**IEEE P802.15**

**Wireless Personal Area Networks**

|  |  |
| --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) |
| Title |  |
| Date Submitted | 20 September 2011 |
| Source | Jeff King (Elster Solutions) | Voice: Fax: E-mail:  |
| Re: | Editorial instructions and figures in support of proposed resolutions to sponsor ballot 4g comments  |
| Abstract | Editor instructions and figures |
| Purpose | Provide instructions to the editor to correct problems identified with the 4g comments in the sponsor ballot. |
| Notice | This document has been prepared to assist the IEEE P802.15. 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 acknowledges and accepts that this contribution becomes the property of IEEE and may be made publicly available by P802.15. |

CID 33:

Reject. The reason for rejection is that the FCS field will be updated when 4g is approved and its changes are rolled into the base specification.

CID 127: Revise. See below.

CID 128: Accept. See below.

CID 129: Revise. See below.

CID 130: Revise. See below.

CID 131: Revise. See below.

CID 278: Revise. See below.

CID 279: Accept. See below.

CID 280: Revise. See below.

CID 281: Revise. See below.

CID 282: Revise. See below.

CID 504: Revise. See below.

CID 506: Accept. See below.

CID 507: Revise. See below.

CID 508: Revise. See below.

On page 10, replace lines 9 through 30 with the following text:

All hopping sequences are referred to by an ID, *macHoppingSequenceID. A macHoppingSequenceID* of 0 denotes the default sequence for a particular PHY. This default sequence is a pseudo-randomly shuffled list of all channels available to the PHY. The mechanism to generate the default sequence is defined as follows:

1. SHUFFLE is a *macHoppingSequenceLength*-sized array. The contents of this array are equivalent to the first *macHoppingSequenceLength* outputs of a 9-bit linear feedback shift register (LFSR) with polynomial x^9 + x^5 + 1 and a starting seed of 255. Each LFSR output is modulo *macHoppingSequenceLength*, so that each entry of SHUFFLE is between 0 and (*macHoppingSequenceLength* - 1), inclusive.
2. CHANNELS is a *macHoppingSequenceLength*-sized array that is initially populated with the monotonically increasing list of channels used by the particular implementation.

	1. For cases where *macHoppingSequenceLength* is greater than the number of channels available to the PHY, this implies that some channels will appear multiple times in the array.
	2. For cases where *macHoppingSequenceLength* is less than or equal to the number of channels available to the PHY, some channels available to the PHY may be excluded from the array, while other channels appear multiple times.
	3. The selection of channels (the subset of available PHY channels, and which, if any, channels are used multiple times in the hopping sequence) is implementation-specific.
3. CHANNELS is shuffled as per figure XYZ.
4. The default sequence (i.e., *macHoppingSequenceList* for *macHoppingSequenceID* = 0) is equivalent to the shuffled CHANNELS array.

i =0

i < *macHoppingSequenceLength?*

No

Yes

Swap CHANNELS[i] and CHANNELS[SHUFFLE[i]]

i = i + 1

Figure XYZ

The use of other sequences (*macHoppingSequenceID* > 0) may be defined by a particular channel hopping system. The *macHoppingSequenceList* for a *macHoppingSequence >* 0 may be longer or shorter than the default sequence, and may be specified algorithmically or set as a predefined channel list. For two hopping devices to communicate, their PHYs must support the same number of channels, and they must either use the default hopping sequence, or agree upon the hopping sequence being used, either through carrying this information in an enhanced beacon frame, or through pre-configuration.

In general, any channel hopping MAC mode can calculate the present channel as follows: