PLENARY SECTION

UDC 004.383

Anatoliy Sergiyenko,
Anastasia Serhienko,
Michael Ukpu.

MULTIPLIERLESS IIR FILTER DESIGN FOR FPGA

The paper deals with a method of IIR filter design, in which the coefficients in the canonical binary number system representation are searched using the simulated annealing algorithm. The IIR filters are designed on the base of the all-pass filter stages, masking filters and multiplied delays in them. The filter coefficients are selected which have no more than three summands in their representation. Therefore, its pipelined implementation in FPGA has the highest clock frequency and minimum hardware volume. The use of the VHDL language in all the steps of the filter design helps to speed-up and improve the filter optimization.

Keywords: VHDL, FPGA, IIR filter, allpass filter.

Fig.: 3. Tabl.: 1. Bibl.: 12.

Introduction.

The infinite impulse response (IIR) filters are widely used in many real-time applications due to their effectiveness [1]. Many IIR filters are built in the IoT applications for their small energy consumption. Most of the CAD tools for the field programmable gate array (FPGA) provide the customers by the filter IP cores. But the IIR filter IP cores are found among them very rarely. So, the new effective IIR filter IP cores are of demand.

A program like Matlab is usually used for the IIR filter coefficient searching. But the found coefficients need for the truncation. And the resulting filter with the truncated coefficients must be tested for its frequency response agreement.

In the paper [2], the VHDL language is proposed both for the filter structure description, and for the coefficient searching, as well as for the frequency response checking. In this work, an approach is proposed for the multiplierless IIR filter synthesis which is based on the allpass filter scheme and on the filter coefficient optimization using the simulated annealing in the VHDL-simulator.

Allpass-Based IIR filters. IIR filters have limited use in the FPGA systems because of the limited throughput and high sensitivity to the coefficient rounding which provokes the filter instability. But the allpass IIR filters have high linearity of the phase characteristic in the passband, minimized group delay, and multiplication number, immunity to small variations of the coefficients [3]. The second-order allpass section has the transfer function

$$H(z) = \frac{b + cz^{-1} + z^{-2}}{1 + cz^{-1} + bz^{-2}}. \quad (1)$$

The IIR filter with the arbitrary characteristics can have the transfer function

$$H(z) = (A_1(z) + A_2(z))/2, \quad (2)$$

where $A_1(z)$, $A_2(z)$ are the transfer functions of the allpass filters. For example, if $A_1(z)$ is the second-order function (1) and $A_2(z) = z^{-1}$, then $H(z)$ is the transfer function of the low pass filter.

The IIR filter algorithm is usually represented by a synchronous dataflow graph (SDF) [4]. An example of optimized SDF which implements the transfer function (2) when $A_1(z)$ is the second-order function (1), and $A_2(z) = z^{-1}$ is illustrated by Fig. 1.

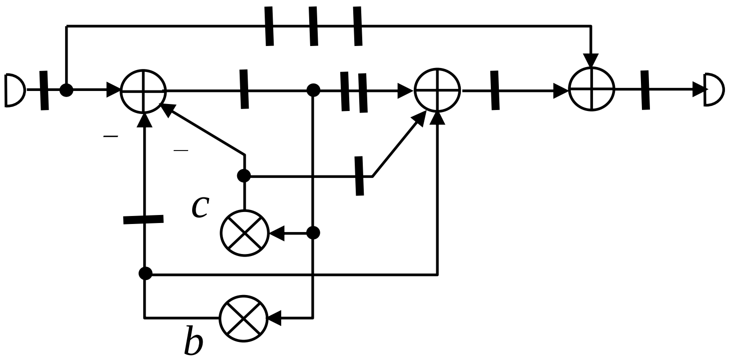

Fig.1. SDF of the low pass IIR filter

Here, the bars represent the register delays, circles do the adders, and coefficient multipliers.

SDF is mapped to the respective pipelined filter structure by the one-to-one mapping. So, the structure derived from the SDF in Fig. 1 has only two multiply units, five adders, and eleven registers. The critical path goes through only a multiplier and an adder. This structure is fully pipelined and its clock frequency is maximized. As a result, the most performance of this filter is achieved when the multiplication to the coefficient c has the minimized delay. This delay achieves the minimum value when this multiplication is performed in the application-specific multiplier based on the adder tree. Such a network is called the multiplierless filter.

More complex IIR filters can be synthesized as a set of stages, each of them is performed as SDF in Fig. 1. Such a filter is designed using the masking filter method, and the method of multiplied delays, which are described in [5]. Fig. 2 illustrates the amplitude-frequency response of the three-staged filter. Then, the resulting response is equal to the multiplication of the stage gains. Thanks to masking, the resulting filter, consisting of simple filter stages, can have a high-quality frequency response.

Each term z^{-k} in the transfer function $H(z)$ (1) corresponds to a delay of k cycles. Consider some prototype filter $H_0(z)$. Then, if the factor k is increased in u

times, then we get a filter with the frequency response $H_n(z) = H_0(z^u)$. The frequency response $H_n(z)$ has the same shape as the prototype filter has but in the range 0 — f_S it is repeated u times, where f_S is the sampling frequency [4]. For example, in Fig. 2, $H_1 = H_0(z)$, $H_2 = H_0(z^2)$, $H_3 = H_0(z^4)$.

The synthesis of the IIR filters using the masking filter and multiplied delay method consists in the selection of the number of stages, factor u, the coefficients for each stage. This is a complex optimization problem. But when the stages are based on the allpass filters then this problem is simplified dramatically [5].

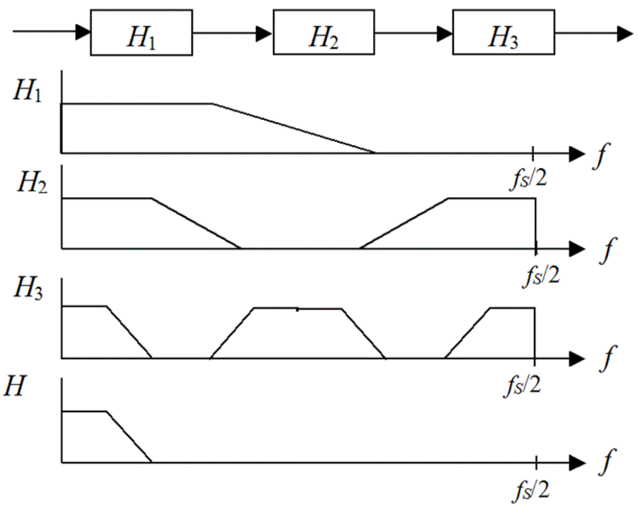

Fig.2. Amplitude-frequency response of the filter with the masking stages

Multiplierless IIR filters. One of the effective methods to speed-up and simplify the IIR filter in FPGA is substituting the hardware multipliers to a set of adders of shifted multiplicands [6]. The modern FPGAs contain the 6-input LUTs, which provide a one-stage network of the three-input adder [7]. Hence, it is preferable to represent the coefficients as the fixed point numbers in the canonical binary number (CBN) system which has up to three terms:

$$\mathrm{c} = \mathrm{d}2^{-\mathrm{p}} + \mathrm{e}2^{-\mathrm{q}} + \mathrm{g}2^{-\mathrm{r}}, \tag{3}$$

where $p < q < r$ are integers, $d, e, g \in \{0, 1, -1\}$.

The optimization problem consists in finding the composite solution vector $\boldsymbol{S} = (\boldsymbol{s}_1,\ldots\boldsymbol{s}_i,\ldots,\boldsymbol{s}_n)$, which belongs to the multidimensional space of solutions, where $\boldsymbol{s}_i = (d_i, e_i, g_i, p_i, q_i, r_i)$, n is the coefficient number. The optimization consists in finding some global extremum of the quality function $\Phi(\boldsymbol{S})$. This function calculates the ripple levels in the passband and stopband of the transfer function $H(z)$. It has to take into account the number of zero parameters d_i, e_i, g_i as well, because it must minimize the filter hardware complexity.

The multiplierless IIR filters, based on the allpass filter like (1), have both the minimized number of the coefficients, and the minimized its bit width $r_i \leq 10$. So, the vector ***S*** has the minimized length too, which substantially simplifies the searching for the optimum CBN coefficients. For example, the three-staged filter like one illustrated by Fig. 2 has no more than $n = 6$ coefficients. The multiplication to each of them is implemented, at least, in a single 3-input adder.

When the coefficient number n is higher than 3 – 4, then some unexact optimization methods have to be used. The nature-inspired evaluation optimization method showed good results in this situation [8]. Usually, they simulate walking in the space of parameters ***S*** in order to find the extremum of $\Phi(\boldsymbol{S})$.

The simulated annealing method can be seen as the extension of such evaluation optimization methods like hill climbing and gradient descent methods [9]. It simulates the diffusion process of a molecule which coordinates are represented by a vector ***S***. By the optimization process, the physical body model with that molecule is firstly heated to the annealing temperature and then is cooled over simulation time. This method provides effective solutions and is used in this work.

The following analogies are accepted [10]: the state of the body is the solution vector ***S***, the quality function $\Phi(\boldsymbol{S})$ means its energy; the change of the body state is the exchange of ***S*** to the next solution $\boldsymbol{S}_n$; temperature is a separate parameter that is decreased during the optimization process, the final temperature $t_{\min}$ is the stopping point of the optimization.

The generalized algorithm for the simulated annealing is as follows:

```
The initial solution $S_n$ is selected,
the temperature $t$
and the cooling coefficient $a$
are assigned;
repeat{
    a random solution $S$ is selected,
    which is near to $S_n$;
    $\Delta\Phi=\Phi(S) - \Phi(Sn)$;//energy decrement
    if $\Delta\Phi < 0$ then
         $S_n = S$;
    else {
         random number $p$ is generated
         if $p < \exp(\Delta\Phi/t)$  then
              $S_n = S$;
```

This algorithm is implemented in a VHDL program using the methods described in [2]. To simplify the searching process, the vector $\boldsymbol{s}_i$ is coded by its origin image c_i calculated by the formula (3). All possible values c_i are sampled into the ROM, i.e., $c_i = f(a_i)$, and the solution is coded as $\boldsymbol{S}' = (a_1,\ldots,a_n)$.

The initial solution $\boldsymbol{S}'_n$ is derived for a set of parameters a_i which represent the coefficients near some exact solution. The random solution $\boldsymbol{S}'$ is derived from the previous one by the addition of some random vector $(\delta_1,\ldots,\delta_n)$. Its elements are exchanged in some range which depends on the temperature t. The function $\Phi(\boldsymbol{S}')$ is calculated as a sum of ripples of the function $|H(\boldsymbol{S}',z)|$ plus the parameter which is proportional to the number of ones in the CBN representation of the coefficients.

Experiments. The method was used to build a set of IIR filters of the order $n = 5,\ldots,9$. These filters were put in the database of the IIR Filter Generator [11]. This Web application generates the synthesizable VHDL models of the filters with the given bit width, stopband frequency. The parameters of some synthesized half band (HB) and low pass (LP) filters are shown in Table 1. Here, the hardware volume is given in the configurable logic block slices (CLBs).

Another example is the synthesis of an LP filter with a cut frequency of 0.025fS, where fS is the sampling frequency. The filter structure corresponds to (2), where A1(z), and A2(z) are transfer functions of the 3-d and 4-th order, respectively. The measured resulting filter transfer function is shown in Fig. 3. The coefficients found by the VHDL program are equal to

$$c_0 = -1.00\bar{1}01; \qquad c_1 = -\,10.0000\bar{1}; \qquad c_2 = -\,10.00\bar{1}; \qquad c_3 = -\,10.0000\bar{1}\,00\bar{1}.$$

$$b_1 = 1.00\bar{1}01; \qquad b_2 = 1.00\bar{1}0\bar{1}; \qquad b_3 = 1.0000\bar{1}01;$$

The results of the synthesis of other HB and LP multiplierless filters which have the equivalent amplitude-frequency characteristics are given in Table 1 too. Comparing to them, the proposed filters have less hardware volume and a much higher clock frequency by the approximately equal suppression level.

Table 1

Parameters of filters configured in Xilinx Kintex FPGA

Filter	*Hardware, CLBs*	*Max. clock frequency, MHz*	*Suppression, db*	*Reference*
HB	203	690	120	
HB	441	107	106	[12]
LP	179	310	54	
LP	203	189	57	[8]

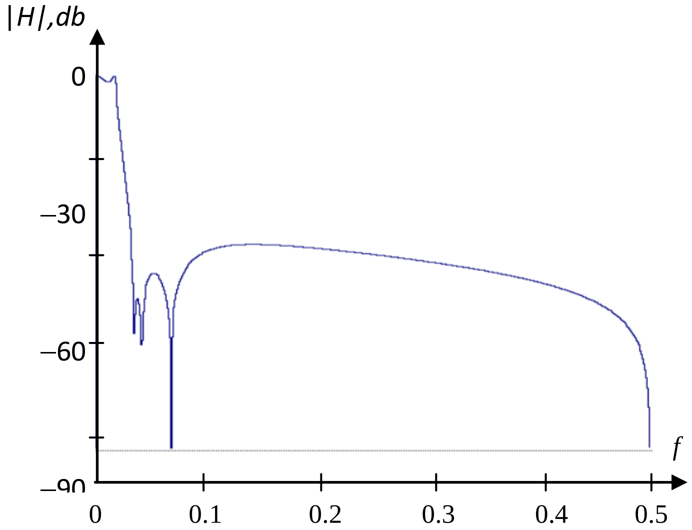

Fig. 3. Frequency responses of synthesized LP filter

Conclusion. The design of the IIR filters using the allpass stages and selecting the filter coefficients in the canonical binary number system representation, which contains no more than three not zero bits is proposed. The IIR filters are designed using the methods of masking filters and multiplied delays. A method of IIR filter coefficient searching is proposed which is based on the simulated annealing optimization algorithm and provides deriving a set of optimized filter coefficients in the canonical binary number system representation.

References

1. Schlichthärle D. Digital Filters. Basics and Design. 2-nd Ed-s. Frankfurt, Main, Germany: Springer, 2010.

2. Sergiyenko A., Serhienko A. VHDL Generation of Optimized IIR Filters. IEEE 2-nd Ukraine Conference on Electrical and Computer Engineering (UKRCON), 2019. P. 1171-1174.

3. Vaidyanathan P. P., Regalia P., Mitra S.K. The Digital All-Pass Filter: A Versatile Signal Processing Building Block. Proc. IEEE. V. 76. 1988. № 1. P. 19–37.

4. Meyer-Baese U. Digital Signal Processing with Field Programmable Gate Arrays. 4-th Ed. Berlin Heidelberg, Germany, Springer, 2014.

5. Krukowski A., Kale I. DSP System Design. Complexity Reduced IIR Filter Implementation for Practical Applications. Germany, Springer. 2004.

6. Milic L. D., Lutovac M. D. Design of multiplierless elliptic IIR filters with a small quantization error. IEEE Trans. on signal processing. V. 47. 1999. № 2. P. 469–479.

7. Guggilla N. K., Dudha C. S. Synthesis. Designing with Xilinx FPGAs Using Vivado, Churiwala, Ed. Springer. 2017. P. 97–110.

8 Anzova V. I., Yli-Kaakinen J., Saramaeki T. An Algorithm for the Design of Multiplierless IIR Filters as a Parallel Connection of Two All-Pass Filters. IEEE Asia Pacific Conf. on Circuits and Systems, APCCAS. 2006. pp. 744-747.

9. Kruse R., Borgelt C., Braune C., Mostaghim S., Steinbrecher M. Computational Intelligence. A Methodological Introduction. 2-nd Ed. London. Springer. 2016.

10 Kirkpatrick S., Gellatt C. D., Vecchi Jr. M. P. Optimization by Simulated Annealing. Science. V. 220. 1983. No5. P.671–680.

11. Sergiyenko A. VHDL design of multiplier-free IIR filters. Kyiv: NTUU “KPI”, 2016. [Online]. URL: http://kanyevsky.kpi.ua/GEN_MODUL/index_eng.php

12. Yeung K. S., Chan S. C. The Design and Multiplier-Less Realization of Software Radio Receivers With Reduced System Delay. IEEE Trans. On Circuits and Systems – Regular Papers, V. 51. 2004. P. 2444–2449.

AUTHORS

Anatoliy Sergiyenko (supervisor) – professor, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: aser@comsys.kpi.ua

Anastasia Serhienko – post-graduate student, Application Specific System Department, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: anastasia.serhienko@gmail.com

Michael Ukpu – student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: askoflight@yahoo.com