

DECIMATION BY IRRATIONAL FACTOR USING CIC FILTER AND LINEAR INTERPOLATION

Djordje Babic*, Jussi Vesma**, and Markku Renfors*

*Telecommunications Lab., Tampere University of Technology, P.O. Box 553, FIN-33101 Tampere Finland
 **Nokia Research Center, P.O. Box 407, FIN-00045 Nokia Group

ABSTRACT

This paper presents an efficient way to implement flexible multirate signal processing systems with high oversampling ratio and adjustable fractional or irrational sampling rate conversion ratio. One application area is a multi-standard communication receiver which should be adjustable for different symbol rates utilized in different systems. The proposed decimation filter consists of parallel CIC (cascaded integrator-comb) filters followed by a linear interpolation filter. The idea in this paper is to use two parallel CIC filters to calculate the two needed sample values for linear interpolation. These samples occur just before and after the final output sample. This corresponds to a system where the linear interpolation is done at the higher input sampling rate.

1. INTRODUCTION

In multistandard receivers, the hardware should be configurable or programmable for the reception of different types of signals having different symbol rates. After the AD-conversion, utilizing the common delta-sigma AD-conversion principle and high oversampling ratio, the sampling rate is reduced to be a (low) integer multiple of the symbol rate. The problem is that the needed decimation factor can be a difficult fractional number or even an irrational number and, for instance, FIR filters used for integer or fractional decimation cannot be efficiently utilized. Another problem is that there can be disturbing channels that are much stronger (e.g. 80-100 dB) than the desired channel. Therefore, the frequency bands which cause aliasing in decimation should have good attenuation. In addition to these requirements, the overall implementation should be simple because this decimation filter is used in the digital front-end of mobile receivers where the sampling rate is high [1].

This work was carried out in the project "Digital and Analog Techniques for Flexible Receivers" funded by the National Technology Agency of Finland (Tekes). The work was also supported by Graduate School TISE and Academy of Finland.

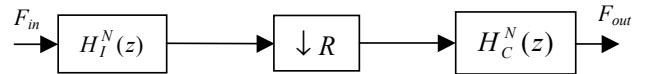


Fig. 1. CIC (cascaded integrator-comb) decimation filter.

Based on these requirements (low complexity and irrational decimation factor), this paper introduces a decimation filter structure which consists of two parallel CIC (cascaded integrator-comb) filters [2], followed by linear interpolation. This structure is easy to implement, e.g., in VLSI circuits, because the CIC filter does not need any multiplications and the linear interpolation requires only one multiplication. The idea is to use two parallel CIC filters that gives samples just before and after the final output sample. Therefore, the linear interpolation is done at the higher input sampling rate.

2. DECIMATION BY INTEGER FACTOR

Cascaded integrator-comb (CIC) filters are commonly used for decimation and interpolation by integer ratio providing efficient anti-image and anti-alias filtering [2]. These filters have a simple regular structure without multipliers. CIC decimation filter (see Fig. 1) consists of N cascaded digital integrator stages operating at high input data rate F_{in} , followed by N cascaded comb or differentiator stages operating at low sampling rate F_{in}/R . Its frequency response is given by

$$H_{CIC}(e^{j\omega}) = e^{-j\omega N(R-1)/2} \left(\frac{\sin(\omega R/2)}{R \sin(\omega/2)} \right)^N, \quad (1)$$

where $\omega = 2\pi f/F_{in}$ is the normalized input frequency.

3. DECIMATION BY NON-INTEGER FACTOR

When the decimation factor is a non-integer number, the filters intended for integer or simple fractional decimation can not be directly used. One solution is to use polynomial-based interpolation filters. Among them, linear interpolation filter has a simple implementation structure, only one multiplication is needed [3].

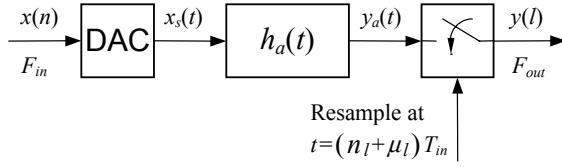


Fig. 2. The hybrid analog/digital model.

Because interpolation is basically a reconstruction problem, polynomial-based interpolation can be analyzed using the hybrid analog/digital model shown in Fig. 2, [3]. In this model, the interpolated output samples $y(l)$ are obtained by sampling the reconstructed signal $y_a(t)$ at the time instants $t = (n_l + \mu_l) T_{in}$. Here n_l is any integer, $\mu_l \in [0,1)$ is the adjustable fractional interval, and T_{in} is the sampling interval of the input signal $x(n)$.

For linear interpolation, the impulse response of the reconstruction filter $h_a(t)$ is a triangular function, and thus, its frequency response is given by

$$H_a(f) = \left(\frac{\sin(\pi f/F_{in})}{\pi f/F_{in}} \right)^2. \quad (2)$$

The digital implementation of the linear interpolation, which needs only one multiplication, can be based on the following equation:

$$y(l) = x(n_l) + [x(n_l + 1) - x(n_l)]\mu_l. \quad (3)$$

4. PROPOSED STRUCTURE FOR NON-INTEGER DECIMATION

Because we consider here a system that can be used also for irrational decimation, the overall decimation factor is determined by

$$R = \frac{F_{in}}{F_{out}} = R_{int} + \varepsilon, \quad (4)$$

where $F_{in} = 1/T_{in}$ and $F_{out} = 1/T_{out}$ are the input and output sampling frequencies, whereas R_{int} is the integer part and $\varepsilon \in [0,1)$ is the decimal part of the overall decimation factor. Figure 3 illustrates the proposed structure for the decimation filter. The input signal $x(n)$ is divided into polyphase components $x_k(m)$ for $k = 0, 1, \dots, R_{int}-1$ by using delay line and parallel CIC filters having the decimation factor of R_{int} . Only few of these R_{int} parallel CIC filters are to work at the same time, hence, the number of parallel CIC filters is reduced. The sampling rate at the output of the CIC filters is F_{in}/R_{int} . After this integer decimation we still have to decimate by $1+\varepsilon/R_{int}$ in order to have overall decimation by R . This final, possibly irrational decimation is done using linear interpolation between some of the two signal pairs $x_k(m)$ and $x_{k+1}(m)$,

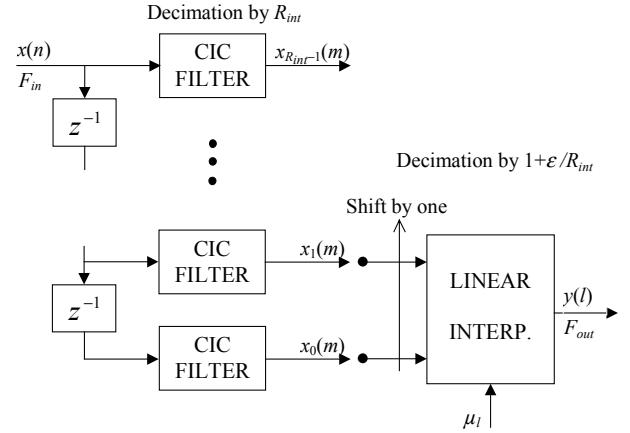


Fig. 3. Model of proposed decimation filter.

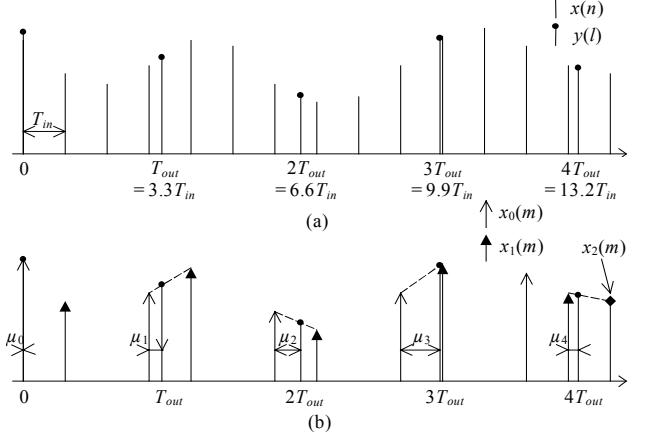


Fig. 4. (a) The input and output samples of the proposed decimation filter for $R = 3.3$. (b) The output samples of the two parallel CIC filter branches $x_0(m)$ and $x_1(m)$.

where \oplus denotes a modulo R_{int} summation.

The linear interpolation block in Fig. 3 is shifted by one branch when needed, according to a certain condition (to be discussed later on). Because of the modulus R_{int} summation mentioned above, the next signal pair for linear interpolation after $x_{R_{int}-2}(m)$ and $x_{R_{int}-1}(m)$ is $x_{R_{int}-1}(m)$ and $x_0(m)$. The fractional interval μ_l is recalculated for each output sample $y(l)$ for $l = 0, 1, 2, \dots$.

The time interval between samples $x_k(m)$ and $x_{k+1}(m)$ equals to T_{in} and, thus, the linear interpolation is done at the high input sampling frequency F_{in} . This means better image attenuation. The CIC filters attenuate the disturbing channels and noise which would cause aliasing in linear interpolation. In other words, the CIC filters and linear interpolation take care of anti-aliasing and anti-imaging property, respectively [1].

As an example, Fig. 4 shows input and output signals as well as some of the polyphase signals for the decimation factor of $R = 3.3$. After four first samples, the next output sample $y(4T_{out})$ falls outside the interval $x_0(m)$ and $x_1(m)$. When this occurs, the linear interpolation is shifted

by one interval (as indicated by an arrow in Fig. 3) and the interpolation is done between signals $x_1(m)$ and $x_2(m)$.

4.1. The frequency response of the overall system

The overall frequency response of the decimation filter structure in Fig. 3 is a product of the frequency responses of the parallel CIC filters and linear interpolation filter. Note that the former response is a periodical whereas the latter is not. The frequency response of the parallel CIC filter stage is simply the same as the response of one CIC filter given by Eq. (1), where, however, R has to be replaced by R_{int} . Since the linear interpolation is done at the higher input rate F_{in} , its frequency response is given by Eq. (2). Consequently, the overall zero-phase frequency response of the proposed decimation filter, relative to the input sampling frequency, is given by

$$H_T(\omega) = H_{CIC}(\omega)H_a\left(\frac{\omega F_{in}}{2\pi}\right) = \left(\frac{\sin\left(\frac{\omega R_{int}}{2}\right)}{R_{int} \sin\left(\frac{\omega}{2}\right)}\right)^N \left(\frac{\sin\left(\frac{\omega}{2}\right)}{\frac{\omega}{2}}\right)^2. \quad (5)$$

where $\omega = 2\pi f/F_{in} = 2\pi f/(RF_{out})$.

4.2. Design examples

In this example, the bandwidth of the input signal is $f_p=0.001F_{in}$ and decimation factor $R=34^{1/34}$. It is required that the frequency bands that cause aliasing to the frequency band of the input signal are attenuated at least by $A_s=80$ dB and the passband distortion is less than $\delta_p=0.01$ (0.086 dB).

These requirements are met by a proposed type of decimation filter having CIC filter of order $N=3$. The overall frequency response of the proposed decimation filter, as well as the responses of the parallel CIC filters and linear interpolation filter, are shown in Fig. 5. Figure 6 presents the bands that cause aliasing to the desired band. As it can be seen, the minimum attenuation of these bands is 84.4 dB. Because the over-sampling factor is still high after decimation, the worst case passband distortion caused by the proposed filter structure is only 0.06 dBs. Figure 7 shows the minimum attenuation of the aliasing bands in a range of the fractional decimation factor ranging from 32 to 34. The minimum attenuation occurs at the edge of the first aliasing band. As it can be seen from Fig. 7, the minimum attenuation of the aliasing bands depends on ε . If ε is close to unity, better attenuation is obtained by expressing R as $R=R_{int}-\varepsilon$. The analysis and filter structures can be easily extended for this negative value of ε as well.

5. IMPLEMENTATION

Linear interpolation is done between two samples (see Fig. 3). Therefore, only two CIC filter branches are needed in

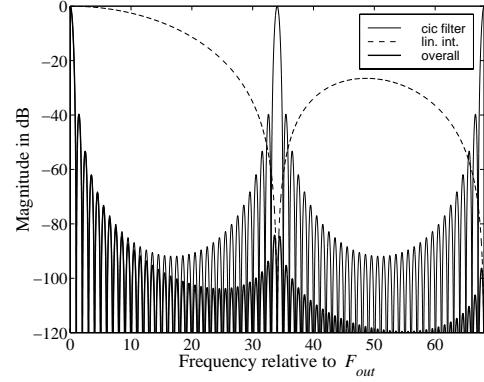


Fig. 5. Frequency response of the CIC decimation filter, linear interpolator filter, and the overall response.

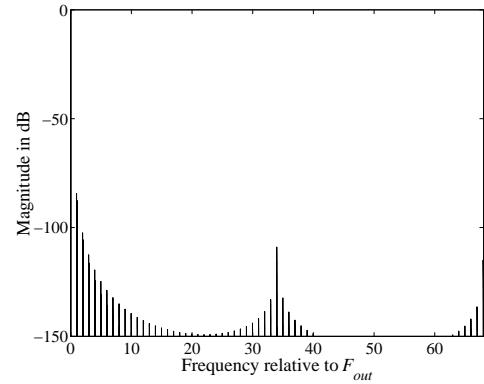


Fig. 6. Aliasing bands of the overall structure.

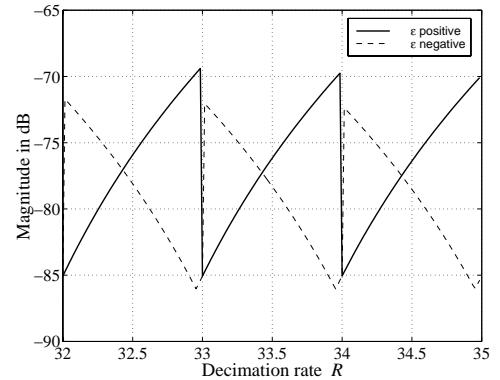


Fig. 7. The maximum value of the aliasing bands for positive and negative ε .

the implementation plus some extra branches for preparing the future samples. The number of extra branches depends on the order of the CIC filter N and the fractional part of the decimation rate ε . Furthermore, the integrator part with the transfer function $H_i^N(z)$ can be shared among the branches. This principle is shown in Fig. 8.

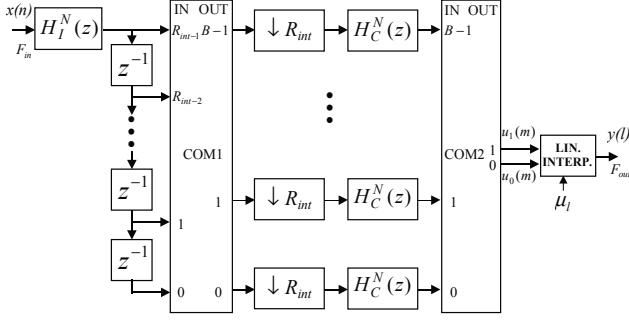


Fig. 8. Implementation structure for the proposed decimation filter.

The commutators COM1 and COM2 are used to select the correct input branch for the B comb sections and for the linear interpolation, respectively. In general case, for any ε , the number of needed comb filter branches is $B=N+2$. Two of them are used for calculating the output samples and the remaining N branches are used for initializing the state-variables of the branches needed later.

In Fig. 9(a), the control logic algorithm is presented for the proposed structure. The first step in this algorithm is the initial set up of the index value l as well as the fractional interval $\mu_0=0$. The next step is the interpolation which is expressed by

$$y(l) = I(\mu_l, u_0(m), u_1(m)), \quad (6)$$

where $I(\cdot)$ denotes the linear interpolation between the samples $u_0(m)$ and $u_1(m)$ with the fractional interval of μ_l . After interpolation, l is incremented by one and the fractional interval can be computed by

$$\mu_l = \mu_{l-1} \oplus \varepsilon, \quad (7)$$

with the initial condition $\mu_0=0$. In Eq. (7), the modulo summation indicates that only the decimal part of the result is used. According to Eq. (7), the calculation of μ_l can be implemented by using an adder with fixed point arithmetic. The shifting in the interpolation has to be performed whenever an overflow occurs while calculating μ_l . Therefore, the overflow bit c_l of the adder can be used as a shifting condition. The shift block in Fig. 9 means that the interpolation is shifted by one branch (see Fig. 3). This shifting operation is implemented using the commutators COM1 and COM2 as it is shown in Fig. 8. The commutator COM1 has R_{int} inputs and B outputs. The commutator COM2 has B inputs and two outputs. In order to describe the function of the commutators we use variables for the outputs of the commutators. There are B variables for the outputs of COM1 denoted by OUT_i^1 for $i=0,1,\dots,B-1$ and two variables for COM2 denoted by OUT_i^2 for $i=1$ and 2. The values of these variables determine what input sample is connected to the i^{th} output.

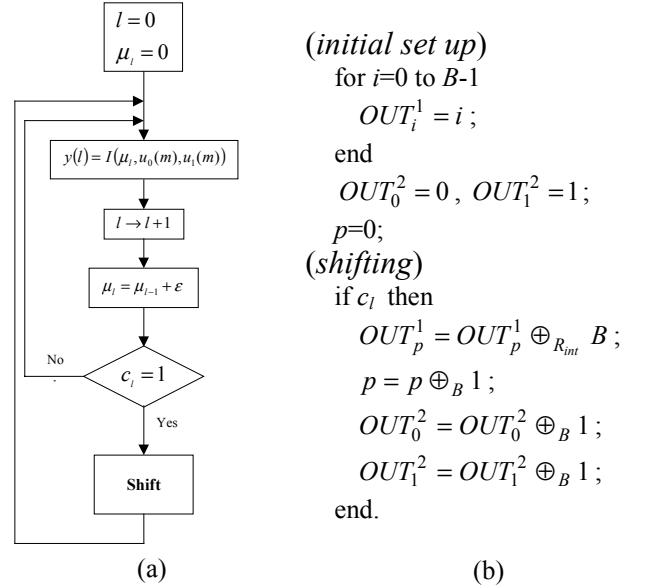


Fig. 9. (a) The state flow diagram of the control logic. (b) Algorithm for switching of COM1 and COM2. Here \oplus_i denotes modulo i summation.

The switching algorithm for COM1 and COM2 is given in Fig. 9(b). When shifting occurs, only one output of COM1, numbered by p , should be switched to the another input. Hence, only the value of the variable OUT_p^1 is changed. In COM2, when shifting occurs, both output branches should be switched to the another input. This is done because the order of the interpolator inputs must be preserved.

6. CONCLUSIONS

We have proposed the structure for an arbitrary ratio decimation that consists of parallel CIC filters and linear interpolation. The whole structure has only one multiplier. Further, the linear interpolation is performed at the lower sampling rate. However, the performance of the proposed structure corresponds to the system where the linear interpolation is done at the higher input sampling rate.

REFERENCES

- [1] T. Hentschel and G. Fettweis, "Software radio receivers," Chapter 10 in *CDMA Techniques for Third Generation Mobile Systems*, edited by F. Swarts, P. van Rooyen, I. Oppermann, and M. P. Lötter, Kluwer Academic Publishers, 1999.
- [2] E. B. Hogenauer, "An economical class of digital filters for decimation and interpolation," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-29, pp. 155-162, April 1981.
- [3] J. Vesma, *Optimization and Applications of Polynomial-Based Interpolation Filters*; Doctoral Thesis, Tampere University of Technology, Publications 254, 1999.