**IEEE P802.15**

**Wireless Personal Area Networks**

|  |  |
| --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) |
| Title | **Kookmin Suggested Resolutions and Revisions on D0** |
| Date Submitted | [September 2016] |
| Source | Trang Nguyen, Nam Tuan Le, and Yeong Min Jang (Kookmin University)Sung-Hee Lee, Eynjun Rhee, and Nam-Kyung Lee (ETRI) |
| Re: | Short form of 460r2 |
| Abstract | Details of Resolutions regarding to the submitted Comments on D0 are suggested.Revisions on Kookmin related Sub-Clauses and Specifications are given.  |
| Purpose | D0 Comments Resolutions and Editorial Revision. |
| 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. |

## **16.6 Kookmin Invisible Mode** –

🡪 Changed to **16.6 Hidden Asynchronous- Quick Link (Hidden A-QL)**

The **Hidden Asynchronous- Quick Link (Hidden A-QL)** operates different PHY modes as shown in table 148.

**Table 148: Hidden A-QL PHY operating modes**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modulation**(m:n) | **RLL Code** | **Optical Clock Rate** | **FEC** | **Bit rate** |
| **Outer code (RS)** | **Inner code (CC)** |
| Hidden A-QL (n x m) | Differential code (1/2) | 10Hz | RS\_option | CC\_option | **mn** x RS\_rate x CC\_rate |

**16.6.1 Reference Architecture**

The reference architecture for Hidden A-QL is specified as shown in figure x1. Data bits are feed into FEC Encoder. The coded sequence is spitted; and then goes to RLL encoder to generate pairs of images according to the data input. The channel encoder will allocate data to fit the screen size.



**Figure x1. Reference Architecture**

**16.6.2 Channel encoder**

Data embedded image

Reference image (no data)

**Figure x2. Hidden A-QL channel encoder output**

The hidden A-QL encoder divides the Screen transmitter into nxm cells as shown in figure x2. The modulation of screen-cells is imperceptible by human eyes but perceptible by the camera receiver.

Four reference cells at corners are modulated by the clock information (2x2 matrix via reference-cells) whereas the other data cells are modulated by the data bits (nxm – 4 bits).

Also, the data block utilizes 1/2-rate line coding (see the sub-clause 16.6.3). The optical clock rate is usually specified at 10Hz and therefore, the block rate is at 5Hz to support 30fps camera receiver that has time-variant frame rate.

The modulation of four reference cells and data cells are shown as table x1.

**Table x1– Hidden A-QL data block format**

|  |  |  |
| --- | --- | --- |
|  | Data block time (i) | Data block time (i+1) |
|  | Reference image | Data emdeded image | Reference image | Data emdeded image |
| $n×m$ cells | invisible 1.png | C:\Users\Trang\Desktop\Warsaw\New folder\invisible\invisible 2.png | invisible 1.png | C:\Users\Trang\Desktop\Warsaw\New folder\invisible\invisible 2.png |
| Clock-and-rotation information atreference-cells | $$R\_{2×2}=\left[\begin{matrix}1&1\\1&0\end{matrix}\right]$$ | $$R'\_{2×2}=\left[\begin{matrix}0&0\\0&1\end{matrix}\right]$$ | $$R\_{2×2}=\left[\begin{matrix}1&1\\1&0\end{matrix}\right]$$ | $$R'\_{2×2}=\left[\begin{matrix}0&0\\0&1\end{matrix}\right]$$ |
| Data bits | $$n×m-4$$ | $$n×m-4$$ |

where nxm is the number of cells in the transmitter.

R2×2 is the states of four reference-cells.

The clock-and-rotation information transmitted by four reference-cells is to support the receiver decoding under the presence frame rate variation and the rotation of receiver.

From the states relationship between four reference-cells, the receiver identifies the rotation of the transmitter on the image for decoding. Also, the change of states of all reference-cells at the rate equaling to the optical clock rate supports the time-variant frame rate receiver decoding by identifying whether the new block coming or not.

The preamble is to start the data frame which consists of multiple data blocks. The preamble for our hidden A-QL is four data-block times long. Each block time of the preamble is specified as the matrix form as follows.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Duration | one block time | one block time | one block time | one block time |
| Preamble | A | A’ | A | A’ |

A and A’ are two inverse forms of the nxm cells matrix.

$$A\_{n×m}=\left(\begin{matrix}1 0&\cdots &1 0\\\vdots &\ddots &\vdots \\0 1&\cdots &0 1\end{matrix}\right); A'\_{n×m}=\left(\begin{matrix}0 1&\cdots &0 1\\\vdots &\ddots &\vdots \\1 0&\cdots &1 0\end{matrix}\right); $$

The preamble is also helpful in helping the receiver to distinguish how many individual cells on the transmitter.

**16.6.3 RLL encoder**

The purpose of the RLL encoder is to enhance the performance of decoding. The RLL encoder utilizes the 1/2-rate line code to generate the reference image for referring to the data image.

Each bit among the block of nxm bits at the reference image are generated by the corresponding nxm bits (including 4 clock information bits and other nxm-4 data bits) at the data image as shown in table x2. Actually, the reference image does not embed any data information but the clock information via the reference-cells.

**Table x2. Proposed line encoding table**

|  |  |
| --- | --- |
| **Binary Input** | **Code Output** |
| “0” | 0 0 |
| “1” | 0 1 |

0 state indicates the cell on the image does not change any intensity, whereas 1 state indicates the cell on the image changes.

Let $D\_{n×m}$denote the matrix of the states of nxm cells in the data image; and $R\_{n×m}$ denote the matrix of the states of nxm cells in the reference image. The matrixes are expressed as:

$$R\_{n×m}=\left(\begin{matrix}1 0&\cdots &0 1\\\vdots &\ddots &\vdots \\1 0&\cdots &0 0\end{matrix}\right); D\_{n×m}=\left(\begin{matrix}0 x\_{11}&\cdots &x\_{1 m-2} 0\\\vdots &\ddots &\vdots \\0 x\_{n1}&\cdots &x\_{n m-2} 1\end{matrix}\right); $$

Once the RLL code is applied, the RLL decoder is needed in the receiver side. Here, the XOR operator (denoted as $⊻$) is applied to every cell from the pair of two matrixes to demodulate data bits follow.

$$D\_{n×m}= D\_{n×m} ⊻ R\_{n×m}$$

**16.6.3 FEC encoder**

The FEC encoder utilizes the Reed-Solomon (RS) as outer code and Convolution Code (CC) as inner code as shown in **tables 149**-1 and **149**-2.

**Table 149-**a: Outer code (RS)

|  |  |  |
| --- | --- | --- |
| **RS\_option** | **RS description** | **RS\_rate** |
| 1 | None | 1 |
| 2 | RS(64,32) | 1/2 |
| 3 | RS(160,128) | 128/160 |
| 4 | RS(15,7) | 7/15 |
| 5 | RS(15,11) | 7/11 |
| 6 | RS(15,2) | 2/15 |
| 7 | RS(15,4) | 4/15 |

**Table 149-b**: Inner code (CC)

|  |  |  |
| --- | --- | --- |
| **CC\_option** | **CC description** | **CC\_rate** |
| 1 | None | 1 |
| 2 | CC(1/4) | 1/4 |
| 3 | CC(1/3) | 1/3 |
| 4 | CC(2/3) | 2/3 |