FIR Filter IP Core - Lattice Radiant Software

User Guide

FPGA-IPUG-02095-1.2

May 2022

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults and associated risk the responsibility entirely of the Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
FIR	Finite Impulse Response

1. Introduction

The Lattice Finite Impulse Response (FIR) Filter IP Core is implemented using high performance sysDSP™ blocks available in Lattice devices. The input data, coefficient and output data widths are configurable over a wide range. The IP core uses full internal precision while allowing variable output precision with several choices for saturation and rounding. The coefficients of the filter can be specified at generation time and/or reloadable during run-time through input ports.

1.1. Quick Facts

Table 1.1 presents a summary of the FIR filter.

Table 1.1. FIR Filter Quick Facts

IP Requirements	FPGA Families Supported	CrossLink™-NX, Certus™-NX, CertusPro™-NX, MachXO5-NX
Resource Utilization	Targeted Devices	LIFCL-40, LIFCL-17, LFD2NX-40, LFD2NX-17, LFCPNX-100, LFMXO5-25
	Supported User Interface	Native interface, see Signal Description section
	Resources	See Table A.1 and Table A.2
Design Tool Support	Lattice Implementation	IP Core v1.x.x - Lattice Radiant Software 2.2 or later
	Synthesis	Lattice Synthesis Engine (LSE)
		Synopsys® Synplify Pro® for Lattice
	Simulation	For a list of supported simulators, see the Lattice Radiant Software User Guide

1.2. Features

The key features of the FIR Filter include:

- Variable number of taps up to 56
- Input data and coefficient widths of 4 to 18 bits
- Signed or unsigned input data and coefficients
- Coefficients symmetry and negative symmetry optimization
- Support for half-band filter
- Re-loadable coefficients support
- Full precision arithmetic
- Selectable output width and precision
- Selectable overflow: wrap-around or saturation
- Selectable rounding: rounding up, round away from zero, round towards zero and convergent rounding
- Width and precision specified using fixed point notations

1.3. Conventions

1.3.1. Nomenclature

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

1.3.2. Signal Names

Signal names that end with:

- _*n* are active low (asserted when value is logic 0)
- _*i* are input signals
- _*o* are output signals
- _*io* are bi-directional input/output signals

1.3.3. Attribute Names

Attribute names in this document are formatted in title case and italicized (*Attribute Name*).

2. Functional Descriptions

2.1. Overview

A top-level block diagram of FIR Filter IP Core is shown in Figure 2.1.

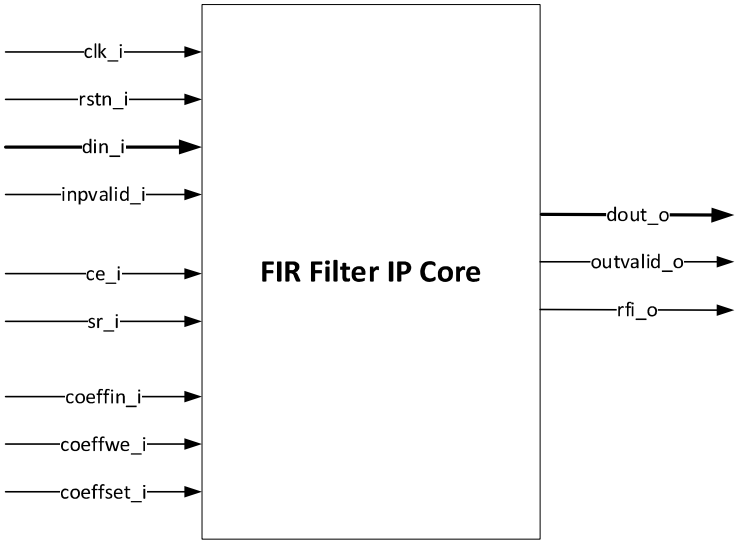

Figure 2.1. FIR Filter IP Core Block Diagram

Functional diagram is shown in Figure 2.2. The data and coefficients are stored in different memories shown as the tap memory and coefficients memory. The symmetry adder, multiplier array and adder tree are implemented using DSP blocks. The symmetry adder is used if the coefficients are symmetric. The adder tree performs the sum of the products. Depending on the configuration, the adder tree, or a part of it, is implemented inside the DSP blocks. The output processing block performs the output width reduction and precision control. This also contains logic to support different types of rounding and overflow. The control logic block manages the scheduling of data and arithmetic operations based on the type of filter.

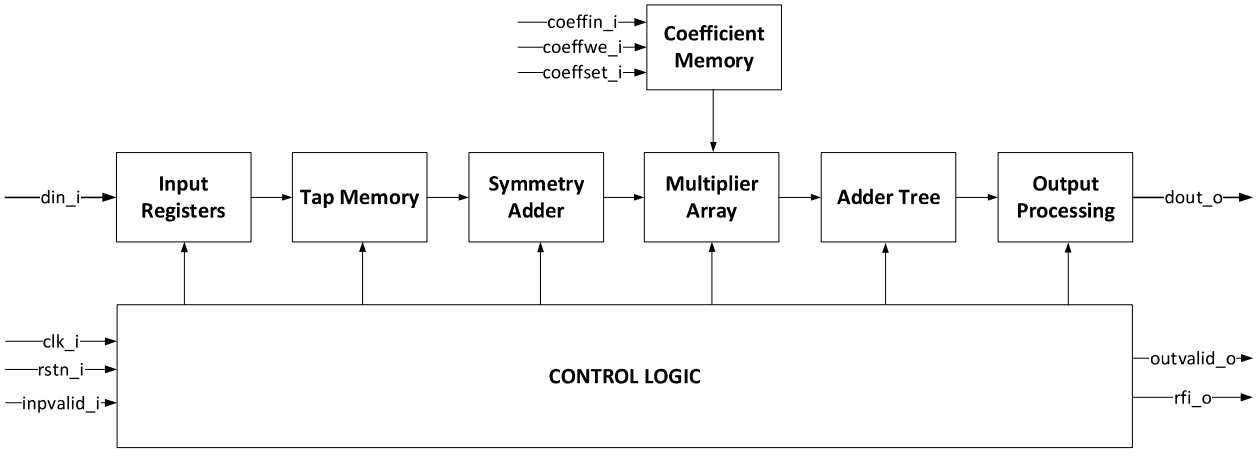

Figure 2.2. FIR Filter IP Core Functional Diagram

2.2. FIR Filter IP Core Architecture

FIR Filter operation on data samples can be described as a sum-of-products operation. For an N-tap FIR filter, the current input sample and (N-1) previous input samples are multiplied by N filter coefficients and the resulting N products are added to give one output sample as shown below:

$$y_n = \sum_{i=0}^{N-1} x_{n-i}h_i = x_n h_0 + x_{n-1}h_1 + \ldots + x_{n-N+1}h_{N-1}$$

wherein: h_n, where n = 0,1,…, N-1, is the impulse response;

x_n, where n = 0,1,…,∞ is the input;

y_n, where n = 0,1,…,∞ is the output;

N-1, number of delay elements, represents the order of the filter;

N, the number of input data samples (current and previous) used for one output sample represents the number of filter taps ().

2.2.1. Direct-form

In the direct-form implementation shown in Figure 2.3, the input samples are shifted into a shift register queue and each shift register is connected to a multiplier. The products from the multipliers are added to get the FIR filter's output sample.

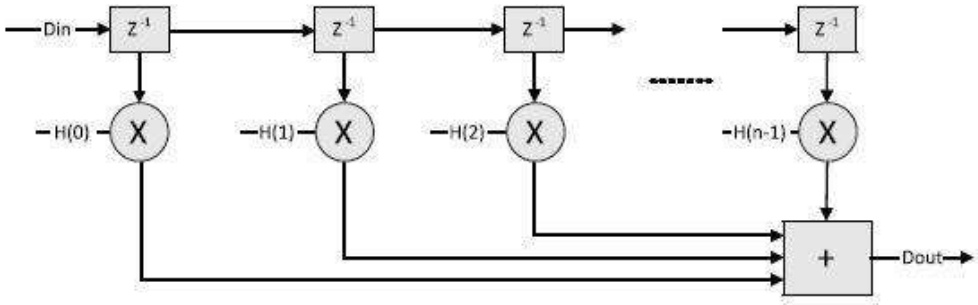

Figure 2.3. Direct-form FIR Filter Implementation

2.2.2. Symmetric

The impulse response for most FIR filters is symmetric. This symmetry can generally be exploited to reduce the arithmetic requirements and produce area-efficient filter realizations. It is possible to use only one half of the multipliers for symmetric coefficients compared to that used for a similar filter with non-symmetric coefficients. An implementation for symmetric coefficients is shown in Figure 2.4.

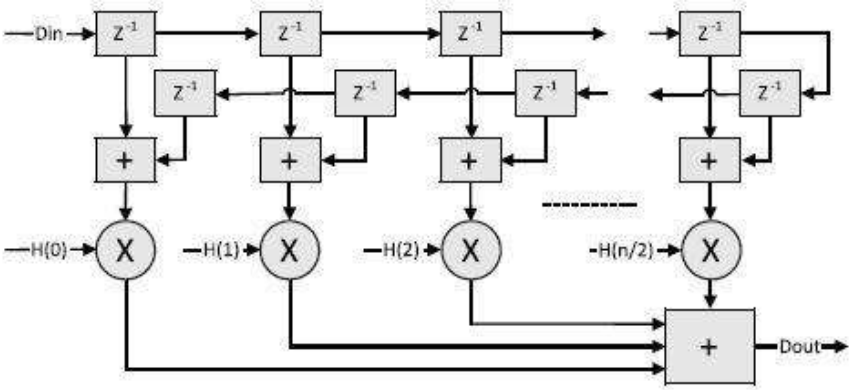

Figure 2.4. Symmetric Coefficients FIR Filter Implementation

2.3. Signal Description

Table 2.1 lists FIR Filter IP Core top-level input and output signals and their descriptions.

Table 2.1. FIR Filter IP Core Signal Description

Port Name	Direction	Bits	Description
Clocks and Reset			
clk_i	Input	1	System clock for data and control inputs and outputs.
rstn_i	Input	1	System wide asynchronous active-low reset signal.
General			
din_i	Input	*Input Data Width*	Input data
inpvalid_i	Input	1	Input valid signal. The input data din_i is read-in only when inpvalid_i is high.
dout_o	Output	*Output Width*	Output data
outvalid_o	Output	1	Output data qualifier. Output data dout_o is valid only when this signal is high.
rfi_o	Output	1	Ready for input. This output, when high, indicates that the IP core is ready to receive the next input data. A valid data may be applied at din only if rfi_o was high during the previous clock cycle.
When *Reloadable coefficients* is selected			
coeffin_i	Input	See ***Notes*** at the end of the table	Coefficients input. The coefficients have to be loaded through this port in a specific order. All the coefficients need to be loaded in one batch, while keeping the signal coeffwe_i high during the entire duration of loading. After all the coefficients are loaded, the input signal coeffset_i must be pulsed high for one clock cycle for the new coefficients to take effect.
coeffwe_i	Input	1	When asserted, the value on bus coeffin_i is written into coefficient memories.
coeffset_i	Input	1	This input is used to signal the filter to use the recently loaded coefficient set. This signal must be pulsed high for one clock cycle after the loading the entire coefficient set using coeffin_i and coeffwe_i.
Optional			
ce_i	Input	1	Clock Enable. While this signal is de-asserted, the core ignores all other synchronous inputs and maintain its current state
sr_i	Input	1	Synchronous Reset. When asserted for at least one clock cycle, all the registers in the IP core are initialized to reset state.

Notes:

1. Width for signed type and symmetric interpolation is Coefficients width +1.
2. Width for unsigned and symmetric interpolation is Coefficients width +2.
3. Width for all other cases is Coefficients width.

2.4. Attribute Summary

Table 2.2 provides the list of user selectable and compile time configurable parameters for the FIR Filter IP Core. The parameter settings are specified using the FIR Filter user interface in Lattice Radiant.

Table 2.2. Attributes Table

Attribute	Selectable Values	Default	Dependency on Other Attributes
Number of Taps	1–56 if *Half Band == True* or *Symmetric Coefficients = True* otherwise, it is 1–28	16	*Half Band == True* or *Symmetric Coefficients == True*
Reloadable Coefficients	Checked, Unchecked	Unchecked	—
Reorder Coefficients Inside	Checked, Unchecked	Unchecked	*Reloadable Coefficients == Checked*
Symmetric Coefficients	Checked, Unchecked	Unchecked	—
Negative Symmetry	Checked, Unchecked	Unchecked	*Symmetric Coefficients == Checked*
Half Band	Checked, Unchecked	Unchecked	—
Coefficients Radix	Binary, Hex, Decimal, Floating Point	Decimal	*Reloadable Coefficients == Unchecked*
Coefficients File	Browser Window	None	*Reloadable Coefficients == Unchecked*
Input Data Type	Signed, Unsigned	Signed	—
Input Data Width	4–18	16	—
Input Data Binary Point Position	–2 – *Input Data Width* + 2	0	—
Coefficients Type	Signed, Unsigned	Signed	—
Coefficients Width	4 - 18	16	—
Coefficients Binary Point Position	–2 – *Coefficients Width* + 2	0	—
Output Width	4 – *Input Data Width* + *Coefficients Width* + ceil ($\log_2$ (*Number of Taps*))		—
Output Binary Point Position	4 + *Input Data Binary Point Position* + *Coefficients Binary Point Position* – Full Precision – *Output Width* + *Input Data Binary Point Position* + *Coefficients Binary Point Position* - 4	0	—
Output Width Full Precision	Calculated	36	Display information only
Output Binary Point Position Full Precision	Calculated	0	Display information only
Overflow	Saturation, Wrap Around	Saturation	*Output Width != Output Width Full Precision*
Rounding	None, Rounding Up, Rounding away from zero, Rounding towards zero, Convergent rounding	None	*Output Width != Output Width Full Precision*
Data Memory Type	EBR, Distributed, Auto	EBR	—
Coefficients Memory Type	EBR, Distributed, Auto	EBR	—
Output Buffer Type	EBR, Distributed, Auto	EBR	—
Synchronous Reset (sr_i)	Checked, Unchecked	Unchecked	—
Clock Enable (ce_i)	Checked, Unchecked	Unchecked	—
Frequency Constraint (MHz)	1–200	200	—

Table 2.3. Attribute Description

Attribute	Description
General Tab	
Filter Specifications	
Number of Taps	Specifies the number of taps
Coefficients Specifications	
Reloadable Coefficients	Indicates whether the coefficients are fixed or reloadable. If checked, the coefficients can be reloaded during core operation using the input port coeffin_i. With this option, the desired coefficients must be loaded before the operation of the filter.
Reorder Coefficients Inside	If this option is selected, the coefficients can be entered in the normal sequence to the core, and the core internally reorders them as required.
Symmetric Coefficients	Indicates that the coefficients are used in symmetric form. If this is checked, the number of taps expected is twice the number of coefficients on the coefficient file.
Negative Symmetry	If this is checked, the coefficients are considered to be negative symmetric. The second half of coefficients are made equal to the negative of the coefficients on the coefficient file.
Half Band	If this is checked, only one half of the number of coefficients (if the number of taps is odd, the half value is rounded to the next higher integer) is read from the initialization file.
Coefficients Radix	This option allows you to specify the radix for the coefficients in the coefficients file with the following considerations: a. For decimal radix, the negative values have a preceding unary minus sign. b. For hexadecimal (Hex) and binary radices, the negative values must be written in 2's complement form using exactly as many digits as specified by the *Coefficients Width* parameter. c. The floating-point coefficients are specified in the form <nn…n>.<dd…d>, where the digits 'n' denote the integer part and the digits' d' the decimal part. The values of the floating-point coefficients must be consistent with the *Coefficients width* and *Coefficients Binary Point Position* parameters. For example, if <nn...n>.<dd...d> is 8.4 and *Coefficients Type* is unsigned, the value of the coefficients should be between 0 and 11111111.1111 (255.9375).
Coefficients File	Indicates the name and location of the coefficients file. This text file has one coefficient per line. If the *Coefficients File* is not specified, the filter is initialized with a default coefficient set consisting of 16 coefficients.
I/O Specifications Tab	
Data	
Input Data Type	Shows whether input data type to be used is signed or unsigned The input data is interpreted as a 2's complement number if the type is signed.
Input Data Width	Specifies the input data width.
Input Data Binary Point Position	This number specifies the bit position of the binary point from the LSB of the input data. If the number is zero, the point is right after LSB, if positive, it is to the left of LSB and if negative, it is to the right of LSB.
Coefficients	
Coefficients Type	This option allows you to specify the coefficients type as signed or unsigned. The coefficient data is interpreted as a 2's complement number if the type is signed.
Coefficients Width	Specifies the coefficients width.
Coefficients Binary Point Position	This number specifies the bit position of the binary point from the LSB of the coefficients. If the number is zero, the point is right after LSB; if positive, it is to the left of LSB and if negative, it is to the right of LSB.
Output	
Output Width	Specifies the output data width. The core's output is usually a part of the full precision output equal to the *Output Width* and extracted based on the different *Output Binary Point Position* parameters.

Attribute	Description
	The format for the internal full precision output is displayed as static text next to the *Output Width Full Precision*.
Output Binary Point Position	Specifies the bit position of the binary point from the LSB of the actual core output. If the number is zero, the point is right after LSB, if positive, it is to the left of LSB and if negative, it is to the right of LSB. This number, together with the parameter *Output Width*, determines how the actual core output is extracted from the true full precision output. The precision control parameters *Overflow* and *Rounding* are applied respectively when MSBs and LSBs are discarded from the true full precision output.
Output Width Full Precision	Shows the full precision of the output width when the resulting value is not rounded.
Output Binary Point Position Full Precision	Shows the binary point position of the output when the resulting value is not rounded.
Precision Control	
Overflow	This can be used whenever there is a need to drop some of the MSBs from the true output. The following options are supported: a. Saturation: The output value is clipped to the maximum if positive or minimum if negative, while discarding the MSBs. b. Wrap-around: The MSBs are simply discarded without making any correction.
Rounding	This option allows you to specify the rounding method when there is a need to drop one or more LSBs from the true output. The following five options are supported for rounding: a. None: Discards all bits to the right of the output least significant bit and leaves the output uncorrected. b. Rounding up – Rounds up to nearest positive number. c. Rounding away from zero – Rounds away from zero if the fractional part is exactly one-half. d. Rounding towards zero – Rounds towards zero if the fractional part is exactly one-half. e. Convergent rounding – Rounds to the nearest even value if the fractional part is exactly one-half.
Implementation Tab	
Memory Type	
Data Memory Type	Specifies the type of memory that is used for storing the input data. The following options are supported: a. EBR: EBR memories are used for storing the data. b. Distributed: Lookup-table based distributed memories are used for storing data. c. Auto: EBR memories are used for memory sizes deeper than 128 locations and distributed memories are used for all other memories.
Coefficients Memory Type	Specifies the type of memory that is used for storing the coefficients. The following options are supported: a. EBR: EBR memories are used for storing the data. b. Distributed: Lookup-table based distributed memories are used for storing data. c. Auto: EBR memories are used for memory sizes deeper than 128 locations and distributed memories are used for all other memories.
Output Buffer Type	Indicates the memory type for the output buffer.
Optional Ports	
Synchronous Reset (sr_i)	When enabled, this signal resets all the registers in the FIR filter IP core.
Clock Enable (ce_i)	When enabled, this can be used for power saving when the core is not being used. Use of clock enable port increases the resource utilization and may affect the performance due to the increased routing congestion.
Synthesis Options	
Frequency Constraint (MHz)	Specifies the constraint frequency for clk_i input.

3. IP Generation and Evaluation

This section provides information on how to generate and synthesize FIR Filter IP Core using Lattice Radiant Software, as well as on how to run simulation, synthesis and hardware evaluation. For more details on the Lattice Radiant Software, refer to the Lattice Radiant Software User Guide.

3.1. Licensing the IP

An IP core-specific license string is required to enable full use of the FIR Filter IP Core in a complete, top-level design.

You can fully evaluate the IP core through functional simulation and implementation (synthesis, map, place and route) without an IP license string. This IP core supports Lattice's IP hardware evaluation capability, which makes it possible to create versions of the IP core, which operate in hardware for a limited time (approximately four hours) without requiring an IP license string. See Hardware Evaluation section for further details. However, a license string is required to enable timing simulation and to generate bitstream file that does not include the hardware evaluation timeout limitation.

3.2. Generation and Synthesis

The Lattice Radiant Software allows you to customize and generate modules and IPs and integrate them into the device architecture. The procedure for generating FIR Filter IP Core in Lattice Radiant Software is described below.

To generate the FIR Filter IP Core:

1. Create a new Lattice Radiant Software project or open an existing project.
2. In the **IP Catalog** tab, double-click on **FIR Filter** under **IP, DSP** category. The **Module/IP Block Wizard** opens as shown in Figure 3.1. Enter values in the **Instance name** and the **Create in** fields and click **Next**.

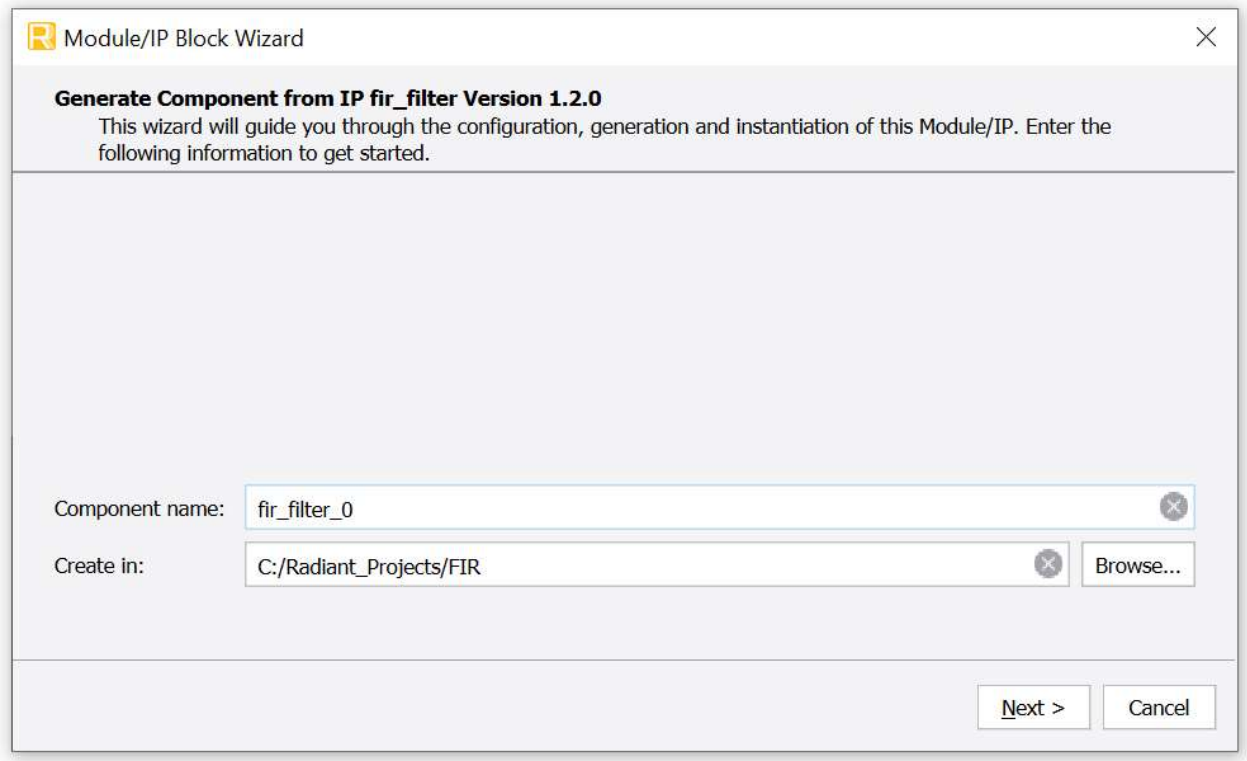

Figure 3.1. Module/IP Block Wizard

3. In the module's dialog box of the **Module/IP Block Wizard** window, customize the selected FIR Filter IP Core using drop-down menus and check boxes. As a sample configuration, see Figure 3.2. For configuration options, see the Attribute Summary section.

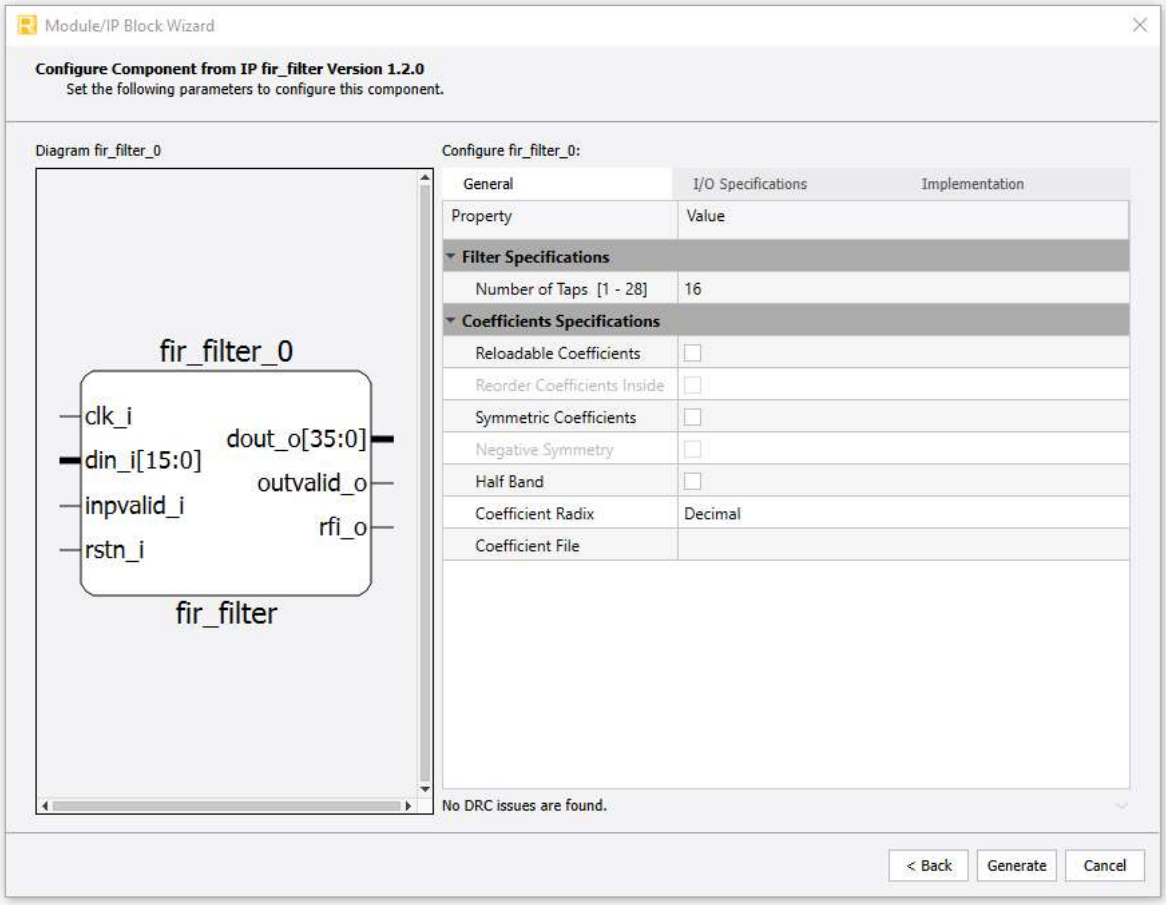

Figure 3.2. Configure User Interface of FIR Filter IP Core

4. Click **Generate**. The **Check Generating Result** dialog box opens, showing design block messages and results as shown in Figure 3.3.

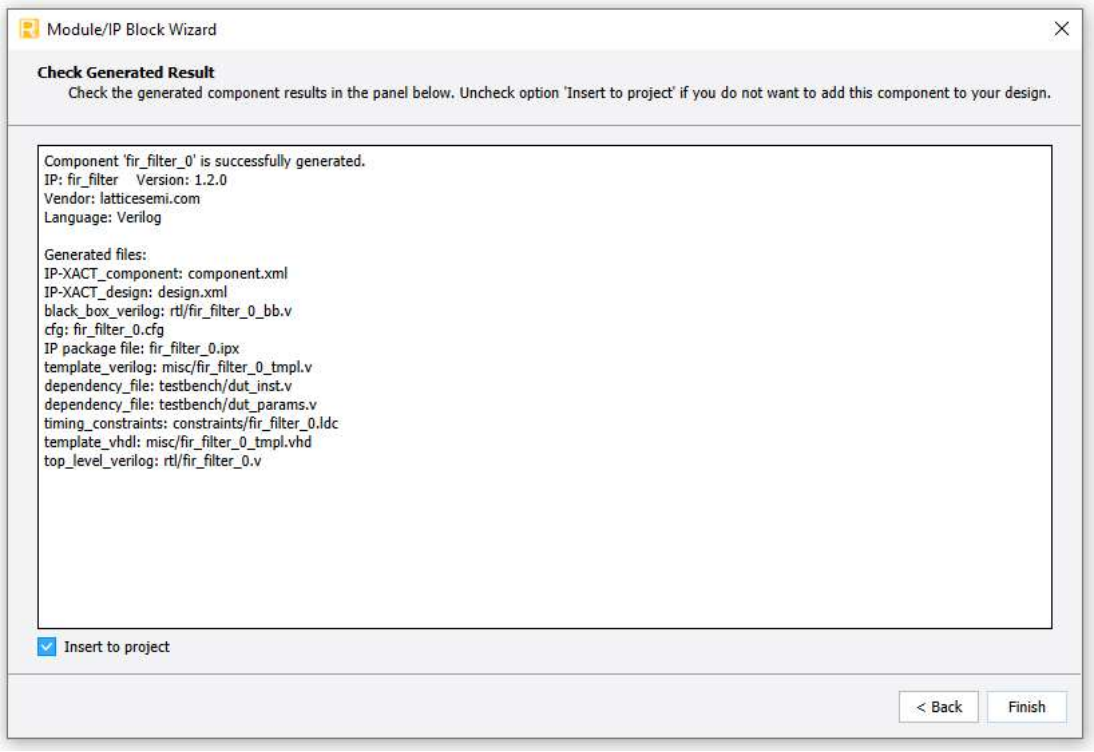

Figure 3.3. Check Generated Result

5. Click the **Finish** button. All the generated files are placed under the directory paths in the **Create in** and the **Instance name** fields shown in Figure 3.1.

The generated FIR Filter IP Core package includes the black box (<Instance Name>_bb.v) and instance templates (<Instance Name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Instance Name>.v) that can be used as an instantiation template for the IP core is also provided. You may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in Table 3.1.

Table 3.1. Generated File List

Attribute	Description
<Instance Name>.ipx	This file contains the information on the files associated to the generated IP.
<Instance Name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/<Instance Name>.v	This file provides an example RTL top file that instantiates the IP core.
rtl/<Instance Name>_bb.v	This file provides the synthesis black box.
misc/<Instance Name>_tmpl.v misc /<Instance Name>_tmpl.vhd	These files provide instance templates for the IP core.

3.3. Running Functional Simulation

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

To run functional simulation using the default simulator:

1. Click the  button located on the **Toolbar** to initiate the **Simulation Wizard** shown in Figure 3.4.

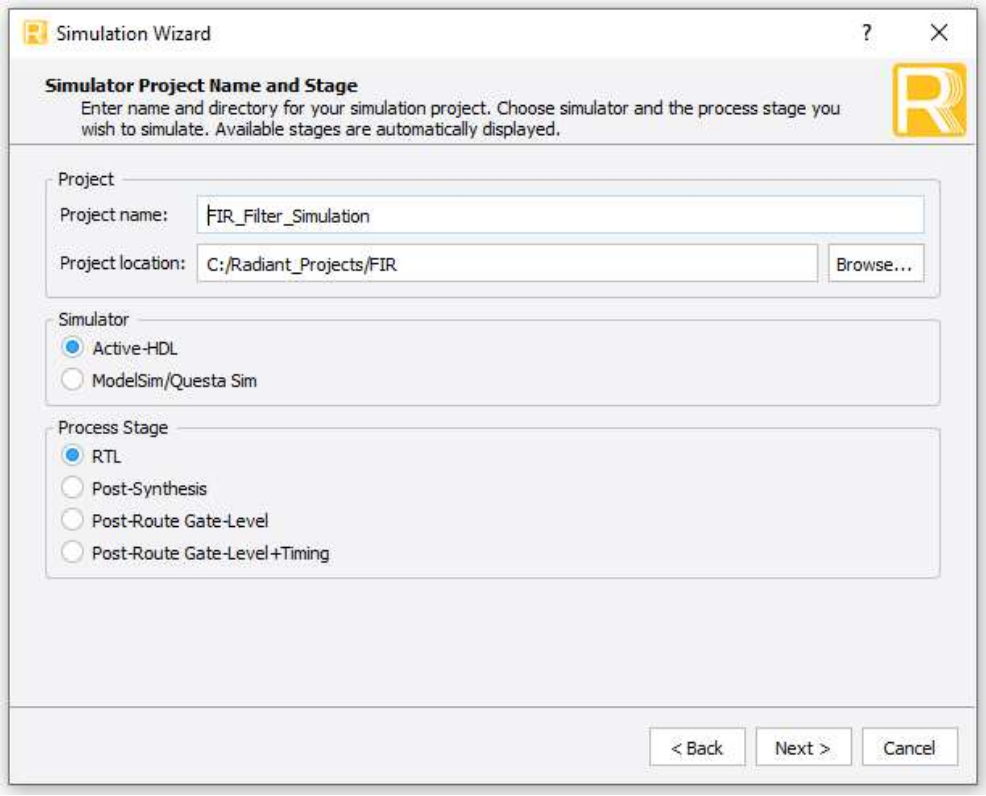

Figure 3.4. Simulation Wizard

2. Click **Next** to open the **Add and Reorder Source** window as shown in Figure 3.5.

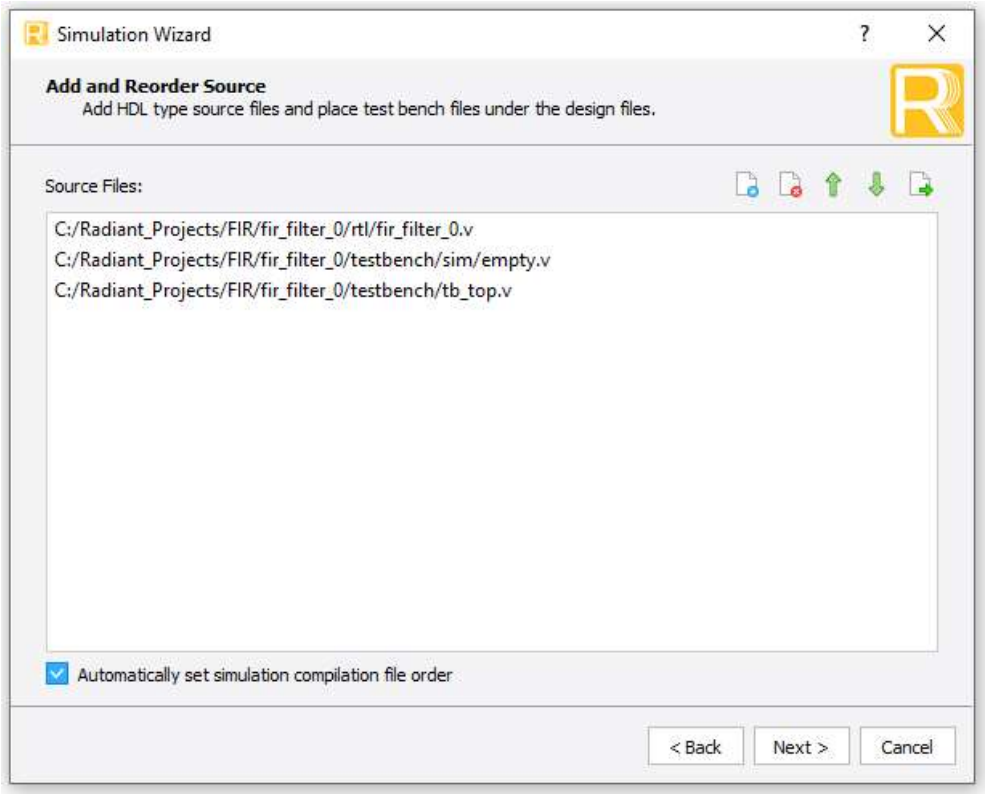

Figure 3.5. Adding and Reordering Source

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

   Note: It is necessary to follow the procedure above until it is fully automated in the Lattice Radiant Software Suite.

   The result of the simulation in our example is provided in Figure 3.6

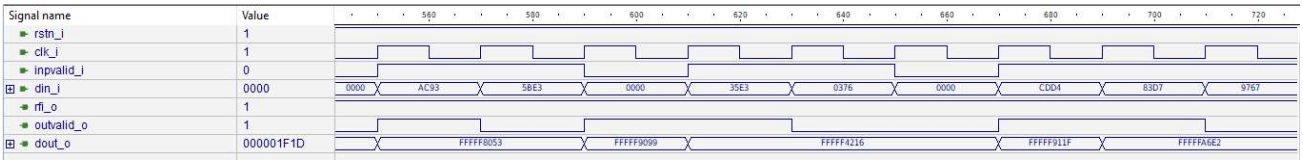

Figure 3.6. Simulation Waveform

3.4. Hardware Evaluation

The FIR Filter IP Core supports Lattice's IP hardware evaluation capability when used with LIFCL and LFD2NX devices. This makes it possible to create versions of the IP core that operate in hardware for a limited period of time (approximately four hours) without requiring the purchase of an IP license. It may also be used to evaluate the core in hardware in user-defined designs. The hardware evaluation capability may be enabled/disabled in the Strategy dialog box. It is enabled by default. To change this setting, go to Project > Active Strategy > LSE/Synplify Pro Settings.

4. Ordering Part Numbers

The Ordering Part Number (OPN) for this IP Core are the following:

- FIR-COMP-CNX-U – FIR Filter Generator for CrossLink-NX - Single Design License
- FIR-COMP-CNX-UT – FIR Filter Generator for CrossLink-NX - Site License
- FIR-COMP-CTNX-U – FIR Filter Generator for Certus-NX - Single Design License
- FIR-COMP-CTNX-UT – FIR Filter Generator for Certus-NX - Site License
- FIR-COMP-CPNX-U - FIR Filter Generator for CertusPro-NX - Single Design License
- FIR-COMP-CPNX-UT - FIR Filter Generator for CertusPro-NX - Site License
- FIR-COMP-XO5-U - FIR Filter Generator for MachXO5-NX - Single Design License
- FIR-COMP-XO5-UT - FIR Filter Generator for MachXO5-NX - Site License
- FIR-COMP-XO5-US - FIR Filter Generator for MachXO5-NX - 1 Year Subscription License

Appendix A. Resource Utilization

Table A.1 shows the resource utilization of the FIR Filter IP Core for the LFMXO5-25-9BBG400I device using Synplify Pro of Lattice Radiant software 3.1. Default configuration is used, and some attributes are changed from the default value to show the effect on the resource utilization.

Table A.1. Resource Utilization (LFMXO5-25-9BBG400I)

Configuration	Clk Fmax (MHz)[1]	Registers	LUTs	EBRs	DSPs
Default	200	60	12	0	16
Reloadable Coefficients: true, *Reorder Coefficients Inside*: true, Others = Default	200	650	32	0	16
Input Data Width: 18, *Precision Control (Overflow)*: Wrap-around, *Precision Control (Rounding)*: Rounding up, Others = Default	200	66	12	0	16
Synchronous Reset: true, *Clock Enable*: true, Others = Default	200	62	66	0	16
Data Memory Type: Distributed, *Coefficients Memory Type*: Distributed, *Output Buffer Type*: Distributed, Others = Default	200	60	12	0	16

Note:

1. Fmax is generated when the FPGA design only contains FIR Filter IP Core, and the target frequency is 200MHz. These values may be reduced when user logic is added to the FPGA design.

Table A.2 shows the resource utilization of the FIR Filter IP Core for the LFMXO5-25-7BBG400I device using Synplify Pro of Lattice Radiant software 3.1. Default configuration is used, and some attributes are changed from the default value to show the effect on the resource utilization.

Table A.2. Resource Utilization (LFMXO5-25-7BBG400I)

Configuration	Clk Fmax (MHz)[1]	Registers	LUTs	EBRs	DSPs
Default	200	60	12	0	16
Reloadable Coefficients: true, *Reorder Coefficients Inside*: true, Others = Default	191	650	34	0	16
Input Data Width: 18, *Precision Control (Overflow)*: Wrap-around, *Precision Control (Rounding)*: Rounding up, Others = Default	190	66	12	0	16
Synchronous Reset: true, *Clock Enable*: true, Others = Default	188	62	66	0	16
Data Memory Type: Distributed, *Coefficients Memory Type*: Distributed, *Output Buffer Type*: Distributed, Others = Default	200	60	12	0	16

Note:

1. Fmax is generated when the FPGA design only contains FIR Filter IP Core, and the target frequency is 200MHz. These values may be reduced when user logic is added to the FPGA design.

References

For complete information on Lattice Radiant Software Project-Based Environment, Design Flow, Implementation Flow and Tasks, as well as on the Simulation Flow, see the Lattice Radiant Software User Guide.

Technical Support Assistance

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

Revision History

Document Revision 1.2, Lattice Radiant SW Version 3.2, May 2022

Section	Change Summary
Introduction	Updated Table 1.1. FIR Filter Quick Facts: • Added MachXO5-NX to FPGA Families Supported • Added LFMXO5-25 to Targeted Devices • Added Table A.2. Resource Utilization (LFMXO5-25-7BBG400I) to Resources
IP Generation and Evaluation	Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure User Interface of FIR Filter IP Core and Figure 3.3. Check Generated Result.
Ordering Part Numbers	Added the following part numbers: • FIR-COMP-XO5-U - FIR Filter Generator for MachXO5-NX - Single Design License • FIR-COMP-XO5-UT - FIR Filter Generator for MachXO5-NX - Site License • FIR-COMP-XO5-US - FIR Filter Generator for MachXO5-NX - 1 Year Subscription License
Appendix A. Resource Utilization	Updated resource utilization for *LFMXO5-25-9BBG400I* and *LFMXO5-25-7BBG400I*.

Document Revision 1.1, Lattice Radiant SW Version 3.0, June 2021

Section	Change Summary

Introduction	• Removed last paragraph.
	• Updated Table 1.1. FIR Filter Quick Facts.
	  • Revised Supported FPGA Families
	  • Revised Targeted Devices
	  • Revised Lattice Implementation.

Ordering Part Numbers	Added part numbers.

Document Revision 1.0, Lattice Radiant SW Version 2.2, October 2020

Section	Change Summary
All	Initial release.

www.latticesemi.com