MachXO4 Hardened Control Functions User Guide

Technical Note

FPGA-TN-02403-1.1

September 2026

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.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Contents

Contents 3
Abbreviations in This Document 6
1. Introduction 7
2. WISHBONE Bus Interface 9
2.1. WISHBONE Protocol 10
2.2. WISHBONE Design Tips 10
3. Generating an EFB Module with IP Catalog 11
4. Hardened I2C IP Cores 15
4.1. Primary I2C 16
4.2. Secondary I2C 18
4.3. Configuring I2C Cores with IP Catalog 19
4.3.1. General Call Enable 20
4.3.2. Wake-up Enable 20
4.3.3. I2C Bus Performance 20
4.3.4. I2C Addressing 20
4.3.5. MachXO4 I2C Usage Cases 20
4.3.6. I2C Design Tips 22
5. Hardened SPI IP Core 23
5.1. SPI Interface Signals 24
5.2. Configuring the SPI Core with IP Catalog 26
5.2.1. SPI Mode 26
5.2.2. SPI Controller Clock Rate 27
5.2.3. SPI Protocol Options 27
5.2.4. Controller Chip Selects 27
5.2.5. SPI Controller Interrupts 27
5.2.6. Wake-up Enable 28
5.2.7. MachXO4 SPI Usage Cases 28
5.2.8. SPI Design Tips 30
6. Timer/Counter 31
6.1. Timer/Counter Modes of Operation 32
6.1.1. Clear Timer on Compare Match Mode 32
6.1.2. Watchdog Timer Mode 32
6.1.3. Fast PWM Mode 33
6.1.4. Phase and Frequency Correct PWM Mode 33
6.2. Timer/Counter IP Signals 34
6.3. Configuring Timer/Counter 34
6.3.1. Timer/Counter Mode 35
6.3.2. Output Function 35
6.3.3. Clock Edge Selection 35
6.3.4. Pre-scale Divider Value 36
6.3.5. Timer/Counter Top 36
6.3.6. Output Compare Value 36
6.3.7. Enable Interrupt Registers 36
6.3.8. MachXO4 Timer/Counter Usage Cases 36
6.3.9. Timer/Counter Design Tips 38
7. Flash Access 39
7.1. Flash Access Ports 40
7.2. Interface to UFM 41
7.2.1. Initializing the UFM with IP Catalog 41
7.3. Configuration Flash 44
7.3.1. Flash Design Tips 45
8. Interface to Dynamic PLL Configuration Settings 46

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

9. MachXO4 RoT Tamper Detection 48
References 49
Technical Support Assistance 50
Revision History 51

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Figures

Figure 1.1. Embedded Functional Block (EFB) 7
Figure 2.1. WISHBONE Bus Interface between the FPGA Core and the EFB Module 9
Figure 3.1. EFB Module in IP Catalog 11
Figure 3.2. Generating an EFB Module with IP Catalog 12
Figure 4.1. I2C Block Diagram 15
Figure 4.2. I2C Primary Block Diagram 16
Figure 4.3. I2C Secondary Block Diagram 18
Figure 4.4. Configuring I2C Functions of the EFB Module with IP Catalog 19
Figure 4.5. I2C Circuit 21
Figure 5.1. SPI Block Diagram 23
Figure 5.2. Configuring SPI Functions of the EFB Module with IP Catalog 26
Figure 5.3. External Controller SPI Device Accessing the Target MachXO4 User SPI 29
Figure 5.4. MachXO4 User SPI Controller Accessing One or Multiple External Target SPI Devices 29
Figure 5.5. External Controller SPI Device Accessing the MachXO4 Configuration Logic 30
Figure 6.1. Timer/Counter Block Diagram 31
Figure 6.2. Timer/Counter Output Waveform 32
Figure 6.3. PWM Waveform Generation 33
Figure 6.4. Phase and Frequency Correct PWM Waveform Generation 34
Figure 6.5. Configuring Timer/Counter 35
Figure 7.1. Flash Block Diagram 40
Figure 7.2. Initializing the UFM Sector with IP Catalog 42
Figure 7.3. UFM in Radiant Spreadsheet View 44
Figure 8.1. EFB Interface to Dynamic PLL 46
Figure 8.2. Interface to Dynamic PLL Configuration Settings 47

Tables

Table 1.1. EFB Memory Map 8
Table 2.1. WISHBONE Target Interface Signals of the EFB Module 9
Table 4.1. Hardened I2C Functionality 15
Table 4.2. I2C Primary – IP Signals 17
Table 4.3. I2C Secondary – IP Signals 18
Table 5.1. Hardened SPI Functionality 23
Table 5.2. SPI – IP Signals 25
Table 6.1. Timer/Counter – IP Signals 34
Table 7.1. MachXO4 Flash Access 39
Table 7.2. UFM Resources in MachXO4 (LFMXO4) Devices 41
Table 7.3. UFM Resources in MachXO4 RoT Devices 41
Table 7.4. Configuration Flash Resources in MachXO4 (LFMXO4) Devices 44
Table 7.5. Configuration Flash Resources in MachXO4 RoT Devices 44
Table 8.1. PLL Interface – IP Signals 47

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
CPU	Central Processing Unit
EBR	Embedded Block Random Access Memory
EFB	Embedded Functional Block
FIFO	First-In First-Out
FPGA	Field Programmable Gate Array
GPIO	General Purpose Input/Output
GSR	Global Set/Reset
I/O	Input/Output
I2C	Inter-Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
IP	Intellectual Property
IRQ	Interrupt Request
JTAG	Joint Test Action Group
LSB	Least Significant Bit
LUT	Look-Up Table
MDF	Mode Fault
MSB	Most Significant Bit
OCRF	Output Compare Flag
OSC	Oscillator
OVF	Overflow Flag
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
RAM	Random Access Memory
ROE	Receiver Overrun Error
RoT	Root of Trust
RTL	Register Transfer Level
SCL	Serial Clock
SDA	Serial Data
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
UFM	User Flash Memory
VHDL	VHSIC Hardware Description Language

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

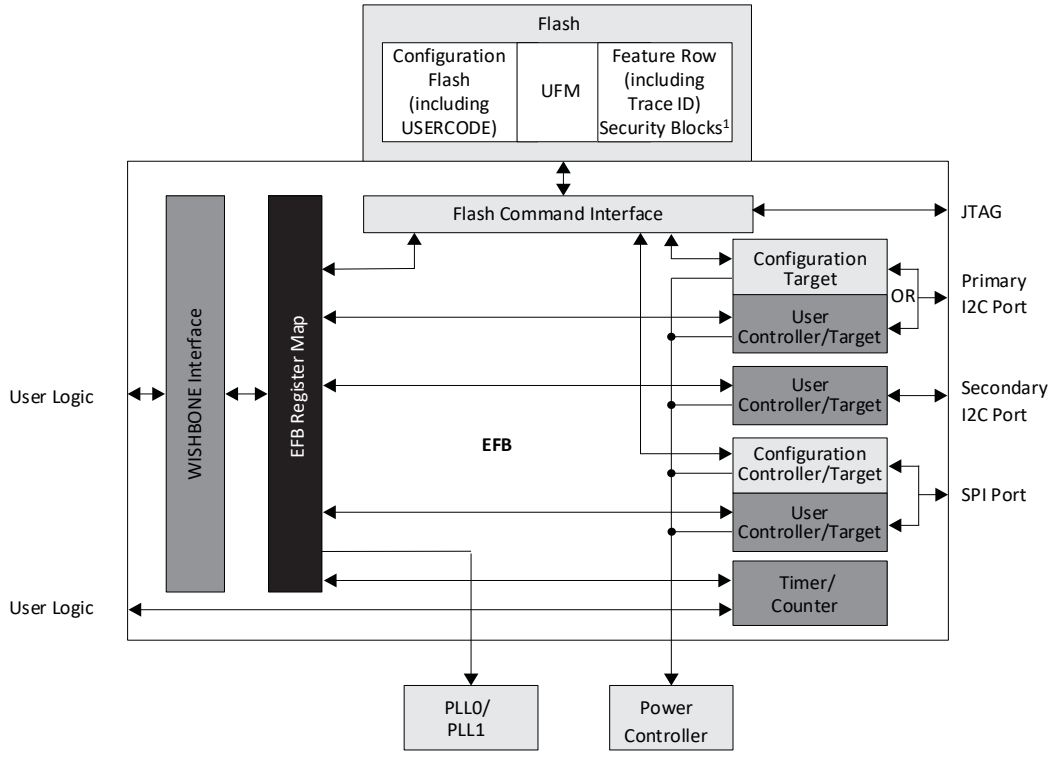
1. Introduction

Note: In this document, *MachXO4* refers to both LFMXO4 and LFMXO4D unless explicitly identified as MachXO4 (LFMXO4). *MachXO4 Root of Trust (RoT)* refers to only *LFMXO4D*.

The Lattice MachXO4™ FPGA family combines a high-performance, low-power FPGA fabric with built-in, hardened control functions. The hardened control functions ease design implementation and save general-purpose resources such as LUTs, registers, clocks, and routing. The hardened control functions are physically located in the Embedded Function Block (EFB). All MachXO4 devices include an EFB module. The EFB block includes the following control functions:

- Two I2C cores
- One SPI core
- One 16-bit timer/counter
- Interface to Flash memory which includes:
  - User Flash Memory
  - Configuration logic
  - Feature row and security blocks for tamper detection (only for MachXO4 RoT devices)
- Interface to Dynamic PLL configuration settings
- Interface to On-chip Power Controller through I2C and SPI.

Figure 1.1 shows the EFB architecture and the WISHBONE interface to the FPGA user logic.



Note:
1. Security blocks are available only for MachXO4 RoT devices.

Figure 1.1. Embedded Functional Block (EFB)

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The hard SPI, I2C, and Timer/Counter IPs contained in the EFB can save more than 500 LUTs when compared to implementing these same functions in FPGA logic using Lattice reference designs.

The EFB Register Map is used to access the EFB hardened functions through the Target WISHBONE bus. Each hard IP has dedicated 8-bit Data and Control registers, with the exception of the Flash Memory (UFM/Configuration), which is accessed through the same set of registers. Ports having access to the EFB Register Map have access to all registers. For example, from the Primary I2C Target port, you can access the Timer/Counter registers. The EFB Register Map is shown below:

Table 1.1. EFB Memory Map

Address Range (Hex)	8-bit Data/Control Registers Function
0x00-0x1F	PLL0 Dynamic Access[1]
0x20-0x3F	PLL1 Dynamic Access[1]
0x40-0x49	I2C Primary
0x4A-0x53	I2C Secondary
0x54-0x5D	SPI
0x5E-0x6F	Timer/Counter
0x70-0x75	Flash Memory (UFM/Configuration)
0x76-0x77	EFB Interrupt Source

Note:

1. There can be up to two PLLs in a MachXO4 device. PLL0 has an address range from 0x00 to 0x1F. PLL1 (if present) has an address range from 0x20 to 0x3F. Refer to the MachXO4 sysCLOCK PLL Design User Guide (FPGA-TN-02391) for details on PLL configuration registers and recommended usage.

The EFB module is represented in design software as a primitive and it is described in this document. Use IP Catalog to configure the EFB and to generate Verilog or VHDL source code. The source code is instantiated in your design.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2. WISHBONE Bus Interface

The WISHBONE bus in the MachXO4 devices is compliant with the WISHBONE standard from OpenCores. The bus provides connectivity between the FPGA user logic and the EFB functional blocks, as well as connectivity between the individual EFB functional blocks. The User Logic must include a WISHBONE Controller interface to communicate with the WISHBONE Target interface of the EFB. An example of a WISHBONE Controller is the LatticeMico8™ soft processor.

The block diagram in Figure 2.1 shows the WISHBONE bus signals between the FPGA core and the EFB. Table 2.1 provides a detailed definition of the signals.

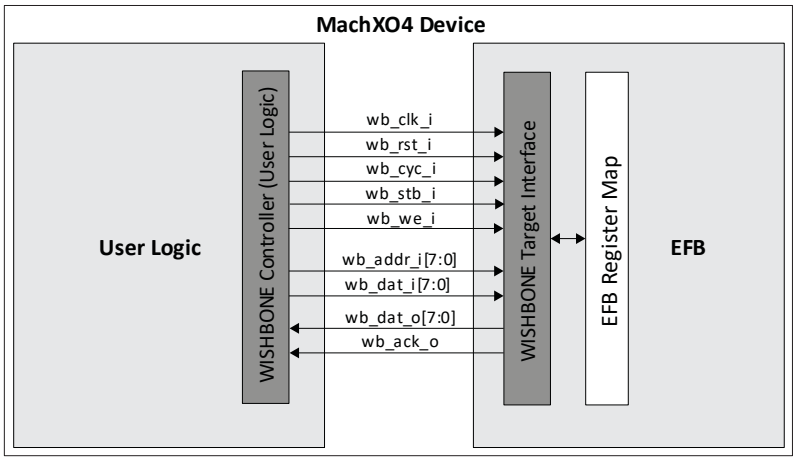


Figure 2.1. WISHBONE Bus Interface between the FPGA Core and the EFB Module

Table 2.1. WISHBONE Target Interface Signals of the EFB Module

Signal Name	I/O	Width	Description
wb_clk_i	Input	1	Positive edge clock used by WISHBONE interface registers and hardened functions within the EFB module. Supports clock speeds up to 133 MHz.
wb_rst_i	Input	1	Synchronous reset signal that resets the WISHBONE interface logic. This signal does not affect the contents of any registers and terminates an active bus cycle. Wait 1 µs after de-assertion before starting any subsequent WISHBONE transactions.
wb_cyc_i	Input	1	Asserted by the WISHBONE controller, indicating a valid bus cycle is present on the bus.
wb_stb_i	Input	1	Strobe signal indicating the WISHBONE Target is the target for the current transaction on the bus. The EFB module asserts an acknowledgment in response to the assertion of the strobe.
wb_we_i	Input	1	Level-sensitive Write/Read control signal. Low indicates a Read operation, and high indicates a Write operation.
wb_adr_i	Input	8	8-bit wide address used to select an EFB specific register.
wb_dat_i	Input	8	A WISHBONE Controller writes data to the addressed EFB register using the wb_dat_i bus during write cycles.
wb_dat_o	Output	8	A WISHBONE Controller receives data from the addressed EFB register using wb_dat_o during read memory cycles.
wb_ack_o	Output	1	Signals the WISHBONE Controller the bus cycle is complete and data written to the EFB is accepted. Data read from the EFB is valid.

To interface to the EFB, you must create a WISHBONE Controller in the User Logic. In a multiple-controller configuration, the WISHBONE Controller outputs are multiplexed in a user-defined arbiter. A LatticeMico8 soft processor can also be used along with the Mico System Builder platform, which can implement multi-controller bus configurations. If two Controllers request the bus in the same cycle, only the outputs of the arbitration winner reach the Target interface.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2.1. WISHBONE Protocol

For more information on the WISHBONE protocol and command sequences, refer to the WISHBONE section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

2.2. WISHBONE Design Tips

- Take note when dynamically turning off components for power savings; many of the EFB features require the MachXO4 device internal oscillator to be enabled even if it is not the source of the WISHBONE Clock. The following features can be used when the OSC is disabled:
  - SPI: User Target or User Controller modes.
  - I2C: After SDA delay is turned off by setting I2C_1_CR[3:2]=11, the OSC can be turned off. Following this, the OSC can be disabled and User Target or User Controller can operate.
  - Timer/Counter: The user clock must be selected.
- If the EFB WISHBONE input signals are not used, connect them to 0.
- To ensure correct operation, wb_cyc_i must be asserted for the entire WISHBONE transaction. For the EFB WISHBONE interface, wb_cyc_i and wb_stb_i may be connected together.
- For more information on the WISHBONE specification, go to the OpenCores website.
- For more information on Lattice reference designs, refer to the Lattice Solutions Reference Designs web page.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3. Generating an EFB Module with IP Catalog

IP Catalog is used to configure the EFB hard IP functions and generate the EFB module. From the Lattice Radiant™ top menu, select **Tools > IP Catalog**. With a MachXO4 device targeted for the Radiant project, the IP Catalog window opens and the EFB module can be found under **Modules > Architecture Modules**.

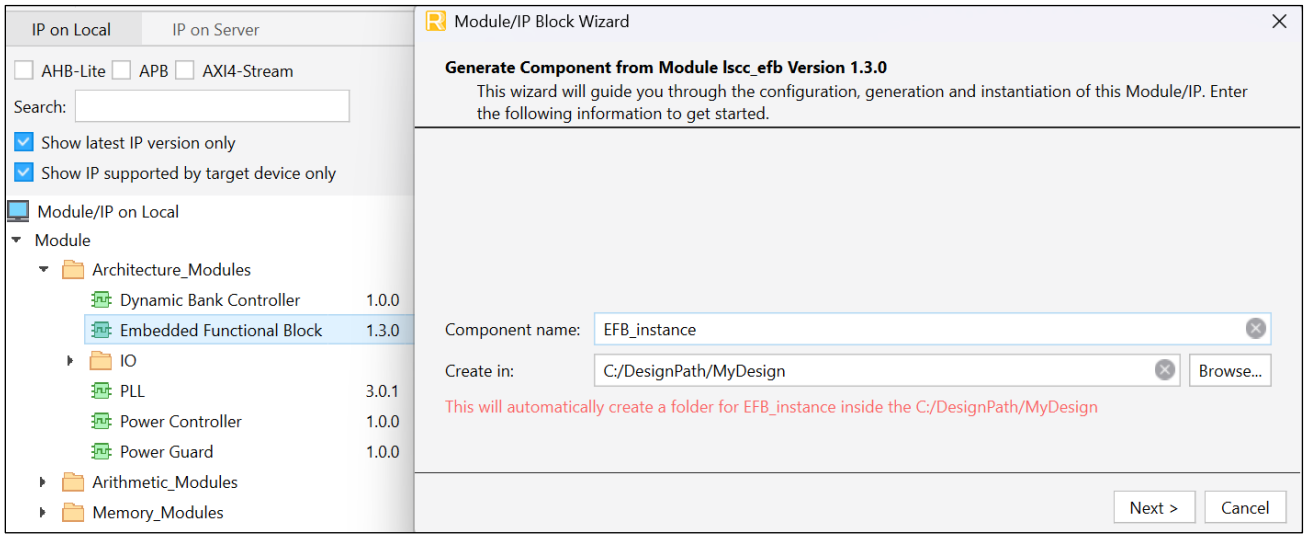


Figure 3.1. EFB Module in IP Catalog

Fill in the **Component name** and **Create in** fields, and click **Next**.

After clicking on the **Next** button, the EFB configuration dialog appears. The left side of the EFB window displays a graphical representation of the I/O associated with each IP function. The I/O pins appear and disappear as each IP is enabled or disabled. The initial tab is used to enable the hardened functions, the dynamic access to the PLL configuration settings, and enter the WISHBONE Clock Frequency. An example EFB with all features enabled is shown in Figure 3.2.

The hardened IP functions have individual tabs in the EFB window for individual configuration settings. These tabs are discussed later in the document, with the technical description of the specific functions. When all functions have been configured, click on the **Generate** button, and the EFB module is generated and ready to be instantiated in your design.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

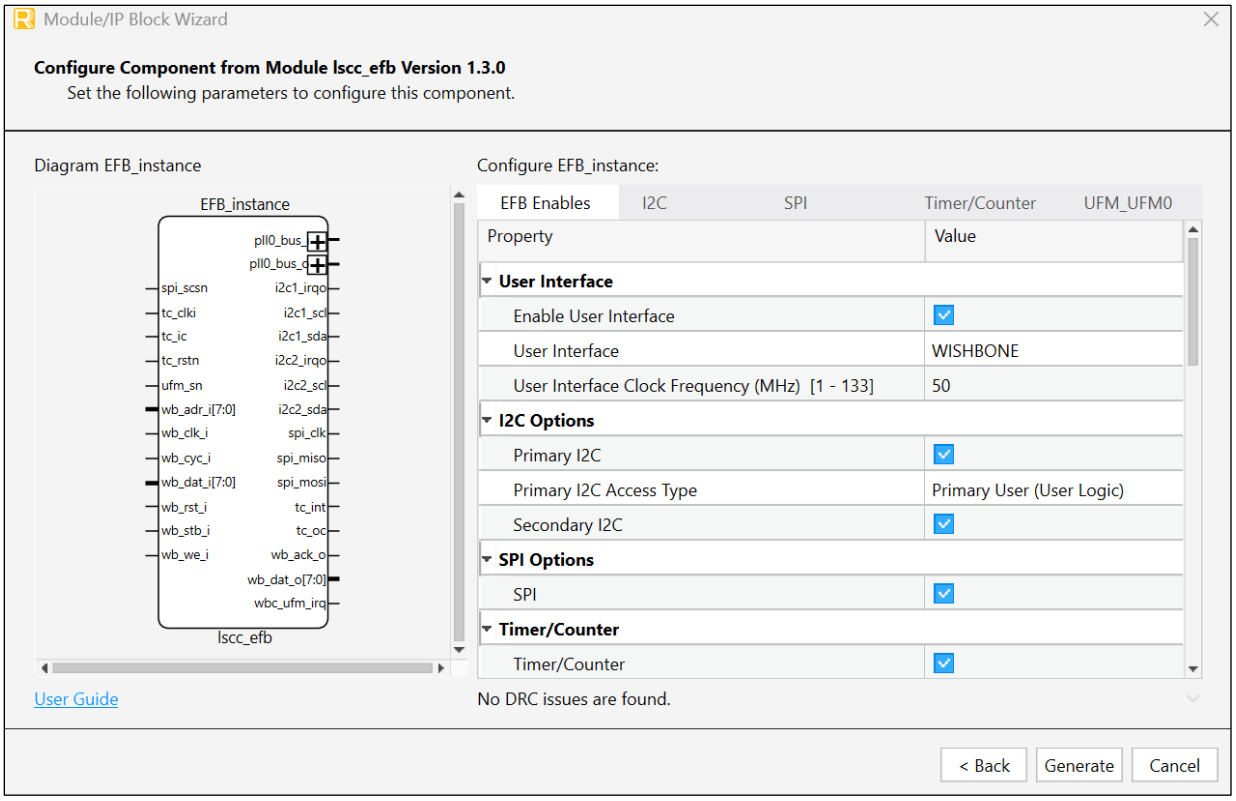


Figure 3.2. Generating an EFB Module with IP Catalog

The number of available PLL modules depends on the device density, and this is reflected in the IP Catalog EFB user interface. The LFMXO4-010, LFMXO4-015, and LFMXO4-025 devices each have one PLL, and the LFMXO4-050, LFMXO4-080, and LFMXO4-110 devices each have two PLLs available for dynamic access through the EFB WISHBONE Target interface.

The default WISHBONE Clock Frequency is set to 50 MHz. You can enter a clock frequency up to 133 MHz. The WISHBONE clock is used by the EFB WISHBONE interface registers and also by the SPI and I2C hardened IP cores. Many of the EFB features require the MachXO4 device internal oscillator to be enabled even if it is not the source of the WISHBONE Clock. The following features can be used when the OSC is disabled:

- SPI: User Target or User Controller modes.
- I2C: After SDA delay is turned off by setting I2C_1_CR[3:2]=11, the OSC can be turned off. Following this the OSC can be disabled and User Target or User Controller can operate.
- Timer/Counter: The user clock must be selected.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Like other modules, the EFB settings can be viewed in the Map Report. Examples for the MachXO4 devices are shown below:

- MachXO4 (LFMXO4) devices:

```
Embedded Functional Block Connection Summary:
-------------------------------------------
Desired WISHBONE clock frequency: 2.0 MHz Clock source:      clk
Reset source:  wb_rst
Functions mode:
I2C #1 (Primary) Function:    ENABLED I2C #2 (Secondary) Function:  DISABLED SPI Function: ENABLED
Timer/Counter Function:       DISABLED
Timer/Counter Mode:    WB
PLL0 Connection:       DISABLED
PLL1 Connection:       DISABLED
I2C Function Summary:
--------------------
I2C Component: PRIMARY
I2C Addressing:7BIT
I2C Performance:       100kHz
Slave Address: 0b0001001
General Call:  ENABLED
I2C Wake Up:   DISABLED
I2C Component: Configuration
I2C Addressing:7BIT
I2C Performance:       100kHz
Slave Address: 0b0001000 SPI Function Summary:
--------------------
SPI Mode:      BOTH
SPI Data Order:LSB to MSB SPI Clock Inversion:       DISABLED SPI Phase Adjust:    DISABLED
SPI Wakeup:    DISABLED
Timer/Counter Function Summary:
-----------------------------
None
```

- MachXO4 RoT devices:

```
Embedded Functional Block Connection Summary:
-------------------------------------------
Desired WISHBONE clock frequency:     2.0 MHz
Clock source:                         clk
Reset source:                         wb_rst
Functions mode:
       I2C #1 (Primary) Function:     ENABLED
       I2C #2 (Secondary) Function:   DISABLED
       SPI Function:                  ENABLED
       Timer/Counter Function:               DISABLED
       Timer/Counter Mode:            WB
PLL0 Connection:                      DISABLED
PLL1 Connection:                      DISABLED
I2C Function Summary:
--------------------
       I2C Component:                 PRIMARY
       I2C Addressing:                7BIT
       I2C Performance:                      100 kHz
       Slave Address:                 0b0001001
       General Call:                  ENABLED
       I2C Wake Up:                   DISABLED
       I2C Component:                 Configuration
       I2C Addressing:                7BIT
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

```
        I2C Performance:                        100 kHz
        Slave Address:                  0b0001000
SPI Function Summary:
--------------------
        SPI Mode:                               BOTH
        SPI Data Order:                 LSB to MSB
        SPI Clock Inversion:            DISABLED
        SPI Phase Adjust:                       DISABLED
SPI Wakeup:                             DISABLED
Timer/Counter Function Summary:
------------------------------
        None
UFM Function Summary:
--------------------
        UFM Utilization:                        EBR Initialization
        Available General
        Purpose Flash Memory:           511 Pages (511×128 Bits)

        EBR Blocks with Unique
        Initialization Data:            6

        WID                     EBR Instance
        ---                     ------------
        0b0000000011    LCDCharMap_inst/LCDCharMap_0_0_0
        0b0000000100
STRING_TABLE_INST/EXT_ROM_INST/pmi_romXhmenusdn8101024_0_0_0
        0b0000000101
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_1_1_0
        0b0000000110
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_0_0_3
        0b0000000111
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_0_1_2
        0b0000001000
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_1_0_1
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

4. Hardened I2C IP Cores

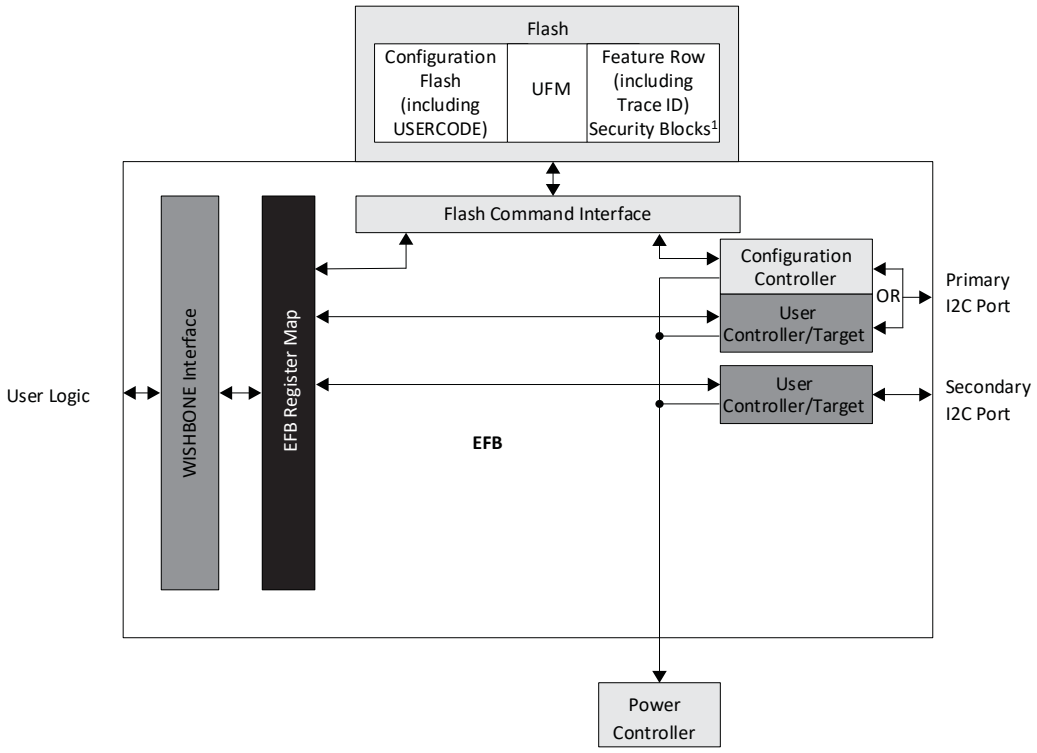
I2C is a widely used two-wire serial bus for communication between devices on the same board. Every MachXO4 device contains two hardened I2C IP cores designated as the Primary and Secondary I2C IP cores. The two cores in the MachXO4 device can operate as an I2C Controller or as an I2C Target. The difference between the two cores is that the Primary core has pre-assigned I/O pins while the ports of the Secondary core can be assigned to any general purpose I/O. In addition, the Primary core also has access to the Flash Memory (UFM/Configuration) through the Flash Command Interface. The hardened I2C IP core functionality and block diagram are shown below.

Table 4.1. Hardened I2C Functionality

	Primary I2C Configuration	Primary I2C User	Secondary I2C User
I2C Port as Controller	No	Yes	Yes
I2C Port as Target	Yes[1]	Yes[1]	Yes
Access the Flash Memory (UFM/Configuration)	Yes[1]	No	No
Access the User Logic	No	Yes	Yes
Must use dedicated I/O	Yes	Yes	No
Wake Power Controller from Standby Mode	Yes	Yes	Yes
Enter Power Controller Standby Mode	Yes	No	No

Note:

1. Primary port can be used as Configuration port or as a User port, but not both.



Note:
1. Security blocks are available only for MachXO4 RoT devices.

Figure 4.1. I2C Block Diagram

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

When an EFB I2C core is a controller, it can control other devices on the I2C bus through the physical interface. When an EFB I2C core is the Target, the device can provide I/O expansion to an I2C Controller. Both MachXO4 Primary and Secondary cores support the following I2C functionality:

- Controller/Target mode support
- 7-bit and 10-bit addressing
- Supports 50 kHz, 100 kHz, and 400 kHz data transfer speed
- General Call support (addresses all devices on the bus using the I2C address 0)
- Interface to User Logic through the EFB WISHBONE Target interface

4.1. Primary I2C

The MachXO4 Primary I2C Controller is shown in Figure 4.2. The main functions of the Primary Controller are:

- Either:
  - I2C Configuration Target provides access to the Flash; or
  - I2C User Target provides access to the User Logic.
- I2C Configuration or User Target provides access to the MachXO4 Power Controller.
- I2C User Controller provides access to peripherals attached to the MachXO4 device.

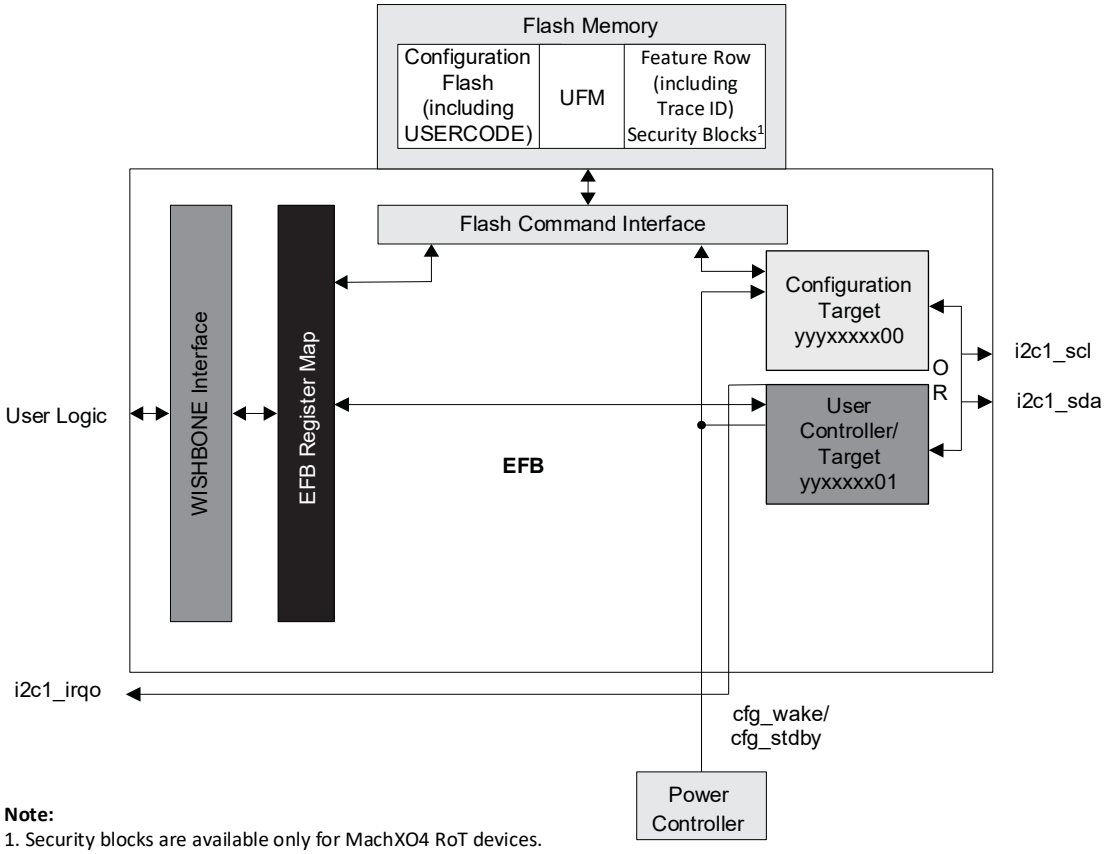


Figure 4.2. I2C Primary Block Diagram

The Primary I2C core can be used for accessing the Flash. However, the Primary I2C port cannot be used for both Flash access and user functions in the same design. The block diagram in Figure 4.2 shows an interface between the I2C block and the Flash. For more information on programming the MachXO4 through the I2C port, refer to the I2C section of the MachXO4 Programming and Configuration User Guide (FPGA-TN-02393).

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Target I2C peripherals on a bus are accessed by the Controller I2C calling the Target's unique addresses. The Primary Configuration address is yyyxxxxx00 and the Primary User address is yyyxxxxx01, where y and x are user programmable from IP Catalog.

The Primary Configuration I2C can be used to wake the Power Controller from Standby or enter Standby. The Primary User can only be used to wake the Power Controller from Standby mode. For more information on the Power Controller, refer to the Power and Thermal Estimation and Management for MachXO4 Devices (FPGA-TN-02409). The I2C Power Controller features can be set up through IP Catalog as documented later, and the register settings are defined in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

Table 4.2 documents the IP signals of the Primary I2C cores.

Table 4.2. I2C Primary – IP Signals

Signal Name	Pre-Assigned Pin Name	I/O	Width	Description
i2c1_scl	SCL	Bidirectional	1	Open drain clock line of the I2C core. The signal is an output if the I2C core is performing a Controller operation. The signal is an input for Target operations. This signal must be brought to the top level of the user RTL design. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of I2C ports in each MachXO4 device.
i2c1_sda	SDA	Bidirectional	1	Open drain data line of the I2C core. The signal is an output when data is transmitted from the I2C core. The signal is an input when data is received into the I2C core. This signal must be brought to the top level of the user RTL design. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of I2C ports in each MachXO4 device.
i2c1_irqo	—	Output	1	Interrupt request output signal of the I2C core. The intended use of this signal is for it to be connected to a WISHBONE Controller (a microcontroller or state machine) and request an interrupt when a specific condition is met. These conditions are described in the I2C section of the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
cfg_wake	—	Output	1	Wake-up signal – Hardwired signal to be connected ONLY to the Power Controller of the MachXO4 device for functional simulation support. The signal is enabled only if the Wakeup Enable feature has been set within the EFB user interface, I2C Tab.
cfg_stdby	—	Output	1	Standby signal – Hardwired signal to be connected ONLY to the Power Controller of the MachXO4 device for functional simulation support. The signal is enabled only if the Wakeup Enable feature has been set within the EFB user interface, I2C Tab.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

4.2. Secondary I2C

The Secondary I2C controller in the MachXO4 devices provides the same functionality as the Primary I2C controller with the exception of access to the MachXO4 Configuration Logic. The i2c2_scl and i2c2_sda ports are routed through the general purpose routing of the FPGA fabric and you can assign them to any General Purpose I/O (GPIO).

The Secondary I2C can be used to wake the Power Controller from Standby mode. For more information on the Power Controller, refer to the Power and Thermal Estimation and Management for MachXO4 Devices (FPGA-TN-02409). The I2C Power Controller features can be set up through IP Catalog as documented later, and the register settings are defined in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

Target I2C peripherals on a bus are accessed by the User Controller I2C calling the Target's unique addresses. The Secondary User address is yyyxxxxx10, where y and x are user-programmable from IP Catalog.

Figure 4.3 shows the block diagram of the Secondary I2C core.

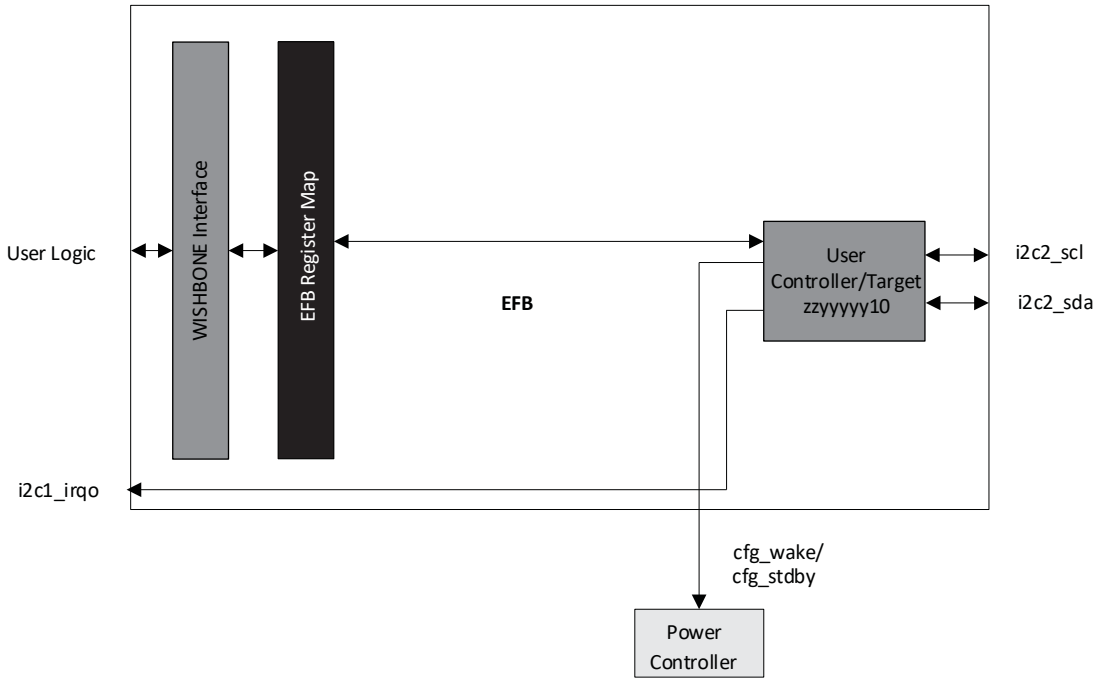


Figure 4.3. I2C Secondary Block Diagram

Table 4.3 documents the IP signals of the Secondary I2C cores. These signals can be routed to any GPIO of the MachXO4 devices.

Table 4.3. I2C Secondary – IP Signals

Signal Name	Pre-assigned Pin Name	I/O	Width	Description
i2c2_scl	—	Bidirectional	1	Open drain clock line of the I2C core. The signal is an output if the I2C core is performing a Controller operation. The signal is an input for Target operations. The signal can be routed to any GPIO of the MachXO4 device.
i2c2_sda	—	Bidirectional	1	Open drain data line of the I2C core. The signal is an output when data is transmitted from the I2C core. The signal is an input when data is received into the I2C core. The signal can be routed to any GPIO of the MachXO4 device.
i2c2_irqo	—	Output	1	Interrupt request output signal of the I2C core. This signal is intended to be connected to a WISHBONE controller (a microcontroller or state machine) and to request an

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Signal Name	Pre-assigned Pin Name	I/O	Width	Description
				interrupt when a specific condition is met. These conditions are described with the I2C register definitions.
cfg_wake	—	Output	1	Wake-up signal – Hardwired signal to be connected ONLY to the Power Controller of the MachXO4 device for functional simulation support. The signal is enabled only if the Wakeup Enable feature has been set within the EFB user interface, I2C Tab.
cfg_stdby	—	Output	1	Standby signal – Hardwired signal to be connected ONLY to the Power Controller of the MachXO4 device for functional simulation support. The signal is enabled only if the Wakeup Enable feature has been set within the EFB user interface, I2C Tab.

4.3. Configuring I2C Cores with IP Catalog

You can configure the I2C cores and generate the EFB module with IP Catalog. Selecting the I2C tab in the EFB user interface displays the configurable settings of the I2C cores. Figure 4.4 shows an example where the I2C cores are configured for an example design.

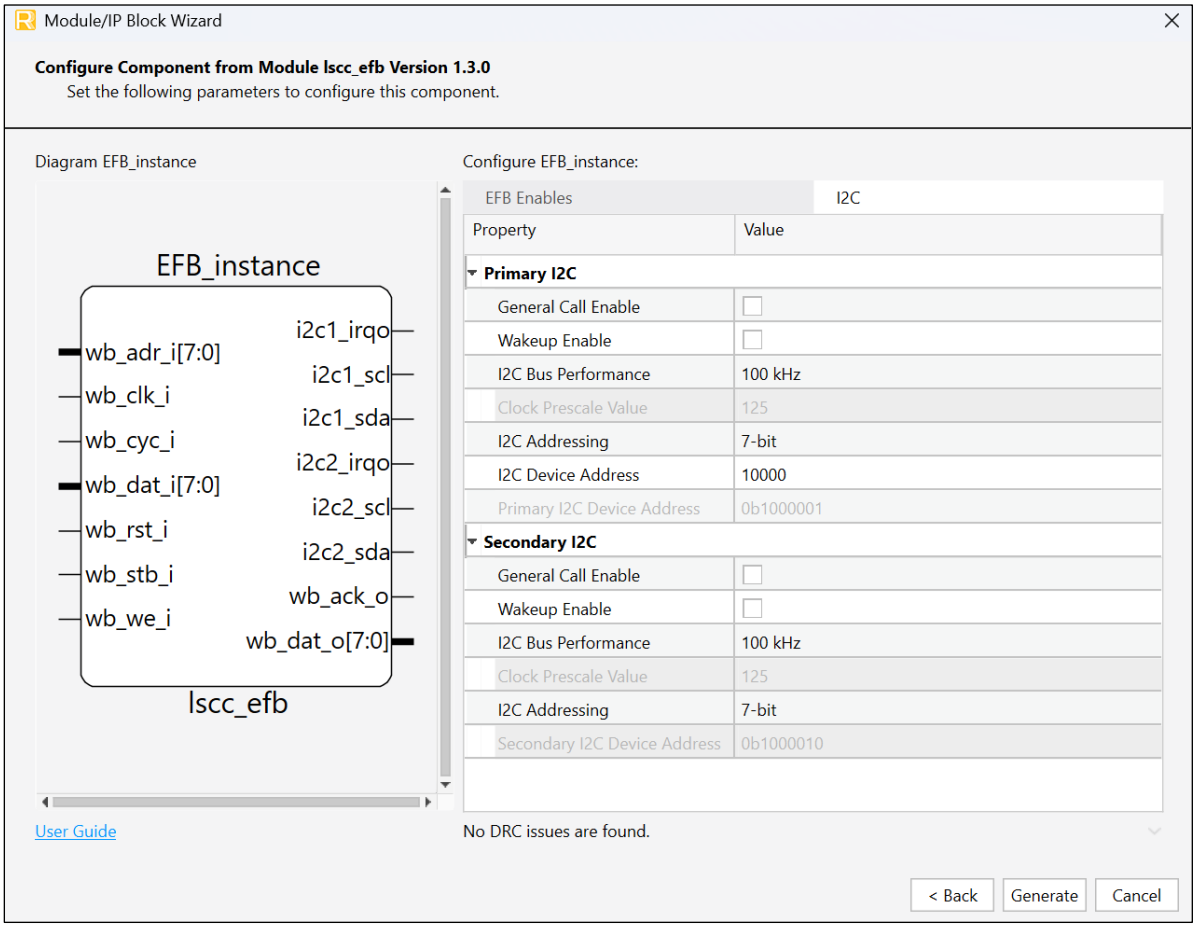


Figure 4.4. Configuring I2C Functions of the EFB Module with IP Catalog

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

4.3.1. General Call Enable

This setting enables the I2C General Call response (addresses all devices on the bus using the I2C address 0) in Target mode. This setting can be modified dynamically by enabling the GCEN bit in the Primary register I2C_1_CR or the Secondary register I2C_2_CR.

4.3.2. Wake-up Enable

For more information on the Power Controller, refer to the Power and Thermal Estimation and Management for MachXO4 Devices (FPGA-TN-02409).

When the Wake-up Enable is selected, an external I2C Controller can cause the MachXO4 device to leave the Standby Power state. There are two methods an external I2C controller can use to wake the MachXO4 device:

- Primary or Secondary Target I2C EFB address match
- Perform a General Call followed by the 0xF3 hex command opcode

The WKUPEN bit in the I2C_1_CR or the I2C_2_CR can be modified dynamically allowing the Wake Up function to be enabled or disabled.

4.3.3. I2C Bus Performance

You can select an I2C frequency of 50 kHz, 100 kHz, or 400 kHz. This is the frequency of the SCL clock on the I2C bus. This user interface value, together with the WISHBONE Clock Frequency attribute from the EFB Enables tab, allows the software to calculate the clock divider value for the 10-bit pre-scale registers using the equation (WB Clock)/(Clock Pre-scale Value). This pre-scale value is modified dynamically by accessing the Primary I2C Baud Rate register pair I2C_1_BR1, I2C_1_BR0 or the Secondary I2C Baud Rate register pair I2C_2_BR1, I2C_2_BR0.

4.3.4. I2C Addressing

You can select between a 7-bit or 10-bit I2C Target addressing scheme. The last two bits of the 7-bit address and 10-bit address are hard-coded and select one of the I2C components. The programmable bits of the I2C address are shared between I2C modules and defined as:

```
yyyxxxxxww
ww bits are hard coded with the following definition
00 = Primary Configuration I2C
01 = Primary User I2C
10 = Secondary User I2C
11 = I2C core reset
```

xxxxx bits are programmable using the IP Catalog user interface and have the default value of 10000.

yyy bits are programmable when 10-bit addressing is selected and have the default value of 000.

The Primary I2C address is the same length (seven or ten bits) as the Primary Configuration I2C address. The Primary and Secondary I2C address sizes can be of differing lengths. For example, the Primary I2C address could be ten bits and the Secondary I2C address could be seven bits.

4.3.5. MachXO4 I2C Usage Cases

The I2C usage cases described below refer to Figure 4.5.

- The Controller MachXO4 I2C Accessing Target External I2C Devices
  - A WISHBONE bus Controller is implemented in the MachXO4 logic.
  - I2C devices 1, 2, and 3 are all Target devices.
  - The WISHBONE bus Controller performs bus transactions to the Primary I2C controller in the EFB to access external Target I2C Device 1 on Bus A.
  - The WISHBONE bus Controller performs bus transactions to the Secondary I2C controller in the EFB to access the external Target I2C Devices number 2 or 3 on Bus B.
  - For more information on the I2C register definitions and command sequences, refer to the I2C section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

- External Controller I2C Device Accessing Target MachXO4 I2C
  - The I2C devices 1, 2, and 3 are I2C Controller devices.
  - The external controller I2C Device 1 on Bus A performs I2C memory cycles to access the EFB Primary I2C controller using address yyyxxxxx01.
  - The external controller I2C Device 2 or 3 on Bus B performs I2C memory cycles to access the EFB Secondary I2C User with the address yyyxxxxx10.
  - A WISHBONE bus controller in the MachXO4 fabric must manage data reception and transmission. The WISHBONE controller can use interrupts or polling techniques to manage data transfer and to prevent data overrun conditions.
  - For more information on the I2C register definitions and command sequences, refer to the I2C section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
- External Controller I2C Device Accessing the MachXO4 Flash Using the Primary I2C Interface
  - The external Controller I2C Device 1 on Bus A performs bus transactions using address yyyxxxxx00. The external controller interacts with the MachXO4 Configuration Logic using this address. The Configuration Logic provides the controls necessary for performing Flash operations.
  - More details on accessing the Flash of the MachXO4 device through I2C are found later in this document and in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
  - For more information on programming the MachXO4 through the I2C port, refer to the I2C section in the Power and Thermal Estimation and Management for MachXO4 Devices (FPGA-TN-02409).
- The above usage cases are not mutually exclusive. For example:
  - External Controller Device 1 on Bus A can access the MachXO4 Configuration Logic at the same time a WISHBONE Controller transfers data to the I2C target devices on Bus B.
  - A WISHBONE controller can transfer data to a microprocessor on Bus A (that is by I2C Device 1), and at some future time the microprocessor can send data back to the WISHBONE Controller.

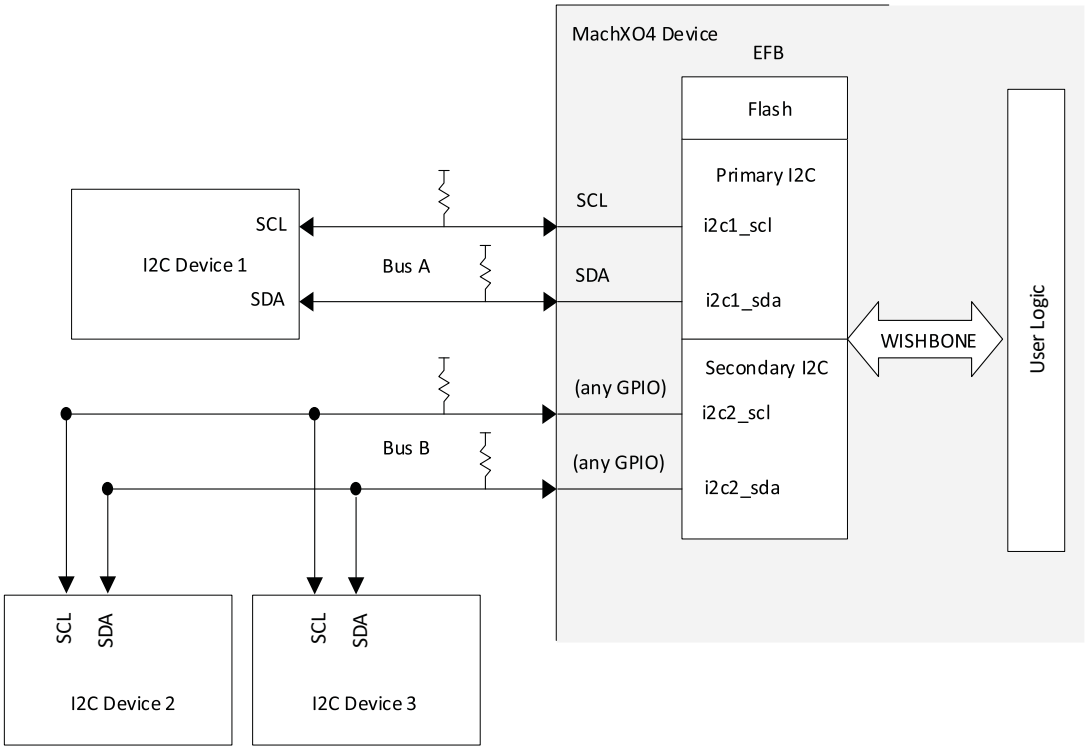


Figure 4.5. I2C Circuit

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

4.3.6. I2C Design Tips

- For more information on the I2C register definitions and command sequences, refer to the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
- Take note that when dynamically turning off components for power savings, the EFB requires the MachXO4 internal oscillator to be enabled even if it is not the source of the WISHBONE Clock. The only exception is after SDA delay is turned off by setting I2C_1_CR[3:2]=11, the OSC can be turned off. Following this, the OSC can be disabled and User Target or User Controller can operate.
- I2C has lower priority than JTAG Port and the Target SPI Port when accessing the NV. Refer to the Flash Access section for details.
- The Primary I2C port cannot be used for both Flash access and user functions in the same design.
- If the Secondary I2C Port is enabled after issuing a Refresh command or toggling PROGRAMN, it is recommended to reset the state machine with an I2C STOP. I2C STOP is performed with a single register write 0x40 to I2C_2_CMDR. This causes a short low-pulse on SCK as the block signals the STOP. Normal I2C activity can commence without additional delay.
- Ensure the correct I2C command for Read UFM (0xCA) is used (for example, 0xCA 00 00 01).
- Ensure the correct I2C command for Read Configuration Flash (0x73) is used (for example, 0x73 00 00 01).
- The MachXO4 input buffer is a generic design capable of handling signals up to 400 MHz. Due to its high-speed performance, slower I2C signals can become more susceptible to noise. This can be compensated by:
  - Using stronger external pull-ups (for example, 2 kΩ)
  - Enabling hysteresis
  - Using a glitch filter as described in the Improving Noise Immunity Serial Interfaces and MachXO4 Hardware Checklist (FPGA-TN-02411) documents.
  - Set **PULLMODE** to *KEEPER* and **HYSTERESIS** to *LARGE* for the I2C pins on the **Port** tab of **Device Constraint Editor**. To open **Device Constraint Editor**, select **Tools** > D**evice Constraint Editor** from the Lattice Radiant software menu bar.
- For more information on Lattice reference designs, refer to the Lattice Solutions Reference Designs web page.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5. Hardened SPI IP Core

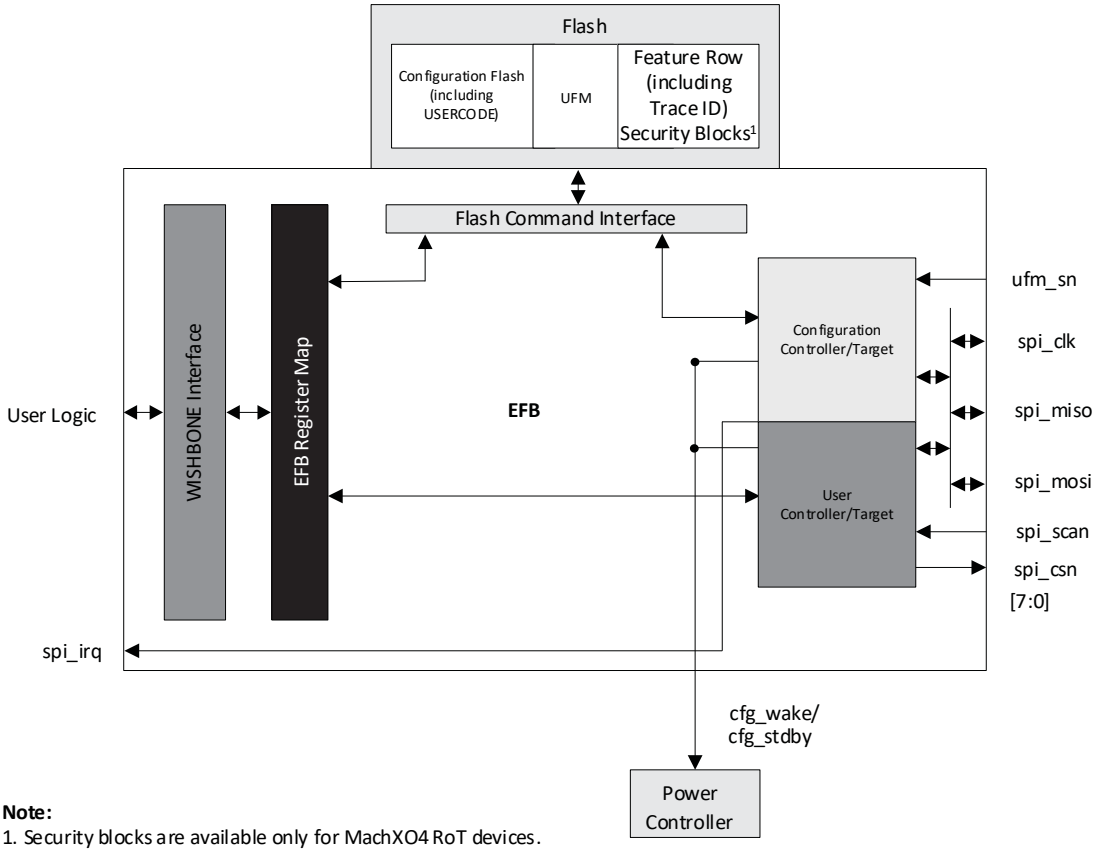
SPI is a widely used four-wire serial bus which operates in full duplex for communication between devices. The MachXO4 EFB contains an SPI Controller that can be configured as an SPI Controller/Target or an SPI Target. When the IP core is configured as a Controller/Target, it is able to control up to eight other devices with Target SPI interfaces. When the core is configured as a Target, it is able to interface to an external SPI Controller device. The SPI core interfaces with the MachXO4 device Configuration Logic or the other User Logic. The hardened SPI IP core functionality and block diagram are shown below.

Table 5.1. Hardened SPI Functionality

	Configuration SPI	User SPI
Target SPI Port	Yes	Yes
Controller SPI Port	No[1]	Yes
Access the Configuration Flash	Yes	No
Must use dedicated I/O	Yes	Yes[2]
Wake Power Controller from Standby Mode	Yes	Yes
Enter Power Controller Standby Mode	No	Yes

Notes:

1. Only for the configuring SRAM.
2. Any GPIO can be used for spi_csn[7:1] and spi_scsn.



Note:
1. Security blocks are available only for MachXO4 RoT devices.

Figure 5.1. SPI Block Diagram

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The SPI IP core on MachXO4 devices supports the following functions:

- Configurable Controller and Target modes
- Mode Fault error flag with CPU interrupt capability
- Double-buffered data register for increased throughput
- Serial clock with programmable polarity and phase
- LSB First or MSB First Data Transfer
- Interface to custom logic through the EFB WISHBONE target interface

In Controller/Target SPI Mode

- The User SPI controller has eight available Controller Chip Selects (spi_csn[7:0]) ports. This allows the control of up to eight external devices with Target SPI interface.
- The Configuration SPI upon power-up, if the SPI port has been enabled to boot the MachXO4 device from an external Target SPI Flash memory, then the SPI port acts as a Controller SPI controller and spi_csn[0] is used as a Controller Chip Select for selecting a specific SPI Flash memory. For more information on programming the MachXO4 device through the SPI port, refer to the SPI section in the MachXO4 Programming and Configuration User Guide (FPGA-TN-02393).

In Target SPI Mode

- The User SPI core has one Target Chip Select (spi_scsn) pin. This allows the User SPI core to be selected by an external device with a Controller SPI interface. User Logic is accessed through the EFB WISHBONE interface by a WISHBONE Controller in the FPGA logic.
- The Configuration SPI has one Target Chip Select (sn) pin. An external SPI Controller can access the MachXO4 device Configuration Logic by asserting the chip select input. The external SPI Controller can reprogram the MachXO4 device Flash by performing bus transfers with SN asserted.

This user guide is focused on the User SPI access. For more information on programming the MachXO4 device through the SPI port, refer to the SPI section in the MachXO4 Programming and Configuration User Guide (FPGA-TN-02393).

The Target Configuration SPI port can be used to wake the Power Controller from Standby or enter Standby. The Target User SPI port can only be used to wake the Power Controller from Standby mode. For more information on the Power Controller, refer to the Power and Thermal Estimation and Management for MachXO4 Devices (FPGA-TN-02409). The SPI Power Controller features can be set up through IP Catalog as documented later, and the register settings are defined in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

5.1. SPI Interface Signals

The SPI interface uses a serial transmission protocol. Data is transmitted serially (shifted out from the transmitting device) and it is received serially, shifted into the receiving device. The controller device selects a specific target device by asserting a chip select, enabling the target device to shift in the commands/data and to respond by shifting out data.

Table 5.2 documents the signals that are associated with the IP core. Each signal has a description of its usage and connection within a design project.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Table 5.2. SPI – IP Signals

Signal Name	Pre-assigned Pin Name	I/O	Width	Description
spi_clk	MCLK/CCLK	Bidirectional	1	The signal is an output if the SPI core is in Controller mode (MCLK). The signal is an input if the SPI core is in Target mode (CCLK). This signal must be brought to the top level of the user RTL design. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of SPI signals in each MachXO4 device.
spi_miso	SPISO/SO	Bidirectional	1	The signal is an input if the SPI core is in Controller mode (SPISO). The signal is an output if the SPI core is in Target mode (SO). This signal must be brought to the top level of the user RTL design. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of SPI signals in each MachXO4 device.
spi_mosi	SISPI/SI	Bidirectional	1	The signal is an output if the SPI core is in Controller mode (SISPI). The signal is an input if the SPI core is in Target mode (SI). This signal must be brought to the top level of the user RTL design. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of SPI signals in each MachXO4 device.
spi_csn[7:0]	CSSPIN	Output	8	Controller Chip Select (Active Low). Up to eight independent target SPI devices can be accessed using the MachXO4 SPI Controller when it is in Controller SPI mode. The signal spi_csn[0] must be brought to the top level of the user RTL design. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of SPI signals in each MachXO4 device.
spi_scsn	—	Input	1	User Target Chip Select (Active Low). An external SPI Controller asserts this signal to transfer data to/from the SPI Controller Transmit Data/Receive Data registers. The signal can be routed to any GPIO of the MachXO4 device.
sn	SN	Input	1	Configuration Logic Chip select (Active Low) is dedicated for selecting the Flash Sectors. The Radiant software automatically routes this signal to its Pre-Assigned pin (no user pin location constraint is necessary). Refer to pin tables in the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for detailed pad and pin locations of SPI signals in each MachXO4 device. SN is an active pin whenever the SPI core is instantiated. SN does not appear on the EFB primitive. Thus, SN cannot be recovered as user I/O. SN can be tied high externally to augment the weak internal pull-up if not connected to an external Controller SPI bus. SN is also active in a blank or erased device.
spi_irq	—	Output	1	Interrupt request output signal of the SPI core. This signal is intended to be connected to a WISHBONE controller (that is by a microcontroller or state machine). It is asserted when specific conditions are met. These conditions are controlled using the SPI register settings.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Signal Name	Pre-assigned Pin Name	I/O	Width	Description
cfg_wake	—	Output	1	Wakeup signal – to be connected only to the Power Controller module of the MachXO4 device. The signal is enabled only if the Wakeup Enable feature has been set within the EFB user interface, SPI Tab.
cfg_stdby	—	Output	1	Standby signal – to be connected only to the Power Controller module of the MachXO4 device. The signal is enabled only if the Wakeup Enable feature has been set within the EFB user interface, SPI Tab.

5.2. Configuring the SPI Core with IP Catalog

IP Catalog is used to configure the SPI Controller and to generate Verilog or VHDL source code for inclusion in your design. Selecting the SPI tab, in the EFB user interface, displays the configurable settings for the SPI core. Figure 5.2 shows an SPI Controller configuration example.

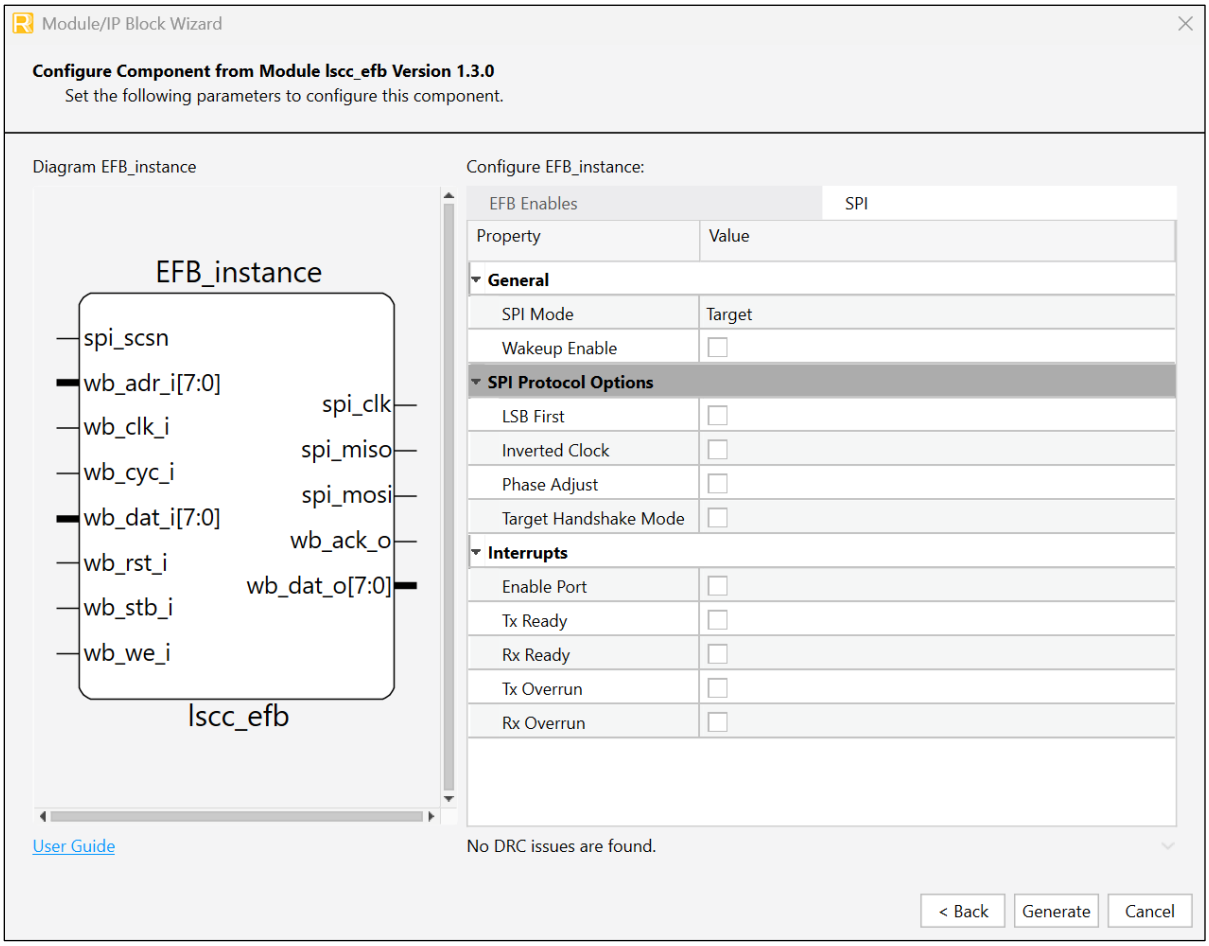


Figure 5.2. Configuring SPI Functions of the EFB Module with IP Catalog

5.2.1. SPI Mode

This option allows you to select between Target or Target and Controller modes for the initial mode of the SPI core. Selecting Target and Controller enables SPI Controller settings, which include Controller Clock Rate and Controller Chip Selects. This option can be updated dynamically by modifying the MSTR bit of the register SPICR2.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5.2.2. SPI Controller Clock Rate

Desired Frequency:

The EFB SPI Controller, when it is configured as an SPI Controller, provides an output clock to the SPI Target devices on the bus. The output frequency uses a power of two value to divide the WISHBONE Clock Frequency. The SPI Controller uses the Controller Clock Rate to time all SPI bus transactions and internal operations. The MachXO4 SPI Controller interface can operate at speeds up to 45 MHz. You input the WISHBONE Clock Frequency on the EFB Enables tab of the dialog.

The divisor can be changed while the FPGA is in user mode. Updating the divider value in the SPIBR register causes the SPI Controller to reset and use a new output clock frequency.

Actual Frequency:

It is not always possible to divide the input WISHBONE clock exactly to the requested frequency. The actual frequency value is returned in this read-only field. When both the desired SPI clock and WISHBONE clock fields have valid data and either is updated, this field returns the value rounded to two decimal places.

For more information regarding the SPI clock rate and SPIBR register control, refer to the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

5.2.3. SPI Protocol Options

LSB First:

This setting specifies the order of the serial shift of a byte of data. The data order (MSB or LSB first) is programmable within the SPI core. This option can be updated dynamically by modifying the LSBF bit in the register SPICR2.

Inverted Clock:

The inverts the clock polarity used to sample and output data is programmable for the SPI core. When selected, the edge changes from the rising to the falling clock edge. This option can be updated dynamically by accessing the CPOL bit of register SPICR2.

Phase Adjust:

An alternate clock-data relationship is available for SPI devices with particular requirements. This option allows you to specify a phase change to match the application. This option can be updated dynamically by accessing the CPHA bit in the register SPICR2.

Target Handshake Mode:

Enables a Lattice proprietary extension to the SPI protocol. This option is used when the internal support circuit (such as WISHBONE host) cannot respond with initial data within the time required, and to make the Target read out data predictably available at high SPI clock rates. This option can be updated dynamically by accessing the SDBRE bit in the register SPICR2.

5.2.4. Controller Chip Selects

The SPI Controller can provide up to eight individual chip select outputs for controller operation.

Each target SPI device accessed by the controller must have its own dedicated chip select. This option can be updated dynamically by modifying the register SPICSR.

5.2.5. SPI Controller Interrupts

TX Ready:

An interrupt which indicates the SPI transmit data register (SPITXDR) is empty. The interrupt bit is

IRQTRDY of the register SPIIRQ. When enabled, indicates TRDY is asserted. Write a *1* to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQTRDYEN in the register SPICSR.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

RX Ready:

An interrupt which indicates the receive data register (SPIRXDR) contains valid receive data. The interrupt is bit IRQRRDY of the register SPIIRQ. When enabled, indicates RRDY is asserted. Write a *1* to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQRRDYEN in the register SPICSR.

TX Overrun:

An interrupt which indicates the Target SPI chip select (SPI_SCSN) was driven low while a SPI Controller.

The interrupt is bit IRQMDF of the register SPIIRQ. When enabled, indicates MDF (Mode Fault) is asserted.

Write a *1* to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQMDFEN in the register SPICSR.

RX Overrun:

An interrupt which indicates SPIRXDR received new data before the previous data. The interrupt is bit

IRQROE of the register SPIIRQ. When enabled, indicates ROE is asserted. Write a *1* to this bit to clear the interrupt. This option can be changed dynamically by modifying the bit IRQROEEN in the register SPICSR.

Enable Port (Interrupts):

This enables the interrupt request output signal (spi_irq_ of the SPI core signal. This signal is intended to be connected to a WISHBONE controller (that is by a microcontroller or state machine) and to request an interrupt when a specific condition is met.

5.2.6. Wake-up Enable

Enables the SPI core to send a wake-up signal to the Power Controller to wake the part from standby mode when the User Target SPI chip select (spi_csn[0]) is driven low. This option can be updated dynamically by modifying the bit WKUPEN_USER in the register SPICR1.

5.2.7. MachXO4 SPI Usage Cases

The SPI usage cases described below refer to the figures below:

- External Controller SPI Device Accessing the Target MachXO4 User SPI:
  - The External Controller SPI is connected to the MachXO4 device using the dedicated SI, SO, CCLK pins. The spi_scsn is placed on any Generic I/O. The EFB SPI Mode is set to Target only.
  - A WISHBONE Controller is implemented in the MachXO4 general purpose logic array. The controller monitors the availability to transmit or receive data by polling the SPI status registers, or by responding to interrupts generated by the SPI Controller. For more information on the SPI register definitions and command sequences refer to the SPI section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
  - The external SPI Controller does not have access to the MachXO4 Configuration Logic because the SN that selects the Configuration Logic is pulled high.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

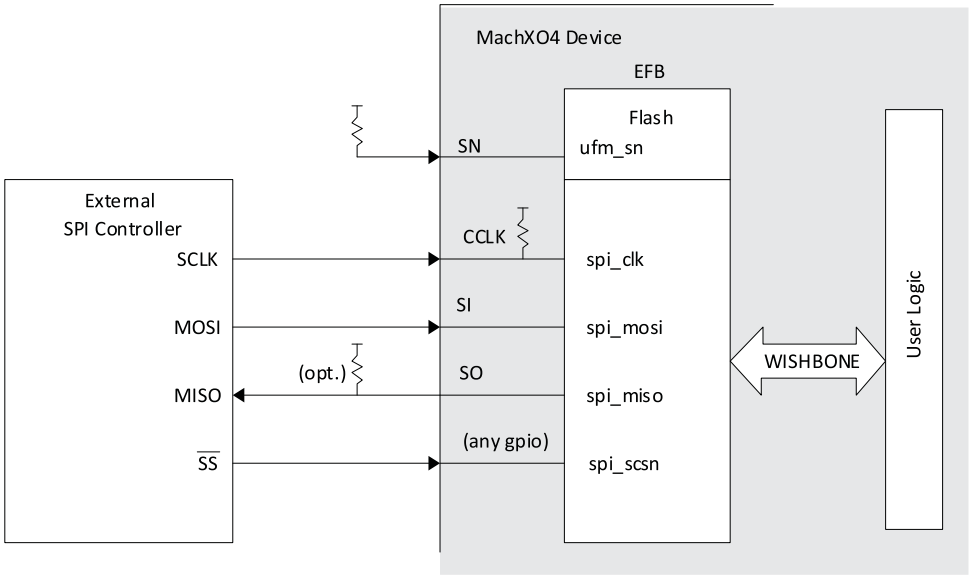


Figure 5.3. External Controller SPI Device Accessing the Target MachXO4 User SPI

- The MachXO4 User SPI Controller accessing one or multiple External Target SPI devices:
  - The MachXO4 SPI Controller is connected to External SPI Target devices using the dedicated SPI port pins. The Chip Selects are configured as follows:
    - The MachXO4 SPI Controller Chip Select spi_csn[0] is placed on the dedicated CSSPIN and connected to the External Target Chip Select.
    - The MachXO4 SPI Controller Chip Select spi_csn[1] is placed on any I/O and connected to another External Target Chip Select.
    - Up to eight External Target SPIs can be connected using spi_csn[7:0]
  - A WISHBONE Controller is implemented in the MachXO4 general logic. It controls transfers to the target SPI devices. It can use a polling method, or it can use SPI Controller interrupts to manage transfer and reception of data.

For more information on the SPI register definitions and command sequences, refer to the SPI section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

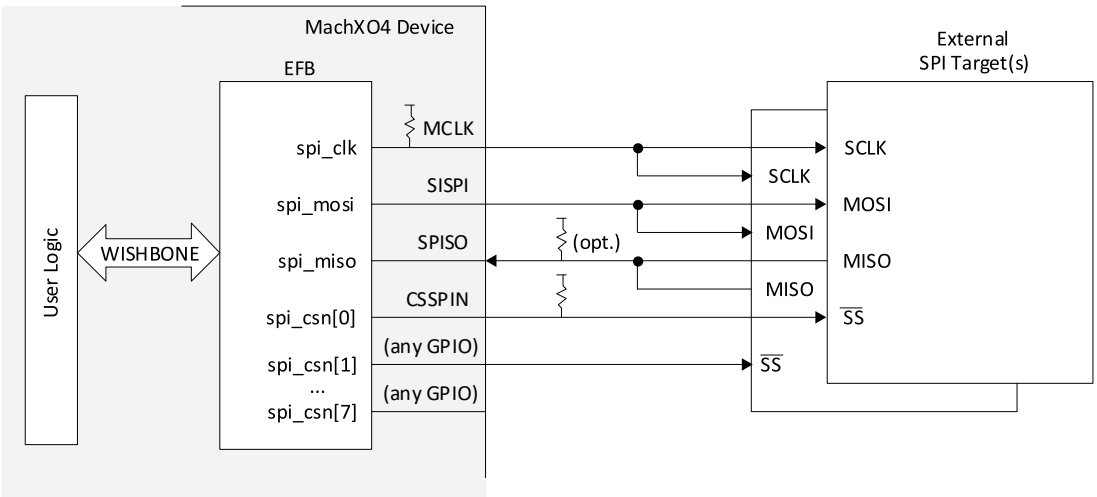


Figure 5.4. MachXO4 User SPI Controller Accessing One or Multiple External Target SPI Devices

- External Controller SPI Device accessing the MachXO4 Configuration Logic

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

- The External SPI Controller is connected to the MachXO4 dedicated target Configuration SPI port pins. The external SPI Controller's chip select controls the SN input that enables the MachXO4 Configuration Logic block. The external controller sends commands to the Configuration Logic block permitting it to interface to the Flash.
- The information on accessing the Flash of the MachXO4 device through SPI can be found later in this document.
- For more information on programming the MachXO4 device through the SPI port, refer to the SPI section in the MachXO4 Programming and Configuration User Guide (FPGA-TN-02393).

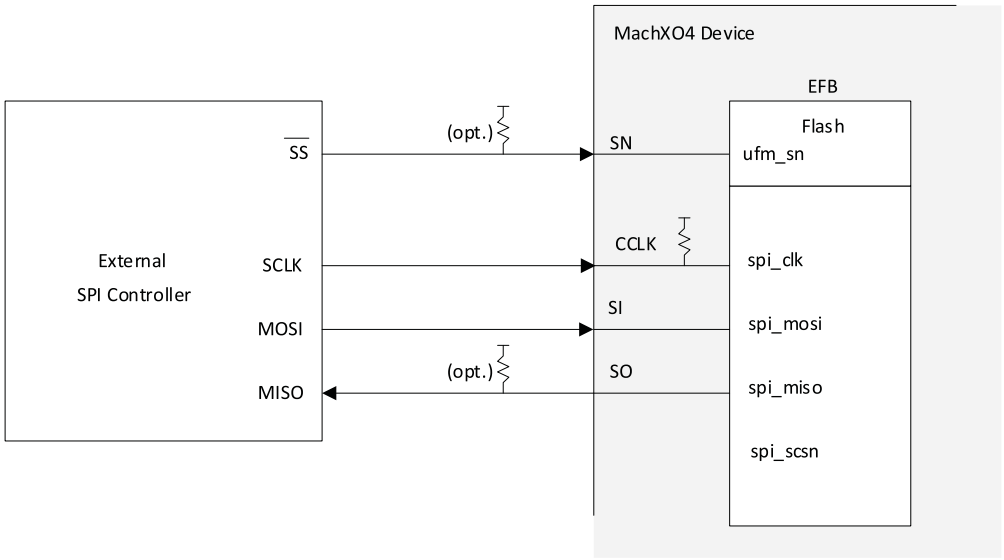


Figure 5.5. External Controller SPI Device Accessing the MachXO4 Configuration Logic

5.2.8. SPI Design Tips

- For more information on the SPI register definitions and command sequences, refer to the SPI section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
- Take note when dynamically turning off components for power savings; the EFB requires the MachXO4 internal oscillator to be enabled even if it is not the source of the WISHBONE Clock. The exception is with User Target or User Controller modes.
- The SPI bus is bidirectional. A byte is received for every byte transmitted. Always discard RX data to avoid Receiver Overrun Error (ROE).
- SN is an active pin whenever the SPI core is instantiated, whether *ufm_sn* appears on the EFB primitive or not. Thus, SN cannot be recovered as user I/O. Tie the SN pin high externally to augment the weak internal pull-up if not connected to an external Controller SPI bus.
- For more information on Lattice reference designs, refer to the Lattice Solutions Reference Designs web page.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6. Timer/Counter

The MachXO4 EFB contains a Timer/Counter function. This Timer/Counter is a general purpose 16-bit Timer/Up Down Counter module with independent output compare units and Pulse Width Modulation (PWM) support. The Timer/Counter supports the following functions:

- Four unique modes of operation:
  - Watchdog timer
  - Clear Timer on Compare Match
  - Fast PWM
  - Phase and Frequency Correct PWM
- Programmable clock input source
- Programmable input clock pre-scale
- Interrupt output to FPGA fabric
- Three independent interrupt sources: overflow, output compare match, and input capture
- Auto reload
- Time-stamping support on the input capture unit
- Waveform generation on the output
- Glitch-free PWM waveform generation with variable PWM period
- Internal WISHBONE bus access to the control and status registers
- Standalone mode with preloaded control registers and direct reset input

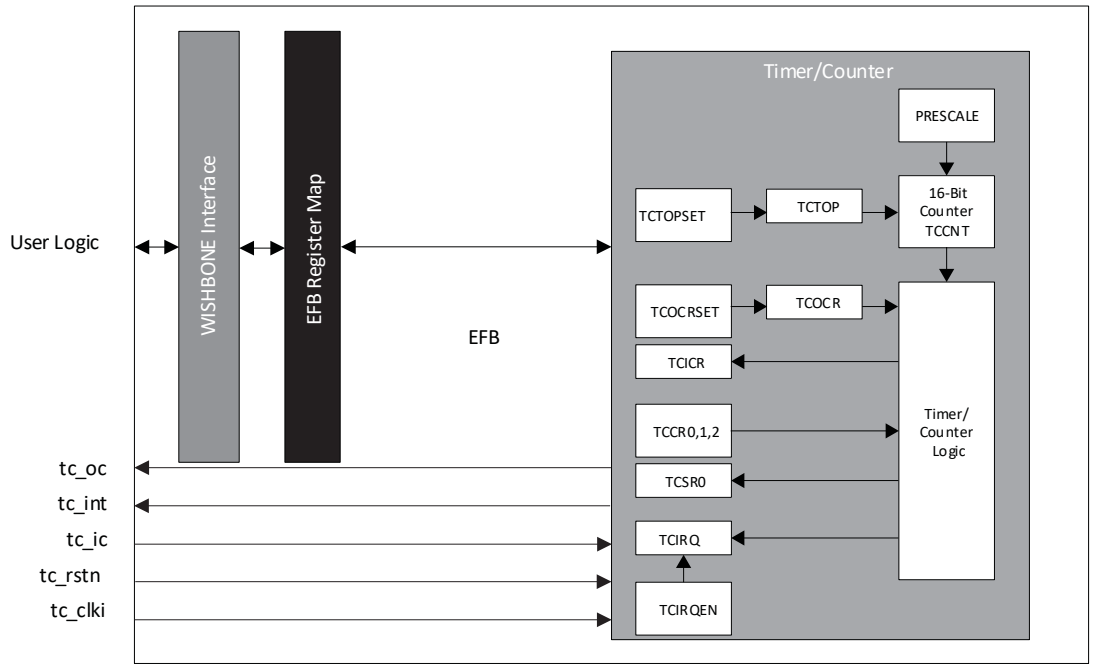


Figure 6.1. Timer/Counter Block Diagram

The Timer/Counter communicates with the FPGA core logic through the WISHBONE interface and specific signals such as clock, reset, interrupt, timer output, and event trigger. The Timer/Counter can be included in a design with or without the WISHBONE interface.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6.1. Timer/Counter Modes of Operation

The Timer/Counter is a flexible function, which has four modes of operation. These modes are designed to meet different system needs related to sequencing events and PWM (Pulse Width Modulation). The four Timer/Counter modes are:

- Clear Timer on Compare Match
- Watchdog Timer
- Fast PWM
- Phase and Frequency Correct PWM

6.1.1. Clear Timer on Compare Match Mode

CTCM (Clear Timer on Compare Match) is a basic counter with interrupts. The counter is automatically cleared to 0x0000 when the counter value, TCCNT register, matches the value loaded in the TCTOP register. The value of the TCTOP register can be dynamically updated through the WISHBONE register, or it can hold a static value that is assigned with IP Catalog at the time of IP generation. The default value of the TCTOP register is 0xFFFF.

The data loaded into the timer counter to define the top counter value is double-registered. The WISHBONE host writes the data to TCTOPSET register, which is then automatically loaded onto the TCTOP register at the moment of auto-clear. Therefore, a new top value can be written to the TCTOPSET register after the overflow flag and during the counting-up to the top value. Updating the value of the TCTOP register changes the frequency of the output signal of the Timer/Counter.

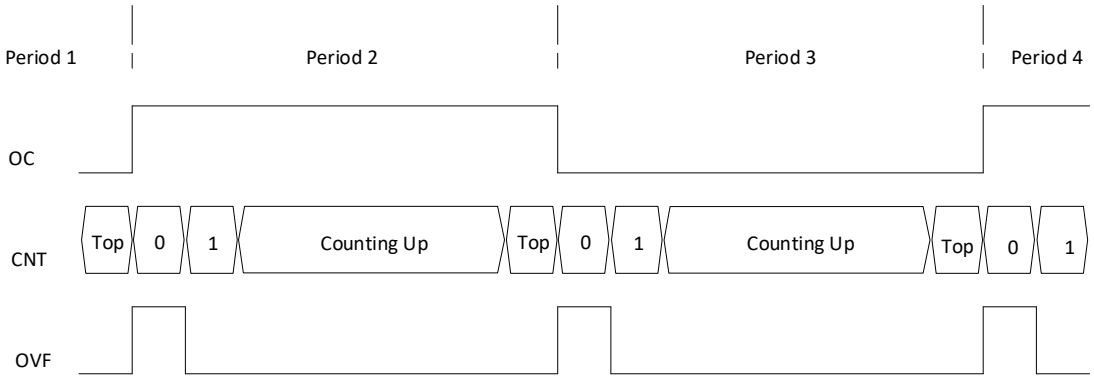


Figure 6.2. Timer/Counter Output Waveform

6.1.2. Watchdog Timer Mode

Watchdog timers are used to monitor a system's operating behavior and provide a reset or interrupt when the system's microcontroller or embedded state machine is no longer operational. One scenario is for a microcontroller to reset the Watchdog Timer to 0x0000 before it begins a process. The microcontroller must complete the process and reset the Watchdog Timer before the timer reaches its terminal count. In the event that the microprocessor does not clear the timer quickly enough, the Watchdog Timer asserts a strobe signaling time expired. The system uses the *time exceeded* strobe to gracefully recover the system.

Another way to use the Watchdog Timer is to periodically turn OFF system modules in order to save power. It can also be used to interact with the on-chip power controller of the MachXO4 device.

The most commonly used ports of the Timer/Counter in Watchdog Timer Mode are the clock, reset, and interrupt.

Optionally, the WISHBONE interface can be used to read time stamps from the TCICR register and update the top value of the counter.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6.1.3. Fast PWM Mode

Pulse-Width Modulation (PWM) is a popular technique to digitally control analog circuits. PWM uses a rectangular pulse wave whose pulse width is modulated, resulting in the variation of the average value of the waveform. You can vary the period and the duty cycle of the waveform by loading 16-bit digital values on the TCTOP register to define the top value of the counter, and the TCOCR register to provide a compare value for the output of the counter.

The output of the Timer/Counter is cleared when the counter value matches the top value that is loaded in the TCTOP register. The output is set when the value of the counter matches the compare value that is loaded into the TCOCR register. The clear/set functions can be inverted. This means that the output of the Timer/Counter is set when the counter value matches the top value and it is cleared when the value of the counter matches the compare value.

The interrupt line can be used for Overflow Flag (OVF) and Output Compare Flag (OCRF).

Figure 6.3 shows the PWM waveform generation. The output of the Timer/Counter is configured to be set when the counter matches the top value and clear when the counter matches the compare value.

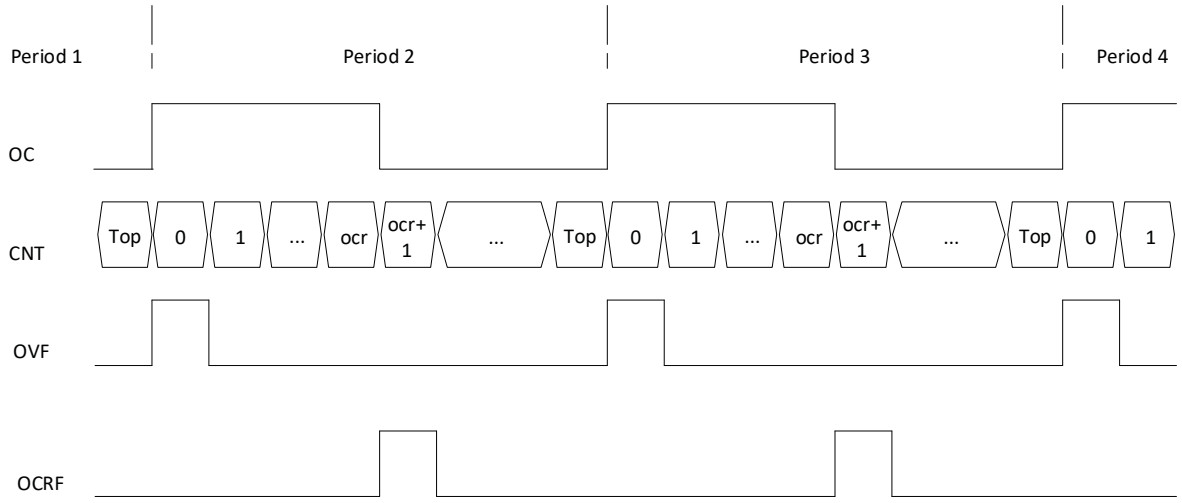


Figure 6.3. PWM Waveform Generation

6.1.4. Phase and Frequency Correct PWM Mode

In phase and frequency correct PWM mode, the counting direction changes from up to down at the moment the counter is incrementing to the top value (top value minus 1). The moment that the counter is decrementing from 0x0001 to 0x0000, the following occurs:

- TCTOP is updated with the value loaded in the TCTOPSET register
- TCOCR is updated with the value loaded in the TCOCRSET register
- Overflow TCSR[OVF] is asserted for one clock cycle

The output of the Timer/Counter is updated only when the counter value matches the compare value in TCOCR register. This occurs twice within one period. The first match occurs when the counter is counting up and the second match occurs when the counter is counting down. The Output Compare Flag TCSR[OCRF] is asserted when both matches occur. The output of the Timer/Counter is set on the first compare match and cleared on the second compare match. The order of set and clear can be inverted.

The mode allows you to adjust the frequency (based on the top value) and phase (based on the compare value) of the generated waveform.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

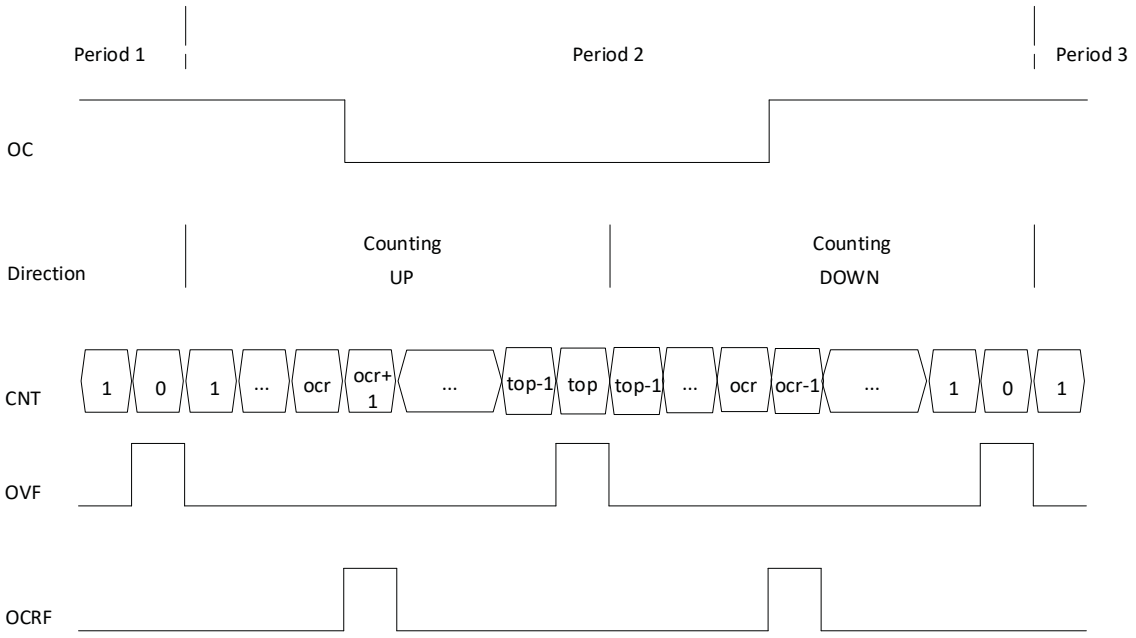


Figure 6.4. Phase and Frequency Correct PWM Waveform Generation

6.2. Timer/Counter IP Signals

Table 6.1 documents the signals that are generated with the IP. Each signal has a description of its usage and connection within a design project.

Table 6.1. Timer/Counter – IP Signals

Signal Name	I/O	Width	Description
tc_clki	Input	1	Timer/Counter input clock signal. Can be connected to the on-chip oscillator. The clock signal is limited to 133 MHz.
tc_rstn	Input	1	This is an active-low reset signal, which resets the 16-bit counter.
tc_ic	Input	1	This is an active-high input capture trigger event, applicable for non-PWM modes with WISHBONE interface. If enabled, a rising edge of this signal is detected and synchronized to capture the counter value (TCCNT Register) and make the value accessible to the WISHBONE interface by loading it into TCICR register. The common usage is to perform a time-stamp operation with the counter.
tc_int	Output	1	This is an interrupt signal, indicating the occurrence of a specific event such as Overflow, Output Compare Match, or Input Capture.
tc_oc	Output	1	Timer/Counter output signal

6.3. Configuring Timer/Counter

IP Catalog is used to configure the Timer/Counter. Selecting the Timer/Counter tab in the EFB user interface displays the configurable settings of the Timer/Counter core.

The Timer/Counter can be used with or without the WISHBONE interface. For usage without the WISHBONE interface, you can select/enter values in the IP Catalog EFB user interface. The values are programmed in the Timer/Counter registers with the programmable bitstream. Using the Timer/Counter with a WISHBONE interface enables you to dynamically update the register content through the WISHBONE interface. The main EFB user interface, EFB Enables tab, allows you to select between using the Timer/Counter with or without a WISHBONE interface.

Figure 6.5 shows an example of how the Timer/Counter is configured for a specific design.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

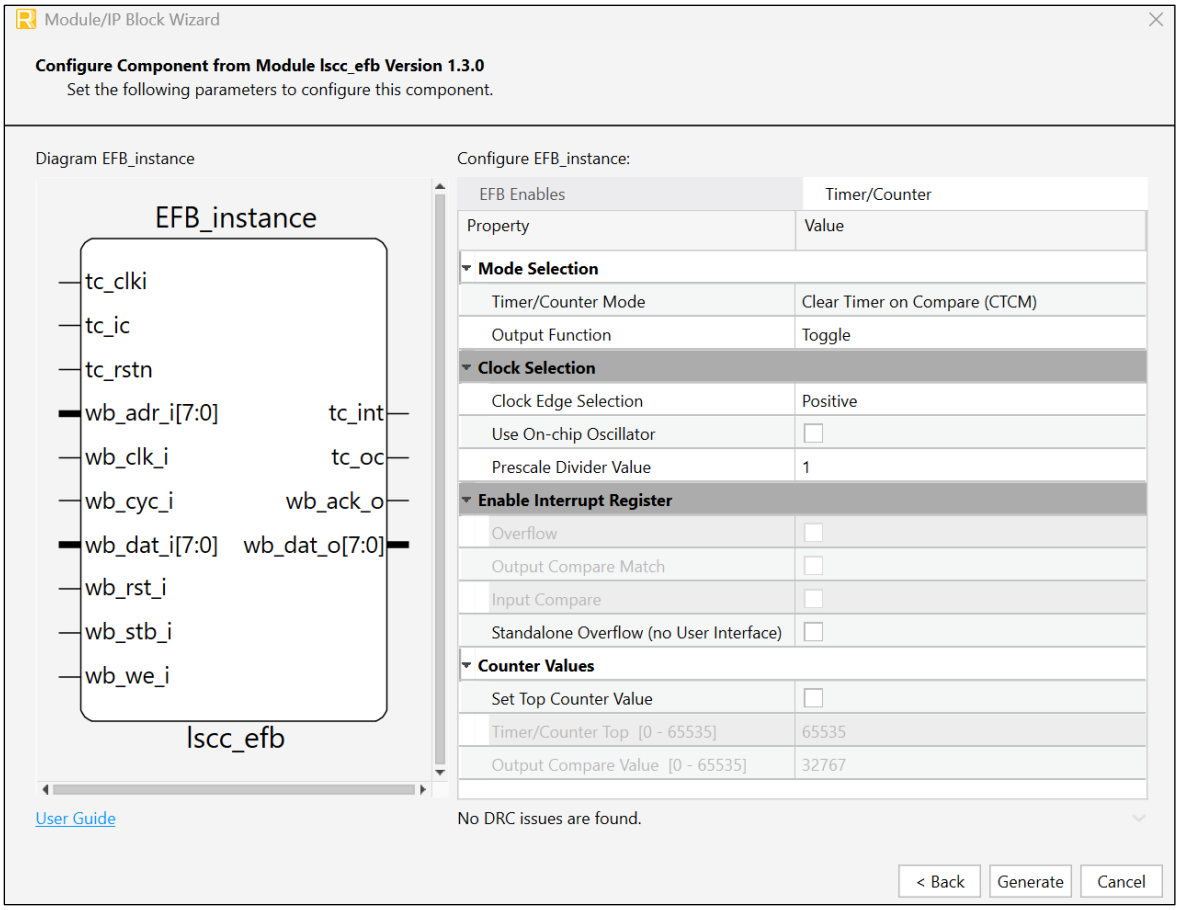


Figure 6.5. Configuring Timer/Counter

6.3.1. Timer/Counter Mode

This option allows you to select one of the four operating modes.

- CTCM – Clear Timer on Compare Match
- WATCHDOG – Watchdog timer
- FASTPWM – Fast PWM
- PFCPWM – Phase and Frequency Correct PWM

This option can be updated dynamically by modifying the bits TCM[1:0] of the register TCCR1.

6.3.2. Output Function

You can select the function of the output signal (tc_oc) of the Timer/Counter IP. The available functions are:

- STATIC – The output of the Timer/Counter is static low.
- TOGGLE – The output of the Timer/Counter toggles based on the conditions defined by the Timer/Counter Mode.
- WAV_GENERATE – The waveform is generated by the Set/Clear on the conditions defined by the Timer/Counter Mode.
- INV_WAV_GENERATE – The waveform is inverted.

This option can be updated dynamically by modifying bits OCM[1:0] of the register TCCR1.

6.3.3. Clock Edge Selection

You can select the edge (positive or negative) of the input clock source as well as enable the usage of the on-chip oscillator. The selections are:

- PCLOCK – Positive edge of the clock from user logic.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

- POSC – Positive edge of the clock from the internal oscillator.
- NCLOCK – Negative edge of the clock from user logic.
- NOSC – Negative edge of the clock from the internal oscillator.

This option can be updated dynamically by modifying the bits CLKSEL and CLKEDGE of the register TCCR0.

6.3.4. Pre-scale Divider Value

Pre-scale divider values (0, 1, 8, 64, 256, 1024) are provided to divide the input clock prior to reaching the 16-bit counter. This option can be updated dynamically by modifying the bits PRESCALE[2:0] of the register TCCR1.

6.3.5. Timer/Counter Top

You can select the top value of the counter. This option can be updated dynamically by modifying the bits TCTOPSET of the registers TCTOPSET1 and TCTOPSET0.

6.3.6. Output Compare Value

You can select the output compare value of the counter. This option can be updated dynamically by modifying the bits TCOCRSET of the registers TCOCRSET1 and TCOCRSET0.

6.3.7. Enable Interrupt Registers

- Overflow – An interrupt which indicates the counter matches the TCTOP0/1 register value. The interrupt is bit IRQOVF of the register TCIRQ. When enabled, indicates OVF was asserted. Write a *1* to this bit to clear the interrupt. This option can be updated dynamically by modifying the bit IRQOVFEN of the register TCIRQEN.
- Output Compare Match – An interrupt which indicates when counter matches the TCOCR0/1 register value. The interrupt is bit IRQOCRF of the register TCIRQ. When enabled, indicates OCRF was asserted. Write a *1* to this bit to clear the interrupt. This option can be updated dynamically by modifying the IRQOCRFEN bit of register TCIRQEN.
- Input Capture – An interrupt which indicates when you assert the TC_IC input signal. The interrupt is bit IRQICRF of the register TCIRQ. When enabled, indicates ICRF was asserted. Write a *1* to this bit to clear the interrupt. This option can be updated dynamically by modifying IRQICRFEN bit of the register TCIRQEN.
- Standalone Overflow – Used without the WISHBONE interface and serves as the only available interrupt request.

6.3.8. MachXO4 Timer/Counter Usage Cases

6.3.8.1. Basic Counter with Interrupts

Configure the Timer/Counter for Static or Dynamic operation. Dynamic operation enables the WISHBONE bus interface allowing a WISHBONE Controller to control the Timer/Counter.

To configure the Timer/Counter for Static or Dynamic operation in the Timer/Counter tab:

1. Configure the Timer/Counter Mode using the Mode Selection controls
   a. Select CTCM (Clear Timer on Compare Match) Mode
   b. Select the Output Function
      - Hold static 0 (STATIC)
      - Toggle (TOGGLE)
2. Update the Clock Selections
   a. Select the clock edge to which the Timer/Counter responds
      - Positive (PCLOCK) or Negative (NCLOCK) edge
   b. Select the Clock Pre-scale Divider
3. Configure the Counter Values
   a. Enable the Top Counter Value
   b. Select a Timer/Counter Top value
4. Enable optional interrupts

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6.3.8.2. Watchdog Timer

Configure the Timer/Counter for Static or Dynamic operation. Dynamic operation enables the WISHBONE bus interface allowing a WISHBONE Controller to control the Timer/Counter.

To configure the Timer/Counter for Static or Dynamic operation in the Timer/Counter tab:

1. Configure the Timer/Counter Mode using the Mode Selection controls
   a. Select WATCHDOG mode
   b. Select Output Function
      - Hold static 0 (STATIC)
2. Update the Clock Selections
   a. Select the clock edge to which the Timer/Counter responds
      - Positive (PCLOCK) or Negative (NCLOCK) edge
   b. Select the Clock Pre-scale Divider
3. Configure the Counter Values
   a. Enable the Top Counter Value
   b. Select a Timer/Counter Top value
   c. Select an Output Compare Value
4. Enable interrupts (TC_INT)
   a. Select Output Compare Match
   b. Select Input Capture

6.3.8.3. PWM Output with variable duty cycle and period

Configure the Timer/Counter for Static or Dynamic operation. Dynamic operation enables the WISHBONE bus interface allowing a WISHBONE Controller to control the Timer/Counter.

To configure the Timer/Counter for Static or Dynamic operation in the Timer/Counter tab:

1. Configure the Timer/Counter Mode using the Mode Selection controls
   a. Select the FASTPWM mode
   b. Select Output Function
      - Hold static 0 (STATIC)
      - PWM (WAVE_GENERATOR)
      - Complement PWM (INV_WAVE_GENERATOR)
2. Update the Clock Selections
   a. Select the clock edge to which the Timer/Counter responds
      - Positive (PCLOCK) or Negative (NCLOCK) edge
   b. Select the Clock Pre-scale Divider
3. Configure the PWM Values
   a. Select the PWM period (Timer Counter Top)
   b. Select the PWM duty cycle (Output Compare Value)
      - Where the duty cycle is (Output Compare Value)/(Timer Counter Top): [(Timer Counter Top) – (Output Compare Value)]/(Timer Counter Top)

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6.3.8.4. PWM output with 50:50 duty variable phase and period

Configure the Timer/Counter for Static or Dynamic operation. Dynamic operation enables the WISHBONE bus interface allowing a WISHBONE Controller to control the Timer/Counter.

To configure the Timer/Counter for Static or Dynamic operation in the Timer/Counter tab:

1. Configure the Timer/Counter Mode using the Mode Selection controls
   a. Select FASTPWM
   b. Select Output Function
      - Hold static 0 (STATIC)
      - PWM (WAVE_GENERATOR)
      - Complement PWM (INV_WAVE_GENERATOR)
2. Update the Clock Selections
   a. Select the clock edge to which the Timer/Counter responds
      - Positive (PCLOCK) or Negative (NCLOCK) edge
   b. Select the Clock Pre-scale Divider
3. Configure the PWM Values
   a. Select the PWM period (Timer Counter Top)
   b. Select the Phase Adjustment (Output Compare Value)
      - Where Phase Adjustment is (360 degrees) × (Output Compare Value) / (Timer Counter Top)

6.3.9. Timer/Counter Design Tips

- For more information on the Timer/Counter register definitions and command sequences, refer to the Timer/Counter section in the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
- Take note when dynamically turning off components for power savings; the EFB requires the MachXO4 internal oscillator to be enabled even if it is not the source of the WISHBONE Clock. The exception is when the user clock is selected.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

7. Flash Access

You can access the Flash interface using the JTAG, SPI, I2C, or WISHBONE interfaces. The Flash access method slightly differs between MachXO4 (LFMXO4) and MachXO4 RoT devices.

The MachXO4 (LFMXO4) Flash consists of three sectors:

- Configuration Flash (Sector 0)
- UFM (Sector 1)
- Feature Row

The MachXO4 RoT Flash is divided into multiple subsectors:

- Two configuration memory sectors (CFG0 and CFG1)
- Four user flash memory sectors (UFM0, 1, 2, and 3)
- One feature row sector (FEA)
- One public key sector (PUBKEY)
- One Advanced Encryption Standard Key sector (AESKEY)
- One configuration security setting sector (CSEC)
- One user security setting sector (USEC)

The ports to access the Flash and the block diagram are shown in Table 7.1.

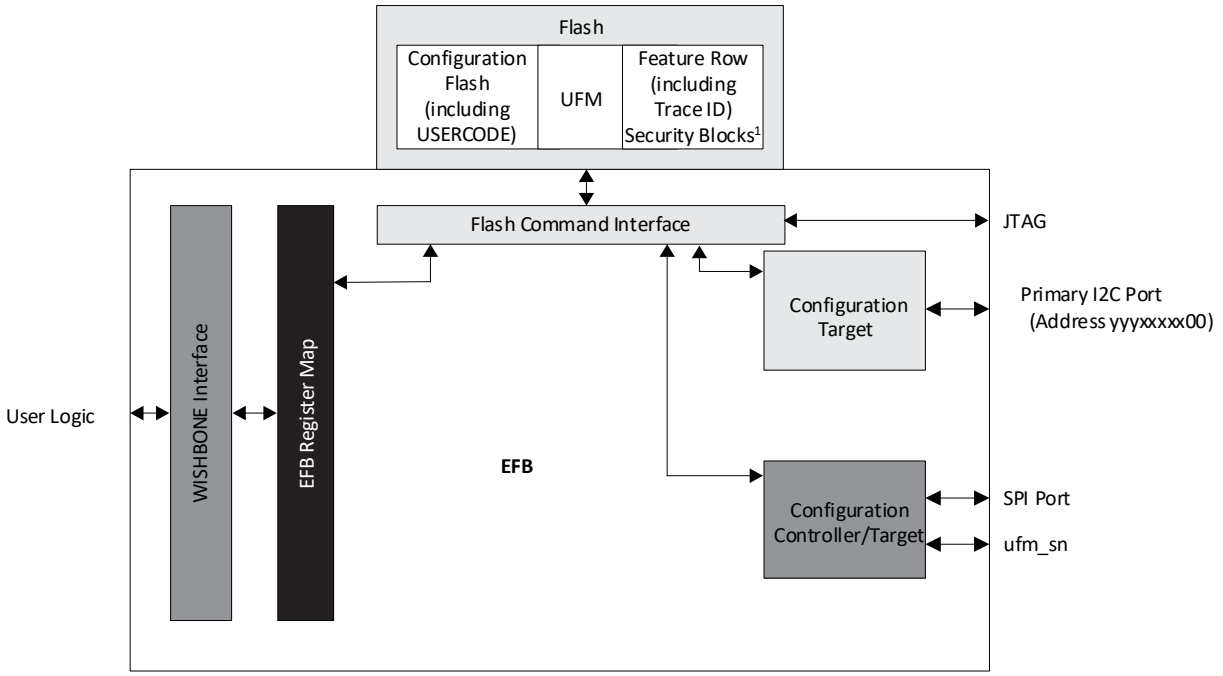
Table 7.1. MachXO4 Flash Access

	UFM[1]	Configuration Flash
JTAG	Yes	Yes
Target SPI Port with Chip Select (sn)	Yes	Yes
Target SPI Port with Chip Select (spi_scsn)	No	No
Primary Target I2C Port address (yyyxxxxx00)	Yes	Yes
Primary Target I2C Port address (yyyxxxxx01)	No	No
Secondary Target I2C Port address (yyyxxxxx10)	No	No
WISHBONE	Yes	Yes

Note:

1. Applicable only to MachXO4 RoT devices.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.



Note:
1. Security blocks are available only for MachXO4 RoT devices.

Figure 7.1. Flash Block Diagram

7.1. Flash Access Ports

The Configuration Logic arbitrates access to the Flash from the interfaces by the following priority. When higher priority ports are enabled, Flash access by lower priority ports is blocked:

- JTAG Port – All MachXO4 devices have a JTAG port, which can be used to perform Read/Write operations to the Flash. The JTAG port is compliant with the IEEE 1149.1 and IEEE 1532 specifications. JTAG has the highest priority to the Flash.
- Target SPI Port – All MachXO4 devices have a Target SPI port, which can be used to perform Read/Write operations to the Flash. Asserting the Flash Target Chip Select (ufm_sn) enables access.
- I2C Primary Port – All MachXO4 devices have an I2C port, which can be used to perform Read/Write operations to the Flash. The Primary I2C Port address is yyyxxxxx00 (where *y* and *x* are user programmable).
- WISHBONE Target Interface – The WISHBONE Target interface of the EFB module enables you to access the Flash from the FPGA user logic by creating a WISHBONE Controller. In addition to the WISHBONE bus signals, an interrupt request output signal is brought to the FPGA fabric. An IRQ (wbc_ufm_irq) is provided to assist the WB Controller to perform UFM/CF programming operations. It is configurable and provides information about the R/W FIFO, and arbitration errors.

Note: Enabling the Flash Interface using Enable Configuration Interface command 0x74 Transparent Mode temporarily disables certain features of the device, including:

- Power Controller
- GSR
- Hardened User SPI Port
- Hardened Primary User I2C Port

Functionality is restored after the Flash Interface is disabled using Disable Configuration Interface command 0x26 followed by Bypass command 0xFF.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

7.2. Interface to UFM

The MachXO4 (LFMXO4) and higher density devices provide one sector of User Flash Memory (UFM). You can access the UFM sector through the Configuration Logic interface using the JTAG, Configuration SPI, Primary Configuration I2C, or WISHBONE interfaces. The UFM is a separate sector within the on-chip Flash Memory and is organized by pages. Each page is 128 bits (16 bytes). Table 7.2 shows the UFM resources in MachXO4 (LFMXO4) devices, represented in bits, bytes, and pages.

The MachXO4 RoT devices provide four sectors of User Flash Memory (UFM0, 1, 2, and 3). You can access the UFM sectors through the Configuration Logic interface using the JTAG, Configuration SPI, Primary Configuration I2C, or WISHBONE interfaces. The UFM sectors are separate sectors within the on-chip Flash Memory and are organized by pages. In addition, the CFG sectors can be used as UFM if they are unused for configuration. Table 7.3 shows the UFM resources in MachXO4 RoT devices, represented in bits, bytes, and pages.

Table 7.2. UFM Resources in MachXO4 (LFMXO4) Devices

Device	UFM Bits	UFM Bytes	UFM Pages
LFMXO4-010	65,408	8,176	511
LFMXO4-015	65,408	8,176	511
LFMXO4-015 256 Ball Package	81,792	10,224	639
LFMXO4-025	81,792	10,224	639
LFMXO4-050	98,176	12,272	767
LFMXO4-050 400 Ball Package	261,888	32,736	2,046
LFMXO4-080	261,888	32,736	2,046
LFMXO4-110	458,496	57,312	3,582

Table 7.3. UFM Resources in MachXO4 RoT Devices

Device	Unit	UFM0	UFM1	UFM2	UFM3	CFG0[1]	CFG1[1]
LFMXO4-040, LFMXO4-050	Bits	98,176	98,176	147,200	24,448	737,024	737,024
	Bytes	12,272	12,272	18,400	3,056	92,128	92,128
	Pages	767	767	1,150	191	5,758	5,758
LFMXO4-080, LFMXO4-110	Bits	458,496	458,496	147,200	24,448	1,605,248	1,605,248
	Bytes	57,312	57,312	18,400	3,056	200,656	200,656
	Pages	3,582	3,582	1,150	191	12,541	12,541

Note:

1. Reclaimed CFG sectors are used as UFM.

The UFM is a general purpose Flash Memory. Refer to the MachXO4 Family Data Sheet (FPGA-DS-02125) or MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126) for the number of Programming/Erase Specifications. The UFM is typically used to store system-level data, Embedded Block RAM initialization data, or executable code for microprocessors and state machines. Use the following tools to partition the UFM resource:

- IP Catalog – Use IP Catalog to enable the UFM and to initialize the memory
- Spreadsheet View (the Radiant software) – The global sysConfig configuration options control the use of the UFM. Read the MachXO4 Programming and Configuration User Guide (FPGA-TN-02393) for a complete description of available options.

7.2.1. Initializing the UFM with IP Catalog

You can initialize the UFM sector with system-level non-volatile data and generate the EFB module through IP Catalog. Selecting the UFM tab in the EFB user interface displays the configurable settings of the UFM.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

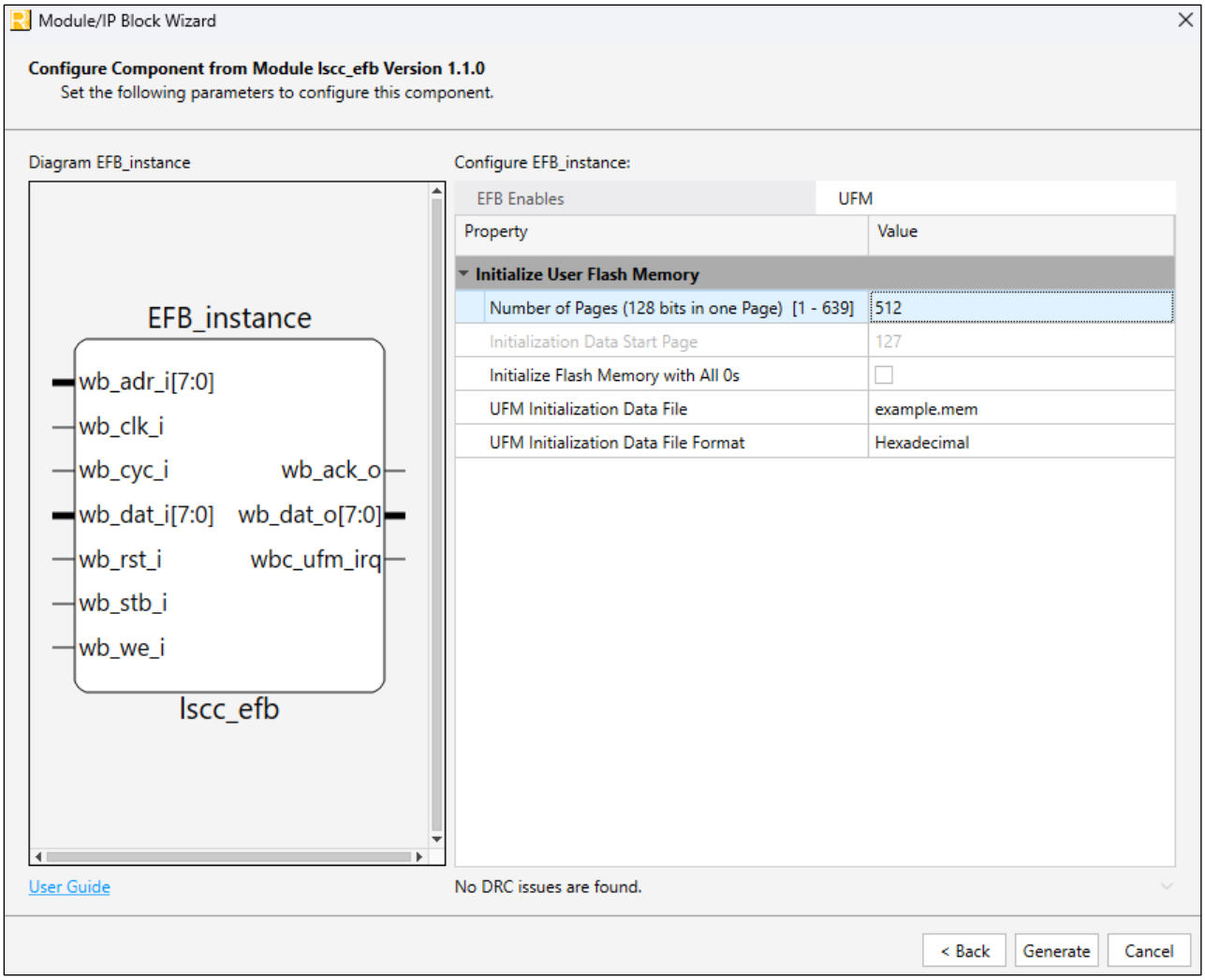
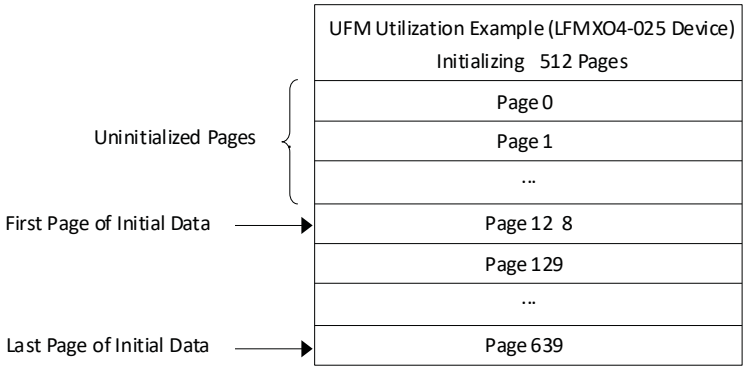


Figure 7.2. Initializing the UFM Sector with IP Catalog

Enter the number of pages to be pre-loaded with initialization data. Because initialization data is *bottom justified*, the read-only field *Initialization Data Starts at Page* informs the address of the first page that is initialized. In the example used for this document, 512 pages of the UFM sector of the LFMXO4-025 device are to be initialized with the content of the memory file **example.mem**.



The LFMXO4-025 device has 640 pages in the UFM sector. The initialization data occupies the bottom (highest) addressable pages of the UFM, while the top (lowest) addressable pages remain uninitialized (all zeros). The format of the memory file is page-based and can be expressed in binary or hexadecimal format.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

7.2.1.1. UFM Initialization Memory File

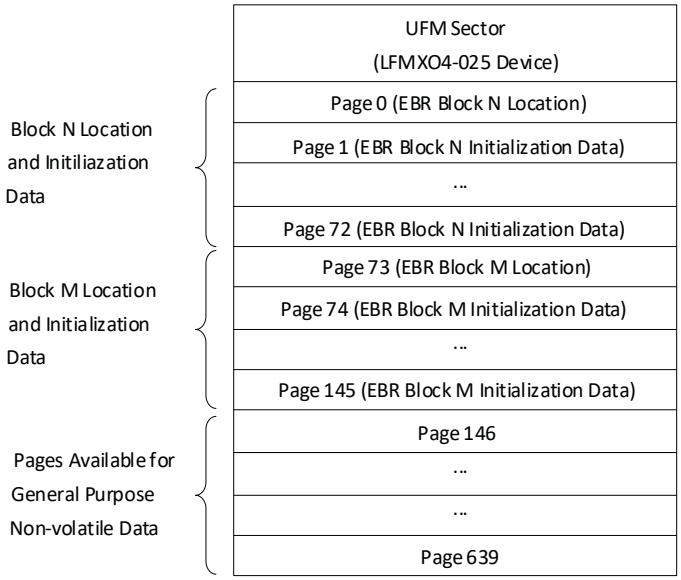
The initialization data file has the following properties and format:

```
Extension:      .mem
Format:         Binary, Hexadecimal
Data Width:     1 page (128 bits in one row of the file)
No. of Rows:    Less than or equal to the number of available pages
Example 1. Binary
1010…1010 (Placed at the starting page of the UFM initialization data, address = N)
1010…1010 (page address = N + 1)
1010…1010 (page address = N + 2)
…
1010…1010 (Placed at the highest UFM page address)
Example 2. Hexadecimal
A…B (Placed at the starting page of the UFM initialization data, address = N)
C…D (page address = N + 1)
E…F (page address = N + 2)
…
A…F (Placed at the highest UFM page address)
```

The most significant byte (byte 15) of the page is on the left side of the row. The least significant byte of the page (byte 0) is on the right side of the row. The most significant bit in a row is the left-most (bit 127), while the least significant is right-most (bit 0).

7.2.1.2. EBR Initialization

If you choose to share the UFM sector with EBR initialization data, reference the sector map below as an example when planning the design. Note at this time the EBR Mapping is not supported in the Radiant software.



Care must be taken when performing a Bulk-Erase operation to prevent the loss of EBR initialization data. Prior to erasing the UFM sector, make a copy of the pages holding the EBR block location and EBR initialization data in a secondary memory resource. After the UFM sector is erased, the data can be written back into the UFM. Upon power-up or a reconfiguration command, the EBR initialization data is automatically loaded in their respective EBR blocks. The EBR Block Location data guides the configuration logic on the location of the EBR block.

7.2.1.3. UFM in Lattice Radiant Software

The UFM sector is one of the Flash Memory resources of the MachXO4 device. The CONFIGURATION option in the Radiant Device Constraint Editor controls how UFM behaves. Each option's impact to the UFM is described below.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

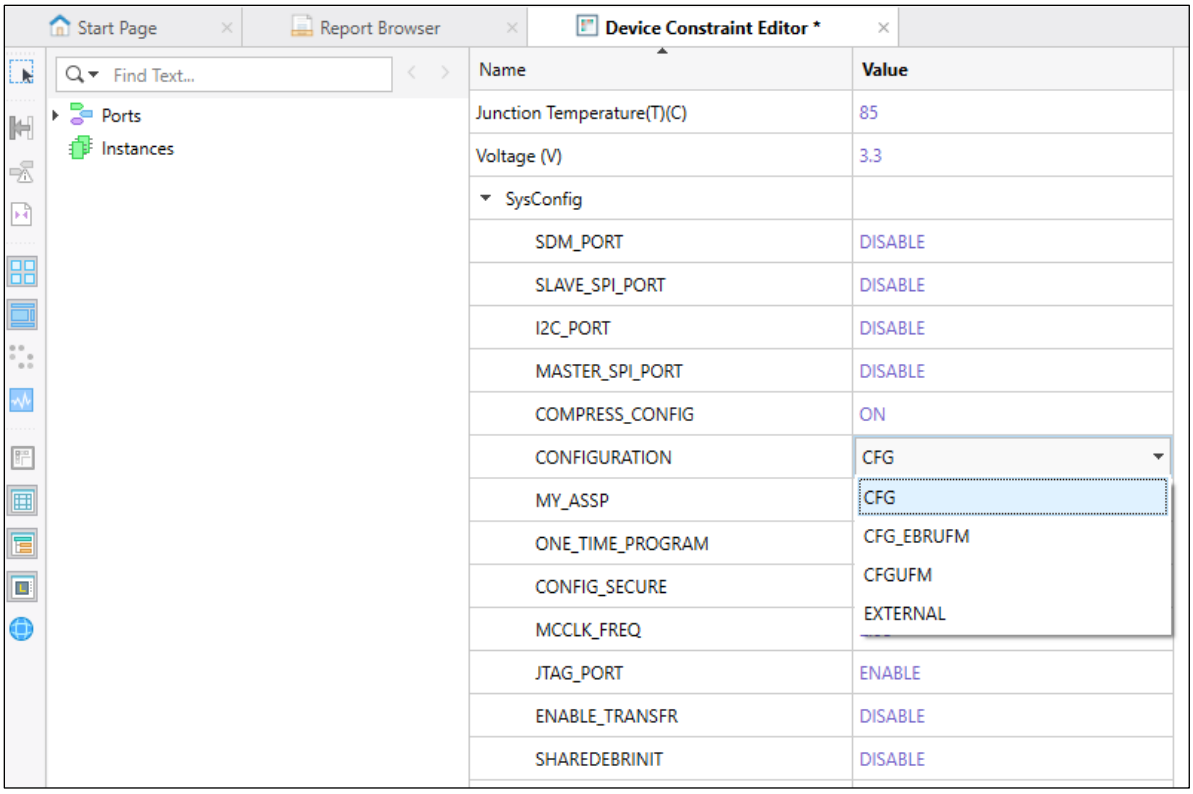


Figure 7.3. UFM in Radiant Spreadsheet View

7.3. Configuration Flash

The WISHBONE interface of the EFB module allows a WISHBONE controller to access the Configuration resources of MachXO4 devices. This is particularly useful for reading data from Configuration resources such as USERCODE and TraceID. A WISHBONE controller can update the Flash memory using the Configuration Logic's transparent mode. The new design is active after power-up or a Configuration Refresh operation.

For more information on programming the MachXO4, refer to the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).

For more information on the TraceID, refer to Using TraceID (FPGA-TN-02084).

Table 7.4. Configuration Flash Resources in MachXO4 (LFMXO4) Devices

Device	CFG Bits	CFG Bytes	CFG Pages
LFMXO4-010	278,400	34,800	2,175
LFMXO4-015	278,400	34,800	2,175
LFMXO4-015 256 Ball Package	409,344	51,168	3,198
LFMXO4-025	409,344	51,168	3,198
LFMXO4-050	737,024	92,128	5,758
LFMXO4-050 400 Ball Package	1,179,008	147,376	9,211
LFMXO4-080	1,179,008	147,376	9,211
LFMXO4-110	1,604,992	200,624	12,539

Table 7.5. Configuration Flash Resources in MachXO4 RoT Devices

Device	Unit	CFG0	CFG1
LFMXO4-040, LFMXO4-050	Bits	737,024	737,024
	Bytes	92,128	92,128

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Device	Unit	CFG0	CFG1
	Pages	5,758	5,758
LFMXO4-080, LFMXO4-110	Bits	1,605,248	1,605,248
	Bytes	200,656	200,656
	Pages	12,541	12,541

7.3.1. Flash Design Tips

- For more information on the Flash register definitions and command sequences, refer to the MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404).
- For more information on programming the MachXO4 devices, refer to the MachXO4 Programming and Configuration User Guide (FPGA-TN-02393).
- For more information on the TraceID, refer to Using TraceID (FPGA-TN-02084).
- Take note when dynamically turning off components for power savings. The EFB requires the MachXO4 internal oscillator to be enabled even if it is not the source of the WISHBONE Clock.
- It is recommended when accessing the Flash, the Feature Row be Read Only after initial programming as it contains the persistent settings for the Configuration ports.
- The Flash Chip Select (sn) is enabled when the SPI functions are enabled. Tie sn inactive (that is by high) if you are not using the Target SPI interface to access the Flash.
- The buses used to access the Flash have a priority. The bus priority is, from highest to lowest, the JTAG Port, Target SPI Port, I2C Primary Port, and WISHBONE Target Interface. When higher priority ports are enabled, Flash access by lower priority ports is blocked. You must define a process that prevents simultaneous access to the Flash by controllers on each configuration port.
- The Primary I2C port cannot be used for both Flash access and user functions in the same design.
- Enabling Flash Interface using Enable Configuration Interface command 0x74 Transparent Mode temporarily disables the Power Controller, GSR, and Hardened User SPI port. Functionality is restored after the Flash Interface is disabled using Disable Configuration Interface command 0x26 followed by Bypass command 0xFF.
- In Flash, *0* defines erased and *1* defines written.
- The smallest unit for a Write operation (bits => 1) is 1 page (16 bytes).
- The smallest unit for an Erase operation (bits => 0) is one sector (there are only two available Flash sectors: Configuration Flash and UFM).
- MachXO4 RoT UFM has a limited number of erase/programming cycles. The number of cycles is described in the MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126). Lattice recommends storing static data, or data that varies infrequently, in the UFM.
- For more information on Lattice reference designs, refer to the Lattice Solutions Reference Designs web page.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

8. Interface to Dynamic PLL Configuration Settings

The WISHBONE interface of the EFB module can be used to dynamically update configurable settings of the Phase-Locked Loops (PLLs) in the MachXO4 devices.

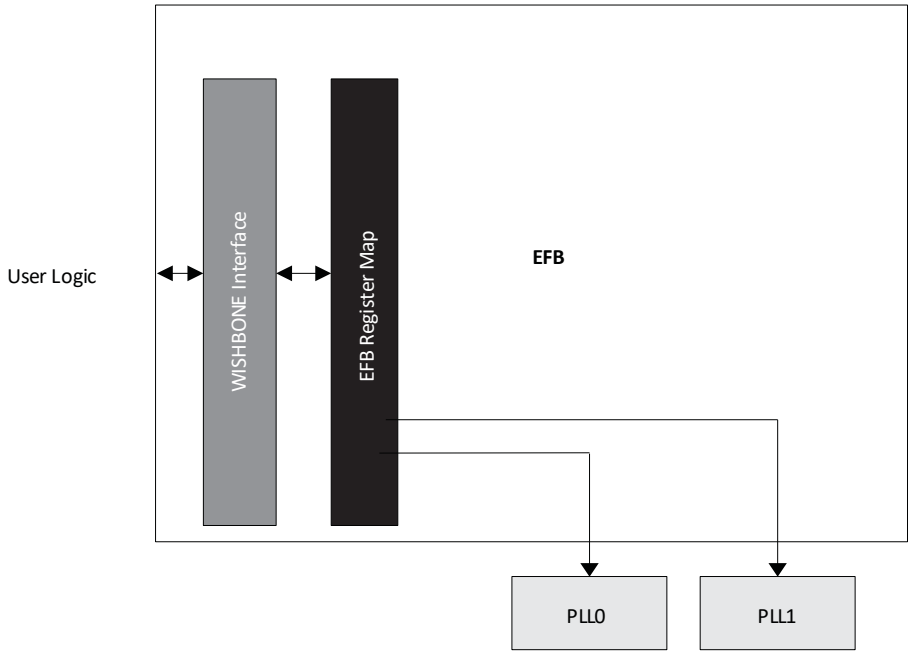


Figure 8.1. EFB Interface to Dynamic PLL

There can be up to two PLLs in a MachXO4 device. PLL0 has an address range from 0x00 to 0x1F in the EFB register map. PLL1 (if present) has an address range from 0x20 to 0x3F in the EFB register map. Refer to the MachXO4 sysCLOCK PLL Design User Guide (FPGA-TN-02391) for details on PLL configuration bits and recommended usage.

You can enable the WISHBONE interface to the PLL components in IP Catalog through the EFB user interface as shown in Figure 8.2.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

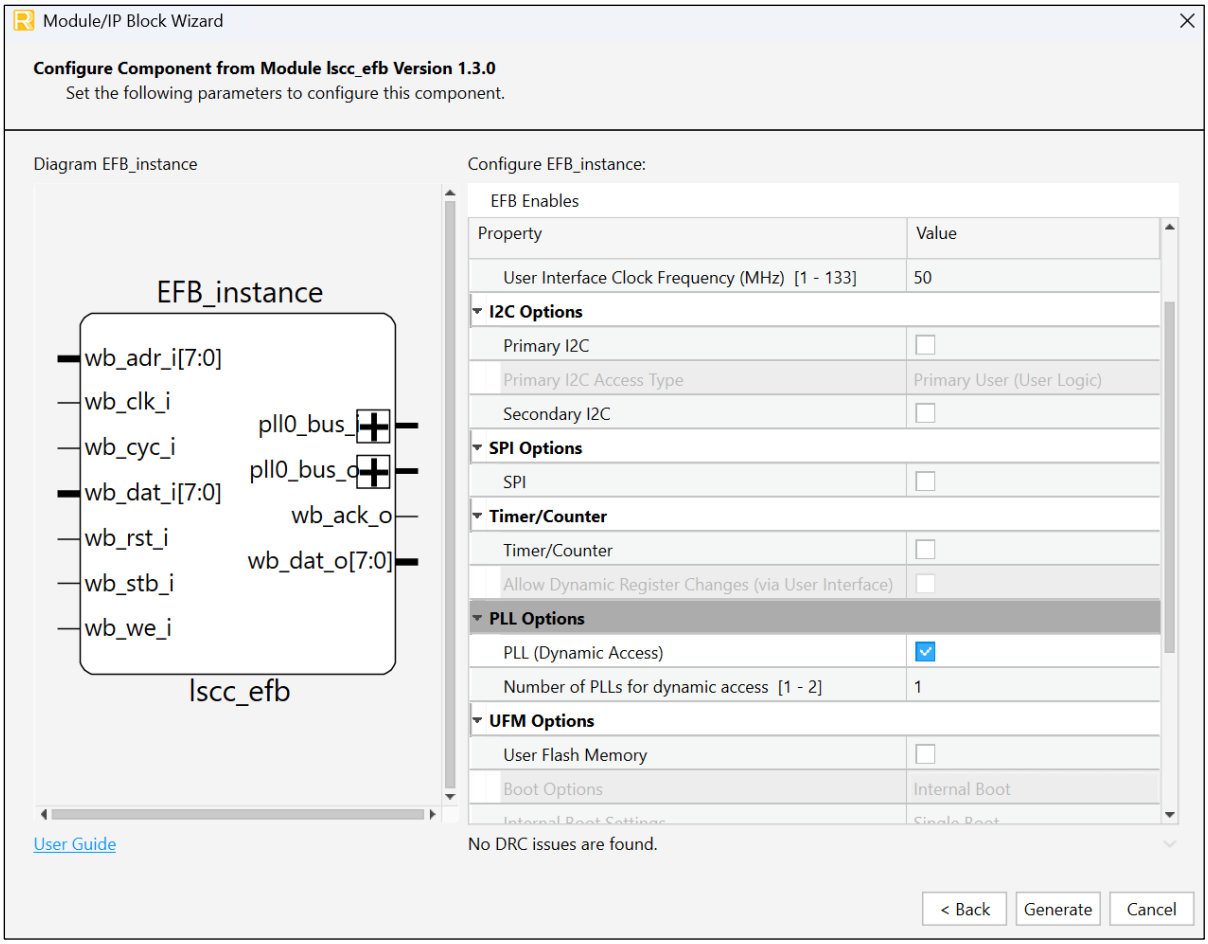


Figure 8.2. Interface to Dynamic PLL Configuration Settings

Enabling the interface to dynamically control PLL settings through the WISHBONE interface generates an IP with the following ports, which are used for dedicated connections to the PLL(s).

Table 8.1 documents the signals that are generated with the IP. Each signal has a description of its usage and connection within a design project.

Table 8.1. PLL Interface – IP Signals

Signal Name	I/O	Width	Description
pll0_bus_i	Input	9	Input data and control bus. You must connect this bus only to a PLL component that is instantiated in the design.
pll0_bus_i	Input	9	Input data and control bus. You must connect this bus only to a PLL component that is instantiated in the design.
pll0_bus_o	Output	17	Output data and control bus. You must connect this bus only to a PLL component that is instantiated in the design.
pll0_bus_1	Output	17	Output data and control bus. You must connect this bus only to a PLL component that is instantiated in the design.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

9. MachXO4 RoT Tamper Detection

Tamper Detection is one of the key features of the enhanced EFB in MachXO4 RoT devices. It creates an alert if an illegal operation is detected, and you can institute measures to protect the device from further threats. When Tamper Detect is enabled, the EFB can report which configuration port caused the attack, and it can lock that port.

For a detailed description of the Tamper Detect feature, refer to the MachXO4 Root of Trust Tamper Detection/Response (FPGA-TN-02434).

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

References

- MachXO4 Family Data Sheet (FPGA-DS-02125)
- MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126)
- MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404)
- MachXO4 Programming and Configuration User Guide (FPGA-TN-02393)
- MachXO4 Root of Trust Tamper Detection/Response (FPGA-TN-02434)
- Power and Thermal Estimation and Management for MachXO4 Devices (FPGA-TN-02409)
- MachXO4 sysCLOCK PLL Design User Guide (FPGA-TN-02391)
- MachXO4 Hardware Checklist (FPGA-TN-02411)
- Improving Noise Immunity Serial Interfaces
- Using TraceID (FPGA-TN-02084)
- MachXO4 web page
- Lattice Radiant Software web page
- Lattice Solutions Reference Designs web page
- Lattice Insights web page for Lattice Semiconductor training courses and learning plans

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

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.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Revision History

Revision 1.1, September 2026

Section	Change Summary
All	Made editorial fixes.
Abbreviations in This Document	• Added *CPU*, *FIFO*, *IRQ*, *JTAG*, *LUT*, *MDF*, *OCRF*, *OSC*, *OVF*, *PWM*, *RAM*, *ROE*, *RoT*, *RTL*, *SCL*, *SDA*, *SRAM*, *UFM*, and *VHDL*. • Removed *DDR*, *LVDS*, *PCB*, and *WLCSP*.
Introduction	• Added a MachXO4 terminology note. • Added a description on security block and tamper detection. • Updated Figure 1.1. Embedded Functional Block (EFB).
WISHBONE Bus Interface	Added a Lattice reference design reference to the WISHBONE Design Tips section.
Generating an EFB Module with IP Catalog	• Updated Figure 3.1. EFB Module in IP Catalog and Figure 3.2. Generating an EFB Module with IP Catalog. • Added Embedded Functional Block Connection Summary for MachXO4 RoT devices.
Hardened I2C IP Cores	• Added the *MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126)* reference to Table 4.2. I2C Primary – IP Signals. • Updated Figure 4.1. I2C Block Diagram, Figure 4.2. I2C Primary Block Diagram, and Figure 4.4. Configuring I2C Functions of the EFB Module with IP Catalog. • Updated the I2C Design Tips section.
Hardened SPI IP Core	• Updated Figure 5.1. SPI Block Diagram, Figure 5.2. Configuring SPI Functions of the EFB Module with IP Catalog, Figure 5.3. External Controller SPI Device Accessing the Target MachXO4 User SPI, and Figure 5.5. External Controller SPI Device Accessing the MachXO4 Configuration Logic. • Added the *MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126)* reference to Table 5.2. SPI – IP Signals. • Added the *MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404)* reference to the SPI Controller Clock Rate section. • Updated the SPI Design Tips section.
Timer/Counter	Updated Figure 6.5. Configuring Timer/Counter.
Flash Access	• Updated this section. • Removed the *MachXO4 (LFMXO4) Configuration Parameter Options* and *MachXO4 (LFMXO4) Configuration Parameter Options* sections.
Interface to Dynamic PLL Configuration Settings	Updated Figure 8.2. Interface to Dynamic PLL Configuration Settings.
MachXO4 RoT Tamper Detection	Added this section.
References	Added *MachXO4 Family RoT Devices Data Sheet (FPGA-DS-02126)*, *MachXO4 Root of Trust Tamper Detection/Response (FPGA-TN-02434)*, and *MachXO4 Hardware Checklist (FPGA-TN-02411)*.

Revision 1.0, December 2025

Section	Change Summary
All	Initial release.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.



www.latticesemi.com