On the design of an OFDM Transmission System with Rapid Prototyping Tools

Guilherme Paulino, Rangel Arthur
University of Campinas, School of Technology, Limeira-SP, Brazil

g117119@dac.unicamp.br, rangal@ft.unicamp.br

Abstract— This paper performs the development of a complete system of digital multicarrier modulation using the DSP Builder tool for FPGA devices. The OFDM modulation was chosen by its wide application in telecommunications systems such as mobile communications and digital television; and may allow the transmission of high speed data. This design proposed divided the system into the following modules: mapping data generating pseudo-random sequences, the symbols creation and applying the inverse Fast Fourier Transform (IFFT).

Keywords—OFDM; DSP Builder; FPGA devices

I. INTRODUCTION

The concept of using parallel data transmission and Frequency-Division Multiplexing (FDM) was proposed in the 1960s [1, 2]. In a parallel classical system of data, the frequency band is divided into N subchannels without overlapping by multiplexing. Each subchannel is modulated with a separate symbol. This eliminates the interchannel interference (ICI). With the mathematical principle of orthogonality, can be increased the efficiency of spectrum usage through a multicarrier modulation technique with overlapping. It also avoids the use of equalization data at high rates, dispersive noise and multipath distortion [3].

In 1971, Weinstein and Ebert applied the discrete Fourier transform (DFT) to parallel data transmission systems as part of a modulation and demodulation process [4]. By applying DFT on the receiver and calculating the correlation values with the frequency of the center of each subcarrier, the data transmitted can be recovered without intercarrier interference. Also, the FDM multiplexing is performed not by low-pass filters but in baseband processing using DFT multicarrier techniques [3].

The implementation of a digital system might be obtained with the use of fast Fourier transform algorithm (FFT) to replace the DFT processing required. These algorithms are more efficient and eliminate oscillator's banks to the subcarriers and for coherent demodulators. Using this method significantly decreases the system complexity and reduces the number of $N^2$ DFT operations to $N \log(N)$ operations in FFT [4].

The Orthogonal Frequency-Division Multiplexing (OFDM) in a transmitter includes the inverse fast Fourier transform (IFFT) operation and also the insertion of cyclic prefix (CP). In an OFDM receiver, the cyclic prefix is removed before the data packet is submitted to the fast Fourier transform (FFT) for demodulation, as shown in Figure 1.

II. RAPID PROTOTYPING TOOLS

For this project the DSP Builder tool was used, which runs on Simulink environment and allows the modeling, simulation and implementation to make tests and measurements on FPGA-based devices. These devices are the state of the art for the industry and R&D in digital processing and transmission.

A. FPGA Device

The Terasic DE1 Development and Education Board was used, which has the FPGA chip Altera Cyclone II model EP2C20F484 with approximately 20,000 logic elements, in addition to peripheral devices that are 8MB SDRAM, 512KB SRAM, 4MB of Flash memory, audio and video interfaces, RS-232, SD card slot, and more.

B. MathWorks Simulink

Simulink allows the modeling of systems through diagrams and offers a complete blockset, for example, to create interactive testbенches, to generate input signals, to plot graphs, to analyze values, etc., and also supports intellectual property (IP) blocks from Altera devices through DSP Builder blockset. The version R2013b was chosen to be compatible with DSP Builder v13.0 and to support Cyclone II device family.
C. **Altera DSP Builder**

DSP Builder is a product for performing Model-Based Design targeting Altera FPGAs. It consists of an advanced and a standard blockset that allows high-performance hardware description generation. Also, it links Simulink to Altera Quartus II implementation environment.

The use of blocks modeling facilitates integration of complex DSP functions and supports automatic generation of VHDL testbenches.

III. **RESULTS**

This project proposes to modularize the system being divided into pseudorandom data sequence generation, mapping, symbol creation and the application of inverse fast Fourier transform (IFFT) using IP blocks Altera.

A. **Pseudorandom Generator**

The pseudorandom input data are generated by a Simulink block called Random Integer Generator, that generates random integers uniformly distributed in an interval of a M-ary number. As the 64-QAM constellation was the digital modulation chosen to generate the input data, the sequence generator block is set to generate integers of maximum amplitude 64. That is, the M-ary number is equal to 64 and the generated numbers have 6 bits per symbol. For this reason, in the OFDM transmitter input interface takes a 6-bits sampling. From this step to the next ones are used DSP Builder IP blocks for modeling.

The assigned initial seed was 37, chosen randomly.

After generation of the random integer values, the numbers are randomized and then quadrature modulation is applied, as shown in the Figure 2.

B. **Randomizer**

The subsystem randomizer receives the pseudo random integer values, after scanning, and creates a vector to be applied to constellation modulation, paralleling the sequence of their inputs. This is based on applying an algorithm of linear feedback shift register (LFSR) and XOR logic ports as shown in Figure 3.

C. **64-QAM Quadrature Modulation**

The quadrature amplitude modulation of 64 symbols is based on the principle which the input vector is split into two signals to be transmitted in parallel, one being the real part and the other one imaginary (see Table 1). The implementation of this algorithm was done using Look-Up Tables as shown in Figure 4.

| Table 1. Analogy Between Mapping 64-QAM Symbols. |
|---------------------------------|---------------------------------|
| **Real Part** | **Imaginary Part** |
| -7 | 000b | -1 | 010b | +7 | 000b | +1 | 010b |
| +7 | 100b | +1 | 110b | -7 | 100b | -1 | 110b |
| -5 | 001b | -3 | 011b | +5 | 001b | +3 | 011b |
| +5 | 101b | +3 | 111b | -5 | 101b | -3 | 111b |

D. **Inverse Fast Fourier Transform**

Figures 5 shows the implementation of IFFT algorithm with 2048 points and radix-2^2. Figure 6 includes the BitReverseCoreC IP block that converts the flow of input data in natural order to a reverse bit sequence, and an IFFT block set up to receive its input bits in reverse sequence and return the output to natural order.

The BitReverseCoreC block performs storage and bit inversion for the IFFT frames. A synthesis with a single time
parameter specifies the $N$ length of the transform. Since the transform length $N = 11$, which results in $2^N = 2048$ points.

The data input of the IFFT block must be based on flow control. That is, for a simple IFFT interaction (with only one block) must be synchronized in consecutive clock cycle and must be a gap between these data blocks. The BitReverseCoreC IP block produces data in blocks on its output, respecting this protocol.

In Figure 7 shows the scope signals output from the scheme in Figure 5, where the valid signal ($x_{\text{valid}}$) is a validation input of the logical type and its value is always 1 during the processing. The channel signal ($x_{\text{channel}}$) is a channel input of the data type unsigned integer with 8-bits precision and receives the counting of the current data block, considering 2048 points per block. The $d$ input of IFFT Core should receive the $x_r$ input in reverse order sequence, that corresponds to the real and the imaginary parts. The $y$ signals are the corresponding ones after IFFT processing.

IV. CONCLUSION

By using a reconfigurable hardware device is possible to implement a multicarrier system for digital transmission, and test and adapt its algorithm behavior for optimization of the parameters using rapid prototyping tools.

The challenge was trying to understand how to develop this complex system of transmission, even limited to simulations. DSP Builder can interface Simulink and FPGA in an easy way. However, the theory applied is the same to understand how industrial systems may be developed using OFDM technology.

ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support given by the agencies CNPq and FUNTTEL.

REFERENCES


Fig. 6. DSP Builder scheme for the Inverse Fast Fourier Transform processing.

Fig. 7. Simulink scope graph plot for the (a) input signals and (b) IFFT output vector.