**IEEE P802.15**

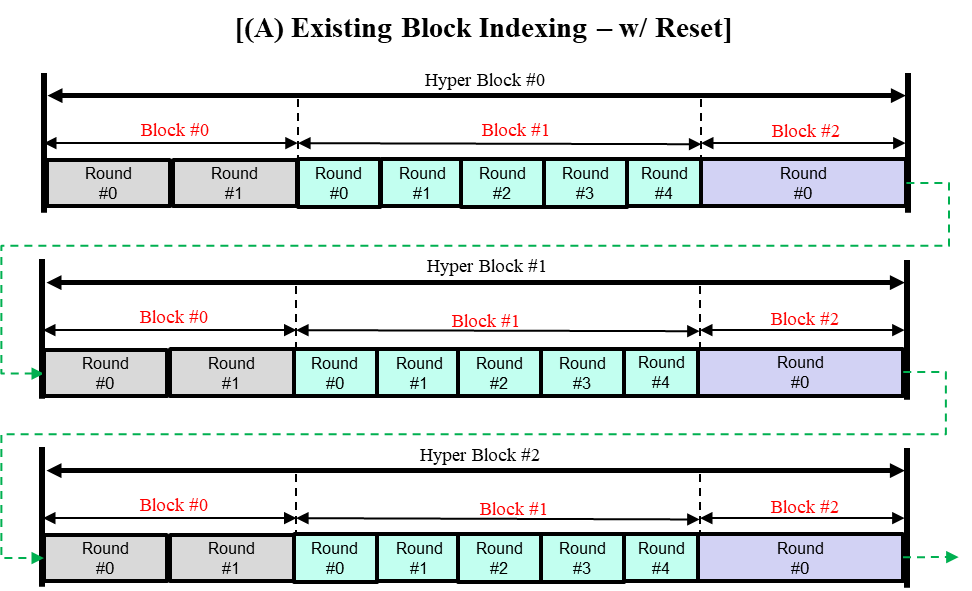
**Wireless Personal Area Networks**

|  |  |  |
| --- | --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) | |
| Title | **Proposed changes for Hyper block indexing**  **(CID 139, 144, 145, 500)** | |
| Date Submitted | May 16 2024 | |
| Sources | Youngwan So (SAMSUNG Elec.)  [youngwan.so@samsung.com](mailto:youngwan.so@samsung.com) |  |
| Re: |  | |
| Abstract |  | |
| Purpose | To propose changes for suggested comments regarding hyper block for “P802.15.4ab™/D (pre-ballot) C Draft Standard for Low-Rate Wireless Networks” . | |
| Notice | This document does not represent the agreed views of the IEEE 802.15 Working Group or IEEE 802.15.4ab Task Group. It represents only the views of the participants listed in the “Sources” field above.It is offered as a basis for discussion and is not binding on the contributing individuals. The material in this document is subject to change in form and content after further study. The contributors reserve the right to add, amend or withdraw material contained herein. | |

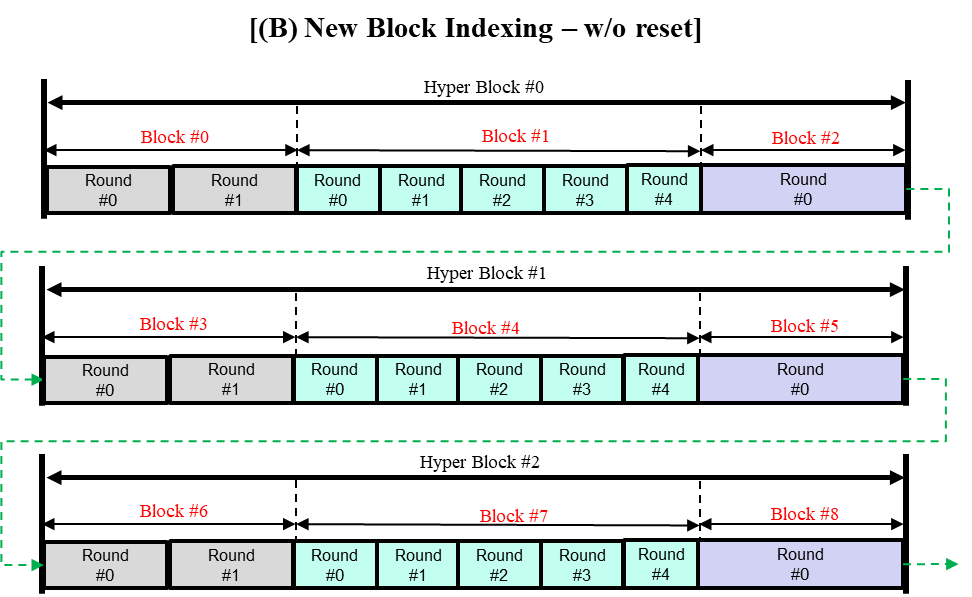
Rev 0: Address Hyper block indexing related comments.

1. **Introduction**

In Hyper Block mode, the Hyper Block index is ever increasing but Block Index inside a Hyper Block is reset to zero whenever new Hyper Block starts as Figure (A) below. But this makes some problems as the index number will have a periodicity, and eventually it may bring security problems.



Therefore, in this contribution we suggest change of block indexing in a way that Block Index inside a Hyper Block DOES NOT reset but increase continuously even it meets new Hyper Block start like Figure (B) below. And we also propose actual changes on the text.



1. **Motivation**

There may be various ways to solve the problem. But we think just allowing Block Index in Hyper Block mode to increase as it does in block-based mode in current specification is the most convenient, so we made Block Index to increase continuously rather than inventing many new solutions.

1. **Affected CIDs**

If we adopt this changes regarding Hyper Block indexing, the following CIDs (from DCN0112) are supposed not to be issue anymore.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Index#** | **Pg** | **Sub-Clause** | **Ln** | **Comment** | **Proposed Change** | **Disposition** |
| Benjamin Rolfe | 139 | 25 | 9.2.12 | 13 | It is possible (e.g. when hyper-block mode is used) for ranging slot, round and block to repeat, and so thus the frame counter value can repeat. This is used (static?) source EUI to form the nonce (9.3.2.4), which should not repeat for a given key. | Clarify how repeating the same value of a nonce is prevented in this processing | Revised |
| Benjamin Rolfe | 144 | 26 | 9.2.13 | 13 | It is possible (e.g. when hyper-block mode is used) for ranging slot, round and block to repeat, and so thus the frame counter value can repeat. This is used (static?) source EUI to form the nonce (9.3.2.4), which should not repeat for a given key. | Clarify how repeating the same value of a nonce is prevented in this processing | Revised |
| Benjamin Rolfe | 145 | 27 | 9.3.2.4 | 5 | Note. In hyper-block mode the block index can repeat. The slot index and round index repeat in every block. This can result in repeating the nonce. | Add to note: When using hyper-block mode will result in repeating a nonce and so key values need to be updated for Hyper Block boundary or the fabric of the universe will unravel due to nonce repetition. | Revised |
| Alex Krebs | 50 | 27 | 9.3.2.4 | 1 | Uniqueness of Nonce not guaranteed for Hyperblock Mode 10.13.3.5 | Clarify how/if encryption applies/does not apply to Hyperblock mode. Alternatively, change 10.13.3.5 to clarify that Hyperblock mode must not use Compact frames. | Revised |

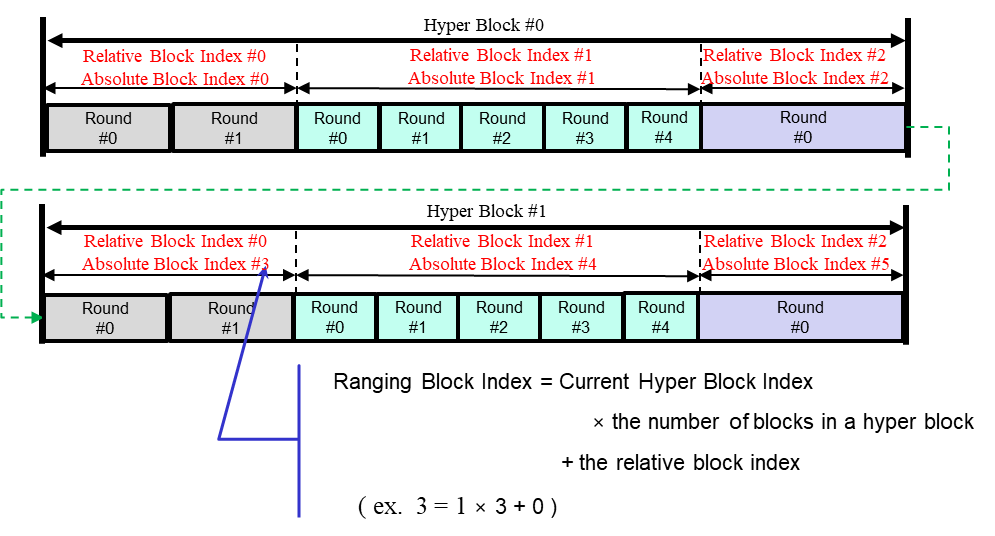
1. **Basic Idea**

In Hyper Block mode,

1. we let HBS IE deliver Block Index which is RELATIVE within a Hyper Block.
2. then we can calculate Block Index which is ABSOLUTE over whole Hyper Blocks, from simple mathematics;

Ranging Block Index = Current Hyper Block Index × the number of blocks in a hyper block

+ the relative block index

****

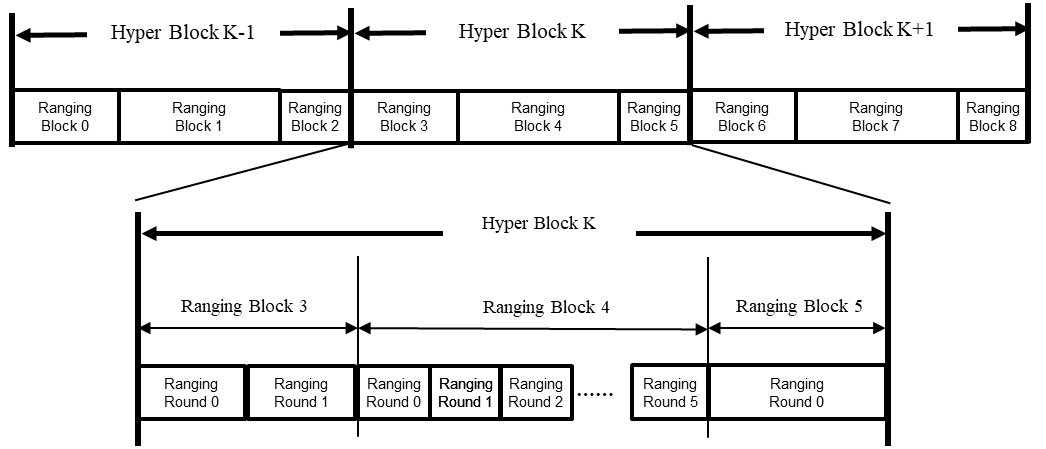
1. **Proposed Changes**

**Proposed text changes on P802.15.4ab™/D (pre-ballot) C:**

*※ NOTE : The RED changes below are implemented by author based on accepted CRs so far, just purely for everyone’s conveniences. This is because there are many accepted CRs that is going to made big changes in the text. Therefore, implementing changes on top of pure original Draft C was meaningless. So please just see the proposed changes from TRACK CHANGES ON only.*

**10.31.3.5 Hyper block mode**

A hyper block is a group of ranging blocks. Hyper block mode uses the time structure that is periodic. Figure 6 shows an example timing diagram of hyper block mode.

****

**Figure 6 – Example of timing diagram of hyper block mode**

Each hyper block consists of a whole number of ranging blocks. In the hyper block mode, the individual ranging blocks within a hyper block may have different configuration for their ranging block duration, ranging round duration, or ranging slot duration, while successive hyper blocks employ the same configuration.

The configuration for the hyper block structure may be repeatedly transmitted in every RCM by the controller. The Hyper Block Structure IE (HBS IE), as defined in 10.31.9.12, may be used to signal the durations of each of the ranging blocks in the hyper block. The RCM with HBS IE may be transmitted in the first slot in every hyper block. The HBS IE specifies the index of the corresponding hyper block and includes a list of the durations of all the ranging blocks within the hyper block. Optionally, round duration and slot duration may also be specified in the HBS IE. On reception of an HBS IE with the RCM, a controlee may assume that hyper block structure is followed. The HBS IE takes effects from the corresponding hyper block where HBS IE exists. Each block structure may be setup by specifying the Ranging Block Duration field, the Ranging Round Duration field, and the Ranging Slot Duration field in the HBS IE and/or the ARC IE within the RCM. If the HBS IE and the ARC IE are both present in the same RCM, the ranging parameters are jointly configured by the HBS IE and the ARC IE, The common parameter can exist both in ARC IE or HBS IE. In that case the corresponding overlapping parameter values should be the same so to make devices avoid confusion. The hyper block structure is determined by the next higher layer.

The hyper block mode is optional for all devices. Each hyper block is identified by hyper block index. This is the total number of hyper blocks that has elapsed since the start of the network and increments by one with each hyper block execution. It is announced by controller with HBS IE. HBS IE also delivers block index which is relative inside a hyper block and it starts from 0 in every hyper block and increases by one with each block. Then, the relationship of the ranging block index and this relative block index is as follows;

Ranging Block Index = Current Hyper Block Index × the number of blocks in a hyper block

+ the relative block index

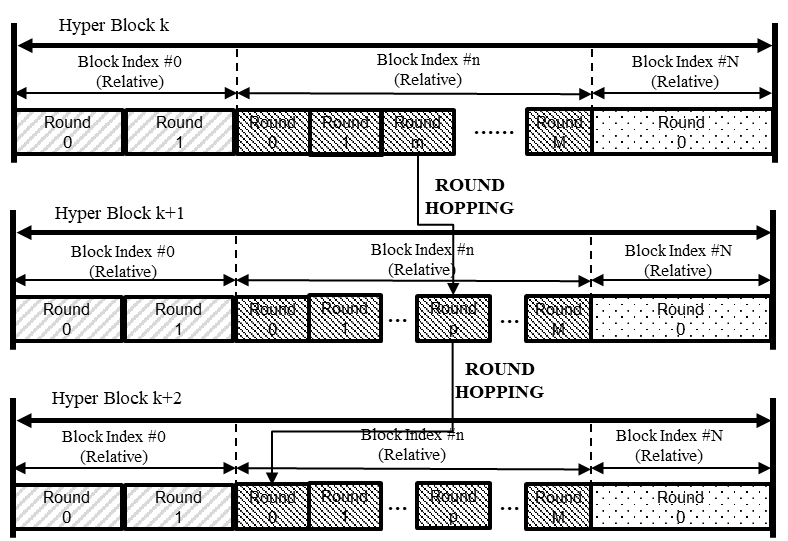
The hyper block index together with the Ranging Block Index is used by devices to maintain synchronization with the block structure.

Different blocks within a hyper block may be allocated for different applications such as ranging or sensing or data communications.

Hyper block keeps the same structure repeated in every hyper block. Round Hopping is optional in hyper block mode. Round hopping may be performed in the hyper block mode in one of the following methods:

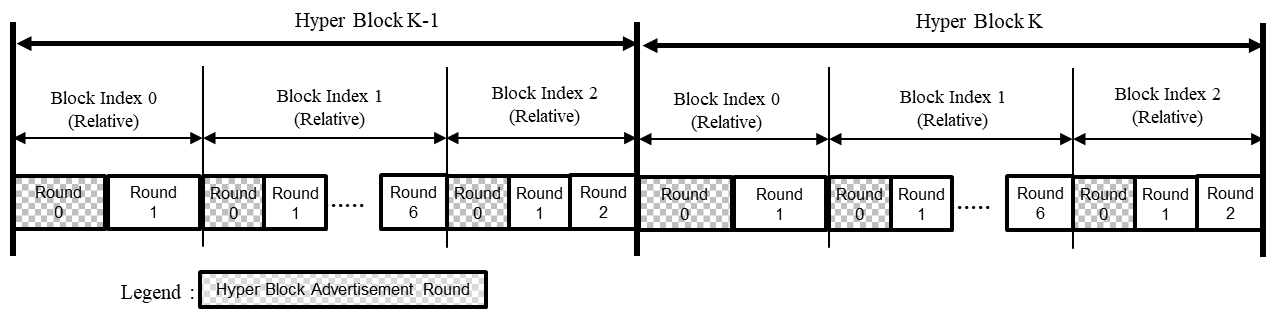
* If a controlee receives an Enhanced Ranging Round IE (ERR IE) (as described in 10.31.9.11) in which the Hopping Mode field is set to one, the controlee may hop to one of the ranging rounds in the ranging block indicated by the ERR IE.
* If the controlee receives a Scheduling IE (as described in 10.31.9.10) with the Scheduling List Type equal to six in which the controlee’s address is present in a Block Assignment field in which the Hopping Mode field is set to one, the controlee may hop to one of the ranging rounds in the ranging block indicated by the Block Assignment field.
* Otherwise, if the controlee receives a second RR IE in its ranging round in which the Hopping Mode field is set to one, the controlee may hop to one of round at the block having the same Block Index number relatively in the next hyper block. The controlee may hop to one of round at the block having the same Block Index number relatively in the next hyper block. Then, transmission at m-th Round in n-th Block relatively within k-th Hyper Block hops to p-th Round in n-th Block relatively within (k+1)-th Hyper Block, (m not equal to p).

Note – If the controlee receives a second RR IE in its ranging round in which the Hopping Mode field is set to one, any other block scheduling method (e.g. Bitmap-based block scheduling) in hyper block mode is not used.



**Figure 7 – Round hopping in hyper block mode**

The Controller may also allocate a hyper block advertisement (HBA) round, at least once in each hyper block, to advertise the assigned block for each participating device or network (e.g., RAN (Ranging Area Network)). The hyper block advertisement round may be fixed as the first round of each ranging block in each hyper block, or it may be a negotiated round in a certain block of each hyper block. (e.g., negotiated during session setup). In each hyper block advertisement round, the Controller transmits a scheduling IE carrying the block assignment schedule, as defined in 10.31.9.10, for that hyper block. An example where the controller allocates a hyper block advertisement (HBA) round in the first round of every ranging block is illustrated in Figure 8.



**Figure 8 – Example of Hyper block advertisement**

In an allocated ranging round of a ranging block within a hyper block, the controller may transmit an Enhanced Ranging Round IE (ERR IE), described in 10.31.9.11, to inform the next ranging block that is assigned to a controlee, the number of rounds in the next assigned ranging block and the ranging round information in the next assigned ranging block. The ERR IE may be included in the RCM or in the last message sent by the controller to the controlees in the current ranging round. The ERR IE will also signal to the controlees whether to hop to a different round and/or use a different transmission offset in the ranging round of the next assigned ranging block. After receiving the ERR IE in the final message of a ranging message sequence or in an RCM, the next higher layer of the controlee is responsible for using the indicated ranging round and transmission offset in the next assigned ranging block. If round hopping is enabled, the controlee may infer the number of rounds in the block based on the Number of Rounds field in the ERR IE and will be able to calculate its allocated round in the block.

If the controlee does not receive the ERR IE (either in the final message of the exchange or in the RCM), for example due to an interference event, the controlee may listen to the channel at the next known hyper block advertisement round to receive the scheduling IE carrying the block assignment schedule for the hyper block. After receiving the block assignment(s), if the controlee finds its address or the address of the network it belongs to in the Scheduling IE, it will know the block that is assigned to it. If round hopping is enabled, it may also calculate the number of rounds in the block based on the Ranging Block Duration field and the Round Duration field in the HBS IE and will be able to calculate its allocated round in the block.

**10.31.9 Nested IEs for multi-node ranging**

**10.31.9.3 Ranging Round IE (RR IE)**

***Change the text of the RR IE clause 10.29.9.3, as shown:***

The RR IE may be used to signal ranging round information for the current ranging round or ranging round information for the next ranging round in both block-based mode and hyper block mode according to the description in 6.9.7.3.3. However, in case of hyper block mode, the "next ranging block" or “ranging block i+1” mentioned in section 6.9.7.3.3 does not specify the next ranging block in the current hyper block (k), but rather specifies the ranging block in the next hyper block with the same block index relatively as the current ranging block (i.e., ranging block I (relative) in hyper block k+1). The Content field of the RR IE shall be formatted as shown in Figure 10-239.



The Ranging Block Index field specifies the index of the ranging block except in hyper block mode when neither block assignment scheduling nor Bitmap-based block scheduling (as described in 10.31.9.10 Scheduling IE) is used. In hyper block mode when neither block assignment scheduling nor Bitmap-based block scheduling is used, the Ranging Block Index field is assumed to specify Hyper Block Index for the ranging hyper block and controlee may assume the block index relative will be the same with previous hyper block.

NOTE—When block assignment scheduling is used in hyper block mode, the ERR IE is used instead of the RR IE to signal the ranging round information for the next ranging round in the next assigned ranging block.

The Hopping Mode field specifies the hop mode for the ranging block, where zero means no hopping and one means hopping.

The Round Index field specifies the ranging round index for the ranging block,

The Transmission Offset field specifies the value of transmission offset of the ranging round in the block, in RSTU. This offset shall be at most the ranging slot duration minus the packet duration.

The RR IE is only used in block-based mode and in hyper block mode without block assignment scheduling.

Devices participating in the ranging exchange have either (a) pre-negotiated a hopping sequence that is known to all devices, or (b) have exchanged all the information necessary such that each device can generate the hopping sequence.

***Insert the new sub-clauses 10.29.9.10 and 10.29.9.12 after 10.29.9.9 as follows:***

**10.31.9.10 Scheduling IE**

The Scheduling IE is used by the controller to schedule blocks or slots to be used by intended device. The Content field of the Scheduling IE shall be formatted as shown in Figure 9.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Bits: 0-2** | **3-6** | **7** | **8** | **9-15** | **Octets: Variable** |
| Scheduling List Type | Scheduling List Length | Address Size | Receiver Address Present | Reserved | Scheduling List |

**Figure 9 – Scheduling IE Content field format**

The Scheduling List Length field indicates the length in octets of the Scheduling List field. The format of the Scheduling List field depends on the value of the Scheduling List Type field.

The Scheduling List Type field specifies how each element of the Scheduling List field is formatted and shall have one of the values specified in Table 4 to select the type of scheduling.

****

When the per-slot scheduling is used, each Scheduling List element schedules one slot to a device.

When the consecutive slot scheduling is used, each Scheduling List element schedules one slot to a device. Since there is no Slot Index field in the Scheduling List element, slots are scheduled in a sequential order. For example, the slot following the slot in which the IE is sent shall be scheduled for the device specified in the first Scheduling List element. There shall be no empty slot between scheduled slots.

When the bitmap-based slot scheduling is used, multiple slots may be scheduled to a device by using one Scheduling List element. A bitmap in each Scheduling List element represents the pattern of scheduled slots to a single device.

When the periodic scheduling is used, multiple slots may be scheduled to a device by using one Scheduling List element. A pattern of scheduled slots shall be represented by the size of scheduling step and the number of scheduling repetitions.

When the RSF scheduling is used, multiple slots may be scheduled to a device by using one Scheduling List element. At a slot, devices shall transmit RSF according to the Scheduling List element, and the composition of RSF is determined by the Scheduling List element.

When the Bitmap-based block scheduling is used, multiple blocks may be scheduled to a device by using one Scheduling List element. A bitmap in each Scheduling List element represents the pattern of scheduled blocks to a single device. For example, Scheduling IE with Scheduling List Type 5 can be transmitted with same ranging round HBS IE, defined in 10.31.9.12, for hyper block mode scheduling and the bitmap in each Scheduling List element represents scheduled blocks to a single device in a hyper block.

When the block assignment scheduling is used, a block may be assigned to one or more device or network using one Scheduling List element as shown in Figure 16. The relative Block Index field in the Scheduling List element identifies the block and the Address List field in the Scheduling List element carries the address of the network or devices that are allocated one or more round in the block.

The Address Size field specifies the size of the Sender Address field or the Receiver Address field or the addresses in the Address List field when the block assignment scheduling is used. If the Address Size field is zero, short address shall be used for the Sender Address field the Receiver Address field and the addresses in the Block Assignment List field. If the Address Size field is one, extended address shall be used for the Sender Address field the Receiver Address field and the addresses in the Block Assignment List field. When the block assignment scheduling is used and the Block Assignment List field carries address of networks, the Address Size field indicates short address.

The Receiver Address Present field when one indicates the presence of the Receiver Address field, or not present when zero.

The format of the Scheduling List field depends on the value of the Scheduling List Type field.

When the Scheduling List Type field is zero, Scheduling List elements shall be formatted as per Figure 10.

****

The Slot Index field is used to assign a slot index to the device identified by the Sender Address field.

The Sender Address field identifies each participating device.

When the Scheduling List Type field is one, Scheduling List elements shall be formatted as per Figure 11.

****

The Sender Address field identifies each participating device.

When the Scheduling List Type field is two, Scheduling List elements shall be formatted as per Figure 12.

****

The Scheduling Bitmap Length field specifies the size of the Bitmap field. The Scheduling Bitmap Length field shall have one of the values specified in Table 5.

****

The Bitmap Offset Present field when one indicates the presence of the Bitmap Offset field, or not present 3 when zero.

The Scheduling Bitmap field contains a binary bitmap string. Each bit of the bitmap maps to the slots following the slot in which the Scheduling IE is transmitted. For example, if the Scheduling IE is sent in the slot whose index is 0 and the Bitmap Offset Present field is set to 0, the first bit corresponds to the slot whose index is 1. The bit is set to 1 to indicate that the corresponding slot is scheduled, otherwise the bit is set to zero to indicate that the corresponding slot is not scheduled. The first bit in time sent in the field refers to the first time slot and the subsequent bits refer chronologically to the subsequent time slots. When the bitmap is larger than the number of slots remaining in the current round, the excess bits of the bitmap shall be ignored.

The Sender Address field identifies the device selected to send in the scheduled slots.

The Receiver Address field, if present, indicates the destination for the frames in the scheduled slots.

The Bitmap Offset field specifies the number of slots between the slot on which the Scheduling IE is sent and the first slot to be scheduled. The first slot to be scheduled corresponds to the first bit in the bitmap. For example, if the Scheduling IE is sent in slot index zero and the Bitmap Offset field is set to five, then the first bit of the bitmap corresponds to a slot index of six.

When the Scheduling List Type field is set to three, Scheduling List elements shall be formatted as per Figure 13

****

The Starting Slot Index field indicates the first slot of the periodic scheduling pattern.

The Scheduling Step field specifies the number of slots in the gap between periodic scheduled slots.

The Scheduling Repetition field specifies the number of repetitions of scheduled slots within the periodic scheduling pattern.

The Sender Address field identifies the device selected to send in the scheduled slots.

The Receiver Address field, if present, indicates the destination for the frames in the scheduled slots.

When the Scheduling List Type field is set to four, the Scheduling List elements shall be formatted as per Figure 14.

****

The Starting Slot Index field marks the first slot of the RSF transmission.

The Scheduling Step field specifies the number of slots in the gap between scheduled slots.

The Scheduling Repetition field specifies the number of scheduled slots within the periodic scheduling pattern.

The Sender Address field identifies the device selected to send in the scheduled slots.

The Receiver Address field, if present, indicates the destination for the frames in the scheduled slots.

The Sequence Index field indicates a code index from Table 16-8, Table 16-9, or Table 50 that is allocated to the device this Scheduling List element relates to.

If sequence index field indicates a code index from Table 50, the Number of Gaps field specifies the length of zeros to insert at the middle and end of the sequence as described in 16.2.11.2. The value of the Number of Gaps field shall be between 0 and 64.

The Sequence Repetition field indicates the number of multi-millisecond ranging sequence (MMRS) symbol repetitions (MSR) in the RSF. The value of this field shall be between 32 and 256.

When the Scheduling List Type field is set to five (Bitmap-based block scheduling), the Scheduling List elements shall be formatted as per Figure 15.

****

The Block scheduling Bitmap Length field specifies the size of the Block Scheduling Bitmap field. The Block Scheduling Bitmap Length field shall have one of the values specified in Table 5.

The Block Scheduling Bitmap field contains a binary bitmap string. Each bit maps to the blocks following and including the block in which the Scheduling IE is transmitted. For example, if there are three blocks in a hyper block, the first, second and third bits correspond to the blocks with indexes 0, 1, and 2 in the hyper block, respectively. A bit in the bitmap is set to 1 to indicate that the corresponding block is scheduled or set to zero to indicate that the corresponding block is not scheduled. When the number of bits sent in the Block Scheduling Bitmap field is greater than the number of remaining blocks, the excess bits shall be ignored.

When the Scheduling List Type field is set to six (block assignment scheduling), the Scheduling List elements shall be formatted as per Figure 16.

|  |  |  |
| --- | --- | --- |
| **Octets: 1** | **1** | **variable** |
| Relative Block Index | Block Assignment List Length | Block Assignment List |

**Figure 16—Scheduling List element format when Scheduling List Type is six**

The Relative Block Index field specifies the index of the ranging block within the hyper block.

The Block Assignment List Length field specifies the number of Block Assignment in the Block Assignment List field.

The Block Assignment Address List field carries a list of one or more Block Assignment field. The Block Assignment field shall be formatted as illustrated in Figure 16B. address of the network or devices that are allocated one or more round in the block identified by the Relative Block Index field. For networks, short address is used.

|  |  |  |
| --- | --- | --- |
| **Octets: 2 or 8** | **Bits: 0** | **1-15** |
| Address | Hopping Mode | Round Index |

**Figure 16B—Block Assignment field format**

The Address field specifies the address of the network or devices that are allocated one or more round in the block identified by the Relative Block Index field. The size of the Address field is specified by the Address Size field.

The Hopping Mode field specifies the hop mode for the assigned ranging block, where zero means no hopping and one means hopping.

The Round Index field specifies the round index for the assigned ranging block when round hopping is not enabled.

If the Scheduling IE is included in the same frame as an RDM IE (as defined in 10.31.9.8) then the Scheduling IE shall be used for the scheduling.

**10.31.9.11 Enhanced Ranging Round IE (ERR IE)**

The ERR IE is used by the controller to inform the next assigned ranging block, the number of rounds in the next assigned ranging block and the ranging round information in the next assigned ranging block to devices. The Content field of the ERR IE shall be formatted as illustrated in Figure 17.



The Hyper Block Index field specifies the index of the hyper block in which the next assigned ranging block is located.

The Relative Block Index field specifies the relative block index of the next assigned ranging block within the hyper block (zero indicates the first ranging block).

The Hopping Mode field specifies the hop mode for the next assigned ranging block, where zero means no hopping and one means hopping.

The Round Index field specifies the round index for the next assigned ranging block when round hopping is not enabled.

The Transmission Offset field specifies the value of transmission offset of the round in the next assigned ranging block, in RSTU. This offset shall be at most the ranging slot duration minus the packet duration.

The Number of Rounds field specifies the number of rounds in the next assigned ranging block and is present when the Hopping mode field is set to one.

**10.31.9.12 Hyper Block Structure IE (HBS IE)**

The HBS IE is used by the controller to send the hyper block structure configuration to controlees in the RCM message. The Content field of the HBS IE shall be formatted as illustrated in Figure 18.

|  |  |  |  |
| --- | --- | --- | --- |
| **Octets: 2** | **1** | **1** | **variable** |
| Hyper Block Index | Content Control | Ranging Block Description List Length | Ranging Block Description List |

The Hyper Block Index field specifies the index of the hyper block,

The Content Control field is formatted as per Figure 19 and indicates the presence of duration fields in the Ranging Block Description List, which is structured as per Figure 20.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bits: 0–1** | **2** | **3** | **4–7** |
| Ranging Block Duration Units | Ranging Round Duration Presence | Ranging Slot Duration Presence | Reserved |

**Figure 19—Content Control field format**

The Ranging Block Duration Units field indicates the Ranging Block Duration field size as per Table 6. Controller can choose any of the units in its own needs, i.e. the number of rounds (=units field value 0), the number of slots (=units field value 1) or RSTU (=units field value 2). Basically, all those three values (Block, Round, Slot Duration) are recommended to be present at HBS IE at the same time. But, as the hyper block structure is supposed to repeat the same, round duration and slot duration can be omitted by setting ‘Round Duration Presence’ and ‘Slot Duration Presence’ bit to be zero and refer the values from the most recent previous hyper blocks which has the value.

**Table 6—Ranging Block Duration Units field**

|  |  |
| --- | --- |
| **Ranging Block Duration Units field value** | **Description** |
| 0 | Size of Ranging Block Description List Length field is 1 octet and the Ranging Block Description List field units are the number of ranging rounds. |
| 1 | Size of Ranging Block Description List Length field is two octets, and the Ranging Block Description List field units are the number of slots |
| 2 | Size of Ranging Block Description List Length field is three octets, and the Ranging Block Description List field units are in RSTU. |
| 3 | Reserved |

The Ranging Round Duration Presence field indicates the presence of the Round Duration field when it is 1, and it is not present when it is 0 as per Figure 20.

The Ranging Slot Duration Presence field indicates the presence of the Slot Duration field when it is 1, and it is not present when it is 0 as per Figure 20.

Ranging Block Description List Length field specifies the number of Ranging Block Description List elements in the Ranging Block Description List field. The number of Ranging Block Description List Elements shall be set equal to the number of blocks in the hyper block.

Ranging Block Description List field contains Ranging Block Description List elements each of which is structured as per Figure 20.

|  |  |  |  |
| --- | --- | --- | --- |
| **Octets: 1** | **1/2/3** | **0/1** | **0/2** |
| Relative Block Index | Ranging Block Duration | Ranging Round Duration | Ranging Slot Duration |

**Figure 20—Ranging Block Description List field format**

The Relative Block Index field specifies the index of the ranging block within the hyper block. This value is relative within a hyper block, so it starts from zero when every hyper block having a new hyper block index begins.

The Ranging Block Duration field is an unsigned integer that specifies the duration of the ranging block. The size and the unit of the Block Duration field are determined by the Ranging Block Duration Unit field as per Table 6.

The Ranging Round Duration field is an unsigned integer that specifies the duration of the round in units of slots, which is the number of slots in the round.

The Ranging Slot Duration field is an unsigned integer that specifies the duration of a slot in RSTU.