Synthesis of the integer FIR filters with short coefficient word length

First Author (Osamah Waleed allawi)
Ministry of Education Directorate General of Education in Baghdad Al-kalam for men Secondary School , Baghdad, Basmaya,Iraq (osamah.waleed@Gmail.com)

Second Author (Mohammed Hassan Abd)
University Baghdad .College Of Education Ibn al-Haytham, Department of Computer Science, Iraq(mohammed.h@uobaghdad.edu.iq)
orcid.org/0000-0002-9294-6383

Abstract— The integer simulation and development finite impulse response (FIR) filters taking into account the possibilities of their realization on digital integer platforms are considered. The problem statement and solution of multifunctional synthesis of digital FIR filters such a problem on the basis of the numerical methods of integer nonlinear mathematical programming are given. As an several examples, the problem solution of synthesis FIR-filters with short coefficient word length has been given. The analysis of their characteristics is resulted. The paper discusses issues of modeling and synthesis of digital FIR filters with provision for the possibilities of their implementation on digital platforms with integer computation arithmetic. The formulation of the problem of multifunctional synthesis of cascade FIR filters using the methods of integer nonlinear mathematical programming is given. The efficiency of this approach is illustrated by examples of solving the problems of synthesizing integer FIR filters with a minimum coefficient word length. The analysis of the synthesized filter characteristics is made.

Keywords— the integer FIR-filter, integer nonlinear programming, multifunctional synthesis, criterion function .

1 INTRODUCTION

A digital FIR filter in its canonical presentation is, as is known, a discrete system, for which the ratio between the input and output time sequences xn and yn is determined by a linear discrete convolution equation.

$$y_n = \sum_{k=0}^{N} h_k \cdot x_{n-k} \quad , \qquad (1)$$

where the constant coefficients hk are counts of the filter impulse response. The input filter window includes N+1 samples, and the value of N determines the order of the FIR filter. Its transfer function is

$H(z) = \sum_{k=0}^{N} h_k \cdot z^{-k}$, will hold, where the complex variable z when turning to the complex representation of the frequency response $K(e^{j\omega}) = \left|K(e^{j\omega})\right| \cdot e^{j\varphi(\omega)}$ take the value $z = e^{j\omega}$, $\omega = 2\pi f / fs$ is the normalized angular frequency, and fs is the signal sampling rate. The main advantage of a FIR filter is the capability of implementing a linear phase frequency response (PFR) under the condition of symmetry (antisymmetry) of the impulse response, and also that FIR filters implemented non-recursively, that is, using direct convolution (1), are always stable.

The total quality of a FIR filter is determined both by its selectivity and running speed, that is, the minimum response time for implementation on a given digital platform. Selectivity is usually understood as the ability to satisfy a set of required filter characteristics in the frequency domain. FIR filters, as is well known, have a relatively low selectivity when impulse response with a large number of samples is required to realize the frequency response of a complex shape with sharp cut-offs, which requires a large amount of digital calculations to calculate the filter response. In this case, the main factors determining the computational costs, as well as running speed, are also the calculation arithmetic and the word length of the coefficients (bitness) of the digital FIR filter [46]. A review of publications on classical FIR filter design methods shows that they are currently dominated by analytical calculation using window weighting methods, frequency sampling methods, methods for calculating optimal (Chebyshev) filter [1], and some other approaches to the synthesis of FIR filters with finite word length coefficients [2]when real arithmetic of calculations is used in digital filtering algorithms. The real format of data representation requires quantization of their values, which, at low small word length of data presentation, leads to a significant distortion of frequency responses and the appearance of quantization noise. In addition, a real solution can be implemented only on a specialized signal processors. The quantization issues in analytic calculation of the FIR filter have been well studied, but in general it can be noted that with a high filter order (N> 100), significant difficulties in meeting the specified specifications arise already when quantizing real data up to 7–8 bits. However, the possibility of direct synthesis of digital integer FIR filters directly in the integer state space can be provided by the integer nonlinear programming (INP) technique, a general description of which is given in [3]. In this case, the state space is understood primarily as the multidimensional space of integer parameters (coefficients) of FIR filters, input and output signals, i.e. integer time sequences, as well as basic integer operations on data in the digital filtering algorithm. Naturally, integer operations on any digital platform are performed much faster than real calculation operations. The number of cycles of the central processor unit required for the implementation of basic integer operations is substantially less. For example, for the C8051F120 microcontroller processor, which can operate with both integer and real data representation formats, basic addition operations are seven times faster, and multiplication operations are more than four times faster for integer arithmetic in comparison with real calculations.

The INP method allows you to design effectively integer filters with a given data representation word length and maximum fulfillment of the requirements for the set of frequency characteristics of the filter with an arbitrary form of their specification.

Currently, 8-bit digital platforms with integer calculation arithmetic is an integral part of available commercial digital platforms. When implementing high-speed integer digital filters (IDF) on specialized platforms or on a chip, the word length of data presentation can be even lower (down to 4 or even 3 bits). And as rightly shown in [4], a reduction in data width even by 1 bit can save up to 50% of RAM, with a significant reduction in the calculation time of the filter response. Therefore, for high-speed chip based filters, the word length of the data representation is largely a determining factor.
The issues of design of recursive IDF with minimum word length by the integer nonlinear programming (INP) technique was discussed in detail in [5], and ensuring the phase linearity of integer FIR filters in [6]. This paper discusses the possibility of solving the issues of synthesis and cascade integer FIR filters with the minimum word length of data representation by the by the INP software (release 3.1), i.e. the examples illustrating the fundamental capabilities of this approach to multifunctional design of digital systems.

1.1 Modeling and synthesis of low word length integer FIR filters

The direct integer FIR filter is defined as follows [11]:

$$y_n = \sum_{k=0}^{N} \frac{b_k}{a_0} \cdot x_{n-k}$$

The principal feature of IDF is that its coefficients bk and ak belong to an alternating series of integers, which can be either natural or binomial (for the normalizing factor a0). The variation change of the coefficients is determined by the word length of the data in the digital platform used. At the same time, the complement is most widely used in computer systems for representing integers. In practice, as for IIR filters, a cascade (series) structure for constructing an FIR filter is often used, having certain advantages over the transversal filter. In this case, the structure of an integer FIR filter is given by the number of its sections in the cascade version, and the vector IX determines the variable parameters (coefficients) of the filter sections in the multidimensional integer parameter space In. The transfer function for a integer FIR filter, consisting of a cascade connection of m second order sections (m = N/2, N is the general order of the filter), has the following form:

$$H(z) = \prod_{i=1}^{m} \frac{b_{0i} + b_{1i} z^{-1} + b_{2i} z^{-2}}{a_{0i}} , \qquad (2)$$

From (2) it is easy to obtain the difference equation for one integer section:

$$y_n = \left(b_0 x_n + b_1 x_{n-1} + b_2 x_{n-2}\right) / a_0 , \qquad (3)$$

where xn, yn are input and output integer time sequence, a0 is a normalization factor.

As can be seen from (1.3), when calculating the filter response, the division by the integer coefficient a0 should be performed, which can be implemented by the bit-shift operation provided that each i-th normalizing factor belongs to the integer binomial series (series of 2n):

$$a_{0i} \in \left\{ 2^q \right\}, \quad q = \overline{0,\, W_k - 1} \quad i = \overline{1, m}, \tag{4}$$

where Wk is the bit word length of integer coefficients, sign including.

Figure 1 shows the structure of an integer section corresponding to the equation (3). As is seen, besides conventional addition, multiplication and clock period delay operations, there is a shift by B=log2a0 bits, used to implement integer division by the normalizing factor a0. Thus, when calculating the FIR-filter response yn, the minimum number of basic operations is used, all of which are integer, which determines the essential higher performance of the filter for real-time operation.

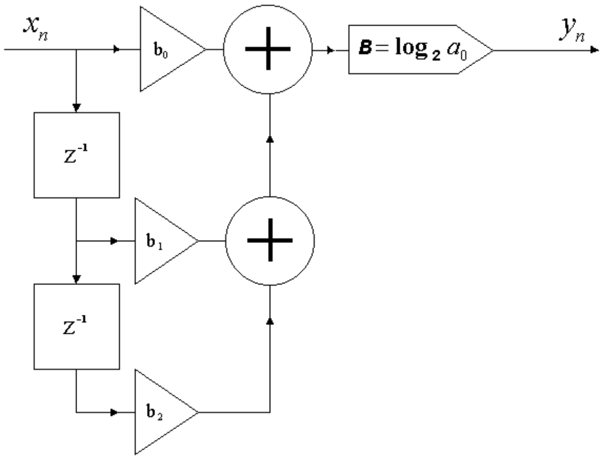


Fig. 1. The Structure FIR section

An important advantage of integer filters is the absence of a data quantization procedure (both for filter coefficients and intermediate integer calculations) in the course of calculating the filter response in real time, and, therefore, the absence of negative data quantization effects listed above. That is, quantization is replaced by integer discretization of the multidimensional space of coefficients before synthesizing a filter with obtaining an integer solution (vector of integer coefficients IX0) with zero implementation error on a digital platform or chip with a given length Wk of the word of integer coefficients.

As for the results of the intermediate calculations, all of them are also integer. And the result of multiplying integers (for example, the current digital sample and the FIR filter coefficient) will be completely in full determinate, not requiring quantization for implementation on the digital platform with a given word length Wk at data representation. For a given bit ness of quantization of the input signal Wx, it is sufficient to select an internal storage register with a bitt ness of Wak= Wx+ Wk bits to store the result of integer accumulative multiplication (MAC), performed according to the algorithm (3). Overflow fluctuations, that is, the occurrence of large limit cycles caused by the word size overflow of the storage register, with such a calculation of its bit ness almost never occurs, especially considering that the accumulation of the result of integer accumulative multiplication in (3) is carried out as algebraic addition, taking into account the sign of the terms, what significantly reduces the bit ness of the result. In the algorithm for calculating the response of an integer FIR section, the only random process (that is, the source of noise) is the normalization operation, dividing by the normalizing binomial factor a0, which, as shown above, is implemented by register right shift by B=log2a0 bits with data loss in the least-significant bits. Note that in the integer filter there is one source of noise for each section, the origin of which is a required shift operation [5]

In digital platforms with real arithmetic of calculations, the representation of the result of multiplying real numbers with a finite number of digits is stochastic in nature and, naturally, requires a quantization procedure, the result of which is stochastic quantization noise.

As is well known, in cascade forms for constructing digital filters, a gain scaling procedure is required, that is, a uniform gain distribution over stages. This allows to operate the filter in a wide dynamic range of input signals. However, for cascade IDFs it is much easier to calculate such gain restriction of an integer section not by using the Lp-norm, but by directly requiring a small spread of the transfer ratio of individual sections [7]As experience shows, no significant dynamic range reduction of a FIR filter will occur if the maximum transfer coefficients of the cascades differ by no more than 5–7 times. With a more coarse amplification scaling, the filter dynamic range reduction becomes noticeable. The requirements of signal gain scaling are formulated by bilateral functional constraints [[5]] for the synthesis extreme problem of the INP synthesis.

In general, the problem of integer nonlinear programming for computer synthesis of a non-recursive IDF with a given word length of data representation can be written as:

$$F^{o}(\boldsymbol{IX^{o}}) = \mathbf{min}\, F(\boldsymbol{IX}) \qquad \boldsymbol{IX} \in I^{4m} \tag{5}$$

$$-\ 2^{Wk-1} < b_{di} < 2^{Wk-1} \quad d = \overline{0,2} \quad i = \overline{1,m}\ , \tag{6}$$

$$a_{0i} \in \left\{2^{q}\right\},\ \ q = \overline{0, W_k - 1} \quad i = \overline{1,m}\ , \tag{7}$$

$$K_i^{\min} \le |K_i(e^{j\omega})| \le K_i^{\max} \quad i=\overline{1,\, m} \quad , \qquad (8)$$

where m is the number of sections of the second order, d is the index of the coefficient of the section transfer function (1.1), IX is the vector of the multidimensional integer parameter space, F(IX) is the objective function, Kimin , Kimax are the acceptable limits of the gain change of the i-th section.

The synthesis extreme problem (5) is written for the integer space I4m of parameters (filter coefficients) of dimension 4m. Constraints (6) set the limits for the variation of integer coefficients, and relation (7) determines the belonging of the a0i coefficients to the binomial series. The constraints (8) scale the transfer coefficients of the sections at a given interval and are implemented in the minimization algorithm using the penalty function.

The multi criteria objective function is most often formed as weighted sum (10) of single objective functions fi(IX), which determine fulfilling functional requirements for a particular kind of frequency response of the filter

$$F(\boldsymbol{IX})=\sum_i \beta_i \cdot f_i(\boldsymbol{IX})\,. \qquad (9)$$

Factor βi sets the significance (weight) of the characteristic (i-th frequency window). The single objective functions fi(IX) are formed in the graphics editor of the synthesis package, usually by the criterion of the mean-square error

, (10)

where Yn(IX) is the current value of the filter frequency response on the n-th discrete frequency, and Y^T_{n} is the required value of the frequency response.

The maximum error criterion

$$f_{\text{i}}(\boldsymbol{IX}) = \max_n \; |Y_n(\boldsymbol{IX}) - Y_n^{\text{T}}| \qquad (11)$$

is less effective and used much more rarely.

The iterative search solution to the external INP problem (5) in a given space of integer is performed using the software-based algorithmic complex [8]with the search for a global extreme on a discrete grid of the Gray code. This complex is adapted to finding solutions in the discrete integer mode of a multidimensional region of parameters (filter coefficients). The vector IXO, minimizing the scalar objective function F(IX) on the set of admissible integer solutions (6), is an effective, solution of the IDF multifunction synthesis problem.

It should be noted that, in contrast to the classical analytical calculation, a search design is, of course, an intellectual process, intellectual design. Many scenarios for solving a complex project problem can be proposed, many specific techniques and skills can be applied by an experienced search designer to successfully solve a complex design problem.

A typical scenario for the search design of cascade IDFs is a dynamic programming script, as a sequence of search tasks with a phased increase in the order of the designed filter. At the first, start-up stage, a low-order structure is used (max. of 6th or 8th order). Naturally, the level of fulfillment of combined requirements by such a filter will be low. At the second stage, already this solution is used as a source. At the same time, the order of the filter is increased by duplication of the section previously found coefficients of one of the section (which can be done automatically within the synthesis package). After several such iterations, the final order of the designed filter is determined, in which the error of complex aggregate requirements is within the specified tolerance.

The possibility of synthesizing integer recursive FIR filters with the minimum word length data representation using the INP method is shown below. The multifunctional synthesis in this case is carried out according to both the criterion of the required frequency selectivity and the criterion of the possible phase linearity in the filter pass band. Thus, the objective function in the design specifications below is formed as a weighted sum of two single objective functions fAFR(IX) and fAFR(IX), which respectively signify the requirements of the filter frequency selectivity and its phase linearity

$$F(\boldsymbol{IX}) = \beta_1 f_{АЧХ}(\boldsymbol{IX}) + \beta_2 f_{ФЧХ}(\boldsymbol{IX}), \quad (12)$$

where the singular objective functions are formed with the additive quadratic criterion (10). The weight coefficients of the objective functions were determined practically during the step solution of external problems of integer programming.

FOUR BIT LOW PASS FIR FILTER

The initial specification for the synthesis of the low-pass filter with the required phase linearity in the integer space of 4-bit coefficients was as follows:

1. Bandwidth of the filter 0 - 0.225 (hereinafter on the scale of relative frequencies f/fs, where fs is the sampling frequency of the signal)
2. The transmission ratio in the pass band 0dB to a tolerance of ± 1.0dB
3. Phase nonlinearity in the band not above 10º
4. Transition band 0.225 - 0.32
5. The suppression level at frequencies below 0.32 not less than 40dB
6. Bit ness (coefficient word length Wk) 4 bits, sign including
7. The FIR filter order is 56
9. Transfer ratios of the sections are in the interval {1.0 – 2.0}

Figure 2a shows the required amplitude-frequency response of the filter (dashed line). When synthesizing a low-pass filter, taking into account the phase linearity, the objective function was formed by (12), and the particular functions were formed using the additive criterion (10).

The problem of integer programming for multifunction synthesizing with 28 cascade second-order sections is formulated as follows:

$$F^o(\boldsymbol{IX^o}) = \min F(\boldsymbol{IX}) \qquad \boldsymbol{IX} \in I^{84} \quad (13)$$

$$-7 \le b_{di} \le 7 \quad i = \overline{1,28}, \quad (14)$$

$$a_{0i} = 8 \quad i = \overline{1,28} \quad , \tag{15}$$

$$1.0 \leq |K_i(e^{j\omega})| \leq 2.0 \quad i = \overline{1,\ 28} \tag{16}$$

Thus, the final minimization of the objective functional was carried out on the 84-dimensional integer space of 4-bit parameters in the allowable domain (14) while scaling (16) the gains of the sections in a given interval. The normalizing factors (15) of all the filter sections were equal. To input the structure of the low-pass filter, the built-in topological editor of the INP-package (10) was use, which allows to create a source data file for solving a specific synthesis task indicating the number of variable coefficients, their initial values and limits of variation, as well as their possible duplication.

1.2 Tables

Table 1 shows the optimum integer 3-bit coefficients of the low-pass filter transfer function with the same significance (β1=β2) requirements for frequency and phase response, as well as the achieved values of the gains of its sections.

Table 1. shows the optimum 4-bit coefficients of the low-pass filter

Section	b0	b1	b2	Kmax
1	6	-6	1	1.4
.	5	-5	5	1.4
3	7	-2	0	1.1
4	-6	-6	4	1.5
5	-6	3	-2	1.4
6	-7	-7	0	1.7
7	-7	4	-3	1.7
8	6	4	0	1.3
9	2	-6	-6	1.3
10	-6	-1	2	1.0
11	-5	6	5	1.5
12	-6	-6	0	1.5
13	5	4	5	1.7
14	-7	-2	0	1.1
15	-7	-5	0	1.5
16	-7	-6	0	1.6
17	-7	-3	1	1.2
18	-6	-5	3	1.3
19	-6	-6	2	1.3
20	-6	-2	-1	1.1
21	-7	-4	0	1.4

22	6	5	-1	1.3
23	6	4	-1	1.1
24	-6	2	-1	1.1
25	4	-4	2	1.2
26	4	5	-4	1.2
27	-6	2	-1	1.1
28	5	6	-3	1.3

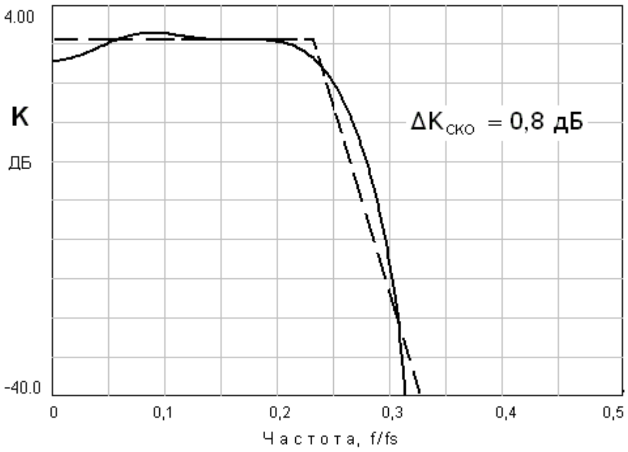
Its frequency responses are shown in Fig. 2 The task time for solving this problem on a standard personal computer under the scenario of dynamic programming and the objective function (12) did not exceed 15 minutes.

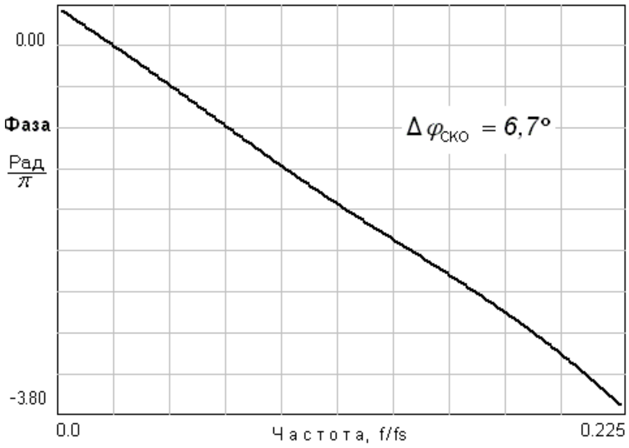
Table 2. shows the optimum 4-bit coefficients of the high-pass filter.

Section	b0	b1	b2	Kmax
1	0	0	-7	0.9
2	7	-7	-2	1.5
3	-1	-2	-7	1.3
4	3	7	-5	1.3
5	7	-3	-1	1.1
6	-2	-7	7	1.5
7	7	-7	-2	1.5
8	7	-7	-2	1.5
9	6	4	1	1.4
10	7	-1	0	1.0
11	7	-7	-2	1.5
12	7	-7	-2	1.5
13	7	-1	0	1.0
14	7	-7	-2	1.5
15	7	-1	0	1.0
16	7	1	1	1.1
17	7	0	0	0.9
18	7	0	0	0.9
19	7	2	1	1.3
20	7	0	0	0.9
21	7	0	0	0.9
22	7	-7	-6	1.9
23	0	-7	2	1.1
24	5	2	2	1.1
25	7	0	0	0.9
26	7	0	0	0.9
27	7	0	0	0.9
28	7	0	0	0.9

Table 2 shows the optimum integer 3-bit coefficients of the high-pass filter transfer function with the same significance (β1=β2) requirements for frequency and phase response, as well as the achieved values of the gains of its sections. Its frequency responses are shown in Figure. 4

2 ILLUSTRATIONS





a b

Fig. 2. Responses of the synthesized LF filter a) frequency response

b) phase response

As for the assessment of phase nonlinearity in the pass band of the synthesized FIR filter, then the application the root-mean-square error (RMS) of phase distortions by the criterion of the influence on the output waveform can be correctly applied,

$$\Delta\varphi_{ско} = \sqrt{\frac{1}{p}\cdot\sum_{n=1}^{p}\left[\varphi_n(\boldsymbol{IX})-\varphi_n^L\right]^2}, \qquad (17)$$

not their estimates by the maximum error criterion

$$\Delta\varphi_{\max} = \max_{n} |\varphi_n(\boldsymbol{IX}) - \varphi_n^L|$$ **, will hold,**

where φn(IX) is the current value of the phase response at the nth discrete frequency of the evaluation range, and is the desired value of the linear phase response.

This follows directly from the Fourier transform, which determines precisely the additive influence of the phase and amplitude spectrum of all harmonics of the processed signal on the distortion of its shape. Therefore, further to assess the phase nonlinearity, as well as the non-uniformity of the frequency response in the pass band of the filter, criterion (17) will be used. In this case the rms non-uniformity of the amplitude- frequency response and the nonlinearity of the phase-frequency response in the pass band of 4-bit HPF were equal, respectively, to ΔKско=0,8 dB and Δφско=6,7º.

The practical implementation of the low-pass filter was carried out on the MSP430F1611 multifunction microcontroller with a RISC-core [20]. The frequency response of the filter was measured using a real signal with an automated panoramic measuring system developed in the Lab VIEW virtual instrument environment. The experimental results of the frequency response measuring of the synthesized LF filter over the entire NY Quist interval for the sampling frequency fs = 1 kHz. are shown in Fig. 3

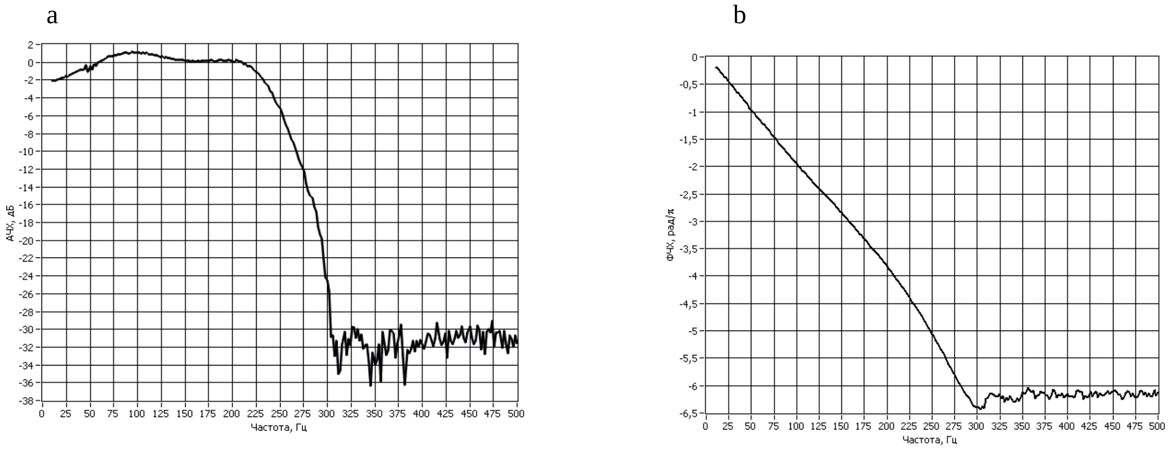


Fig. 3. Experimental measurements of the amplitude-frequency response (a) and phase response (b) of a 3-bit low-pass filter

Thus, all functional requirements for the INP-synthesis of a four-bit low-pass filter were met.

FOUR BIT HIGH PASS FIR FILTER

The specification for the synthesis of the high-pass filter with the required phase linearity in the integer space of 4-bit coefficients was as follows:

1. Filter bandwidth 0.225 to 0.5
2. The transmission ratio in the pass band 0dB to a tolerance of ± 0.8dB
3. Phase nonlinearity in the band not above 10º
4. Transition band 0.08 - 0.225
5. The suppression level at frequencies below 0.08 not less than 40dB
6. The word length of the coefficients Wx is 4 bits
7. The FIR filter order is 56
9. Transfer ratios of the sections are in the interval 0.8 – 2.0

The required frequency response of the filter is shown in Fig. 4a (dotted line). When synthesizing a HPF with regard to phase linearity, the objective function was formed as a weighted sum of partial objective functions fAFR(IX) и fPFR(IX) by (12), particular functions were formed using the additive criterion (10). The weight coefficients of the window objective functions were determined practically during the phased solution of the following external problems of integer programming for the multifunctional HPF synthesis in the form of a cascade connection of 28 second-order sections:

$$F^{o}(\boldsymbol{IX^{o}}) = \min F(\boldsymbol{IX}) \qquad \boldsymbol{IX} \in I^{84} \tag{18}$$

$$-7 \le b_{di} \le 7 \quad i = \overline{1,28} \quad , \tag{19}$$

$$a_{0i} = 8 \quad i = \overline{1,28} \quad , \tag{20}$$

$$0.8 \le K_i(e^{j\omega}) \le 2.0 \quad i = \overline{1,\,28} \quad . \tag{21}$$

Thus, the final minimization of the objective functional was also carried out on the 84-dimensional integer space of 4-bit parameters in the allowable domain (19) while scaling (21) the gains of the sections in a given interval. The normalizing factors (20) of all the filter sections were equal

a b

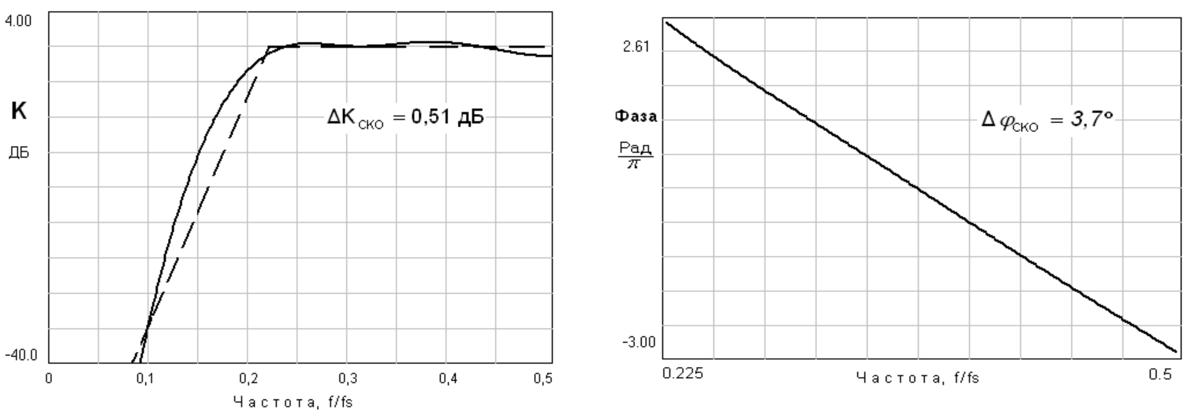


Fig. 4. Responses of the synthesized HF filter a) frequency response, b) phase response

The filter was also practically implemented on a MSP430F1611 microcontroller. The results of the frequency response automated measuring of the filter over the entire NY Quist interval for the sampling frequency fs = 1 kHz. are shown in Figure. 5 Since, the output signal was taken directly from the DAC, there is also a noise level increase near the NY Quist frequency, where the effects of input signal quantization are most pronounced. As can be seen, the experimental characteristics of the low bit ness high-pass FIR filter quite correspond to the synthesis data.

a b

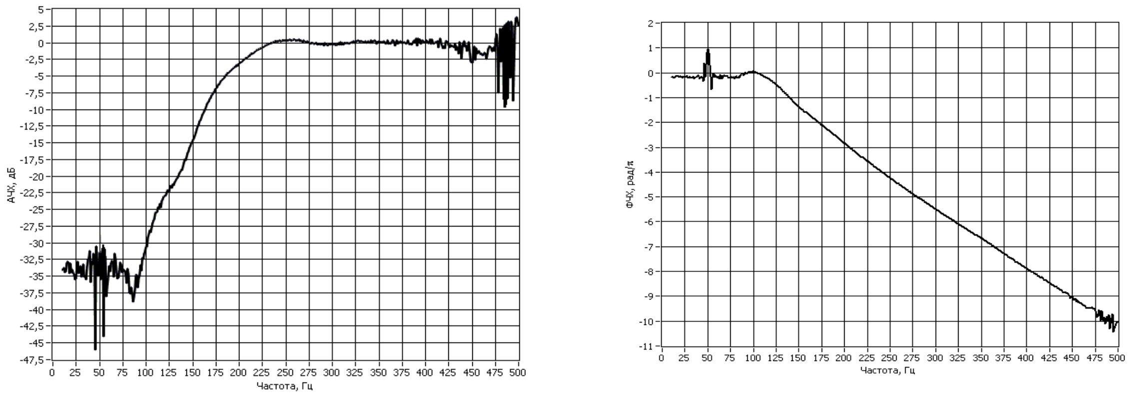


Fig. 5. Experimental measurements of the amplitude-frequency response (a) phase response (b) of a 4-bit high-pass filter

Thus, the functional requirements for the INP synthesis of a 4-digit recursive high-pass filter with the same significance requirements for the amplitude-frequency response and phase-frequency response of the filter were met. It is interesting to compare the MFR of the synthesized 4-bit high-pass filter with free coefficients with the option of synthesizing the same filter with symmetric coefficients, which can easily be implemented by the INP methodology [39]. Its frequency responses are shown in Figure. 6

a b

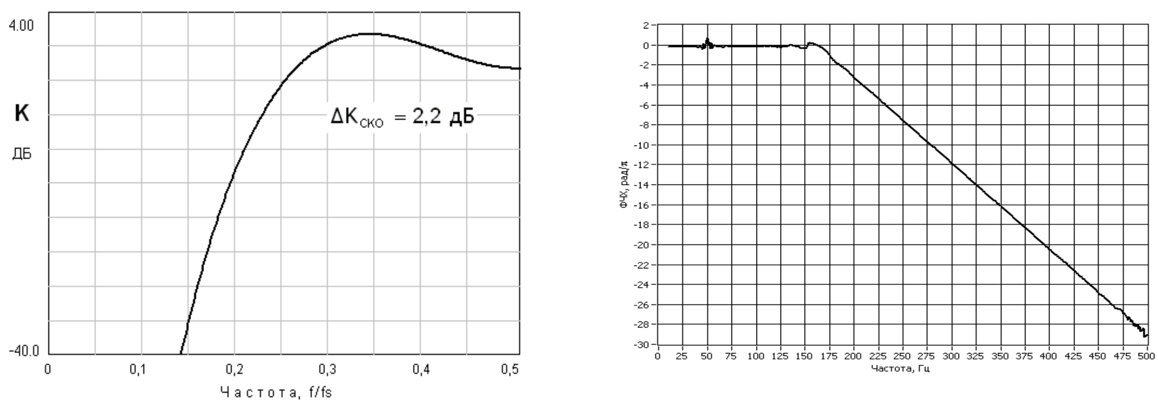


Fig. 6. (a) MFR of the 4-bit high-pass filter for synthesis with symmetric coefficients (b) experimental PFR measurements

As can be seen, with full phase linearity, the root-mean-square band pass non-flatness of the MFR filter in the pass band was ΔKrms= 2.2 dB for the same order of the FIR filter. Thus, the condition of the impulse response symmetry is a very strict restriction to the parameters (filter coefficients), and, consequently, to the selectivity of the designed filter. In this example, the synthesis of the FIR high-pass filter shows that with symmetric coefficients its selectivity acc. criterion (17) is 4 times less compared to the solution with free coefficients. Obviously, this effect will be even stronger for narrow-band filters, where phase distortions are small by themselves and in a narrow bandwidth they are much easier to eliminate by direct INP synthesis with free coefficients, without losing the filter selectivity. Thus, with INP-synthesis of a cascade FIR filter with phase linearity only in a given band, the filter length can be significantly reduced if the functional requirements for the frequency and phase response of the filter are fully met.

3 Acknowledgment

We gratefully recognize all this universities and contributors in this research regarding giving us a part of their time.

4 Conclusion

Methods of integer nonlinear programming as applied to the problems of designing linear digital filters are a modern and alternative to traditional methods of designing digital FIR filters. The fundamental distinction of INP-synthesis is the use of modern numerical methods of machine design, allowing to work not with an analytical, but with a discrete representation of the characteristics of the designed filter, when both the initial required and current characteristics are tabulated with a given discreteness of their representation in the frequency domain and in the computing system represented by two-dimensional arrays (vectors). This makes it possible, on the one hand, to calculate with a given accuracy all the required filter characteristics (including GDT and dispersion) using numerical methods. On the other hand, to apply effective search methods of discrete programming for the synthesis of a technical solution, allowing to design digital FIR filters directly in the integer parameter space. The search criterion for this is that the current functioning of the synthesized filter complies with the required functioning in terms of the set of frequency responses. Modern algorithmic complexes of integer minimization make it possible to solve such design problems reliably and efficiently while fulfilling all external requirements and constraints to the operation of a digital filter, which makes it possible to significantly improve the quality of designed filters and reduce the time to develop them.

From the above content it can be seen that, when compared with traditional approaches, the synthesis of FIR filters using integer nonlinear programming allows:

1.To carry out the filter synthesis according to the frequency characteristics specified in any form and at a given frequency scale (linear, logarithmic etc.);

2.Designing filters with a short data word length (down to 3 bits) directly in the FIR filter parameter (coefficient) space;

3.The global model search method determines the high reliability of finding an effective solution of the INP external problem. No good initial approximation (prototype) is required here. As a rule, for design specifications with complex selective requirements, the optimal solution is determined not from the starting point specified by the user, but from the point generated by the search algorithm for solving the problem itself.

4.The necessary gain scaling in cascade structures can be provided directly during the INP synthesis of an integer filter. There is no need to use indirect methods of gain scaling using, for example, Lp-norms.

5.The integer discretization of the filter parameter (coefficient) space allows to obtain design integer solutions, which ensures maximum performance when the filter operates in real time and removes all restrictions on the arithmetic of calculations when implementing it on any digital platforms (signal processors, controllers, FPGA) with a given bit ness of data presentation, as well as on chips of custom-made or semi-custom-made VLSI's.

5 References

[1] V. Zahariev, E. Azarov, and K. Rusetski, "An approach to speech ambiguities eliminating using semantically-acoustical analysis," Открытые семантические технологии проектирования интеллектуальных систем, no. 8, pp. 211-223, 2018.

[2] N. Alekseenko, V. Burov, and O. Rumyantseva, "Solution of the three-dimensional acoustic inverse scattering problem. The modified Novikov algorithm," Acoustical Physics, vol. 54, no. 3, pp. 407-419, 2008.

[3] N. Bykhovskiy, "ìDigital Signal Processingî No. 1-2018."

[4] N. Gadawe, T. Fathi, S. Qaddoori, and R. Hamad, "Synthesis and Implementation of IIR Filter using VHDL."

[5] M. Bykhovskiy, "19 International Scientific Research Conference “Digital Signal Processing and its Application-DSPA-2017” In the issue:-multirate signal processing-synthesis of the integer IIR filters-radar signal processing."

[6] B. Voinov, "Information technology and systems: the search for the best, original and rational decisions," ed: Moscow: Nauka, 2007.

[7] K. Nakayama, "A discrete optimization method for high-order FIR filters with finite wordlength coefficients," IEICE TRANSACTIONS (1976-1990), vol. 70, no. 8, pp. 735-743, 1987.

[8] Y. Lim, S. Parker, and A. Constantinides, "Finite word length FIR filter design using integer programming over a discrete coefficient space," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 30, no. 4, pp. 661-664, 1982.

[9] Sundhararaj, V., & Meenakshipriya, B. (2021). An Efficient Discrete Wavelet Transform Based Hybrid Image Watermarking Algorithm Using Human Visual Model.

[10] Begum, M., Ferdush, J., & Uddin, M. S. (2021). A Hybrid robust watermarking system based on discrete cosine transform, discrete wavelet transform, and singular value decomposition. Journal of King Saud University-Computer and Information Sciences.

[11] Article submitted 16 October 2021. Published as resubmitted by the authors 29 November 2021.
W. Chen, M. Huang, W. Ye, and X. Lou, "Cascaded Form Sparse FIR .

[12] Filter Design," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 67, no. 5, pp. 1692–1703, 2020.

[13] H. Wang, Z. Zhao, and L. Zhao, "Matrix decomposition based lowcomplexity fir filter: Further results," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 67, no. 2, pp. 672–685, 2019.

Article submitted 1 November 2022. Published as resubmitted by the authors 12 December. 2022.