УДК 004.932.4

MATHEMATICAL SIMULATION OF DIGITAL FILTERS WITH FINITE PULSE CHARACTERISTICS

Pesetskaya A.A., Sosna I.A.

Belarusian State University of Informatics and Radioelectronics,

Minsk, Belarus

Daneiko T.M. – PhD in Physics and Mathematics

Abstract. The article discusses the methodology for optimizing the structure of decimation filters using mathematical modeling. This can significantly reduce the requirements for FPGA resources. The study of the architecture of the FIR resampling filter with a buffer is divided into stages.

A finite impulse response (FIR) digital resampling filter is a filter which impulse response (response to any finite length input) has a finite duration, since it ends to zero in a finite time.

An FIR filter can be used to implement almost every kind of frequency response in digital form and is itself implemented with help of using multipliers, adders and delay elements [1].

When implementing FIR filters in field-programmable gate arrays (FPGA), their capabilities are significantly limited by some FPGA resources. The most "deficient" resource usually turns out to be a resource, such as multipliers.

Due to the lack of multipliers, developers have to use more expensive FPGAs, reduce the filter order, and reduce the number of available digital strip values. As a result, this negatively affects the technical characteristics of the final product.

The methodology for optimizing the structure of FIR decimation filters consists in performing the following actions:

– Calculate the corresponding coefficients of the FIR filter and the order of the digital filter.

The characteristic of a classical FIR filter can be represented as:

$$y(n) = \sum_{i=0}^{n} b_i x(n - \mathrm{i}) \qquad (1)$$

where n is the order of the filter,

b_i is filter coefficient.

The block diagram of a non-recursive FIR filter is shown in Figure 1.

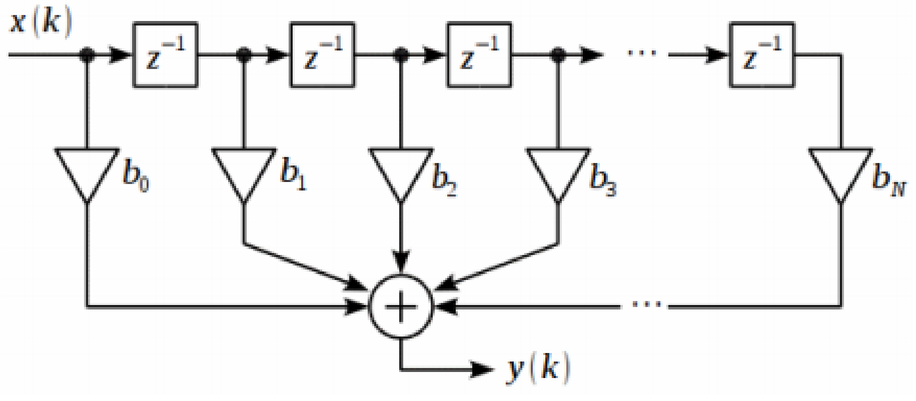

Fig. 1 – Block diagram of a non-recursive FIR filter

An order n FIR filter contains **n** delay lines and **n + 1** coefficients. If the coefficient **b0 = 1**, then we have an FIR filter of order **n**, for which multiplication by **b0 = 1** will be trivial.

The impulse response of an FIR filter is always finite and exactly matches the filter coefficients. The array of such filters allows us to implement **m** different nominal values of digital bands, where **m** is any integer.

– Estimate the need for resources (such as multipliers) for this filter array:

$$M = (n + 1) \times m \qquad (2)$$

– Consider as an alternative solution the architecture of the buffer FIR decimation filter shown in Figure 2.

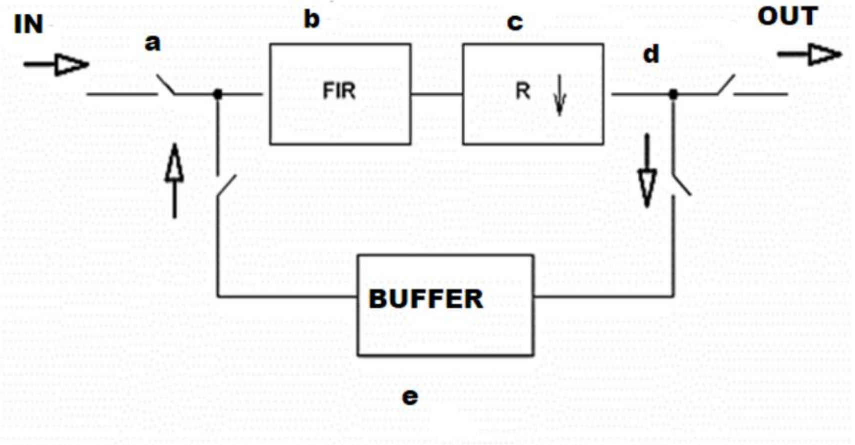

Fig. 2 – Block diagram of the buffer FIR filter:

a - input logic switch;

b - halfband FIR filter;

c - decimation block with a coefficient R = 2;

d - output logic switch;

e - buffer.

The input for this filter are samples of a digital signal, pre-converted to 0 Hz.

- Evaluate the data processing scheme with this filter.

1. Data enters the input logic switch a) and is redirected to the input of the FIR filter b), which is designed primarily to remove aliasing in the input signal during decimation.

2. After filtering, the clock frequency of the signal is halved using the decimator c).

3. After decimation, the data goes to the output switch d). If the required signal bandwidth is not reached, the data is redirected to the buffer e).

4. After being filled, the buffer returns the data block through the input switch a) for the next iteration to narrow the bandwidth.

5. Steps 2-4 are repeated until the required signal bandwidth is reached. It is obvious that for block data, that due to the cyclic structure of signal processing, it is possible to implement any decimation coefficients on one filter, if they are multiples of **2k**, where **k** = 1, 2, 3, ...

This filter allows you to process streaming (continuous) data with decimation by half and block data (data blocks with a fixed number of samples) with any decimation factor divisible by **2k**, i.e. it allows you to create unlimited digital signal bands.

For streaming data, the specified filter can be applied with a decimation factor of 2. In this case, the filter passes the data through itself once without using a buffer. The use of a filter with decimation coefficients of more than 2 is possible, but not so effective due to a significant reduction in suppression in the attenuation band.

- Consider additional possibilities for optimizing the FIR filter requirements for the number of multipliers required for its implementation in modern field-programmable gate arrays.

For the analysis, it is necessary to introduce the following restrictions into the filter structure: let us assume that every second filter coefficient is equal to zero and the filter coefficients are symmetric about the central coefficient. This filter requires only one multiplier for every four orders of magnitude for implementation, in contrast to the classical FIR filter.

- Take into account when designing the filter that a necessary and sufficient condition for the equality of every second coefficient to zero is that the filter should be half-band, or in other words, cutoff frequency (**Wpass**) and stop frequency (**Wstop**) should be symmetric about the half the sampling frequency (**Fs / 2**) [2].

The converse is also true. For a half-band filter with cutoff frequencies symmetric about the frequency **Fs / 2**, every second coefficient is equal to zero. It should be noted, however, that the cutoff and stop frequencies should be as close as possible to the half of the sampling rate as possible. Otherwise, the equality of every second sample to zero can significantly distort the filter characteristics.

- When implementing a filter buffer, it is necessary to take into account that an FIR filter is a non-recursive filter or a convolution filter. FIR filters convolve their coefficients with a sequence of input data samples, while the resulting data volume increases according to the formula:

$$К = К_d + К_f - 1 \qquad (3)$$

where **K** is the number of elements in the output sequence;

$\mathbf{K_d}$ is the number of input data samples;

$\mathbf{K_f}$ is the number of filter coefficients.

In this case, the first $\mathbf{K_f}$ **-1** samples of the output sequence are not valid.

The fact of an increase in the number of samples at the output can be explained as follows: the FIR filter is designed as a delay line, the outputs of which are multiplied by coefficients and summed up. The length of the delay line is equal to the order of the filter. Therefore, until the entire delay line is full, the output data is not valid.

In order for all output data to be valid and constant in size, the first $\mathbf{K_f}$ **-1** samples must be discarded. It must be done also when data is returning to the next iteration.

One of the important properties of the filter is the linearity of its phase-frequency characteristic (PFC). The non-linear PFC of the filter distorts the signal, i.e. different frequencies acquire different phase shifts and, accordingly, different time delays at the filter output. This should be taken into account, since, for example, in coherent processing, phase distortion is unacceptable. Therefore, the more linear the PFC – the better the filter.

The proposed architecture has some restrictions on its application, namely: the filter can be effectively applied with a decimation factor of more than 2 only with block data; decimation coefficients can only be multiples of **2k**; the filter has a relatively "rigid" structure, namely, every second coefficient is equal to zero, and the coefficients must be symmetric about the central coefficient [3]. However, this structure is easily implemented by using standard mathematical modeling tools.

Thus, completing of an independent computational study for optimizing the structure of FIR decimation filters, according to the proposed method, will allow students to effectively master the material.

Sources

1. Karbushov, Ch.S. Development of an FIR filter using a distributed arithmetic architecture // Technical sciences: problems and prospects: materials of the V International scientific conference. - 2017.

2. Steven W. Smith, The Scientist and Engineer’s Guideto Digital Signal Processing, Second Edition, 1999, California Technical Publishing, P.O. Box 502407, San Diego, CA 92150.

3. Theory and practice of digital signal processing [Digital resource]: Structures of digital filters and their characteristics. – Access mode: http://www.dsplib.ru/content/filters/ch10/ch10.html