CAD SUBSYSTEM FOR DESIGN OF EFFECTIVE DIGITAL FILTERS IN FPGA

Pavel Plotnikov

Vladimir State University, Russia, Gorky str., 87, 600000, plotnikov_pv@inbox.ru

In given article analyze of DF design flows, implemented in modern CAD systems is considered. Design flow of effective DF in FPGA, intended for decreasing logic resources, based on optimization of DF coefficients is observed in given paper.

Keywords: Digital filter, CAD, FPGA, Optimization.

1. INTRODUCTION

Field programmable gate arrays (FPGA) become recently more and more popular hardware basis for digital signal processing (DSP) devices. Digital filters (DF) are the basic devices of many DSP systems. For design of DF for the systems implemented on the FPGA developer must obtain required characteristics with a minimum number of used logical cells. Successful solution of this task results effective DF implementation, and unsuccessful - to irrational use of logical and routing resources of chip, to increasing of power consumption, to decreasing of performance and rising of cost.

The purpose of given paper – analyze of existing design flows of DF in FPGA and overview of design flow, implemented in CAD subsystem, intended for design of effective DF.

2. ANALYSIS OF EXISTING DESIGN FLOWS OF DF IN FPGA

The result of DF design is software or hardware DF realization, which meet all initial requirements. CAD manufacturers offer two typical design flow of DF in FPGA (fig. 1).

On the basis of initial requirements designer chooses type of the filter (recursive or nonrecursive), and defines requirements to the gain-frequency function. For example, for nonrecursive low-pass filters (LPF) the most popular requirements for gain-frequency functions are:

- boundary frequency of a passband;
- passband ripple;
- boundary frequency of a stopband;
- stopband gain;

The key moment is that *any decision if it corresponds to initial requirements in passband and stopband is satisfactory* [1]. Requirements to DF gain-frequency function in system level CAD are used for calculation of DF coefficients (samples of impulse response). Various synthesis algorithms of DF which details of realization, as a rule, are hidden in system level CAD software are applied for this purpose. Also in

system level CAD there are opportunities for quantization of DF coefficients, i.e. transition from float point representation to fixed point format.

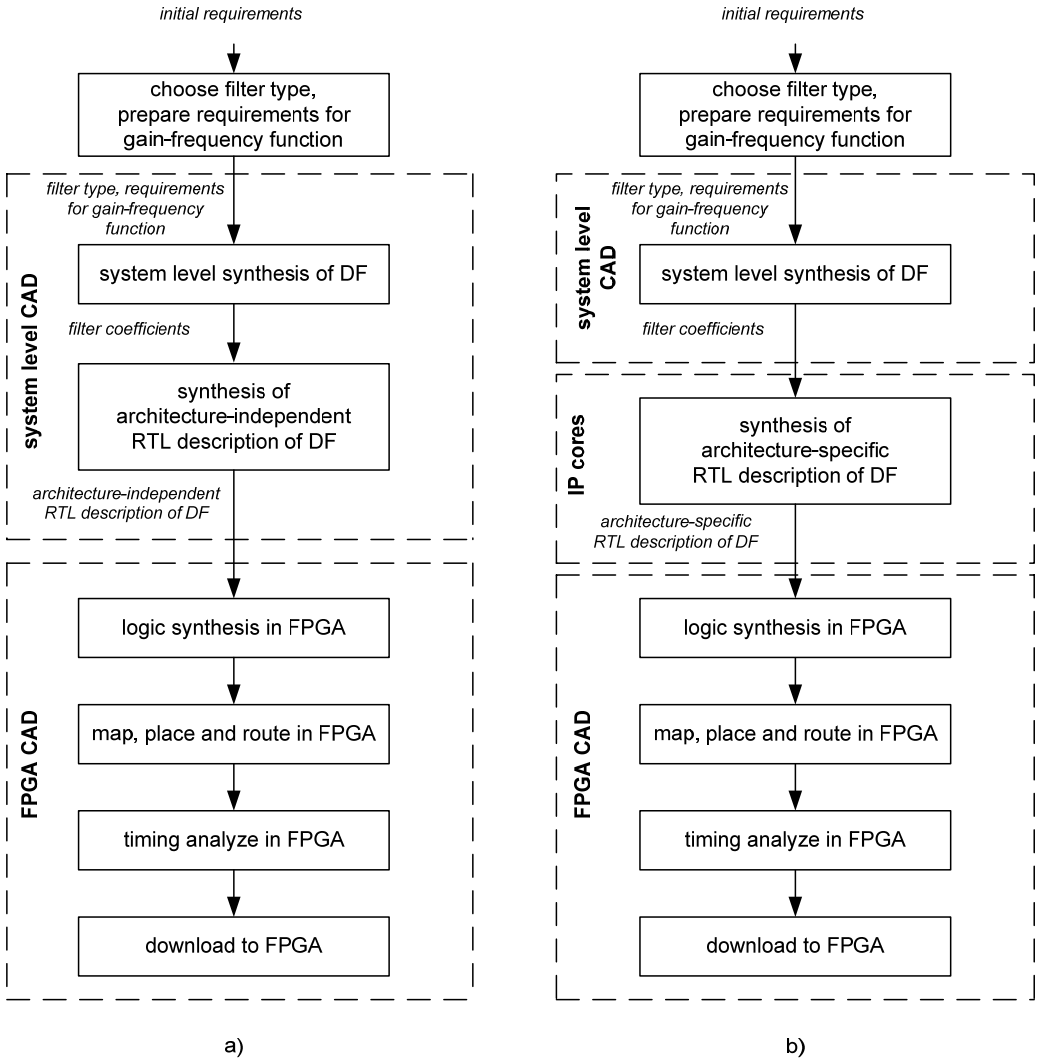

Fig. 1. Typical design flows of DF in FPGA.

The following stage - synthesis of the DF description at register transfer level (RTL) - can be solved in the various ways. Developers of many modern system level CADs supplement the systems with modules of DF HDL description generation (fig. 1a). For example it is Matlab with toolbox Filter Design HDL Coder, SystemView with module HDL Design Studio, Advanced Design System with module HDL Models and Code Generation, etc. For all systems independence of generated DF HDL description from specific FPGA architecture, and consequently low optimality of implementation is typical. Using of IP cores is other popular approach (fig. 1b). IP cores are the complete components allowing easily including of them in user project for creation of more complex system. For creation of DF with IP cores, designer sets DF coefficients, chooses structure and concurrency of hardware implementation of the DF. Existing IP cores allow generating DF HDL description, optimized for specific FPGA architecture. Advantage of IP cores is essentially higher optimality of

using of logic and routing resources of FPGA. As example of IP cores is Parallel FIR filter of CORE Generator from Xilinx and FIR Compiler from ALTERA.

Logic synthesis it is translation of DF HDL description to netlist and optimization of netlist for specific FPGA architecture. Many FPGA CAD systems contain logic synthesis module (for example, XST from Xilinx, Quartus from Altera), and also allow to use of third party synthesis subsystems (such as Leonardo Spectrum from Mentor Graphics).

Map, place and route in FPGA are carried out in FPGA CAD systems in automatic mode. At these stages usually user sets requirements and constraints only.

The basic feature of design flows of DF in FPGA is *separate and independent usage of various CAD systems at different stages, which reduce optimality of final implementation of DF in FPGA*. DF design flows, implemented in modern CAD systems have some disadvantages:

- independent usage of system level CAD and FPGA CAD;
- initial requirements for DF's gain-frequency function not considered during design on register transfer level;
- DF's optimization criterion, used during system level design, not referenced to characteristics of hardware implementation in FPGA.

Presence of these disadvantages essentially worsens such characteristics of DF hardware implementation in FPGA as quantity of used logic resources and maximum clock frequency. The offered DF design flow, implemented in CAD subsystem for effective DF in FPGA has not these disadvantages and allows to improve parameters of DF implementation in FPGA.

3. DESIGN FLOW OF EFFECTIVE DF IN FPGA

Design flow of effective DF, implemented in offered CAD subsystem is shown on fig.2.

After choice of filter type and definition of gain-frequency function requirements, system level synthesis of DF is carried out. Result of synthesis is the vector of DF coefficients.

After system level synthesis, optimization of DF coefficients is performed. DF coefficients are used in constant multipliers which basically define the logic resource necessary for filter implementation in FPGA. The main criterion, defining quality of constant multiplier implementation in FPGA is quantity of necessary look-up tables (LUT) [2]. For estimation of LUT quantity, specific FPGA library is used. During optimization constraints for gain-frequency function are considered, that guarantees, that the resulting decision will meet all initial requirements.

The given optimization method based on a DF coefficient (samples of impulse response) variation for minimization of used FPGA logical resource. Variation of filter coefficients provides its gain-frequency function variation, therefore limits for pass band ripple and stop band gain are used during optimization. It guarantees, that obtained after optimization DF will meet the initial gain requirements.

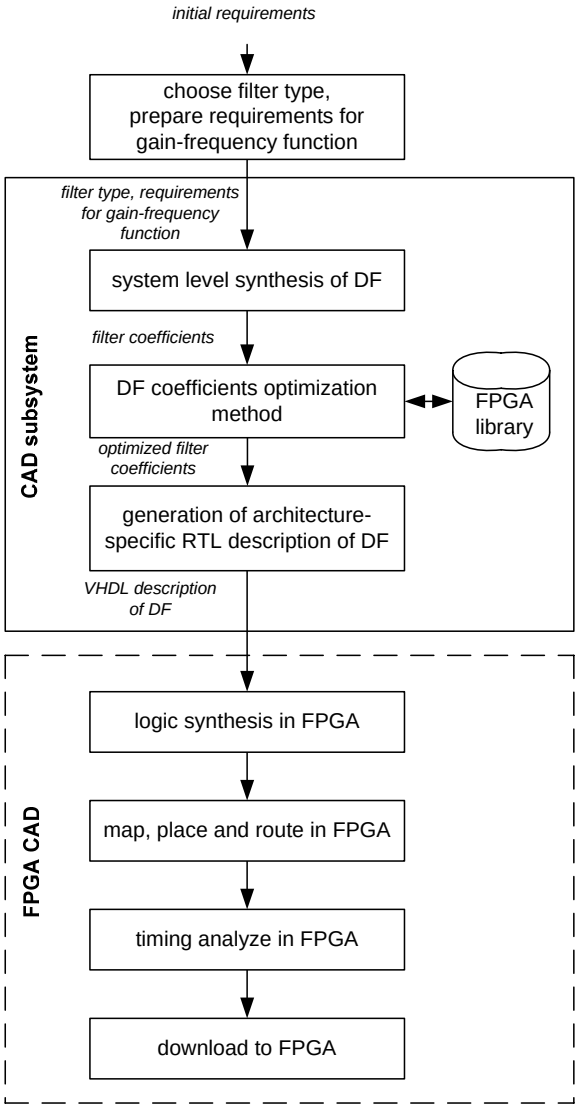

Fig. 2. Design flow of effective DF in FPGA

The big dimension of the optimization task (quantity of coefficients of the standard digital filter is 10-50) and a bad determinacy of the minimized function does not allow to use traditional optimization methods, based on derivatives. Therefore evolutionary algorithms have been selected. Evolutionary algorithms are a simple heuristic method for optimization of multivariate, badly defined functions [3]. Given method uses genetic algorithms - part of evolutionary algorithms.

In the given context of genetic algorithms application each individual represents DF and is characterized by the chromosome - sequence of genes. Each gene of a chromosome is DF coefficient, i.e. the chromosome contains the complete information about a DF impulse response. Goal function defines for each individual and designates quantity of FPGA logical cells, necessary for DF implementation.

As initial value of goal function was taken quantity of LUT, necessary for implementation of DF's constant multipliers with coefficients obtained after system level synthesis. From each generation the individual with minimum value of the goal

function was selected, and the given value was considered as the best for current generation.

Let us consider application of the given optimization method for design of DECT quadrature DF. These filters are necessary for construction of DECT digital receivers which implementation features are overviewed in [4]. The filter with finite impulse response, which has such advantages as absolute stability, linearity of phase-frequency function and simplicity of software or hardware implementations has been selected for realization. After system level synthesis of DF quantity of coefficients was 44. These coefficients were quantized for a 12-bit word length.

Value of the best individual goal function is shown depending on number of generation in fig. 3.

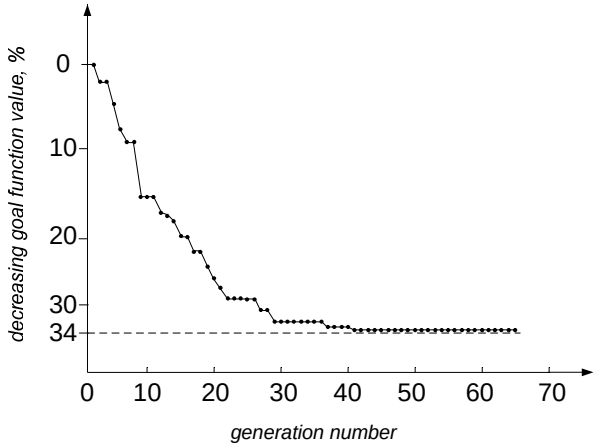

Fig. 3. Dependence value of best individual goal function from number of generation

For resulting DF, LUT quantity, necessary for constant coefficient multipliers, was less on 34 % in comparison with initial solution. Usually after optimization this value decreases on 25-45 %.

Each generation consists from some hundreds of individuals and simulation of one generation takes approximately 1-2 second. Therefore total optimization time not exceed 2 minute for 65 generations.

Impulse response and gain-frequency function of the initial and resulting filters is shown on fig. 4.

Initial DF characteristics are shown by a dashed line, and for DF after optimization – by continuous. Two impulse responses practically merge and it is possible to see a difference only at magnifying of a fragment. Meanwhile, such insignificant difference can decrease logic resource more than 30%. For gain-frequency function differences are more appreciable, but it is necessary to pay attention, that the filter obtained after optimization, as well as initial, completely meets initial requirements. Phase-frequency function of both filters remains linear since the symmetry of impulse response after optimization is saved.

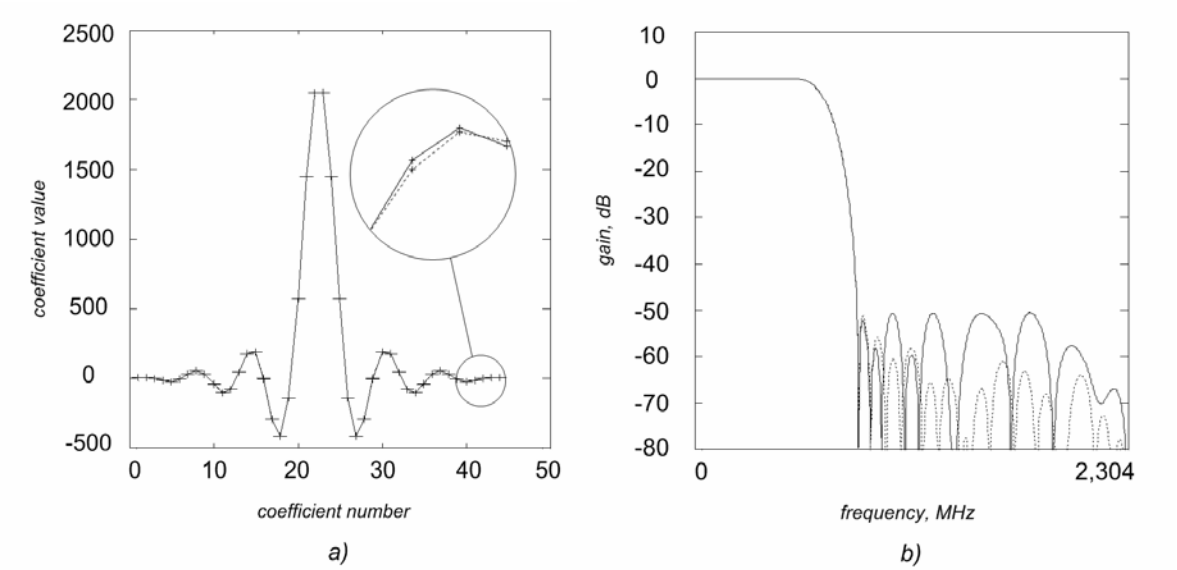

Fig. 4. Impulse response (a) and gain-frequency function (b).

Optimized DF coefficients are used for generation of architecture-specific RTL description of DF. During generation a synthesized VHDL description of DF is created. Logic synthesis, map, place and route are performed in FPGA CAD, such as ISE 6.3 from Xilinx.

Main advantages of an offered design flow of effective DF in FPGA:

- During optimization both initial gain function requirements and quantity of FPGA logic resources are considered.
- Criterion of optimization is LUT quantity, which objectively indicates an optimality of the obtained DF implementation in the FPGA.

4. CONCLUSION

In given article analyze of design flows of DF, implemented in modern CAD systems is considered. Design flow of effective DF in FPGA, intended for decreasing logic resources, based on optimization of DF coefficients is observed in given paper. This design flow, implemented in CAD subsystem, can decrease logic resource, necessary for DF's constant multiplier implementation on 25-45%. At the same time DF, obtained after optimization, completely met all initial requirements to gain-frequency function.

5. REFERENCES

[1] Foundations of digital signal processing: Lections / Authors: A.I.Solonina, D.A.Ulahovich, C.M.Arbuzov, E.B.Solov'eva / Publ. 2 – SPb.: BHV-Peterburg, 2005. – 768 p.: il. (in Russian)

[2] Virtex-II Platform FPGA Complete Data Sheet, http://www.xilinx.com.

[3] Batichshev D.I. Genetic algorithms for decision of extreme problems. Voronezh, 1995. (in Russian)

[4] Merkutov A.S. Automated design of digital FM receivers / Mathematical method, informational technologies, physic experiments in science and industry, materials of regional science conference - Vladimir: VlGU. – 2003. p. 74 – 75.