Nataliya Maslennikova
Computer Engineering Department
Politechnika Koszalińska
E–mail: olegmas@wp.pl
Anatolij Sergiyenko
Computer Engineering Department
National Technical University of Ukraine„KPI"
E–mail: aser@comsys.kpi.ua

Tunable Infinite Impulse Responce Filters in FPGA

Keywords: IIR, FPGA, allpass filter, structure synthesis

Introduction

Infinite impulse response (IIR) filters are often used in digital signal processing (DSP). To adjust the IIR filter parameters dynamically different techniques are used. The most common method is based on calculating a set of coefficients, which are stored in ROM. The IIR filter tuning is the complex transition process that distorts the output signal proportionally to the quality factor of the filter and the step of the parameter changing [1].

The tunable narrow-band filters are usually based on the multistage decimation, and interpolation filters, and often use the two-step heterodyne frequency transfer [1,5]. But the combination of narrowband, and broadband filters require complex technical solutions.

IIR filters based on a field programmable gate array (FPGA) have high throughput and quality. But to implement the tunable IIR filter the usual computer system should include a microcontroller to calculate the coefficients, or coefficient ROM of a large volume, which significantly increases the hardware costs.

In the work the implementation of a dynamically tunable IIR filter is proposed, which is based on the phase filter having a smooth exchange of its parameters through the use of the frequency masking effect and filters with multiple delays.

1. Masking filters

The cascaded connection of the filter stages has the resulting frequency response, which is the intersection of frequency responses of these stages. In this cascaded filter the frequency response of one stage can mask frequency responses of other stages as is shown in Fig. 1. Because of the masking effect, the resulting filter, consisting of simple filter stages, has the high-quality frequency response H_{res} [2,3].

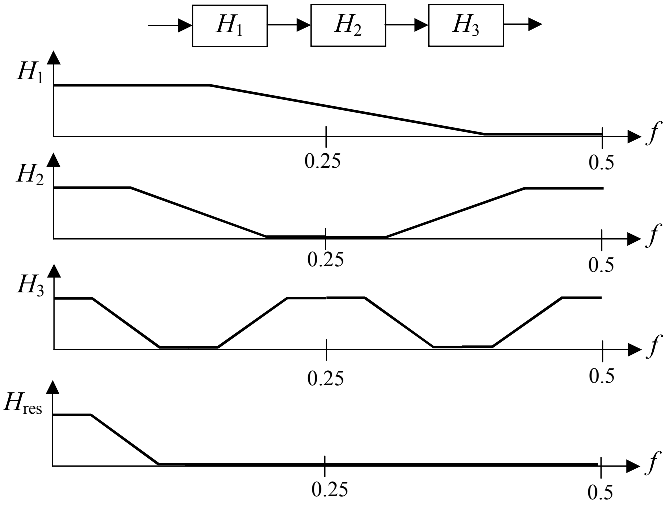

Fig. 1. Frequency response of the three-staged filter

2. Multiple delays filter

Each variable z^{-k} in the transfer function $H_0(z)$ corresponds to a delay of k cycles in the filter signal graph, or to a chain of k delay registers in the filter structure. If the number of delay registers in the IIR filter is increased to n times, then the filter transfer function is exchanged as $H_n(z) = H_0(z^n)$. FRC of the modernized filter has the same shape as that of the filter – prototype $H_0(z)$, but in the range of $0 - f_S$ it is repeated n times, where f_S is the sampling frequency [4]. The diagrams in Fig. 1 represent $H_1 = H_0(z)$, $H_2 = H_0(z^2)$, $H_3 = H_0(z^4)$.

3. IIR filters based on the phase filters

The phase filter has a transfer function magnitude, which is equal to $|H(z)| = 1$. If the signals from two-phase filters are added, then the resulting signal is suppressed at the frequencies, for which the phase difference is equal to π. The resulting transfer function is

$$H_S = (H_1(z) \pm H_2(z))/2, \qquad (1)$$

and corresponds to a low pass filter (LPF), high pass filter (HPF), bandpass, or notch filter according to the summation sign, and the order of the phase filters [5]. Consider the transfer function $H_1(z)$ has the phase shift, which is equal to π in some frequency f_R, and $H_2(z) = z^{-1}$. Then we have LPF when the sign is plus, and HPF when we have the opposite sign, and f_R is the cutoff frequency.

IIR filters based on the phase filters are distinguished by high stability, low immunity to the coefficient precision, high linearity of the pass band characteristic, and high speed. The parameters of frequency response, such as the position of the cutoff frequency, the slope of the transition band, are directly dependent on the filter coefficients [1,5].

4. Dynamically tuned LPF

To construct LPF, which is tunable over a wide frequency range, the filter structure is proposed, which includes the stages of masking filters $H(z),\ldots, H(z^k)$ and a stage of the forming filter $H(z^k,a,b)$, as it is shown in Fig. 2. The frequency response $H(z)$ is the response of the bireciprocal filter [5]. The example of the frequency response of the filters $H(z)$, $H(z^2)$, and $H(z^4)$ is shown in Fig. 1. The transfer function of a first stage is based on the phase filter:

$$H(z) = 0{,}5z^{-1} + 0{,}5\frac{a + b(1+a)z^{-1} + z^{-2}}{1 + b(1+a)z^{-1} + az^{-2}}, \qquad (2)$$

where the first, and second terms are $H_1(z)$ and $H_2(z)$ in the equation (1). The coefficient b adjusts the cutoff frequency f_R, and the coefficient a specifies the width of the transition band Δf [5], where

$$b = \cos(2\pi f_R); \qquad a = (1 - t)/(1 + t); \qquad t = \mathrm{tg}(\pi\Delta f). \qquad (3)$$

Thus, by changing b in (2), the cutoff frequency is adjustable in the range of $(0{,}1 - 0{,}4)f_S$ providing the suppression up to 50 dB in the suppression band. It should be noted, that all of the filter stages are calculated by a single algorithm but with different coefficients a, b, and the number of delay registers k.

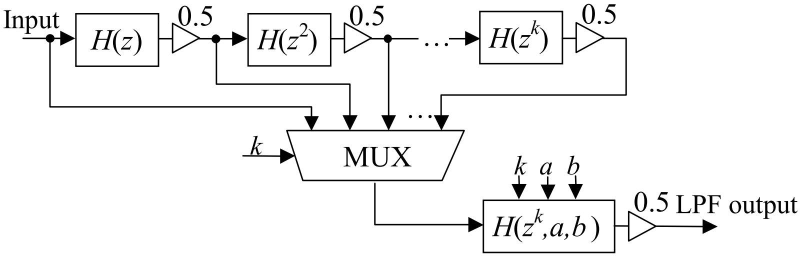

Fig. 2. Structure of the low pass filter with the tunable parameters

5. Dynamically tunable HPF

A narrow band HPF can be obtained by subtracting the LPF results, described above, from the delayed input signal. However, due to the non-ideal low pass transfer characteristic, the attenuation of such a filter does not exceed 30–40 dB.

Consider the filter with the property of double complementary filter characteristics. Two filters are called complementary if the pass band frequency response of one filter corresponds to the stop band frequency response of another filter. If additionally the total energy of the output signals of these filters is equal to the power of the input signal, then the filters are called double complementary ones. And the fact is that LPF, and HPF, given by the formula (1), are double complementary filters [5]. Thus, adding such filter output signals, we get the copy of the input signal not taking into account the calculation errors.

Fig. 3 shows the structure of HPF with the complementary LPF, which is constructed using the double complementary filter property. The first filter stage separates the input signal into two bands: the low pass band (upper branch) and high pass band (lower branch). The second stage does the same calculations with the low pass band output of the first stage. The resulting signal of the second stage is formed as the sum of the high-frequency signal, and the delayed high pass signal of the first stage. The output signal of HPF in Fig. 3 has a bandwidth of $7{\cdot}f_S/16$. The HPF bandwidth is tuned as well as in the LPF described above. The coarse passband is selected as the i-th stage output, and the precise cutoff frequency f_R is regulated by the coefficients of the final stage.

The bandpass filters can be constructed as the combination of LPF, and HPF, which is described above. Two outputs of the filter in Fig.3 can be effectively used in the filter bank design.

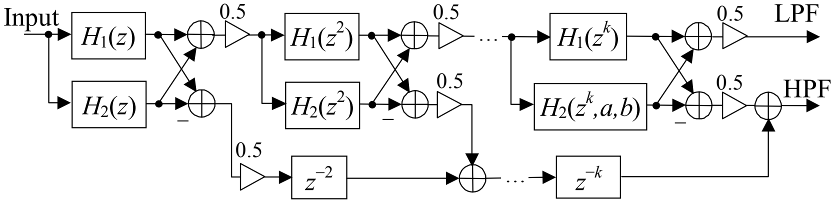

Fig. 3. HPF structure based on the double complementary filters

6. Example of LPF implementation

To implement the dynamically tunable LPF, the structure in Fig. 2 was chosen. To obtain the stopband attenuation of over 80 dB, and the transition band slope of 100 dB per octave the filter stage has to implement the $10k$-th order transfer function

$$H(z^k)=\left[z^{-2k}+\frac{a+b(1+a)z^{-k}+z^{-2k}}{1+b(1+a)z^{-k}+az^{-2k}}\cdot\frac{c+z^{-k}}{1+cz^{-k}}\right]^2. \quad (4)$$

Here, the coefficient c infers the cutoff frequency f_R, and slightly smoothes out the frequency response. The approximate value of the cutoff frequency and the transition band slope of the filter are defined as in the equations (3). In a single filter stage, which is implementing the transfer function (4) for k = 1, the cutoff frequency is tuned in the range of (0,125 – 0,4)f_S. Three stages of the masking filters expand this range to (0,015 – 0,4) f_S.

The filter structure synthesis is performed using the method of the synchronous dataflow graph scheduling, which provides the formal design of pipelined structures with high throughput and minimized hardware volume. The method consists in placing the signal flow graph in the multidimensional index space, and mapping it in the subspaces of structures, and time events. Limitations on the mapping process help to minimize both the clock period and hardware costs of the filter structure, which is configured in FPGA [6,7].

The resulting filter structure has three multiplication units for the coefficients a, b, c, which are shared in time between four filter stages. The structure operates with a period of 8 clock cycles, calculating 24 multiplications per period. Due to the fully pipelined filter structure, the clock frequency f_{CLK} achieves high values.

A special calculation unit is added to the filter structure, which calculates the coefficients a, b, c, depending on the cutoff frequency f_R due to the formulas (3). The frequency f_R is inputted as 12-bit width word, two bits of it control the multiplexer in Fig. 2. The new cutoff frequency is set immediately after the frequency code is registered in the filter, and begins to act after the respective signal transition process. The filter is stable for all possible combinations of the frequency codes.

Fig. 4 shows the frequency response of the filter, depending on the cutoff frequency that demonstrates its high quality. The frequency response is obtained using the filter testbench procedure, presented in [8]. The filter has a high suppression level and a narrow transition band at different settings. The passband ripple is not succeeded 0,23dB.

The characteristics of the filter for the 16-bit data, implemented in Xilinx FPGAs, are presented in Table. 1. Note that each implementation contains only three DSP48 multiplication blocks.

The module of the low-pass filter, which is described in VHDL, is freely accessible on the website of open IP cores [9].

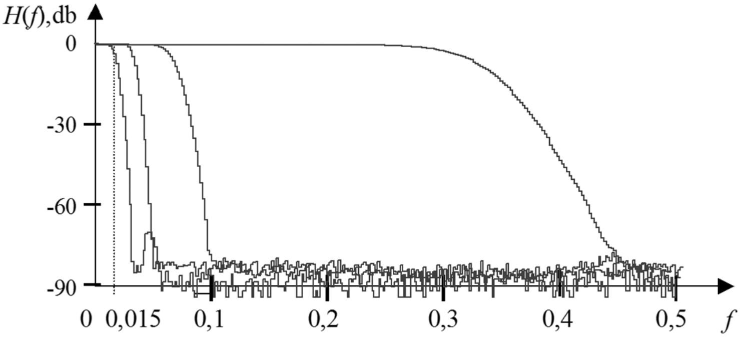

Fig. 4. Characteristics of LPF for different k values

Table 1. Characteristics of the filter, implemented in Xilinx FPGAs

FPGA	LUTs	CLBs	Maximum f_{CLK}, MHz	Maximum f_S, MHz
Spartan-3e	842	700	134	16
Spartan-6	692	268	199	25
Kintex-7	708	298	316	39

7. Conclusions

It is proven that the use of phase IIR filters with multiple delays, and frequency masking effect, it is possible to design the high throughput dynamically tunable IIR filter. The filter structure is performed using the method of mapping the spatial synchronous data flow graph. The proposed filter is effectively implemented in FPGA, where it has small hardware volume and high clock frequency due to the pipelined structure. Such filters can be implemented in a program as well.

References

1. Oppenheim, A.V., Schafer, R.W., *Discrete-Time Signal Processing,* 3-d Ed., Prentice-Hall, New Jersey, 2009.
2. Lim Y.C., Lian Y., *Frequency-response masking approach for digital filter design: Complexity reduction via masking filter factorisation*, IEEE Trans. Circuits Syst. II: Analog and Digital Signal Processing, Vol. 41, N4, P. 518–525, 1994.
3. Lu W.-S., Hinamoto T., *Optimal Design of Frequency-Response-Masking Filters Using Semidefinite Programming*, IEEE Trans. Circuits Syst. I: Fundamental theory and applications, Vol. 41, N4, Aug, P. 557–568, 2003.
4. Crochiere, R.E., Rabiner, L.R., *Multirate digital signal processing*, Englewood Cliffs, NJ, Prentice-Hall, 1983.
5. Regalia P.A., Mitra S.K., Vaidyanathan P.P., *The Digital All-Pass Filter: A Versatile Signal Processing Building Block,* Proc. IEEE, Vol.76, №1, P.19–37, 1988.
6. Sergiyenko A.M., Simonenko V.P. *Otobrajenie perioditsheskich algorythmov v programmiruemye logitsheskie integralnye schemy*, Electronic Modeling, Vol. 29, №2, P.49–61, 2007. (In Russian)
7. Maslennikowa N., Sergiyenko A., *Scheduling of synchronous dataflow graphs for datapath synthesis*, Zeszyty Naukowe Wydziału Elektroniki i Informatyki, Nr 8, Koszalin, P. 53–60. 2015.
8. Sergiyenko A., *Testbench for the filter testing.* Available at http://kanyevsky.kpi.ua/useful_core/testbench_for_filter.html
9. Sergiyenko, A., Uzenkov, O., *Low-Pass IIR Filter*, 2010. Available at http://opencores.org/ project,lp_iir_filter

Summary

Features of the dynamically tuned IIR filters, which are configured in FPGA, are considered. The filters utilize the frequency masking properties of the all-pass digital filters, which have the delay factors z^{-k}. The mapping of the filter algorithm is implemented using pipelining and retiming techniques, based on the spatial synchronous dataflow graph, which provides the small hardware volume, and high clock frequency. The smooth stopband frequency tuning is provided by the built-in coefficient calculator.

Streszczenie

Dany artykuł poświęcony jest właściwościom filtrów cyfrowych, zrealizowanych w programowalnych logicznych układach scalonych. Dzięki wykorzystaniu filtrów fazowych, efektów maskowania, potokowości oraz rozproszeniu zasobów otrzymano małe nakłady aparaturowe oraz wysoką częstotliwość taktowania filtrów. Strukturę filtra otrzymano metodą odwzorowania grafu przestrzennego synchronicznych potoków danych algorytmu filtracji. Płynna zmiana częstotliwości przekroju jest osiągana poprzez szybkie obliczenie współczynników filtra we wbudowanym kalkulatorze współczynników.

Słowa kluczowe: filtry IIR, FPGA, filtry fazowe, synteza struktury.