Parallel Distributed Arithmetic FIR Filter

December 30, 1998 | Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: coregen@xilinx.com
URL: www.xilinx.com

Features

- Input data words from 4 to 16 bits
- Coefficient width from 4 to 24 bits
- Taps from 2 to 20 for symmetric filters
- Taps from 2 to 10 for non-symmetric filters
- Support for cascading multiple filters together to create larger filters
- Support for symmetric, non-symmetric, and negative symmetric filters
- Full precision output
- Scaleable output
- Pipelined construction for higher throughput
- 2's complement input data
- Uses Fast Carry logic for high speed
- Drop-in modules for the XC4000E, EX, XL, XV and Spartan families
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Available in Xilinx CORE Generator

General Description

FIR filters are one of the most basic building blocks used in digital signal processing, taxing the performance that DSP hardware can deliver. Multiply accumulates must be performed at an ever-increasing rate and demands in the billions of MACs per second range are now common.

Field programmable gate arrays (FPGAs) using distributed arithmetic algorithms can implement large numbers of taps at MHz data sample rates, outperforming DSP processors by one or two orders of magnitude. Sample rates can be efficiently handled over a wide range by applying the proper structure that just meets the required performance with the least number of FPGA configurable logic blocks (CLBs).

Functional Description

Data presented at the filter DATA_IN port is stored within the filter module in an array of internal registers - one per tap. Filter coefficients provided by the user are stored in internal look-up tables and accessed during filter operation in accordance with the parallel distributed arithmetic algorithm. Partial results from each look-up table are summed to form a final result at the filter output port DATA_OUT. See the functional block diagram below. Internally, the filter retains full precision throughout all stages of computation resulting in zero noise at the filter output. Given information about the bit-width of the input data, coefficients, and the number of filter taps, the parameterization window indicates the number of bits necessary at the output port to encompass the dynamic range of the filter output. The output port may be truncated to fewer bits if desired. Truncation occurs by trimming as many LSBs from the result as necessary to deliver the requested number of bits at the output port. Truncation does not significantly reduce the amount of logic required by the filter since, internally, full precision is always maintained.

Selecting the highest possible value for the output bit-width (i.e. no truncation) and applying a unit impulse at the data input port causes the filter to reproduce its own impulse response at its output.

The filter accepts a new input data word, and produces a new filtered result, on every clock cycle.

Pinout

Port names for the schematic symbol are shown in Figure 2 and described in Table 1.

Cascading Multiple PDA FIR Filters

Filters requiring numbers of taps greater than the limit imposed by this module can be constructed by ‘cascading’ several filter modules and summing the results of each to form a final output. Cascading filter modules is performed using the DATA_IN, C_M_O, C_M_I and C_D_O pins. See the diagram below.

Since the latency between the DATA_IN and DATA_OUT ports of each filter module is related to the number of taps, to ensure that data arrives at the summation correctly, make all cascaded sections the same number of taps. Only the last filter in the chain may have an odd number of taps, i.e. when the total number of taps desired is odd, give each filter module an even numbers of taps, except the last one

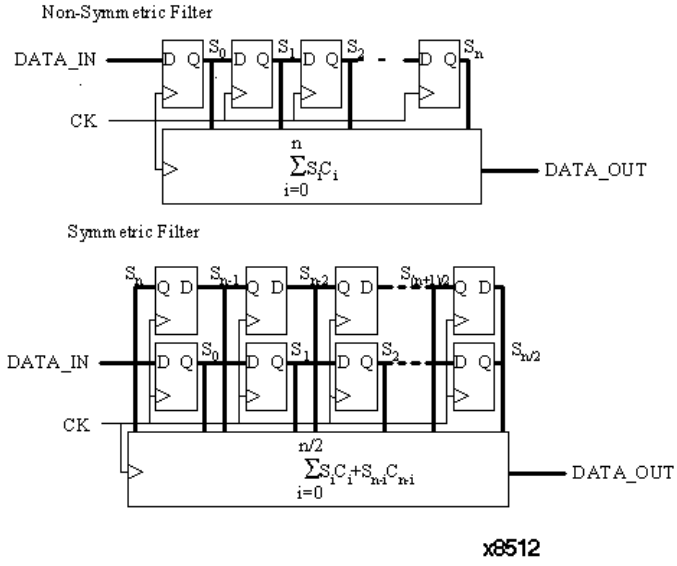

Figure 1: Functional Block Diagram

which must be odd. (The same number of taps as the other sections minus 1.)

Core Generator Parameters

The CORE Generator parameterization window for this macro is shown in Figure 5. The parameters are as follows:

- **Component Name**: Enter a name for the component.
- **Input Width**: Select an input bit width from the pull-down menu. The valid range is 4-16.
- **Signed Input Data:** Selects Signed or Unsigned input data.
- **Output Width**: Full arithmetic precision is carried throughout the filter but the output port may be truncated. This parameter selects the number of bits to bring out of the filter. The valid range depends on factors such as coefficient width, data width, number of taps, and Trim Empty ROMs.
- **Symmetry:** Selects Symmetric or Non-symmetric coefficients. The Anti-Symmetry option is only enabled if you choose a Symmetric impulse response and the input data is Signed.
- **Anti-Symmetry**: Valid only for symmetric filters. When checked, the reflected half of the impulse response is negated.
- **Cascade**: Allows this module to be combined with other filter modules to form a larger filter.
- **Number of Taps:** For symmetrical filters the range is 2 to 20. For non-symmetrical filters the range is 2 to 10.
- **Coefficient Width:** This determines the width of the internal look-up tables. The tables automatically grow from the width specified to accommodate the bit growth as the coefficient sums are calculated. The valid range is 4-24.
- **Trim Empty ROMs:** Since not all coefficients will require all 'coefficient width' bits to express them, this option allows look-up tables to be optimized to the smallest possible dimensions, given the actual coefficient values. Selecting this option may reduce the maximum output bitwidth.
- **Load Coefficients:** Get coefficients from a specified file.
- **Show Coefficients**: Display the coefficients after they have been loaded.
- **.coe file**: Displays the name of the coefficient file. This field is read-only.

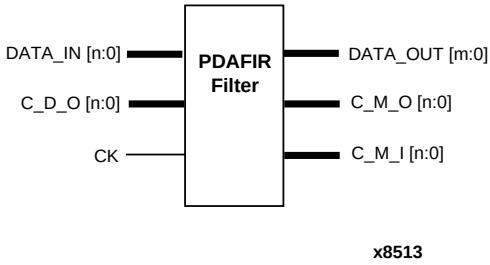

Figure 2: Core Schematic Symbol

Specifying Filter Coefficients

Not all of the parameters necessary to completely specify a filter's characteristics can be specified from the parameterization window. Specifically, the filter coefficients must be loaded from a text file (known as a COE file) described below. In addition, all the parameters visible on the parameterization window may be assigned values in the COE file. COE files may take any root filename but must end with the extension ".COE".

Table 1: CORE Signal Pinout

Signal	Signal Direction	Description
DATA_IN [n:0]	Input	PARALLEL DATA INPUT
CK	Input	CLOCK - with the exception of asynchronous control inputs (where applicable), control and data inputs are captured, and new output data formed on rising clock transitions.
c_m_i [n:0]	Input (optional)	CASCADE MID INPUT – used when cascading multiple filters. Leave unconnected if not cascading symmetric filters. Parallel data from next cascaded filter.
c_m_o [n:0]	Output (optional)	CASCADE MID OUTPUT – used when cascading multiple filters. Leave unconnected if not cascading symmetric filters. Parallel data to next cascaded filter.
c_d_o [n:0]	Output (optional)	CASCADE DATA OUTPUT – used when cascading multiple filters. Leave unconnected if not cascading filters. Parallel data to next cascaded filter.
DATA_OUT [m:0]	Output	FILTER RESULT – Parallel Data Out.

To select and load a COE file, press the "Load Coefficients…" button on the filter parameterization window and choose the desired file from the dialog-box. To review the filter's coefficient-values at any time, press the "Show Coefficients…" button. Any field on the parameterization window that is assigned a value in the COE file will loose its previous value when the COE file is loaded. Changing a parameter value that was previously loaded from a COE file causes the COE file's name to be highlighted in red, indicating that the settings have changed since the file was loaded.

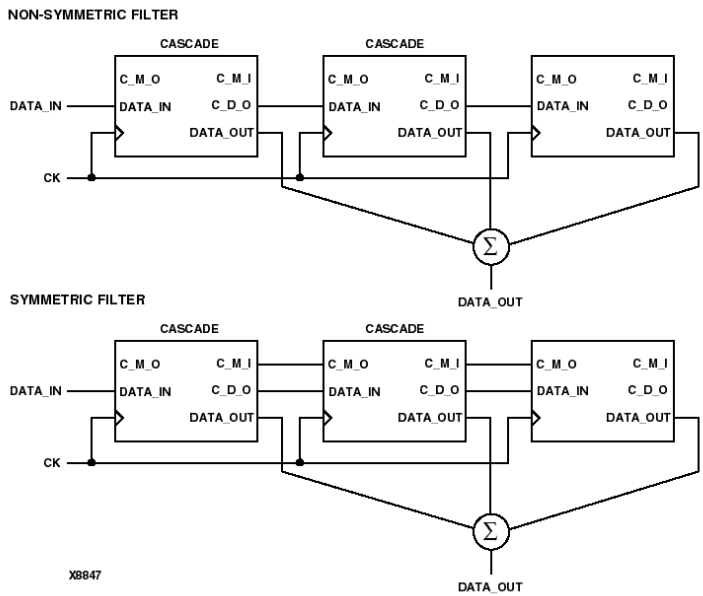

Figure 3: Cascading Multiple PDA FIR Filter

For a detailed description of COE file syntax, please refer to the Xilinx CORE Generator User Guide. The list of COE file keywords supported by the Parallel Distributed Arithmetic FIR Filter are shown in the Parameter File Information table at the end of this datasheet. An example COE file is shown in Figure 4.

When specifying filter coefficients in a COE file, the Radix is assumed to be 10. You may however specify COEFDATA as hexadecimal by adding the command RADIX=16.

```
component_name=pdafir;
number_of_taps=8;
radix=10;
input_width=7;
coef_width=8;
output_width=20;
symmetry=true;
antisymmetry=false;
cascade=false;
trim_empty_roms=false;
signed_input_data=false;
coefdata=41,35,15,99;
```

Figure 4: Example COE File For the PDA FIR Filter

Filter Output Bit-Width

The filter calculates the number of bits required at its RSLT output port on the basis of the parameters provided by the user. The width of the filter's RSLT port will always be sufficient to express the largest magnitude result that may ever arise during the filter's operation. However, the bit-width calculation is conservative and a RSLT port with more bits than necessary may be created.

The process used to determine the filter's output port bit-width is:

Largest Coefficient Value (Given the coefficient bit-width provided by the user)

$2^{\text{Coefficient Bit Width} - 1}$

Largest Sample Value (Given the data bit-width provided by the user)

$2^{(\text{Data Bit Width1})} - 1$ for signed data,
$2^{\text{Data Bit Width}} - 1$ for signed data

Largest Filter Output Value (Given the number of taps provided by the user)

Largest Sample Value * *Largest Coefficient Value* * Number Of Taps

Number Of Bits Required At Output Port

Log_2(*Largest Filter Output Value*) + 2 ...Rounded down to the previous whole number

Given that not all coefficients will be as large as the *Largest Coefficient Value* calculated above, the filter's RSLT output port may, at times, be wider than necessary. Given a specific set of coefficients, the ideal output port bit-width may be found manually by substituting the sum of the magnitudes of the coefficients for the product of the *Largest Coefficient Value* and the Number Of Taps. That is,

Largest Sample Value (Given the data bit-width provided by the user)

$2^{(\text{Data Bit Width1})} - 1$ for signed data,
$2^{\text{Data Bit Width}} - 1$ for signed data

Largest Filter Output Value (Given the coefficients provided by the user)

Largest Sample Value * ($|C_0| + |C_1| + |C_2| + ... + |C_n|$)

Number Of Bits Required At Output Port

Log_2(*Largest Filter Output Value*) + 2 ...Rounded-down to the previous whole number

The difference between the conservative bit-width calculated by the Core Generator and the ideal bit-width calculated manually represents the number of unnecessary MSBs that are included in the filter's output port. These bits may safely be dropped from the filter's output, and a small amount of logic trimming may occur as a result.

As an example, take a symmetric 6-tap filter, with coefficients 1,-8,12,12,-8 and 1. The center coefficients require 5 bits to express, but the other coefficients require only 3 and 2 bits. Assuming 6-bit sample data, the Core Generator conservatively calculates that this filter requires a 13-bit RSLT port. Given the specific coefficients however, only 12 bits are actually necessary. In this application therefore, the MSB of the filter's output may be discarded.

Core Resource Utilization

The PDA FIR filter processes one data word every clock cycle. The sample rate is therefore equal to the clock rate.

Tables 2 and 3 show the approximate number of CLBs required for various example bit widths. Tables 4 and 5 show the maximum speed (in MHz) for data sample rates using XC4036XL-1 devices.

Table 2: CLB Utilization for Symmetrical PDA FIR Filters.

# Taps	# Bits (Data and Coefficients)		
	8	10	12
4	142	191	252
8	185	241	309
10	252	333	430

Table 3: CLB Utilization for Non-symmetrical PDA FIR Filters

# Taps	# Bits (Data and Coefficients)		
	8	10	12
2	95	156	204
3	110	173	224
4	115	325	231
5	165	252	331

Table 4: Performance for Symmetrical PDA FIR Filters (MHz)

# Taps	# Bits (Data and Coefficients)		
	8	10	12
4	88	78	77
8	80	74	65
10	70	74	60

Table 5: Performance for Non-symmetrical PDA FIR Filter (MHz)

# Taps	# Bits (Data and Coefficients)		
	8	10	12
2	78	81	68
3	74	77	64
4	74	73	64
5	75	66	70

Ordering Information

This macro comes free with the Xilinx CORE Generator.

For additional information contact your local Xilinx sales representative, or e-mail requests to coregen@xilinx.com.

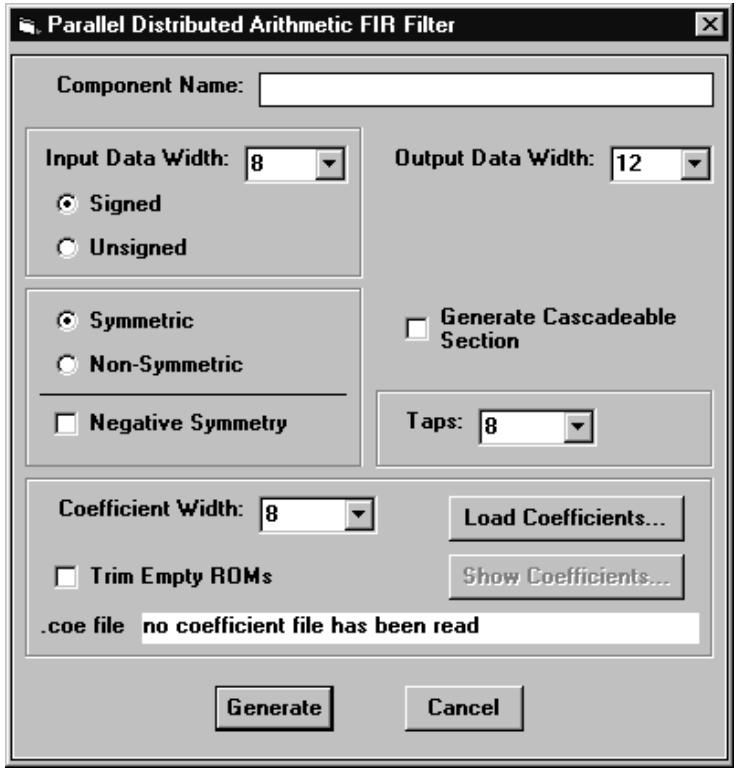

Figure 5: Parameterization Window

Parameter File Information

Component Name	Type	Notes
Component_Name	String	
Input_Width	Integer	4 - 16
Output_Width	Integer	
Number_of_Taps	Integer	2 - 20
Coef_Width	Integer	4 - 24
Symmetry	Boolean	True/False
Antisymmetry	Boolean	True/False
Signed_Input_Data	Boolean	True/False
Cascade	Boolean	True/False
Trim_Empty_ROMS	Boolean	True/False
CoefData{n} 0<=n<=9	Integer	
Radix*	Integer	10 or 16

*COE file only.