Interactive design of digital filter using LabVIEW

Valentina Markova [1]

Abstract – **LabVIEW is powerful and flexible programming language for data acquisition, analysis and presentation of results. This paper introduces LabVIEW as intuitive tools for design, analysis and implementation of digital filters. It allows students and researchers to design and analyse DSP systems easily and for a shorter time, compared to Matlab. The aim of this paper is to present a set of VIs for design of digital filters with different interactive design options.**

Keywords – **Digital filter, LabVIEW, Virtual instruments.**

I. Introduction

LabVIEW is a graphical programming language, which has become prevalent among researchers and engineers in the industry during the last years. LabVIEW programs are called virtual instruments (VIs) and have two parts - a front panel and a block diagram. The front panel is the user interface. The block diagram contains the program code [1]. In case of filter design the VI reads the desired parameters of the filters entered by the user on the front panel and determines its characteristics or filter coefficients [2].

The NI Digital Filter Design Toolkit (DFDT) extends LabVIEW with tools for design, analysis and implementation of variety of IIR and FIR filters. It contains very useful teaching tools, which allow immediate feedback after specifying the desired filter requirements. The toolkit also includes a set of Filter Analysis VIs for evaluation of all filter characteristics – frequency and phase response, impulse and step response, group delay and zero/pole placement [3]. LabVIEW follows a dataflow execution model, which makes the learning process easier. Working within LabVIEW environment has many advantages, such as the ability to perform filter testing with simulated signals or real signals, acquired by real data acquisition device or sound card.

The DFDT includes a comprehensive set of design algorithms ranging from classical (Butterworth, Chebyshev, window, etc.) to modern optimizing options, including Remez exchange and least Pth norm [4].The Special Filter Design VIs helps for designing special digital filters like IIR notch filters, IIR comb filters, maximally flat filters, narrowband filters and delay compensators. The toolkit also includes the Multirate Filter Design VIs appropriate for design, analysis and implementation of single-stage and multistage multirate filters.

The aim of this paper is to present a set of VIs for design of digital filters with different interactive design options. The effect of the repeated interactive design changes is that the designer has the possibility to observe how different design parameters affect the filter performance.

[1]Valentina Markova is with the Faculty of Electronics, Department of Telecommunications at Technical University of Varna, 1 Studentska str., Varna 9010, Bulgaria.
E-mail: via@tu-varna.bg.

II. Design of FIR filters with LabVIEW

Finite impulse response (FIR) filters, also known as non-recursive filters operate only on the current and the past input values of the signal. There are some advantages in using FIR filter, since it can be designed with exact linear phase and the filter structure is always stable with quantized filter coefficients [5]. The price for this linearity of the phase response of the filter is the higher computational requirements.

FIR filters perform a convolution of the filter coefficients with a sequence of input values and produce an equally numbered sequence of output values, as shown in Eq.1:

$$y_i = \sum_{k=0}^{n-1} h_k x_{i-k} \qquad (1)$$

where x is the input sequence, y – the output sequence, and h is the FIR coefficients.

The DFDT facilitates the design of the digital filter by specifying its parameters, as well as by offering numerous display options. The design of a FIR filter is based on a direct approximation of the specified magnitude response. The toolkit includes a variety of VIs and functions for the design of FIR filters. The linear phase FIR filters can be designed using either Remez exchange method or windowed Fourier series. An example of FIR windowed design is shown on Fig.1.

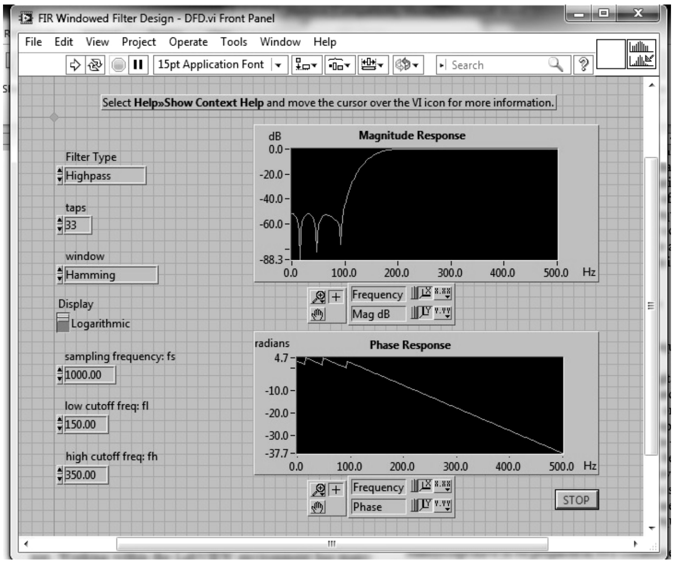


Fig.1. FIR Windowed Filter Design VI

Using FIR Windowed Filter Design VI, the design of the filter can be specified by entering the filter specifications (sampling frequency – fs, low cutoff frequency – fl, high cutoff frequency – fh and window function).

The design results are displayed immediately in the shape of magnitude and phase response. The filter information also contains the order of the filter. The DFDT supports variety of smoothing windows (Hanning, Hamming, Triangular, Kaiser-Bessel and so on).

The other VI - FIR Windowed coefficients VI, generates the set of filter coefficients.

The second design method - object of the current manuscript is the equiripple linear phase FIR design, using Remez exchange algorithm. Using DFD Remez Design VI (Fig. 2), it is possible to design a variety of filter types by adjusting the filter specifications by changing the passband and stopband parameters or the number of the taps. Equiripple design equally weights the passband and stopband ripple and produces filters with a linear phase characteristic.

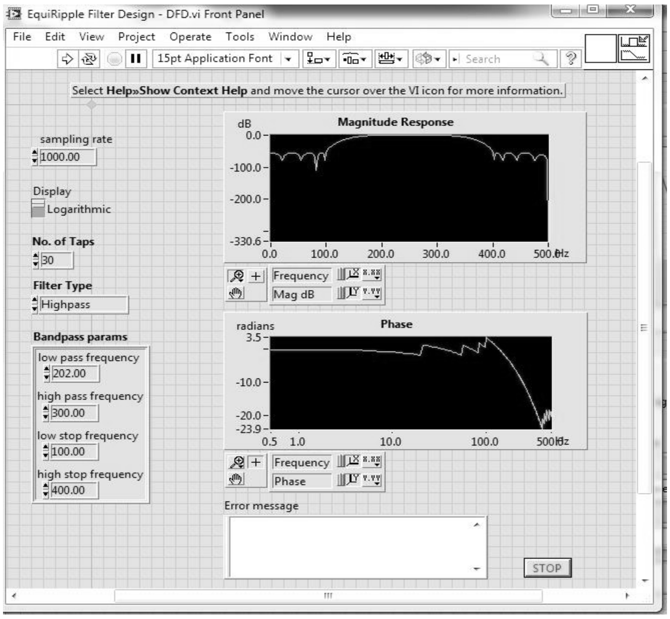


Fig.2. EquiRipple FIR Filter Design VI

The block diagram of the described VI is presented in Fig.3.

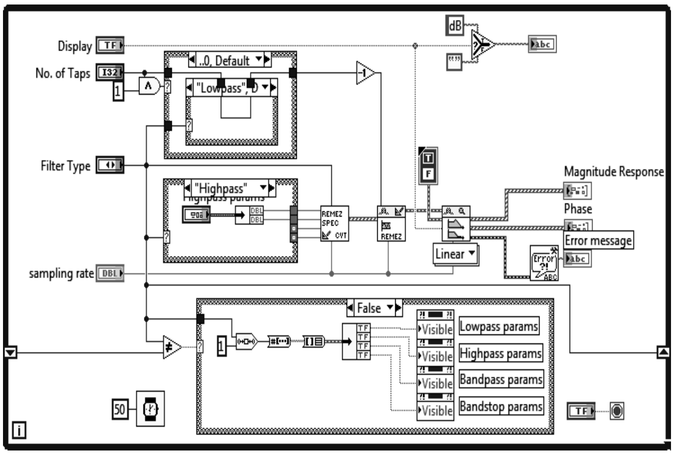


Fig.3. The screenshot of the block diagram of EquiRipple FIR Filter Design VI

III. Design of IIR filters with LabVIEW

Infinite impulse response (IIR) filters, also known as recursive filters, operate on the current and the past input values as well as the current and the past output values of the signal. The IIR filters can achieve the same level of attenuation as the FIR filters, but with less number of coefficients. Therefore, an IIR filter can provide a significantly faster and more efficient filtering than an FIR filter.

DFDT offers a big variety of design options. In the first group VIs the design of the filter can be performed by entering the filter specifications. The filter characteristics or its coefficients are directly available in forms defined by used VI. There is also an option for designing of a filter by entering directly the transfer function by the user in the front panel. The final design option includes defining of the filter characteristics, by placing the poles and zeros of the transfer function in the pole zero diagram.

Express VIs are special tools in DFDT used for quick creation of digital filters by interactively specifying the filter parameters.

A. Design of IIR filter by specifications

The most common design methods are based on classical analogue filter approximations –Bessel, Butterworth, Chebishev, Inv. Chebishev and elliptic.

Using IIR Digital Filter VI (Fig.4), the designer can design a variety of filter types (Lowpass, Highpass, Bandpass, Stopband) by changing the design method, or by adjusting the filter specifications, by entering different values of the input parameters in the Front Panel of the VI. The available design methods are Elliptic, Butterworth, Bessel [6], The filter order is estimated automatically. The execution of the VI returns the magnitude and phase responses of the filter.

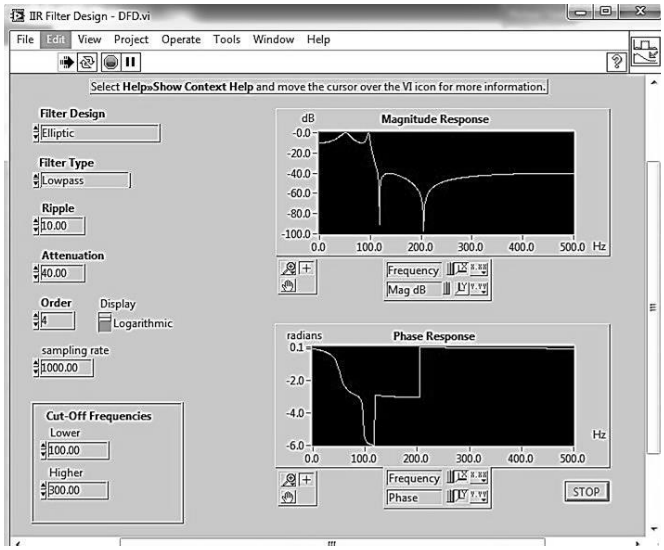


Fig.4. The screenshot of the Front panel of IIR Filter Design VI

B. Design of IIR filter from transfer function

DFDT also has VI for creating filters by entering directly the filter coefficients, as presented in Fig.5.

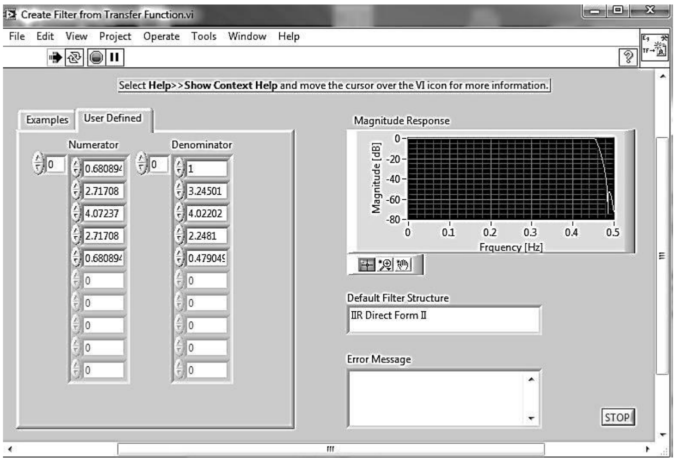


Fig.5. The screenshot of the Front panel of IIR Filter Design VI

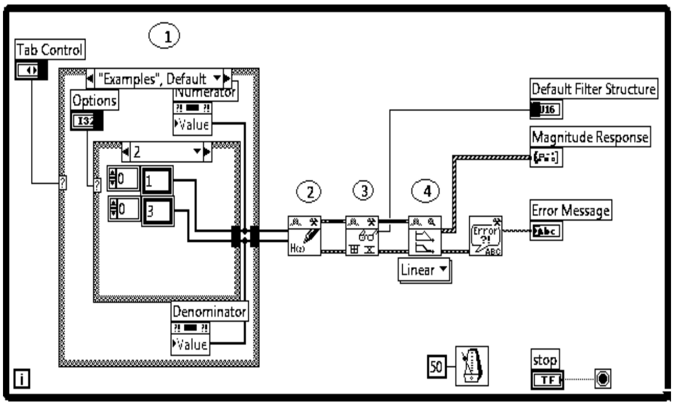


Fig.6. The screenshot of the Front panel of IIR Filter Design VI

Running this VI returns the magnitude response and the used structure for realization of the filter. The design procedure in Fig.6 is following:

1. All the values of the coefficients in the numerator and the denominator of the filter are entered by user in two arrays in the Case Structure [nihepl].

2. The next function creates a filter from the corresponding transfer function.

3. The next step is to retrieve the filter structure.

4. The final step is plotting the frequency responses of the filter by using waveform graph.

C. Design of IIR filter from Pole Zero Placement

Express VIs are very useful tools in DFDT, that help for the quick creation of digital filters. For instance, with Pole-Zero Placement VI you can interactively place zero and pole on the z-plane by entering the exact values or clicking on the pole-zero graph to move or place them with the mouse. Every change of the mentioned above parameters, affects immediately the shape of the magnitude response shown on fig. 7.

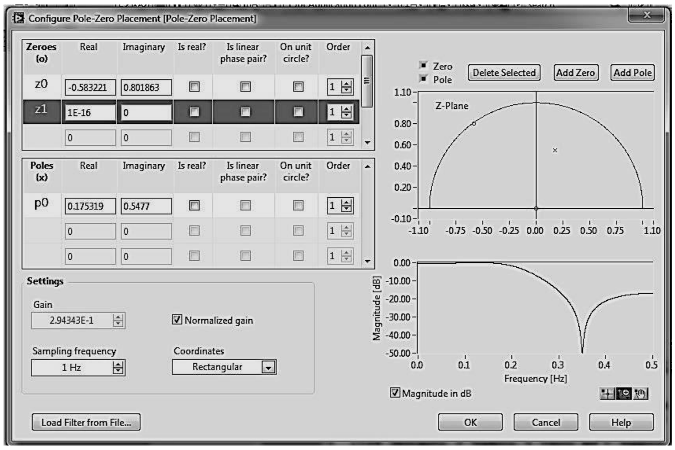


Fig.7. The screenshot of the Front panel of IIR Filter Design VI

IV. RESULTS

Design and implementation of digital filters with LabVIEW DFDT consist of several interactive steps. Some of the advantages of LabVIEW are, that the design parameters can be changed at the time of execution of the program and the execution results are immediately displayed. The result of the constantly repeating (with different input parameters) graphical design procedures is that the designer can see how these parameters affect the filter performance.

In this section, the efficiency, flexibility and interactivity of the LabVIEW environment is demonstrated by an example.

Example: Design a lowpass digital filter with following specifications: passband frequency – f_{pass} = 3.4kHz, stopband frequency– f_{stop} = 3.8kHz, passband ripple – 0.1, stopband attenuation – 60dB.

For design, analysis and implementation of the filter, the VI Design Filter step by step is used. The interface supports the following design methods: for FIR – Kaiser window, the Dolph-Chebyshev window and equiripple, for IIR – elliptic, Chebyshev, Inverse Chebyshev, and Butterworth.

The design starts with entering the filter specifications in “design a new filter” window, shown in fig.8. Initially, with the design method set to equiripple FIR, the order of the filter fulfilled the required parameters is 46. The respective characteristics of the filters such as the impulse response, step response, phase response, magnitude response, group delay, pole-zero diagram are immediately plotted in the window “Design result”. Analysis of these characteristics is possible by using the dropping menu (marked as 1 on fig.8).

However, it is possible to select other design methods and to get immediately the new filter order, as well as the actual filter shape. For the given specification, as it is expected the elliptic design requires the minimum filter order – 5. Some of the result characteristics of the filter are shown in fig.9.

With the previous design method, the FIR filter requires higher filter order, but in behalf of it has linear phase response. The choice between both filter classes (IIR or FIR linear phase) depends on all given criteria. With the present example it is ease to observe the performance of both filter classes. This flexibility of the design process, as well as the option for constant change of the parameters and analysis of the respective new characteristics, makes this software perfect for education. The main advantages and disadvantages of every specific design method and respectively the structure used for its realisation is visualised in a simple and understandable manner.

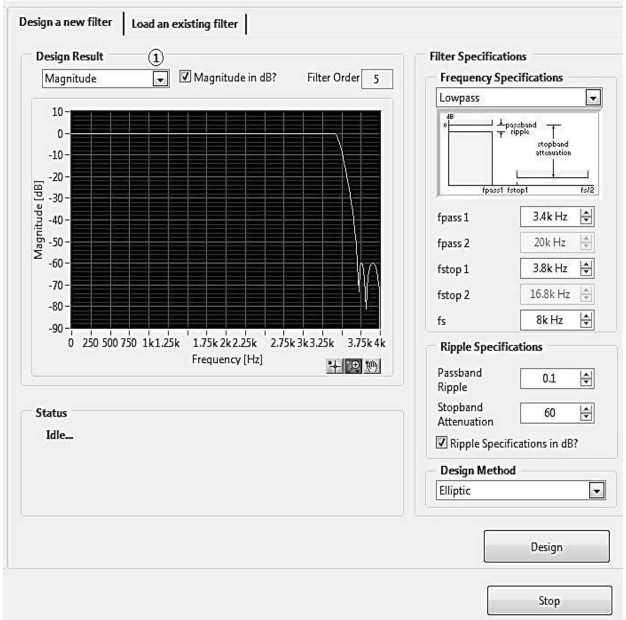
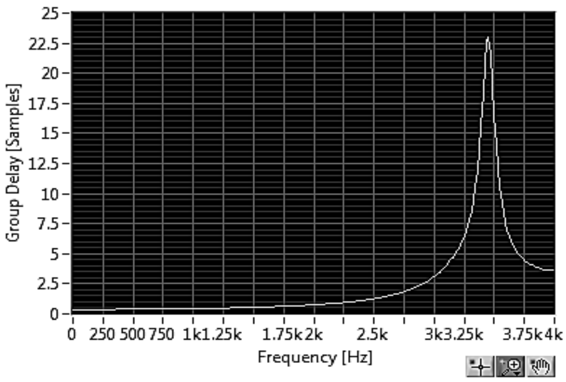
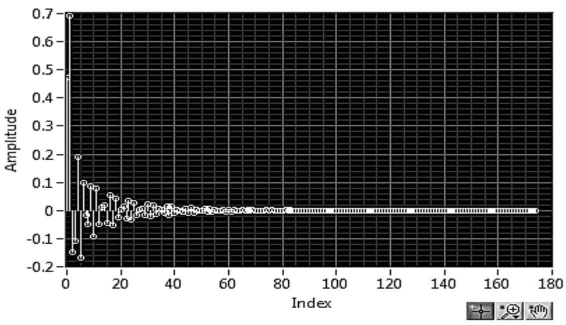


Fig.8. The screenshot of the Front panel of Filter Design VI



a) Group delay of Lowpass IIR filter



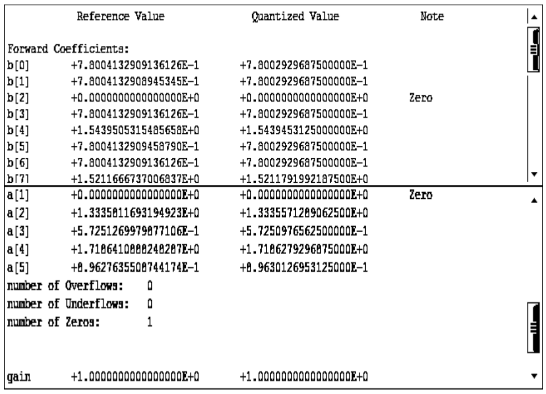
b) Impulse responseof Lowpass IIR filter

Fig.9. The screenshot of the Front panel of IIR Filter Design VI

The coefficients of the obtained filter are directly available in forms such as direct I and II, cascade, lattice form and so on. The filter coefficients that are result of direct form I realization of the Elliptic IIR design, are presented on fig.9, where b[i] are forward coefficients and a[j] – reverse coefficients of the from Eq.2.

$$H(z) = \frac{\sum_{j=0}^{m} b_m z^{-m}}{1+\sum_{i=1}^{n} a_n z^{-n}} \qquad (2)$$

The final step is the implementation of the best filter using a general-purpose PC, a DSP or in an FPGA.



	Reference Value	Quantized Value	Note
Forward Coefficients:			
b[0]	+7.8004132909136126E-1	+7.8002929687500000E-1	
b[1]	+7.8004132908945345E-1	+7.8002929687500000E-1	
b[2]	+0.0000000000000000E+0	+0.0000000000000000E+0	Zero
b[3]	+7.8004132909136126E-1	+7.8002929687500000E-1	
b[4]	+1.5439505315485658E+0	+1.5439453125000000E+0	
b[5]	+7.8004132909458790E-1	+7.8002929687500000E-1	
b[6]	+7.8004132909136126E-1	+7.8002929687500000E-1	
b[7]	+1.5211666737006837E+0	+1.5211791992187500E+0	
a[1]	+0.0000000000000000E+0	+0.0000000000000000E+0	Zero
a[2]	+1.3335811693194923E+0	+1.3335571289062500E+0	
a[3]	+5.7251269979877106E-1	+5.7250976562500000E-1	
a[4]	+1.7186410888248287E+0	+1.7186279296875000E+0	
a[5]	+8.9627635508744174E-1	+8.9630126953125000E-1	
number of Overflows:	0		
number of Underflows:	0		
number of Zeros:	1		
gain	+1.0000000000000000E+0	+1.0000000000000000E+0	

Fig.8. The screenshot of the Front panel with the Filter's coefficients

V. Conclusion

LabVIEW is a flexible, intuitive graphical event-based programming language. LabVIEW programs are called virtual instruments. In this paper a set of VIs for design of digital FIR and IIR filters with different interactive design options are given. Some of the advantages of LabVIEW are, that the design parameters can be changed at the time of execution of the program and the execution results are immediately displayed. The effect of the repeating interactive design changes is that the designer has the possibility to observe, how different design parameters affect the filter performance. The interactive tools are excellent teaching tools, because they provide immediate feedback after specifying the desired filter parameters.

References

[1] J. Jerome, *Virtual Instrumentation using LabVIEW*, PHI Learning Pvt., 2010

[2] Y. Singh, S. Tripathi and M. Pandey, "Analysis of Digital IIR filter with LabVIEW", International Journal of Computer Applications, Vol. 10, No. 06, pp.23-30, 2010.

[3] National Instuments, LabVIEW Digital Filter Design Toolkit User Manual, 2012.

[4] National Instuments, Interactively Design Digital Filters with LabVIEW, 2014.

[5] S. Mitra, *Digital signal processing*, A Computer based approach, 4th edition, Sept. 2010.

[6] P. Podder, Md. M. Hasan, Md. R. Islam, M. Sayeed, "Design and implementation of Butterworth, Chebyshev –I and Elliptic filter for speech signal analysis", Int. Journal of Computer Applications, Vol.98, No.7, July2014.