

Design and Implementation of Digital Front End Module of MIMO-OFDM on FPGA

Syed Muneeb ur Rehman¹, P. Cyril Prasanna Raj²

¹M.Sc. (Engg.) Student, ²Assistant Professor and Course Manager
VLSI System Design Centre
M.S.Ramaiah School of Advanced Studies, Bangalore 560 054

Abstract

MIMO introduces third dimension space into the traditional frequency-time domain. The idea is to transmit multiple streams of data on multiple antennas at the same frequency, to increase throughput. Typically, multiple receiver antennas are used as well, since this configuration achieves high data rates, multiplied by the number of channels between either ends. This principle is called Multiple Input Multiple Output (MIMO). They are particularly attractive because they do not require any additional transmission bandwidth, and unlike traditional systems use multi-path interference to their benefit.

The present paper focuses on FPGA implementation of a particular multi-antenna scheme, a 2×2 MIMO System, employing Alamouti Technique (Space Time Block coding). The initial simulation was done in MATLAB for the proof of concept and specification was derived for the design before proceeding to hardware design. The number format used in the design is IEEE floating point format. The MIMO communication System Blocks were then designed using Verilog HDL. The modules which were designed are 16 QAM Modulator, MIMO Encoder, Wireless channel model, MIMO Decoder, 16 QAM Demodulator, Floating Point multiplier, adder, complex Multiplier.

This paper has resulted in the development of a hardware prototype of a MIMO Transmitter, Receiver and channel, which works at the speed of 200MHz on a Spartan-3 FPGA board. As the number format was floating point, there was a need to develop a separate function which will show the equivalent real numbers for the corresponding floating point number. This made the task of debugging a lot easier. Test benches for individual model were developed and tested it for its correct functionality. All the modules were to form complete MIMO system. The functional simulation was carried out for the entire design. The entire design was mapped on to FPGA. Transmitter block, Receiver block and Channel block were each tested separately on FPGA. The results were compared with the MATLAB results and were found to be the same. The Tools used are MATLAB, ModelSim, and Active-HDL and Xilinx.

Keywords: MIMO, Matlab, FPGA, Alamouti, Digital Front End

1. INTRODUCTION

Radio transmissions traditionally use one antenna at the transmitter and one antenna at the receiver. This system is termed **Single Input Single Output (SISO)**. Both the transmitter and the receiver have one RF chain (that's coder and modulator). SISO is relatively simple and cheap to implement and it has been used age long since the birth of radio technology. It is used in radio and TV broadcast and our personal wireless technologies (e.g. Wi-Fi and Bluetooth). Figure 1.1 shows SISO Model. One antenna at both the transmitter and the receiver employs no diversity technique.



Fig. 1.1 Single Input Single Output (SISO)

To improve performance, a multiple antenna technique has been developed. A system which uses a single antenna at the transmitter and multiple antennas at the receiver are named **Single Input Multiple Output (SIMO)**. The receiver can either choose the best antenna to receive a stronger signal or combine signals from all antennas in such a way that maximizes SNR (Signal to

Noise Ratio). The first technique is known as switched diversity or selection diversity. The latter is known as maximal ratio combining (MRC). Figure 1.2 shows the SIMO Model, One antenna at the transmitter, two antennas the receiver. Employs a receive diversity technique

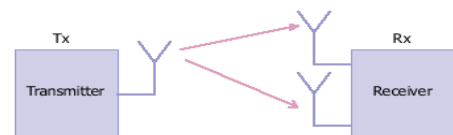


Fig. 1.2 Single Input Multiple Output (SIMO)

A system which uses multiple antennas at the transmitter and a single antenna at the receiver is named as Multiple Input Single Output (MISO). A technique known as **Alamouti STC (Space Time Coding)** is employed at the transmitter with two antennas. STC allows the transmitter to transmit signals (information) both in time and space, meaning the information is transmitted by two antennas at two different times consecutively.

Multiple antennas (each with an RF chain) of either SIMO or MISO are usually placed at a base station (BS). This way, the cost of providing either a receive

diversity (in SIMO) or transmit diversity (in MISO) can be shared by all subscriber stations (SSs) served by the BS. Two antennas at the transmitter, one antenna at the receiver, Employs a transmit diversity. Figure 1.3 shows the MISO model.

To enhance throughput of a radio link, multiple antennas (and multiple RF chains accordingly) are put at both the transmitter and the receiver. This system is termed Multiple Input Multiple Output (MIMO). A MIMO system with similar count of antennas at both the transmitter and the receiver in a point-to-point (PTP) link is able to multiply the system throughput linearly with every additional antenna. For example, a 2x2 MIMO will double the throughput.

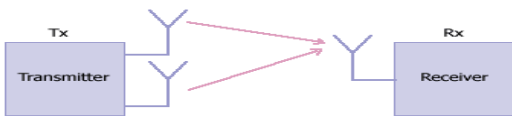


Fig. 1.3 Multiple Input Single Output (MISO)

A MIMO system takes advantage of the spatial diversity that is obtained by spatially separated antennas in a dense multi path scattering environment. MIMO systems may be implemented in a number of different ways to obtain either a diversity gain to combat signal fading, or to obtain a capacity gain.

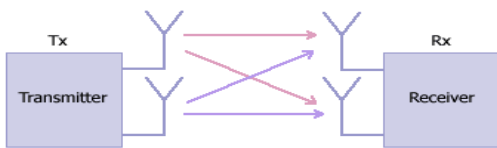


Fig. 1.4 Multiple Inputs & Outputs, MIMO, 2x2

Generally there are three categories of MIMO Techniques. The first aims to improve the power efficiency by maximizing spatial diversity. Such techniques include delay diversity, space-time block codes (STBC) [4], [5] and space-time trellis codes (STTC) [6]. The second class uses a layered approach to increase capacity. One popular example of such a system is V-BLAST suggested by Foschini *et al.* [7] where full spatial diversity is usually not achieved. Finally, the third type exploits the knowledge of channel at the transmitter. It decomposes the channel coefficient matrix using Singular Value Decomposition (SVD) and uses these decomposed unitary matrices as pre and post filters at the transmitter and the receiver to achieve near capacity [8].

In this paper, an efficient FPGA design of a (2x2) MIMO System (STBC) is presented. This paper is organized as follows. Section 2 presents about the number representation used in the design. Section 4 presents an overview of MIMO Systems. Section 3 describes the design of MIMO. Section 5 discusses the results and we conclude in Section 6 deals with the conclusion.

2. MULTIPLE INPUT MULTIPLE OUTPUT

A third dimension, space, is introduced into the traditional frequency-time domain. The idea is to transmit multiple streams of data on multiple antennas at the same frequency, to increase throughput. Typically, multiple receiver antennas are used as well, since this configuration achieves high data rates, multiplied by the number of channels between either end. This principle is called Multiple Input Multiple Output (MIMO) [1]. There are two methods widely used for transmitting MIMO data. If the channel has a negligible error rate, we can send several data simultaneously over multiple antennas. This is known as spatial multiplexing, which utilizes the spectrum very efficiently. In contrast, if the environment has high error rate, we transmit the same data over multiple antennas. This is called as space-time coding. The purpose of this approach is to increase the diversity of MIMO to combat signal fading.

The essential purpose of a MIMO system is to determine which antenna is corresponding to which data on the receiver side. As shown in Figure 1, Rx1 receives data from all the transmitter antennas, Tx1, Tx2, and Tx3. Thus, we must have a special decoding algorithm to identify which antenna has transmitted which data to Rx1.

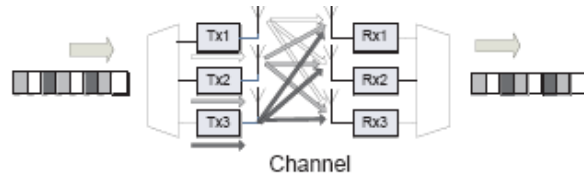
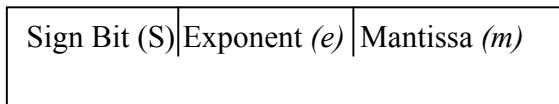


Fig. 2.1 The MIMO Principle [1]

3. NUMBER REPRESENTATION

The number system being used in the paper is floating point number representation, this is one of the important specification one should take care before starting the actually design. An 8 bit custom floating point format is used for representing the numbers.

Floating-point systems were developed to provide high resolution over a large dynamic range. Floating-point systems often can provide a solution when fixed-point systems, with their limited precision and dynamic range fail Floating-Point systems; however, bring a speed and complexity penalty. Most floating-point systems comply with the published single or double – precision IEEE floating –point standard A standard floating-point word consists of a sign-bit S, exponent e, and an unsigned (fractional) normalized mantissa m, arranged as follows: [3]



Algebraically, a floating-point word is represented by

$$X = (-1)^S \times 1.m \times 2^{e-bias} \quad ..(2.1)$$

This is a signed magnitude format. The hidden one in the mantissa is not present in the binary coding of the floating point number. If the exponent is represented with E bits then the bias is selected to be

$$bias = 2^{E-1} - 1 \quad ..(2.2)$$

3.1 Eight bit Custom Floating Point Format

Custom floating point format is used, i.e. 3 bits are used for representing mantissa, 4 bits for exponent and the other is the sign bit. To illustrate, let us determine decimal equivalent of 3.75 in the 8 bit custom floating point format

Consider a floating point representation with sign bit, E=4 bit exponent width and M=3-bit for mantissa (not including the hidden one). Let us now determine the representation of 3.75_{10} in this (1, 4, 3) floating point format using (2.2) the bias is

$$bias = 2^{E-1} - 1$$

$$bias = 2^{4-1} - 1 = 2^3 - 1 = 7$$

and mantissa needs to be normalized according to 1.m format i.e

$$3.75_{10} = 11.11_2 = 1.111 \times 2^3$$

|
mantissa(m)

The biased exponent is therefore represented with $E = 1 + bias = 1 + 7 = 1000_2$

Finally, we can represent 3.75_{10} in the (1, 4, 3) floating point format is represented as

Sign Bit(S)	Exponent (e)	Unsigned mantissa (m)
0	1000	111

Hence $3.75 = 01000111$ in floating point format

Due to the large gate count capacity of current FPGA's the design of floating point arithmetic has become a viable option. In addition, the introduction of the embedded 18 x 18 bit array multiplier Xilinx Virtex 2, Spartan 3 and Altera Stratix FPGA device families allows an efficient design of custom floating point arithmetic.[10]

Floating point Multiplier and adder were developed according to the standard IEEE floating point algorithms.

4. MIMO ARCHITECTURE DESIGN

The Top block of the entire system is shown in figure 4.1; it consists of the following Components Input and Output Memory (for storing input and output bit stream) MIMO Encoder (Alamouti, Space time encoder) 16 QAM Modulator 16 QAM Demodulator, MIMO Decoder (space time decoder) and Wireless channel model. In the following section the design of each module and its design is been discussed in detail.

The input bit stream which is to be transmitted is stored in a text file, from where it is stored in a memory. And the output of the memory is the input of the modulator; similarly the output from demodulator is stored in the output memory.

The input bits are stored in a memory, from where the bits are passed to the Modulator these bits acts as input to modulator. At every clock pulse bits stored in memory are passed through the modulator, which gives output symbols (i.e. real and imaginary values), in the same way at every trigger of clock the bits keep on passing from memory to modulator and the QAM demodulator, the output bits coming out of demodulator are stored in the output memory and a comparison is done between output and input memory to find bit error rate. The Modulation Scheme which has been employed is 16-QAM.

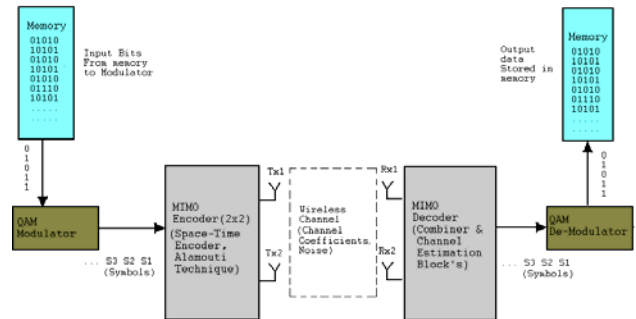


Fig. 4.1 MIMO-Top Level Block Diagram

The MIMO Encoder Module is as shown in figure 4.2. The port connections are also shown in the figure below. Input to MIMO encoder comes from QAM modulator block.

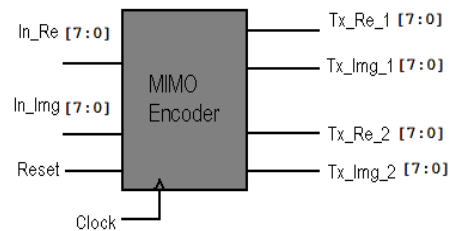


Fig. 4.2 MIMO Encoder Module

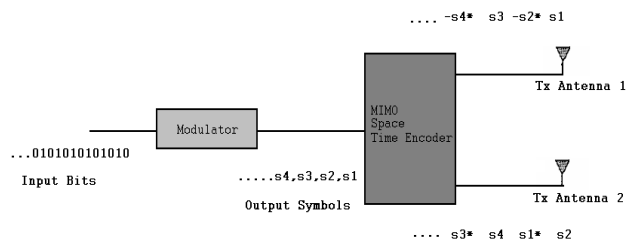


Fig. 4.3 MIMO Encoding and Transmitting scheme

Hence, 000000110001 = 0.56250 as shown in simulation in figure 5.1. This makes the task easier for verification. This is a function written in verilog. This function is only used for verification of results and cannot be synthesized. We can plug in this function anywhere in the system and check the equivalent real values. Below are the snapshots of simulation results.

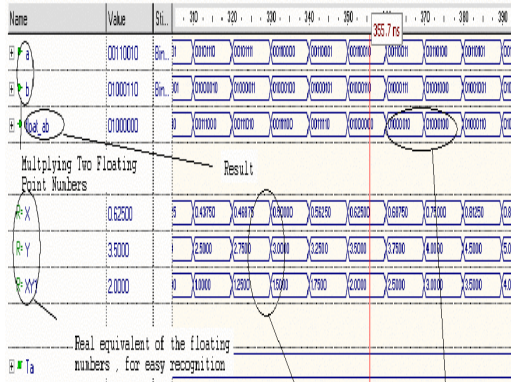


Fig. 5.2 Floating Point Multiplier Simulation

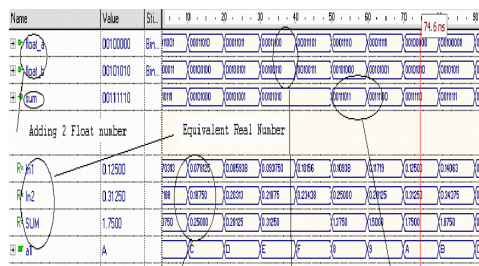


Fig. 5.3 Floating Point Adder Simulation

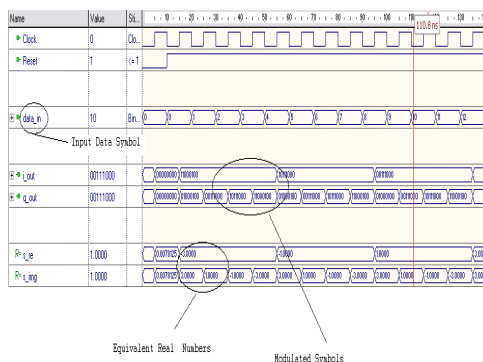


Fig. 5.4 QAM Modulator Simulation

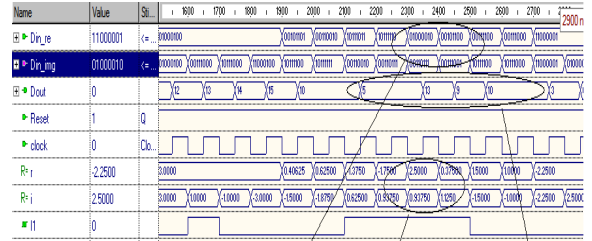


Fig. 5.5 QAM Demodulator Simulation

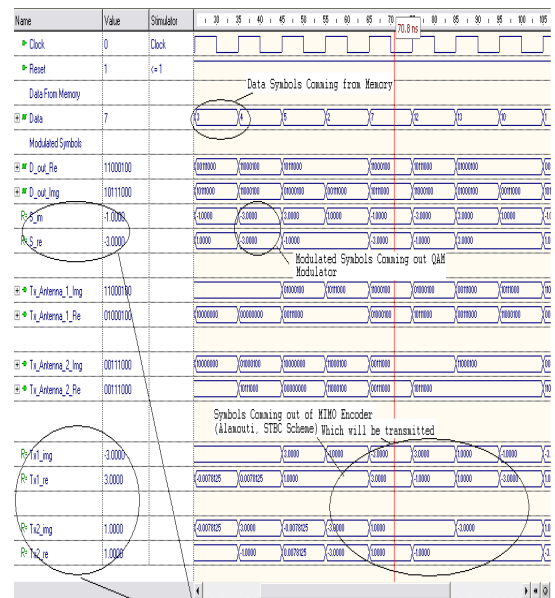


Fig. 5.6 MIMO Transmission Simulations

6. CONCLUSIONS

In this paper MIMO system design has been discussed in detail, including all its modules. Verilog HDL code was written for all the modules within the system. Each individual module was tested for its correct functionality and then all the modules were integrated to form a one complete system.

This paper has resulted in the development of a hardware prototype of a MIMO Transmitter, Receiver and channel, which works at the speed of 200MHz on a Spartan-3 FPGA board, the number representation used, is IEEE floating point unlike fixed point, which is most commonly used. Floating point and the results are discussed. Floating-point systems were developed to provide high resolution over a large dynamic range. Floating-point systems often can provide a solution when fixed-point systems, with their limited precision and dynamic range fail Floating-Point systems; however,

bring a speed and complexity penalty. But we tried to reduce the complexity and increase the speed, by choosing a custom floating point format. The results are discussed and the FPGA Resource utilization are tabulated in Table 1.

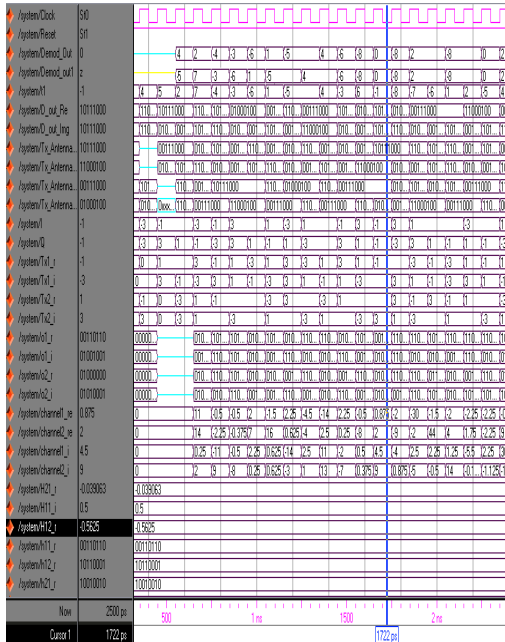


Fig. 5.7 Top Level Simulation

Table 1 Device Utilization Summary

Device Spartan 3 FPGA	Slices	Flip-Flops	4 input LUT's	IOB's	Block Rams	GClock's
Available Resources	1920	3840	3840	173	12	8
Resources Utilized by Each Module						
Input Memory	38 (1%)	32 (0%)	73 (1%)	6 (3%)	1 (8%)	1 (12%)
16 QAM Modulator	3 (0%)	12 (0%)	6 (0%)	22 (12%)	-	1 (12%)
MIMO STBC Encoder	111 (5%)	104 (2%)	203 (5%)	50 (28%)	1 (8%)	2 (25%)
MIMO STBC Decoder	1418 (73%)	96 (2%)	2532 (65%)	130 (75%)	-	3 (37%)
16 QAM Demodulator	15 (0%)	-	27 (0%)	22 (12%)	-	1 (12%)
Channel Model	1060 (55%)	-	1888 (49%)	65 (37%)	-	1 (12%)
Floating point Complex Multiplier	274 (14%)	-	490 (12%)	48 (27%)	-	1 (12%)
Floating Point Adder	71 (3%)	-	129 (3%)	24 (13%)	-	1 (12%)
Floating Point Multiplier	33 (1%)	-	59 (1%)	24 (1%)	-	1 (12%)

REFERENCES

[1] G. L. Stuber, J. R. Barry, S. W. McLaughlin, Y. (G.) Li, M. A. Ingram, and T. G. Pratt, "Broadband MIMO-OFDM Wireless Communications," Proceedings of the IEE, vol. 92, no. 2, pp. 271-294, Feb 2004.

[2] Jeoong S. Park, Hong-Jip Jung and Viktor K. Prasanna "Efficient FPGA-based Implementations of the MIMO-OFDM Physical Layer", IEEE Transactions on Communications, vol. 32, no. 18, pp. 571-577, August 2003

[3] Uwe Meyer-baese" digital signal processing with field programmable gate arrays"3rd edition ISBN: 978-3-540-72612-8

[4] S. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communications," *IEEE JSAC, Vol. 16, No. 8, October 1998.*

[5] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. on Information Theory*, vol. 45, pp. 1456-1467, July 1999.

[6] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. on Information Theory*, vol. 44, pp. 744-765, Mar. 1998.

[7] P. W. Wolniansky, G. J. Foschini, G. D. Golden and R. A. Valenzuela, "V-Blast: An Architecture for Realizing Very High Data Rates Over the Rich-Scattering Channel," *International Symposium on Signals, Systems and Electronics, ISSE 1998*, pp. 295-300, Sept. 29th - Oct. 2nd, 1998.

[8] J. Ha, A. N. Mody, J. H. Sung, J. Barry, S. McLaughlin and G. L. Stuber, "LDPC Coded OFDM with Alamouti/SVD Diversity Technique," *Kluwer Journal on Wireless Personal Communications*, 23(1):183-194, October 2002.

