CIC Filter Introduction

Matthew P. Donadio
m.p.donadio@ieee.org

18 July 2000

For Free Publication by Iowegian

1 Introduction

As data converters become faster and faster, the application of narrow-band extraction from wideband sources, and narrow-band construction of wideband signals is becoming more important. These functions require two basic signal processing procedures: decimation and interpolation. And while digital hardware is becoming faster, there is still the need for efficient solutions. Techniques found in [CR83] work very well in practice, but large rate changes require very narrow band filters. Large rate changes require fast multipliers and very long filters. This can end up being the largest bottleneck in a DSP system.

In [Hog81], an efficient way of perfoming decimation and interpolation was introduced. Hogenauer devised a flexible, multiplier-free filter suitable for hardware implementation, that can also handle arbitrary and large rate changes. These are known as *cascaded integrator-comb* filters, or CIC filters for short.

This paper sumarizes the findings published in [Hog81]. An overview can also be found in [Fre94]. An extension of CIC filters has been published in [KJW97], and is briefly mentioned here. When in doubt, the reader should refer to these sources.

2 Building Blocks

The two basic building blocks of a CIC filter are an integrator and a comb. An integrator is simply a single-pole IIR filter with a unity feedback coeficient:

$$y[n] = y[n-1] + x[n] \tag{1}$$

This system is also known as an accumulator. The transfer function for an integrator on the z-plane is

$$H_I(z) = \frac{1}{1 - z^{-1}} \tag{2}$$

Using the equations from [OS89] for a single pole system, we can determine that

$$\begin{array}{ll} |H_I(e^{j\omega})|^2 & = \frac{1}{2(1-\cos\omega)} \\ \mathrm{ARG}[H_I(e^{j\omega})] & = -\tan^{-1}\left[\frac{\sin\omega}{1-\cos\omega}\right] \\ \mathrm{grd}[H_I(e^{j\omega})] & = \begin{cases} \text{undefined} & \omega = 0 \\ -\frac{1}{2} & \omega \neq 0 \end{cases} \end{array} \tag{3}$$

The power response is basically a low-pass filter with a -20 dB per decade (-6 dB per octave) rolloff, but with infinite gain at DC. This is due to the single pole at $z = 1$; the output can grow without bound for a bounded input. In other words, a single integrator by itself is unstable.

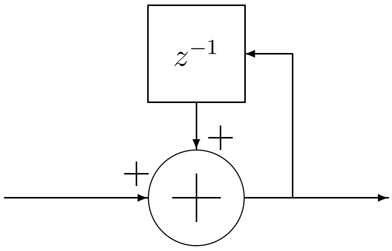

Figure 1: Basic Integrator

A comb filter running at the high sampling rate, f_s, for a rate change of R is an odd-symetric FIR filter described by

$$y[n] = x[n] - x[n - RM] \tag{4}$$

In this equation, M is a design parameter and is called the *differential delay*. M can be any positive integer, but it is usally limited to 1 or 2. The corresponding transfer at f_s

$$H_C(z) = 1 - z^{-RM} \tag{5}$$

Again, we can determine that

$$\begin{array}{ll} |H_C(e^{j\omega})|^2 & = 2(1 - \cos RM\omega) \\ \mathrm{ARG}[H_C(e^{j\omega})] & = -\frac{RM\omega}{2} \\ \mathrm{grd}[H_C(e^{j\omega})] & = \frac{RM}{2} \end{array} \tag{6}$$

When $R = 1$ and $M = 1$, the power response is a high-pass function with 20 dB per decade (6 dB per octave) gain (after all, it is the inverse of an integrator). When $RM \neq 1$, then the power response takes on the familiar raised cosine form with RM cycles from 0 to 2π.

When we build a CIC filter, we cascade, or chain output to input, N integrator sections together with N comb sections. This filter would be fine, but we can simplify it by combining it with the rate changer. Using a technique for multirate analysis of LTI systems from [CR83], we can "push" the comb sections through the rate changer, and have them become

$$y[n] = x[n] - x[n - M] \tag{7}$$

at the slower sampling rate $\frac{f_s}{R}$. We accomplish three things here. First, we have slowed down half of the filter and therefore increased efficiency. Second, we have reduced the number of delay elements needed in the comb sections. Third, and most important, the integrator and comb structure are now independent of the rate change. This means we can design a CIC filter with a programmable rate change and keep the same filtering structure.

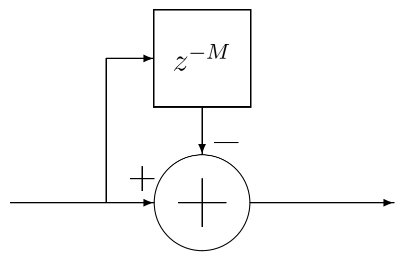

Figure 2: Basic Comb

To summarize, a CIC decimator would have N cascaded integrator stages clocked at f_s, followed by a rate change by a factor R, followed by N cascaded comb stages running at $\frac{f_s}{R}$. A CIC interpolator would be N cascaded comb stages running at $\frac{f_s}{R}$, followed be a zero-stuffer, followed by N cascaded integrator stages running at f_s.

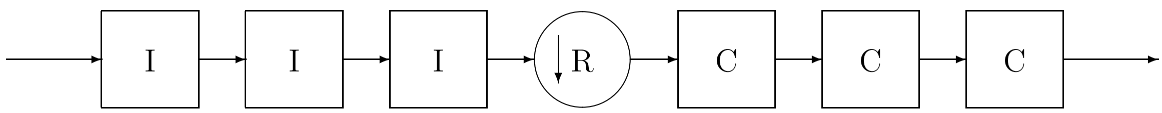

Figure 3: Three Stage Decimating CIC Filter

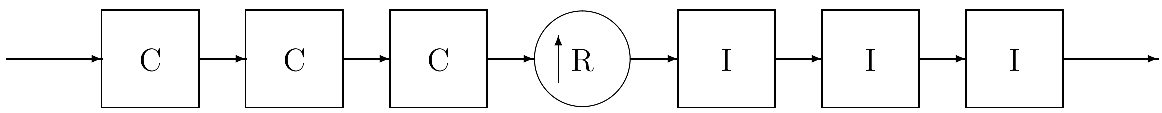

Figure 4: Three Stage Interpolating CIC Filter

3 Frequency Characteristics

The transfer function for a CIC filter at f_s is

$$H(z) = H_I^N(z)H_C^N(z) = \frac{(1 - z^{-RM})^N}{(1 - z^{-1})^N} = \left(\sum_{k=0}^{RM-1} z^{-k} \right)^N \tag{8}$$

This equation shows that even though a CIC has integrators in it, which by themselves have an infinite impulse response, a CIC filter is equivalent to N FIR filters, each having a rectangular impulse response. Since all of the coeficients of these FIR filters are unity, and therefore symetric, a CIC filter also has a linear phase response and constant group delay.

The magnitude response at the output of the filter can be shown to be

$$|H(f)| = \left| \frac{\sin \pi M f}{\sin \frac{\pi f}{R}} \right|^N \tag{9}$$

By using the relation $\sin x \approx x$ for small x and some algebra, we can approximate this function for large R as

$$|H(f)| \approx \left| RM \frac{\sin \pi M f}{\pi M f} \right|^N \text{ for } 0 \le f < \frac{1}{M} \tag{10}$$

We can notice a few things about the response. One is that the output spectrum has nulls at multiples of $f = \frac{1}{M}$. In addition, the region around the null is where aliasing/imaging occurs. If we define f_c to be the cutoff of the usable passband, then the aliasing/imaging regions are at

$$(i - f_c) \le f \le (i + f_c) \tag{11}$$

for $f \le \frac{1}{2}$ and $i = 1, 2, \cdots, \lfloor \frac{R}{2} \rfloor$. If $f_c \le \frac{M}{2}$, then the maximum of these will occur at the lower edge of the first band, $1 - f_c$. The system designer must take this into consideration, and adjust R, M, and N as needed.

Another thing we can notice is that the passband attenuation is a function of the number of stages. As a result, while increasing the number of stages improves the imaging/alias rejection, it also increases the passband "droop." We can also see that the DC gain of the filter is a function of the rate change.

4 Bit Growth

For CIC decimators, the gain G at the output of the final comb section is

$$G = (RM)^N \tag{12}$$

Assuming two's complement arithmetic, we can use this result to calculate the number of bits required for the last comb due to bit growth. If B_{in} is the number of input bits, then the number of output bits, B_{out}, is

$$B_{\text{out}} = \lceil N \log_2 RM + B_{\text{in}} \rceil \tag{13}$$

It also turn out that B_{out} bits are needed for each integrator and comb stage. The input needs to be sign extended to B_{out} bits, but LSB's can either be truncated or rounded at later stages. The analysis of this is beyond the scope of this tutorial, but is fully described in [Hog81].

For a CIC interpolator, the gain, G, at the ith stage is

$$G_i = \begin{cases} 2^i & i = 1, 2, \cdots, N \\ \frac{2^{2N-i}(RM)^{i-N}}{R}, & i = N+1, \cdots, 2N \end{cases} \tag{14}$$

As a result the register width, W_i, at ith stage is

$$W_i = \lceil B_{\text{in}} + \log_2 G_i \rceil \tag{15}$$

and

$$W_N = B_{\text{in}} + N - 1 \tag{16}$$

if $M = 1$. Rounding or truncation cannot be used in CIC interpolators, except for the result, becuase the small errors introduced by rounding or truncation can grow without bound in the integrator sections.

It is now worth revisiting the unstable aspect of the integrator stages. It turns out that it is not a problem. For decimators, integrator overflow is not a problem as long as two's complement math is used and we don't expect an overall system gain > 1. For interpolators, the comb stages and zero stuffing will prevent integrator overflow.

5 Implementation Details

Because of the passband droop, and therefore narrow usable passband, many CIC designs utilize an additional FIR filter at the low sampling rate. This filter will equalize the passband droop and perform a low rate change, usually by a factor of two to eight.

In many CIC designs, the rate change R is programmable. Since the bit growth is a function of the rate change, the filter must be designed to handle both the largest and smallest rate changes. The largest rate change will dictate the total bit width of the stages, and the smallest rate change will determine how many bits need to be kept in the final stage. In many designs, the output stage is followed by a shift register that selects the proper bits for transfer to the final output register. A system designer can use the equation for B_{out} for a decimator and W_{2N} for an interpolator to calculate proper shift values.

For a CIC decimator[1], the normalized gain at the output of the last comb is given by

$$g = \frac{(RM)^N}{2^{\lceil N \log_2 RM \rceil}} \tag{17}$$

This lies in the interval $(\frac{1}{2}, 1]$. Note that when R is a power of two, the gain is unity. This gain can be used to calculate a scale factor, s, to apply to the final shifted output.

$$s = \frac{2^{\lceil N \log_2 RM \rceil}}{(RM)^N} \tag{18}$$

which lies in the interval $[1, 2)$. By doing this, the CIC decimation filter can have unity DC gain.

6 Sharpened CIC Filters

Filter sharpening can be used to improve the response of a CIC filter. This technique applies the same filter several times to an input to improve both passband and stopband

[1]This paragraph is an generalization of equations found in the datasheet for the Harris/Intersil HSP50016.

charecteristics. If $H(z)$ is a symetric FIR filter, then a sharpened version, $H_S(z)$, can be expressed as

$$H_S(z) = H^2(z)[3 - 2H(z)] \tag{19}$$

The magnitude response of a sharpened CIC filter would then be

$$|H(f)| = \left| 3\left(\frac{\sin \pi M f}{\sin \frac{\pi f}{R}}\right)^{2N} - \left(2\frac{\sin \pi M f}{\sin \frac{\pi f}{R}}\right)^{3N} \right| \tag{20}$$

The interested reader is referred to [KJW97] for more details. Please note that it uses different parameters and implements a CIC filter a bit differently than [Hog81].

7 Conclusion

Since their inception, CIC filters have become an important building block for DSP systems. They have found a particular niche in digital transmitters and receivers. They are currently used in highly integrated chips from Intersil, Graychip, Analog Devices, as well as other manufacturers and custom designs. This paper has attempted to summarize key points found in [Hog81] and provide some insight into designs. While many journal submissions are of limited value to an engineer, this paper was written for designers. As such, the reader should try to locate [Hog81] as the definitive reference for CIC filters.

References

[CR83] Ronald E. Crochiere and Lawrence R. Rabiner. *Multirate Digital Signal Processing*. Pretice-Hall Signal Processing Series. Prentice Hall, Englewood Cliffs, 1983.

[Fre94] Marvin E. Frerking. *Digital Signal Processing in Communication Systems*. Kluwer Academic Publishers, Boston, 1994.

[Hog81] E. B. Hogenauer. An economical class of digital filters for decimation and interpolation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-29(2):155–162, 1981.

[KJW97] Alan Y. Kwentus, Zhongnong Jiang, and Alan N. Wilson, Jr. Application of filter sharpening to cascaded integrator-comb decimation filters. *IEEE Transactions on Signal Processing*, 45(2):457–467, 1997.

[OS89] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Pretice-Hall Signal Processing Series. Pretice-Hall, Englewood Cliffs, 1989.