IOSR Journal of Engineering
Apr. 2012, Vol. 2(4) pp: 573-576

Implementation of unified architecture of 802.11a and 802.16a PHY layers using Verilog HDL (R & D)

Brijesh Darji[1], Bhavna Pancholi[2]

[1]*Electrical Engineering Department, Faculty of Technology and Engineering, M.S.University of Baroda, Gujarat, India.*
[2]*Electrical Engineering Department, Faculty of Technology and Engineering, M.S.University of Baroda, Gujarat, India.*
[1]*18, Sunrise classic society, B/H Arunoday Society, Vallabh Vidyanagar, Gujarat*

Abstract— WiFi and WiMAX are widely used wireless technologies for accessing internet. This paper elaborates the implementation of unified physical layers of WiFi and WiMAX technologies which are compliant to IEEE standards 802.11a and 802.16a respectively. The PHY specifications of these standards are described with block schematics. The scope of this paper is limited to the digital signal processing involved in the PHY layers of WiFi and WIMAX technologies.

Keywords: *OFDM, WiFi, WiMAX, Verilog*

1. INTRODUCTION

WiFi and WiMAX are the well developed and standardized technologies working on OFDM platform. Their physical layer architecture are much similar accept WiMAX physical layer has RS encoder & Decoder at transmitter and receiver respectively. In this paper the physical layer specification similarities and differences of IEEE 802.11a (WiFi) and IEEE 802.16a (WiMAX) are discussed. We used the conventions for WiFi as 11a and WiMAX as 16a for the entire paper.The architecture of PHY layer of the IEEE 802.16a is similar to IEEE 802.11a except some differences are stated as below.

Table 1. Architectural differences between IEEE 802.11a and IEEE 802.16a

Blocks of Unified PHY layer	**Change done in IEEE 802.11a(WiFi) configuration upon high level on WiFi/WiMAX select line**	**Change done in IEEE 802.16a(WiMAX) configuration upon low level on WiFi/WiMAX select line**
Scrambler	will be configured to 7 bits LFSR	will be configured to 15 bits LFSR
Descrambler	will be configured to 7 bits LFSR	will be configured to 15 bits LFSR
Reed-Solomon coder	N/A	will be enabled
Reed-Solomon decoder	N/A	will be enabled
Convolutional encoder	configuration does not change	configuration does not change
Viterbi decoder	configuration does not change	configuration does not change
Puncture	Appropriate puncture module will be selected (supporting 1/2, 2/3, 3/4 code rates)	Appropriate puncture module will be selected. (supporting 1/2, 2/3, 3/4 and 5/6 code rates)
De-puncture	Appropriate de-puncture module will be selected. (supporting 1/2, 2/3, 3/4 code rates)	Appropriate de-puncture module will be selected (supporting 1/2, 2/3, 3/4 code rates)
Interleaver	Appropriate interleaver module will be selected	Appropriate interleaver module will be selected
De-interleaver	Appropriate de-interleaver module will be selected	Appropriate de-interleaver module will be selected
Mapper	Appropriate mapper module will be elected. (BPSK, QPSK, 16 QAM, 64 QAM)	Appropriate mapper module will be selected (BPSK, QPSK, 16 QAM, 64 QAM)
De-mapper	Appropriate de-mapper module will be selected (BPSK, QPSK, 16 QAM, 64 QAM)	Appropriate de-mapper module will be selected (BPSK, QPSK, 16 QAM, 64 QAM)
Serial to parallel (transmitter)	will be configured to 48 bits SIPO register	will be configured to 200 bits SIPO register
parallel to serial with CP insertion (transmitter)	will be configured to 64 + CP insertion value, bits POSI register	will be configured to 256 + CP insertion value, bits POSI register
Serial to parallel with CP removal (receiver)	will be configured to 64 + CP removal value, bits SIPO register	will be configured to 256 + CP removal value, bits SIPO register
parallel to serial (receiver)	will be configured to 48 bits POSI register	will be configured to 200 bits POSI register
IFFT (transmitter)	will be configured to 64 point IFFT	will be configured to 256 point IFFT
FFT(receiver)	will be configured to 64 point FFT	will be configured to 256 point FFT

The organization of paper is as follows. Section 1 contains the comparison of physical layers of 11a and 16a. Section 2 contains Implementation of different blocks of unified architecture of 11a and 16a in verilog HDL. We conclude in section 3.

2. IMPLEMENTATION OF DIFFERENT BLOCKS

Scrambler:

Randomize input data to eliminate long streams of ones and zeros.

Scrambler of 802.11a

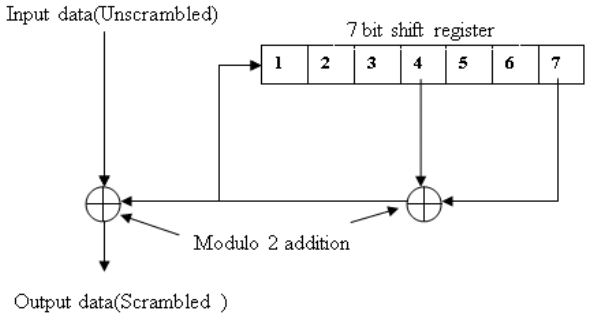

Fig 1 Scrambler of 802.11a

The scrambler removes the long streams of zeros and ones at transmitter which causes loss of synchronization at receiver. This is done by randomizing the input data with performing modulo 2 additions of input and the LFSR output having polynomial

$$S(x) = x^7+x^4+1$$

Here the initial state is used 1111111. i.e. all ones.

Scrambler of 802.16a

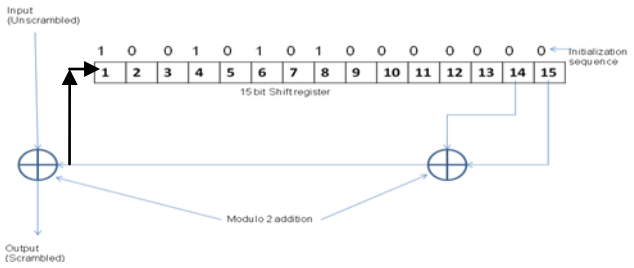

Fig 2 Scrambler of 802.16a

Here 15 bit LFSR is used with polynomial as below mentioned.

$$S(x) = x^{15}+x^{14}+1$$

The initial state used is 100101010000000.

Unified scrambler

As shown in figure , the unified scrambler has same hardware of 802.16a scrambler in addition with one 2:1 mux and one modulo 2 adder. This scrambler gets configured to 802.11a or 802.16a as per the select line WiFi/WiMAX. If the line is 1, then the WiFi data is selected at output and for low level on the line WiMAX data is selected at the output.

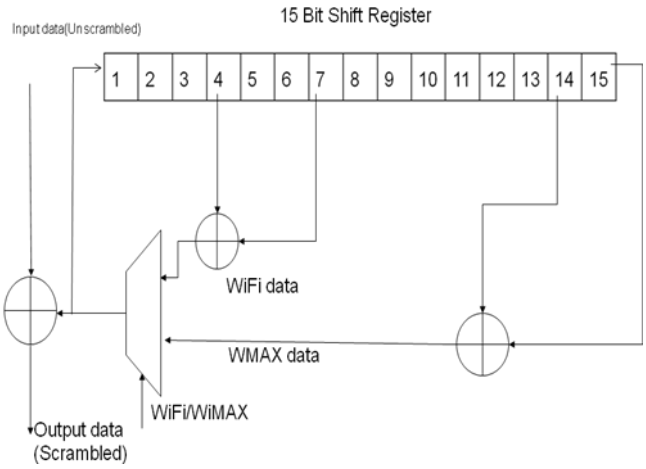

Fig 3 Unified scrambler

Functional simulation results of unified scrambler

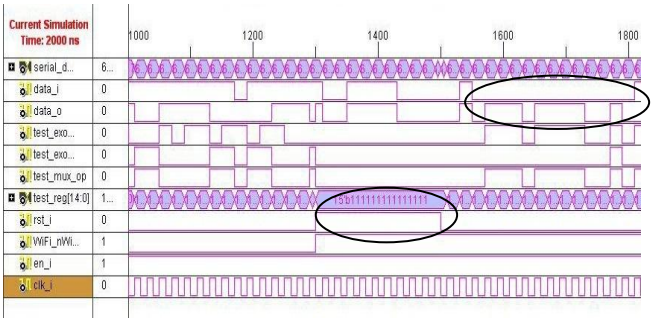

Fig 4 Waveform for WiFi with long stream of 0's as input

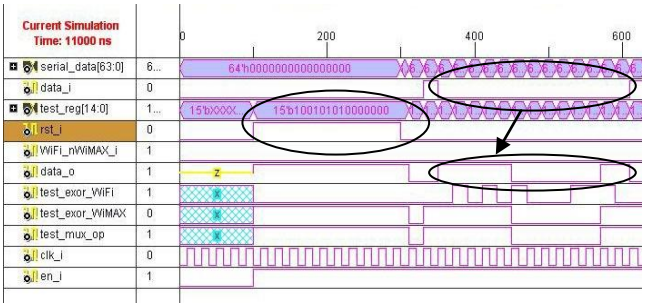

Fig 5 Waveform for WiMAX with long stream of 0's as input

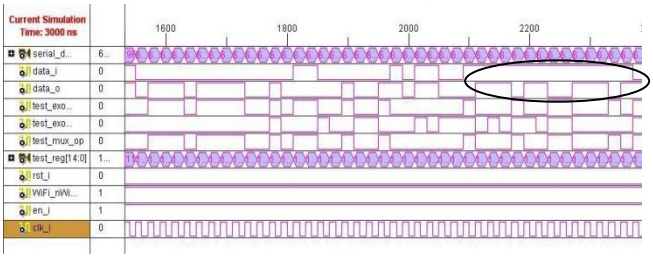

Fig 6 Waveform for WiFi with long stream of 1's as input

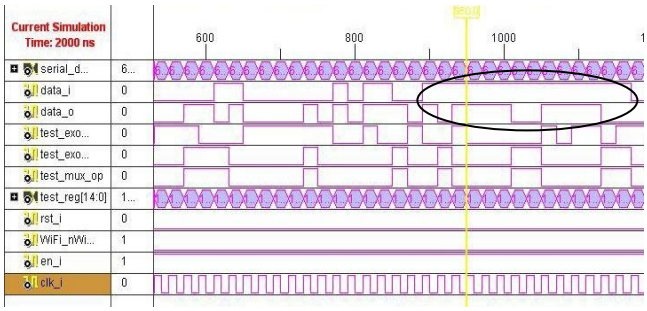

Fig 7 Waveform for WiMAX with long stream of 1's as input

Data_i shows input data stream. data_o shows output data stream.WiFi_nWiMAX is a selection line when it is 1 then WiFi is selected and when it is 0 then WiMAX is selected. test_reg is used to initialize shift register. rst_i is the reset input when it is high shift reg is reseted and initialized by 111111111111111 and if WiFi is selected then it is initialized by 100101010000000.

Implementation of Interleaver and de-interleaver

Interleaver is used for spreading the coded information over a block. This helps in preventing the information against the burst noise. For an uninterleaved coded data the burst noise can corrupt many bits and hence the viterbi decoder may not recover the original information. If the coded is interleaved then even in case of burst noise when the data is de-interleaved the burst noise is spread over entire block giving the viterbi decoder more chances of decoding the correct information. This is shown in fig 8. The same hardware is used for de-interleaver purpose with input and output swapped.

As shown in fig 8 the interleaver takes the input data row wise and outputs it column wise spreading the data over a block which is a size of interleaver.

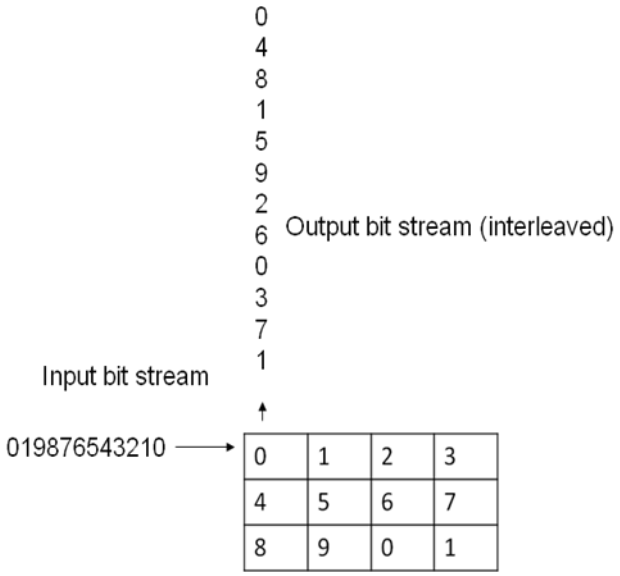

Fig 8 Working of interleaver

Specifications of interleaver for 802.11a

Block sizes: 48, 96, 192 and 288

Specifications of interleaver for 802.16a

Block sizes: 192, 384, 768 and 1154

Functional simulation waveforms of Interleaver and deinterleaver block

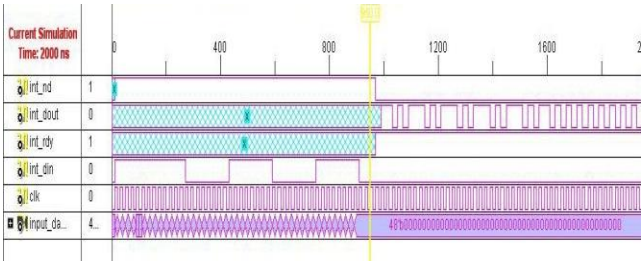

Fig 9 Waveform of interleaving for block size 48

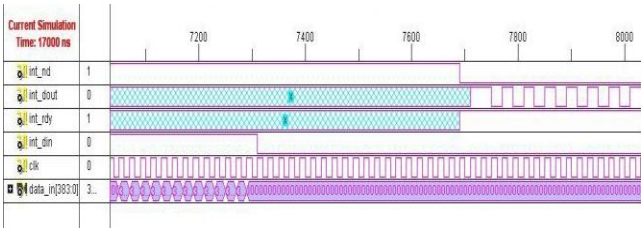

Fig 10 Waveform of interleaving for block size 384

'int_din' is the serial data from puncture block.'int_nd' remains high when interleaver is taking data from puncture block.'int_rdy' high informs that interleaved data is available on output data line 'int_dout' Here interleaver is implemented of block size 48 for WiFi (6-rows,8 columns) and 384 for WiMAX (16-rows,24-columns).

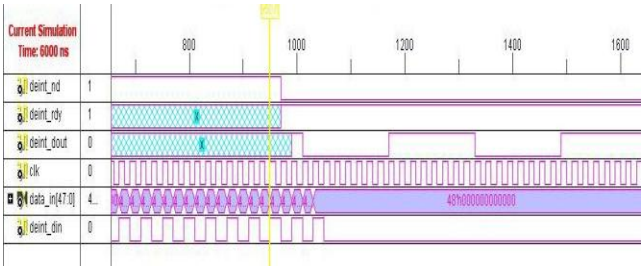

Fig 11 Waveform of deinterleaving for block size 48

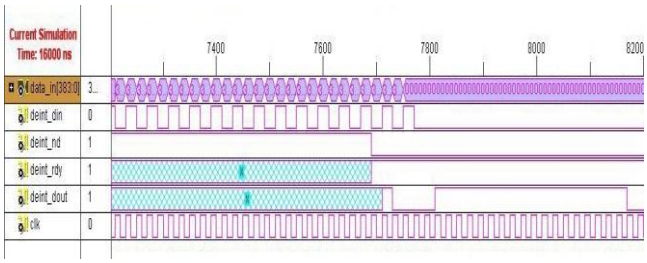

Fig 12 Waveform of deinterleaving for block size 384

'deint_din' is the serial data from demapper block. 'deint_nd' remains high when deinterleaver is taking data from demapper block.'deint_rdy' high informs that deinterleaved data is available on output data line 'deint_dout' Here deinterleaver is implemented of block size 48 for WiFi (8-rows,6-columns) and 384 for WiMAX (24-rows,16-columns).

3. CONCLUSIONS

This work derives the necessary results for the unified PHY layer implementation of 802.11a and 802.16a. Following are the key points to summarize the work done so far.

1) Understanding of PHY layers of 11a and 16a standards is achieved.
2) Implementation of unified scrambler in verilog HDL(Hardware Descriptive Language)
3) Implementation of interleaver in verilog HDL(Hardware Descriptive Language)
4) Implementation of deinterleaver in verilog HDL(Hardware Descriptive Language)

REFERENCES

[1] An Evaluation of Software Defined Radio – An Overview, QinetiQ Ltd., 2006.
[2] Alan C. Brooks, Stephen J. Hoelzer, Design and Simulation of Orthogonal Frequency Division Multiplexing (OFDM) Signaling, Final Report, May 15, 2001.
[3] Clause 17, IEEE Std 802.11a, Part 11:Wireless LAN Medium Access Control (MAC) and Physical Layer(PHY) Specifications, High-Speed Physical Layer in the 5 GHz Band. 2007.
[4] Subclause 8.3 and appendix B.1.2, IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems, 2009.
[5] Michael F Finneran, WiFi vs WiMAX: A comparison of Technologies, Markets and Business Plans, June 1, 2004.