**IEEE P802.15**

**Wireless Personal Area Networks**

|  |  |
| --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) |
| Title | **(Taylor) Format Conventions Proposal** |
| Date Submitted | [30 Sept., 2014] |
| Source | [Larry Taylor][DTC (UK)][ ] | Voice: [ ]Fax: [ ]E-mail: [larry.taylor@acm.org] |
| Re: | [Proposed Comment Resolutions to CID #’s – all bit order related comments] |
| Abstract | **Resolutions to multiple bit order comments** |
| Purpose | [see Re:] |
| 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. |

## *Insert the following new clause, suggest before clause 4 as either a new top level clause or a sub-clause of clause 3:*

## X. Format conventions

Throughout this standard, **unless otherwise stated**, data structures exposed in interfaces are represented using the conventions and formats defined in this clause.

Note: It is important to note that interfaces are depicted in this standard as bit serial by convention but may be implemented in other forms e.g. 4 or 8-bit parallel etc.

Within the PHY, data structures are passed to the reference modulator input in bit-serial convention. PHY specific coding may change the bit-order and encode multiple information bits into symbols. Hence the over-the-air interface may express the data structure content in a manner different than the conventions defined here.

## x.1. Fields

The general format of a data structure is shown in Figure 1. Each field is represented by a column in the Figure giving the size of the field and the name and/or type of the datum encoded in the field. The convention of a data structure is recursive in that a field may contain a data structure which is in turn composed of (sub-)fields.

|  |  |  |  |
| --- | --- | --- | --- |
| Octets:2 | Bit:0-5 | 6-7 | … |
| Field Name / Type (Number datum) | Field Name / Type (Bit string datum) | Field Name / Type (Bit string datum) | …. |

Figure 1: General Data Structure Format

The form <octet:n> means the field is n octets long. The form <bit:n> or <bits:m-n> means bit number n or bits m to n of the data structure. The terms octet and bit may also be written as octets or bits.

Fields are concatenated into larger data structures e.g. Figure 1 represents a data structure with 3 fields of length 2 octets, 6 bits and 2 bits for a total size of 3 octets.

Order of representation is strict with the leftmost field as shown in any Figure occurring before the next field to its right. The 2nd field from the left occurs before the 3rd field and so on.

Significance, as expressed in the terms LSB and MSB, only applies to numbers as defined in x.2. All other fields are treated as strings.

The convention of ‘processing a data structure’ or ‘transmitting a data structure' mean treatment of the content of the data structure in the order defined in the following sub-clauses..

## x.2. Numbers

Numbers are represented in binary (0bnnn) or hexadecimal (0xnnn) notation with the number of digits indicated. For example, 0b001 is a 3 digit binary number of value 1 and 0x001 is a 3 digit hexadecimal number of value 1. Numbers encoded in fields may be signed or unsigned integers. Other formats (e.g. a filed containing a floating point number) are stated in the field definition where applicable.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bit:0 | .. | 7 | Bits 8-(n-2) | n-1 |
| LSB | - | - | Subsequent octets | MSB |

Figure 2: General Number Field Format

When represented in fields, numbers are expressed with LSB leftmost and MSB rightmost as shown in Figure 2 for an n-bit unsigned integer. (Note that this is the inverse of the number representations 0bnnn and 0xnnn and of common number representations used in mathematical notations.) For example, the number 0x47 would be represented in an 8-bit field as 1110 0010.

Numbers of size greater than 1 octet occur with the octet containing the least significant bits first (leftmost) followed by octets containing bits of increasing significance to the right.

## x.3. Strings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bit:0 | .. | .. | 7 | Bits 8-k-1 |
| String bit 0 |  |  | String bit 7 | String bits 8 to k-1 |

Figure 3: General String Field Format

A string of length k bits is represented as a bit sequence numbered from 0 to k-1. Bit 0 occurs first and is represented as the leftmost bit of a string field. Bit 1 occurs to the right of bit 0 and so on until bit (k-1).

A string of length greater than 1 octet is represented by the octet containing the lowest numbered bits first and leftmost, followed by octets containing increasing bit numbers to the right.

## x.4 Reserved fields and values

Each bit within any Reserved field shall be set to zero on transmission and shall be ignored on reception.

No decision should be made on the contents of any Reserved field or field containing a reserved value.

## *Delete paragraphs 1 & 2 of clause 6.1.*

## *Delete the first line of clause 11.1*

## *Delete the 2nd and 3rd sentences of the first paragraph of clause 15.2.*

## *Delete the 2nd paragraph of clause 19.2*

## *Delete the 2nd sentence of the first paragraph of clause 19.2.3*

## *Delete the 2nd sentence of the first paragraph of clause 19.2.4*

## *Delete the 2nd sentence of the first paragraph of clause 20.2*

## *Clarify the meaning of ‘MSB first’ in clauses 20.2.3, 21.3.3, 23.1.3, 24.1.3, 26.1.3, 27.2.2*

## *Delete the 2nd paragraph of clause 23.1*

## *Delete the 2nd paragraph of clause 24.1*

## *Delete the last paragraph of clause 25.1.3*