OpenCPI
CIC Interpolator Component Data Sheet

OpenCPI Release: v2.4.7

Summary - CIC Interpolator

Name	cic_int
Worker Type	Application
OpenCPI Release	v2.4.7
Last Update	4/2019
Component Library	ocpi.assets.dsp_comps
Workers	cic_int.hdl
Tested Platforms	alst4, isim, Matchstiq-Z1(PL), ml605, modelsim, xsim, ZedBoard(PL)

Functionality

The CIC interpolator has `N` cascaded comb stages with an input data rate of $\frac{f_s}{R}$, followed by a rate change by a factor `R`, followed by `N` cascaded integrator stages with an output data rate of f_s. The differential delay, `M`, affects the slope of the transition region. Figure 1 diagrams the interpolating CIC filter.

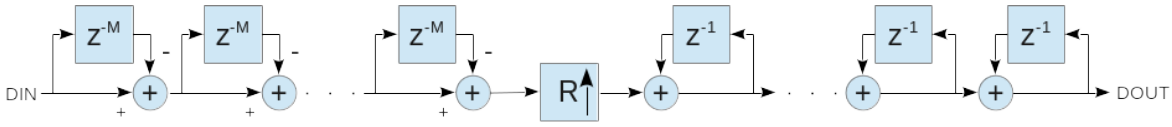

Figure 1: Cascaded Integration Comb Decimation filter Block Diagram

Worker Implementation Details

cic_int.hdl

Number of Stages

The generic `N` sets the number of integrators and comb stages in the filter. Increasing the number of stages increases the attenuation in the sidelobes and decreases the 3dB bandwidth of the passband. The recommended range for this parameter is 3 to 6. Consult the reference material for an in depth discussion of the frequency response of the filter as a function of the generics in this module.

Bit Growth

For this design, the output data width for the comb stages is configurable via `ACC_WIDTH`. To adjust for bit growth in the data path and to ensure no quantization error at the output, this equation should be used to determine the value of `ACC_WIDTH`.

$$ACC\_WIDTH = N * CEIL(log_2(R * M)) + DIN\_WIDTH \quad (1)$$

Theory

A CIC filter is comprised of `N` integrator sections cascaded together with `N` comb sections. Combining the transfer functions for all sections results in the system response function seen in Equation 2. Note that the filter has zeros at integer multiples of $\frac{f_s}{RM}$ Hz.

$$H(z) = \Big[H_{int}(z)\Big]^N \Big[H_{comb}(z)\Big]^N = \Big[\frac{1}{(1-z^{-1})^N}\Big]\Big[(1-z^{-RM})^N\Big] = \frac{(1-z^{-RM})^N}{(1-z^{-1})^N} \quad (2)$$

Block Diagrams

Top level

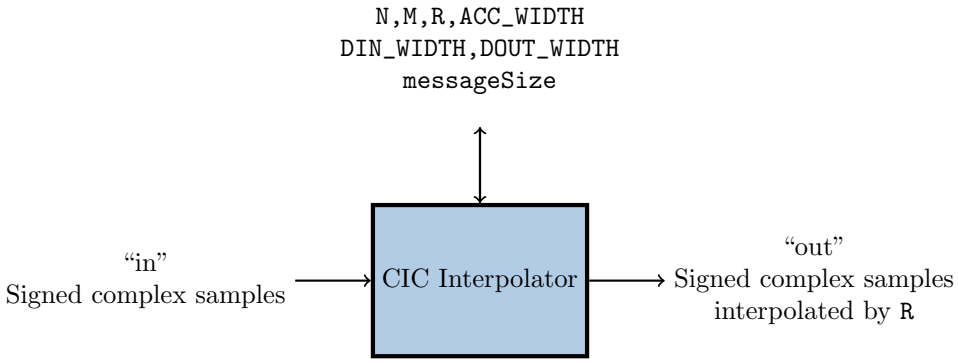

Figure 2: Top Level Block Diagram

Source Dependencies

cic_int.hdl

- assets/components/dsp_comps/cic_int.hdl/cic_int.vhd
- assets/hdl/primitives/dsp_prims/dsp_prims_pkg.vhd
  assets/hdl/primitives/dsp_prims/cic/src/cic_int_gen.vhd

Component Spec Properties

Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
N	UChar	-	-	Readable	-	-	Number of Stages
M	UChar	-	-	Readable	-	-	Differential Delay
R	UShort	-	-	Readable	-	-	Interpolation Factor
ACC_WIDTH	UChar	-	-	Readable	-	-	Accumulation Width *(2)
DIN_WIDTH	UChar	-	-	Readable	-	-	Input data width
DOUT_WIDTH	UChar	-	-	Readable	-	-	Output data width
messageSize	UShort	-	-	Readable, Writable	-	8192	Number of bytes in output message

Worker Properties

cic_int.hdl

Type	Name	Type	SequenceLength	ArrayDimensions	Accessibility	Valid Range	Default	Usage
SpecProperty	N	-	-	-	Parameter	3-6	3	Number of Stages
SpecProperty	M	-	-	-	Parameter	1-2	1	Differential Delay
SpecProperty	R	-	-	-	Parameter	4-8192	4	Decimation Factor
SpecProperty	DIN_WIDTH	-	-	-	Parameter	16	16	Input Data Width
SpecProperty	ACC_WIDTH	-	-	-	Parameter	*	22	Accumulation Width *(2)
SpecProperty	DOUT_WIDTH	-	-	-	Parameter	16	16	Output Data Width
Property	CHIPSCOPE_p	Bool	-	-	Readable, Parameter	Standard	false	Include ISE ChipScope circuit
Property	VIVADO_ILA_p	Bool	-	-	Readable, Parameter	Standard	false	Include Vivado Integrated Logic Analyzer circuit

Component Ports

Name	Producer	Protocol	Optional	Advanced	Usage
in	False	iqstream_protocol	False	-	Complex signed samples (Q0.15 I, Q0.15 Q).
out	False	iqstream_protocol	False	-	Complex signed samples (Q0.15 I, Q0.15 Q).

Worker Interfaces

cic_int.hdl

Type	Name	DataWidth	Advanced	Usage
StreamInterface	in	32	ZeroLengthMessages=true	Signed complex samples
StreamInterface	out	32	ZeroLengthMessages=true	Signed complex samples

Control Timing and Signals

The CIC Interpolation filter HDL worker uses the clock from the Control Plane and standard Control Plane signals.

This worker has a latency of `N*2+1` valid input data clock cycles.

Latency
`N*2+1`

Worker Configuration Parameters

cic_int.hdl

Performance and Resource Utilization

cic_int.hdl

Test and Verification

Two test cases are implemented to validate the CIC Interpolator component:

1. Unity gain response to DC: The CIC Interpolator gain is calculated using the following equation:

$$CIC\ Gain = \frac{(R * M)^N}{2^{CEIL(N * log_2(R * M))}} \tag{3}$$

2. Tone waveform: A waveform containing a tone at 50 Hz is sampled at 1024000/R and processed by the worker. The output data (interpolated waveform) is checked to ensure the 50 Hz tone is present.

For the plots below, a CIC Interpolator with the following parameter set was used: `N`=3, `M`=1, `R`=2048, and `ACC_WIDTH`=49.

For Case #1, the plots below show the input with the I-leg zoomed in the show the amplitude is 32767, and output data with the I-leg zoomed to show an amplitude of 32767, which can be calculated using 3, shown below, and the Q-leg showing the worker delay before reaching it steady-state value.

$$OutputAmplitude = 32767 * \frac{(2048 * 1)^3}{2^{CEIL(3 * log_2(2038 * 1))}} = 32767 * 1 = 32767 \quad (4)$$

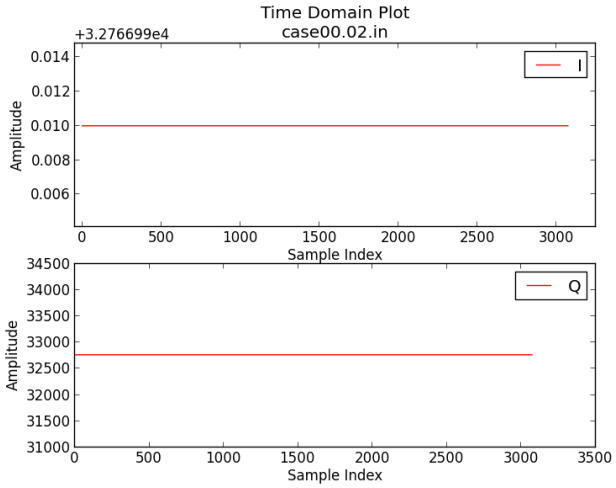

Figure 3: Time Domain: DC with amp=32767

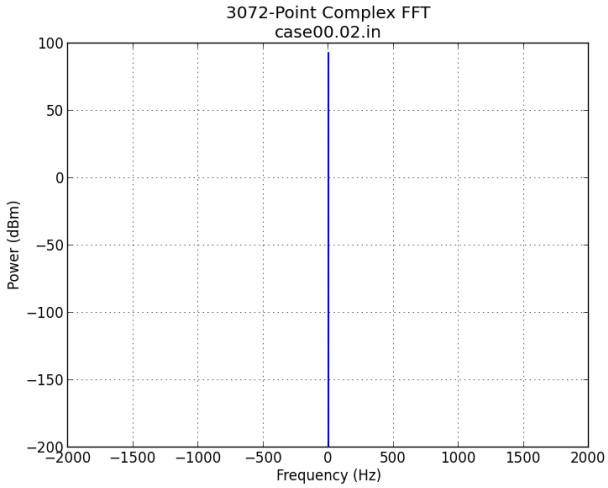

Figure 4: Frequency Domain: 0 Hz

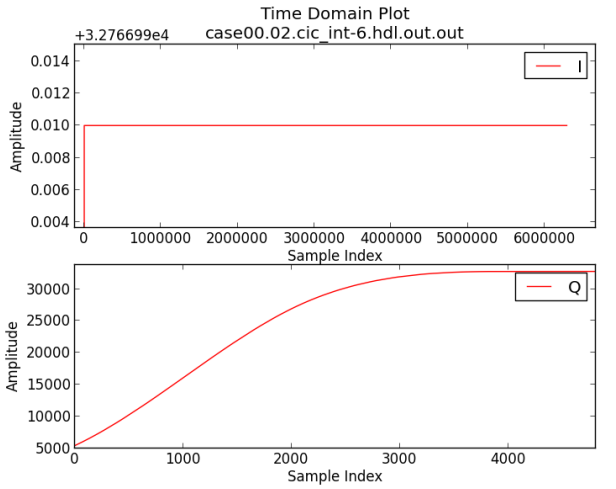

Figure 5: Time Domain: DC with amp=32767

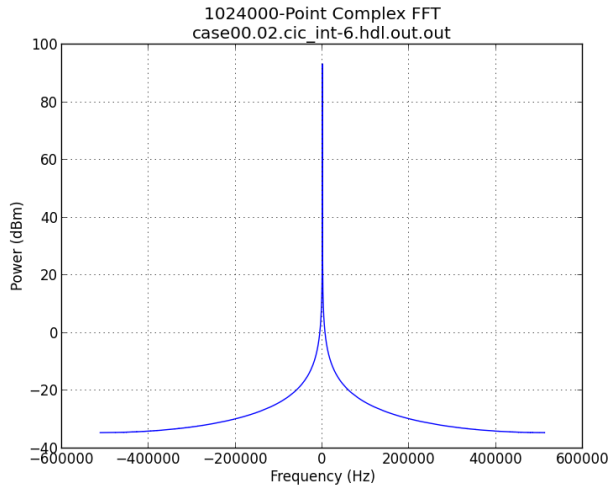

Figure 6: Frequency Domain: 0 Hz

The input time-domain plot below shows the I-leg zoomed into one cycle and Q-leg showing all samples of a 50 Hz tone sampled at 1024000/R=1024000/2048=500 Hz, which results in 10 samples/cycle. The input freq-domain plot shows the generated tone at 50 Hz. The output time-domain plot below shows the I-leg zoomed into approximately two cycles and Q-leg showing all samples of a 50 Hz tone sampled at (1024000/R)*R=(1024000/2048)*2048=1024000 Hz, which results in 20480 samples/cycle. The output freq-domain plot shows the expected tone at 50 Hz.

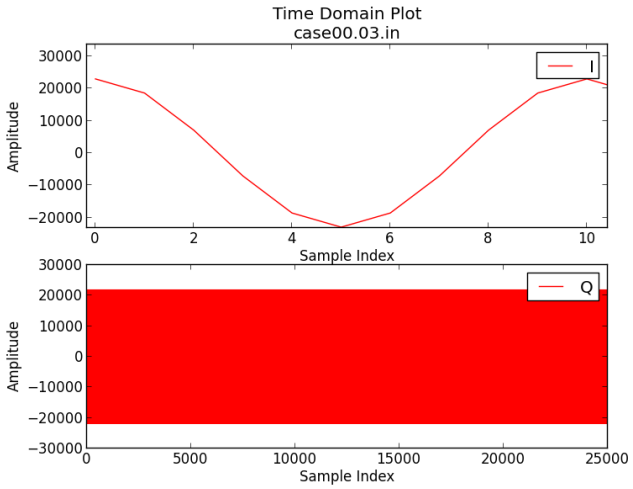

Figure 7: Time Domain

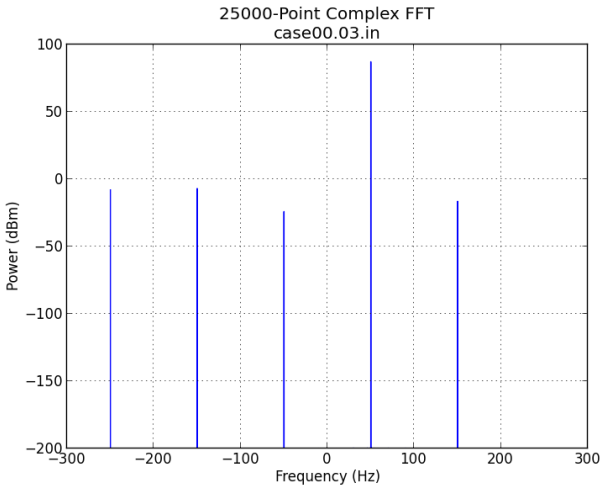

Figure 8: Frequency Domain: 50 Hz

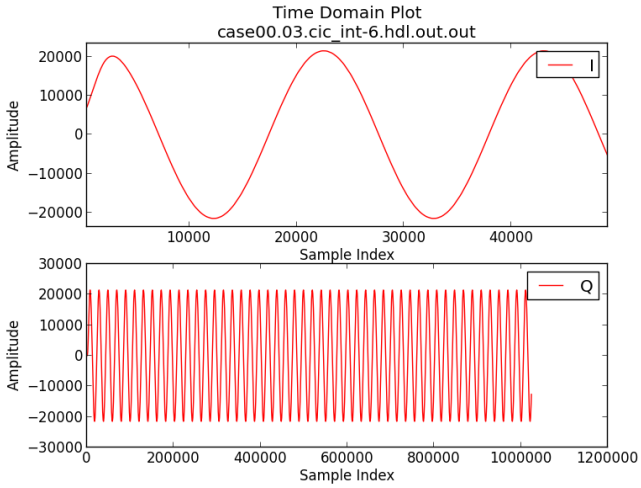

Figure 9: Time Domain

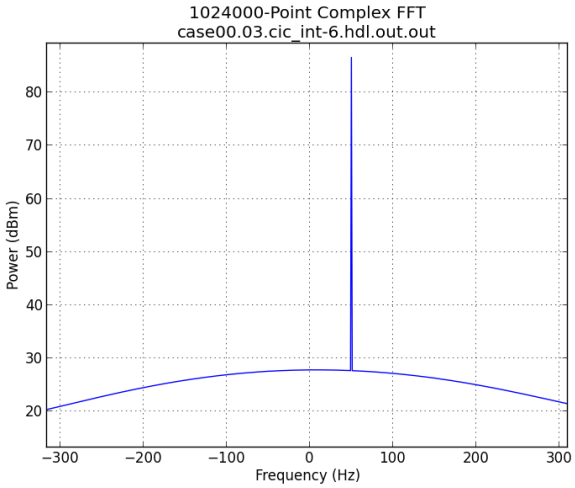

Figure 10: Frequency Domain: 50 Hz

References

(1) Ronald E. Crochiere and Lawrence R. Rabiner. Multirate Digital Signal Processing. Prentice-Hall Signal Processing Series. Prentice Hall, Englewood Cli_s, 1983.
(2) Eugene B. Hogenauer, An Economical Class of Digital Filters for Decimation and Interpolation, IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-29, No. 2, April 1981.
(3) Matthew P. Donadio, CIC Filter Introduction, http://home.mit.bme.hu/ kollar/papers/cic.pdf