Compensation of Non-Linearly Gain of Tunable Decimation CIC Filters

Kvachev Maxim, Puzyrev Pavel
Omsk State Technical University
Omsk, Russia
redmaxer@yandex.ru

Abstract – In this article described the procedure of design CIC decimation filters with variable decimation factor used in DSP with more economical and operating efficiency. The described method of compensating amplitude degradation due variable decimation factor allows to reduce FPGA resources or die area in comparison with other methods. The considered filter has 2x greater efficiency of FPGA resource usage and 60% higher operating frequency. Recommendations for design, application use and a design example with the necessary requirements is given.

Index Terms – digital filtering; FPGA; digital down converter; CIC; resampling

I. INTRODUCTION

In the process of implementing digital receivers, it is necessary to lower the sampling rate of the signal at the mixer output for subsequent processing at a low frequency. A resampling filter performs this operation, which is part of the DDC (Digital Down Converter).

In systems for resampling signals with large values of changing the sampling rate, CIC (Cascaded Integrator-Combs) filters are commonly used. A large number of authors [1 – 6] describes these filters, and only recently, they have gained spread due a rapid increase in the use of SDR (Software-Defined Radio).

The DDC transfers the signal spectrum to a lower frequency and performs a resampling of the signal for transmission to subsequent digital processing devices.

Based on these filters, performing filtering on the overlap of the signal spectra when a change in the sampling rate. Their distinctive feature from other types of filters is the ability to change the sampling rate in a wide range (mostly from 8 to 8192, but it is also possible to extend the range up to 32000 [7]), and in the absence of multiplication operations. These advantages make CIC filters attractive for implementation in FPGA (Field-Programmable Gate Array), DSPs (Digital Signal Processors) and IC (Integrated Circuit), is the processing at a sufficiently high frequency.

When implementing CIC filter on FPGA is necessary to reduce the word length of the intermediate calculations in order to save logic modules (ALM), and implement variable decimation factor, which entails a problem of change signal level at the filter output nonlinearly.

In [1] describes the possibility of design CIC filters with discarding of least significant bits on each stage of filtering, with a small accuracy loss.

Most of software for generate HDL description of CIC filters gives a feature to change decimation factor in the runtime for full precision internally computations only. Also, this feature is available with reduced registers length, but resource usage of device in many cases is ineffective.

The main difficult of compensation unwanted output signal level decreasing is impossibility amplifying signal at output of filter, because for large decimation factor deviation useful signal may be lost.

Section III describes a theory for design required CIC filter, with amplitude degradation compensation of signal on the output of filter due variable decimation factor.

In Section IV outlined results of simulating and comparison of designed in this paper and automatically generated CIC filters, by resource of FPGA used for implement considered filters and maximum operating frequency also.

II. PROBLEM STATEMENT

Due change decimation factor of CIC filter, amplitude of output signal changing non-linear and for large deviation of decimation factor from initial, useful information may corrupted.

The objective of this work is to provide a design methodology for a CIC filter with variable decimation factor, a reduced internal registers bit length by Hogenauer pruning theory and compensation of non-linearly changing CIC gain.

To assess the FPGA resource usage and give recommendations when implementing CIC using the following method.
III. THEORY

CIC filter has a cascaded $N$ ideal integrator stage and combs filters. Basic structure showed in Fig. 1.

Transfer function of $N$ order CIC filter [1] is

$$H(z) = H_T^N(z)H_C^N(z) = \frac{(1 - z^{-RM})^N}{(1 - z^{-1})^N},$$

(1)

where $R$ – decimation factor, $M$ – delay in comb section.

Used a Z-transform how it’s shown in [8] from (1) let’s move on to magnitude response of CIC filter $N$ order (2), response is shown by Fig. 2

$$H_{CIC}(f) = \left[\frac{\sin(\pi fM)}{\sin(\pi f/R)}\right]^N,$$

(2)

where $f$ – sample rate of the input signal.

For many cases, when $R$ is large, use a simplified equation $\sin(x) \approx x$ and magnitude response (2) approximated as shown in (3)

$$H_{CIC}(f) = \left[\frac{RM}{\pi Mf}\right]^N \text{ for } 0 \leq f < \frac{1}{M}.$$  

(3)

Filter reject all harmonics upper then $f_{\text{stop}}$ and can be find as

$$f_{\text{stop}} = \frac{f_s}{RM}.$$  

(4)

Zeros of filter placed on frequencies multiplies $f_{\text{stop}}$ and defined by equation (5)

$$f_{\text{zeros}} = k_if_{\text{stop}}, \ 0 < k_i \leq RM.$$  

(5)

How it is seen in Fig. 2, on zero-frequency CIC filter has a large gain, what is the reason unwanted growth internally register and adder length.

Maximum output magnitude of CIC filter $G_{\text{max}}$ due zero-frequency gain can be find by

$$G_{\text{max}} = (RM)^N.$$  

(6)

Maximum register length with sign bit (6) is

$$B_{\text{max}} = \left[\log_2 RM + B_{\text{in}}\right].$$  

(7)

To reduce large internally bit word length to the required output width, we can discard the LSB’s in the amount is

$$B_{\text{out}} = B_{\text{max}} - B_{\text{out}}.$$  

(8)

However, solution (8) generate output signal distortion, a white noise with a large amplitude [9].

Method of reduce register widths suggested by Hogenauer in [1] is equal mean error from $2N$ error sources to error produced by last error source on $2N+1$ stage

$$\sigma^2_{T_j} \leq \frac{1}{2N} \sigma^2_{T_{2N+1}},$$  

(9)

where $\sigma^2_{T_{2N+1}}$ – mean error variance produced by discarded bits in $2N+1$.

Truncating $B_j$ bits on internal stage produced white noise with mean is

$$E_j = 2^{B_j}.$$  

(10)

and variance is

![Fig. 2. Frequency response of CIC filter with $R = 2048$, $N = 4$, $M = 1$ and $f_s = 100$ MHz](image-url)
\[ \sigma_j^2 = \frac{1}{12} E_j^2, \quad (11) \]

Variance of noise introduced by truncated bits on each stage and can be find as

\[ \sigma_j^2 = \sigma_j^2 F_j^2, \quad (12) \]

where \( F_j^2 \) is mean error gain on \( j \)-stage

\[ F_j^2 = \begin{cases} \sum_{k=0}^{(RM-1)N+j-1} h_j^2(k), & j = 1, \ldots, N \\ \sum_{k=0}^{2N+j-1-j} h_j^2(k), & j = N + 1, \ldots, 2N \\ 1, & j = 2N + 1 \end{cases}, \quad (13) \]

Where \( h_j(k) \) is

\[ h_j(k) = \begin{cases} [k:RM] \sum_{i=0}^{N} (-1)^i \binom{N-j+k-REMl}{k-REMl}, & j = 1, 2, \ldots, N \\ (-1)^i \binom{2N + 1 - j}{k}, & j = N + 1, \ldots, 2N \end{cases}, \quad (14) \]

So used the foregoing equations we can achieve the next expression, detailed in [1], number of bits will be discarded on each stage with reasonable noise variance can be find as

\[ B_j = \left[ -\log_2 F_j + \log_2 \sigma_j^2 + \frac{1}{2} \log_2 \frac{6}{N} \right]. \quad (15) \]

Change of decimation factor leads to change of CIC filter gain non-linearly as (6) and Fig. 3 are demonstrated this. For keep amplitude of signal in a certain range and save information of useful signal, compensation is variable gain is needed.

Due to the fact then in runtime, register width remains unchanged, it is necessary to amplify input signal. Amplifying on the final stage or any of internally stage is ineffective.

Amplifying on internal stage is ineffective because amplify internal computations with decreased register width becomes amplifying white noise with large variance \( \sigma_j^2 \).

Therefore, last way to compensation changing gain of filter is amplify signal on the input of first stage by

\[ K_B = N \log_2 \frac{R_{base}^M}{R_{new}}, \quad (16) \]

where \( R_{base} \) is the decimation factor used for design CIC filter, \( R_{new} \) is the decimation factor different from initial.

Compensation may be applied by multiply of input signal and gain coefficient (14 case 1), or by bit-shift register (14 case 2)

\[ K_B = \begin{cases} 2^{K_B}, & \text{if multiplier} \\ \left\lfloor \frac{K_B}{2} \right\rfloor, & \text{if bit - shifting} \end{cases}. \quad (17) \]

From (16) and (17) becomes clear, bit-shift method can fully compensate decreasing of amplitude when decimation factor is power-of-2.

Compensation method by multiplier has a better precision then bit-shift register, but high frequency multipliers are needed or free space of IC, for few cases, it is unacceptable.

### IV. EXPERIMENTAL RESULTS

#### A. Computer Modeling Results

For research CIC filter, will design one with follow requirements \( f_s = 100 \text{ MHz}, f_{out} = 50 \text{ kHz}, N = 4, M = 1, B_m = 16, B_{out} = 20 \).

By (4) find required decimation factor \( R = 2000 \). For next computations take \( R = 2048 \).

Bit-length of non-truncated adders can be found by (7) \( B_{max} \) = 60 and total truncated bits by (8) \( B_{N+1} = 40 \).
By (11), (12), (13) and (14) define bits truncated at each $2N$ stages 0, 11, 21, 30, 35, 36, 37, 38 and 40 (final $2N+1$ truncation).

Compensation coefficient $GAIN$ can be found by (16) and (17) every time when decimation factor is changed, or in the design process developer find compensation coefficients for factors $R$ and save his in the memory.

Structure of CIC filter with pruning by Hogenauer and compensation of decreasing gain showed in Fig. 4.

Gain compensation may be implement by two ways: bit-shift register or multiplier.

Dependence of gain by decimation factor with different compensation methods, showed in Fig. 5.

Markers on the Fig. 5 pointing when both methods give a same compensation to output signal.

Due limited bit width of multipliers, compensated gain has some unevenness showed in Fig. 6.

B. HDL-Synthesis Results

According to the above is synthesis HDL-description for CIC filters with variable decimation factor from various vendors.

Resource usage of FPGA and maximum operating frequency for various filters given in Table I.

To perform this part was used a following instruments MATLAB HDL-Coder, Quartus Prime 18.1 and Timing Analyzer. The operation of filters for crystals of different speeds is guaranteed at the frequencies indicated in the column Slow, according to [10].

V. DISCUSSION OF RESULTS

The dependence shown in Fig. 5 confirm conclusion in (17) that is fully compensation of amplitude degradation of signal by shift-bit register is possible only when decimation factors be equal power-of-two.

When implement CIC filters with compensation of decreasing gain in hardware with large relation $R_{new}$ to $R_{old}$, due limited bit width of DSP-multipliers, is convert to cascaded mult/adders is may needed, how is describes in [11] and [12].

In Section IV.B, the information on the use of FPGA resources show that the use of this compensation technique does not critically affect the maximum operating frequency of the filter, but it saves a significant amount of ALM and memory blocks compared with similar solutions.
Computer modeling of designed CIC filter confirmed performance of compensation method described in this paper. However most accurate method by multiplier-compensated is most expensive for some applications, because multiplier is operating on high frequency of input sample rate and use a few DSP-blocks.

Comparing filter what been designed in this work with filters suggested by vendors of software or IP cores for automatic generate HDL-description of CIC filter shows efficient of compensation method suggested in this paper. Filter designed in this work more economical for resource usage of FPGA and have maximum operating frequencies.

In the process of design CIC filter with large deviation $R_{new}$ to $R_{base}$ is bit width of multiplier may spread and convert multiplier to cascaded form may needed, so choice of compensation mode is necessary and must to base on free resources of the device.

**REFERENCES**

[1] E. B. Hogenauer “An Economical Class of Digital Filters for Decimation and Interpolation” in IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-29, pp. 155-162, April 1981

[2] U. H. Meyer-Bäuse, “Hogenauer CIC Filters” in Digital Signal Processing with FPGA, 3rd ed., Springer, Berlin, Germany, ch. 5, sec. 3, pp. 256–272, 2007.

[3] R. G. Lyons. (2005, March 31) Understanding cascaded integrator-comb filters [Online]. Available: https://www.embedded.com/design/configurable-systems/4006446/Understanding-cascaded-integrator-comb-filters

[4] R. G. Lyons, “Cascaded Integrator-Combs Filters” in Understanding Digital Signal Processing, 2nd ed., Prentice Hall, New Jersey, ch. 10, sec. 5, pp. 397–407

[5] M. P. Donadio. (2010, July 18) CIC Filter Introduction [Online]. Available: http://dsfguru.com/files/cic.pdf

[6] S. A. Bahurin. (2010, June 9) Hogenauer CIC Filters and characteristics [Online]. Available: http://www.dsplib.ru/content/filters/ch9/ch9.html (in Russian)

[7] CIC IP Core User Guide, 6th ed., Intel, 2017

[8] S. A. Bahurin. (2010, June 10) Laplace transform. Z-transform [Online]. Available: http://www.dsplib.ru/content/filters/ch9/ch9.html (in Russian)

[9] A. V. Oppenheim et al., “Round-off Noise In Digital Filters” in Discrete-Time Signal Processing, 2nd ed., Prentice Hall, New Jersey, ch. 6, sec. 8, pp. 391–402

[10] Guaranteeing Silicon Performance with FPGA Timing Models, 2nd ed., Intel, 2017

[11] DSP Blocks in Starix-IV Devices”, 2nd ed., Altera, 2004

[12] Using Embedded Multipliers in Spartan-3 FPGAs, 2nd ed., Xilinx, 2003

**TABLE I. RESOURCE USAGE OF FPGA AND MAXIMUM OPERATING FREQUENCY**

| CIC Filter Type | ALM | Registers | DSP Blocks | Memory | Slow Fmax, MHz | Fast Fmax, MHz |
|-----------------|-----|-----------|------------|--------|---------------|---------------|
| Without compensation and bit pruning | 509 | 1508 | 0 | 0 | 254 | 372 |
| Bit shifted compensation with pruning | 207 | 403 | 0 | 0 | 281 | 504 |
| Multiplier compensation with pruning | 157 | 387 | 2 | 0 | 275 | 495 |
| Intel FPGA CIC IP Core. Multiplier compensation with pruning | 363 | 686 | 0 | 868 | 170 | 315 |
| MATLAB HDL Coder. Bit shifted compensation with full precision | 1009 | 581 | 0 | 0 | 62 | 156 |

VI. CONCLUSION

In this paper have been describes a method of design CIC decimation filters with decreased register and adders bit-length by Hogenauer pruning theory, programmable decimation factor and compensation non-linearly CIC filter gain changing.

Computer modeling of designed CIC filter confirmed performance of compensation method described in this paper. However most accurate method by multiplier-compensated is most expensive for some applications, because multiplier is operating on high frequency of input sample rate and use a few DSP-blocks.

Comparing filter what been designed in this work with filters suggested by vendors of software or IP cores for automatic generate HDL-description of CIC filter shows efficient of compensation method suggested in this paper. Filter designed in