

## Research Article

### The Design and Simulation of Embedded FIR Filter based on FPGA and DSP Builder

Zhao Zeng-Rong

Hebei Normal University College of Career Technology, Hebei, Shijiazhuang 050024, China

**Abstract:** The aim of this study is to introduce a new way to design an embedded FIR Filter whose parameters can be adjusted easily to meet different need. FIR Filter plays an important role in the digital signal processing which can implement the function such as low pass filter, pass band selection and etc. A 37 steps low pass FIR filter is designed and simulated in DSP Builder and MATLAB by Filter IP Core which can be converted into VHDL file to be used in Quartus II and FPGA device as a embedded model quickly and easily. The simulation result shows the FIR Filter meets the requirement of parameters.

**Keywords:** DSP builder, embedded FIR filter, FPGA Quartus II

#### INTRODUCTION

In the procedure of digital signal processing, signal comes from the data acquisition card which would be mixed with noise, disturbance and clutter wave signals, so the vital signal must be extracted to be extracted to be used in the next circuit. FIR Filter plays an important role in digital signal processing which could be used to as the Low Pass Filter, Pass Band Selection, Anti Aliasing, Extraction and Interpolation and etc. In digital signal processing, traditional filter is realized by high speed multiplication accumulator where only limited operation can be done during the next sampling period, so the bandwidth is limited and the speed is low. With the development of technology, digital filter can be designed and simulated through the simulation software MATLAB. Digital filter has many advantages such as high precision, high speed, excellent cutoff characteristic and so on. And the digital filter can be embedded in Single chip microcomputer or FPGA. And the structure of FPGA is the ideal carrier of processing digital signal by improving sample speed, so the performance of the whole system is improved. And Now FIR Filter automatic generation tool based on FPGA has been published, where MATLAB, DSP Builder and FPGA are combined and the filter can be designed, simulated and transformed into VHDL language program to operate in FPGA. In this study, a simple way is gave to show how to design FIR Filter with high speed and performance quickly by DSP Builder and FPGA whose parameters are automatically realized by the Filter Compiler and its simulation result meets the need.

#### FIR FILTER WORKING PRINCIPLE

FIR means finite impulse response, its system formula is:

$$H(z) = \sum_{k=0}^M b_k z^{-k} \quad (1)$$

where,

M = The zero count-delay section which is called order number of FIR Filter.

FIR's basic formula is:

$$y(n) = \sum_{i=0}^{L-1} x(n-i)h_i \quad (2)$$

where, x(n) is the input sampling sequence, h(i) is the coefficient of FIR Filter, L is the coefficient length, y(n) is the output sequence of FIR Filter. IF all the coefficient h(i) is known, FIR Filter is finished. So how to determine these coefficients are very important, but it is too complex to a new hand to design a suitable FIR Filter and it will be more difficult with the order number's increasing by traditional design method. Now with the development of FPGA and EDA tools, it is possible to design FIR Filter simply by DSP Builder and FIR IP Core provided by Altera. DSP Builder is a system tool face to DSP development supported by Altera Company. It is used as a Simulink Toolbox of MATLAB which can design DSP system such as FIR and FFT in a graphical interface by calling related models in the DSP Builder Toolbox. Here we adopt FIR Compiler to resolve these problems. Fir Compiler is a fir Filter Core provided by Altera Company which is combined with DSP Builder closely. DSP Builder

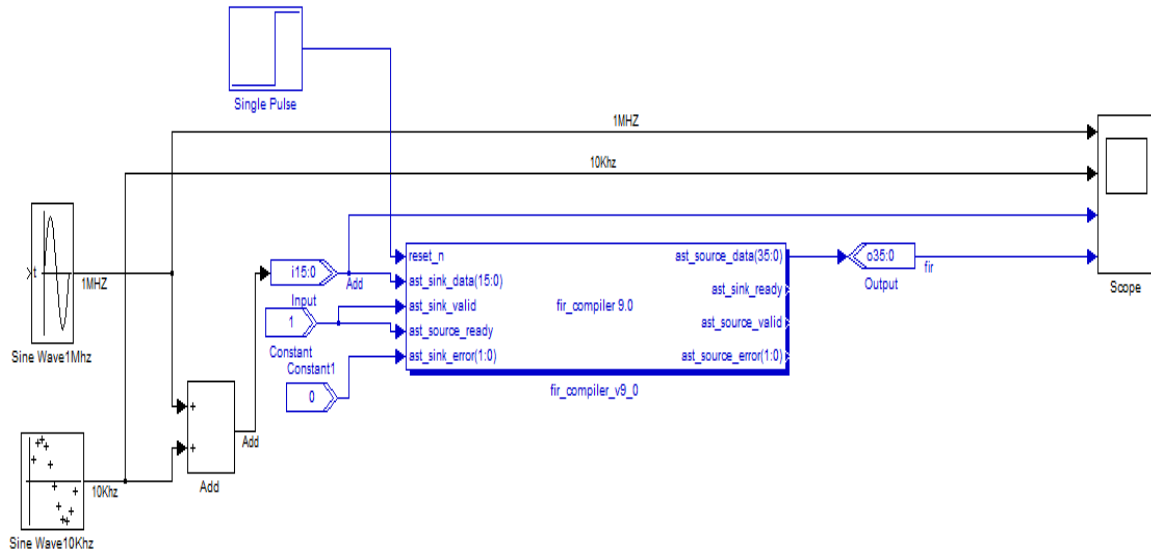


Fig. 1: FIR filter based on DSP builder

provides the application and simulation environment of FIR Core (Pan and Huang, 2009).

### FIR FILTER BASED ON DSP BUILDER

Before the FIR Core is used, all the software MATLAB, DSP Build, Quartus and IP Core must be installed correctly, so we can call the model fir-compiler-90 from the library of Altera DSP Builder Blocket/MegaCore Functions to compose the FIR Filter as shown in Fig. 1 (Pan and Huang, 2003).

In the Fig. 1, the FIR Filter includes 5 inputs; they are reset-n, ast-sink-data, ast-sink-valid, ast-source-ready and ast-sink-error. In which, the ast-sink-data is input wave, here it is the superimposed signal of two sine wave, other signals come from the step pulse, constant 0 and constant 1. The scope is used to show the wave of sine wave 10 Mhz, sine wave 100khz, superimposed signal ADD and the signal fir after filter.

**FIR Compiler's function:** Altera FIR Compiler is a tool to custom the Finite Impulse Response (FIR) filter, its main functions are as following:  
It supports Avalon Streaming interface;

It supports four different system structures, they are full parallel distributed arithmetic, Full serial distributed arithmetic and Multibit serial distributed arithmetic and Multicycle variable structure. Here we adopt Multicycle variable structure which can make the filter to calculate the output result by multiple clocks.

It supports single rate, rising sample and falling sample.

It can optimize design to reduce logic cell's number by using Memory blocks such as M512, M4k, M9k, MLAB and DSP block (Zhi-Gang, 2007).



Fig. 2: FIR compiler

As we known, FIR Filter's parameters are balanced and symmetric, so its design could be optimized according to this characteristic. And there will be an extra clock cell to calculate after its optimization.

**FIR Filter's Parameter is set up as following:** After the Filter Compiler is adopted from the library of DSP Builder/Mega Core, we double click it to open the Mega Core of FIR Compiler as show in Fig. 2. There are two steps needed to finish FIR Filter design, step1 is responsible for parameterize Filter and step2 is responsible for filter's generation and changing Filter's appearance into a model with input and output port. Then click the step 1 in Fig. 2 to open the parameters setting up window. Filter Compiler's parameters need to be set in different window. Here, we need two windows as in Fig. 3 and 4 (Qing *et al.*, 2008).

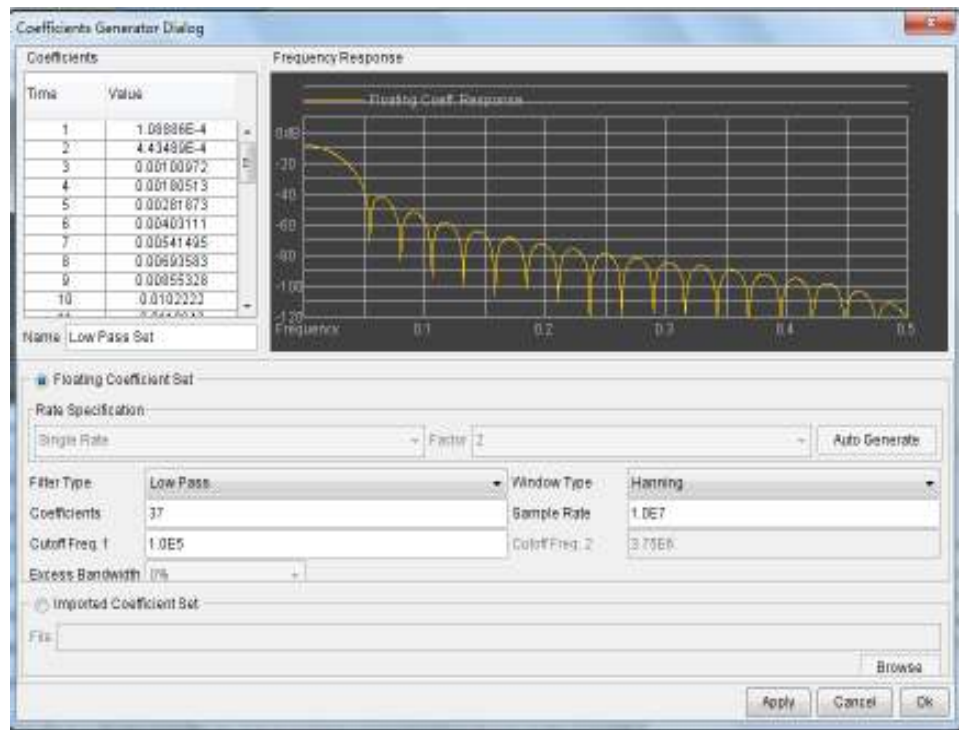


Fig. 3: Coefficients generator dialog window

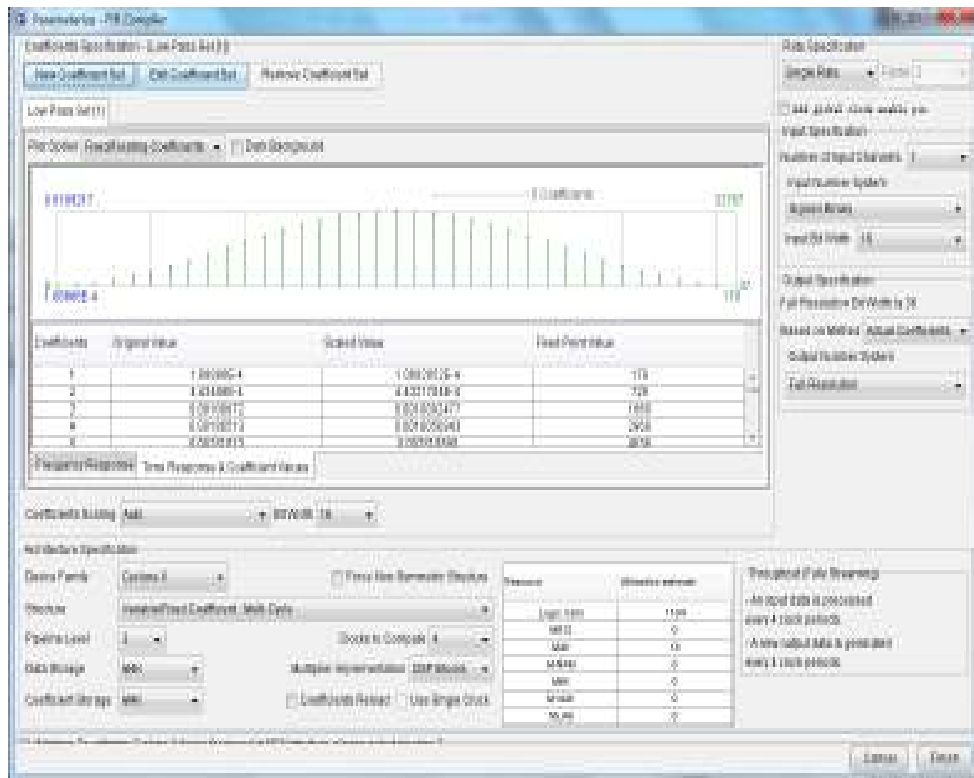


Fig. 4: Coefficients specification window

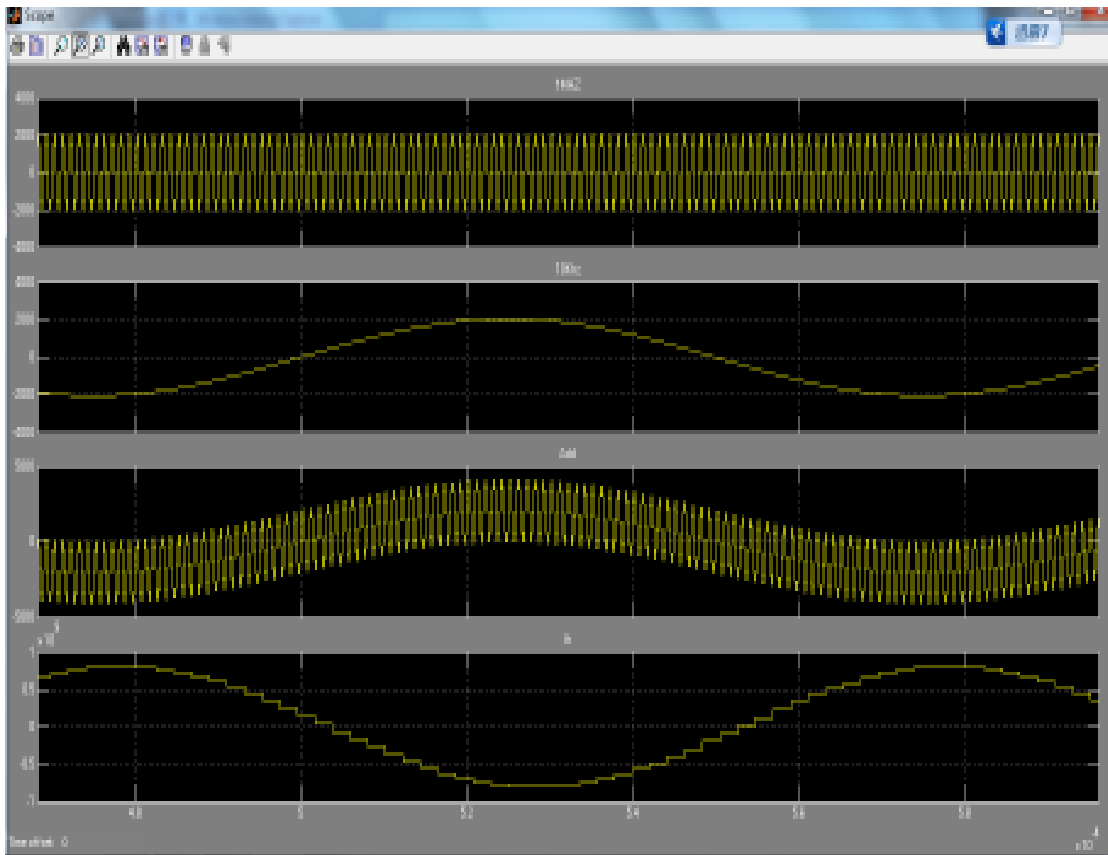


Fig. 5: Simulation result

- **Coefficients denerator dialog:** Under the window of Coefficients Generator Dialog as Fig. 3, parameters are set up as following: Filter type is Low pass, Coefficients is 37, Window type is Hanning, Sample rate is 10 MHz, Cut off frequency is 100 khz. Then click the button “Apply”, the frequency response curve is shown as in Fig. 4. Click the button “OK” to return.
- **Coefficients specification:** Under the window of Coefficients Specification as Fig. 4: Bit width is 16, Pipeline level is 3, Structure is Multi-cycle which can compromise between area and rate by adjusting calculation time to consume less LEs. The time response&coefficients curve are shown as in the figure and we can see there are 13 M4K and 1134 logic cells used in the FPGA device.

After all these parameters are set up, click the button “ok” to return to the Fig. 2 and click step 2 to create a required filter. The filter model is changed into a new image with input ports and output ports as shown in the Fig. 1.

**Other models’ parameters:** Two sine waves are adopted from the library of Simulink/Source, their

sample time are  $125 \times 10^{-9}$  seconds, samples per period are 8 and 800, so their frequency are 1Mhz and 10khz differently.

The add model is adopted from the library of Simulink/Math operation; here its parameters are saved as default.

The single pulse model is adopted from the library of Altera DSP Builder Block/Gate and Control, its delay is 50 which are larger than filter’s order 37.

Constant 1 and constant 0 come from the library of Altera DSP Builder Block/IO and Bus.

Input and output ports come from the library of Altera DSP Builder Block/IO and Bus, their width are 16 and 36 differently.

## SIMULATION RESULTS

After the FIR Filter has been designed as in the Fig. 1 and all the parameters are set up, a clock model should be played on the figure to make sure all the model work at the same frequency and the Signal Compiler model ensures the DSP Builder work correctly. To simulate the FIR Filter, Firstly click the command Simulation/configuration parameters to set the related parameters as following: start time is 0

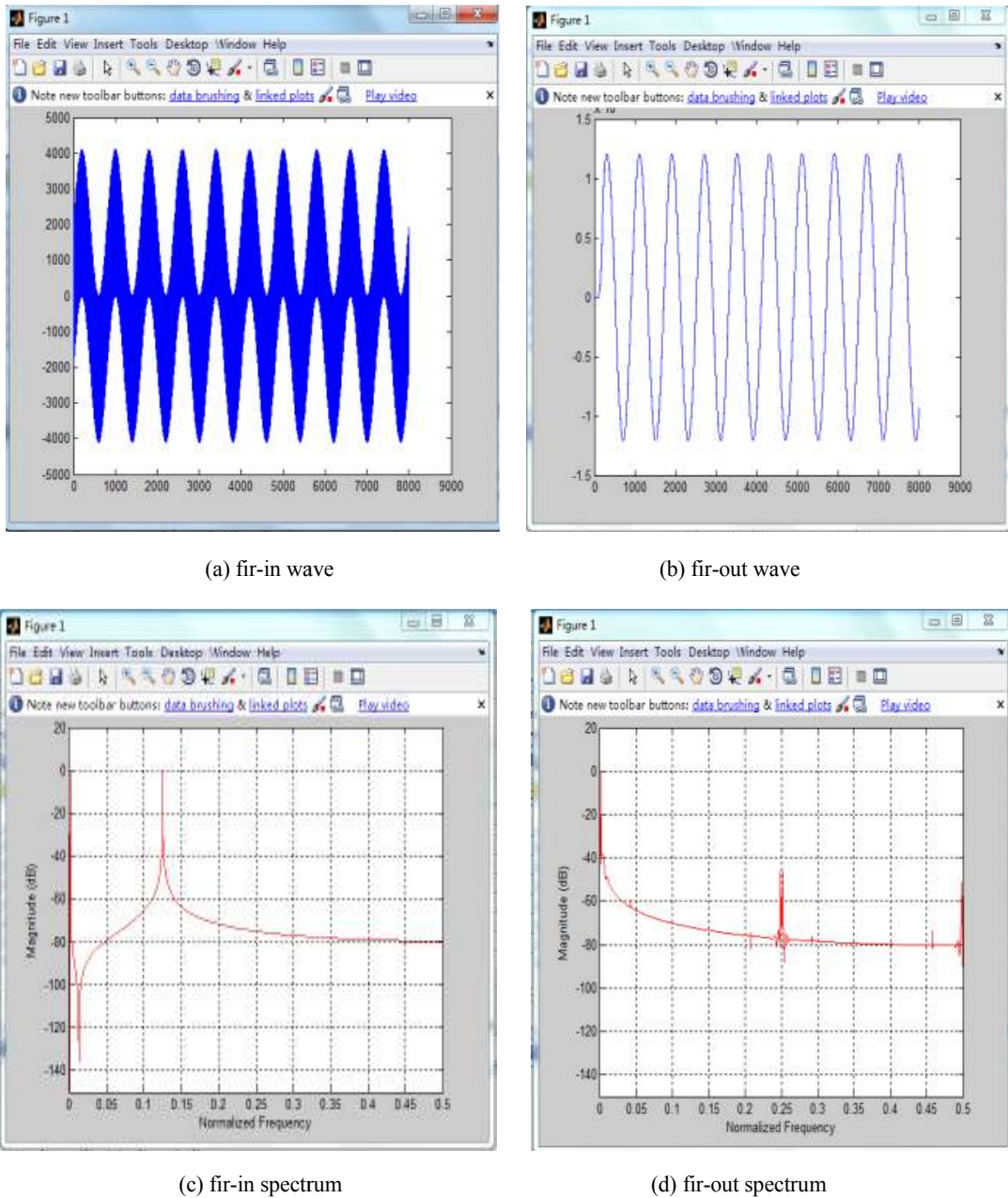


Fig. 6: Data analysis result in the workspace of MATLAB

second, end time is  $1e-4$ , type is Fixed-step and solver is discrete (no continuous status). Secondly, click Simulation/Start to do simulation. When the simulation is over, double click the Scope to open simulation curves to observe simulation result as in Fig. 5.

There are four curves in the Fig. 5: signal 1 MHz, signal 10 kHz, signal ADD and signal fir. The simulation result shows that the high frequency signal 1 MHz is filtered from the add signal after FIR Filter and

it is delayed than input signal add, so the FIR Filter designed by the FIR Compiler is effective and correct. To show the result clearly, we set the signal filter-in array and filter-out array to the MATLAB Workspace and plot their spectrum curves by commands plot and fftplot; their curves are shown as Fig. 6 a, b, c, d (Zi-Ting and Hai-Yang, 2008). It shows in the figure that the signal 1 MHz is attenuated down to 60 dB after the FIR Filter. It also proves that the FIR Filter designed

works well in another view. After the simulation result is correct, it is easy to convert the FIR Filter in simulink to VHDL file which can be used in QUARTUS II by clicking model- Signal Compiler. So a 37 steps, low pass FIR Filter is finished, you can download it into the FPGA device appointed in the Compiler to work in hardware environment.

### **CONCLUSION**

From the process of designing FIR Filter, it shows that adopting DSP Builder to design filter in the system level improve the traditional design method in circuit level largely, it can do a system design based on the hardware design language, simulate and analyze by the Simulink tool of MATLAB and even plot related curve in MATLAB, at last the design model is converted into VHDL file to realize specific hardware circuit by QUARTUS II. During this course, you needn't know clearly about the filter's inner structure and hardware structure, so it is a system design in fact and the design efficiency increases in a large scale. If another filter is needed, it is simple to set filter parameters in step 2 to change the filter type and related parameters in Fig. 3 and Fig. 4 to meet your need.

### **ACKNOWLEDGMENT**

This study is supported by Hebei science and technology project (11213501D) and Hebei Normal University Campus Fund (L2010Y14).

### **REFERENCES**

- Pan, S. and J. Huang, 2003. Modern DSP Techniques. Publishing House of Xi'an Electronic and Science University, Xian.
- Pan, S. and J. Huang, 2009. EDA Technology and VHDL. 3rd Edn., Publishing House of Tsinghua University, Beijing.
- Qing, L., X. Wei and L. Yan, 2008. Design and realization of the lattice FIR filter using DSP builder. *J. Northwest University Natural Sci.*, 38(5): 727-730.
- Zhi-Gang, Z., 2007. FPGA and SOPC Technology-DE2 Practice. Publishing House of Xi'an Electronic and Science University, Xian.
- Zi-Ting, W. and G. Hai-Yang, 2008. Digital filter IP core's design and realization based on matlab/simulink. *J. Lan Zhou JiaoTong University*, 27(3): 111-113.