**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 | [November 2016] |
| Source | Trang Nguyen, Nam Tuan Le, and Yeong Min Jang (Kookmin University) |
| Re: |  |
| Abstract | Details of Resolutions regarding to the submitted Comments on D0 are suggested.  Revisions 2 on Kookmin related Sub-Clauses and Specifications are given. |
| Purpose | D0 Comments Resolutions |
| 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. |

# #1: Low-Clock-Rate OOK Amplitude Dimming

|  |
| --- |
| **Comment:**  Low-Clock-Rate OOK Amplitude Dimming |
| **Resolution**  4.5.3.1.4 OOK dimming  **4.5.3.1.5 Low-Clock-Rate OOK amplitude dimming** //we insert the new sub-clause  Since the flicker on OOK modulation is mitigated by using symmetric symbols, a number of ones and zeros in bits “1” and “0” are equal maintaining the average bright of the light source at 50%. For some Image Sensor Communication PHY modes, the clock rate of modulation is constant at a low frequency, hundreds of Hz on S2-PSK and S8-PSK PHY modes or less than 5kHz on OOK PHY modes; hence, the pulse amplitude modulation is performed to adjust the average brightness.  Figure 11 shows an example of amplitude dimming for OOK modulation. The brightness of ones is controlled to achieve the average brightness at a desired dimming level. The output dimming level is the average value of the one and the zero brightness. Compared to the OOK dimming approach by inserting compensation symbols, the data rate of Low-Clock-Rate OOK amplitude modulation approach is maintained during dimming.  C:\Users\Trang\Dropbox\00- - - - -  - - IEEEr1 Proposal\0-----for 2016 July meeti\Figures\Fig 11.png  **Figure 11 – Example of OOK amplitude dimming applied for low-clock-rate OOK PHY modes.** |

# #2: PHY A,B,C tables

|  |
| --- |
| **Comment #**  PHY A, B and C tables |
| **Resolution**  //Please replace our text (red color) into the merged tables of PHY operating modes.  **Table xx. PHY A operating modes**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Modulation** | **RLL** | **Optical clock rate** | **FEC** | | **Bit Rate** | | UFSOOK |  |  |  | |  | | Twinkle VPPM |  |  |  | |  | | Offset-VPWM |  |  |  | |  | | S2-PSK | TBD | 200Hz | Temporal repeat code  (10 symbol/sec) | Spatial code rate = 1/2 | Uncoded date rate is equal to camera frame rate. | | S8-PSK | TBD | 800Hz | Temporal repeat code  (10 symbol/sec) | Spatial code rate = 3/4 | Uncoded date rate is triple camera frame rate. | | S2+DSM-PSK | None | 800n Hz | Temporal repeat code  (10 symbol/sec) | Spatial code rate = 3/8  Additional Outer code: TBD | Uncoded date rate is triple camera frame rate. |   **Table xx. PHY B operating modes**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Modulation** | **RLL** | **Optical clock rate** | **FEC** | | **Bit Rate** | | RS-FSK |  |  |  | |  | | 3 mode  PWM/PPM |  |  |  | |  | |  |  |  |  | |  | | OOK | Manchester | 2.2 kHz | Temporal repeat code: DS=100 | Code rate = (N-4)/N  where N is the number of bits encoded per sub-packet.  (4 Ab bits overhead per sub-packet)  RS(TBD, TBD) | 60 | | 4B6B | 2.2 kHz | Temporal repeat code: DS=60 | 150 | | Manchester | 4.4 kHz | Temporal repeat code: DS=60 | 580 | | 4B6B | 4.4 kHz | Temporal repeat code: DS=60 | 700 | | FSK | None | Variable | Temporal repeat code: (10 symbol/sec) | Code rate = (K-1)/K | 10 ×(K-1)  where K is the number of bits encoded per frequency symbol. |   **Table xx. PHY C operating modes**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Modulation** | **RLL** | **Optical clock rate** | **FEC** | | **Bit Rate** | | 2D sequential 8-color code | None | 10 Hz | Spatial code rate = 252/256  RS (TBD) | | 30 ×(#\_data LEDs) | | Invisible data embedded |  |  |  | |  | | VTASC |  |  |  | |  | | PAPM |  |  |  | |  | | Kookmin Invisible Code | None | 10Hz | RS() | CC() |  | |

# #3: PHY A,B,C modes dimming

|  |
| --- |
| **Comment #**  PHY A, B, C modes dimming – text modification |
| **Resolution**  **//PHY A dimming specifications ----- Please add those text**  **9.5.2.4.4 S2-PSK dimming**  S2-PSK achieves dimming by controlling the amplitude of ones or zeros in OOK signal. Without controlling the amplitude, ones are at full brightness (totally “ON”), and zeros are at full darkness (totally “OFF”), enabling the average brightness at 50%. The configuration of ones’ amplitude allows the average brightness output at the dimmed level (<50%). Likewise, the configuration of zeros’ amplitude allows the average brightness output at the bright level (>50%). The desired dimming level is the average brightness of one and zero.  The clock rate of OOK modulation is fixed at a low frequency (200Hz for indoor or 125Hz for outdoor) throughout the transmission time and dimming control. Consequently, the amplitude controlling is performed. Hence the S2-PSK dimming is called Low-Clock-Rate OOK amplitude dimming.  **9.5.2.4.5 S2+DSM-PSK dimming**  DSM-PSK dimming is performed by controlling the “ON” time pulse width. The DSM-PSK dimming approach is the same as VPPM dimming and performed at a high clock rate (hundreds of kHz). The step of DSM-PSK dimming depends on the step of duty cycle being controlled by VPPM.  During an interval of time, VPPM dimming generates a dimmed level. After, VPPM dimming generates a bright level on the next interval of time. The process is repeated generating zeros (dimmed levels) and ones (bright levels) of S2-PSK signal.  Finally, the brightness that human eyes perceive is the average brightness of ones and zeros of the S2-PSK signal.  **//PHY B dimming specifications ----- Please add those text**  **9.5.2.5.2 FSK dimming**  // (TBD) The FSK dimming should be the same as RS-FSK dimming. Let us discuss and find out the common text for both.  // (TBD) The common FSK dimming method for ISC would be addressed in 4.5.3.1 the sub-clause. Any difference between RS-FSK dimming and Kookmin FSK would be addressed in PHY B specifications separately.  **9.5.2.5.3 Two mode OOK dimming**  The preamble symbol and data symbols are all symmetric symbols, and the average brightness of those is constant at 50%. The optical clock rate is also constant at a considerable low frequency, 2.2kHz or 4.4kHz.  Rolling shutter OOK PHY modes achieve dimming by controlling the amplitude of ones or zeros in OOK signal. The configuration of ones’ amplitude generates the average brightness output at the dimmed level (<50%). Meanwhile, the configuration of zeros’ amplitude achieves the average brightness output at the bright level (>50%). The achieved dimming level is the average brightness of one and zero.  **//PHY C dimming specifications ----- Please add those text**  **9.5.2.6.1 2D-sequential color code**  **(TBD)**  Dimming is not supported for the eight-color code sequential communications.  Any change in color inside the code area interferences to the red, green, and blue channel of data communication. In contrast, any change in the outside area of the code does not affect any to the link performance if the code is successfully detected.  **9.5.2.6.5 Kookmin invisible code**  **(TBD)**  The change in color background does not affect to the data communication. Dimming is hence supported by controlling the background color. The delay time of dimming control should be synchronized to the symbol rate of transmission. |

# -- # MAC frame structure

# #4: PHY constants and attributes table

|  |
| --- |
| **Comment #**  PHY constants and attributes table |
| **Resolution**  **// Empty**  **TBD** |

# #5: PHY A specifications

|  |
| --- |
| **Comment #**  PHY A specifications |
| **Resolution**  **14.3 S2-PSK**  **14.3.1 S2-PSK Encoder**  **14.3.1.1 Bit-to-symbol mapping**    **Figure 261 –(Left) A pair of LEDs transmitter, and (right) An example of bit-to-symbol mapping**  **Table x: Bit-to-symbol mapping table**   |  |  |  | | --- | --- | --- | | **Bit** | **LED-1 symbol** | **LED-2 symbol** | | 0 | 1 0 1 0 … 1 0 | 1 0 1 0 … 1 0 | | 1 | 1 0 1 0 … 1 0 | 0 1 0 1 … 0 1 |   The pair of LEDs collaborates to transmit a bit at an interval of time. The bit-to-symbol mapping to control a pair of LEDs is as shown in table x.  The selection of the optical clock rate is just to be non-flicker. However, a lower optical clock rate gains a lower probability of bad-sampling which caused by capturing at the pulse switching time. Usually, the constant value of the optical clock rate is chosen at 200Hz for the indoor environment and 125Hz for the outdoor environment.  The symbol to each LED is a multiple-times repetition of a symmetric Manchester symbol (i.e. multiple times repetition of 01 or 10). After mapping, the symbol rate (equivalent to bit rate) is constant and chosen to be no greater than the camera frame rate (e.g. 10 symbol-per-second) to ensure that at least a transmitted symbol is sampled once.  **14.3.1.2 Decoding principle (applied for a random sampling)**  At a random sampling time ts, the camera captures the states of LED-1 and LED-2, x1(ts) and x2(ts) respectively. A bit is de-mapped as follow  bit = ~XOR (x1(ts); x2(ts));  **14.3.1.3 Decoding example**    **Figure 262 –An example of decoding**  **Figure 262** illustrates an example of decoding. The decoding result is non-affected by the values of captured states of the LEDs but by the comparison (i.e., XOR operator). This means a receiver does not need to know which LED is a reference LED and which one is data LED; data is output from a comparison of two captured states of LEDs. Consequently, the decoding is suitable for a random sampling under the presence of frame rate variation.  **14.3.1.4 Line Coding**  **// (TDB)** To be used alone, S2-PSK requires a line coding. We wish to update later.  **14.3.2 S2-PSK Error Correction**    **Figure 263 –Bad-sampling error in S2-PSK modulation**  Even if the optical clock rate (e.g. 200Hz) is much lower than the capturing speed of an image (e.g. several kHz or MHz), a bad-sampling may still occur due to the capture on the transition of OOK signal. Forward error correction is required to correct the bad-sampling error.  The spatial approach that has been used to encode a bit by modulating a pair of LEDs can be considered as a type of partial error correction. Moreover, a temporal repetition code is proposed, allowing the receiver correcting a limited amount of error caused by the presence of bad-sampling. Usually, a temporal repetition holds the bit rate (e.g. 5bps or 10bps) at much less than the camera frame rate (e.g. ~30fps).  **14.3.2 S2-PSK dimming Support**  S2-PSK dimming is achieved by amplitude modulation as described in the sub-clause 4.5.3.1.5 Low-Clock-Rate OOK amplitude dimming. The benefit of amplitude dimming is that the data rate of transmission is maintained while performing dimming. In contrast, it requires hardware support.    **14.4 S2+DSM-PSK**  **14.4.1 DSM-PSK**  **14.4.1.1 S8-PSK**  **14.4.1.1.1 S8-PSK Encoder**  A group of four-LEDs is used to transmit a phase which encoded by 3-bits data. Herein, a spatial phase, S\_Phase (of an LEDs-group) is determined by a waveform created from a set of four-sates of LEDs on the group at a time slot. **Figure and table** below show the determination of the spatial phase value being used for S8-PSK encoding later.   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | Spatial-Phase Determination Table   |  |  | | --- | --- | | **4-States waveform** | **S\_Phase** | | 1000 | 1 | | 1100 | 2 | | 1110 | 3 | | 1111 | 4 | | 0111 | 5 | | 0011 | 6 | | 0001 | 7 | | 0000 | 8 | |   **Figure 264 – (Left) A group of LEDs waveforms; and (right) Spatial-Phase Determination Table**  We have two groups of LEDs being used to encode 3 bits each time. For a simple decoding, the waveforms of four LEDs in a group are maintained as shown in figure 264 (left), called the reference group. The phases of waveforms of the other four LEDs in the other group (called the data group) vary on the value of 3 bits. The **Global Phase Shift** is defined by the shift value of spatial phases between the data group and the reference group at any point in time.  The mapping from 3 bits to the value of the global phase shift is as shown in **table x below.**  **Table x. Encoding table**   |  |  | | --- | --- | | 3-bits  **Input** | Global Phase Shift  **Output** | | 000 | 0 | | 001 | 1 | | 010 | 2 | | 011 | 3 | | 100 | 4 | | 101 | 5 | | 110 | 6 | | 111 | 7 |     **14.4.1.1.2 S8-PSK Decoder**  At the sampling time, four LED-states of the reference group and the data group are captured. The S\_Phase values of those groups are determined from the **Spatial-Phase Determination Ta**ble.  The shift value between S\_Phase values of those groups is calculated as follow:  S\_Phase\_Shift = S\_Phase(data) – S\_Phase(reference)  The de-mapping from S\_Phase\_Shift into 3 bits is presented following two possible cases.  **Case 1:** Decoding under none-presence of bad-sampling     |  |  | | --- | --- | | (S\_Phase\_Shift)  **Input** | 3-bits  **Output** | | 0 | 000 | | 1 | 001 | | 2 | 010 | | 3 | 011 | | 4 | 100 | | 5 | 101 | | 6 | 110 | | 7 | 111 |     **Figure 265 - (left) S8-PSK none-bad-sampling Decoding; and (right) A decoding example**  **Case 2:** Decoding under presence of *bad-sampling*  A bad-sampling generates a presence of an unclear state (x\_state). A new determination of S\_Phase value under the presence of x-state is a shown as the re-defined Spatial-Phase table as follow.  Spatial-Phase Determination Table (Re-defined with x-state)   |  |  | | --- | --- | | **4-States waveform** | **S\_Phase** | | 1x00 | 1 | | 11x0 | 2 | | 111x | 3 | | x111 | 4 | | 0x11 | 5 | | 00x1 | 6 | | 000x | 7 | | x000 | 8 |   The decoding is the same as presented. After the S\_Phase of each LEDs-group is determined, the shift value between S\_Phase values of those groups is calculated. And then, 3 bits are mapped from the value of S\_Phase\_Shift.    **Figure.266. An example of decoding under the bad-sampling condition**  **14.4.1.1.4 S8-PSK Error Correction**  The spatial approach encodes 3 bit by modulation two groups (each has 4 LEDs) can be considered as a type of partial error correction. The spatial approach allows the camera decoding successfully even under the presence of the bad-sampling due to long exposure time. The proposed correction of bad-sampling does not reduce the data rate.  Also, a temporal repetition code is employed, allowing the receiver correcting a limited amount of error caused by the presence of bad-sampling. The clock rate to output a symbol of 3-bits is at 10 Hz, enabling the majority voting scheme on a typical 30fps camera.  **14.4.1.1.5 S8-PSK Dimming**  S8-PSK dimming is achieved by amplitude modulation as described in the sub-clause 4.5.3.1.5 Low-Clock-Rate OOK amplitude dimming. The benefit of amplitude dimming is that the data rate is maintained while dimming is performed. In contrast, it requires hardware support.  **14.4.1.2 DS8-PSK**  **14.4.1.2.1 DS8-PSK Encoder**  A group of eight-LEDs is used to transmit a phase which encoded by 3-bits data. Herein, a spatial phase, S\_Phase (of a LEDs-group) is determined by a set of four-sates of LEDs on the group. **Figure and table** below show the determination of the spatial phase value being used for DS8-PSK encoding later.    **Figure 269 –** **Spatial-Phase Determination Tables at different dimming levels**  We have a pair of groups of LEDs being used to encode 3 bits each time. For a simple encoding, the waveforms of eight LEDs in a group are unchanged, called the reference group. The phases of waveforms to drive the other eight LEDs in the other group (called the data group) are controlled by the value of 3 bits Input. The **Global Phase Shift** is defined by the shift value of spatial phases between the data group and the reference group at any point in time.  The mapping from 3 bits to the value of the global phase shift is the same as the mapping table which has presented for S8-PSK**.**  **14.4.1.2.2 DS8-PSK Decoder**  The decoder for DS8-PSK is the same as for S8-PSK. After being captured, eight LED-states of the reference group and the data group are mapped into the S\_Phase values. And then, the shift value between S\_Phase values of those groups is also calculated, S\_Phase\_Shift. The de-mapping from S\_Phase\_Shift Input into 3bits Output is also the same as S8-PSK.  For the condition of bad-sampling, the spatial-Phase Determination tables are also re-defined as S8-PSK did.    **Figure 270 –Re-defined Spatial-Phase Determination Tables at different dimming levels with x\_state**  **14.4.1.2.3 DS8-PSK Error Correction**  The spatial approach encodes 3 bit by modulating a pair of LED-groups (each has 8 LEDs) can be considered as a type of partial error correction. The spatial approach allows the camera decoding successfully even under the presence of the bad-sampling due to long-exposure time. The proposed correction of bad-sampling decoding does not reduce the data rate.  Also, a temporal repetition code is employed, allowing the receiver correcting a limited amount of error caused by the presence of bad-sampling. The clock rate to output a symbol of 3-bits is at 10 Hz, enabling the majority voting scheme on a typical 30fps camera.  **14.4.1.2.4 DS8-PSK Dimming**  DS8-PSK dimming is supported in steps of 1/8 (12.5%). The dimming control is performed by the pulse width, not the amplitude. DS8-PSK dimming is a sub-set of VPPM dimming.  To support dimming during transmission, the receiver needs to select a proper table for later decoding among seven S\_Phase Determinationtables and seven re-defined S\_Phase Determinationtables based on the dimmed level. The decoding procedure is as shown as follow:   |  | | --- | | **Decoding procedure under dimming condition:** | | **Step 1:** Choose the proper S\_Phase decoding Table (among seven tables) according to the dimming level:   * + Dimming level (or under the presence of x\_state)   + Select the proper S\_Phase Determination table among 14 tables. | | **Step 2:** Map with the selected decoding table to find S\_Phase(data); S\_Phase(reference) and S\_Phase\_Shift  **Input:** The discrete waveforms of a 8-LEDs groups (a reference group and data groups)  **Output:** Spatial Phase Values   * + S\_Phase(reference)   + S\_Phase(data)   + S\_Phase\_Shift = S\_Phase(data) - S\_Phase(reference) | | **Step 3:** Data decoding using Phase-to-Bits table  **Input:** S\_Phase\_Shift  **Output**: 3 data bits |   **14.4.1.3 Twinkle VPPM**  **14.4.1.3.1 Twinkle S2-PSK and DS8-PSK Encoder**  The DSM-PSK dims the light sources at the low dimming level and the high dimming level evenly, therefore generates an AM signal at a low frequency of 200Hz (or 125Hz). The AM signal is modulated following the encoding rule of S2-PSK. The bit rate for AM signal is usually 10Hz; therefore any camera has the frame rate at no less than 20fps can demodulate the AM signal.  For a dual-camera receiver system, the twinkle signal can be demodulated as below:   * A low frame rate camera (i.e. low-cost camera) is to detect the S2-PSK signal.   + Can be either a global or a rolling shutter camera   + Can be either a slow exposer or a quick exposer camera. A higher shutter speed camera is better for removing environmental noise and detecting LEDs. * A high-speed camera (i.g. a global shutter and high frame rate camera) is to decode data from the DS8-PSK signal.     **Figure 2xx –Hybrid modulation schemes for dual M-LEDs and dual cameras system**    **Figure 2xx – Hybrid modulation schemes for LED Signage transmitter**  **14.4.1.3.2 Twinkle S2-PSK and DS8-PSK Error Correction**  The error correction for high-speed data link using DS8-PSK modulation is the same as the presented.  For S2-PSK, the temporal repetition code is applied. The frequency of AM signal is multiple times less than the camera frame rate to ensure that the majority voting is performed.  **14.4.1.3.2 Twinkle S2-PSK and DS8-PSK Dimming Support**  Dimming is supported by adjusting the low dimmed level and high dimmed level of DS8-PSK scheme to output a desired dimming level.  Output dimming level = ½ (low dimmed level + high dimmed level) |

# #6: PHY B Specifications

|  |
| --- |
| **Comment #:**  PHY B specifications |
| **Resolution**  **15.2 Compatible M-FSK Modulation**  **15.2.1 Reference Architecture**  The M-FSK modulation scheme is applied to a system as shown in **figure 288.** An LED panel is to transmit frequency symbols which modulated data, and a rolling shutter camera is to receive data. The system considers applying for various types of rolling shutter image sensors which may have differences in frame rates variation, sampling rates, and rolling exposure times.    **Figure 288– Reference Architecture for clock transmission in frequency modulation**  **15.2.2 M-FSK Encoder**  A bit of clock information (Ab) is inserted at the beginning of a data packet to become a packet of bits. The packet of bits is then mapped into a frequency symbol. Figure 289 illustrates an example of encoding procedure to map data into frequency symbol.    **Figure 289– Data and clock information merging in frequency domain**    **Figure 290– Frequency band used for M-FSK Encoder**  Typically, a rolling shutter camera, such as Smartphone camera has a fixed shutter speed at 8kHz. The cut-off frequency of the camera is the upper limit value for the frequency band. The frequency band is also lower limited by the eye cut-off frequency to ensure being invisible to human eye.  The frequency separation, however, has limited by the sampling rate of the image sensor. The sampling rate here is the pixel read-out rate of the image sensor, therefore depending on the selected resolution as well. The frequency separation is chosen to be wide enough for the commercial cameras, such as Smartphone cameras being able to distinguish the transmitting in-band frequencies. **Figure 290** illustrates the selected frequency band and frequency separation.  The width of the frequency band and the separation of frequencies together decide the number of frequencies for transmission. The bit rate is calculated as the function of the frequencies number being used.  **15.2.3 CM-FSK asynchronous transmission**    **Figure 292– Data packet structure**  To support a varying frame rate camera can demodulate data from frequency symbols, an asynchronous transmission is proposed. An asynchronous bit (Ab) represents the clock information of a data packet allowing the camera distinguishing a repeated frequency symbol and a new frequency symbol coming without knowing the sampling times of those symbols.  **15.2.3 32-FSK Modulation**  32-FSK encodes a symbol of data, including one asynchronous bit and four data bits, into a frequency among selected 32 frequencies. The structure of symbol and the bits-to-symbol mapping table are as shown in figure 295.  **C32-FSK encoding table**   |  |  | | --- | --- | | Baud symbol | Frequency symbol | | fSF | **fo** | | 00000 | f1 | | 00001 | f2 | | . . . | . . . | | 11110 | f31 | | 11111 | f32 | | f'SF | f33 |   **Symbol structure**   |  |  | | --- | --- | | Bits: 1 | 4 | | Ab | Data |   **Figure 295– Symbol structure and 32-FSK encoding table**  Beside 32 frequencies are selected to encode a symbol of five bits, two additional frequencies are used as preamble symbols. The calculation of data frequencies and preamble frequencies is as follow:   * Data frequency: fi = fSF + i.∆f (i=1; 2;…; 32) * Preamble frequency: f’SF = fSF + 33.∆f   where ∆f is the frequency separation value; fSF and f’SF are two preambles.    **Figure 296– Frequency allocation for 32-FSK**  The allocation of frequencies is as shown in figure 296. In the receiver side, the determination of two preamble frequency values allows the calculation of the other 32 data frequencies for decoding.  **15.2.4 64-FSK Modulation**  64-FSK encodes a symbol of data, including one asynchronous bit and five data bits, into a frequency among selected 64 frequencies. The structure of symbol and the bits-to-symbol mapping table are as shown in figure 297.  **64-FSK encoding table**   |  |  | | --- | --- | | Baud symbol | Frequency symbol | | fSF | **fo** | | 00000 | f1 | | 00001 | f2 | | … | | | 11110 | f31 | | 11111 | f32 | | f'SF | f33 | | 010000 | f34 | | 010001 | f35 | | … | | | 111110 | f64 | | 111111 | f65 |   **Symbol structure**   |  |  | | --- | --- | | Bits: 1 | 5 | | Ab | Data |   **Figure 297– Symbol structure and 64-FSK encoding table**  The 64-FSK frequency band is a twice extension of the 32-FSK frequency band. The first 32 data frequencies and two preamble frequencies are the same values as addressed in the 32-FSK modulation. Also, the other 32 frequencies are additionally allocated on the right side of the 32-FSK modulation band to achieve a higher capacity of data per frequency symbol.  **15.2.5 Hybrid Frequency-Phase Shift Keying**  The M-FSK is achieved by allocating different frequencies on the selected band. PSK modulation is additionally used on the hybrid modulation to tackle the higher link capacity. The number of phases is two (zero-phase and inverse-phase) to modulate a square wave.  Figure 297 shows a design of LED lighting in using pairs of LEDs for transmitting a hybrid signal of 2-PSK and M-FSK. By using a pair of LEDs, 2-PSK is additionally achieved by modulating the phase relationship between the two frequency signals to drive a pair of LEDs. Bit “0” is mapped by modulating the same phase signals to a pair of LEDs, while bit “1” is mapped by modulating the inverse-phase signals to the pair. Both LEDs on the pair are at the same frequency, enabling fully advantage of M-FSK modulation.  At a distance that a camera cannot distinguish a frequency from the others due to the limited size of LED on the captured image, the camera is still able to demodulate the 2-PSK signal by comparing the states of two LEDs of a pair. The design of LED-lighting as shown in figure 297 along with the hybrid 2-PSK/M-FSK earn an advantage in communication distance.    **Figure 297– A lighting system design using pairs of LEDs for hybrid PSK/FSK modulation**  **15.3 C-OOK**  **15.3.1 C-OOK Encoder**  A packet of data is modulated using OOK modulation. The optical clock rate is at 2.2 kHz or 4.4 kHz.  The data packet structure is as shown in figure 299. A packet consists of multiple similar data sub-packets to avoid missing data in between the gap time of adjacent images. The number of repetition depends on the communication mode specified later (see **section x)**.  Every Data Sub-packet (DS) has its preamble symbol and payload. The definition of preamble symbol depends on the kind of line coding on the payload. Hence, the preamble for Manchester coded payload is shorter than that of 4B6B coded payload to reduce the amount of overhead. Table 142 shows the symbol definitions.  The payload section of a DS is fragmented into three subsections. The middle subsection is for data, while the head and the tail subsection are for the clock information (Asynchronous bits). The amount of asynchronous bits (Ab) at each subsection is fixed at 2 bits, while the amount of data on the body of payload is varied, and specified upon the communication mode (see **section x)**.    **Figure 299– Data packet structure**  T**able 142: Definition of SF symbol (Preamble symbol) and sub-packet structure**   |  |  |  |  | | --- | --- | --- | --- | | **SF symbol (preamble)** | **DS Payload** | | | | **Ab** | **Data** | **Ab** | | 011100 | 2 bits | Manchester coding | 2 bits | | 0011111000 | 4B6B coding |   **15.3.2 C-OOK Asynchronous Decoder**  To demodulate the entire data sub-packet DS, the distance from a camera to the LED transmitter should be close enough. Figure 301 shows the relationship between the amount of data being captured by the camera and the distance from the camera to the LED transmitter.    **Figure 301– Decoding scenario**  From the figure 301, the maximum distance achieved is the distance at which the camera gets the amount of data equal to the amount of the sub-packet.  **Decoding case 1: Fuse incomplete parts of a sub-packet into a complete one**  At this distance far, the distance d1 as shown in figure 301, the camera detects the preamble symbol and then demodulates the amount of data enough for a sub-packet; however, the uncertainty whether the forward part and the backward part counted from the position of the preamble belong to a sub-packet or not is problematic. The problem of a small amount of data also happens at a shorter distance when the transmitted sub-packet is long.  Asynchronous bits representing the clock information of the packet are used for the asynchronous decoding algorithm in this case.    **Figure 302– Decoding algorithm at a far distance**  Figure 302 illustrates the decoding algorithm to recover a packet of data from the forward part and the backward part of an image when the size of LED is small in the captured image. By observing the values of an asynchronous bit before and an asynchronous bit after the preamble SF, two statements of fusing those two parts of image are addressed:   * Case 1- *Inter-frame data fusion*: Fusing two sub-parts of a packet at two different images into a complete packet.   This type of data fusion is applied in case two Ab on an image are different.   * Case 2- *Intra-frame data fusion*: Recovering a complete packet from an image.   This type of data fusion is applied in case two Ab on an image are similar.  **Decoding case 2: Combination of Data Fusion and Majority Voting**  When the camera goes closer to the LED transmitter, the amount of data being captured per image is greater than that of a sub-packet. Therefore, the extra amount of data is used for correcting the possible error by applying a majority vote.  At distance d2 on figure 301, the amount of data equivalent to two sub-packets is captured. The majority voting is used in this case to correct the error throughout the entire sub-packet.  Figure 303 shows an experimental example of decoding under *Intra-frame data fusion.* The extra data after fusion a sub-packet is used for correcting the error by voting.  Assume that the camera frame rate may vary but be greater than the packet rate of transmission. Therefore, any extra data after fusion is useful for the error correction by grouping multiple images which belong to a sub-packet to vote. The voting is on the amount of data grouped from all of the forward parts and backward parts of images as well as extra data.    **Figure 303– An example of decoding employing intra-frame fusion along with error correction.**  **15.3.3 Missing packet detection on frame rate drop**  The decoding algorithm in sub-clause 15.3.2 was proposed under the assumption of the receiver frame rate greater than the transmitting packet rate. In some circumstance, the frame rate may drop to less than the packet rate, causing to an entire packet is missed. The detection of the missed packet is proposed herein for a later process.  The core idea comes from the usage of asynchronous bits inserted into the payload of every sub-frame. Two bits (Ab1Ab2) are inserted at the forward and the backward of the body payload as shown in Figure 303. Those two bits together bring the clock information of the sub-packet and being modulated as shown in Figure 304.   |  |  |  |  | | --- | --- | --- | --- | | **Preamble (SF)** | **Ab (front)** | **Body payload** | **Ab (rear)** | |  | 2 bits (Ab1Ab2) | Variable | 2 bits  (Ab1Ab2) |   **Figure 303– Data Sub-Packet Structure**    **Figure 304–Asynchronous bits transmission and a missed-symbol Detection**  Ab1 and Ab2 are square signals. Ab1 changes from zero/one into one/zero every time of single data packet, while Ab2 changes every time of two data packets.  The combination of two Ab, Ab1 and Ab2, generates four different values, 00 01 10 and 11. Therefore, the usage of those two Ab enables the detection of 2 missed packets continuously. It means the detection of missed packets is 100% successful for any frame rate drop to no less than 1/3 of the packet rate. For example, a packet rate at 10Hz with 2 Ab allows the frame rate drops to 3.3fps while all the missed packets are detectable.  **15.3.3 Packet Structure Specification Modes**  The maximum distance of transmission depends on the size of data sub-packet. A shorter length of the DS permits a longer distance at maximum. Here, the length of DS in time is specified by the value of sub-packet rate. Table below presents some parameters for OOK modes. More detail of the sub-packet structure is given in table xx.  **Table – Parameters Specifications**   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | **Mode 1** | **Mode 2** | **Mode 3** | **Mode 4** | | Optical clock rate | 2.2 kHz | 2.2 kHz | 4.4 kHz | 4.4 kHz | | Sub-Packet rate | 100 DS/s | 60 DS/s | 60 DS/s | 60 DS/s | | Packet rate | 10 packet/s | 10 packet/s | 10 packet/s | 10 packet/s | | Bit rate  (error corrected) | 60 bps | 150 bps | 580 bps | 700 bps | | Max distance (limited by exposure) | 10% of (image-frame interval) | 16.7% of (image-frame interval) | 33% of (image-frame interval) | 33% of (image-frame interval) |   **Table – Sub-Packet Structure Specifications**   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | **Mode 1** | **Mode 2** | **Mode 3** | **Mode 4** | | Preamble (SF) | 6B | 10B | 6B | **10B** | | Ab (front) | 2B | 2B | **4B** | **4B** | | Payload (body) | 8 bits | 13 bits | 33 bits  (24B) | 41 bits  **(62B)** | | Ab (rear) | 2B | **2B** | **4B** | **4B** | |

# #7: PHY C Specifications

|  |
| --- |
| **Comment #**  PHY C specifications |
| **Resolution**  **16.1 2D-sequential color code**  A reference architecture of the 2D-sequential color code is shown in Figure x. The system considers two types of the 2D code for sequential transmission: (i) existing QR code; and (ii) a new 2D color code. The re-use of QR code for sequential transmission requires a sequential communication protocol. Besides, the proposed 2D color code is to minimize the overhead in communications and speed-up the processing on a receiver.  Figure 2--MIMO sequential-- - detail  **Figure x - Reference architecture for 2D-sequential color transmission system**  **16.1.1 A new 2D color code design**  Two dimensions design of color code for sequential transmission is shown inFigure x. Especially four LEDs at four corners of the code are called reference LEDs to transmit reference signals. The other LEDs are to transmit data. All the LEDs are surrounded by four high-gradient lines to differentiate the code area and the outside area.    **Figure x - An example design of 16x16 LEDs transmitter**  The purposes of four reference LEDs are (i) to transmit clock information to help a varying frame-rate receiver in performing asynchronous decoding; (ii) to detect and remove the rolling shutter affected images; and (iii) to help a receiver decoding under rotation.  The purposes of four high gradient lines surrounding are (i) to help the detection of the code area; and (ii) to mitigate perspective distortion on the receiver.  **16.1.2 2D color code Encoder**  The clock information is transmitted through four reference LEDs while data is transmitted through all the other LEDs. Figure x shows the change of clock is synchronized with the time a block of data is clocked out.  Table x shows the encoding table for reference LEDs and data LEDs.  C:\Users\Trang\Dropbox\00- - - - -  - - IEEEr1 Proposal\0-----for 2016 July meeti\Figures\Fig. 18.png  **Figure x – Clock information controlling the block of data output (Redrawn)**  **Table x – 8 colors encoding true table**   |  |  |  |  | | --- | --- | --- | --- | | **Reference LEDs** | Red channel | 0 and 1 evenly | | | **Data LEDs** |  | data bit =“0” | data bit = “1” | | Red channel | 0 | 1 | | Green channel | 0 | 1 | | Blue channel | 0 | 1 |   After encoding, each of channel red, green or blue has a data bit. Together 3 bits of data is mapped into a color is shown in table x. 3 bits are transmitted per LEDs each time.  Four reference LEDs transmit the clock signal through red channel.  **Table x – 8 colors encoding true table**   |  |  | | --- | --- | | **3 bits**  **Input** | **Color**  **Output** | | 000 | Black | | 100 | Red | | 010 | Green | | 001 | Blue | | 110 | Yellow | | 101 | Magenta | | 011 | Cyan | | 111 | White |   **16.1.3 QR Code Encoder**  QR code has no reference LED. Therefore, the insertion of the clock information is performed before mapping data into color. The demodulator de-maps the block of data and then splitting the data block into the clock information and data section.  The mapping from data into color among eight colors is the same as for 2D color code encoder.  The QR code interface follows the existing standard which has published years ago. For sequential communications, the type of error correction is chosen at the lowest level to reduce overhead.  **16.1.2 2D-color-code Decoder**  **16.1.2.1 Perspective Mitigation**  The LEDs matrix created from 4-corner positions Ai(w,h) is the main idea to mitigate perspective distortion. Figure x shows an example of perspective distortion mitigation. The procedure for mitigating distortion is as two following steps:  EMB000022ec35b8  **Figure x - LEDs extraction matrix using line detection under perspective distortion**  **Step 1**: 4-Edges detection using image processing  Edges are detected by using Hough transform  The position of 4 corners and matrix positions of LEDs  **Step 2:** 16x16 LED-positions Matrix forming  **Input:** 4-corner positions Ai(w,h)  **Output:** 16x16 matrix of LED-positions   |  |  | | --- | --- | | Fig | | | (a) Rotating angle is 0 at phase 0 | (a) Rotating angle is 0 at phase π |   **16.1.2.2 360-degree Rotation Decoding**  **Figure x – Rotation mitigation**  To help a camera in decoding under the rotation, the reference LEDs are gain used.  At any time, a state of a reference LED is always different from the other three. The rotation angle is identified easily by checking the reference LEDs and their states.  Previously, a red channel was used to transmit the clock information. Here, another channel, such as blue channel, is applied to transmit a signal that allows a receiver in identifying the presence of rotation.  **16.1.2.3 Rolling effect detection and cancellation**  The rolling shutter mechanism causes a problem in capturing multiple LEDs. All LEDs are synchronized by the clock; however, the vertical rolling operation delays the sampling time of a lower LED compared to the sampling time of an upper LED. As a consequence, on the same image, LEDs may be captured on different time clocks. The rolling affected image requires to be detected for cancelling the error.  Figure x shows an example of the image which is rolling affected by using the reference LEDs. The detection is still correct under rotation.  EMB000022ec35cd  **Figure x –Rolling Effect Detection for Removal**   |  | | --- | | **Rolling effect detection algorithm:** | | **If** (Mk != 0 && Mk != 1)  **then** (the image has rolling effect) |   whererepresents the matrix of reference LEDs on the captured image.  **16.1.2.4 Varying frame rate resolution**  Assume that a camera frame rate is no less than the clock rate of transmission to ensure every block of data is sampled at least once. Herein, the identification of images those are sampled on the same block of data is necessary to group those images.  The grouping images is proposed by using the clock information (Asynchronous bit, Ab) transmitted through the reference LEDs. All adjacent images those have Ab=1 are grouped for voting a block of data; and all those have Ab=0 are grouped together for voting another block of data.  Figure x shows an example of proposed scheme. Ab bit is used to group images those belong to a block of data and then all grouped images are voted.  EMB000022ec35d0  **Figure x –Asynchronous decoding**  **16.1.2.5 Error correction**  EMB000022ec35dd  **Figure x –Spatial Convolution coding (channel coding)**  A temporal error correction is applied. The block rate is 10 block/s, much less than the frame rate of camera to ensure that every block of data is sampled more than once. The majority voting of all images those sampled on the block of data is to correct the error.  A spatial Error Correction Coding is considerd as a channel coding for the noisy channel. Convolutional coding (CC) is TBD. |

## **16.6 Kookmin Invisible Mode** –

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

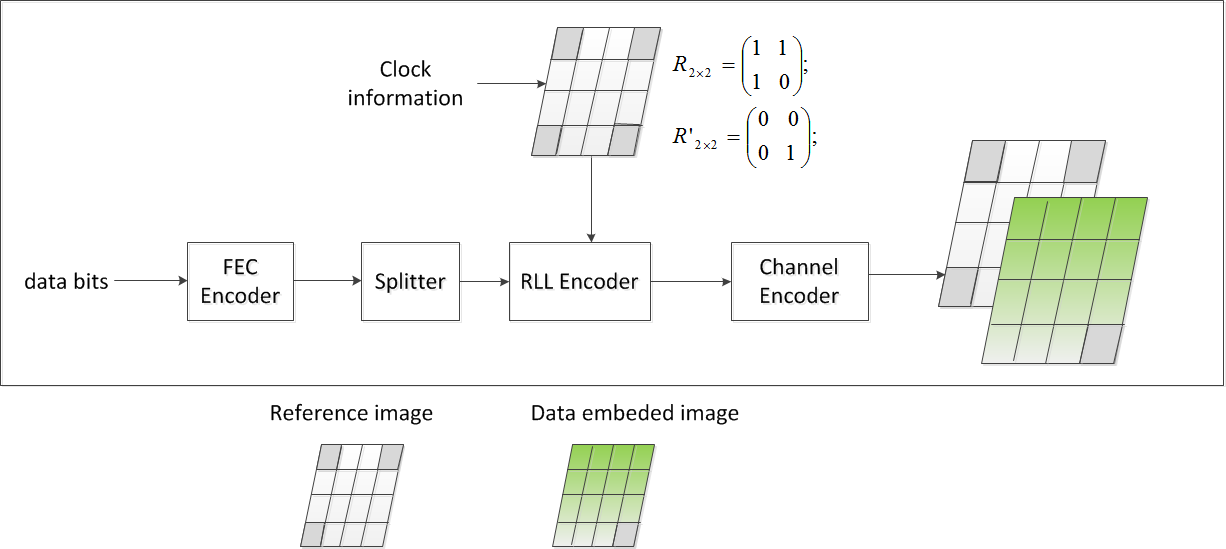
The **Hidden Asynchronous- Quick Link (Hidden A-QL)** operates different PHY modes as shown in table 148.

**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 |
| 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 at  reference-cells |  |  |  |  |
| Data bits |  | |  | |

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.

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 denote the matrix of the states of nxm cells in the data image; and denote the matrix of the states of nxm cells in the reference image. The matrixes are expressed as:

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.

**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 |