Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

LATTICE SEMICONDUCTOR

LatticeCORE

FIR Filter IP Core User's Guide

April 2014
IPUG79_01.4

Table of Contents

The Lattice FIR (Finite Impulse Response) Filter IP core is a widely configurable, multi-channel FIR filter, implemented using high performance sysDSP™ blocks available in Lattice devices. In addition to single rate filters, the IP core also supports a range of polyphase decimation and interpolation filters. The utilization versus throughput trade-off can be controlled by specifying the multiplier multiplexing factor used for implementing the filter. For example, setting the multiplier multiplexing factor to the maximum value supported by the GUI results in the best resource utilization, whereas setting the multiplier multiplexing factor to 1 results in the best throughput. The FIR Filter IP core supports up to 256 channels, with each having up to 2048 taps.

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.

The FIR Filter IP core can also be generated using the Lattice FIR Filter Simulink® Model. For information on the Simulink flow, refer to the FPGA Design with ispLEVER tutorial.

Quick Facts

Table 1-1 through Table 1-2 give quick facts about the FIR Filter IP core for LatticeXP2™, LatticeECP3™, and ECP5™ devices.

Table 1-1. FIR Filter IP Core for LatticeXP2 Devices Quick Facts

		FIR IP Configuration		
		4 Channels 64Taps 1 Multiplier	**1 Channel 32 Taps 32 Multipliers**	**1 Channel 32 Taps 8 Multipliers**
Core Requirements	FPGA Families Supported	LatticeXP2		
	Minimal Device Needed	LFXP2-5E	LFXP2-40E	LFXP2-8E
Resource Utilization	Targeted Device	LFXP2-40E-7F672C		
	LUTs	96	311	163
	sysMEM EBRs	1	0	8
	Registers	114	322	252
	MULT18X18	1	32	8
Design Tool Support	Lattice Implementation	Lattice Diamond® 3.2		
	Synthesis	Synopsys® Synplify Pro® for Lattice 2013.09L-SP1		
	Simulation	Aldec® Active-HDL™ 9.2 Lattice Edition		
		Mentor Graphics® ModelSim® SE 6.3F		

Table 1-2. FIR Filter IP Core for LatticeECP3 Devices Quick Facts

		FIR IP Configuration		
		4 Channels 64 Taps 1 Multiplier	**1 Channel 32 Taps 32 Multipliers**	**1 Channel 32 Taps 8 Multipliers**
Core Requirements	FPGA Families Supported	Lattice ECP3		
	Minimal Device Needed	LFE3-35EA		
Resource Utilization	Targeted Device	LFE3-70E-8FN672CES		
	LUTs	150	48	132
	sysMEM EBRs	2	0	8
	Registers	174	98	432
	DSP Slice	2	16	5
Design Tool Support	Lattice Implementation	Lattice Diamond 3.3		
	Synthesis	Synopsys Synplify Pro for Lattice D-2013.09L-SP1		
	Simulation	Aldec Active-HDL 9.2 Lattice Edition		
		Mentor Graphics ModelSim SE 6.3F		

Table 1-3. FIR Filter IP Core (Low-Speed Mode) for ECP5 Devices Quick Facts

		FIR IP Configuration		
		4 Channels 64 Taps 1 Multiplier	**1 Channel 32 Taps 32 Multipliers**	**1 Channel 32 Taps 8 Multipliers**
Core Requirements	FPGA Families Supported	ECP5		
	Minimal Device Needed	LFE5U-45FEA		
Resource Utilization	Targeted Device	LFE5U-85F-8MG756I	LFE5U-85F-8MG756I	LFE5U-85F-8MG756I
	LUTs	143	47	133
	sysMEM EBRs	2	0	8
	Registers	174	98	432
	DSP Slice	2	16	5
Design Tool Support	Lattice Implementation	Lattice Diamond 3.2		
	Synthesis	Synopsys Synplify Pro F-2013.09L-beta		
	Simulation	Aldec Active-HDL 9.2 Lattice Edition		
		Mentor Graphics ModelSim SE PLUS 6.3f		

Table 1-4. FIR Filter IP Core (High-Speed Mode) for ECP5 Devices Quick Facts

		FIR IP Configuration		
		1 Channels 64 Taps 16 Multipliers	**1 Channel 24 Taps 6 Multipliers**	**1 Channel 48 Taps 12 Multipliers**
Core Requirements	FPGA Families Supported	ECP5		
	Minimal Device Needed	LFE5U-45FEA		
Resource Utilization	Targeted Device	LFE5U-85F-8MG756I	LFE5U-85F-8MG756I	LFE5U-85F-8MG756I
	LUTs	737	372	633
	sysMEM EBRs	24	9	18
	Registers	1018	643	929
	DSP Slice	10	5	8
Design Tool Support	Lattice Implementation	Lattice Diamond 3.2		
	Synthesis	Synopsys Synplify Pro F-2013.09L-beta		
	Simulation	Aldec Active-HDL 9.2 Lattice Edition		
		Mentor Graphics ModelSim SE PLUS 6.3f		

In high-speed mode, for partially-folded/full-folded cases, 1 channel means I,Q channels.

Features

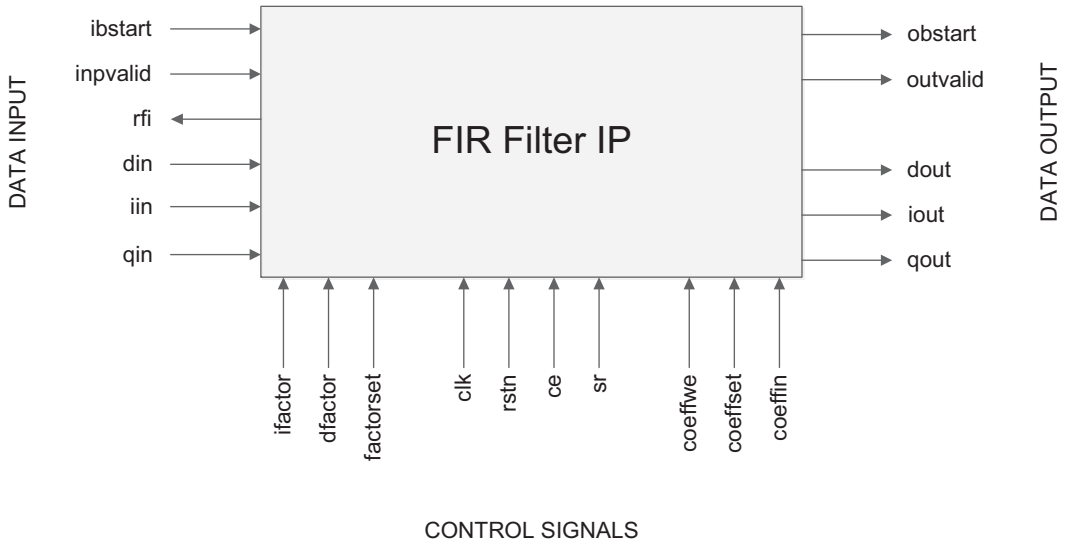
- Variable number of taps up to 2048
- Input and coefficients widths of 4 to 32 bits
- Multi-channel support for up to 256 channels
- Decimation and Interpolation ratios from 2 to 256
- Support for half-band filter
- Configurable parallelism from fully parallel to serial
- Signed or unsigned data and coefficients
- Coefficients symmetry and negative symmetry optimization
- Re-loadable coefficients support
- Full precision arithmetic
- Selectable output width and precision
- Selectable overflow: wrap-around or saturation
- Selectable rounding: truncation, round towards zero, round away from zero, round to nearest and convergent rounding
- Width and precision specified using fixed point notations
- Handshake signals to facilitate smooth interfacing
- In ECP5, support high-speed. For low speed, support for half-band filter

This chapter provides a functional description of the FIR Filter IP core.

Interface Diagram

The top-level interface diagram for the FIR Filter IP core is shown in **Figure 2-1**.

Figure 2-1. Top-Level Interface for the FIR Filter IP Core



FIR Filter 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.

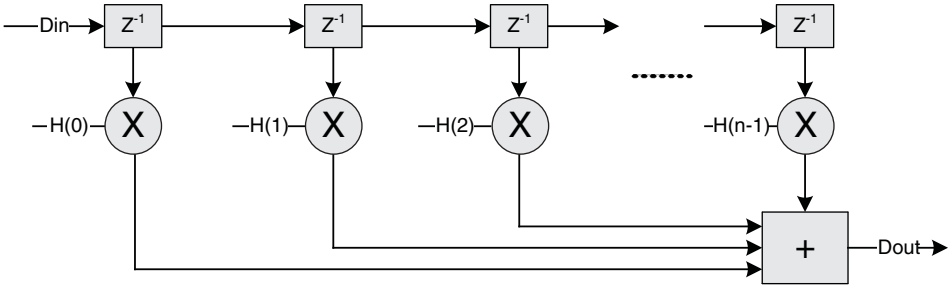
$$y_n = \sum_{i=0}^{N-1} x_{n-i} h_i = x_n h_0 + x_{n-1} h_1 + \ldots + x_{n-N+1} h_{N-1} \quad (1)$$

In the above equation hn, n=0,1,…,N-1 is the impulse response, x_n, n=0,1,…,×, is the input and y_n, n=0,1,…,×, is the output. The number of delay elements (N-1) represents the order of the filter. The number of input data samples (current and previous) used in the calculation of one output sample represents the number of filter taps (N).

Direct-form Implementation

In the direct-form implementation shown in **Figure 2-2**, the input samples will be shifted into a shift register queue and each shift register is connected to a multiplier. The products from the multipliers are summed to get the FIR filter's output sample.

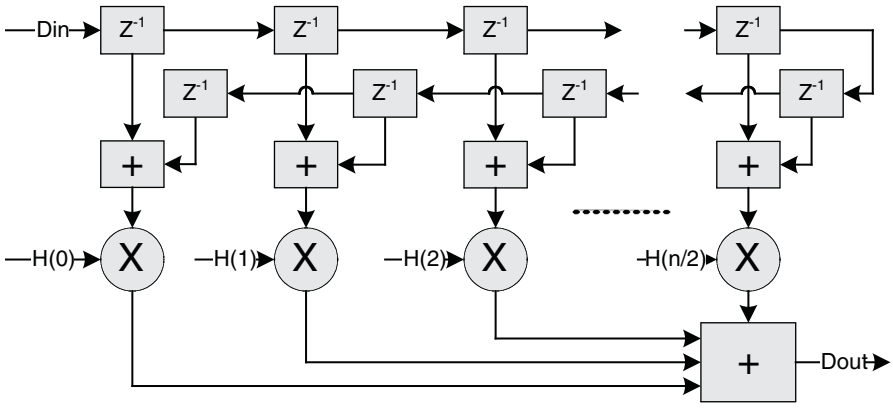
Figure 2-2. Direct-form FIR Filter



Symmetric Implementation

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 2-3**.

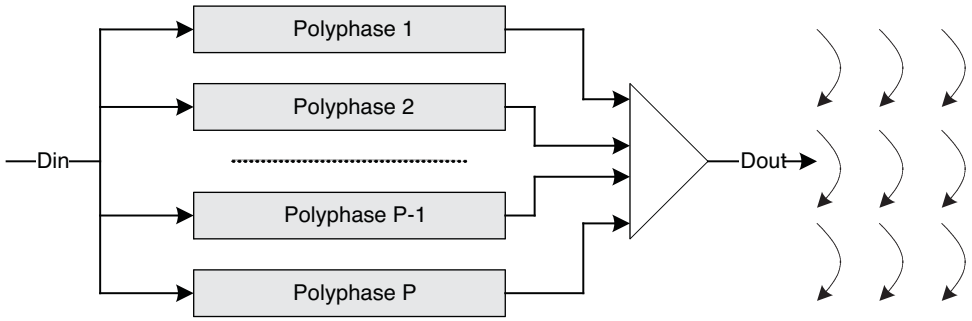
Figure 2-3. Symmetric Coefficients FIR Filter Implementation



Polyphase Interpolation FIR Filter

The polyphase interpolation filter option implements the computationally efficient 1-to-P interpolation filter shown below where P is an integer greater than 1. **Figure 2-4** shows a polyphase interpolator, where each branch is referred to as a polyphase.

Figure 2-4. Polyphase Interpolator

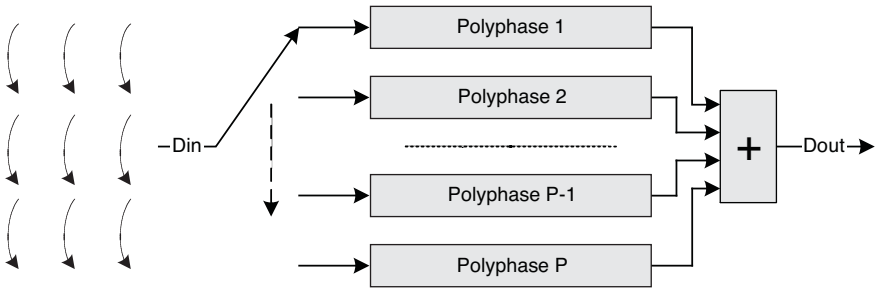


In this structure, the input data will be loaded into each poly-phase at the same time and the output data of each polyphase will be unloaded as an output sample of the FIR. The number of polyphases is equal to the interpolation factor. The coefficients are assigned to all polyphases evenly.

Polyphase Decimation FIR Filter

The polyphase decimation filter option implements the computationally efficient P-to-1 decimation filter shown in Figure 2-5, where P is an integer greater than 1.

Figure 2-5. Polyphase Decimator



In this structure, the input sample is loaded sequentially into each of the polyphases with only one polyphase fed at a time. When all the polyphases are loaded with a sample, the result from the polyphases are summed and unloaded as the FIR filter's output. In this scheme, P input samples generate one output sample, where P is the decimation factor.

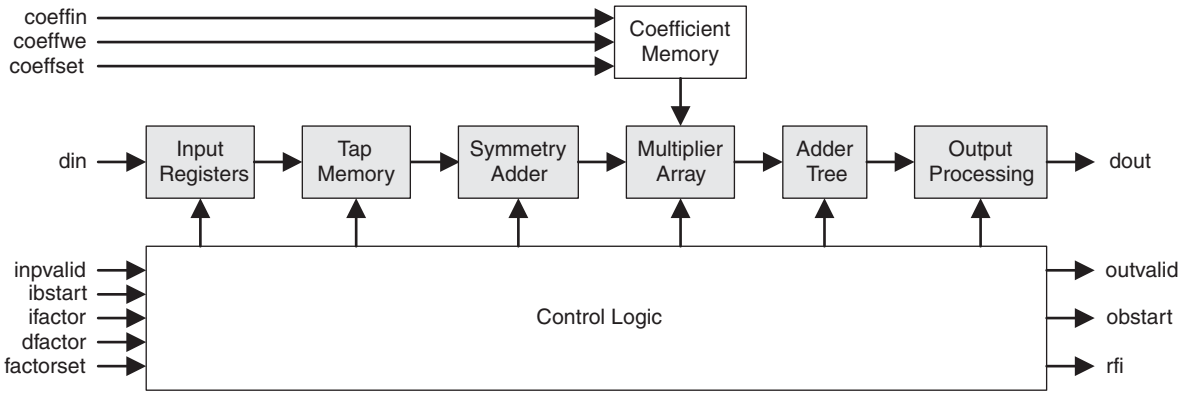
Multi-channel FIR Filters

It is very common to see FIR filters used in multi-channel processing scenarios. The maximum possible throughput of a FIR filter implementation is often much higher than the throughput required for a single channel being processed. For such applications, it is desirable to use the same resources in a time multiplexed way to realize multi-channel FIR filters. Except in fully parallel implementations, where enough multipliers are used to perform all the necessary computations in one clock cycle, the FIR filter uses independent tap and coefficient memories to feed each multiplier. Hence, multi-channel implementations result in lower memory usage compared to multiple instantiations of FIR filters. For cases, where all the channels use the same coefficient set, using a multi-channel FIR filter has the clear advantage of requiring smaller coefficient memories.

Implementation Details

Figure 2-6 shows the functional block diagram of the FIR Filter IP core.

Figure 2-6. Functional Block Diagram of the FIR Filter IP Core

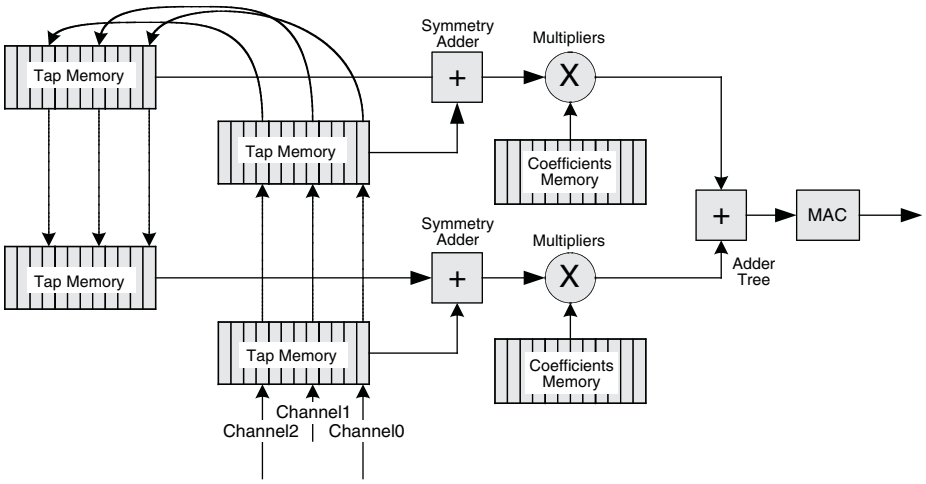


The data and coefficients are stored in different memories shown as tap memory and coefficients memory in the above diagram. The symmetry adder is used if the coefficients are symmetric. The multiplier array contains one or more multipliers depending on the user specification. The adder tree performs the sum of products. Depending on the configuration, the adder tree, or a part of it, is implemented inside DSP blocks. The output processing block performs the output width reduction and precision control. This block contains logic to support different types of

rounding and overflow. The block labeled “Control Logic” manages the scheduling of data and arithmetic operations based on the type of filter (interpolation, decimation or multi-channel) and multiplier multiplexing.

The tap and coefficient memories are managed differently for different configurations of the FIR filter. **Figure 2-7** shows the memory assignments for a 16-tap, 3-channel, symmetric FIR filter with two multipliers.

Figure 2-7. Tap and Coefficient Memory Management for a Sample FIR Filter



In the diagram, there are two tap memories and a coefficient memory for each multiplier. The depth of each memory is ceil(taps/multiplier)*channel, where the operator, ceil(x), returns the next higher integer if the argument x is fractional. The memory depth is 12 in this example.

Configuring the FIR Filter IP Core

Architecture Options

The options for number of channels, number of taps and filter type are independent and directly specified in the **Architecture** tab of the IP core GUI (see “Parameter Settings” on page 21 for details). If a polyphase decimator or interpolator is required, the decimation or interpolation factor can be directly specified in the GUI. The decimation or interpolation factor can also be specified through input ports during operation by selecting the corresponding **Variable** option. If the **Variable decimation** (or **Variable interpolation**) factor option is selected, the decimation (or interpolation) factor can be varied from two to **Decimation factor** (or **Interpolation factor**) through the input port.

Coefficients Specification

The coefficients of the filter are specified using a coefficients file. The coefficients file is a text file with one coefficient per line. If the coefficients are symmetric, the check box **Symmetric Coefficients** must be checked so the IP core uses symmetry adders to reduce the number of multipliers used. If the **Symmetric Coefficients** box is checked, only one-half of the coefficients are read from the coefficient file. For an n-tap symmetric coefficients filter, the number of coefficients read from the coefficients file is equal to ceil(n/2). For multi-channel filters, the coefficients for channel 0 are specified first, followed by those for channel 1, and so on. For multi-channel filters, there is an option to specify whether the coefficients are different for each channel or the same (common) for all the channels. If the coefficients are common, only one set of coefficients needs to be specified in the coefficients file. The coefficient values in the file can be in any radix (decimal, hexadecimal or binary) selected by the user. A unary negative operator is used only if the coefficients are specified in decimal radix. For hexadecimal and binary radices, the numbers must be represented in twos complement form. An example coefficients file in decimal format for an 11-tap, 16-bit coefficients set is given below. In this example, the coefficients binary point is 0.

-556
-706
-857
-419
1424
5309
11275
18547
25649
30848
32758

An example coefficients file in floating point format for the above case when the Coefficients binary point position is 8, is given below. The coefficients will be quantized to conform to the 16.8 fractional number, in which 16 is the full width of the coefficients and 8 is the width of the fractional part.

-2.1719
-2.7578
-3.3477
-1.6367
5.5625
20.7383
44.043
72.45
100.0191
120.5
127.96

If the check box **Reloadable Coefficients** is checked, the coefficients can be reloaded to the FIR filter during the operation of the core. With this option, the desired coefficients must be loaded before the operation of the filter. The coefficients must be loaded in a specific order that is determined by the program supplied with the IP core. The IP core can also optionally do the reordering internally, albeit using more resources. If this option is desired, the check box **Reorder Coefficients Inside** can be checked. With this option, the coefficients can be loaded in the normal sequential order to the core.

Multiplier Multiplexing Factor

The throughput and the resource utilization can be controlled by assigning a proper value to the **Multiplier Multiplexing Factor** parameter. Full parallel operation (one output data per clock cycle) can be achieved by setting the **Multiplier Multiplexing Factor** to 1. If the **Multiplier Multiplexing Factor** is set to the maximum value displayed in the GUI, full series operation is supported and it takes up to n clocks to compute one output data sample, where n is the number of taps for a non-symmetric FIR filter and half the number of taps for a symmetric FIR filter. The maximum value of the **Multiplier Multiplexing Factor** for different configurations of an n-tap FIR filter is given in Table 2-1.

Table 2-1. Maximum Multiplier Multiplexing Factor for Different Configurations[1]

FIR Type	Single Rate	Interpolator with Factor=i	Decimator with Factor=d
Non-symmetric	n	Ceil(n/i)	Ceil(n/d)
Symmetric	Ceil(n/2)	Ceil(n/2i)	Ceil(n/2d)
Half-band	floor((n+1)/4)+1	floor((n+1)/4)	floor((n+1)/8)+1

1. The operator floor (x) returns the next lower integer, if x is a fractional value.

I/O Specification Options

The controls in the I/O Specifications GUI tab are used to define the various widths and precision methods in the data path. The width and binary point positions of the input data and coefficients can be defined independently. From the input data width, coefficient width and the number of taps, the full precision output width and true location of the output binary point automatically get fixed. The full precision output is converted to user specified output width by dropping some least significant (LS) and some most significant (MS) bits and by performing the specified rounding and overflow processing. The output is specified by the output width and the output binary point position parameter.

Rounding

The following five options are supported for rounding:

- **None** – Discards all bits to the right of the output least significant bit and leaves the output uncorrected.
- **Rounding up** – Rounds to nearest more positive number.
- **Rounding away from zero** – Rounds away from zero if the fractional part is exactly one-half.
- **Rounding towards zero** – Rounds towards zero if the fractional part is exactly one-half.
- **Convergent rounding** – Rounds to the nearest even value if the fractional part is exactly one-half.

Implementation Options

Memory Type

The FIR Filter IP core uses memories for storing delay tap data, coefficients and for some configurations, input or output data. The number of memory units used depends on several parameters including data width, number of taps, filter type, number of channels and coefficient symmetry. In most cases, each multiplier requires one data memory unit and one coefficient memory unit. Interpolation or decimation filters may additionally use input or output buffers. The memory type GUI option can be used to specify whether EBR or distributed memory is used for data, coefficient, input and output storage. The option called “Auto” leaves that choice to the IP generator tool, which uses EBR if the memory is deeper than 128 locations and distributed memory otherwise.

Signal Descriptions

A description of the Input/Output (I/O) ports for the FIR Filter IP core is provided in **Table 2-2**. The top-level interface diagram for the FIR Filter IP core is shown in **Figure 2-1**.

Table 2-2. Top-Level Port Definitions

Port	Bits	I/O	Description
General I/Os			
clk	1	I	System clock for data and control inputs and outputs.
rstn	1	I	System wide asynchronous active-low reset signal.
din	Input data width	I	Input data.
iin	Input data width	I	I channel input data for High Speed mode. In single channel high speed mode, interleaved data feed into the core from iin and qin, while data on iin is d0,d2,d4,d6... and d1,d3,d5,d7 are on qin. In multi-channel high speed mode, iin and qin are data for different channel.
qin	Input data width	I	Q channel input data for High Speed mode. As above.
inpvalid	1	I	Input valid signal. The input data is read-in only when inpvalid is high.
dout	Output width	O	Output data for Low speed mode.
iout	Output width	O	Output data for high speed mode, in single channel mode, interleaved data out from the core, while d0,d2,d4,d6 are on iout, and d1,d3,d5,d7 are on qout. In mult-channel mode, data on iout and qout are for different channel.
qout	Output width	O	Q channel output data in high speed mode.

Table 2-2. Top-Level Port Definitions

Port	Bits	I/O	Description
outvalid	1	O	Output data qualifier. Output data dout is valid only when this signal is high.
rfi	1	O	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 was high during the previous clock cycle.
When Reloadable coefficients is selected			
coeffin	Notes 1	I	Coefficients input. The coefficients have to be loaded through this port in a specific order. Refer to the section “Interfacing with the FIR Filter IP core” for details.
coeffwe	1	I	When asserted, the value on bus coeffin will be written into coefficient memories.
coeffset	1	I	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 and coeffwe.
When Number of channels is greater than 1			
ibstart	1	I	Input block start. For multi-channel configurations, this input identifies channel 0 of the input.
obstart	1	O	Output block start. For multi-channel configurations, this output identifies channel 0.
When Variable interpolation factor or Variable decimation factor is checked			
ifactor	ceil(Log2(Interpolation factor+1))	I	Interpolation factor value
dfactor	ceil(Log2(Decimation factor+1))	I	Decimation factor value
factorset	1	I	Sets the interpolation factor or the decimation factor.
Optional I/Os			
ce	1	I	Clock Enable. While this signal is de-asserted, the core will ignore all other synchronous inputs and maintain its current state
sr	1	I	Synchronous Reset. When asserted for at least one clock cycle, all the registers in the IP core are initialized to reset state.

Notes 1
a. Width for signed type and symmetric interpolation is Coefficients width +1.
b. Width for unsigned and symmetric interpolation is Coefficients width +2.
c. Width for all other cases is Coefficients width.

Interfacing with the FIR Filter IP core

Data interface

In low speed mode, data is feed into the core through din and out from the core through dout. In high speed mode, data is feed into the core through iin&qin while out from the core through iout&qout, and in single channel mode, the data on iin&qin/iout&qout is interleaved.

Multiple Channels

For multi-channel implementations, two ports, ibstart and obstart, are available in the IP core to synchronize the channel numbers. The input ibstart is used to identify channel 0 data applied at the inputs. The output obstart goes high simultaneously with channel 0 output data.

Variable Interpolation/ Decimation Factor

When the interpolation (or decimation) factor is variable, the ports ifactor (or dfactor) and factorset are added to the IP core. The interpolation (or decimation) factor applied on the port ifactor (or dfactor) is set when the strobe signal factorset is high. When the interpolation (or decimation) factor changes, the output rfi goes low for a few cycles. When it becomes high again, the filter performs as an interpolating (or decimating) filter corresponding to the new factor value.

Reloadable Coefficients

When **Reloadable Coefficients** is selected, the two added ports, *coeffin* and *coeffwe*, are used to reload the coefficients. All the coefficients need to be loaded in one batch, while keeping the signal *coeffwe* high during the entire duration of loading. After all the coefficients are loaded, the input signal *coeffset* must be pulsed high for one clock cycle for the new coefficients to take effect.

There are two ways in which coefficients can be applied for reloading the coefficients memory, as specified by the **Reorder Coefficients Inside** parameter.

When **Reorder Coefficients Inside** is not selected, the coefficients have to be applied in a particular sequence for reloading the coefficients memory. The raw coefficients, as specified in the coefficients file, can be converted to the reloadable sequence by using the coefficients generation program coeff_gen.exe (for Windows) available under the “gui” folder in the IP installation directory (for example, under the `C:\LatticeCore\fir_core_v4.2\gui` folder). The names of the coefficient generation program for UNIX and Linux are *coeff_gen_s* and *coeff_gen_l* respectively. For Windows, the program is invoked as follows:

```
coeff_gen.exe  <IP_file_name>.lpc
```

Note: If in lpc file, the value of parameter "varcoeff=" is "Yes", please change it to "No" before generating ROM files manually.

This command converts the coefficients in the input file[1], as referred by the *coefffile=* parameter in the lpc file, to the loadable coefficients sequence file called *coeff.mem*. Note that the output file may contain more coefficients than there originally were due to inserted zero coefficients. All the coefficients in the output file, including the zeros, have to be applied sequentially through the *coeffin* port. To obtain the sequence of application of coefficients, edit the input coefficients file with sequential numbers 1,2,… and run the *coeff_gen.exe* command. In the reloadable coefficients mode, the core will not be ready for operation (the *rfi* output will not be high) until the coefficients are loaded and *coeffset* is asserted high.

When the parameter **Reorder Coefficients Inside** is selected, the coefficients will be reordered inside the IP core without requiring manual reordering described previously. With this option, reordering logic is added to the IP core and the user can apply the coefficients in the normal sequence.

With this capability, if the parameter **Symmetric Coefficients** is selected, only half of the coefficients provided will be used. For example, if the raw coefficient input sequence is: "1 2 3 4 5 6 5 4 3 2 1," the coefficients that will be used will be "1 2 3 4 5 6."

Similarly, if **Half Band** is selected with the this capability, all of the input coefficients in even locations, except the last one, will be discarded. For example, if the raw coefficient input sequence is: "1 0 2 0 3 0 4 0 5 6 5 0 4 0 3 0 2 0 1," the coefficients that will be used will be "1 2 3 4 5 6."

1. If the parameter *varcoeff=* in the lpc file is set to "yes", change it to "no" before generating the new coefficients file.

Timing Specifications

Timing diagrams for the FIR Filter IP core are given in **Figure 2-8** through **Figure 2-21**. Note that there are different timing specifications for certain FIR filter applications using LatticeECP3 devices. **Figures 2-8** through **2-11** apply to all FIR applications. **Figures 2-12** through **2-14** apply to all FIR applications using LatticeXP2 devices and the following applications using LatticeECP3 devices: negative symmetry, half band, factor variable interpolation and decimation and applications using 36x36 multipliers. **Figures 2-15** through **2-17** apply to all other LatticeECP3 applications. Figure 2-18 through 2-21 apply to High Speed mode using ECP5.

Timing Specifications Applicable to All Devices

Figure 2-8. Single Channel, Single Rate FIR Filter with Continuous Inputs

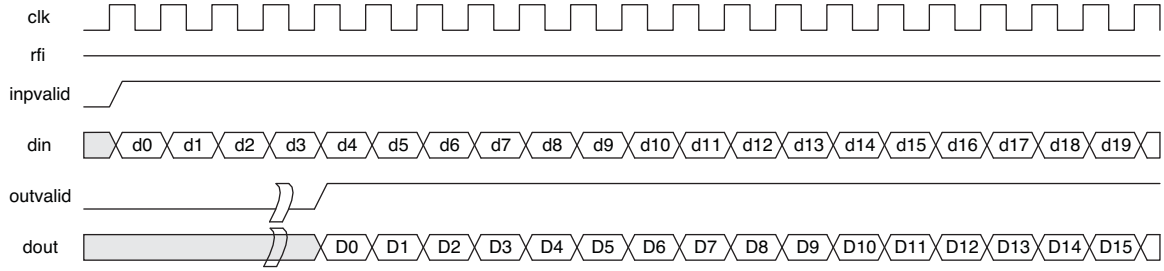


Figure 2-9. Single Channel, Single Rate FIR Filter with Gaps in Input

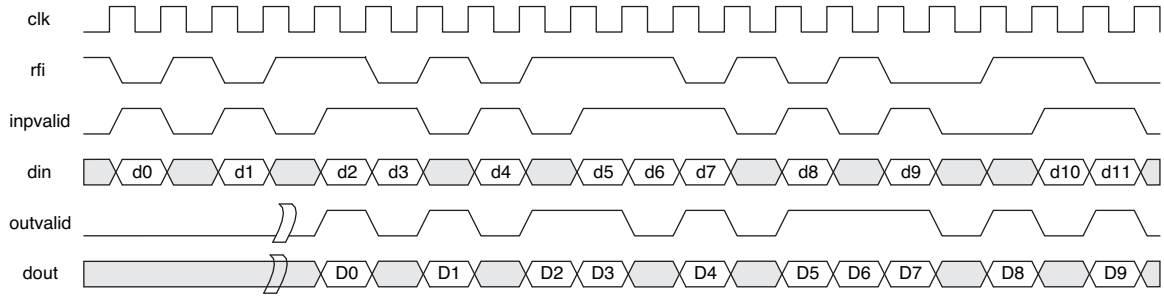


Figure 2-10. Factorset Signals

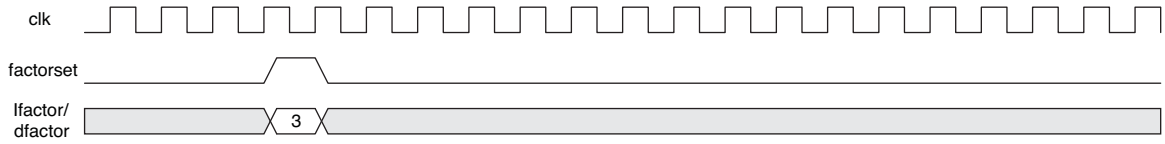
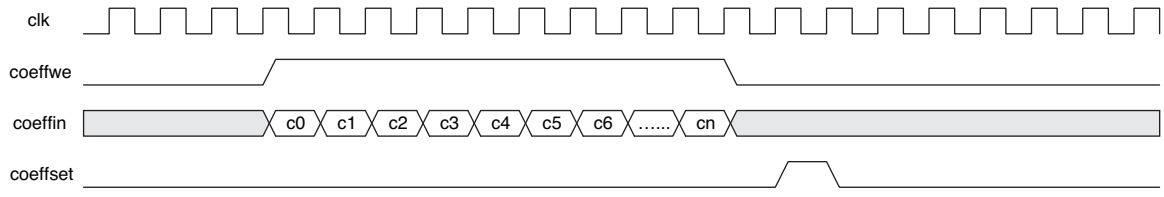


Figure 2-11. Coefficient Reloading



Timing Specifications Applicable to LatticeXP2 and Certain LatticeECP3 Implementations

As indicated previously, **Figures 2-12** through **2-14** apply to all FIR applications using LatticeXP2 devices and the following applications using LatticeECP3 devices: negative symmetry, half band, factor variable interpolation and decimation and applications using 36x36 multipliers.

Figure 2-12. Multi-Channel Single Rate FIR Filter (3 Channels)

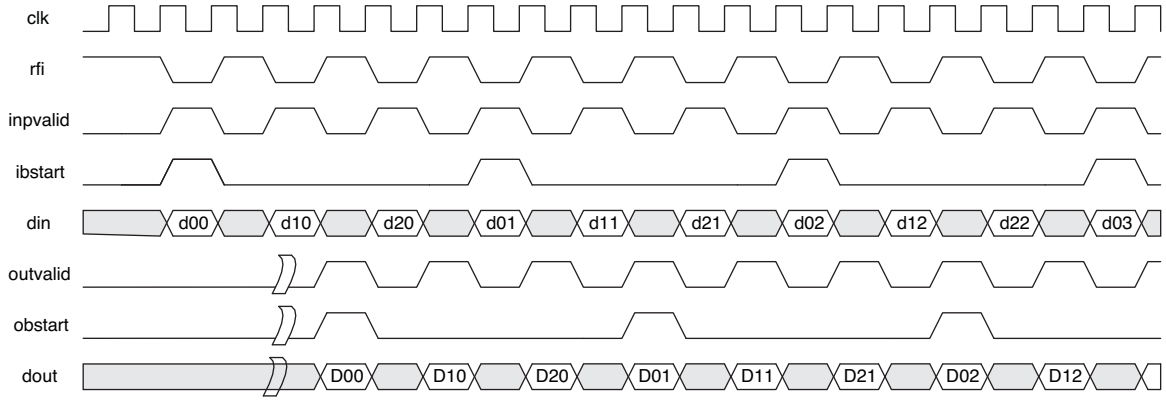


Figure 2-13. Multi-Channel (3 Channels) Interpolator (Factor of 3)

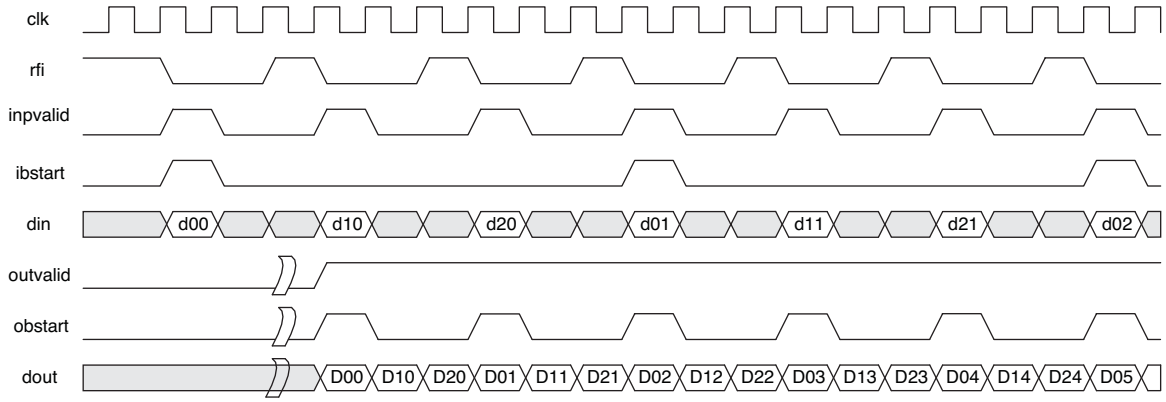
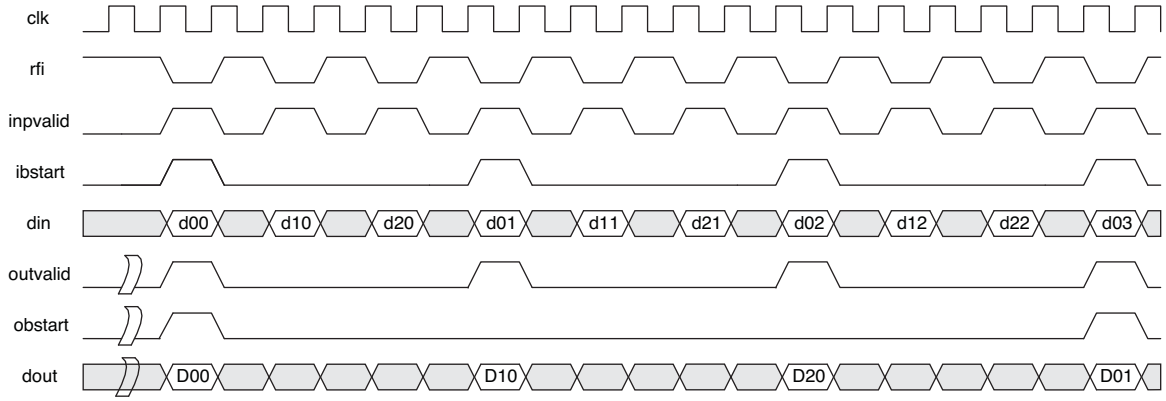


Figure 2-14. Multi-Channel (3 Channels) Decimator (Factor of 3)



Timing Specifications Applicable to Certain LatticeECP3 Implementations

As indicated previously, **Figures 2-15** through **2-17** apply to all LatticeECP3 applications other than those specifically listed in the previous section.

Figure 2-15. Multi-Channel Single Rate FIR Filter (3 Channels)

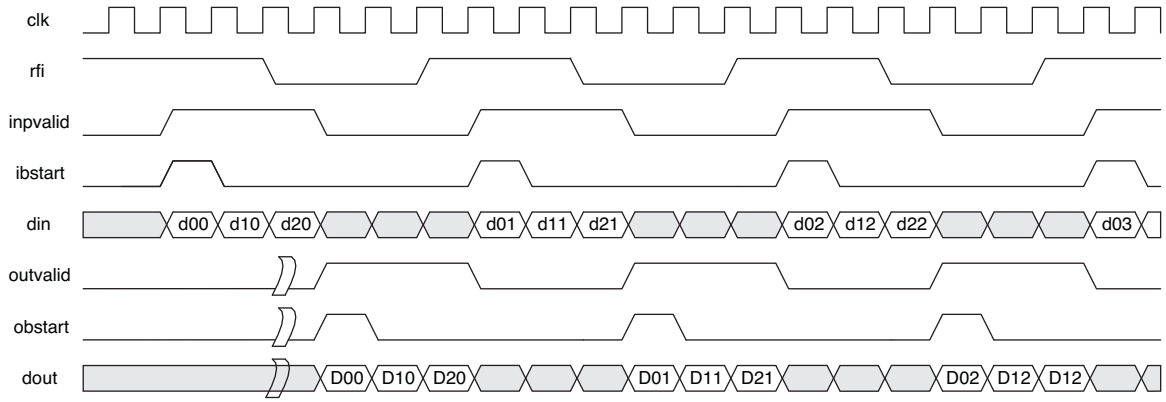


Figure 2-16. Multi-Channel (3 Channels) Interpolator (Factor of 3)

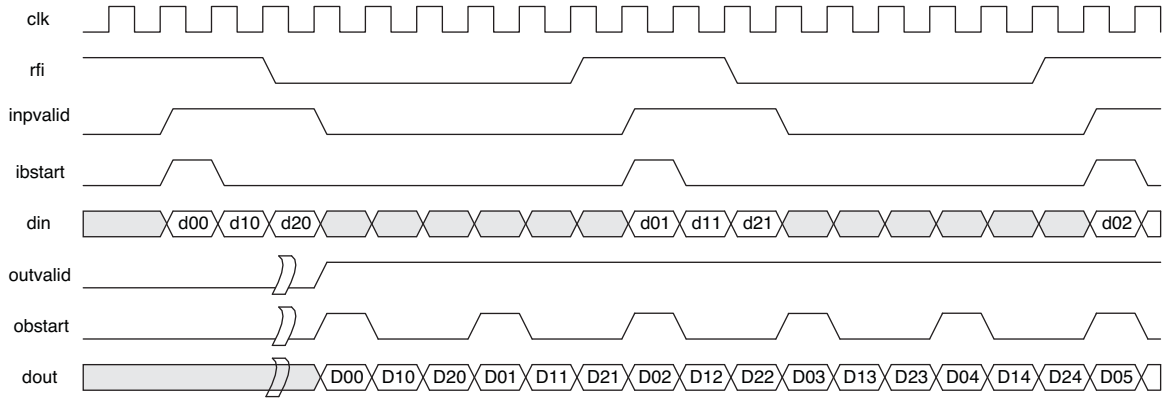
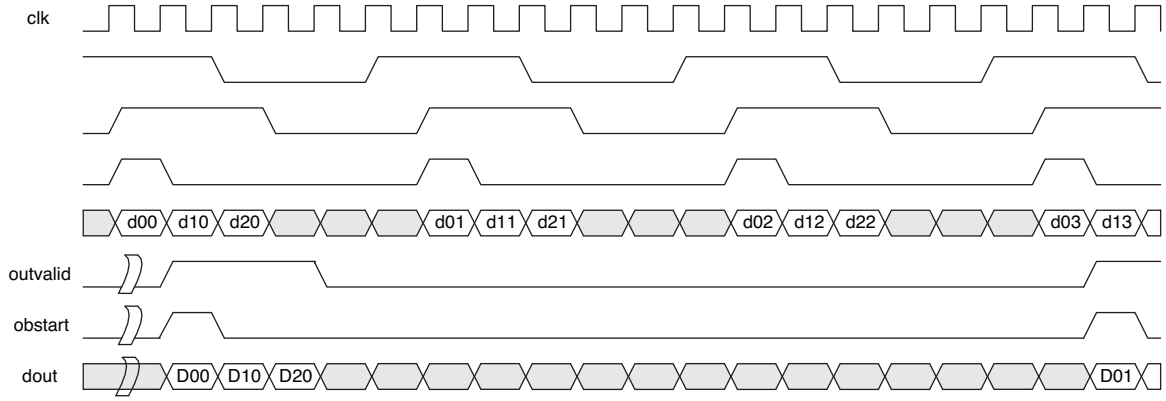


Figure 2-17. Multi-Channel (3 Channels) Decimator (Factor of 3)



Timing specifications Applicable to ECP5 Implementations

As indicated previously, **Figures 2-18** through **2-21** apply to ECP5 High speed mode application.

Figure 2-18. Single Channel in High Speed Mode

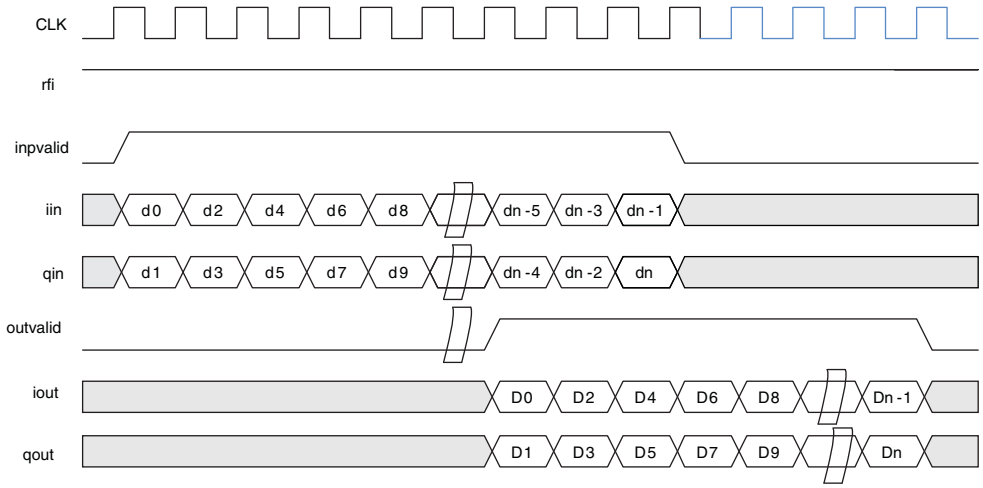


Figure 2-19. Multi-Channel (4 Channels) in High Speed Mode

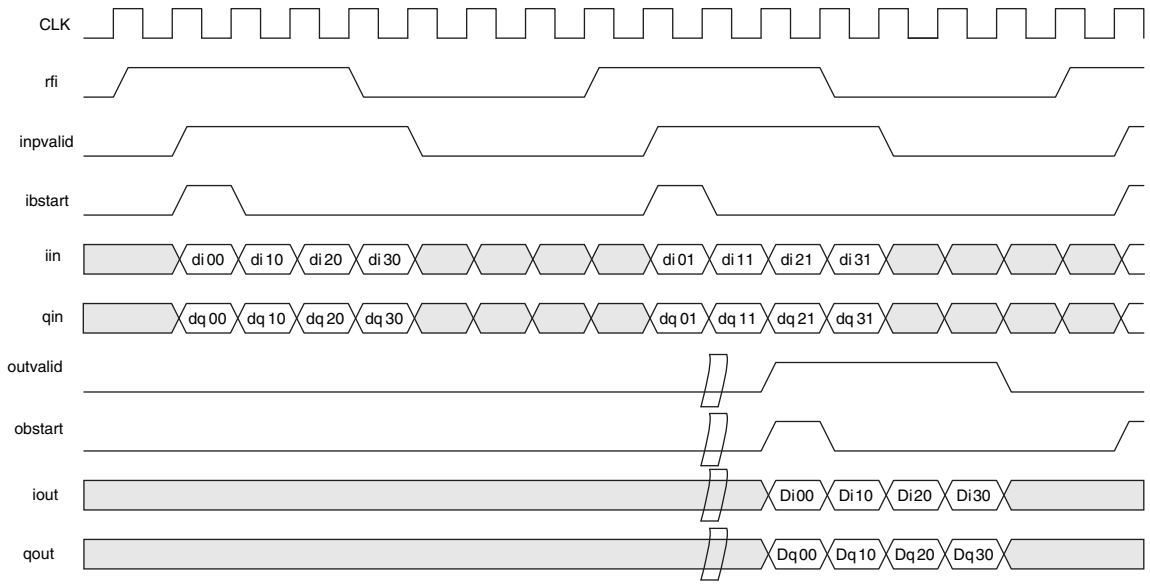


Figure 2-20. Multi-Channel (2 Channels) DHIGHSPEED I&Q Polyphase-Interpolator (1:4)

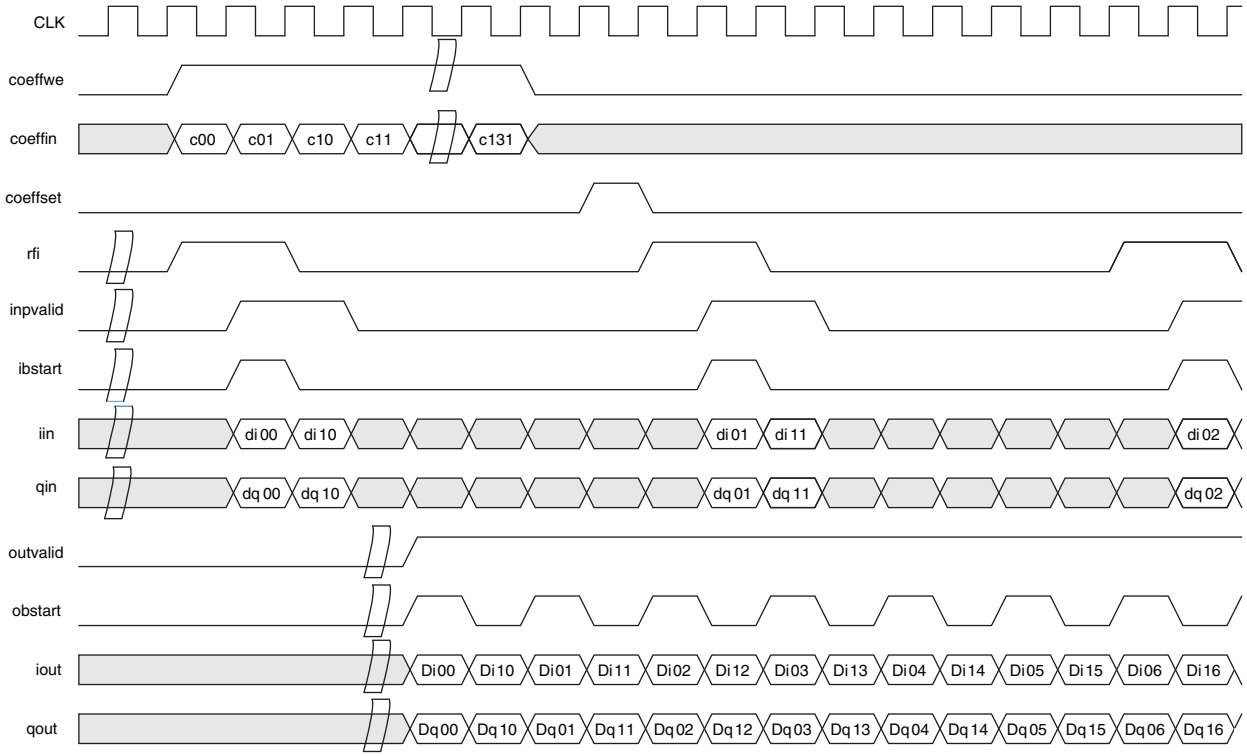
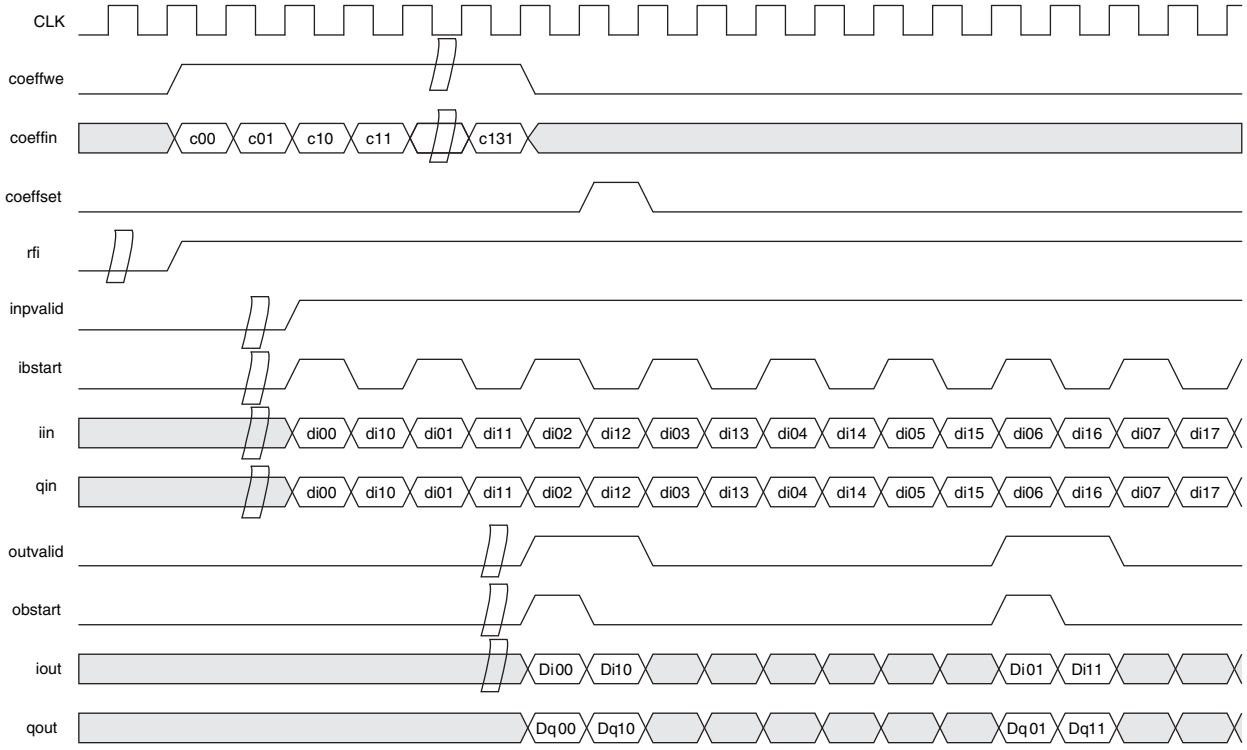


Figure 2-21. Multi-Channel (2 Channels) DHIGHSPEED I&Q Polyphase-Decimator (4:1)



Parameter Settings

The IPexpress and Clarity Designer tools are used to create IP and architectural modules in the Diamond software. Refer to "IP Core Generation and Evaluation" on page 29 for a description on how to generate the IP.

Table 3-1 provides the list of user configurable parameters for the FIR Filter IP core The parameter settings are specified using the FIR Filter IP core Configuration GUI in IPexpress or Clarity Designer. The numerous FIR Filter IP core parameter options are partitioned across multiple GUI tabs as described in this chapter.

Table 3-1. Parameter Specifications for the FIR Filter IP Core

Parameter	Range	Default
Filter Specifications		
Number of channels	1 to 256	4
Number of taps	1 to 2048	64
Filter type	{Single rate, Interpolator, Decimator}	Single rate
Interpolation factor	2 to 256	2
Variable interpolation factor	{Yes, No}	No
Decimation factor	2 to 256	2
Variable decimation factor	{Yes, No}	No
Single channel in High Speed	{Yes, No}	No
Multi-channel in High Speed	{Yes, No}	No
Coefficients Specifications		
Reloadable coefficients	{Yes, No}	Yes
Reorder coefficients inside	{Yes, No}	No
coefficients set	{Common, One per channel}	Common
Symmetric coefficients	{Yes, No}	No
Negative symmetry	{Yes, No}	No
Half band	{Yes, No}	No
Coefficient radix	{Floating point, Decimal, Hex, Binary}	Decimal
Coefficients file	Type or Browse	-
Advanced Options		
Multiplier Multiplexing factor	Note 2, Note 3	Note 3
Number of SysDSP blocks in a row	5^4	Note 4
I/O Specifications		
Input data type	{Signed, Unsigned}	Signed
Input data width	4 to 32	16
Input data binary point position	-2 to Input data width + 2	0
Coefficients type	{Signed, Unsigned}	Signed
Coefficients width	4 to 32	16
Coefficients binary point position	-2 to Coefficients width + 2	0
Output width	4 to Max Output Width	38
Output binary point position	(4+Input data binary point position + coefficient binary point position – Max output width) to (Output width + Input data binary point position + Coefficient binary point position - 4)	0

Table 3-1. Parameter Specifications for the FIR Filter IP Core (Continued)

Parameter	Range	Default
Precision control		
Overflow	{Saturation, Wrap-around}	Saturation
Rounding	{None, Round-up, Round away from zero, Round towards zero, Convergent rounding}	None
Memory Type		
Data memory type	{EBR, Distributed, Auto}	EBR
Coefficient memory type	{EBR, Distributed, Auto}	EBR
Input buffer type	{EBR, Distributed, Auto}	EBR
Output buffer type	{EBR, Distributed, Auto}	EBR
Optimization	{Area, Speed}	{Area}
Optional Ports		
ce	{Yes, No}	No
sr	{Yes, No}	No
Synthesis Options		
Frequency constraint	1 – 400	300

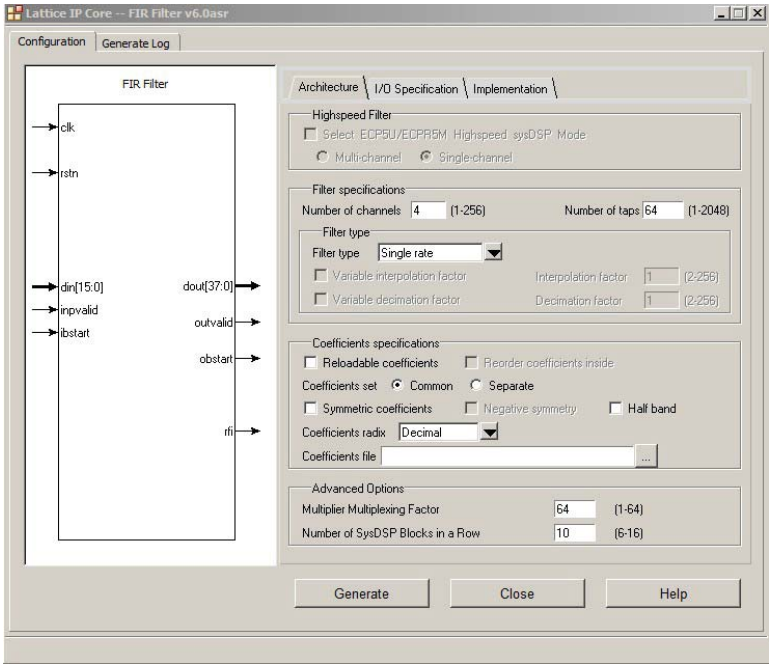
1.
2. The **Multiplier Multiplexing Factor** is limited by the number of DSP blocks in a device (A) and the actual number of DSP blocks a design needs (B). When A>B, the **Multiplier Multiplexing Factor** is set to 1; otherwise the value will be greater than 1.
3. See "Multiplier Multiplexing Factor" on page 25 for details.
4. Maximum number of DSP blocks available in a row in the selected device.

The default values shown in the following pages are those used for the FIR Filter reference design. IP core options for each tab are discussed in further detail.

Architecture Tab

Figure 3-1 shows the contents of the Architecture tab.

Figure 3-1. Architecture Tab of the FIR Filter IP Core GUI



Select ECP5 High Speed sysDSP Mode

This option allows users to select whether to use high speed mode or not.

Multi-channel

This option allows users to specify whether the filter should be a multiple channels filter in high speed mode

Single-channel

This option allows users to specify whether the filter should be a single channel filter in high speed mode.

Number of Channels

This option allows the user to specify the number of channels.

Number of Taps

This option allows the user to specify the number of taps.

Filter Type

This option allows the user to specify whether the filter is single rate, interpolator or decimator.

Interpolation Factor

This option allows the user to specify the value of the fixed interpolation factor. When FIR type is interpolation, the value should be 2 to 256, otherwise, it will be set to 1 automatically.

Variable Interpolation Factor

This option allows the user to specify whether the interpolation factor is fixed at the time of IP generation or variable during run-time. If this is checked, the interpolation factor is set through the input port *ifactor* when *factorset* is high.

Decimation Factor

This option allows the user to specify the value of the fixed decimation factor. When FIR type is decimation, the value should be 2 to 256, otherwise, it will be set to 1 automatically.

Variable Decimation Factor

This option allows the user to specify whether the decimation factor is fixed at the time of IP generation or variable during run-time. If this is checked, the decimation factor is set through the input port *dfactor* when *factorset* is high.

Reloadable Coefficients

This option allows the user to specify whether the coefficients are fixed or reloadable. If checked, the coefficients can be reloaded during core operation using the input port *coeffin*.

Reorder Coefficients Inside

When coefficients are reloadable, they need to be entered in a particular order. The reordering can be done using the program supplied along with the IP core. However, the core also provides for optional hardware reordering at the expense of additional hardware resources. If this option is selected, the coefficients can be entered in the normal sequence to the core and the core will internally reorder hem as required. This option is not available when Filter type is interpolator and Symmetric coefficients is enabled.

Coefficients set

This option allows the user to specify whether the same coefficient set is used for all channels or an independent coefficient set is used for each channel.

Symmetric Coefficients

This option allows the user to specify whether the coefficients are symmetric. If this is checked only one half of the number of coefficients (if number of taps is odd, the half value is rounded to the next higher integer) is read from the initialization file.

Negative Symmetry

If this is checked, the coefficients are considered to be negative symmetric. That is the second half of the coefficients are made equal to the negative of the corresponding first-half coefficients.

Half Band

This option allows the user to specify whether a half band filter is realized. 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.

Coefficient Radix

This option allows the user to specify the radix for the coefficients in the coefficients file. For decimal radix, the negative values have a preceding unary minus sign. 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. 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

This option allows the user to specify the name and location of the coefficients file. If the coefficients file is not specified, the filter is initialized with a default coefficient set.

Multiplier Multiplexing Factor

This option allows the user to specify the **Multiplier Multiplexing Factor**. This parameter should be set to 1 for full parallel applications and to the maximum value supported in the GUI for full series applications.

Number of sysDSP Blocks in a Row

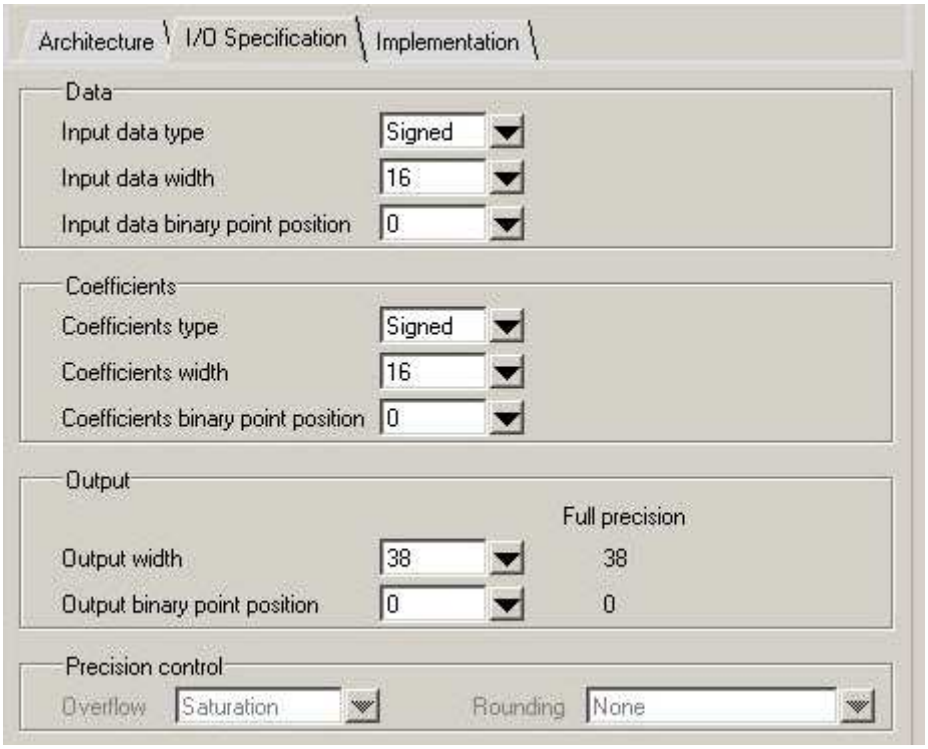
This parameter allows the user to specify the maximum number of DSP multipliers to be use in a DSP row to achieve optimal performance. For example, if the targeted device has 20 multipliers in a DSP row and the design requires 22 multipliers, the user can select to use all 20 multipliers in one row and two multipliers in another row, or fewer than 20 multipliers in each row (e.g. 8), which may yield better performance. Multipliers spread across a maximum of three DSP rows may be used in a single FIR instance.

This parameter is only valid on LatticeECP3 devices.

I/O Specification Tab

Figure 3-2 shows the contents of the I/O Specification tab.

Figure 3-2. I/O Specification Tab of the FIR Filter IP Core GUI



Input Data Type

This option allows the user to specify the input data type as signed or unsigned. If the type is signed, the data is interpreted as a two's complement number.

Input Data Width

This option allows the user to specify input data width.

Input Data Binary Point Position

This option allows the user to specify the location of the binary point in the input data. 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.