Multistage biquadratic filter for measuring of electrical power consumption parameters

O I Gordeev and V A Zakharenko

Omsk State Technical University, 11, Mira, 644050 Omsk, Russia

Abstract. The relevance of developing technical solutions to reduce the number of calculation operations and the amount of memory required for the operation of typical structural elements of programs that implement digital signal processing in problems of continuous measurement of electrical parameters is shown. The use of the multi-stage biquadratic filter for solving these problems is proposed. The theoretical comparison of the structure of the above-mentioned filter with the known structures of digital filters is carried out. An experimental comparison of a specific implementation of the multi-stage biquadrate filter with the similar implementation of a digital filter of a known structure is carried out.

1. Introduction
To develop modern systems for commercial or technical electricity metering, it is necessary to solve the problem of continuous measurement of electrical parameters (including power consumption parameters) with a given accuracy and speed of reaction to controlled parameter changes. This imposes significant restrictions on the choice of technical means used to solve this problem. The situation is complicated by the currently observed trends to increase the functional content of measuring instruments, stricter requirements for metrological characteristics, and the need to reduce the cost of the final product. Ignoring the above points may lead to the fact that the developed system will be unclaimed due to a significant level of competition. Sometimes this leads to the fact that requirements for the product can be repeatedly changed during the process of development or modernization.

There are many variants for implementing measuring devices (such as measuring transducers or electricity meters) that can be used in modern systems for commercial or technical electricity metering (for example, [1–2]). As a rule, such devices are based on microcontrollers [1] or digital signal processors [2]. Given that the differences between these components of electronic circuits from the point of view of the subject of this article are not principled, in the future both types of such components will be called "microcontrollers".

The use of microcontrollers allows you to divide the initial task of measuring electrical parameters into two parts: hardware and software. The hardware part is responsible for converting the measured signals to digital form with the required accuracy, and the software part is directly responsible for calculating the required parameters. This separation, in some cases, allows you to make changes to the product by changing only the software part, without affecting the hardware, what reduces the cost of prototyping. Such cases may include, for example, an increase in the number of additional parameters calculated on the basis of directly measured parameters, or the addition of a new data exchange protocol over a digital interface. However, such changes to the software may lead to the fact that the product that until recently met almost all the requirements (except for those that caused the changes)
will no longer meet them. Since this is a continuous measurement, an increase in the number of calculated parameters may lead to the fact that the product will not be able to perform all the necessary calculations during a given cycle (before new data is received from the measuring circuit), due to the fact that the total time for performing calculation operations exceeds the maximum allowed. Another reason for non-compliance with the requirements may be a lack of random access memory or long-time memory.

The above problems arise when the existing stock of characteristics laid down at previous stages of development is not enough to meet the changed technical requirements for the product. It should be noted that in this case we are talking about a specific hardware and software implementation that was considered satisfactory at the previous stages of development (with the same, unchanged technical requirements). The solution to these problems is, respectively, to change the hardware and (or) software implementation. At the same time, changing the hardware implementation entails additional costs both for purchasing new components (usually with improved characteristics, and, as a result, more expensive), and for making prototypes and conducting their tests (this is especially important when upgrading an already manufactured product). Therefore, this option is often undesirable. On the other hand, changing only the software implementation lacks the above disadvantages, but it is difficult to assert about any specific limits when no change in the software part can ensure compliance with the requirements without changing the hardware part. This is especially true in cases where the identified lack of memory or time for calculations is relatively small.

At the same time, the search for a adequate solution to change the software part in some cases may take much more time and resources than would be spent if the decision was made immediately to replace the hardware part with one that has redundant (for this task) characteristics. Therefore, it makes sense at the early stages of development to consider possible options for changing the software part of the product in order to reduce the resources consumed by the microcontroller while maintaining the main functional characteristics. This will allow you to set much more precisely the possible limits of changing the software part when changing technical requirements. At the same time, the software part that uses less memory or completes calculations faster (with the same accuracy) can itself become one of the competitive advantages of the product as a whole.

2. Problem statement

When calculating the average and root mean square values of current, voltage, and power, we use well-known formulas for finding the average and root mean square values based on an accumulated sample of instantaneous values of the corresponding parameters [2]. From the point of view of using digital signal processing methods, the calculation using such formulas is the use of digital low-pass filters (hereinafter referred to as LPF) [3]. There is a huge amount of work devoted to solving the problem of improving various characteristics of digital filters (for example, [4–8]). Most often, such works either suggest new implementations (or design methods) of filters [4–6], or investigate the characteristics of already known filter implementations in relation to a specific problem [7–8]. At the same time, in each specific case, only a separate set of parameters that the researcher considers important from the point of view of the task before him is considered out of the full set of characteristics of digital filters. All other characteristics that are not included in the set are considered to have no significant impact by default. Thus, in [4–5], when comparing various design methods, stability parameters, accuracy, time and frequency characteristics are evaluated, but practical implementation issues such as the required word length of coefficients, the required memory size, computational complexity, and others are not considered. In [6], a variant of implementing a digital filter with integer coefficients is proposed, which reduces the number of logic elements when implemented on an FPGA. In [7], two types of digital filters are compared in terms of minimizing the word length of coefficients and their frequency characteristics, and in [8], four types of filters are compared in terms of their application for audio signal processing. All these works have their own scope of application and it is impossible to create a filter implementation that would satisfy all application tasks without exception. In relation to the problem of using digital LPF in measuring
devices for real-time calculation of electrical parameters in industrial frequency networks, no work has been found that offers any solutions other than the well-known structures of digital filters [3].

In this regard, the task was set to determine the advantages and disadvantages of using the multistage biquadratic filter proposed by the authors [9] for measuring the electric power consumption parameters in comparison with recursive filters designed using standard methods. The characteristics of the compared filters will be considered from the point of view of their implementation in a measuring device (such as, for example, in [2]). In this case, the calculation of various parameters will use, in fact, the same type of LPF [2–3]. This means that even a slight improvement in the parameters of such a LPF will have a significant impact on the program parameters as a whole (the effect will be a multiple of the number of identical filters).

3. Theory

As is known [3], the relationship between the input and output samples of any digital filter is determined by its transfer characteristic \( H(z) \). In general, for a biquadrate filter (which in some sources also has the name ”filter with an infinite impulse response (IIR filter) of the second order”), the transfer characteristic has the form:

\[
H(z) = \frac{b_0 + b_1z^{-1} + b_2z^{-2}}{1 - a_1z^{-1} - a_2z^{-2}}.
\]

where \( b_0, b_1, b_2 \) are the coefficients of the feedforward loop; \( a_1, a_2 \) are the coefficients of the feedback loop; \( z \) is a complex variable, and the expression \( z^{-k} \) indicates the signal delay for \( k \) sampling cycles; this is usually implemented by storing the value in random access memory until the value is needed to perform calculations.

It is also known [3] that when two linear digital filters with transfer characteristics \( H_1(z) \) and \( H_2(z) \) are connected in series (cascade), the total transfer characteristic \( H_0(z) \) of the resulting system will be equal to:

\[
H_0(z) = H_1(z) \cdot H_2(z).
\]

One of the most well-known and most frequently used structures of IIR filters in general and biquadratic filters in particular are "Direct form I" and "Transposed direct form II", shown in Figure 1 and Figure 2 respectively [3].

![Figure 1. The structure "Direct form I" of a biquadratic filter.](image)

According to [3], The "Direct form I" structure is the most stable and less sensitive to rounding errors, while the "Transposed direct form II" structure contains fewer delay elements (designated as \( z^{-k} \)), which means that it requires less amount of random access memory for its operation.
If we analyze the "Direct form I" structure, we can pay attention to the fact that it is a series connection of a feedforward loop and a feedback loop, while the feedforward loop is a filter with a finite impulse response (FIR filter). If we assume that the feedback loop can also be considered as a separate filter, then, given the formula (1) and (2), it can be argued that biquadratic filter structures "Direct form I" is, in fact, the series connection of the FIR filter with three coefficients and the IIR-filter with a missing feedforward loop (we can also say that this is an IIR filter with one coefficient in feedforward loop, which is equal to one). Considering also that the three-coefficient IIR filter does not provide any acceptable filtration quality in the vast majority of implementations [3], we can conclude that the parameters of the IIR filter with the missing feedforward loop are the determining parameters for the biquadratic filter as a whole.

It follows that replacing the three coefficients in the feedforward loop with one equivalent coefficient will most likely not have a significant impact on the characteristics of the biquadratic filter, but it will reduce both the number of calculation operations required to obtain a single output sample \( y(n) \), and reduce the amount of memory required for normal operation of the filter by reducing the number of delay elements. This is especially true when several biquadratic filters are connected in series, which are usually used instead of high-order digital filters due to the excessive complexity of ensuring stability and the increased requirements for the bit rate of the latter [3].

Based on the above considerations, the multistage biquadratic filter was developed, the structure of a single cascade of which is shown in Figure 3.

**Figure 2.** The structure "Transposed direct form II" of a biquadratic filter.

**Figure 3.** The structure of a single cascade of the multistage biquadratic filter.
The transfer characteristic of the structure shown in Figure 3 has the form:

\[ H(z) = \frac{b_0}{1 - a_1 z^{-1} - a_2 z^{-2}}. \]  

(3)

Comparing the structure of a single cascade of the multistage biquadratic filter with the well-known structures "Direct form I" and "Transposed direct form II", we can note that the latter require more operations and a larger amount of memory for storing coefficients. The number of delay elements that also require allocation of a certain amount of memory to implement their functionality are shown in Figure 2 and Figure 3 are the same, but the structure in Figure 1 — twice as much, that is, the "Direct form I" structure definitely requires more memory costs. Therefore, considering also that, according to [3], the structures "Direct form I" and "Transposed direct form II" from the theoretical point of view have the same characteristics, the multistage biquadratic filter will be compared with a multistage filter having the structure "Transposed form II".

4. Experimental results

Let's look at a specific example. Two LPF filters are compared: a filter with the "Transposed direct form II" structure, whose coefficients were calculated using the bilinear transformation method based on the analog Butterworth filter (hereinafter referred to as the LPF-1), and a multistage biquadratic filter with identical coefficients (hereinafter referred to as the LPF-2). Shared data:

- both filters were implemented as a program for the STM32F373CC microcontroller, written in the C programming language;
- microcontroller clock frequency: 64 MHz;
- sampling rate: 10.245 kHz;
- number of cascades of biquadratic filters: 3;
- cutoff frequency of each of the cascades of biquadratic filters: 25 Hz;
- data storage format: float (floating-point number);
- number of bits: 32.

The coefficients of the filters are shown in Table 1.

| Coefficient | Value for LPF-1 | Value for LPF-2 |
|-------------|-----------------|-----------------|
| \(b_0\)     | 0.000058136319157505 | 0.000232545276630021 |
| \(b_1\)     | 0.000116272638315011 | – |
| \(b_2\)     | 0.000058136319157505 | – |
| \(a_1\)     | 1.97831828719267000 | 1.97831828719267000 |
| \(a_2\)     | -0.97855083246930000 | -0.97855083246930000 |

To calculate ten output samples based on a block of ten input samples, the LPF-1 will require 1696 bytes of memory and 11.875 microseconds (760 clock cycles), and the LPF-2 will require 1568 bytes of memory and 8.375 microseconds (536 clock cycles). In other words, LPF-2 requires 128 bytes (7.5 %) less memory and performs calculations 1.4 times faster than LPF-1. The presented values were obtained by analyzing the compiled code of programs that implement the functionality of each individual filter, and confirmed by software debugging tools for the code executed by the microcontroller.

At the same time, the speed of performing calculation operations and the amount of memory used are not the only characteristics that can be compared to give preference to a particular type of filter. In
In general, from the point of view of application in real problems, the main parameters of digital filters are [3]:

- frequency response;
- phase response;
- step response (or impulse response);
- group delay time.

For the tasks of measuring parameters of electrical signals, comparison by phase response and group delay time makes sense in cases when the phase distortion of individual harmonic components has a significant impact on the results of calculations (this applies primarily to power quality analyzers), or when using complex schemes of interaction of filters with different parameters. In other words, for most of the measurement tasks, it is sufficient to compare filters for frequency and step responses.

The frequency response was calculated based on formulas (1) and (3) using the method presented in [3]. The results of the frequency response calculations are shown in Figure 4 and Table 2.

![Figure 4. Frequency responses of the investigated filters.](image)

| Frequency, Hz | Magnitude for LPF-1, dB | Magnitude for LPF-2, dB | Difference between magnitudes, dB |
|---------------|-------------------------|-------------------------|----------------------------------|
| 0             | 0                       | 0                       | 0                                |
| 25            | -9.0312                 | -9.0296                 | -0.0015                          |
| 50            | -36.9172                | -36.9111                | -0.0061                          |
| 100           | -72.3142                | -72.2897                | -0.0245                          |
| 200           | -108.4394               | -108.3413               | -0.0981                          |
| 500           | -156.5343               | -155.9193               | -0.6150                          |
| 1000          | -193.9175               | -191.4279               | -2.4896                          |
| 2000          | -235.5469               | -225.0618               | -10.4851                         |
| 4000          | -307.3146               | -250.7003               | -56.6143                         |
The differences between the frequency response of LPF-1 and LPF-2 become significant as we approach a frequency equal to half the sampling frequency (Nyquist frequency, which equal, in this case, 512.5 Hz). The difference does not exceed 0.1 dB at frequencies less than 200 Hz, but at frequencies above 1 kHz it reaches a value of 2.5 dB or higher. On the other hand, at frequencies above 1 kHz, the filter suppression level exceeds 190 dB, which is more than enough for most measurement tasks. In other words, as the difference between the frequency responses of LPF-1 and LPF-2 increases, the practical effect of this increase on the measurement results becomes less significant. However, specific conclusions about the permissibility of using LPF-2 should be made only when analyzing its frequency response from the point of view of a specific task.

Step response allows you to judge the speed of the product's response to a sudden change in the input signal. This characteristic is quantified by a parameter known as "response time", which characterizes the duration between the moment when the input signal transitions to a new steady state and the same moment for the output signal. The moment when the output signal enters the steady state is considered to be the moment when the output signal takes the expected value within the permissible error and remains within these limits until the input signal changes again.

The step response of the investigated filters was determined by applying a constant signal equal to one to the input after a long period of absence of the input signal. All output samples of the filters were fixed in software memory. The general view of the obtained step responses is shown in Figure 5, and the scaled view is shown in Figure 6. The response time of each of the filters, determined by the step responses, is shown in Table 3.

![Figure 5. Step responses of the investigated filters (general view).](image1)

![Figure 6. Step responses of the investigated filters (scale view).](image2)
Table 3. Response times of the investigated filters at various required setting limits.

| Required setting limits | Response time for LPF-1, ms (samples) | Response time for LPF-2, ms (samples) | Difference between response times, ms (samples) |
|-------------------------|-------------------------------------|-------------------------------------|-----------------------------------------------|
| ±1.00 %                 | 38.555 (396)                        | 38.263 (393)                        | 0.293 (3)                                     |
| ±0.50 %                 | 59.834 (614)                        | 59.541 (611)                        | 0.293 (3)                                     |
| ±0.20 %                 | 66.374 (681)                        | 66.081 (678)                        | 0.293 (3)                                     |
| ±0.10 %                 | 69.302 (711)                        | 69.009 (708)                        | 0.293 (3)                                     |
| ±0.05 %                 | 86.286 (885)                        | 85.993 (882)                        | 0.293 (3)                                     |

As can be seen from Table 3, the response time of the LPF-2 is less than the response time of the LPF-1 by 0.293 ms (3 clock cycles), and this difference does not depend on the required setting limits, therefore, the LPF-2 can respond faster to changes in the input signal than the LPF-1.

5. Discussion of results
As a result of experimental comparison of LPF-1 and LPF-2, it was found:

- the LPF-2 performs computing operations 1.4 times faster than LPF-1;
- the LPF-2 requires 128 bytes (7.5 %) less memory than the LPF-1;
- the LPF-2 frequency response is worse than the LPF-1 frequency response. The difference between the frequency response depends on the frequency and is greatest in the immediate vicinity of the Nyquist frequency;
- the response time of the LPF-2 is less by 0.293 ms (3 clock cycles) than that of the LPF-1, regardless of the required setting limits.

6. Conclusion
A theoretical and experimental comparison of the multistage biquadratic filter and digital filters with known structures is carried out on the example of a specific implementation of a LPF for continuous measurement of electrical parameters. The multistage biquadratic filter has a higher speed of performing calculation operations and requires less memory for its operation. In addition, it was found that the response time of the multistage biquadratic filter is less than that of a similar filter with the "Transposed direct form II" structure. The disadvantage of a multi-stage biquadrate filter is the degraded frequency response (compared to a similar filter), but this degradation is significant only at frequencies close to the Nyquist frequency.

References
[1] Surajudeen-Bakinde N T, Ayodele S O, Oloruntoba T D, Otuoze A O and Faruk N 2017 Development of an Internet based prepaid energy meter 2017 IEEE AFRICON, Cape Town 1370–1373 DOI: 10.1109/AFRICON.2017.8095681
[2] Hribik J, Fuchs P, Hruškovic M, Michálek R and Lojko B 2005 Digital Power Network Parameters Measurement Radioengineering Vol. 14 No. 2 19–26
[3] Lyons R G 2010 Understanding digital signal processing (London: Pearson Education) p 954
[4] Dumitrescu B and Niemisto R 2004 Multistage IIR filter design using convex stability domains defined by positive realness IEEE Transactions on Signal Processing Vol. 52 No. 4 962–974 DOI: 10.1109/TSP.2004.823497
[5] Saha S K, Kar R, Mandal D and Ghoshal S P 2014 Gravitation search algorithm: Application to the optimal IIR filter design Journal of King Saud University — Engineering Sciences 26 73–78 DOI: 10.1016/j.jsues.2012.12.003
[6] Karimov T I, Butusov D N, Andreev V S, Rybin V G and Kaplun D I 2018 Compact Fixed-Point Filter Implementation 22nd Conference of Open Innovations Association (FRUCT), Jyvaskyla 73–78 DOI: 10.23919/FRUCT.2018.8468289

[7] Mingazin A T 2004 Minimal coefficient wordlength of two half-band IIR filter structures DSPA 2004, Moscow 1 46–47

[8] Sutradhar S R, Sayadat N, Rahman A, Munira S, Haque A K M F and Sakib S N 2017 IIR based digital filter design and performance analysis 2nd International Conference on Telecommunication and Networks (TEL-NET), Noida 1–6 DOI: 10.1109/TEL-NET.2017.8343596

[9] Gordeev O I Multistage biquadratic filter Patent RU 2716902 C1 2020

Acknowledgments
The reported study was funded by RFBR, project number 19-38-90166.