Power and Area Efficient Hardware Architecture for WiMAX Interleaving

Zuber M. Patel
Dept. of Electronics Engg., S.V. National Institute of Technology, Surat, India
Email: zuber_patel@rediffmail.com

***Abstract*—In this paper, area and power efficient design of interleaver/deinterleaver for IEEE 802.16 (WiMAX) networks is presented. Interleaving plays an important role in wireless networks in combating burst errors. It spreads burst error among multiple code words, thus reduces erroneous bits per code word symbol which can be corrected by forward error correction (FEC) decoder. The paper proposes an efficient hardware design that avoids look-up table (LUT) ROM and complex address generator logic. It uses only simple linear address generator circuit and multiplexer (MUX) based efficient intra-column permutation logic. The design supports all modulation schemes and sub-channelization. ASIC implementation results reveal that total number of gate count for interleaver is 25.9k and for deinterleaver is 26.1k. The combined system takes core chip area of 1.11mm^2 and consumes power of 0.586mW at 5MHz frequency.**

***Index Terms*—burst-error, interleaving, permutation, WiMAX**

I. Introduction

Success of wireless broadband technologies depends on ability to reduce bit error rate (BER). The wireless technologies such as WiFi and IEEE 802.16 [1] use OFDM/OFDMA techniques since it is resistant to fading and uses spectrum efficiently. However, when frequency selective channel is in deep fade [2], subcarriers may suffer from strong noise interference causing large burst-errors. Deep fading also reduces FEC capability of channel coding. The impact of deep fading can be minimized by using technique of interleaving [3]. The basic function of interleaver is to protect transmitted data from burst errors. The interleaver reorders the data such that error burst is spread over many code words so that each received code word exhibits only few symbol errors which can be corrected by FEC decoder. Interleaving is usually followed by channel encoding unit and is quite effective in combating burst-errors. There are two fundamental methods for interleaving. The first method is to divide data stream into blocks and permute data within each block. This type of interleaving is referred to as block interleaving. The block interleaver operates on block of input data bits at a time and there is no interleaving between the blocks. A simplest kind of block interlever writes data bits row wise in memory and then reading data column wise. But more general interleaver supports configurable rows and columns and permutation. The second type of interleaving is called convolutional interleaving [4] which reorders the data stream in a regular sliding window approach. A convolutional interleaver consists of a set of shift registers, each with a fixed delay. These delays are nonnegative integer multiples of a fixed integer. Each new symbol from the input signal feeds into the next shift register and the oldest symbol in that register becomes part of the output signal. Digital Video Broadcasting (DVB) standard uses convolutional interleaving as its outer interleaving method. In this paper, block interleaving is addressed.

In block interleaving, reordering of data may involve complex operations. Interleaver permutes data according to mapping and correspondingly deinterleaver uses inverse mapping to produce original sequence of data. The hardware of interleaver/deinterleaver normally consists of memory, address generator logic and permutation logic. Depending upon block size and complexity of mapping function, this hardware may occupy considerable real estate silicon area. Hence, many paper authors in research literature [5], [6] and [7] have addressed issue of efficient hardware implementation of interleaver/deinterleaver. In this paper, hardware design of IEEE 802.16 interleaver/deinterleaver is addressed with the aim of reducing area and power. Proposed design eliminates look-up table (LUT) ROM and uses simple address generator circuitry. The design supports various modulation schemes and sub-channelization specified in WiMAX standard.

The rest of the paper is organized as follows. Interleaving adopted in IEEE 802.16 standard is discussed in Section 2. Section 3 presents proposed architecture for hardware implementation of WiMAX interleaver and deinterleaver. ASIC Implementation results are presented in Section 4. Finally, the conclusion remarks are given in Section 5.

II. Interleaving for IEEE 802.16

Due to its capability to combat burst-error, interleaving finds its place in the IEEE 802.16 PHY layer after FEC encoder. By distributing burst-error, interleaver improves BER performance of wireless communication system. WiMAX PHY layer overview is in Fig. 1 with all baseband processing units. The very first unit is scrambler that avoids long stream of 1s and 0s which may

Manuscript received July 24, 2014; revised September 22, 2014.

doi: 10.12720/ijsps.3.1.60-64

cause timing synchronization problem at receiver. The randomized data is then sent to Reed Solomon Convolutional Concatenated (RS-CC) FEC encoder that adds structured redundant bits to achieve error correction on receiver side. RS-CC encoded data are interleaved by block interleaver. Different block size and permutation schemes are operated in interleaving depending on the modulation scheme, rate and sub-channelization. After interleaving, data is sent to mapper and IFFT units. On the receiver side, reverse operations are performed to get back original data.

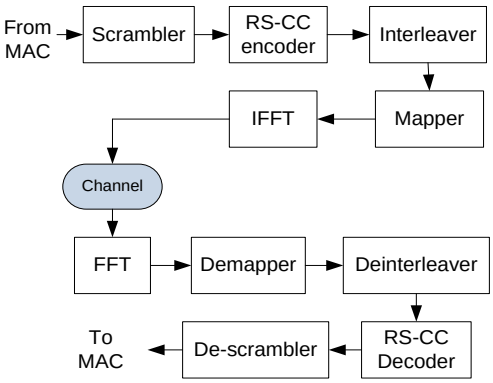

Figure 1. Overview of WiMAX PHY layer baseband processing

In WiMAX, block interleaving is used which operate on block of encoded data. The block size is corresponding to the number of coded bits per the allocated per OFDM symbol i.e. N_{cbps}. Let N_{cpc} be the number of the coded bits per subcarrier which is equal to 2, 4 and 6 for QPSK, 16-QAM and 64-QAM schemes respectively. Defining $s = ceil(N_{cpc}/2)$ we get s equal to 1, 2 and 3 for above mentioned three modulation schemes. The interleaving operation is specified by two step permutation. We denote index k of coded bits before first permutation, m after first permutation and j after the second permutation. The first permutation step is defined by following equation.

$$m = (N_{cpbs}/16) \bullet k_{mod(16)} + \lfloor k/16 \rfloor \quad (1)$$

where $k = 0, 1, 2, ..., N_{cpbs}-1$

The second permutation is defined by equation

$$j = s \bullet \lfloor m/s \rfloor + (m + N_{cbps} - \lfloor 16m/N_{cbps} \rfloor)_{mod(s)} \quad (2)$$

The first permutation ensures that adjacent coded bits are mapped onto nonadjacent subcarriers. The second permutation ensures that adjacent coded bits are mapped alternately onto less or more significant bits of the constellation, thus avoiding long runs of low reliable bits. Closer study reveals that the first permutation given in (1) defines a block interleaver with 16 columns and a variable number of rows. In other words, (1) can be considered as matrix transposition operation which transposes original data into $(N_{cbps}/16)\times16$ matrix. The block size for IEEE 802.16 interleaver i.e. N_{cbps} depends on the modulation and subchannel partition and is listed in Table I. With fixed number columns, number of rows varies depending on size of N_{cbps}.

TABLE I. THE VALUES OF N_{CBPS} SUPPORTED IN WIMAX

Modulation Scheme	Number of Coded bits per symbol N_{cbps}		
	1 subchannel	2 subchannels	4 subchannels
QPSK	96	192	384
16-QAM	192	384	768
64-QAM	288	576	1152

The second permutation given in (2) defines intra-column permutations which may also differ between columns [8]. For s equals 1 (case of QPSK), this step of permutation actually does not alter the data sequence in columns. For the other cases, it reorders the data in the columns of the matrix except for the data in those columns with the column coordinate which can be dividable by s. In these columns, data is divided into small groups of data with size s, and data in group are locally permutated. An example of data permutation is illustrated in Fig. 2 for the case of 16-QAM and in Fig. 3 for case of 64-QAM for 1 subchannel. The number in the matrix is used to denote the index of corresponding input data for the associated matrix.

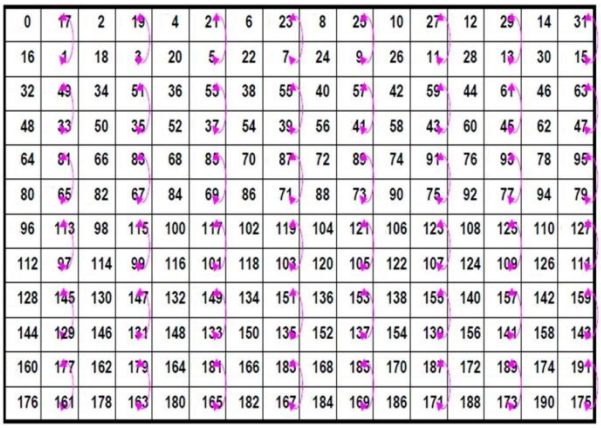

Figure 2. 12x16 block interleaving for 16-AQAM

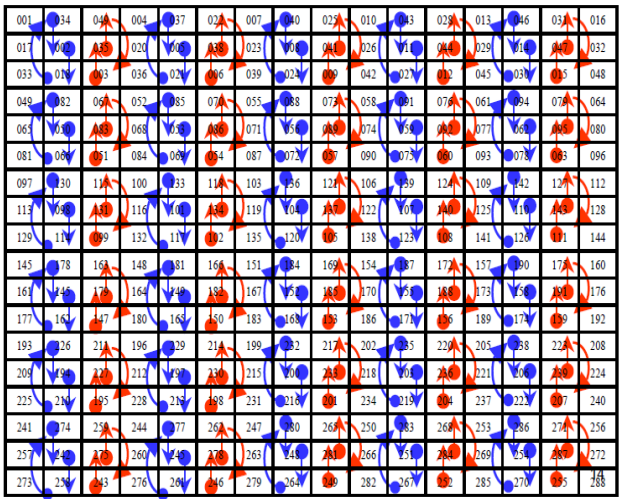

Figure 3. 18x16 block interleaving for 64-QAM

To reorder the interleaved data sequence back into the original one, the deinterleaver has to reverse the operation of the second permutation defined in (2), followed by the reverse of the first permutation defined in (1). Some works have devised multimode architectures that can support multiple wireless standards. The proposal [9] has addressed interleaving for both DVB and IEEE 802.16 networks whereas other work [10] supports IEEE 802.11 and 802.16 standards.

III. Proposed Hardware Architecture for Interleaving

In this section, we shall discuss interleaving implementation in hardware. Traditional block interleaver implementation comprises of a bit addressable memory and a ROM (or look up table) for storing the interleaving sequence. This structure is very general and supports many different type of interleaving sequence. However it requires that every bit has to read and written one at a time and ROM takes very large area as it has to store entire interleaving sequence. This will increase area and power dissipation excessively. The proposed architecture uses multi-bank 2D memory where a complete row or column can be read or written in single clock cycle. With efficient permutation logic and ROM less hardware, proposed hardware architecture realizes interleaving with low area and power.

A. Block Interleaver

The basic units of block interleaver are data memory, address generator and/or permutation logic. The memory is used to store data block of size N_{cbps} (which may vary depending upon modulation scheme and sub-channel partition) and address logic computes address and control signals for data memory. In earlier designs [5], [6], address generators use look-up table (LUT) ROM that stores order of addresses for obtaining desired interleaved output data stream. Depending on modulation scheme, LUT address is appropriately modified by logic to construct correct memory address. This approach has two disadvantages. First, it needs ROM for storing look-up table which increases silicon area. Second, it causes high switching activity on address lines of data memory as address changes for each bit to be read.

In proposed design, LUT ROM is completely avoided and address generator circuits is made very much simple. The basic interleaver structure (Fig. 4) consists of two data memory blocks each divided into 12 banks. The bank size is determined from smallest block size N_{cpbs} supported in IEEE 802.16 standard which is 96 bits. Hence each bank is organized as 6 rows and 16 columns. The largest block size (1152 bits) needs 12 banks and occupies entire memory block. Each memory block has two address inputs; row address (r_add) for writing data stream in memory and column address (c_add) for reading data. These addresses are used to read/write data in the selected memory bank. The row address is 3-bit wide and column address is 4-bit wide. During operation, when interleaver writes entire block of size N_{cbps} in one memory row-wise as 16-bit words, it also reads data column-wise from each bank. It reads 1st column of all banks then 2nd column of all banks and so on. Thus, it outputs 6-bit data during each read cycle which is sent to permutation logic. Our architecture writes a complete row and reads a complete column of memory bank in single clock cycle. Hence, it needs less address lines and less number of clock cycles to complete interleaving operation as compared to ROM based implementation. This reduces power consumption of proposed interleaver.

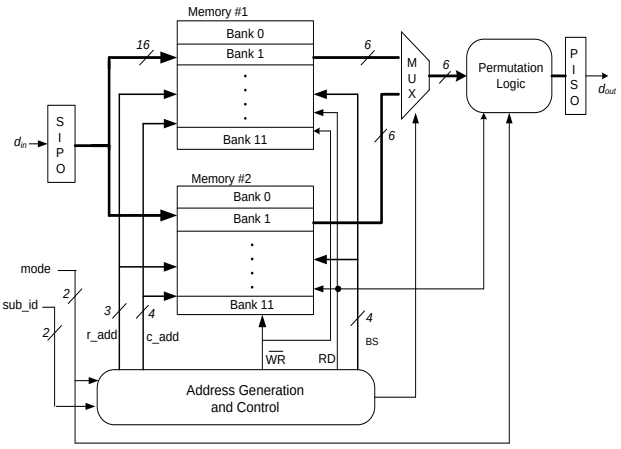

Figure 4. An architecture of proposed interleaver

In proposed architecture, address generator updates address linearly and hence it requires less hardware. Additionally, it generates separate RD and WR control signals and bank select (BS) control signal for data memory. The RD control signal is also used in permutation logic block and as load enable in output 6-bit parallel in serial out (PISO) register. The WR signal is connected as load enable in serial in parallel out (SIPO) register to place 16-bit data on data bus. The input signal mode (M_1M_0) signifies type of modulation and sub_id (I_1I_0) signifies the sub-channel partition presently used. The sub_id input is used by address generator and control to determine range of 4-bit value of BS. For largest block size of 1152, BS ranges from 0000 to 1011.

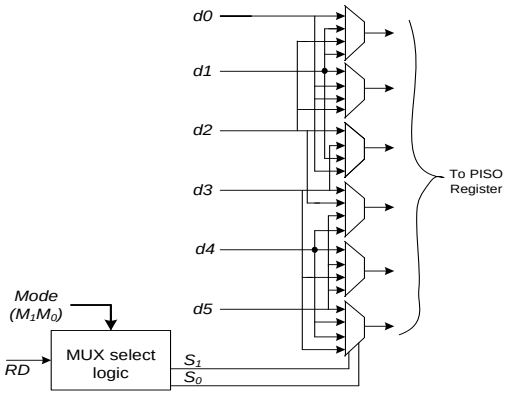

Figure 5. Permutation logic for proposed interleaver

During memory read operation, 6-bit data are produced which are fed to permutation logic. The permutation logic (Fig. 5) performs different intra-column permutation for different modulation schemes as defined by IEEE 802.16

standard. It consists of group of six 4x1 multiplexers and a MUX select logic. Depending upon the current modulation scheme, MUX select logic places appropriate value on select lines (S_1S_0) to achieve desired permutation. The select lines are connected to select input of all six multiplexers. S_1S_0 is updated at every memory read operation and they cycle in different sequence of states depending on modulation scheme. Since no intra-column permutation is performed for QPSK, the MUX select logic places fixed value 00 on select lines S_1S_0. For 16-QAM, S_1S_0 cycles in 00 and 01 states and for 64-QAM S_1S_0 cycles in three states 00, 10 and 11 at every memory read operation.

B. Block Deinterleaver

The architecture of deinterleaver is quite similar to interleaver as it has to reverse the operation of interleaving. Input bits are grouped into 6-bits by input SIPO register. Permutation logic performs data swapping on this 6-bit data using group of six 4x1 multiplexers according to modulation scheme. Hence, in deinterleaver data are permuted before it is written to memory. The data at the output of permutation logic are then written column-wise in the banks of memory. In deinterleaver, 1st column of all banks are written first, then 2nd columns of all banks and so on. Once memory block #1 is written completely, control enables write operation in memory block #2 and starts reading 16-bit words from memory block #1 at the same time. The deinterleaver reads 16-bit data row-wise from memory and loads it to PISO register that outputs deinterleaved data serially.

TABLE II. ASIC PERFORMANCE COMPARISON

Parameter	This Design	Paper Ref. [11]
CMOS Library	0.18µ UMC	0.18µ TSMC
Area (core)	1.11 mm^2	1.32mm^2
Power	0.586mW@5MHz	1.1mW@5MHz
Max. Freq.	280MHz	-
Gate Count	52062 (combined)	-

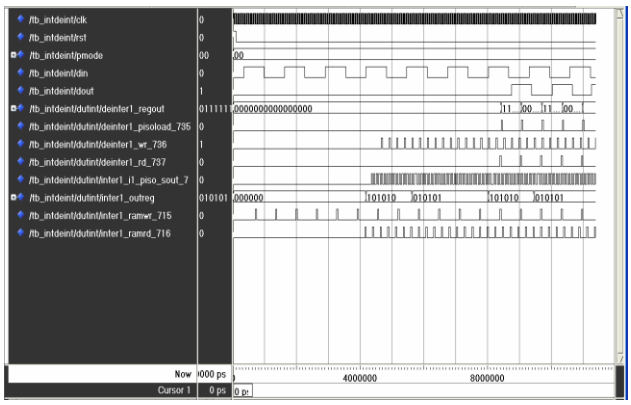

Figure 6. Post layout simulation for Interleaving/deinterleaving operation for mode 00 (QPSK)

IV. ASIC IMPLEMENTATION

The combined system involving both interleaver and deinterleaver is developed in VHDL and synthesized with Synopsis Design Complier synthesis tool using 0.18µ CMOS technology. The netlist obtained after synthesis is imported in Cadence SoC encounter tool to complete layout. Implementation results show that interleaver consumes 25,942 gates whereas deinterleaver consumes 26,120 gates in total. The combined system can run at frequency of 280 MHz and dissipates 0.586mW of power at 5MHz. Our results are compared with other similar design in Table II which reveals improvement of proposed design in terms of area and power. Fig. 6 shows post layout simulation with QPSK modulation mode.

V. CONCLUSION

The paper presents an efficient hardware implementation of block interleaver/deinterleaver module defined at PHY layer of IEEE 802.16 standard. The proposed architecture entirely eliminates the need of look-up table ROM used in earlier designs. Besides, it uses less complex address generator circuit with linear increment in addresses. The overall hardware with permutation logic occupies less silicon area. The ASIC implementation shows improvement in both area and power compared to others. Post layout simulation verifies the functionality of our design.

ACKNOWLEDGMENT

This work was supported by SMDP - II VLSI project from Ministry of Communication and Information Technology, Dept. of Information Technology, Govt. of India.

REFERENCES

[1] Local and Metropolitan Networks — Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1, IEEE Std 802.16e-2005, 2006.

[2] J. Li and M. Kavehrad, "Effects of time selective multipath fading on OFDM systems for broadband mobile applications," *IEEE Communication Letters*, vol. 3, no. 12, pp. 332-334, Dec. 1999.

[3] V. D. Nguyen and H.-P. Kuchenbecker, "Block interleaving for soft decision viterbi decoding in OFDM systems," in *Proc. IEEE 54th Vehicular Technology Conference*, Oct. 2001, pp. 470-474.

[4] F. Daneshgaran, M. Laddomada, and M. Mondin, "Interleaver design for serially concatenated convolutional codes: Theory and application," *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1177-1188, Jun. 2004.

[5] R. Asghar and D. Liu, "2D realization of WiMAX channel interleaver for efficient hardware implementation," in *Proc. World Academy of Science, Engineering and Technology*, Hong Kong, Mar. 2009, pp. 25-29.

[6] B. K. Upadhyaya and S. K. Sanyal, "Novel design of WiMAX multimode interleaver for efficient FPGA implementation using finite state machine based address generator," *International Journal of Communications*, vol. 6, no. 2, pp. 27-36, 2012.

[7] A. A. Khater, M. M. Khairy, and S. E.-D. Habib, "Efficient FPGA implementation for the IEEE 802.16e interleaver," in *Proc. Int. Conf. on Microelectron*, Marrakech, Morocco, 2009, pp. 181-184.

[8] E. Tell and D. Liu, "A hardware architecture for a multi mode block interleaver," in *Proc. International Conference on Circuits and Systems for Communications (ICCSC)*, Moscow, Russia, Jun. 2004.

[9] Y. N. Chang, "A low cost dual mode de-interleaver design," *IEEE Transaction on Consumer Electronics*, vol. 54, no. 2, pp. 326-332, May 2008.
[10] Y.-W. Wu and P. ting, "A high speed interleaver for emerging wireless communications," in *Proc. IEEE International Conference on Wireless Networks, Communications and Mobile Computing*, Jun. 2005, pp. 1192-1197.
[11] M. C. Ng, M. Vijayaraghavan, N. Dave, and Arvind, "From WiFi toWiMAX: Techniques for high-level IP reuse across different OFDM protocols," in *Proc. 5th IEEE/ACM International Conference on Formal Methods and Models for Codesign (MEMOCODE 2007)*, May 2007, pp. 71-80.

Zuber M. Patel received bachelor's degree in Electronics from National Institute of Technology (NIT), Surat (formerly REC, Surat) and Master's degree from Indian Institute of Technology Bombay (IITB), Powai, Mumbai. Presently, he is working as an assistant professor in the Department of Electronics Engineering, NIT Surat. He is also pursuing PhD from the same institution. He has published more than 8 research papers in International Journals and Conferences. His current research areas are VLSI and Embedded design of Wireless transceiver, Digital VLSI Design and FPGA.

