Design and Implementation of an Area Efficient Interleaver for MIMO-OFDM Systems

[1]D.Nagaraju, [2]I.Sharath Chandra, [3]V.Karunakar Reddy

Assistant Professor, Department of ECE,
[1,3]Matrusri Engineering College, [2]GNIT
Hyderabad. Telangana, India.

Abstract- **This work is based on a memory-efficient and faster interleaver implementation technique for MIMO- OFDM communication systems on FPGA. The IEEE 802.16 standard is used as a reference for simulation and analysis.This is the method for interleaver design on FPGA and its memory utilization. This project work concentrate on efficient interleaver design for IEEE 802.16 system implemented on FPGA. Our goal is to achieve minimum memory usage, faster interleaving, and increased speed of the overall system. [80%]**

***Keywords*:** **MIMO-OFDM, IEEE 802.16, FEC, INTERLEAVING.**

I. INTRODUCTION

The IEEE 802.16 defines the standard for broadband wireless access covering the physical layer and medium access specifications for wireless metropolitan area networks (WMAN). The IEEE 802.16 Air Interface Standard is a technology that is playing a key role in fixed broadb and wireless MAN. The forward error correction (FEC) mechanism in the standard plays a very important role in its performance. A number of techniques are being used to achieve highly effective error-control coding such as Turbo codes and concatenated codes . However, interleaving also plays a major role in the FEC mechanism. The aim of interleaving is to reorder the incoming data and make the adjacent bits non -adjacent by a factor, to cope with the burst errors occurring during the transmission of data over the channel. Memory utilization and frequent memory access time are the crucial part of Interleaver design, targeting less memory utilization and reduced memory access in order to reduce the power dissipation of the overall system.[88%]

This paper is organized as follows .Section II presents an overview of proposed system. Section III presents the implementation details of MIMO-OFDM transmitter. Section IV discusses the hardware resource utilization of proposed system and Section V concludes the paper.[100%]

II. SYSTEM DISCRIPTIO N

A. BLOCK DIAGRAM

The basic OFDM communication system's block diagram is shown in Figure 1. The forward error correction (FEC) blocks include convolution encoding, puncturing, and interleaving. A modification of the system described in Figure 1[2] is to use two separate data streams to enhance the data rate and possibly increase the number of antennas by using spatial as well as transmit diversity.[100%]

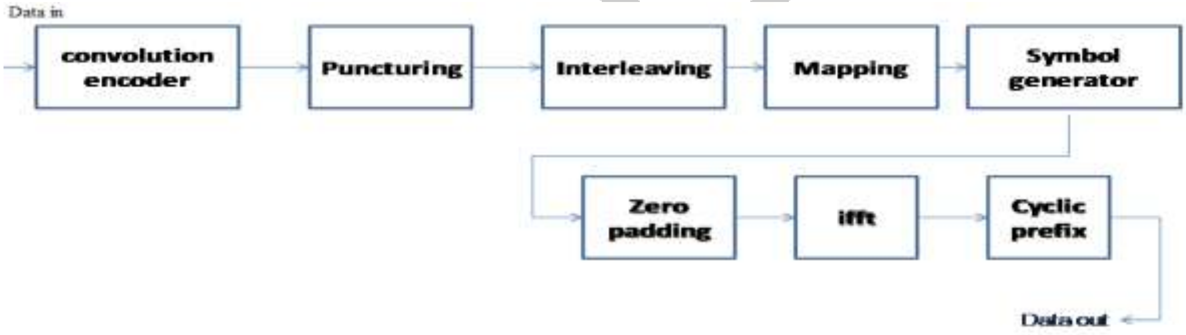

Figure 1: Block diagram for OFDM communication systems.

B. CONVOLUTIONAL ENCODING.

In telecommunication, convolution code is a type of error-correcting code in which

- ○ Each m-bit information symbol (each m-bit stream) to be encoded is transformed into an n - bit s symbol, where m/n is the code rate (n>m).

- ○ The transformation is a function of the k information symbols.[100%]

To convolutionally encode data, start with a *k* memory registers, each holding one bit input, All memory registers start with a value of zero, unless otherwise specified, The encoder has *n* modulo-2adders (a modulo 2 adder can be implemented with a single Boolean XOR gate, where the logic is : 0+0 = 0, 0+1 = 1, 1+0 = 1, 1+1 = 0), and *n* generator polynomials, one for each adder [3](see Figure2 below). An input bit *m*1 is fed into the leftmost register.[78%]

Using the generator polynomials and the existing values in the registers, the encoder output is n bits. Now shift all register values to the right and wait for the next input bit. If there are no remaining input bits, the encoder continues output until all registers have returned to the zero state.

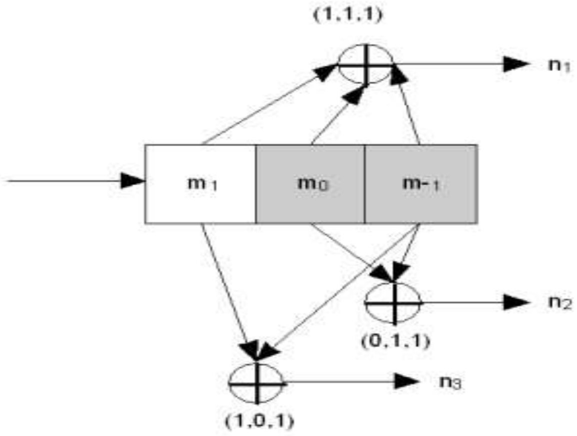

Figure 2: Rate 1/3 non-recursive, non systematic convolutional encoder with constraint length 3.

The Figure 2 shown above is a rate 1/3 (*m*/*n*) encoder with constraint length (*k*) of 3. Generator polynomials are *G*1 = (1, 1, 1), *G*2 = (0, 1, 1), and *G*3 = (1, 0,1). Therefore, output bits are calculated (modulo 2) as follows:

- □ *n*1 = *m*1 + *m*0 + *m-1*
- □ *n*2 = *m*0 + *m*-1
- □ *n*3 = *m*1 + *m*-1.[100%]

C. PUNCTURING.

In FEC, puncturing is the process of removing some of the redundant bits after encoding with an error-correction coding technique. This has the same effect as encoding with an error-correction code with a higher rate, or less redundancy. With puncturing the same decoder can be used regardless of how many bits have been punctured. Puncturing is often used with Viterbi algorithm in coding system.

A punctured code is obtained by eliminating encoded symbols from ordinary encoded symbols; this process is known as puncturing process.

If a rate of '1/n' parent encoder is punctured by eliminating some of the 'np' encoded bits to 'p' information bits .[4] After puncturing symbol from the encoded sequence corresponding to same amount of information is reduced therefore the rate of encoder is increased by puncturing process[100%].

Representation of punctured codes.

For rate 1/n parent encoder the puncturing pattern can be represent as 'nxp' matrix p whose elements are 1's and 0's with1indicating inclusion and 0 indicating deletion.

Example**:** given encoder generator polynomial.

G= [1+D^2 , 1+D+D^2] And punctured matrix as

$$P=\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$$

It indicates that within two encoded blocks, the first bit of second encoded block is eliminated.[86%]

D. INTERLEAVING

Interleaving is a process that makes a system more efficient, fast and reliable by arranging data in a non-contiguous manner. Uses of interleaving,

- Storage: As hard disks and other storage devices are used to store user and the system data, there is always a need to arrange the stored data in an appropriate way.
- Errors in data can be corrected through interleaving.
- Multi-Dimensional Data Structures.

Interleaver designs include:

- Rectangular (or uniform) interleaver (similar to the method using skip factors described above.)
- Convolutional Interleavers.
- Random interleavers [6]
- S-random interleaver (where the interleaver is a known random permutation with the constraint that no input symbols within distance S appear within a distance of S in the output)
- Another possible construction is a contention-free quadratic permutation polynomial. It is used for example in 3GPP long term evolution mobile telecommunication standard.[86%]

III . SYSTEM IMPLEMENTATION.

The IEEE 802.16 (WiMAX) system is implemented on FPGA for design emulation and verification. Convolutional encoder and puncturing blocks are implemented using shift registers and *XOR* gates. For QPSK, 16-QAM, and 64-QAM mapping, puncturing reduces the code rate to 3/4 as described in [1]. There is no puncturing used for BPSK mapping as the code rate is always rate 1/2. The interleaver is implemented using RAM blocks in the FPGA and employing logic cells (LC) for the state machine of the address generator for read/write operations. [89%]

Double buffering technique is used to implement the interleaver to eliminate the delay in the interleaving process. After the first block of symbols is stored in the buffer, the address generator[7] starts generating read addresses and starts reading data from the buffer. In the mean time, the second buffer is filled with incoming data and the interleaver will start reading from the second buffer after the first one is read out completely. Table I shows the buffer sizes for different interleaving schemes used in our system. The number of buffers increase with the increase in modulation symbol size, so that we can write/read data from them at the same time.[78%]

TABLE I
Buffer sizes for different modulation schemes:

Modulation schemes	BPSK	QPSK	16-QAM
Buffer size	384	384	384

A. Interleaver for BPSK Mapping:

For BPSK mapping, the interleaver is implemented using a single memory block of double the required size. For example, an interleaver of size 192 is implemented using a buffer of 384 bits as shown in table I. Incoming bits are first stored in the RAM until 192 bits are filled and then the read - out is enabled. A state machine generates write addresses for the RAM to write data to it. After 192 writes to the RAM, it asserts the read enable signal and starts generating read addresses for the RAM while continuing with the write process. This way the next part of the buffer is filled when the interleaver finishes reading the first 192 bits and then it starts reading the next 192 bits. [80%]

B. Interleaver for QPSK Mapping:

For QPSK mapping, the interleaver is implemented using two memory blocks of double the required size. A state machine generates a single address for the two RAMs to write data to them simultaneously. The pattern thus formed can be read out column-wise from RAM1 and RAM2 alternatively to implement the interleaver function. After 192 writes to each RAM, it asserts the read enable signals and starts generating read addresses for each RAM while continuing with the write process.
The address generator generates two read addresses successively with an increment of 6, in order to read 2 locations from one RAM during one read operation. The first column of RAM1 is read first, followed by the first column of RAM2 and this process continues until the last location is read.[9] This technique enables us to write 3 bits to the buffer at the same time and read 3 bits successively to generate a 3- bit symbol for QPSK mapping.[78%]

C. Interleaver for 16-QAM Mapping

For 16-QAM mapping the interleaver is implemented using four memory blocks of double the required size. The method of read and write address generation is the same as explained before except the fact that now we have four separate RAMs that are first filled simultaneously and then the data is read out column-wise from RAM1, RAM2, RAM3, and RAM4 alternatively to implement the interleaver function. The address generator generates four read addresses successively with an increment of 3, in order to read 4 locations from one RAM during one read operation. This technique enables us to write 5 bits to the buffer at the same time[10] and read 5 bits successively to generate a 5-bit symbol for 16-QAM mapping.[76%]

IV. HARDWARE RESOURCE UTILIZATION.

A. Interleaver Memory Utilization

Table II shows the RAM resource utilization for the different types of modulation schemes.

TABLE II

Mapping type	No. of Slices Used out of 5472	No. of 4 input LUTs out of 10944	No. of slice Flip flops out of 10944	Maximum Frequency (MHz)
BPSK	1025	1048	1188	44.592
QPSK	889	1226	981	42.592
16-QAM	1094	1323	1275	92.520

Each look-up table (LUT) is 32 bit and BRAM is 36 Kb, which can also be used in separate blocks of 18 Kb. As we can see, the implementation is very efficient in terms of RAM resource utilization if Distributed RAM extraction method is used during the synthesis of our design. However, if Auto RAM extraction is used, the synthesizer uses Block RAM resources to implement some of the memory for interleavers in order to save distributed RAM resources which results in an increase in the operating frequency of the overall system. In this implementation, most of the block RAM resources are wasted as the modulation size increases to 64-QAM.[89%]

From the table II , We can see that memory utilization is less for QPSK modulation scheme in terms of number of slices used in the look -up table[LUT] and 16-QAM is better in terms of frequency of operation.

V. CONCLUSION

In this paper, an efficient way to design the IEEE 802.16 transmitter on FPGA is presented. A special design method is used to implement the interleaver with minimum memory requirement and initial latency. This approach can also be used to design other high-speed communication systems or to improve their speed.

The proposed optimizations could be utilized in real time applications since they only require to replace the current interleaving parameters and do not involve any hardware alteration. The transmitter using different modulation schemes have been coded and simulated and compared with respect to area, frequency and power utilizations .

REFERENCES

[1] IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air Interface for Broadband Wireless Access Systems, IEEE Std. 802.16-2009.

[2] Fernandon H. Gregorio "OFDM phy coding and interleaving." May 2007.

[3] Y.-N. Chang, " A low-cost dual-mode deinterleaver design," IEEE Transactions on Consumer Electronics, May. 2008.

[4] X. Yin and J. Liu, " Design and implementation of an improved 3G turbo codes interleaver for 3GPP system," in 9th International Conference on Electronic Measurements and Instruments (ICEMI) , Beijing, China, Aug 2009.

[5] N. Crisan, L. C. Cremene, E. Puschita, and T. Palade, “Spectral Efficiency Improvement for the under -11 GHz Broadband Wireless Access,” in International Conference on Telecommunications, (ICT), Jun. 2008, pp.

[6] S.-J. Park and J.-H. Jeon, “Interleaver optimization of convolutional turbo code for 802.16 systems,” IEEE Communication Letters, May. 2009

[7] Z. Iqbal and S. Nooshabadi, “ Effects of Channel Coding and Interleaving in MIMO-OFDM Systems,” in 54th IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), Seoul, Korea, Aug. 2011.

[8] C. Berrou, Y. Saouter, C. Douillard, S. Kerouedan, and M. Jezequel, “ Designing good permutations for turbo codes: toward a single model,” in Proc. IEEE ICC, Paris, France, June 2004.

[9] S. Haene, D. Perels, and W. Fichtner, "System-Level Characterization of a Real-Time 4x4 MIMO-OFDM Transceiver on FPGA," in European Signal Processing Conference, Poznan, Poland, Sep. 2007.

[10] S. Haene, A. Burg, D. Perels, P. Luethi, N. Felber, and W. Fichtner, "FPGA Implementation of Viterbi Decoders for MIMO-BICM," in Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers, Oct. 28 - Nov. 01 2005.

[11] A. Jamin, and P .Mahonen, “ Wavelet packet based modulation for wireless communications,” Wiley Wireless Communications and Networking Journal, vol. 5, no. 2, pp. 123–137, Mar. 2005.

[12] F. S. Al-kamali , M. I. Dessouky, B. M. Sallam, F. E. Abd El-Samie , and F. Shawki, " An Efficient Transceiver scheme for SC-FDMA System Using A Wavelet Transform", IET Commun., 2010, Vol. 4, Iss. 1, pp. 69–79.

[13] H.G.Myung,“Single Carrier Orthogonal Multiple Access Technique for Broadband Wireless Communications,” Ph.D. Dissertation, Polytechnic University,Jan. 2007.

[14]F. Adachi, D. Garge, S. Takaoka, and K. Takeda, “ Broadband CDMA techniques,” IEEE Wireless Communs., Vol. 12, Issue 2, pp. 8-18, April 2005.

[15] X. Zhu, R.D. Murch, “ Novel Frequency-Domain Equalization Architectures for a Single-carrier Wireless MIMO system”, in proc. IEEE VTC, pp. 874-878, Fall 2002.