Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Padding in WirelessMAN-OFDM with subchannelization**
Date Submitted	**2004-11-01**
Source(s)	Jon Labs, Rene Janicek, Rainer Ullmann Wavesat Inc. 1375 Trans-Canada Highway Dorval, QC H9P 2W8 Canada Voice: 514-684-0200 Fax: 514-684-0211 jlabs@wavesat.com
Re:	Call for Comments on Maintenance Issues regarding IEEE Standard 802.16 2004-09-24
Abstract	Document discusses an ambiguity in padding for OFDM-256 PHY with subchannelization.
Purpose	Consider recommended corrections for corrigendum.
Notice	
Release	
Patent Policy and Procedures	

Padding in WirelessMAN-OFDM with Subchannelization

Jon Labs
Rene Janicek
Rainer Ullmann
Wavesat, Inc.

The Issue

For a full BW OFDM symbol with 192 data subcarriers, the number of coded bits per OFDM symbol are either 192, 384, 768 or 1152, depending on whether the modulation is BPSK, QPSK, 16 QAM or 64 QAM, respectively. These coded block sizes are all divisible by 8, and they are therefore an integer number of bytes. Furthermore, when one determines how many bytes they correspond to before coding, one finds that an integer number of bytes is still obtained. For example, consider the case for 64-QAM. The number of bytes after coding is 1152/8 = 144. The two coding rates for 64-QAM are 2/3 and _, so before coding the number of bytes is 144*2/3 = 96 and 144*3/4=108 respectively.

With subchannelization, however, the numbers do not always divide evenly. The reason is the smallest granularity for subchannelization is 12 subcarriers. Non-integer byte allocations are an issue because the MAC PDU sizes are always in bytes.

The bit size for the coded subchannelized blocks that make up an OFDM symbol is given in Table 221 of the standard in Section 8.3.3.3, p. 441, line 29:

Table 221—Block sizes of the Bit Interleaver

	Default (16 subchannels)	8 subchannels	4 subchannels	2 subchannels	1 subchannel
	N_{cbps}				
BPSK	192	96	48	24	12
QPSK	384	192	96	48	24
16-QAM	768	384	192	96	48
64-QAM	1152	576	288	144	72

To take an example, consider two subchannels with QPSK. The number of bytes after coding is 48/8 = 6. The coding rates for QPSK are _ and _, so before coding the number of bytes is 6*1/2 = 3 bytes for QPSK-1/2 which is fine, but 6*3/4 = 4.5 bytes for QPSK-3/4 which is an issue.

We claim that the standard is ambiguous on the issue of how to pad. There are a number of pieces of text in the standard that discuss this issue:

1. *"Allocated space within a data burst that is unused shall be initialized to a known state. This may be accomplished by setting each unused byte to the stuff byte value (0xFF)."*—Section 6.3.3.7, p. 131, line 24

The only thing mandatory is in the first sentence, which says that the padding must use a known value. The text does not say one must use '1'b for the bits left over in non-integer allocations.

2. *"If the amount of data to transmit does not fit exactly the amount of data allocated, padding of 0xFF ('1' only) shall be added to the end of the transmission block."*—Section 8.3.3.1, p.430, line 56

This text is in the section on randomization for the OFDM-256 PHY. Randomization occurs before the error correction encoding. By saying "padding of 0xFF", the text suggest that the padding should be done only to fill in the unused *bytes* in the allocation, but one could argue it means for all unused *bits*.

3. *"When the total number of data bits in a burst is not an integer number of bytes, zero pad bits are added after the zero tail bits. The zero pad bits are not randomized. Note that this situation can occur only in subchannelization."* —Section 8.3.3.2.1, p. line 55

This text is specifically talking about the remaining bits in an allocation that does not make a full byte. Unfortunately this criteria does not cover all bases. Consider the example of a subchannelized allocation defined by 1 subchannel over 9 symbols using QPSK-1/2. For QPSK, there are 2 bits per subcarrier, so the number of coded bits is:

(9 symbols) x (1 subchannel/symbol) x (12 subcarriers/subchannel) x (2 bits/subcarrier)

= 216 bits

= 27 bytes

Before coding this becomes (given that one byte is reserved for the tail byte):

(27 bytes) x (1/2) –1 = 12.5 bytes

Therefore assume the MAC has passed to the PHY 12 bytes. Encoding and modulating 12 bytes gives:

[(12 bytes) + (1 byte)]*2 = 26 coded bytes = 208 bits

after encoding, i.e. we get an integer number of coded bytes, but we still do not fill in the allocation of 9 symbols of 1 subchannel.

What is important is that the number of bits coming out of the Convolutional encoder is modulo the number of bits required per interleaver block as given in Table 221 of the standard. This requirement leads to the recommended changes in the next section.

Recommended Changes

In IEEE P802.16-REVd/D5, Section 8.3.3.1, p. 430, line 56, change the text to:

"If the amount of data to transmit does not fit exactly the amount of data allocated, padding of 0xFF ('1' only) shall be added to the end of the transmission block <u>*for the unused integer bytes*</u>*. For RS-CC and CC encoded data (see 8.3.3.2.1), padding will be added to the end of the transmission block, up to the amount of data allocated minus one byte, which shall be reserved for the introduction of a 0x00 tail byte by the FEC. For BTC (8.3.3.2.2) and CTC (8.3.3.2.3), if implemented, padding will be added to the end of the transmission block, up to the amount of data allocated."*

And in Section 8.3.3.2.1, p.433, line 51, change the text to:

"The encoding is performed by first passing the data in block format through the RS encoder and then passing it through a convolutional encoder. A single 0x00 tail byte is appended to the end of each burst. This tail byte shall be appended after randomization. In the RS encoder, the redundant bits are sent before the input bits, keeping the 0x00 tail byte at the end of the allocation. When the total number of data bits in a burst is not an integer number of bytes, zero <u>*('0'b)*</u> *pad bits are added after the zero tail bits* <u>*until the number of bits after the convolutionial encoder is divisible by*</u> <u>N_{cbps}</u> <u>*as specified in Table 221*</u>*. The zero pad bits are not randomized. Note that this situation can occur only in subchannelization. In this case the RS encoding is not employed."*