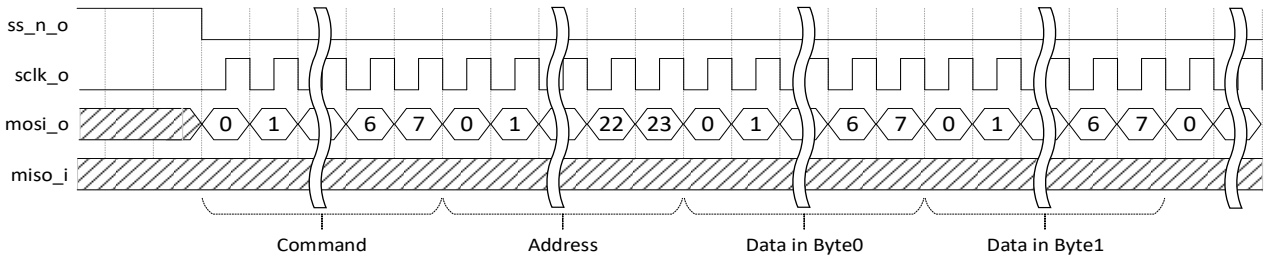

Figure 3.7. SPI Flash Write Transaction with LSB as First Transmitted Bit

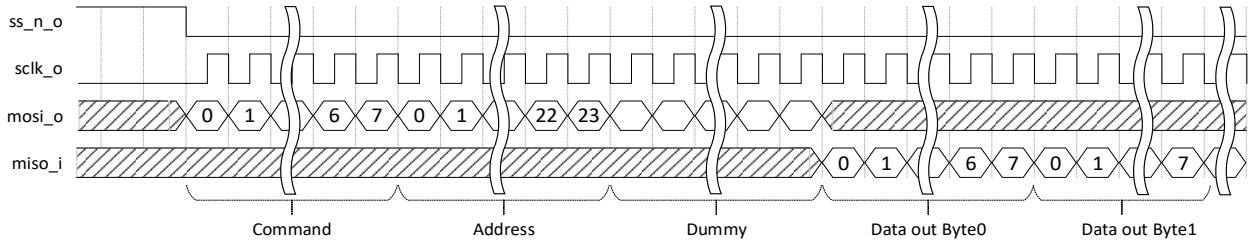

Figure 3.8. SPI Flash Read Transaction with LSB as First Transmitted Bit

3.2.3. AHB-Lite Byte Endianness

Table 3.5 shows the data port of AHB-Lite interface with different endianness. For example, the Data is {Byte0, Byte1, Byte2, Byte3} with an address starting from 0x0.

Table 3.5. AHB-Lite Interface Data Endianness

	Data Stored	
Byte Address	**Little Endian**	**Big Endian**
0x0000_0000	Byte3	Byte0
0x0000_0001	Byte2	Byte1
0x0000_0002	Byte1	Byte2
0x0000_0003	Byte0	Byte3

Figure 3.9 and Figure 3.10 show sample SPI flash write and read transaction with word transfer size. All sample waveforms have clocking mode 0.

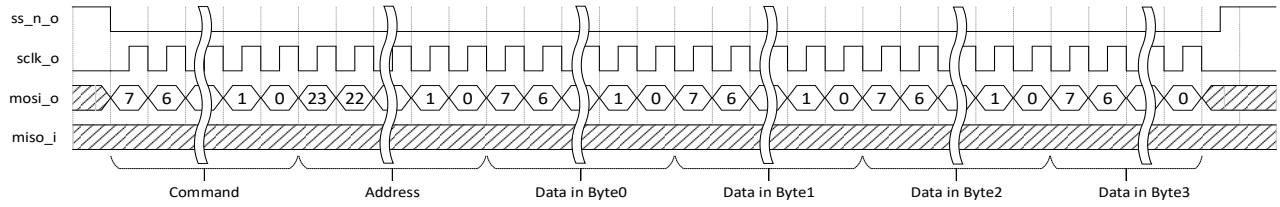

Figure 3.9. SPI Flash Write Transaction

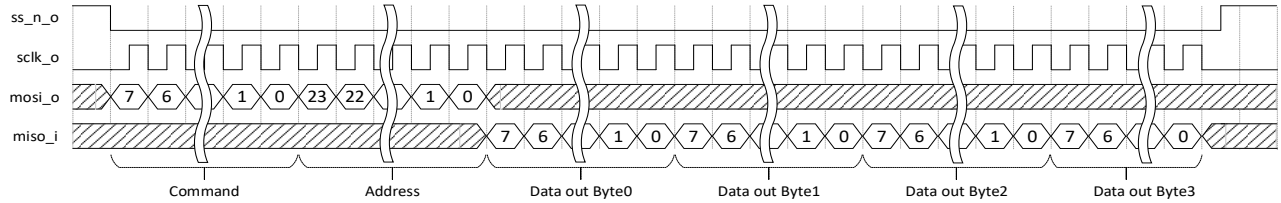

Figure 3.10. SPI Flash Read Transaction

3.2.4. Flash Command

Figure 3.11 shows the resulting SPI write transaction with clocking mode 0 using Block erase type 1 flash command. The following are the APB interface register accesses used to perform this transaction:

1. Register write to '0x040' with '32-bit flash address'.
2. Register write to '0x044' to set the Flash Addressing mode to 32-bits.
3. Register write to '0x008' to initiate Block Erase Type 1.
4. Register read from '0x1c0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation.

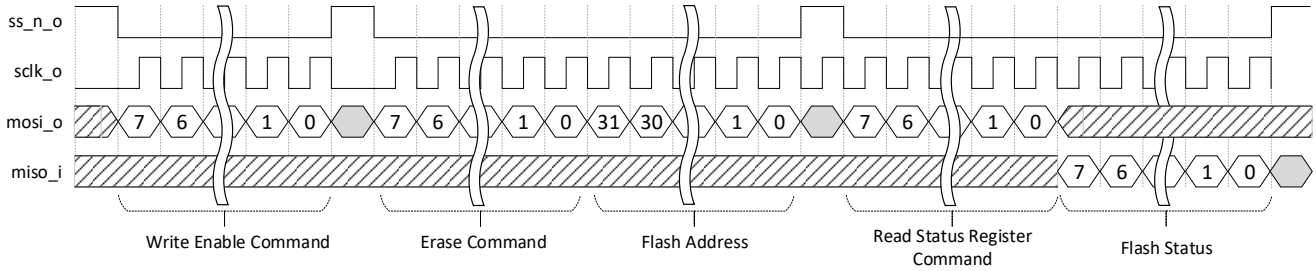

Figure 3.11. SPI Transaction for Block Erase Type 1

Figure 3.12 shows the resulting SPI read transaction with clocking mode 0 using Status read flash command. The following are the APB interface register accesses used to perform this transaction:

1. Register read from '0x020' to read status.
2. Register read from '0x1c0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation.

Note that the Status Read flash command is limited to read 8 bits data only. Use user-defined command for more than 1 byte of returned data.

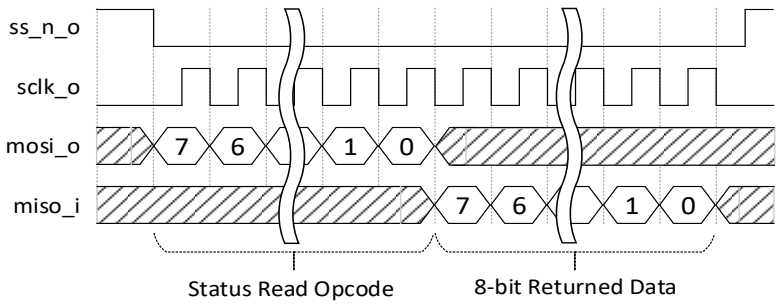

Figure 3.12. SPI Transaction for Status Read

3.2.5. User Command

To send user command to SPI flash memory, user command registers listed in Table 5.1 must be set accordingly.

Below is a sample procedure to perform an SPI flash write transaction.

1. Register write to '0x044' to set the Flash Addressing mode to 32-bit.
2. Register write to '0x080' to set the command opcode to 0x55.
3. Register write to '0x084' to set the address to the target 32-bit flash address.
4. Register write to '0x088' to set the lower 4 bytes of the write data {byte3, byte2, byte1, byte0}.
5. Register write to '0x08c' to set the upper 2 bytes of the write data {byte5, byte4}.
6. Register write to '0x090' to set the data length to 6 bytes.
7. Register write to '0x094' to set the direction control to write and enable the address.
8. Register write to '0x0a0' to start the user initiated command.
9. Register read from '0x1c0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation.

These procedures result to SPI transaction shown in Figure 3.13. The waveform below has clocking mode 0.

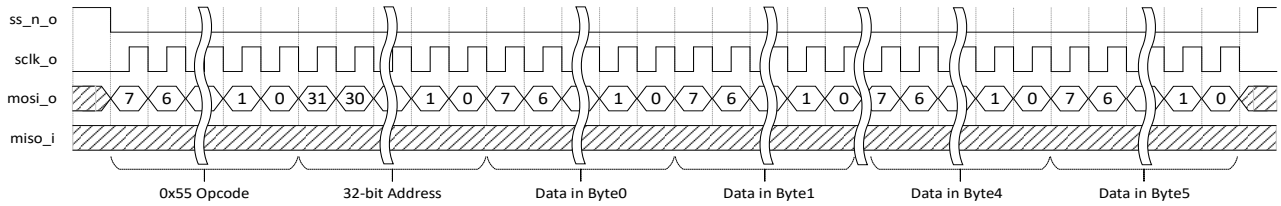

Figure 3.13. SPI Write Transaction using User-Defined Command

Below is a sample procedure to have an SPI flash read transaction.

1. Register write to '0x044' to set the Flash Addressing mode to 24 bits.
2. Register write to '0x080' to set the command opcode to 0x50.
3. Register write to '0x084' to set the address to the target 24-bit flash address.
4. Register write to '0x090' to set the data length to 5 bytes and to set the dummy length to 4 bytes.
5. Register write to '0x094' to set the direction control to read and enable both the address and dummy bytes.
6. Register write to '0x0a0' to start the user-initiated command.
7. Register read from '0x1c0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation.
8. Register read from '0x098' for the lower 4 bytes of the read data {byte3, byte2, byte1, byte0}.
9. Register read from '0x09c' for the upper byte of the read data byte4.

These procedures result to SPI transaction shown in Figure 3.14. Below waveform has clocking mode 0.

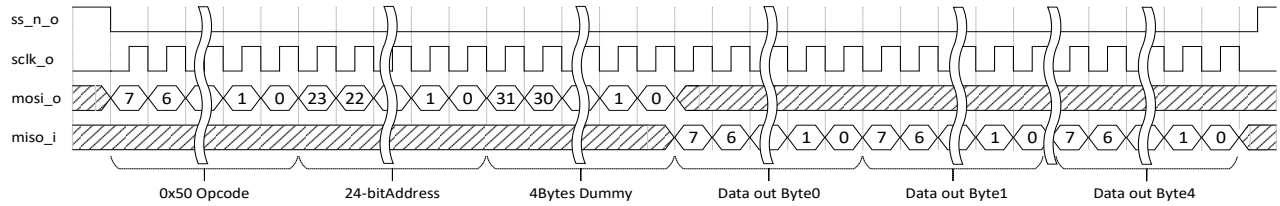

Figure 3.14. SPI Read Transaction using User-Defined Command

4. Signal Description

Table 4.1 lists the input and output signals of the IP.

Table 4.1. SPI Flash Memory Controller IP Core Signal Description

Port Name	I/O	Size	Description
Clock and Reset			
ahb_hclk_i	I	1	AHB-Lite input clock
apb_pclk_i	I	1	APB input clock
rst_i	I	1	Asynchronous active high reset When asserted, output ports and registers are forced to their reset values.
SPI Interface			
mosi_o	O	1	Serial data from SPI flash controller (FPGA) to SPI flash
miso_i	I	1	Serial data to SPI flash controller (FPGA) from SPI flash
ss_n_o	O	1	SPI flash chip select
sclk_o	O	1	Serial clock to SPI flash
wpj_o	O	1	SPI flash write protect
AHB-Lite Interface			
ahbl_haddr_i	I	*Data Port AHB Address Width*	AHB-Lite address signal
ahbl_hwdata_i	I	*Data Port AHB Data Width*	AHB-Lite write data signal
ahbl_hsize_i	I	3	AHB-Lite size signal Indicates the size of the transfer that is typically byte, halfword, or word.
ahbl_hwrite_i	I	1	AHB-Lite transfer direction signal 1 – Write transfer 0 – Read transfer
ahbl_hsel_i	I	1	AHB-Lite select signal Indicates that the current transfer is intended for the selected subordinate.
Ahbl_hburst_i	I	3	AHB-Lite burst type signal The burst type indicates if the transfer is a single transfer or forms part of a burst.
ahbl_htrans_i	I	2	AHB-Lite transfer type signal Indicates the transfer type of the current transfer. This can be: • IDLE • BUSY • NONSEQUENTIAL • SEQUENTIAL
ahbl_hlock_i	I	1	AHB-Lite lock signal
ahbl_hready_i	I	1	AHB-Lite ready input signal
ahbl_hrdata_o	O	*Data Port AHB Data Width*	AHB-Lite read data signal
ahbl_hresp_o	O	1	AHB-Lite transfer response signal When low, the signal indicates that the transfer status is okay. When high, the signal indicates that the transfer status is error.
ahbl_hready_o	O	1	AHB-Lite ready output signal Indicates that a transfer has finish on the bus. The subordinate uses this signal to extend an AHB-Lite transfer.

Port Name	I/O	Size	Description
APB Interface			
apb_psel_i	I	1	APB select signal Indicates that the completer device is selected, and that a data transfer is required.
apb_paddr_i	I	11	APB address signal
apb_pwdata_i	I	*Control port APB Data Width*	APB write data signal
apb_pwrite_i	I	1	APB direction signal 1 – APB write access 0 – APB read access
apb_penable_i	I	1	APB enable signal Indicates the second and subsequent cycles of an APB transfer.
apb_pready_o	O	1	APB ready signal The completer uses this signal to extend an APB transfer.
apb_pslverr_o	O	1	APB error signal Indicates a transfer failure. This signal is tied to 1'b0.
apb_prdata_o	O	*Control port APB Data Width*	APB read data signal

5. Register Description

5.1. Overview

The host can control the IP Core by writing to and reading from the available registers. These registers can be configured at the run-time.

Table 5.1 lists the address map and specifies the registers available.

Table 5.1. Summary of SPI Flash Memory Controller IP Core Registers

APB Offset	Register Name	Access Type	Description
Registers to Issue SPI Flash Commands			
0x000	Page Program	Write-only	This register is used to initiate the page program command.
0x004	Page Read	Write-only	This register is used to initiate the page read command.
0x008	Block Erase Type 1	Write-only	This register is used to initiate the block erase type 1 command.
0x00c	Block Erase Type 2	Write-only	This register is used to initiate the block erase type 2 command.
0x010	Block Erase Type 3	Write-only	This register is used to initiate the block erase type 3 command.
0x014	Chip Erase	Write-only	This register is used to initiate the chip erase command.
0x018	Write Enable	Write-only	This register is used to initiate the write enable command.
0x01c	Write Disable	Write-only	This register is used to initiate the write disable command.
0x020	Status Read	Read-only	This register is used to initiate the read status register command.
0x024	Status Write	Write-only	This register is used to initiate the write status register command.
0x028	Power Down	Write-only	This register is used to initiate the power down command.
0x02c	Power Up	Write-only	This register is used to initiate the power up command.
0x030	Manufacturer ID	Read-only	This register is used to initiate the read manufacturer ID command.
Registers to set SPI Flash Address			
0x040	SPI Flash Address	Read/Write	This register is used to set the SPI flash address.
0x044	SPI Flash Addressing Mode	Read/Write	This register is used to set the SPI flash addressing mode.
Registers to Issue a User-Defined Command to SPI Flash			
0x080	USER CMD Opcode	Read/Write	This register is used to set the flash command opcode for user-initiated command.
0x084	USER CMD Address	Read/Write	This register is used to set the flash address for user-initiated command.
0x088	USER CMD Write Data 0	Read/Write	This register is used to set the flash data bytes, Byte3 – Byte0, for user-initiated command.
0x08c	USER CMD Write Data 1	Read/Write	This register is used to set the flash data bytes, Byte7 – Byte4, for user-initiated command.
0x090	USER CMD Data and Dummy Length	Read/Write	This register is used to set the flash data byte length for both write and read, and dummy byte length for read for user-initiated command.
0x094	USER CMD Configurations	Read/Write	This register is used to set the configurations for user-initiated command.
0x098	USER CMD Read Data 0	Read-only	This register is used to read the returned data, Byte3 – Byte0, for user-initiated command.
0x09c	USER CMD Read Data 1	Read-only	This register is used to read the returned data, Byte7 – Byte4, for user-initiated command.
0x0a0	USER CMD Start	Write-only	This register is used to initiate user-defined command.

APB Offset	Register Name	Access Type	Description
Registers to Configure SPI Flash Controller Internals			
0x100	Read Speed	Read/Write	This register is used to program read commands as fast read or slow read commands.
0x104	Page Program Size	Read/Write	This register is used to program page program size for page program commands.
0x108	Page Read Size	Read/Write	This register is used to program page read size for read commands such as fast read or slow read commands.
0x10c	Write Protect Register	Read/Write	This register is used to enable/disable write protect signal assertion.
0x110	SCLK Rate	Read/Write	This register is used to set the factor for deriving sclk_o from input apb_pclk_i.
0x114	SPI First Transmitted Bit	Read/Write	Specifies the shifting order of SPI bits when transmitting or receiving data.
0x118	AHB-L Page Buffer Enable	Read/Write	This register is used to enable/disable page buffer write or read in AHB-L interface.
0x11c	SCLK Polarity	Read/Write	This register is used to set the SCLK polarity during idle transaction.
0x120	SCLK Phase	Read/Write	This register is used to set the SCLK phase during active transaction. This register indicates the edge clock for sampling and shifting out of data.
Registers to Configure SPI Flash Controller for an SPI Flash Instruction Set			
0x180	Slow Read CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's slow read command.
0x184	Fast Read CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's fast read command.
0x18c	Page Program CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's page program command.
0x190	Block Erase Type 1 CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's block erase type 1 command.
0x194	Block Erase Type 2 CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's block erase type 2 command.
0x198	Block Erase Type 3 CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's block erase type 3 command.
0x19c	Chip Erase CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's chip erase command.
0x1a0	Write Enable CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's write enable command.
0x1a4	Write Disable CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's write disable command.
0x1a8	Read Status CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's read status register command.
0x1ac	Write Status CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's write status register command.
0x1b0	Deep Power Down CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's deep power down command.
0x1b4	Resume from Power Down CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's resume from power down command.
0x1b8	Read Manufacturer ID CFG	Read/Write	This register is used to configure SPI flash controller for an SPI flash's read manufacturer ID command.

APB Offset	Register Name	Access Type	Description
Status Registers			
0x1c0	Transaction Status	Read-only	This register indicates the transaction status of the controller.
0x1c4	Page Read Buffer Data Count Status	RW1C	This register tracks the number of data written on the buffer and available to be read during page read.
0x1c8	Page Read Buffer Address Status	RW1C	This register tracks the last address read on the page read buffer.
Page Program and Read Buffer			
0x200-0x3FF	Page Program Buffer	Write-only	This register is used to program data into the page program buffer for page program command.
0x400-0x5FF	Page Read Buffer	Read-only	This register is used to read data from the page read buffer for page read commands such as fast read or slow read.

The behavior of registers to write and read access is defined by its access type, which is defined in Table 5.2.

Table 5.2. Access Type Definition

Access Type	Behavior on Read Access	Behavior on Write Access
Read-only	Returns register value	Ignores write access
Write-only	Returns 0	Updates register value
Read/Write	Returns register value	Updates register value
RW1C	Returns register value	Writing 1'b1 on register bit clears the bit to 1'b0. Writing 1'b0 on register bit is ignored.

5.2. Page Program Register 0x000

Table 5.3 shows the Page Program register.

Table 5.3. Page Program Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Writing a 1 to this bit starts a page program operation.

5.3. Page Read Register 0x004

Table 5.4 shows the Page Read register.

Table 5.4. Page Read Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Writing a 1 to this bit starts a page read operation.

5.4. Block Erase Type 1 Register 0x008

Table 5.5 shows the Block Erase Type 1 register.

Table 5.5. Block Erase Type 1 Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Writing a 1 to this bit starts the block erase operation on specific flash memory size using the block erase type 1 command.

5.5. Block Erase Type 2 Register 0x00c

Table 5.6 shows the Block Erase Type 2 register.

Table 5.6. Block Erase Type 2 Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Writing a 1 to this bit starts the block erase operation on specific flash memory size using the block erase type 2 command.

5.6. Block Erase Type 3 Register 0x010

Table 5.7 shows the Block Erase Type 3 register.

Table 5.7. Block Erase Type 3 Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Writing a 1 to this bit starts the block erase operation on specific flash memory size using the block erase type 3 command.

5.7. Chip Erase Register 0x014

Table 5.8 shows the Chip Erase register.

Table 5.8. Chip Erase Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Write a 1 to this bit to initiate an SPI flash chip erase operation.

5.8. Write Enable Register 0x018

Table 5.9 shows the Write Enable register.

Table 5.9. Write Enable Register

Name	Bits	Access Mode	Description
Enable	0	Write-only	Write a 1 to this bit to initiate an SPI flash “Write Enable” command.

5.9. Write Disable Register 0x01c

Table 5.10 shows the Write Disable register.

Table 5.10. Write Disable Register

Name	Bits	Access Mode	Description
Disable	0	Write-only	Write a 1 to this bit to initiate an SPI flash “Write Disable” command.

5.10. Status Read Register 0x020

Table 5.11 shows the Status Read register.

Table 5.11. Status Read Register

Name	Bits	Access Mode	Description
Value	7:0	Read-only	8-bit return data after reading this register, indicates the status of SPI flash.

5.11. Status Write Register 0x024

Table 5.12 shows the Status Write register.

Table 5.12. Status Write Register

Name	Bits	Access Mode	Description
Value	7:0	Write-only	8-bit write data to be written to the status register of SPI flash.

5.12. Power Down Register 0x028

Table 5.13 shows the Power Down register.

Table 5.13. Power Down Register

Name	Bits	Access Mode	Description
Start	0	Write-only	Initiate a “Deep Power Down” command in SPI flash by writing a 1.

5.13. Power Up Register 0x02c

Table 5.14 shows the Power Up register.

Table 5.14. Power Up Register

Name	Bits	Access Mode	Description
Value	0	Write-only	Initiate a “Resume from Power Down” command in SPI flash by writing a 1.

5.14. Manufacturer ID Register 0x030

Table 5.15 shows the Manufacturer ID register.

Table 5.15. Manufacturer ID Register

Name	Bits	Access Mode	Description
Value	7:0	Read-only	8-bit manufacturer ID of the SPI flash after reading this register.

5.15. SPI Flash Address Register 0x040

Table 5.16 shows the SPI Flash Address register.

Table 5.16. SPI Flash Address Register

Name	Bits	Access Mode	Description
Address	31:0	Read/Write	24-bit or 32-bit flash address. For AHB-Lite interface, if *AHB Address Width* is set to 24, only 24-bit flash addressing is available. For 24-bit addressing mode, both APB and AHB-Lite interface should set the flash address on the lower 24-bit of this register.

5.16. SPI Flash Addressing Mode Register 0x044

Table 5.17 shows the SPI Flash Addressing Mode register.

Table 5.17. SPI Flash Addressing Mode Register

Name	Bits	Access Mode	Description
Addressing Mode	0	Read/Write	Indicates the flash addressing mode. 0 : 24-bit 1 : 32-bit If *AHB Address Width* is set to 24, only 24-bit flash addressing is available.

5.17. User CMD Opcode Register 0x080

Table 5.18 shows the User CMD Opcode register.

Table 5.18. User CMD Opcode Register

Name	Bits	Access Mode	Description
User Opcode	7:0	Read/Write	The 8-bit command opcode of a user-defined command that is issued on the SPI flash.

5.18. User CMD Address Register 0x084

Table 5.19 shows the User CMD Address register.

Table 5.19. User CMD Address Register

Name	Bits	Access Mode	Description
User Address	31:0	Read/Write	The 24-bit or 32-bit address of a user-defined command that is issued to the SPI flash. For 24-bit address, the lower 24-bit of this register should be set.

5.19. User CMD Write Data 0 Register 0x088

Table 5.20 shows the User CMD Write Data 0 register.

Table 5.20. User CMD Write Data 0 Register

Name	Bits	Access Mode	Description
User Write Data 0	31:0	Read/Write	SPI flash write data for the user command. Write Data 0: byte3, byte2, byte1, byte0.

5.20. User CMD Write Data 1 Register 0x08c

shows the User CMD Write Data 1 register.

Table 5.21. User CMD Write Data 1 Register

Name	Bits	Access Mode	Description
User Write Data 1	31:0	Read/Write	SPI flash write data for the user command. Write Data 1: byte7, byte6, byte5, byte4.

5.21. User CMD Data and Dummy Length Register 0x090

Table 5.22 shows the User CMD Data and Dummy Length register.

Table 5.22. User CMD Data and Dummy Length Register

Name	Bits	Access Mode	Description
User Dummy Length	7:4	Read/Write	SPI dummy length for flash read. Actual length = User Dummy Length + 1.
User Data Length	3:0	Read/Write	SPI data length for both flash write and read. Valid values are 0-8 only.

5.22. User CMD Configurations Register 0x094

Table 5.23 shows the User CMD Configurations register.

Table 5.23. User CMD Configurations Register

Name	Bits	Access Mode	Description
User Dummy Enable	2	Read/Write	Set to 1 to include the dummy bytes in the user command.
User Address Enable	1	Read/Write	Set to 1 to include the address in the user command.
User Direction	0	Read/Write	Set to 0 to perform flash read. Otherwise, set to 1 for flash write.

5.23. User CMD Read Data 0 Register 0x098

Table 5.24 shows the User CMD Read Data 0 register.

Table 5.24. User CMD Read Data 0 Register

Name	Bits	Access Mode	Description
User Read Data 0	31:0	Read	SPI flash read data for the user command. Read Data 0: byte3, byte2, byte1 and byte0.

5.24. User CMD Read Data 1 Register 0x09c

Table 5.25 shows the User CMD Read Data 1 register.

Table 5.25. User CMD Read Data 1 Register

Name	Bits	Access Mode	Description
User Read Data 1	31:0	Read	SPI flash read data for the user command. Read Data 1: byte7, byte6, byte5 and byte4.

5.25. User CMD Start Register 0x0a0

Table 5.26 shows the User CMD Start register.

Table 5.26. User CMD Start Register

Name	Bits	Access Mode	Description
Command Start	0	Write-only	Writing a 1 to this bit starts a user-defined command to SPI flash.

5.26. Read Speed Register 0x100

Table 5.27 shows the Read Speed register.

Table 5.27. Read Speed Register

Name	Bits	Access Mode	Description
SPI Dummy	7:4	Read/Write	SPI dummy length for flash fast read. Actual length = SPI Dummy + 1
Enable Fast Read	0	Read/Write	Set to 1 to enabled flash fast read.

5.27. Page Program Size Register 0x104

Table 5.28 shows the Page Program Size register.

Table 5.28. Page Program Size Register

Name	Bits	Access Mode	Description
Size	9:0	Read/Write	The number of bytes to be written to the SPI flash on a Page Program. The minimum value is 1 and maximum value is PAGE_SIZE. The SPI flash controller writes the number of bytes specified in this field from the Page Program Buffer to the SPI flash.

5.28. Page Read Size Register 0x108

Table 5.29 shows the Page Read Size register.

Table 5.29. Page Read Size Register

Name	Bits	Access Mode	Description
Size	9:0	Read/Write	The number of bytes to be read from SPI flash on a Page Read. The minimum value is 1 and maximum value is PAGE_SIZE. The SPI flash controller reads the number of bytes specified in this field from SPI flash then store to Page Read Buffer.

5.29. Write Protect Register 0x10c

Table 5.30 shows the Write Protect register.

Table 5.30. Write Protect Register

Name	Bits	Access Mode	Description
Write Protect Enable/Disable	0	Read/Write	Set to 1 to enable write protect signal assertion on SPI transaction.

5.30. SCLK Rate Register 0x110

Table 5.31 shows the SCLK Rate register.

Table 5.31. SCLK Rate Register

Name	Bits	Access Mode	Description
SCLK Rate	3:0	Read/Write	Set the factor for deriving sclk_o from input apb_clk_i. This register should not be set dynamically during SPI transfer. 0 : sclk_o has the same rate as apb_clk_i 1 – 15 : sclk_o = apb_clk_i/(2*SCLK Rate value)

5.31. SPI First Transmitted Bit Register 0x114

Table 5.32 shows the SPI First Transmitted Bit register.

Table 5.32. SPI First Transmitted Bit Register

Name	Bits	Access Mode	Description
First Transmitted Bit	0	Read/Write	Specifies the shifting order of the SPI bits when transmitting/receiving data. 0: MSB 1: LSB

5.32. AHB-L Page Buffer Enable Register 0x118

Table 5.33 shows the AHB-L Page Buffer Enable register.

Table 5.33. AHB-L Page Buffer Enable Register

Name	Bits	Access Mode	Description
Page Buffer Enable	0	Read/Write	This register is used to enable/disable page buffer write and read in AHB-L interface. 0: Disable 1: Enable

5.33. SCLK Polarity Register 0x11c

Table 5.34 shows the SCLK polarity register.

Table 5.34. SCLK Polarity Register

Name	Bits	Access Mode	Description
SCLK Polarity	0	Read/Write	This register is used to set the logic level of sclk_o during idle. When the value is 0, sclk_o is low during idle. When the value is 1, sclk_o is high during idle.

5.34. SCLK Phase Register 0x120

Table 5.35 shows the SCLK phase register.

Table 5.35. SCLK Phase Register

Name	Bits	Access Mode	Description
SCLK Phase	0	Read/Write	This register is used to set the sclk_o phase during SPI active transaction. This specifies the sclk_o edge for shifting out and sampling data.

5.35. Slow Read CFG Register 0x180

Table 5.36 shows the Slow Read CFG register.

Table 5.36. Slow Read CFG Register

Name	Bits	Access Mode	Description
Slow Read Flash Command Opcode	7:0	Read/Write	Slow read flash command opcode of the off-chip SPI flash memory device.

5.36. Fast Read CFG Register 0x184

Table 5.37 shows the Fast Read CFG register.

Table 5.37. Fast Read CFG Register

Name	Bits	Access Mode	Description
Fast Read Flash Command Opcode	7:0	Read/Write	Fast read flash command opcode of the off-chip SPI flash memory device.

5.37. Page Program CFG Register 0x18c

Table 5.38 shows the Page Program CFG register.

Table 5.38. Page Program CFG Register

Name	Bits	Access Mode	Description
Page Program Flash Command Opcode	7:0	Read/Write	Page Program flash command opcode of the off-chip SPI flash memory device.

5.38. Block Erase Type 1 CFG Register 0x190

Table 5.39 shows the Block Erase Type 1 CFG register.

Table 5.39. Block Erase Type 1 CFG Register

Name	Bits	Access Mode	Description
Block Erase Type 1 Flash Command Opcode	7:0	Read/Write	Block Erase Type 1 flash command opcode of the off-chip SPI flash memory device.

5.39. Block Erase Type 2 CFG Register 0x194

Table 5.40 shows the Block Erase Type 2 CFG register.

Table 5.40. Block Erase Type 2 CFG Register

Name	Bits	Access Mode	Description
Block Erase Type 2 Flash Command Opcode	7:0	Read/Write	Block Erase Type 2 flash command opcode of the off-chip SPI flash memory device.

5.40. Block Erase Type 3 CFG Register 0x198

Table 5.41 shows the Block Erase Type 3 CFG register.

Table 5.41. Block Erase Type 3 CFG Register

Name	Bits	Access Mode	Description
Block Erase Type 3 Flash Command Opcode	7:0	Read/Write	Block Erase Type 3 flash command opcode of the off-chip SPI flash memory device.

5.41. Chip Erase CFG Register 0x19c

Table 5.42 shows the Chip Erase CFG register.

Table 5.42. Chip Erase CFG Register

Name	Bits	Access Mode	Description
Chip Erase Flash Command Opcode	7:0	Read/Write	Chip Erase flash command opcode of the off-chip SPI flash memory device.

5.42. Write Enable CFG Register 0x1a0

Table 5.43 shows the Write Enable CFG register.

Table 5.43. Write Enable CFG Register

Name	Bits	Access Mode	Description
Write Enable Flash Command Opcode	7:0	Read/Write	Write Enable flash command opcode of the off-chip SPI flash memory device.

5.43. Write Disable CFG Register 0x1a4

Table 5.44 shows the Write Disable CFG register.

Table 5.44. Write Disable CFG Register

Name	Bits	Access Mode	Description
Write Disable Flash Command Opcode	7:0	Read/Write	Write Disable flash command opcode of the off-chip SPI flash memory device.

5.44. Read Status CFG Register 0x1a8

Table 5.45 shows the Read Status CFG register.

Table 5.45. Read Status CFG Register

Name	Bits	Access Mode	Description
Read Status Flash Command Opcode	7:0	Read/Write	Read Status flash command opcode of the off-chip SPI flash memory device.

5.45. Write Status CFG Register 0x1ac

Table 5.46 shows the Write Status CFG register.

Table 5.46. Write Status CFG Register

Name	Bits	Access Mode	Description
Write Status Flash Command Opcode	7:0	Read/Write	Write Status flash command opcode of the off-chip SPI flash memory device.

5.46. Deep Power Down CFG Register 0x1b0

Table 5.47 shows the Deep Power Down CFG register.

Table 5.47. Deep Power Down CFG Register

Name	Bits	Access Mode	Description
Deep Power Down Flash Command Opcode	7:0	Read/Write	Deep power down flash command opcode of the off-chip SPI flash memory device.

5.47. Resume from Power Down CFG Register 0x1b4

Table 5.48 shows the Resume from Power Down CFG register.

Table 5.48. Resume from Power Down CFG Register

Name	Bits	Access Mode	Description
Resume from Power Down Flash Command Opcode	7:0	Read/Write	Resume from power down flash command opcode of the off-chip SPI flash memory device.

5.48. Read Manufacturer ID CFG Register 0x1b8

Table 5.49 shows the Read Manufacturer ID CFG register.

Table 5.49. Read Manufacturer ID CFG Register

Name	Bits	Access Mode	Description
Read Manufacturer ID Flash Command Opcode	7:0	Read/Write	Read manufacturer ID flash command opcode of the off-chip SPI flash memory device.

5.49. Transaction Status Register 0x1c0

Table 5.50 shows the Transaction Status Register.

Table 5.50. Transaction Status Register

Name	Bit	Access Mode	Description
IP Transaction Status	0	Read only	When read as 1, the controller is currently executing SPI transaction to the flash. When read as 0, the controller's SPI transaction to flash is already done and it is ready to receive new transaction request.

5.50. Page Read Buffer Data Count Status Register 0x1c4

Table 5.51 shows the Page Read Buffer Data Count Status register. This register can be accessed to check page read buffer status whenever data is available and for number of available buffer data. After reading all the buffer data, you need to write 1 on this register to clear/reset register value.

Table 5.51. Page Read Buffer Data Count Status Register

Name	Bits	Access Mode	Description
Reserved	[31:19]	Read-only	—
Buffer Content	[18:8]	RW1C	Indicates the number of 8-bit or 32-bit data available on the page read buffer.
Reserved	[7:1]	Read-only	—
Buffer Status	[0]	RW1C	Indicates whether the buffer is empty or not. 1'b1: buffer is not empty. 1'b0: buffer is empty.

5.51. Page Read Buffer Address Register 0x1c8

Table 5.52 shows the Page Read Buffer Address register. This register can be accessed to check the last buffer address where read is performed. After reading all the buffer data, you need to write 1 on this register to clear/reset register value.

Table 5.52. Page Read Buffer Address Register

Name	Bits	Access Mode	Description
Address of Last Buffer Read	[31:0]	RW1C	Indicates the last page read buffer address read through the interface. When the buffer interface is APB, values are in the range of 0x400-0x5FF. When the buffer interface is AHB-Lite, actual 24-bit or 32-bit flash address is the return value.

5.52. Page Program Buffer Register 0x200-0x3FF

Table 5.53 shows the Page Program Buffer register.

Table 5.53. Page Program Buffer Register

Name	Bits	Access Mode	Description
Page Program Buffer Register	[9:0]	Write-only	Page program buffer register. When apb_paddr_i[9:8] == 2'b10, maximum page program size is 256. When apb_paddr_i[9:8] == 2'b11, maximum page program size is 512.

5.53. Page Read Buffer Register 0x400-0x5FF

Table 5.54 shows the Page Read Buffer register.

Table 5.54. Page Read Buffer Register

Name	Bits	Access Mode	Description
Page Read Buffer Register	[10:0]	Read-only	Page read buffer register. When apb_paddr_i[10:8] == 3'b100, maximum page read size is 256. When apb_paddr_i[10:8] == 3'b101, maximum page read size is 512.

6. Example Design

A system design with the SPI Flash Memory Controller IP can be developed to test the IP with a RISC-V processor.

The SPI Flash Memory Controller example design allows you to compile, simulate, and test the SPI Flash Memory Controller IP on the following Lattice evaluation board:

- CrossLink-NX PCIe Bridge Board

6.1. Example Design Supported Configuration

The following IP configuration is used during IP generation in a Propel project but some configuration attributes equivalent to parameters such as SCLK Polarity, SCLK Phase, and SCLK Ratio are also set during the C code test sequence. Clock mode 0, clock mode 3, and SCLK Ratio of 0 and 1 are also used. The Page Buffer Interface is also set to APB and AHBL.

The system clock frequencies used are 30 MHz for the APB clock and 100 MHz for the AHBL clock.

Note: In the table below, ✓ refers to a checked option in the SPI Flash Memory Controller IP example design and ✕ refers to an unchecked option or a non-applicable option in the SPI Flash Memory Controller IP example design.

Table 6.1. SPI Flash Memory Controller IP Configuration Supported by the Example Design

SPI Flash Memory Controller IP GUI Parameter	SPI Flash Memory Controller IP configuration supported in example demo design
Control Port Settings	
Enable Page Program Buffer	✓
Enable Page Read Buffer	✓
Page Program Buffer Memory Type	Distributed Memory
Page Read Buffer Memory Type	Distributed Memory
Page Buffer Interface	APB/AHBL
Control Base Address	0x80000000
SPI Flash Settings	
Page Size	256
Flash Memory Map Size (KB)	4
SCLK Rate	0
SCLK Polarity	0
SCLK Phase	0
First Transmitted Bit	MSB
Data and Control Bus	
Data Port AHB Address Width	32
Data Port AHB Data Width	32
Data Port AHB Data Byte Endianness	Little-Endian
Control Port APB Address Width	11
Control Port APB Data Width	32
SPI Flash Command Opcodes	
Supported Flash Commands	Default values

6.2. Overview of the Example Design and Features

The example design discussed in this section is created using a RISC-V MC SoC Project template in the Lattice Propel Development Suite. The generated project includes the following components:

- Processor – RISC-V MC with Programmable Interrupt Controller (PIC)/Timer
- GPIO
- Asynchronous SRAM
- UART – Serial port
- Oscillator and PLL
- Glue Logic

Figure 6.1 shows how the SPI Flash Memory Controller IP is instantiated on the template. The AHB-L interface of the IP is connected to the AHB-L Interconnect system bus while the APB interface of the IP is connected to the APB Interconnect system bus. The SPI interface ports remain. Clock inputs connect from PLL output clocks. The reset input connects from the inverted system reset.

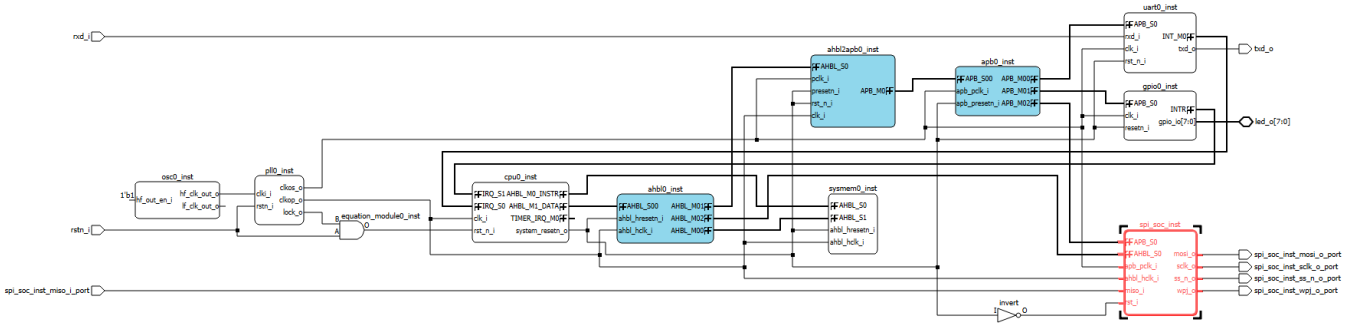

Figure 6.1. SPI Flash Memory Controller IP Example Design in SoC Project

6.3. Example Design Components

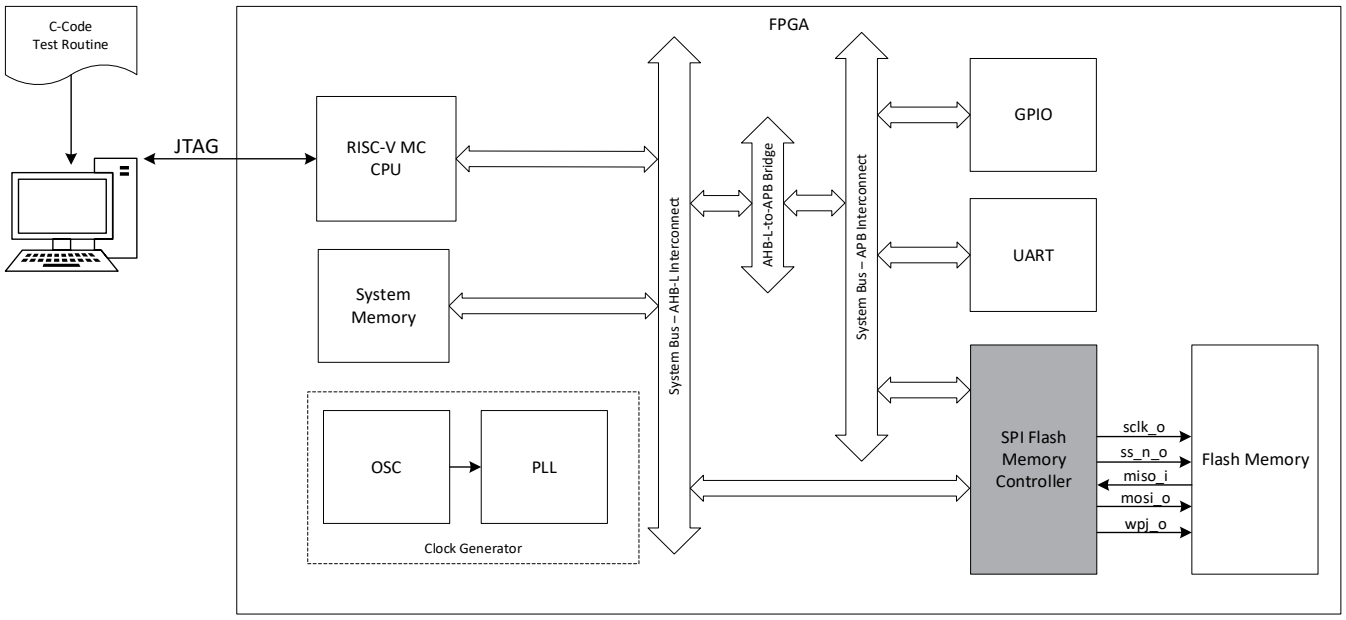

Figure 6.2. SPI Flash Memory Controller Example Design Block Diagram

The SPI Flash Memory Controller IP example design includes the following blocks:

- RISC-V MC CPU – processes data and instructions.
- System Memory – stores the commands to be done for testing.
- Oscillator and PLL – clock generator
- System bus – AHB-L and APB interconnects manage transfers from the processor and memory to the data port and control port of the IP, respectively.
- Interface bridge – translates AHB-L to APB transactions for GPIO, UART, and SPI Flash Memory Controller control port.
- GPIO and UART – serve as debug modules. UART can be used to display debug messages while GPIO output can be used to indicate successful load of bitstream on the FPGA board.
- SPI Flash Memory Controller IP – instance of the IP that will be connected to the flash device available on the FPGA board.
- Flash Memory – SPI target. This is not an instance of any submodule. The SPI interface ports of the IP will be connected to the on-board flash device through the physical design constraint (PDC) file.

6.4. Generating the Example Design

The Lattice Propel Builder and Propel SDK can be used to create an embedded system which consists of a System-on-Chip (SoC) design with an embedded processor and system software. The procedure for generating an example design for the SPI Flash Memory Controller is described below.

A workspace is generated on the Propel SDK. The SoC project is then created in Propel Builder. For more detailed instructions, refer to A Step-By-Step Approach to Lattice Propel (FPGA-AN-02052) or the Lattice Propel Builder User Guide and Lattice Propel SDK User Guide.

To create an SoC Project:

1. Launch Propel SDK. The **Lattice Propel Launcher** opens as shown in Figure 6.3. Browse to a directory for the **Workspace** field and click **Launch**.

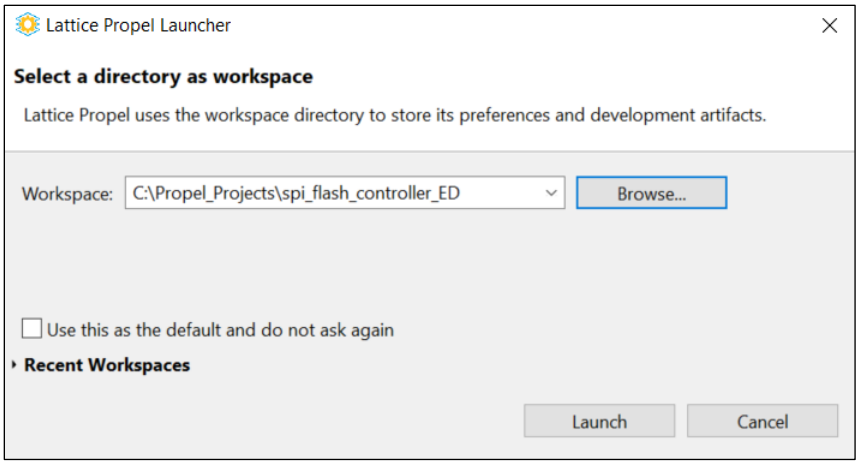

Figure 6.3. Lattice Propel Launcher

2. Create a new SoC design project by selecting **File > New > Lattice SoC Design Project**. The **Create SoC Project** window opens as shown in Figure 6.4.

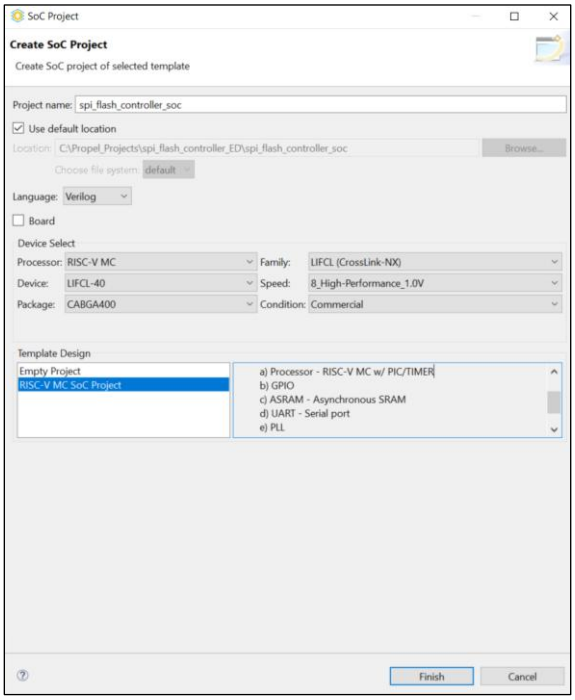

Figure 6.4. Create SoC Project

3. Enter a name for the SoC project in the **Project name** field. Customize the options under **Device Select** using the drop-down menus or configure them through the **Board** option. Select the template design from the **Template Design** list. For this example design, RISC-V MC processor, LIFCL-40-8CABGA400 board, and RISC-V MC SoC Project are used.

4. Click **Finish**.

5. Select the generated project, then run Propel Builder by clicking the icon or selecting **LatticeTools > Open Design in Propel Builder.** Propel Builder opens and loads the design template.

6. In the **IP Catalog** tab, instantiate the SPI Flash Memory Controller IP by generating the IP with the configuration as shown in Table 6.1.
7. After generating the IP, the **Define Instance** window opens as shown in Figure 6.5. Modify the instance name if needed, then click **OK**.

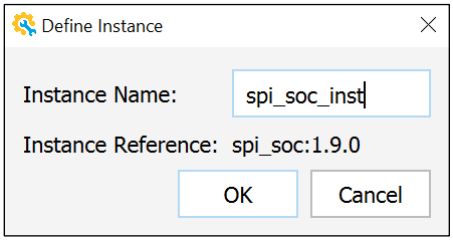

Figure 6.5. Define Instance

8. Connect the instantiated IP to the system (refer to Figure 6.1). From the default system components setting, modify the following:
   - Oscillator HCLK Frequency to 30 MHz.
   - PLL Input Clock, CLKOP, and CLKOS Frequency to 30 MHz, 60 MHz, and 30 MHz, respectively.
   - AHB-L Interconnect Total AHB-Lite Slaves to 3.
   - APB Interconnect Total APB Slaves to 3.
9. View the memory space of the project by selecting the **Address** tab as shown in Figure 6.6. Enable **Lock** for the address space settings so that the settings cannot be manually or automatically updated.

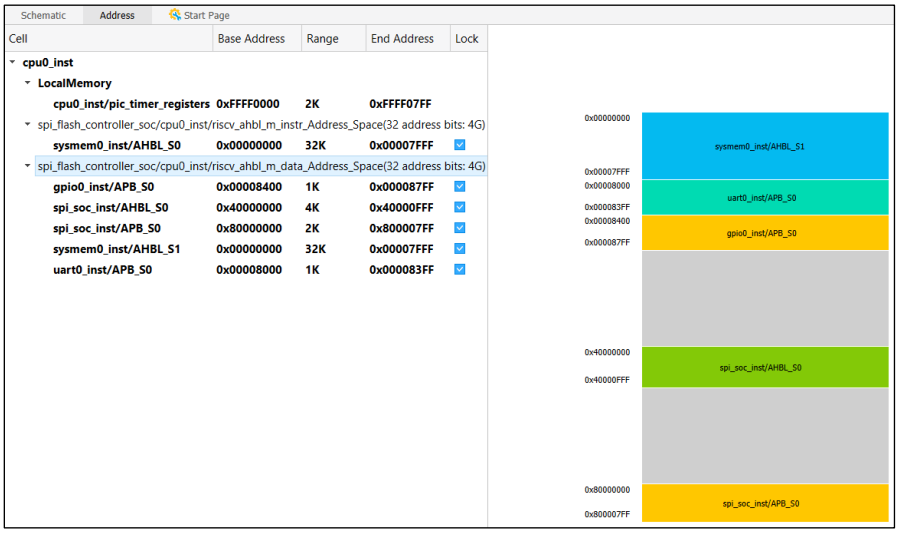

Figure 6.6. Address Tab

10. Click the icon to perform design rule checking (DRC).
11. Save the SoC project.

To create a Radiant Project:

1. Click the icon or select **Design > Run Radiant** to launch the Lattice Radiant Software.
2. Update the IP constraints file. On the PDC file, connect the SPI ports of the IP to the on-board flash pins. Set the sysCONFIG MASTER_SPI_PORT to DISABLE to use the on-board flash as target device.
3. Generate the programming file.

To create a C/C++ Project:

1. In the Lattice Propel software, build your SoC project to generate the system environment needed for the embedded C/C++ project by selecting the generated SoC project, then select **Project > Build Project**.
2. Check the build result from the **Console** view as shown in Figure 6.7.

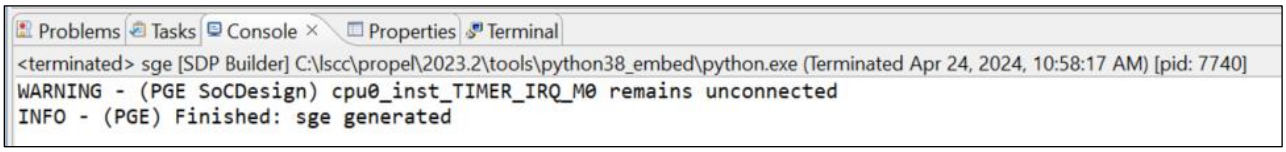

Figure 6.7. Build SoC Project Result

3. Generate a new Lattice C/C++ project by selecting **File > New > Lattice C/C++ Project**. The **C/C++ Project Load System and BSP** window opens as shown in Figure 6.8. Update the **Project Name** field, then click **Next** followed by **Finish**.

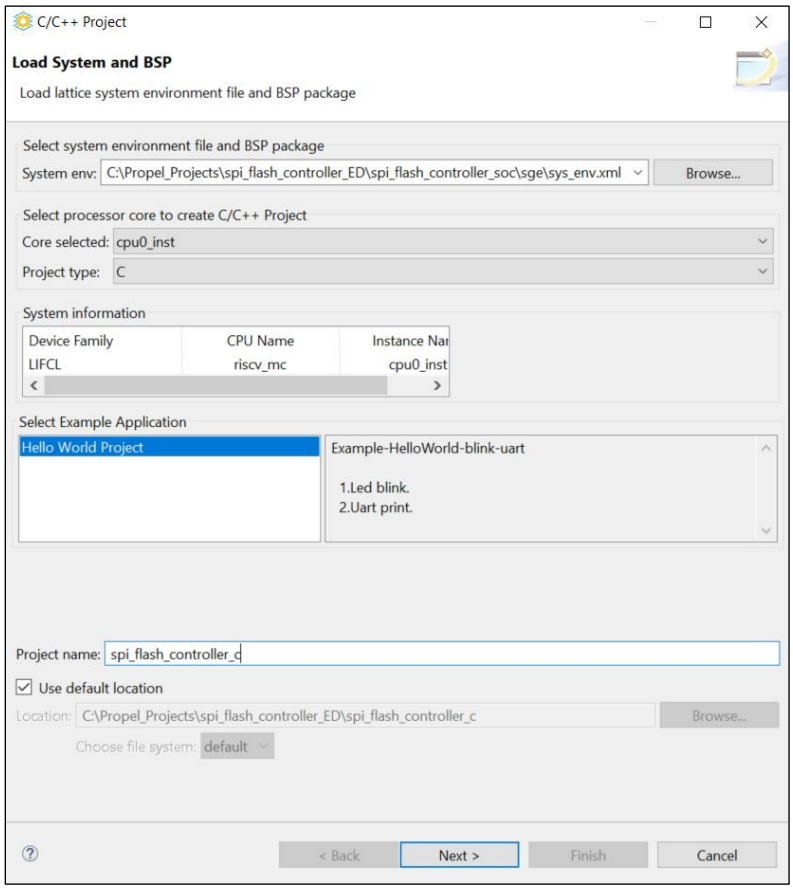

Figure 6.8. C/C++ Project Load System and BSP

4. Select the C/C++ project to build followed by **Project > Build Project**.
5. Check the build result from the **Console** tab as shown in Figure 6.9.

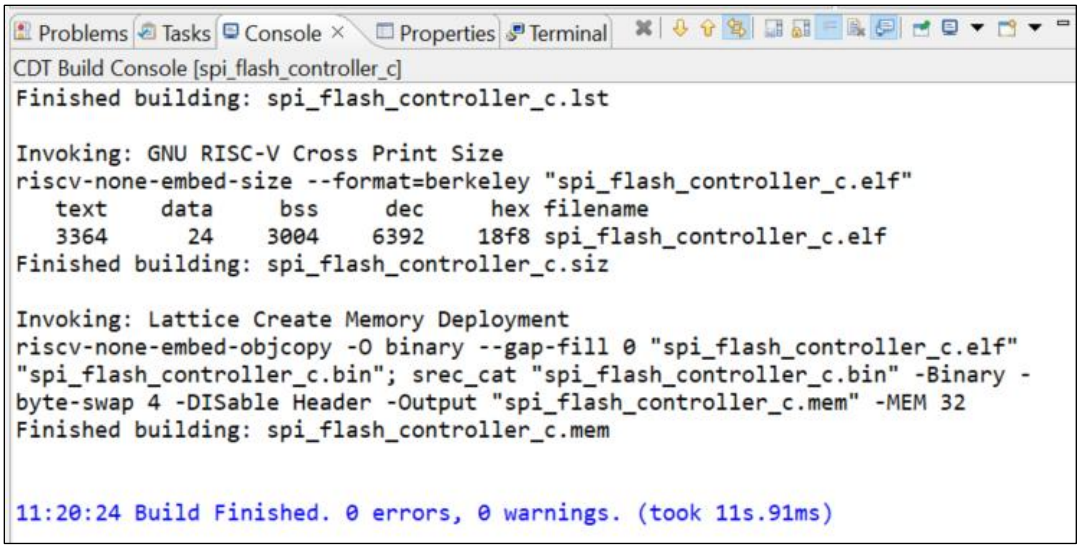

Figure 6.9. Build C/C++ Project Result

This environment is now ready for running on the FPGA board. Refer to the Propel SDK Flow On-Chip Debugging section of A Step-By-Step Approach to Lattice Propel (FPGA-AN-02052).

6.5. Simulating the Example Design

The Lattice Propel Builder software also has a verification project mode which can be used to generate a simulation environment. The procedure for generating a verification project for the SPI Flash Memory Controller is described below.

1. From the SoC Project, perform the pre-simulation requirements enumerated in the Verification Project Flow section of A Step-By-Step Approach to Lattice Propel (FPGA-AN-02052).
2. Click the icon to switch to verification project.
3. Reload dut_inst by double clicking on it. The **Reload sbx** window opens as shown in Figure 6.10. Click **Yes** to continue.

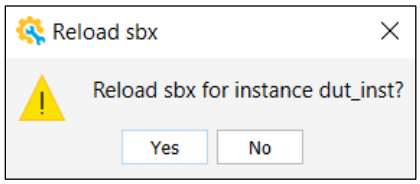

Figure 6.10. Reload sbx

The verification project schematic generated is shown in Figure 6.11.

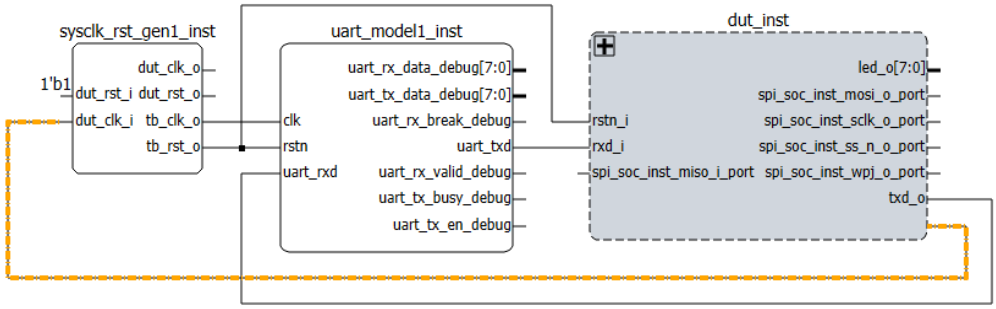

Figure 6.11. Verification Project Schematic

4. Click the **Generate** icon to generate the simulation environment. The testbench file including scripts for the chosen simulator, file lists, and other files are generated in the verification folder of the SoC project.
5. To see the full IP simulation behavior for this example design, connect a flash simulation model to the IP instance in the generated testbench file as shown in Figure 6.12 and add the flash simulation model in the generated file list.

spi_flash_controller_soc_v.sv

```
97   /*-----------------------Instantiation Blocks-----------------------*/
98   /* This section cover the instantiation of DUT, VIPs, Simulation  */
99   /* Models, and other IP/Components.                               */
100  /*----------------------------------------------------------------*/
101
102   flash_model flash_model_inst(
103        .SCLK  (spi_soc_inst_sclk_o_port),
104        .CS    (spi_soc_inst_ss_n_o_port),
105        .SI    (spi_soc_inst_mosi_o_port),
106        .SO    (spi_soc_inst_miso_i_port),
107        .WP    (spi_soc_inst_wpj_o_port),
108        .RESET (1'h0),
109        .SIO3  ());
110
111  spi_flash_controller_soc
112  dut_inst
113  (
114    .rstn_i(sysclk_rst_gen1_inst_tb_rst_o_net),
115    .rxd_i(uart_model1_inst_uart_txd_net),
116    .spi_soc_inst_miso_i_port(spi_soc_inst_miso_i_port),
117    .spi_soc_inst_mosi_o_port(spi_soc_inst_mosi_o_port),
118    .spi_soc_inst_sclk_o_port(spi_soc_inst_sclk_o_port),
119    .spi_soc_inst_ss_n_o_port(spi_soc_inst_ss_n_o_port),
120    .spi_soc_inst_wpj_o_port(spi_soc_inst_wpj_o_port),
121    .txd_o(uart_model1_inst_uart_rxd_net),
122    .led_o()
123  );
```

Figure 6.12. Edited Testbench File

6. Click the **Launch Simulation** icon to run the simulation.
   Note: The simulation waveform shown in Figure 6.13 results when the generated main.c file in the C/C++ project is updated to perform the operation of an IP.

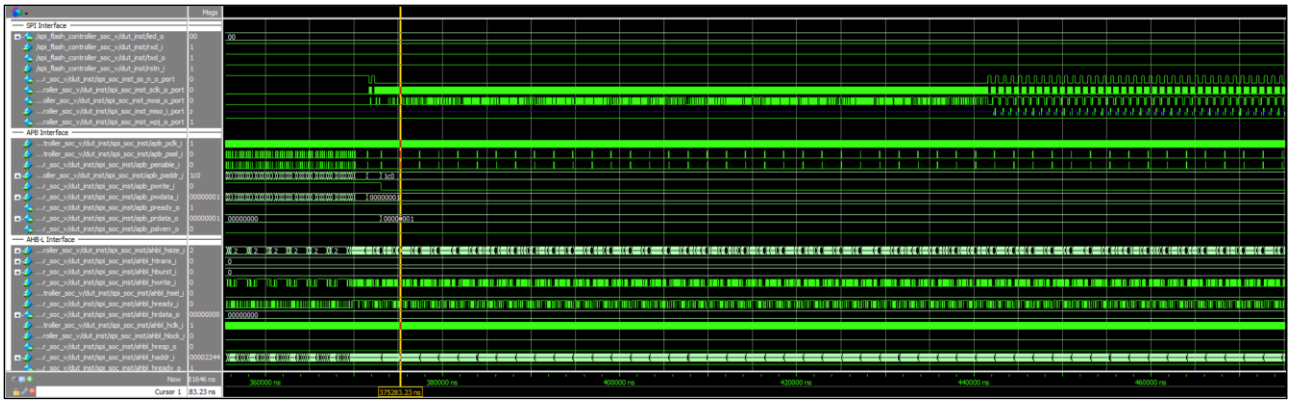

Figure 6.13. Simulation Waveform

6.6. Hardware Testing

The generated bitstream file from the procedure in the Generating the Example Design section is downloaded to the CrossLink-NX PCIe Bridge Board via the Radiant programmer. The Reveal analyzer is added to the Radiant project to check the output behavior of the IP.

7. Designing with the IP

This section provides information on how to generate the IP Core using the Lattice Radiant software and how to run simulation and synthesis. For more details on the Lattice Radiant software, refer to the Lattice Radiant Software User Guide.

7.1. Generating and Instantiating the IP

The Lattice Radiant software can be customized to generate modules and IPs and integrate them into the device's architecture. The procedure for generating the SPI Flash Memory Controller IP Core in Lattice Radiant software is described below.

To generate the SPI Flash Memory Controller IP:

1. Create a new Lattice Radiant software project or open an existing project.
2. In the **IP Catalog** tab, double-click **SPI Flash Controller** under **IP, Processors_Controllers_and_Peripherals** category. The **Module/IP Block Wizard** opens as shown in Figure 7.1. Enter values in the **Component name** and the **Create in** fields and click **Next**.

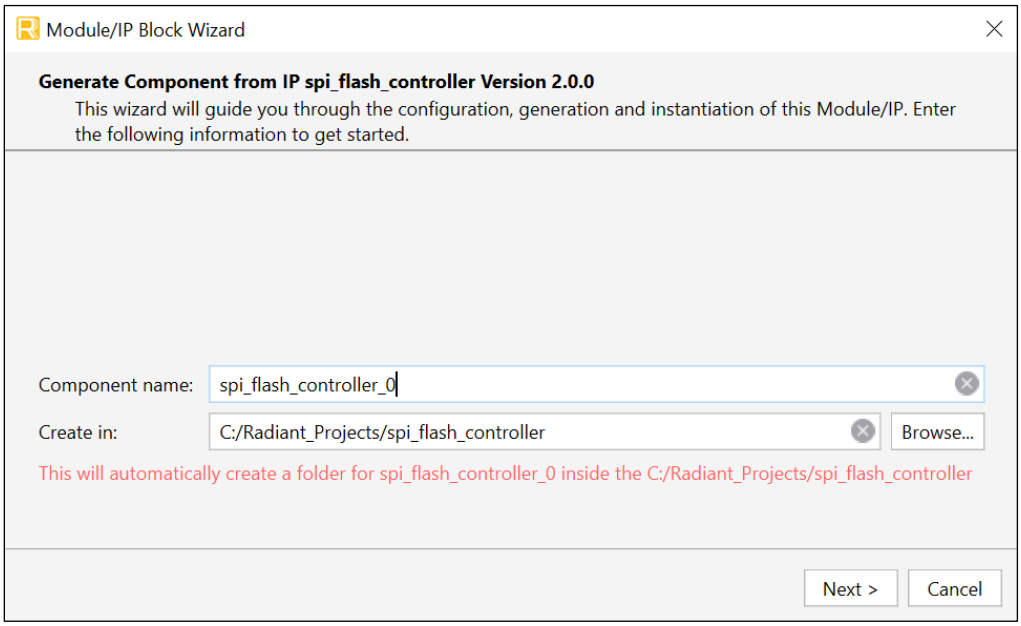

Figure 7.1. Module/IP Block Wizard

3. In the module's dialog box of the **Module/IP Block Wizard** window, customize the selected SPI Flash Memory Controller IP Core using drop-down menus and check boxes. For a sample configuration, see Figure 7.2. For configuration options, see the Attributes Summary section.

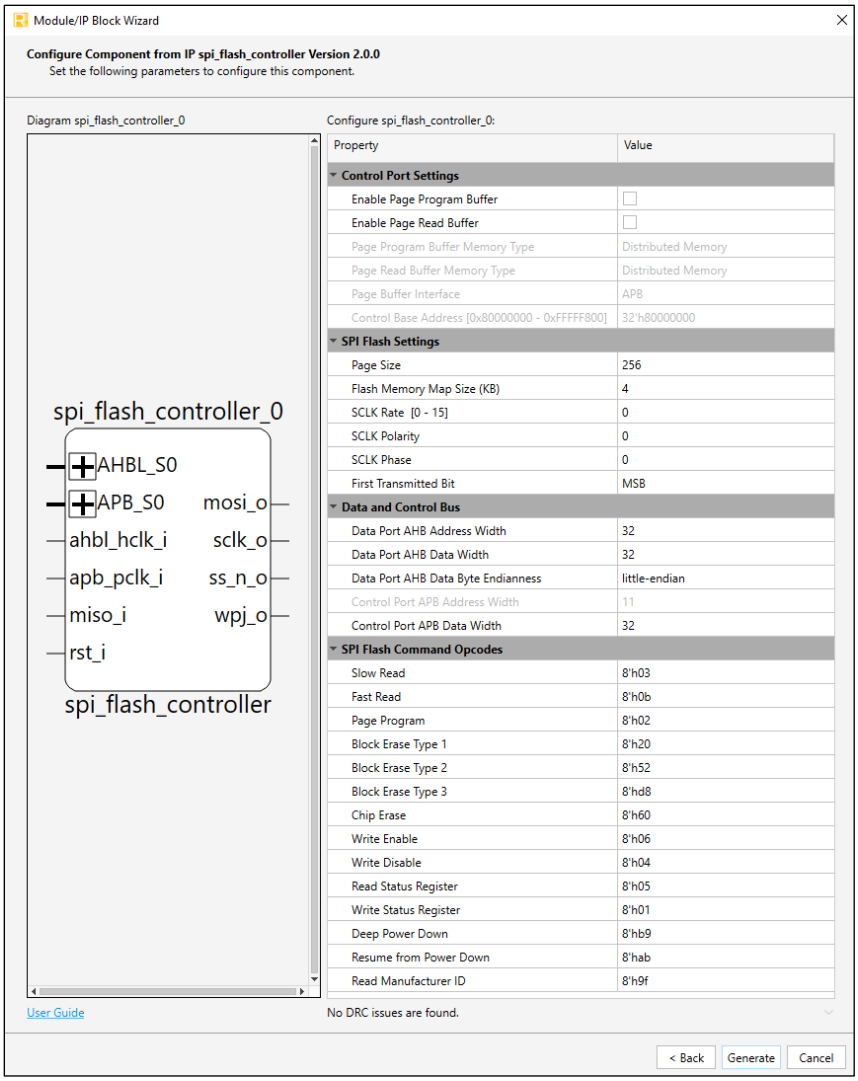

Figure 7.2. Configure User Interface of SPI Flash Memory Controller IP Core

4. Click **Generate**. The **Check Generated Result** dialog box opens, showing design block messages and results (Figure 7.3).

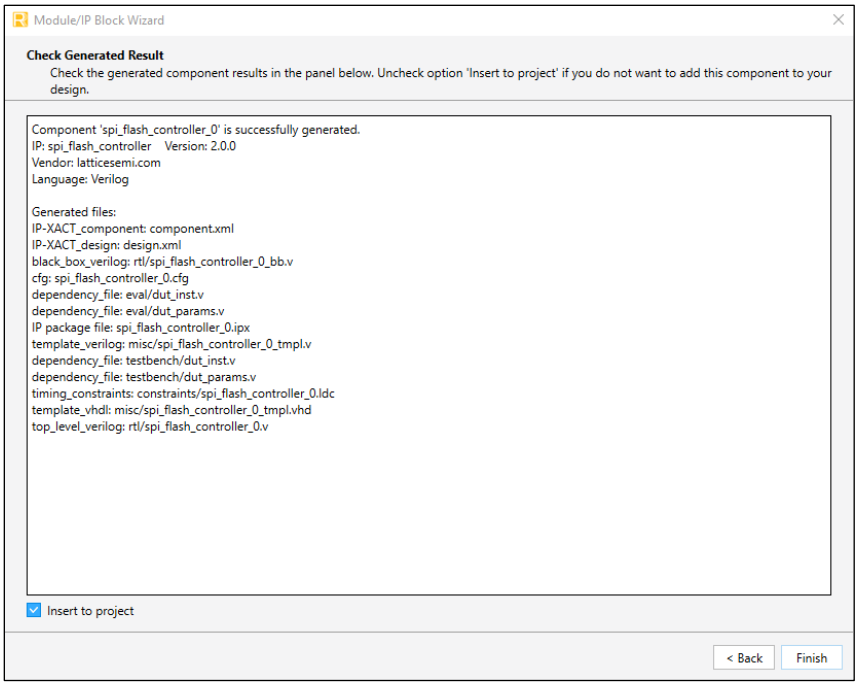

Figure 7.3. Check Generated Result

5. Click **Finish**. All the generated files are placed under the directory paths in the **Create in** and the **Component name** fields shown in Figure 7.1.

7.1.1. Generated Files and File Structure

The generated SPI Flash Memory Controller module package includes the black box (<Component name>_bb.v) and instance templates (<Component name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Component name>.v) that can be used as an instantiation template for the module is also provided. You may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in Table 7.1.

Table 7.1. Generated File List

Attribute	Description
<Component name>.ipx	This file contains the information on the files associated to the generated IP.
<Component name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/<Component name>.v	This file provides an example RTL top file that instantiates the module.
rtl/<Component name>_bb.v	This file provides the synthesis black box.
misc/<Component name>_tmpl.v misc /<Component name>_tmpl.vhd	These files provide instance templates for the module.
eval/constraint.pdc	IP constraint file.
testbench/dut_inst.v	This file contains the IP instance instantiation.
testbench/dut_params.v	This file contains the parameter values set during IP configuration which is used by the tb_top.v for simulation.
testbench/tb_top.v	Top-level testbench file of this IP.
testbench/test_stimuli.v	This file serves as the main testbench file which generates stimuli to the IP configuration and then checks and compares the IP output to the expected output generated within.

7.2. Running Functional Simulation

After the IP is generated, functional simulation can be performed using different available simulators. The default simulator already has pre-compiled libraries ready for simulation. Choosing a non-default simulator, however, may require additional steps.

To run functional simulation using the default simulator:

1. Click the  icon located on the **Toolbar** to initiate the **Simulation Wizard** shown in Figure 7.4.

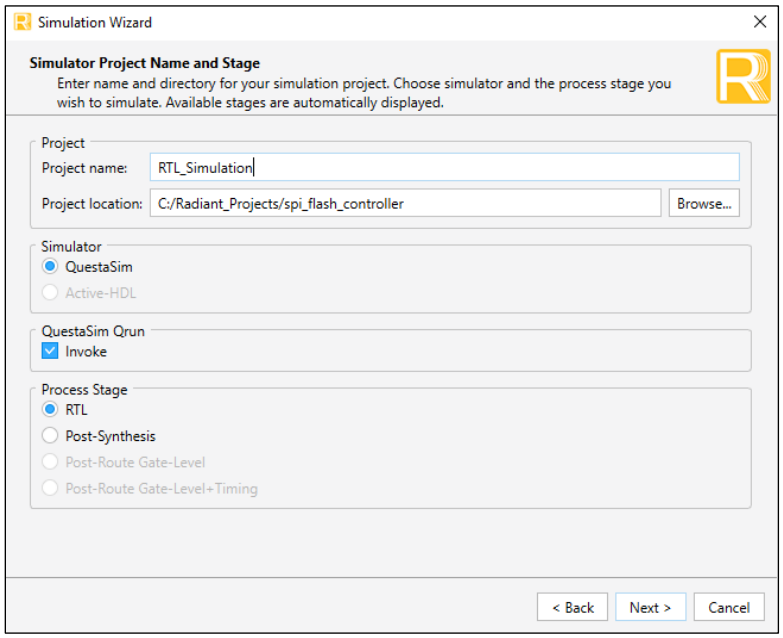

Figure 7.4. Simulation Wizard

2. Click **Next** to open the **Add and Reorder Source** window, as shown in Figure 7.5.

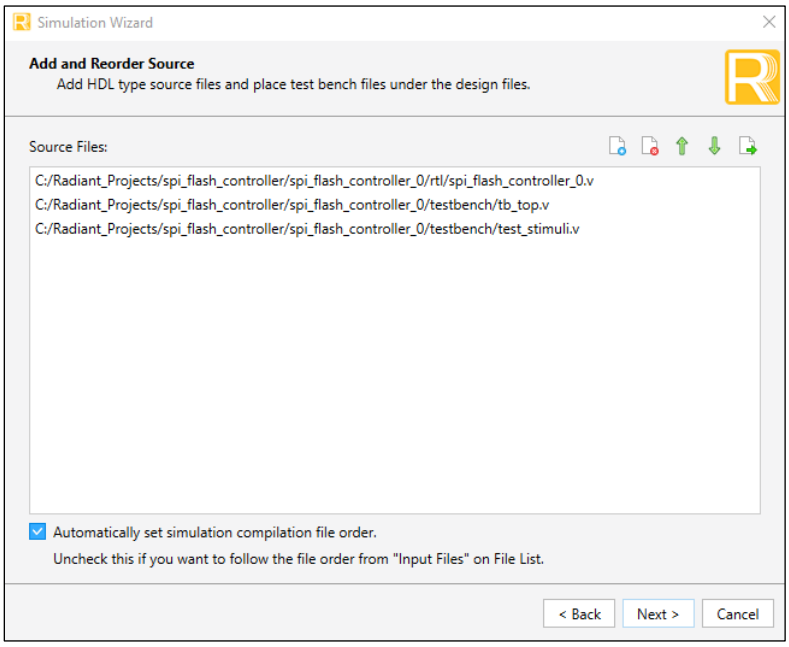

Figure 7.5. Add and Reorder Source

3. Click **Next**. The **Summary** window is shown.
4. Click **Finish** to run the simulation.

Note: It is necessary to follow the procedure above until it is fully automated in the Lattice Radiant software suite. The results of the simulation in our example are provided in Figure 7.6.

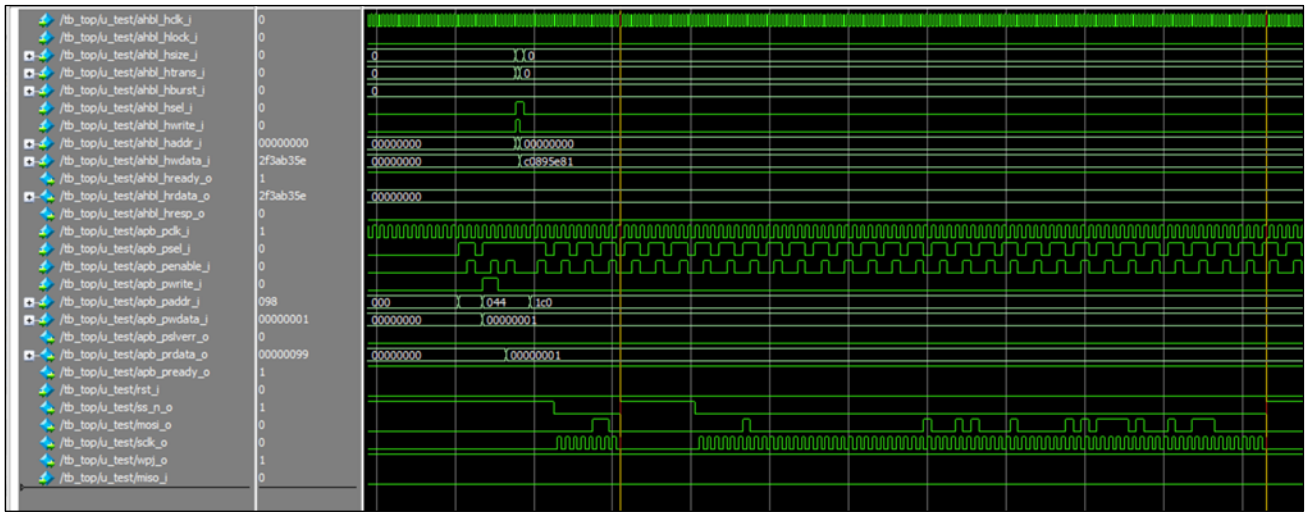

Figure 7.6. Simulation Waveform

7.3. Constraining the IP

It is your responsibility to provide proper timing and physical design constraints to ensure that the design meets the desired performance goals on the FPGA. The content of the following IP constraint file can be added to the user design constraints:

<IP_Instance_Path>/<IP_Instance_Name>/eval/constraint.pdc.

The above constraint file has been verified during IP evaluation with the IP instantiated directly in the top-level module. The constraint in this file can be modified given a complete understanding of the effect of the constraint.

To use this constraint file, copy the content of constraint.pdc to the top-level design constraint for post-synthesis.

Refer to Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059) for details on how to constraint the design.

Appendix A. Resource Utilization

Table A.1 and Table A.2 show the resource utilization of the SPI Flash Memory Controller IP Core using different devices using Synplify Pro of Lattice Radiant software. The default configuration is used, and some attributes are changed from the default value to show the effect on resource utilization.

Table A.1. Resource Utilization using LAV-AT-E70-3LFG1156I

Configuration	ahb_hclk_i Fmax (MHz)[1]	apb_pclk_i Fmax (MHz)[1]	Registers	LUTs[2]	EBRs
Default	250	81.16	982	1492	0
Enable Page Program Buffer: Checked, *Enable Page Read Buffer*: Checked, Others = Default	250	108.32	1083	2055	0
Enable Page Program Buffer: Checked, *Page Program Buffer Memory Type*: EBR, Others = Default	250	126.39	1028	1592	1
Enable Page Read Buffer: Checked, *Page Read Buffer Memory Type*: EBR, Others = Default	250	113.2	1043	1587	1
Enable Page Program Buffer: Checked, *Enable Page Read Buffer*: Checked, *Page Buffer Interface:* AHBL, *SCLK Rate*: 1, Others = Default	250	106.406	1153	2226	0

Notes:

1. Fmax is generated when the FPGA design only contains SPI Flash Controller IP Core, and the target frequency is 100 MHz for ahb_hclk_i and 50 MHz for apb_pclk_i. These values may be reduced when user logic is added to the FPGA design.
2. The distributed RAM utilization is accounted for in the total LUT4s utilization. The actual LUT4 utilization is distribution among logic, distributed RAM, and ripple logic.

Table A.2. Resource Utilization using LAV-AT-E70-1LFG1156I

Configuration	ahb_hclk_i Fmax (MHz)[1]	apb_pclk_i Fmax (MHz)[1]	Registers	LUTs[2]	EBRs
Default	250	63.19	982	1492	0
Enable Page Program Buffer: Checked, *Enable Page Read Buffer*: Checked, Others = Default	250	74.93	1083	2055	0
Enable Page Program Buffer: Checked, *Page Program Buffer Memory Type*: EBR, Others = Default	250	88.22	1028	1592	1
Enable Page Read Buffer: Checked, *Page Read Buffer Memory Type*: EBR, Others = Default	250	77.56	1043	1587	1
Enable Page Program Buffer: Checked, *Enable Page Read Buffer*: Checked, *Page Buffer Interface:* AHBL, *SCLK Rate*: 1, Others = Default	235.85	72.59	1153	2226	0

Notes:

1. Fmax is generated when the FPGA design only contains SPI Flash Controller IP Core, and the target frequency is 100 MHz for ahb_hclk_i and 50 MHz for apb_pclk_i. These values may be reduced when user logic is added to the FPGA design.
2. The distributed RAM utilization is accounted for in the total LUT4s utilization. The actual LUT4 utilization is distribution among logic, distributed RAM, and ripple logic.

References

- SPI Flash Memory Controller IP Release Notes (FPGA-RN-02039)
- A Step-By-Step Approach to Lattice Propel (FPGA-AN-02052)
- Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059)
- Avant-E web page
- Avant-G web page
- Avant-X web page
- Certus-NX web page
- CertusPro-NX web page
- CrossLink-NX web page
- MachXO5-NX web page
- SPI Flash Memory Controller IP Core web page
- CrossLink-NX PCIe Bridge Board web page
- Lattice Radiant Software web page
- Lattice Propel Design Environment web page
- Lattice Solutions IP Cores web page
- Lattice Solutions Reference Designs web page
- 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 2.1, IP v2.0.0, July 2025

Section	Change Summary
Introduction	In Table 1.1. Summary of the SPI Flash Memory Controller IP: • Added LFD2NX-15 and LFD2NX-25 to *Targeted Devices*.
References	• Added SPI Flash Memory Controller IP Core web page. • Updated listing name to Lattice Radiant Software web page.

Revision 2.0, IP v2.0.0, December 2024

Section	Change Summary
Cover	Added IP version.
Abbreviations in This Document	Updated section title, description, and table header.

Introduction	• In Table 1.1. Summary of the SPI Flash Memory Controller IP:
	  • Added Certus-NX-RT and CertusPro-NX-RT to *Supported FPGA Family*.
	  • Added IP Changes row.
	  • Removed IP Version row.
	  • Added LFD2NX-9 and LFD2NX-28 to *Targeted Devices*.
	  • Added Resources row.
	  • Updated IP core and software versions in *Lattice Implementation*.
	  • Added note on *Lattice Implementation*.
	  • Made minor editorial change.
	• Added the IP Support Summary section.
	• Removed the IP Validation Summary section.
	• Added the Hardware Support section.

Example Design	Added introductory paragraph to list the evaluation board used for the example design.
Designing with the IP	• Updated Figure 7.1. Module/IP Block Wizard, Figure 7.2. Configure User Interface of SPI Flash Memory Controller IP Core, Figure 7.3. Check Generated Result, Figure 7.4. Simulation Wizard, and Figure 7.5. Add and Reorder Source. • Removed the IP Evaluation section.
References	• Added SPI Flash Memory Controller IP Release Notes and Lattice Radiant Timing Constraints Methodology. • Removed Lattice Radiant Software User Guide, Lattice Propel Builder User Guide, and Lattice Propel SDK User Guide. • Added CrossLink-NX PCIe Bridge Board, Lattice Solutions IP Cores, and Lattice Solutions Reference Designs web pages. • Made minor editorial change.

Revision 1.9, June 2024

Section	Change Summary
All	Updated document title.
Acronyms in This Document	Added items.

Introduction	• In Overview of the IP section:
	  • Removed slave designation in relation to ports.
	  • Added additional description for control port.
	• Updated IP version and Lattice Implementation in Table 1.1. Summary of the SPI Flash Memory Controller IP in Quick Facts section.
	• Updated bus interface names to Data Port AHB-Lite Subordinate interface and Control Port APB Completer interface in Features section.
	• In IP Validation Summary section:

Section	Change Summary
	• Minor editorial changes. • Updated Lattice Avant IP version in Table 1.2. IP Validation Level.
Functional Description	• Updated Figure 2.1. SPI Flash Memory Controller IP Core Functional Diagram in IP Architecture Overview section. • Updated Byte/Halfword/Word Direct Read or Write section. • Updated description for Advanced Peripheral Bus Interface in Table 2.1. User Interfaces and Supported Protocols in User Interfaces section.

IP Parameter Description	• Minor formatting changes to the following figures in SPI First Transmitted Bit section:
	  • Figure 3.5. SPI Flash Write Transaction with MSB as First Transmitted Bit
	  • Figure 3.6. SPI Flash Read Transaction with MSB as First Transmitted Bit
	  • Figure 3.7. SPI Flash Write Transaction with LSB as First Transmitted Bit
	  • Figure 3.8. SPI Flash Read Transaction with LSB as First Transmitted Bit
	• Minor formatting changes to the following figures in AHB-Lite Byte Endianness section:
	  • Figure 3.9. SPI Flash Write Transaction
	  • Figure 3.10. SPI Flash Read Transaction
	• Minor formatting changes to the following figures in User Command section:
	  • Figure 3.13. SPI Write Transaction using User-Defined Command
	  • Figure 3.14. SPI Read Transaction using User-Defined Command

Signal Description	• Updated the description for the following ports in Table 4.1. SPI Flash Memory Controller IP Core Signal Description:
	  • ahbl_hwrite_i
	  • ahbl_hsel_i
	  • ahbl_hburst_i
	  • ahbl_hresp_o
	  • ahbl_hready_o
	  • apb_psel_i
	  • apb_pwrite_i
	  • apb_pready_o
	• Added the ahbl_hready_i port in Table 4.1. SPI Flash Memory Controller IP Core Signal Description.

Example Design	Added Example Design section.
Designing with the IP	In Generating and Instantiating the IP section: • Updated Figure 7.1. Module/IP Block Wizard. • Updated Figure 7.2. Configure User Interface of SPI Flash Memory Controller IP Core. • Updated Figure 7.3. Check Generated Result.
References	• Minor editorial changes. • Added reference A Step-By-Step Approach to Lattice Propel (FPGA-AN-02052). • Added link for Lattice Propel Software user guides and web page. • Added links for Avant-G and Avant-X web pages.

Revision 1.8, December 2023

Section	Change Summary
All	• Changed document title to *SPI Flash Memory Controller IP Core*. • Restructured the whole document upon the new IP user guide template.
Disclaimers	Updated this section.
Introduction	• Restructured all the subsections, adding the Quick Facts, Licensing and Ordering Information, IP Validation Summary sections and changing the Naming Conventions section title to the current. • General update to the features in the Features section.
Functional Description	• General update to the IP Architecture Overview section.

Section	Change Summary

	• Updated Figure 2.1. SPI Flash Memory Controller IP Core Functional Diagram.
	• Newly added the Clocking, Reset, and User Interfaces sections.
	• Other IP Specific Blocks/Layers/Interfaces section:
	  • restructured the whole section upon the original Section 2.1.1 to Section 2.1.4 in the previous version and made general update to contents;
	  • newly added User-Defined Flash Operations and IP Transaction Status sections.

IP Parameter Description	• Restructured the Attributes Summary section upon the original Section 2.3 by:
	  • updating Table 3.1. Attributes Summary, changing Dependency on Other Attributes for Control Port APB Address Width to *Display information only*;
	  • updating Table 3.2. Attributes Descriptions:
	  • changing Description for Enable Page Program Buffer to *Setting this to 1 indicates that page write data are locally stored before writing the entire page to SPI flash*;
	  • changing Description for Control Base Address to *Specifies the base address for the Control Port APB interface*;
	  • updating Description for First Transmitted Bit, Data Port AHB Address width, Data Port AHB Data Width, Data Port AHB Data Byte Endianness, Control Port APB Address Width, and Control Port APB Data Width attributes.
	• Restructured the IP Parameter Settings for Example Use Cases section upon the original Section 2.5 and made general update to the descriptions.
	• Removed the original Table 2.61 and Table 2.62 and newly added Table 3.5. AHB-Lite Interface Data Endianness.
	• Updated the Flash Command section by:
	  • adding the following APB interface register access to perform the SPI write transaction using Block Erase Type 1:
	  Register read from '0x1C0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation;
	  • updating Figure 3.11. SPI Transaction for Block Erase Type 1 upon the original Figure 2.12 in the previous version;
	  • changing the APB register access to perform the SPI read transaction using Status Read from *Wait for apb_ready_o* to "*Register read from '0x1C0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation*".
	• Updated the User Command section by:
	  • adding *"Register read from '0x1C0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation*" to the sample procedure for an SPI flash write transaction;
	  • adding *"Register read from '0x1C0' to check the IP transaction status while performing the flash operation. Wait until the read value is "0x0" to indicate end of the flash operation*" to the sample procedure for an SPI flash read transaction.

Signal Description	• Moved the original Section 2.2 to this section. • In Table 4.1. SPI Flash Memory Controller IP Core Signal Description, changed *S Port AHB-lite Interface* to *AHB-Lite Interface* and *C Port APB Interface* to *APB Interface*.

Register Description	• Moved the original Section 2.4 to this section.
	• Changed Register *0x00* to *0x000*, *0x04* to *0x004*, *0x08* to *0x008*, *0x0c* to *0x00c*, *0x10* to *0x010*, *0x14* to *0x014*, *0x18* to *0x018*, *0x1c* to *0x01c*, *0x20* to *0x020*, *0x24* to *0x024*, *0x28* to *0x028*, *0x2c* to *0x02c*, *0x30* to *0x030*, *0x40* to *0x040*, *0x44* to *0x044*, *0x80* to *0x080*, *0x84* to
	0x084, *0x88* to *0x088*, *0x8c* to *0x08c*, *0x90* to *0x090*, *0x94* to *0x094*, *0x98* to *0x098*, *0x9c* to *0x09c*, *0xa0* to *0x0a0*.
	• Table 5.1. Summary of SPI Flash Memory Controller IP Core Registers:
	  • updated Description of USER CMD Read Data 0 and USER CMD Read Data 1;
	  • removed the Byte Program CFG register;

Section	Change Summary
	• changed Register Name of 0x190, 0x194, and 0x198 to *Block Erase Type 1 CFG*, *Block Erase Type 2 CFG*, and *Block Erase Type 3 CFG* accordingly; • changed Register Name of 0x1b0 to *Deep Power Down CFG*, and updated its description; • changed Register Name of 0x1b4 to *Resume from Power Down CFG*, and updated its description. • In Table 5.5. Block Erase Type 1 Register, Table 5.6. Block Erase Type 2 Register, Table 5.7. Block Erase Type 3 Register, Table 5.8. Chip Erase Register, Table 5.11. Status Read Register, and Table 5.12. Status Write Register, Table 5.15. Manufacturer ID Register, and Table 5.16. SPI Flash Address Register, and Table 5.19. User CMD Address Register, updated Description of the corresponding register. • In Table 5.28. Page Program Size Register, updated the Bits column. • In Table 5.29. Page Read Size Register, updated the Bits and Description columns. • Removed the original Section 2.4.37. • In the Deep Power Down CFG Register 0x1b0 section, updated the register name to *Deep Power Down CFG*. • In the Resume from Power Down CFG Register 0x1b4 section, updated the register name to *Resume from Power Down CFG*. • In the Read Manufacturer ID CFG Register 0x1b8 section, updated the register name *to Read Manufacturer ID*. • In Table 5.50. Transaction Status Register, updated the Name column. • Updated Bits, Access Mode, and Description in Table 5.53. Page Program Buffer Register and Table 5.54. Page Read Buffer Register.
Designing with the IP	• Changed the section title from *IP Generation, Simulation, and Validation* to *Designing with the IP*. • Changed the section title of Generating and Instantiating the IP to the current. • Updated Figure 6.1. Module/IP Block Wizard, Figure 6.2. Configure User Interface of SPI Flash Memory Controller IP Core, Figure 6.3. Check Generated Result, and Figure 6.6. Simulation Waveform. • Restructured information of generated files of the IP core into Generated Files and File Structure. • In Table 6.1. Generated File List, newly added the *eval/constraint.pdc* attribute. • Newly added the Constraining the IP section.
Resource Utilization	• Removed the original Table A.1 and Table A.2. • Changed device name from *LAV-AT-500E-3LFG1156I* to *LAV-AT-E70-3LFG1156I* and updated the corresponding resource utilization in Table A.1. Resource Utilization using LAV-AT-E70-3LFG1156I. • Changed device name from *LAV-AT-500E-1LFG1156I* to *LAV-AT-E70-1LFG1156I* and updated the corresponding resource utilization in Table A.2. Resource Utilization using LAV-AT-E70-1LFG1156I.

Revision 1.7, August 2023

Section	Change Summary
Functional Description	• Added Flash Memory Map Size attribute in Table 2.2. Attributes Table and Table 2.3. Attributes Descriptions. • Added offset addresses Transaction Status, Page Read Buffer Data Count Status, and Page Read Buffer Address Status for status registers in Table 2.4. Summary of SPI Flash Memory Controller IP Core Registers. • Added RW1C access type in Table 2.5. Access Type Definition. • Added Transaction Status Register 0x1C0, Page Read Buffer Data Count Status Register 0x1C4, and Page Read Buffer Address Register 0x1C8 sections.
IP Generation, Simulation, and Validation	• Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure User Interface of SPI Flash Memory Controller IP Core, and Figure 3.3. Check Generated Result as per IP

Section	Change Summary
	spi_flash_controller version 1.7.0. • Added attributes dut_inst.v, dut_params.v, tb_top.v and test_stimuli.v in Table 3.1. Generated File List.

Revision 1.6, May 2023

Section	Change Summary
All	Removed sector size information.
Functional Description	• Updated subsection headings under Register Description. • Updated 0 to 1 in the description of Table 2.31. Page Program Size Register and Table 2.32. Page Read Size Register.
IP Generation, Simulation, and Validation	Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure User Interface of SPI Flash Memory Controller IP Core, and Figure 3.3. Check Generated Result as per IP spi_flash_controller Version 1.6.0.

Revision 1.5, April 2023

Section	Change Summary
Introduction	Added *Supports all SPI clocking modes (combination of clock polarity and clock phase)* in Features section.

Functional Description	• Added SCLK polarity and SCLK phase attributes in Table 2.2. Attributes Table and Table 2.3. Attributes Descriptions.
	• Added SCLK polarity and SCLK phase registers in Table 2.4. Summary of SPI Flash Memory Controller IP Core Registers.
	• Updated from location to bit in Table 2.11. Chip Erase Register, Table 2.12. Write Enable Register, and Table 2.13. Write Disable Register.
	• Updated description from Indicates the current status of SPI flash after read to this register to Read the Status Register of SPI flash in Table 2.14. Status Read.
	• Updated below figures to correct the vertical broken lines:
	  • Figure 2.2. Clocking Mode 0 (SCLK Polarity=0, SCLK Phase=0)
	  • Figure 2.3. Clocking Mode 1 (SCLK Polarity=0, SCLK Phase=1)
	  • Figure 2.4. Clocking Mode 2 (SCLK Polarity=1, SCLK Phase=0)
	  • Figure 2.5. Clocking Mode 3 (SCLK Polarity=1, SCLK Phase=1)
	  • Figure 2.6. SPI Flash Write Transaction with MSB as First Transmitted Bit
	  • Figure 2.7. SPI Flash Read Transaction with MSB as First Transmitted Bit
	  • Figure 2.8. SPI Flash Write Transaction with LSB as First Transmitted Bit
	  • Figure 2.9. SPI Flash Read Transaction with LSB as First Transmitted Bit
	  • Figure 2.10. SPI Flash Write Transaction
	  • Figure 2.11. SPI Flash Read Transaction
	  • Figure 2.12. SPI Transaction for Block Erase Type 1
	  • Figure 2.13. SPI Transaction for Status Read
	  • Figure 2.14. SPI Write Transaction using User-Defined Command
	  • Figure 2.15. SPI Read Transaction using User-Defined Command
	• Added SCLK Polarity Register and SCLK Phase Register sections.
	• Added Clocking Modes (SCLK Polarity and SCLK Phase) section.
	• Deleted Figure 2.2. Basic transaction on SPI Flash Controller.
	• Added SCLK to the title of Clocking Modes (SCLK Polarity and SCLK Phase) section.

IP Generation, Simulation, and Validation	Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure User Interface of SPI Flash Memory Controller IP Core, and Figure 3.3. Check Generated Result as per IP spi_flash_controller Version 1.5.0.
Technical Support Assistance	Added FAQ website link in Technical Support Assistance section.

Revision 1.4, January 2023

Section	Change Summary
Technical Support Assistance	Added reference link to Lattice Answer Database.
Revision History	Updated to incorporate Lattice standards and guidelines.

Revision 1.3, November 2022

Section	Change Summary
Introduction	• Removed Quick Facts section. • Control port feature is fixed to enable.
Functional Description	• Updated Figure 2.1. SPI Flash Memory Controller IP Core Functional Diagram. • Updated Table 2.1. SPI Flash Memory Controller IP Core Signal Description to add ahb_hclk_i and apb_pclk_i. • Added the attributes First Transmitted Bit and Data Port AHB Data Byte Endianness to Table 2.2. Attributes Table. • Removed Control Port from Table 2.2. Attributes Table and Table 2.3. Attributes Descriptions. • Updated Table 2.4. Summary of SPI Flash Memory Controller IP Core Registers based on added registers. • Updated Register Description section based on added registers. • Added transaction waveforms in Operations Details section.
IP Generation, Simulation, and Validation	• Updated section names. • Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure User Interface of SPI Flash Memory Controller IP Core, Figure 3.3. Check Generated Result, Figure 3.4. Simulation Wizard, Figure 3.5. Add and Reorder Source, and Figure 3.6. Simulation Waveform.
Appendix A	Added the Resource Utilization section.

Revision 1.2, May 2021

Section	Change Summary
Introduction	Added MachXO5-NX to the Supported FPGA Family, and LFMXO5-25 to the Supported User Interfaces in Table 1.1. Quick Facts.
Core Generation, Simulation, and Validation	Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure User Interface of SPI Flash Memory Controller IP Core, and Figure 3.3. Check Generated Result to show the latest version 1.2.0.
Appendix A. Resource Utilization	• Added Table A.1. Resource Utilization for resource utilization of the SPI Flash Controller IP Core for the LFMXO5-25-9BBG400I. • Added Table A.2. Resource Utilization for resource utilization of the SPI Flash Controller IP Core for the LFMXO5-25-7BBG400I.

Revision 1.1, June 2021

Section	Change Summary
All	Minor adjustments in formatting.
Introduction	Updated section content, including Table 1.1. Quick Facts to add CertusPro-NX support.
References	Added webpage for CertusPro-NX.

Revision 1.0, December 2020

Section	Change Summary
All	Initial release.

www.latticesemi.com