**IEEE P802.15**

**Wireless Specialty Networks**

|  |  |  |
| --- | --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Specialty Networks (WSNs) – 802.15.4ab | |
| Title | **Proposed Text for 4ab MAC – Block Assignment in Hyper Blocks** | |
| Date Submitted | 18 May 2023 | |
| Source | Rojan Chitrakar, Lei Huang, Kuan Wu, Bin Qian, David Xun Yang (Huawei)  Hongwon Lee, Insun Jang, Jinsoo Choi, Hangyu Cho(LG Electronics)  Kangjin Yoon(Meta)  Youngwan So, Mingyu Lee, Taeyoung Ha, Aniruddh Rao Kabbinale and Clint Chaplin (Samsung Electronics) | E-mail: [rojan.chitrakar@huawei.com]    E-mail: [hongwon.lee@lge.com]  E-mail: [kyoon@meta.com]  E-mail: [youngwan.so@samsung.com] |
| Re: | Developing technical content for actual specification text. | |
| Abstract | This document provides details of MAC features for 4ab especially for Hyper block-based mode | |
| Purpose | Support development of technical content for the draft | |
| 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. | |

The baseline for this TFD is 15-23-0155-01-04ab-status-update-on-hyper-block-based-mode-v2. The modifications compared to the baseline are shown as markups.

**6.9.7.3.5 Hyper block-based mode**

***Modify the subclause as follows (Track changes ON):***

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



**Figure 6-XXX – Example of timing diagram of hyper block-based mode**

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

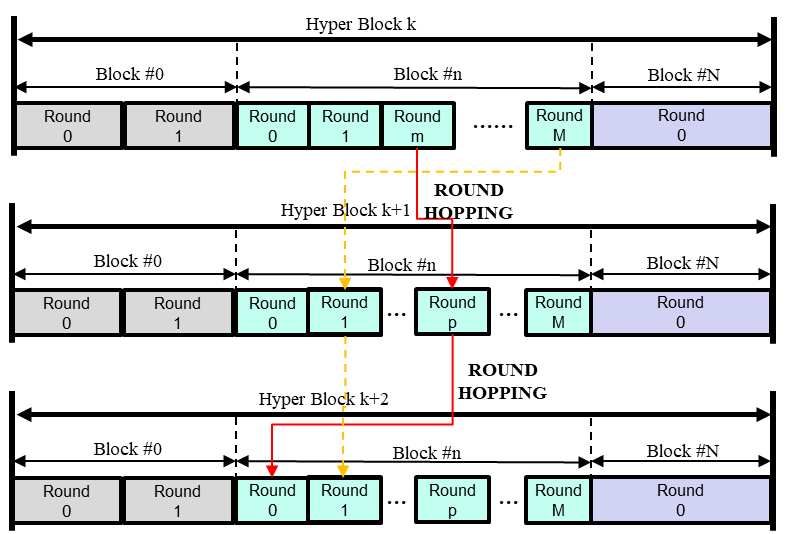
The configuration for the hyper block structure can be repeatedly transmitted in every RCM by the controller. The Hyper Block Structure IE (HBS IE), as defined in 7.4.4.56, can be used to signal the durations of each of the ranging blocks in the hyper block. The HBS IE specifies the index of the corresponding ranging block and includes a list of the durations of all the ranging blocks within the hyper block. Optionally, round duration, slot duration can also be specified in the HBS IE. On reception of an HBS IE with the RCM, a controlee can assume that hyper block structure is followed. Each block structure can 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. The hyper block structure is determined by the next higher layer.

The hyper block-based mode is optional. 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. It is used by devices as hyper block counter to identify where it is now, as ranging block index restarts from 0 again in every hyper block.

The packets sent in the ranging block/round/slots may be used for ranging and/or sensing and/or data communications as well.

Hyper block keeps the same structure repeated in every hyper block, typically. When bitmap-based block scheduling or block assignment scheduling (as described in 7.4.4.X Scheduling IE) is not used, to do round hopping, the controlee should hop to one of round at the block having the same Block Index number in the next hyper block.

Then, transmission at m-th Round in n-th Block within k-th Hyper Block hops to p-th Round in n-th Block within (k+1)-th Hyper Block (m!=p)



**Figure ooo-ooo — Round hopping in hyper-block based 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 7.4.4.X (Schedule IE), 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 6-OOO.



Figure 6-OOO

In an allocated ranging round of a ranging block within a hyper block, the controller may transmit an Enhanced Ranging Round IE (ERR IE) (as described in 7.4.4.57) 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 controlee next higher layer is responsible for using the indicated ranging round and transmission offset in the next assigned ranging block. If round hopping is enabled, it can know 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 can 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 can also calculate the number of rounds in the block based on the Block Duration field and the Round Duration field in the HBS IE and will be able to calculate its allocated round in the block.

***Revise the sub-clause 7.4.4.X Scheduling IE in 15-23-0062-03-04ab-text-for-scheduling-ie as follows:***

**7.4.4.X 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 7-X.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bits: 0–3 | 4–6 | 7 | 8 | 9–15 | Octets: variable |
| Scheduling List Length | Scheduling List Type | Address Size | Receiver Address Present | Reserved | Scheduling List |

**Figure 7-X – Scheduling IE Content field format**

The Scheduling List Length field indicates the number of elements in the Scheduling List field, each of which is formatted as per Figure 7-XX or Figure 7-XXX depending 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. The Scheduling List Type field shall have one of the values specified in Table 7-Y.

**Table 7-Y – Values of the Scheduling List Type field in the Scheduling IE**

|  |  |
| --- | --- |
| Scheduling List Type field value | The type of Scheduling List field |
| 0 | Per-slot scheduling |
| 1 | Consecutive slot scheduling |
| 2 | Bitmap-based slot scheduling |
| 3 | Periodic scheduling |
| 4 | RSF scheduling |
| 5 | Bitmap-based block scheduling |
| 6 | Block assignment scheduling |
| 7 | Reserved |

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 Control Message 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 (defined in x.x.x) 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 may be transmitted with same cycle of HBS IE defined in 7.4.4.56 for hyper block-based 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 7-Z2. The Ranging 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 and the Receiver Address field. If the Address Size field is one, extended address shall be used for the Sender Address field and the Receiver Address field. When the block assignment scheduling is used and the Address 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 set to 0, Scheduling List elements shall be formatted as per Figure 7-Y.

|  |  |
| --- | --- |
| Octets: 1 | 2/8 |
| Slot Index | Sender Address |

**Figure 7-Y – Scheduling List element format when Scheduling List Type is 0**

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 set to 1, Scheduling List elements shall be formatted as per Figure 7-YY.

|  |
| --- |
| Octets: 2/8 |
| Sender Address |

**Figure 7-YY – Scheduling List element format when Scheduling List Type is 1**

The Sender Address field identifies each participating device.

When the Scheduling List Type field is set to 2, Scheduling List elements shall be formatted as per Figure 7-XX.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Bits: 0–1 | 2 | 3–7 | Octets: Variable | 2/8 | 0/2/8 | 0/1 |
| Scheduling Bitmap Length | Bitmap Offset Present | Reserved | Scheduling Bitmap | Sender Address | Receiver Address | Bitmap Offset |

**Figure 7-XX – Scheduling List element format when Scheduling List Type is 2**

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 7-X.

**Table 7-X – Values of the Scheduling Bitmap Length field in the Scheduling IE**

|  |  |
| --- | --- |
| Scheduling Bitmap Length field value | The size of Scheduling Bitmap field |
| 0 | 8 bits bitmap |
| 1 | 16 bits bitmap |
| 2 | 32 bits bitmap |
| 3 | 64 bits bitmap |

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

The Scheduling Bitmap field contains a binary bitmap string. Each bit 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 number of bits sent in the Scheduling Bitmap field is greater than the number of remained slots, the last excess bits sent shall be ignored.

The Sender Address field identifies which device can send frames in scheduled slots.

The Receiver Address field, if present, indicates the device to which frames will be sent in 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 the slot whose index is 0 and the Bitmap Offset field is set to 5, the first bit corresponds to the slot whose index is 6.

When the Scheduling List Type field is set to 3, Scheduling List elements shall be formatted as per Figure 7-XXX.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bits: 0–6 | 7–10 | 11–15 | Octets: 2/8 | 0/2/8 |
| Starting Slot Index | Scheduling Step | Scheduling Repetition | Sender Address | Receiver Address |

**Figure 7-XXX – Scheduling List element format when Scheduling List Type is 3**

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 which device can send frames in scheduled slots.

The Receiver Address field, if present, indicates the device to which frames will be sent in scheduled slots.

When the Scheduling List Type field is set to 4, Scheduling List elements shall be formatted as per Figure 7-YYY.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bits: 0–6 | 7–10 | 11–15 | Octets: 2/8 | 0/2/8 | 1 | 1 | 1 |
| Starting Slot Index | Scheduling Step | Scheduling Repetition | Sender Address | Receiver Address | Sequence Index | Number of Gaps | Sequence Repetition |

**Figure 7-YYY – Scheduling List element format when Scheduling List Type is 4**

The Starting Slot Index field marks the first transmission slot after trigger step of multiple RSF transmission in the recurring periodic transmission pattern in unit of slots.

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 which device can send frames in scheduled slots.

The Receiver Address field, if present, indicates the device to which frames will be sent in scheduled slots.

The Sequence Index field indicates the code index of MMRS that allocated to the device in this Scheduling List element relate to.

If sequence index field indicates the code indices of MMRS based on the length-128 complementary set, Number of Gaps field shall be used to specify the length of zero between two parts of the length-128 complementary set. The value of this field shall be between 0 and 64.

The Sequence Repetition field indicates the number of MMRS repetitions in RSF (i.e., N\_MSR), and the value of this field shall be between 32 and 256.

When the Scheduling List Type field is set to 5(Bitmap-based block scheduling), Scheduling List elements is formatted as per Figure 7-Z.

|  |  |  |  |
| --- | --- | --- | --- |
| Bits: 0–1 | 2–7 | Octets: Variable | 2/8 |
| Block scheduling Bitmap Length | Reserved | Block Scheduling Bitmap | Sender Address |

**Figure 7-Z – Scheduling List element format when Scheduling List Type is 5**

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 7-X.

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 defined in 6.9.7.4.5, the first, second and third bits correspond to the block whose index are 0, 1, and 2 in a Hyper Block respectively. The bit is set to 1 to indicate that the corresponding block is scheduled, otherwise the bit is 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 remained blocks, the last excess bits sent shall be ignored.

The Sender Address field identifies which device can send frames in scheduled blocks.

When the Scheduling List Type field is set to 6 (Bitmap-based block scheduling), Scheduling List elements is formatted as per Figure 7-Z2.

|  |  |  |
| --- | --- | --- |
| Octets: 1 | 1 | variable |
| Ranging Block Index | Address List Length | Address List |

**Figure 7-Z2 – Scheduling List element format when Scheduling List Type is 6**

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

The Address List Length field specifies the number of addresses in the Address List field.

The Address List field carries a list of address of the network or devices that are allocated one or more round in the block identified by the Ranging Block Index field. For networks, short address is used.

If RDM IE defined in 7.4.4.44 and the scheduling IE defined in 7.4.4.x exist in the same control message, then each ERDEV(s) in enhanced HPRF mode shall only be scheduled by the scheduling IE defined in 7.4.4.x.

**7.4.4.38 Ranging Round IE (RR IE)**

***Change 7.4.4.38 as follows:***

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 according to the description in 6.9.7.3.3. The Content field of the RR IE shall be formatted as shown in Figure 7-106g.

|  |  |  |  |
| --- | --- | --- | --- |
| **Octets:2** | **Bits : 0** | **1-15** | **Octets : 2** |
| Ranging Block Index | Hopping Mode | Round Index | Transmission Offset |

**Figure 7-106g—RR IE Content field format**

The Ranging Block Index field specifies the index of the ranging block, in case of block-based mode. In case of Hyper Block-based mode, when assignment scheduling (as described in 7.4.4.X Scheduling IE) is not used, it is assumed to specify Hyper Block Index for the ranging hyper block and controlee can assume the block index will be the same with previous hyper 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. 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 subclause*** ***7.4.4.57 as follows (Track changes ON):***

**7.4.4.57 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 6-NNN.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Octets: 2 | 1 | Bits: 0 | 1 - 15 | Octets: 2 | 0 or 1 |
| Hyper Block Index | Ranging Block Index | Hopping Mode | Round Index | Transmission Offset | Number of Rounds |

Figure 6-NNN – ERR IE Content field format

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

The Ranging Block Index field specifies the 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.