

Project	IEEE 802.16 Broadband Wireless Access Working Group < http://ieee802.org/16 >		
Title	CRC32 Clarifications		
Date Submitted	2005-07-14		
Source(s)	Ilan Zohar Intel corp.	Ilan.zohar@intel.com Voice: +972-3-9205842	
Re:	Sponsor Ballot on IEEE P802.16-2004/Cor1/D3		
Abstract	While in wide use, the definition of CRC32 has always required understanding of the context or the help of test vectors. This contribution offers a standalone definition of CRC32.		
Purpose	This contribution clarifies the definition of CRC32		
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.		
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.		
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures < http://ieee802.org/16/ipr/patents/policy.html >, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair < mailto:chair@wirelessman.org > as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site < http://ieee802.org/16/ipr/patents/notices >.		

CRC32 clarifications

Ilan Zohar – Intel Corporation

1. Motivation

The standard uses CRC32 to detect possible PDU transmission errors and refers to 802.3 for its definition. CRC32 is used in other contexts as well (e.g. 802.11). In both cases implementing CRC32 requires either relying on test vectors or understanding not only of the definition of CRC32 itself but its context as well (such as the manner of transmission in 802.3).

This contribution presents a standalone definition of CRC32 (the same CRC used in 802.16, 802.3 and 802.11) conforming to the already existent test vectors in 802.16 and 802.3.

2. Proposed changes

Modify text in 6.3.3.5 as follows:

6.3.3.5 CRC calculation

A service flow may require that a CRC be added to each MAC PDU carrying data for that service flow (11.13.12). In this case, for each MAC PDU with HT=0, a CRC~~32~~, as defined in ~~6.3.3.5.1 IEEE 802.3~~, shall be appended to the payload of the MAC PDU; i.e., request MAC PDUs are unprotected. The CRC shall cover the generic MAC header and the Payload of the MAC PDU. The CRC shall be calculated after encryption; i.e. the CRC protects the Generic Header and the ciphered Payload.

6.3.3.5.1 CRC32 calculation

The data (input) bytes shall be flipped (for each byte exchange bit0 ↔ bit7, bit1 ↔ bit6, bit2 ↔ bit5, and bit3 ↔ bit4).

The CRC32 shall be calculated using the following standard generator polynomial of degree 32:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

The CRC32 is the 1's complement of the sum (modulo 2) of the following:

1. The remainder of x^k ($x^{31} + x^{30} + x^{29} + \dots + x^2 + x + 1$) divided (modulo 2) by $G(x)$, where k is the number of bits in the input data, and
2. The remainder after multiplication of the bit-flipped input data (treated as a polynomial) by x^{32} and then division by $G(x)$.

The CRC32 field shall then be transmitted bit-flipped commencing with the most significant byte (the first transmitted byte will have in its bit7 the coefficient of x^{24} and in bit0 the coefficient of x^{31} . The fourth byte will have the coefficient of x^0 in bit 7 and the coefficient of x^7 in bit0).

As a typical implementation, at the transmitter, the initial remainder of the division is preset to all 1's and is then modified by division of the bit-flipped data by the generator polynomial $G(x)$. The 1's complement of this remainder is then bit flipped byte after byte when transmitted, with the most significant byte first.

At the receiver, the initial remainder is preset to all 1's and the input bytes shall be flipped first and then treated as

coefficient of a polynomial. When divided by $G(x)$, this polynomial shall result in the absence of transmission errors, in a unique nonzero remainder value. The unique remainder value is the polynomial:

$x^{31} + x^{30} + x^{26} + x^{25} + x^{24} + x^{18} + x^{15} + x^{14} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^4 + x^3 + x + 1$
(or as its hexadecimal representation 0xC704DD7B)