Key Design Features

- Synthesizable, technology independent VHDL Core
- Half-band polyphase decimation filter with a configurable decimation factor from 2 to 2^N
- FIR filter core implemented as an optimized 48-tap systolic array (24-taps per phase) for ultimate speed and minimal resource use
- Configurable data and coefficient widths
- Symmetric arithmetic rounding limits DC-bias problems
- Saturation of output samples - no wrap
- Ships with 14-bit coefficients giving 80 dB stop-band attenuation
- Only 12 H/W multipliers used per decimate-by-2 stage
- Supports input sample rates of up to 300 MHz+

Applications

- Decimation of signals after digital-down-conversion
- Decimation by a wide range of factors from 2 to 2^N
- Reduction of input sample rate to make subsequent signal processing easier

Generic Parameters

Generic name	*Description*	*Type*	*Valid range*
num_stages	Number of decimate-by-2 filter stages (N)	integer	≥ 1
dw	Width of input/output data samples	integer	≥ 2
cw	Width of coefficients	integer	≥ 2
fw	Number of coefficient fraction bits	integer	≥ 0 (fw < cw)

Pin-out Description

Pin name	*I/O*	*Description*	*Active state*
clk	in	System clock (F_S)	rising edge
reset	in	System reset	low
en	in	Clock enable in	high
en_out	out	Clock enable out	high
x_in	in	Input samples (signed number)	data
y_out	out	Decimated output samples (signed number)	data

Block Diagram

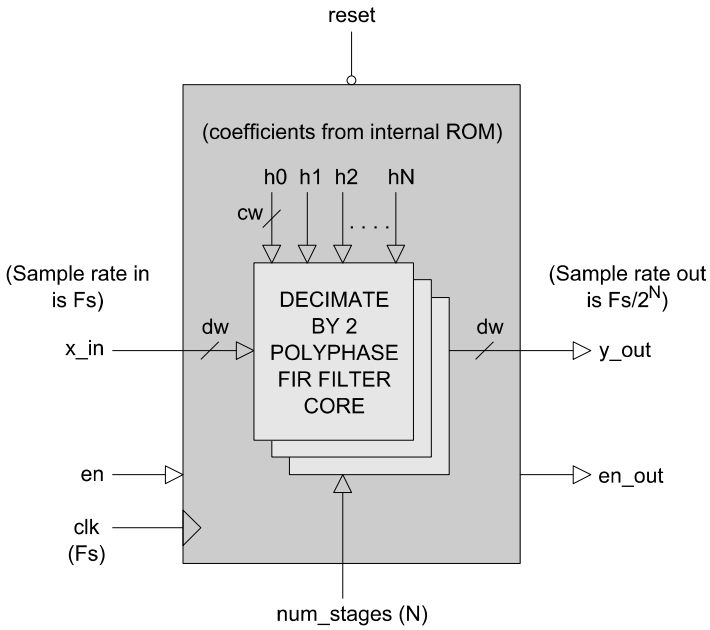


Figure 1: Decimation Filter Architecture

General Description

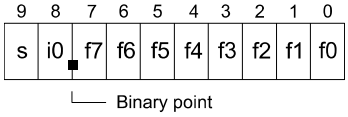
FIR_DEC_N is a polyphase decimation filter that permits the down-sampling of an input signal by any power of 2. The filter core is organized as a highly optimized systolic array, allowing the user to specify very large decimation factors while keeping resource costs to a minimum.

Input data is sampled on the rising clock-edge of *clk* when *en* is active high. Internally, the samples are filtered and decimated then presented at the output interface, *y_out*.

The output signal *en_out* is the output clock-enable signal that indicates when an output sample is valid. For instance, when decimating by a factor of 2, then *en_out* will have a duty cycle of 50% relative to the input clock-enable signal. When decimating by a factor of 4, the duty cycle will be 25% and so on.

Filter coefficients and I/O specification

Filter coefficients[1] are defined as signed fixed-point numbers in [*cw fw*] format where *cw* is the total number of coefficient bits and *fw* is the number of bits in the fractional part. In all cases, *cw* must be at least 2 bits and *fw* must be less than *cw* to accommodate the sign bit. For instance, a coefficient in [10 8] format would be arranged as follows:



1 The design is supplied with Matlab® scripts for the easy generation of different coefficient sets using FDAtool®. Please see application note: app_note_zc002.pdf for more details.

The standard design ships with 14-bit coefficients in [14 13] format. This coefficient set is sufficient to obtain up to 80dBs of stop band attenuation with a 48-tap half-band filter. Using coefficients with a fewer number of bits will result in a smaller design, but will also compromise filter performance[2].

The number of bits in the input and output samples is controlled by the parameter *dw*. Inputs and outputs are signed values (their format is purely relative).

Sampling frequency considerations

The system clock frequency is the sampling frequency of the internal filter core. This is the same as the input sample rate. The sample rate of the output data is a function of the input sampling frequency, F_S , and the decimation factor, N, where:

$$F_S(at\ output) = \frac{F_S}{N}$$

$$where\ ,$$

$$N = 2^{\text{num\_stages}}$$

Filter characteristic

Each decimate-by-2 filter section is a 48-tap half-band Nyquist filter with 24-taps per phase. The magnitude response, impulse response and step response of the filter are shown in Figures 2, 3 and 4.

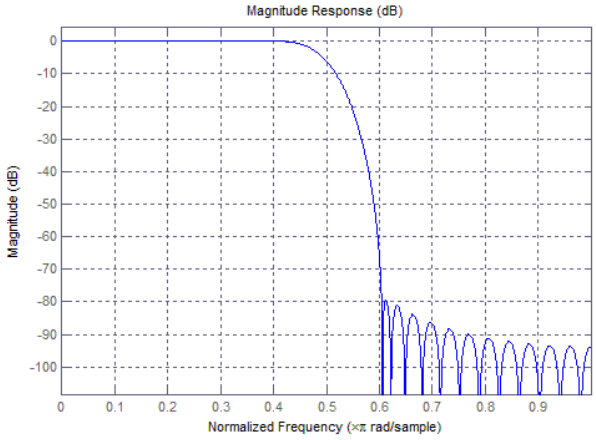


Figure 2: Magnitude response – 48-tap half-band Nyquist filter

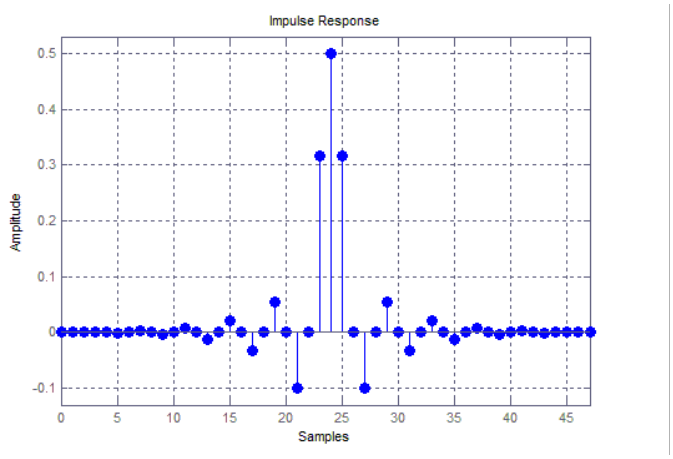


Figure 3: Impulse response – 48-tap half-band Nyquist filter

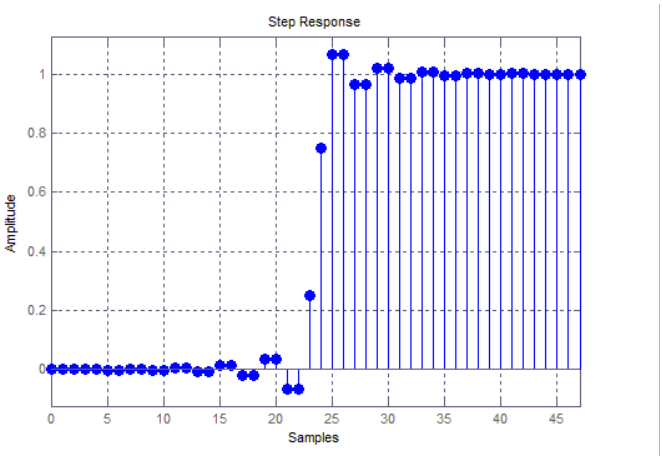


Figure 4: Step-response - 48-tap half-band Nyquist filter

Functional Timing

Figure 5 shows a sequence of input and output samples for a decimation factor of 2. Notice that output samples are valid every 2nd clock cycle. Outputs should be sampled on the rising clock-edge of *clk* when *en_out* is active high.

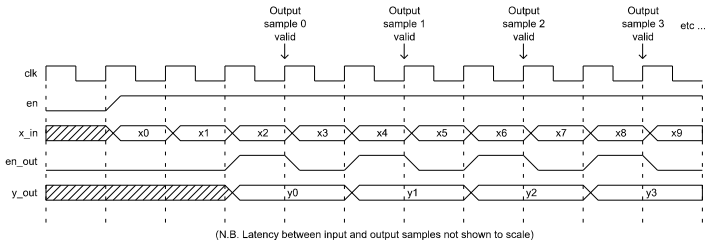


Figure 5: Timing waveform - downsample by 2

2 We recommend that the parameters in the filter package are left alone without modification. If the user requires a different filter response, then please contact Zipcores for advice.

Figure 6 shows a similar sequence, but this time for a decimation factor of 4. In this case, the output samples are valid every 4th clock cycle.

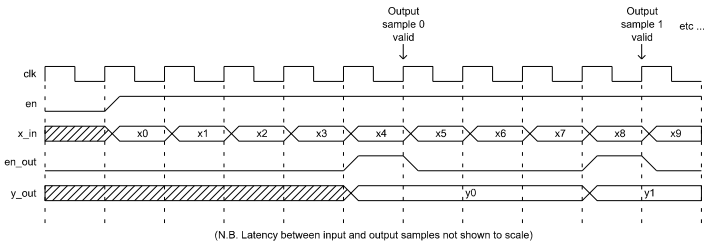


Figure 6: Timing waveform - downsample by 4

Likewise, successive decimation factors of 8, 16, 32, etc. have similar timing waveforms with the *en_out* signal being asserted every 8th, 16th and 32nd clock cycle respectively.

Source File Description

All source files are provided as text files coded in VHDL. The following table gives a brief description of each file. Note that all generic parameters are defined in the package 'fir_dec_pack.vhd'.

Source file	*Description*
fir_dec_pack.vhd	Package containing all generic parameters - including coefficients
fir_dec_mad.vhd	Multiply-add block
fir_dec_mad_centre.vhd	Multiply-add block - centre tap
fir_dec_mad_zero.vhd	Multiply-add block - zero coeffs
fir_dec_rnd.vhd	Rounding block
fir_dec_sat_vhd	Saturation block
fir_dec_s0.vhd	Filter polyphase section 0
fir_dec_s1.vhd	Filter polyphase section 1
fir_dec.vhd	Decimate-by-2 filter
fir_dec_n.vhd	Top-level component
fir_dec_n_bench.vhd	Top-level test bench

Functional Testing

An example VHDL testbench is provided for use in a suitable VHDL simulator. The compilation order of the source code is the same as the source file description above.

The test provided is configured for a single decimate-by-2 filter with the generic parameter *num_stages* set to '1'. The simulation must be run for at least 1 ms during which time the impulse response and step response of the filter is tested.

The simulation generates a text file called 'fir_dec_n_out.txt' that contains the output samples captured during the course of the test. Figures 7 and 8 respectively demonstrate the impulse response and step response outputs for the given test example.

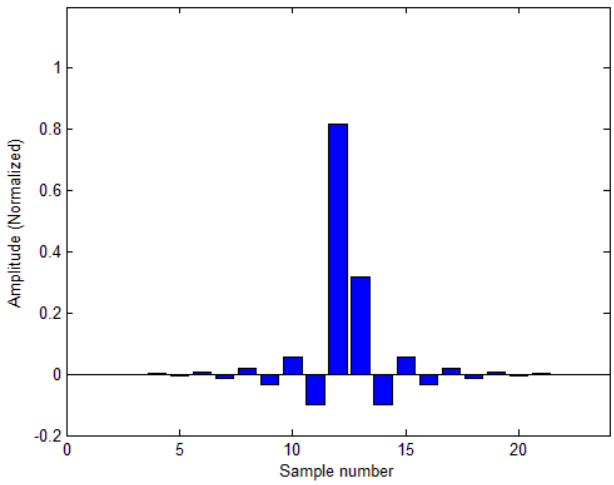


Figure 7: Impulse response (2 input samples) - testbench example

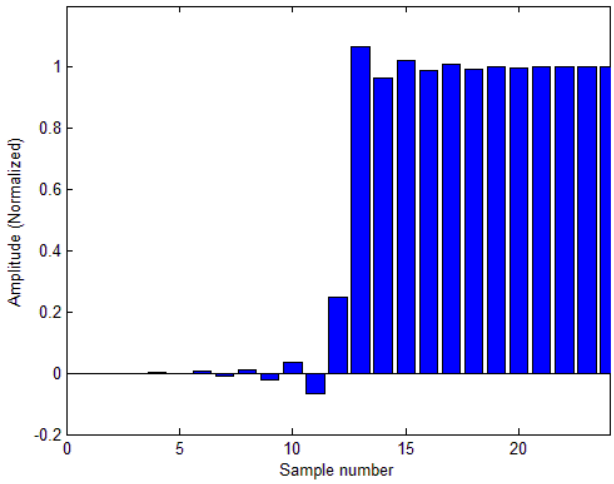


Figure 8: Step response - testbench example

Synthesis

The files required for synthesis and the design hierarchy is shown below:

- fir_dec_pack.vhd
- fir_dec_n.vhd
  - fir_dec.vhd
    - fir_dec_s0.vhd
    - fir_dec_s1.vhd
      - fir_dec_mad.vhd
      - fir_dec_mad_centre.vhd
      - fir_dec_mad_zero.vhd
    - fir_dec_rnd.vhd
    - fir_dec_sat_vhd

The VHDL core is designed to be technology independent. However, as a benchmark, synthesis results have been provided for the Xilinx Virtex 5 and the Altera Stratix III series of FPGA devices. The lowest and highest speed grade devices have been chosen in both cases for comparison.

Smaller and faster designs will generally be achieved by using coefficient sets with less bits. However, this will compromise filter performance. It is recommended that a coefficient width of at least 14-bits is used to achieve ~80dBs of attenuation in the stop-band.

Trial synthesis results are shown with the generic parameter *num_stages* set to '1'. Resource usage is specified after Place and Route.

VIRTEX 5

Resource type	***Quantity used***
Slice register	320
Slice LUT	447
Block RAM	0
DSP48	14
Clock frequency (worst case)	255 MHz
Clock frequency (best case)	330 MHz

STRATIX III

Resource type	***Quantity used***
Register	1211
ALUT	667
Block Memory bit	336
DSP block 18	20
Clock frequency (worse case)	190 MHz
Clock frequency (best case)	275 MHz

Revision History

Revision	***Change description***	***Date***
1.0	Initial revision	17/02/2011
1.1	Updated synthesis results in line with minor source code changes	13/04/2011
1.2	Made *num_stages* a generic parameter	25/10/2011

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Development Software ***category:***

Click to view products by Zipcores ***manufacturer:***

Other Similar products are found below :

SRP004001-01 SW163052 SYSWINEV21 WS01NCTF1E W128E13 SW89CN0-ZCC IP-UART-16550 MPROG-PRO535E AFLCF-08-LX-CE060-R21 WS02-CFSC1-EV3-UP SYSMAC-STUDIO-EIPCPLR 1120270005 MIKROBASIC PRO FOR FT90X (USB DONGLE) MIKROC PRO FOR FT90X (USB DONGLE) MIKROBASIC PRO FOR AVR (USB DONGLE LICEN MIKROBASIC PRO FOR FT90X MIKROC PRO FOR DSPIC30/33 (USB DONGLE LI MIKROPASCAL PRO FOR ARM (USB DONGLE LICE MIKROPASCAL PRO FOR FT90X MIKROPASCAL PRO FOR FT90X (USB DONGLE) MIKROPASCAL PRO FOR PIC32 (USB DONGLE LI SW006021-2H ATATMELSTUDIO 2400573 2702579 2988609 SW006022-DGL 2400303 88970111 DG-ACC-NET-CD 55195101-101 55195101-102 SW1A-W1C MDK-ARM SW006021-2NH B10443 SW006021-1H SW006021-2 SW006022-2 SW006023-2 SW007023 MIKROE-730 MIKROE-2401 MIKROE-499 MIKROE-722 MIKROE-724 MIKROE-726 MIKROE-728 MIKROE-732 MIKROE-734