# Rate Converter Product Specification

## OVERVIEW

This document describes a specific variant in RFEL's range of Multirate Filter cores. This specific core processes interleaved complex input data in continuous real time, with no gaps in the data, at complex data rates of up to 100MS/s. An alternative variant provides real filtering at rates of up to 200MS/s.

Rate conversion is necessary in systems where a precise output sample rate is required, which is not an integer division of the input sample rate. One example of a scenario where this is particularly important is in communication systems where demodulators commonly require the output rate to be an exact multiple of the symbol rate, but the output of the channelisation process does not match this.

The range of cores described provide fractional sample rate conversion for input and output sample rates that are related by a simple integer ratio. That is, $f_{out} = L/M\, f_{in}$ where $f_{in}$ and $f_{out}$ are the input and output sample rates and $L$ and $M$ are integers referred to as the upsampling and downsampling factors. This includes simple integer upsampling (i.e. $M=1$) and simple integer downsampling (i.e. $L=1$) as special cases. Designs are available for non-integer ratios. In practice, the cores described here provide an adequate solution for the vast majority of practical cases.

Another class of rate conversion filter supplied by RFEL, the CIC, is described in [1].

For the example core described here, the complex input signal is upsampled by a factor $L=5$ and downsampled by $M=8$ to provide an output rate of $f_{out} = 5/8 f_{in}$ . For a complex input rate of 100MHz, this yields an output sample rate of 62.5MHz. This core was part of a larger channeliser core and permitted the output of a signal at the exact required demodulator rate.

RFEL's Multirate Filter cores are intended for use in applications where processing speed is critical and optimum use of available silicon is required. The core is available for licence in netlist form as a component ready to be combined with customer's own IP. Alternatively RFEL can provide the core as part of a larger channeliser core in either netlist or bitstream format.

## FEATURES

- Continuous real-time processing of interleaved complex data at up to 100MS/s
- Continuous real-time processing of real or complex data at up to 200MS/s
- Rate conversion of multiple interleaved low-rate channels is also possible
- Fully pipelined design
- Targeted at Xilinx and Altera FPGA families
- Bit-widths and bit-growths adjustable at factory
- Fully bit-true parameterisable models are available
- Other sample rate conversion ratios available

 

## APPLICATIONS

- Wide-band filter banks
- Communications systems with exact required bin spacing
- Electronic warfare (radar, sonar, surveillance)
- Medical instrumentation
- Test instrumentation
- Real-time spectral analysis

## GENERAL DESCRIPTION

The multirate filter core takes an input sample stream at a rate $f_{in}$ and passes it through a polyphase filter structure which upconverts to a rate of $f_{mid} = Lf_{in}$, filters to eliminate aliases and images and then downconverts to a rate of $f_{out} = L/M\, f_{in}$. Because of the polyphase filter structure used, all processing is done at the input sample rate $f_{in}$. There is no need for the FPGA device to run at the $f_{mid}$ sample rate.

The process of rate conversion is covered in detail in [1] and [3], so only a brief summary will be given here. The input signal must be bandlimited such that it complies with Nyquist for both the input rate and the output rate (see Figure 1(a) and (e)). The full procedure is as follows:

1) The input sample stream is upsampled $L$ times by inserting $L$-1 zero samples between each input data sample. Figure 1(b) shows that this results in the generation of multiple images in the frequency domain.

2) If the signal were decimated directly without filtering, the images that appear in the alias regions shown in Figure 1(d) would fold into the output signal corrupting the output and destroying the useful data. Therefore a filter is used with the shape shown in Figure 1(b, c, d) that eliminates all signal power in the alias regions. The filtering process removes all the images as shown in Figure 1(c).

3) The signal may then be downsampled $M$ times by discarding $M$-1 out of every $M$ samples as shown in Figure 1(e). The resultant signal does not suffer from corruption in the passband, since all the frequency content in the alias bands was removed at the intermediate frequency.

It is important to note that it would be wasteful and in many cases impractical to carry out the rate conversion process in a stepwise fashion as described above. The core provided by RFEL uses an efficient polyphase structure to carry out this rate conversion process thereby achieving the same performance without the need to use a high intermediate clock rate.

The design is synchronous and the rate conversion described here is a rate reduction, so the clock rate $f_s = f_{in}$ ($f_s = 2 \times f_{in}$ for interleaved complex inputs) and the output is on an enable domain, such that it is clocked at rate $f_s$ with the first 5 samples in every 8-sample block enabled (interleaved for complex). The filter generates this enable signal internally and outputs it along with the data stream; Figure 2 is a timing diagram example for a rate convertor applied to a real signal.

Conversely, for a rate increase, the output rate would match the clock rate and the input signal would be on an enable domain with the first $M$ samples in every $L$-sample block enabled.

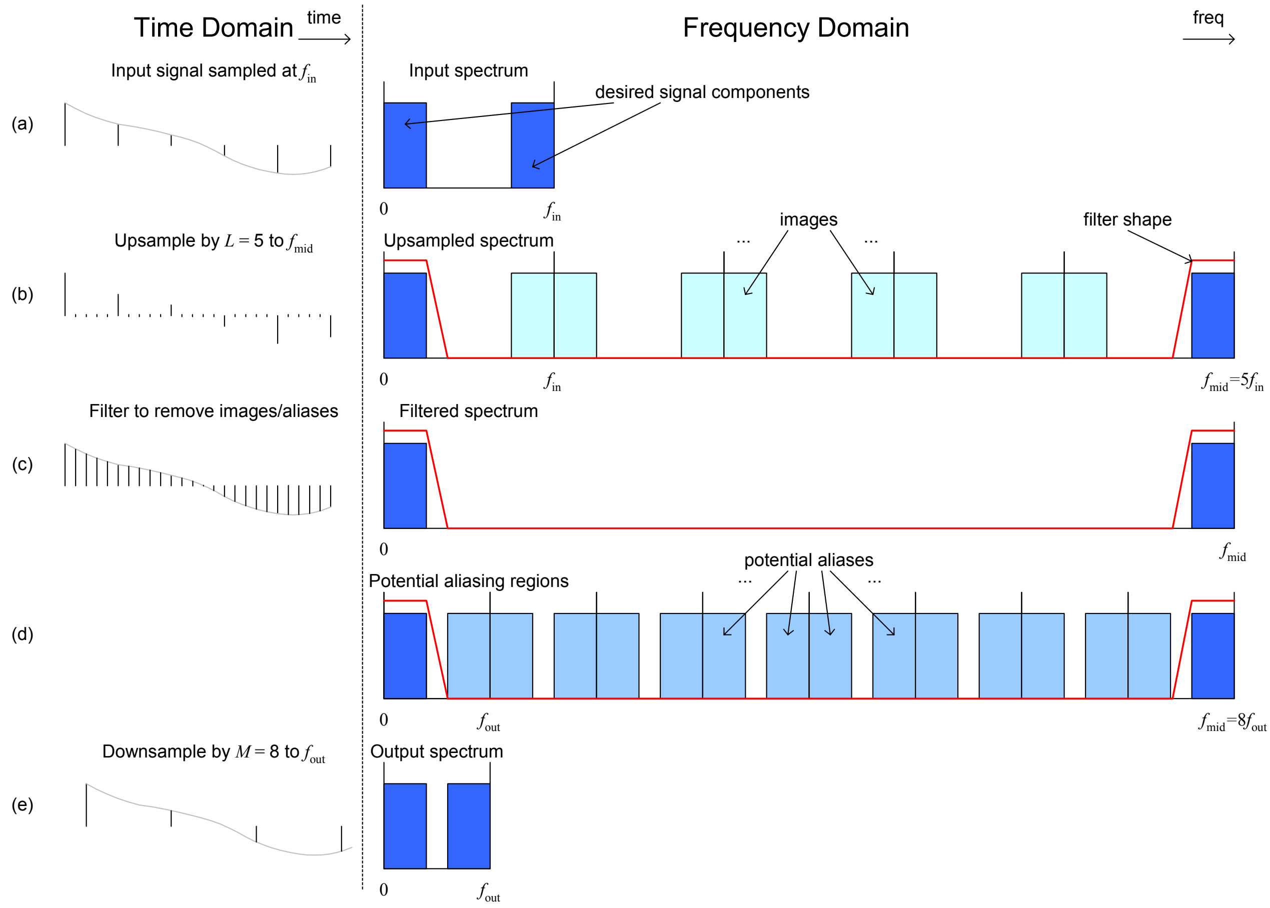


**Figure 1: Time and frequency domain representations of 5/8 rate conversion (a) Input signal; (b) Upsampled signal; (c) Filtered signal; (d) Locations of alias regions; (e) Output signal**

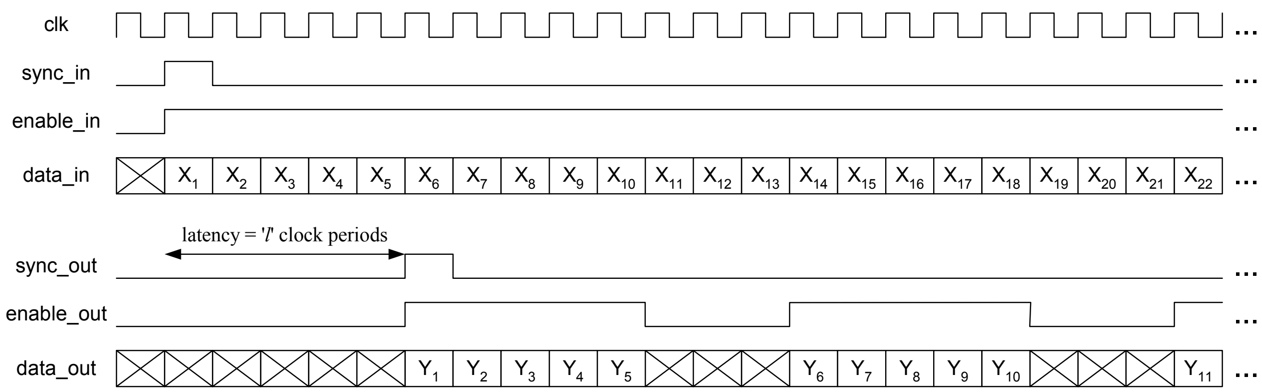


**Figure 2: Multirate filter I/O timing diagram (real signals)**

 

## CASE STUDY

The case study given here is for a multirate filter that formed part of a communication signal channeliser. For this application, the input sample rate was fixed by the ADC rate and the output sample rate of the final core had to be an exact integer multiple of the symbol rate. The specified rate convertor takes interleaved I and Q samples as inputs and provides interleaved I and Q samples as outputs.

The core parameters are shown in Table 1.

| Parameter | Value |
|---|---|
| Upsample factor | 5 |
| Downsample factor | 8 |
| Filter passband ripple | < 0.01dB |
| Stopband rejection | > 70dB |
| Number of filter taps | 280 |
| Input width | 14 |
| Output width | 15 |
| Tap width | 17 |
| Max clock rate | 200MHz |
| Max input sample rate (interleaved) | 100MHz |
| Max output sample rate (interleaved) | 62.5MHz |
| Max signal bandwidth | 54.5MHz |
| Device | Xilinx Virtex-II Pro 50 -5 |
| Slices | 1528 |
| Multipliers | 35 |
| RAMs | 0 |

**Table 1: Example core parameters**

## CORE INTERFACE DESCRIPTION

| Signal | Direction | Type | Width | Function |
|---|---|---|---|---|
| clk | IN | std logic | 1 bit | The core clock rate is equal to $f_s$ where $f_s$=2$f_{in}$ and $f_{in}$ is the input sample rate. |
| sync_in | IN | std logic | 1 bit | Active-high pulse marking the first sample of a new input block[1]. |
| enable_in | IN | std logic | 1 bit | Active-high signal asserted for the block to keep processing. Asserted from first sample of input data. |
| data_in | IN | std logic vector | (data_width_in -1 downto 0) | 2's complement data input. |
| sync_out | OUT | std logic | 1 bit | Active-high pulse marking the first sample of the first output block. Coincident with the first sample of the first output block. |
| enable_out | OUT | std logic | 1 bit | Active-high signal marking the valid output samples. |
| data_out | OUT | std logic vector | (data_width_out -1 downto 0) | 2's complement data output. |

**Table 2: Interface specification**

## DELIVERABLES

| Supplied Item | Description |
|---|---|
| Design | EDIF netlist |
| Constraints File | UCF (User Constraints File) |
| Instantiation Template | VHDL |
| Verification | VHDL test bench including ModelSim script and test data files.<br>Compiled RTL VHDL Model.<br>Bit-true Matlab model and scripts.<br>Placement reports. |

**Table 3: Items provided with each core**

Electronic transfer is used to deliver the cores and supporting documentation.

Optional design support services are available to help incorporate the core into larger designs.

[1] The sync_in only needs to be strobed at the start of the first frame. Internal counters then control the processing.

## GLOSSARY

| | |
|---|---|
| **ADC** | Analogue to Digital Converter |
| **EDIF** | Electronic Data Interchange Format |
| **FPGA** | Field Programmable Gate Array |
| **I/O** | Input / Output |
| **MS/s** | Million Samples Per Second |
| **RFEL** | RF Engines Limited |
| **RTL** | Register Transfer Level |
| **UCF** | User Constraints File |
| **VHDL** | Very High Speed IC Hardware Description Language |
| **VITAL** | VHDL Initiative Toward ASIC Libraries |

**Table 4: Glossary**

Please refer to the RFEL web site www.rfel.com for details of other signal processing IP cores.

Specifications are subject to change without notice.

## REFERENCES

[1] CIC Product Specification, RF Engines, April 2004.

[2] Multirate Digital Signal Processing, Crochiere R., and Rabiner L., 1983, Prentice-Hall.

[3] Multirate Systems and Filter Banks, Vaidyanathan, P.P., 1993, Prentice-Hall.