DESIGN OF DIGITAL MODULATORS WITH MATLAB SUPPORT

R.Marsalek
Department of Radio electronics
Brno University of Technology

1. Introduction

Digital modulations are nowadays widely used, for example in cellular systems or local area networks. In our department, digital modulations are one of the centers of interests for research as well as for education. In this paper, we will briefly discuss a design flow for FPGA modulator implementation (supported by MATLAB software) and describe our experimental setup prepared for simple FPGA modulators verification.

2. Implementation of digital modulators in FPGA's and the role of MATLAB software

Basically, a design of a FPGA modulators with MATLAB support can consist of these steps:

I) Modulator design in Matlab

Basic digital modulators differ mainly in mapping used (QPSK, M-QAM, 8PSK, etc.) and filtering (pulse shaping) of signal. One of the three classical filters - rectangular, cosine, root raised cosine filter is often used. The length of filter impulse response is the compromise between "accuracy" (signal distortion) and implementation complexity. An example of cosine filter impulse response and resulting 8PSK signal constellation is shown in figure1.

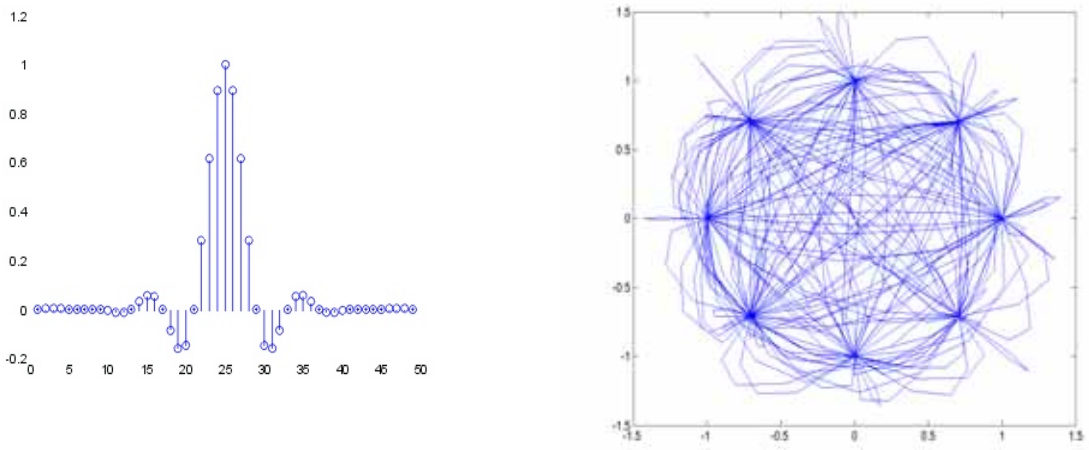

Fig. 1 : FIR filter impulse response (left) and 8PSK constellation (right)

II) VHDL implementation

VHDL components for most types of mapping can be easily written. An efficient implementation of the shaping filter can be generally more complicated task, but a designer can profit from the components included in Xilinx CoreGen library. In our designs, we have successfully used a Distributed arithmetic filter. It can be easily configured according requirements on filter speed or amount of chip resources. CoreGen filters need to define the filter coefficients in .COE file. To make the filter design more flexible, we have written a MATLAB function 'write_to_coe.m' with these inputs – vector of real filter coefficients, filename and length (n) of data representation. Note that this function supposes binary *Qn* data format. The function 'write_to_coe.m' can be split into two parts – calculation of *Qn* data from MATLAB data format and writting of data to .COE file. The example of .COE file generated using our function is shown in figure 2.

III) Simulation

For the simulation of VHDL code, e.g. Modelsim software can be used. The results can be compared with MATLAB simulation for example using MATLAB link for Modelsim or, as in our case, by writing little function for data conversion from Modelsim workspace to MATLAB.

IV) FPGA programming

using standard FPGA utilities, e.g. Foundation etc.

V) Verification, measurements

Next section will be devoted specially to verification of FPGA design.

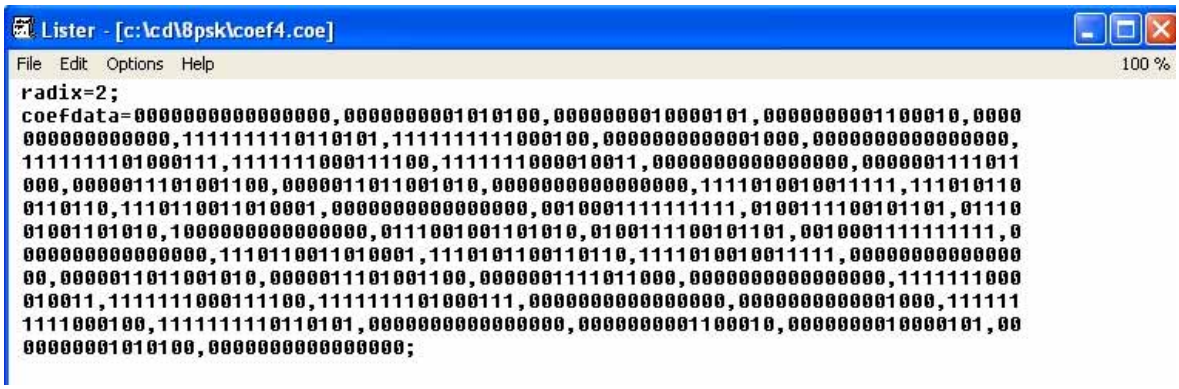

Fig. 2 : COE file generated using our script

3. Design verification

For the verification of simple digital modulators implemented in FPGA devices, we are currently working on the experimental setup shown in figure 3. The main components of the experiment are:

I) FPGA development board

Our laboratory is equipped with MEMEC Virtex-II V2MB1000 development kit populated with 1 million gate Xilinx device XC2V1000. This board includes also some additional components like DDR memory, RS232 port, etc.

II) A/D, D/A development board

consisting of 2 165Ms D/A and 2 53Ms A/D converters. The connections to the FPGA development board is made by P160 interface.

III) Signal generator SMIQ used as the vector modulator

working from 300kHz to 2.2 GHz, fabricated by Rohde-Schwartz. IQ modulator impairments can be also emulated.

IV) Real-time signal analyzer

3086, manufacturer Sony Tektronix 3086. Operating from DC to 3GHz in RF, baseband or IQ mode. Analyzer is equipped with 12,1 inch color display. It can perform downconversion and constellation diagram, eye diagram, RF spectrum can be displayed. Moreover, the quality of modulator can be also evaluated by EVM value, that can be also calculated using this analyzer.

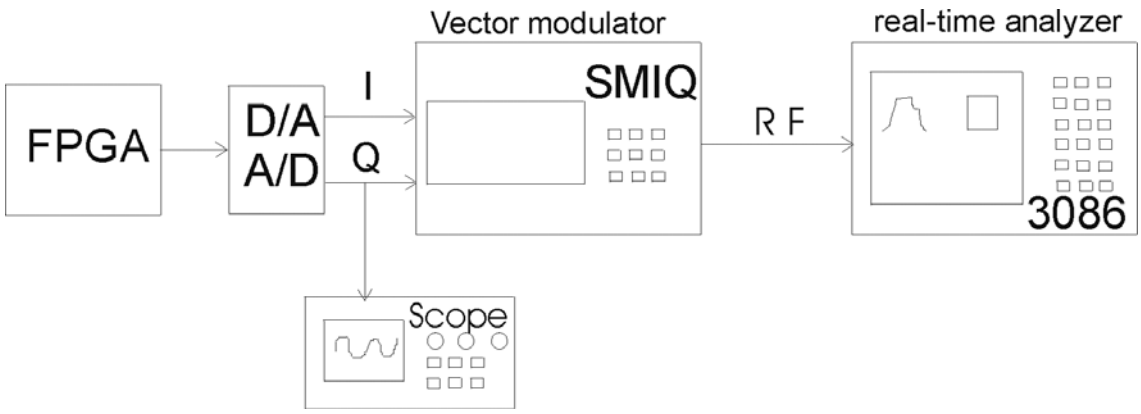

Fig. 3 : Experimental setup for digital modulators verification

4. Conclusions

In this paper, we have briefly discussed possible design flow of simple FPGA digital modulators and the role of MATLAB in this process. Our currently prepared experimental setup for modulator verification has also been presented.

Acknowledgements

This work has been supported by the grant GACR (Czech Science Foundation) No. 102/04/0557 "Development of the digital wireless communication resources" and the grant MSM 262200011.

References

[1] Virtex-II V2MB1000 Development Board User's guide, MEMEC Design, December 2002
[2] MEMEC P160 Analog Module User Guide, MEMEC design, July 2003.

Contact: marsaler@feec.vutbr.cz