IEEE P802.15
Wireless Specialty Networks

|  |
| --- |
| IEEE P802.15.13Text parts for updated clause 7 |
| Date: 2019-07-15 |
| Author: |
| Name | Affiliation | Address | Phone | Email |
| Kai Lennert Bober | Fraunhofer HHI |  |  | kai.lennert.bober@hhi.fraunhofer.de |
|  |  |  |  |  |
|  |  |  |  |  |

Abstract

# This document contains proposed text for clause 7 of IEEE P802.15.13.

1.
2. MAC services

The IEEE 802.15.13 MAC offers its service to the higher protocol layers and DME through the MCPS-SAP and MLME-SAP respectively. The MCPS-SAP includes primitives that support the integration of IEEE 802.15.13 networks in bridged LANs in accordance with IEEE Std 802.1AC. The MLME-SAP exposes basic management functions and further advanced functionality to the DME.

A primitive invocation originating from the service user, i.e. higher layer, carries the suffix **.request**. Hence, it requests the start of a service, i.e. action, on the service provider, which is the next lower layer. The immediate response to a **.request** is a **.confirm** primitive, returned by the service provider, i.e. the MAC or MLME.

Externally caused events at the MAC or MLME are indicated to the higher layer through primitives carrying the **.indication** suffix. Such events may originate from unrequested actions, e.g. the reception of a specific management frame, or as an asynchronous response to a finished service invocation through a preceding service request. The response to an **.indication** is **.response** primitive invocation**.**

A number of PIB attributes defines the behavior of the MAC and reflects the current system state. MAC PIB Attributes are covered in clause 7.3.

Moreover, capabilities indicate subparts of functionality in this standard that are supported by a given device implementation. Those capabilities are used to negotiate functionality that can be used while a device is associated with a given OWPAN. Capabilities are covered in clause 7.4.

* 1. MCPS-SAP

The MCPS-SAP supports the transport of MSDUs between the MACs of peer IEEE 802.15.13 devices through the primitives listed in Table 15. MSDUs shall be in LLC EtherType Protocol Discrimination (EPD) format (see IEEE Std 802-2014). Table 15: MCPS-SAP primitives provides a summary of all MCPS-SAP primitives.

|  |  |  |
| --- | --- | --- |
| **MCPS-SAP primitive** | **Request** | **Indication** |
| MCPS-DATA | 7.1.1 | 7.1.2 |

Table 16: MCPS-SAP primitives

* + 1. MCPS-DATA.request

The MCPS-DATA.request primitive is used by the higher layer to request the transfer of data to another device.

The parameters of the primitive are listed in Table 16.

|  |  |  |
| --- | --- | --- |
| **Parameter name** | **Range** | **Parameter description** |
| DestinationAddress | 48-bit MAC addresses | The destination address of the MSDU. |
| SourceAddress | 48-bit MAC addresses | The source address of the MSDU. |
| Msdu | Octet Sequence | The actual MSDU. |
| Priority | [0, 7] | The priority of the MSDU. |
| Protected | TRUE, FALSE | Whether the associated MSDU shall be transmitted protected. |

Table 17: Parameters of the MCPS-DATA.request primitive

* + 1. MCPS-DATA.indication

MCPS-DATA.indication primitive is issued by the MAC of a device upon reception of a MSDU from a peer device.

The parameters of the primitive are listed in Table 18.

|  |  |  |
| --- | --- | --- |
| **Parameter name** | **Range** | **Parameter description** |
| DestinationAddress | 48-bit MAC addresses | The destination address of the MSDU. |
| SourceAddress | 48-bit MAC addresses | The source address of the MSDU. |
| Msdu | Octet Sequence | The actual MSDU data. |
| Priority | [0, 7]  | The priority associated with the MSDU |

Table 19: Parameters of the MCPS-DATA.indication primitive

* 1. MLME-SAP

The MLME-SAP supports the management and usage of a device’s MLME functionality through the DME. The primitives supported by the MLME-SAP are given in Table 18. Primitives in brackets are optional for devices and must be implemented by devices that can act as coordinators (i.e. implement the *capCoordinator* capability).

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MLME-SAP primitive** | **Request** | **Confirm** | **Indication**  | **Response**  |
| MLME-ASSOCIATE | 7.2.1.1 | 7.2.1.2 | (7.2.1.3) | (7.2.1.4) |
| MLME-DISASSOCIATE | 7.2.2.1 | 7.2.2.2 | 7.2.2.3 |  |
| MLME-GET | 7.2.3.1 | 7.2.3.2 |  |  |
| MLME-SET | 7.2.4.1 | 7.2.4.2 |  |  |
| MLME-SCAN | 7.2.5.1 | 7.2.5.2 |  |  |
| MLME-START | (7.2.6.1) | (7.2.6.2) |  |  |
| MLME-STOP | (7.2.7.1) | (7.2.7.2) |  |  |

Table 20: Parameters of the MCPS-DATA.indication primitive

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

* + 1. MLME-ASSOCIATE

The MLME-ASSOCIATE primitive serves the association process of a device with an OWPAN as described in clause 5.4.5.

All devices shall provide an interface for the request and confirm association primitives. The indication and response association primitives are optional for a device that is not a coordinator.

* + - 1. Request

The MLME-ASSOCIATE.request is issued by the DME to the device MAC to initiate the association process with a given OWPAN. Upon reception of the primitive, the MLME shall start the association procedure as detailed in 5.4.5.

The parameters of the primitive are listed in Table 20.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| OwpanId | 48-bit MAC address | OWPAN IDs observed in a preceding scan | The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.  |

Table 21: Parameters of the MLME-ASSOCIATE.request primitive

If the MLME of a device receives multiple MLME-ASSOCIATE.request primitives for different target OWPAN IDs, it shall discard all but the first request and wait for its completion or timeout before accepting another request.

* + - 1. Confirm

The MLME-ASSOCIATE.confirm primitive is issued by the MAC layer of a device to report the result of the previously requested association attempt to the DME.

The parameters of the primitive are listed in Table 21.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| OwpanId | 48-bit MAC address | OWPAN IDs | The OWPAN ID for which association was requested in the preceding MLME-ASSOCIATE.request. |
| ShortAddress | 16-bit short address | [1, 65534] | The short address allocated to the device. |
| Status | integer | Status IDs according to Table 8 and Table 22: Status codes for local results of the | The result of the association request. |

Table 22: Parameters of the MLME-ASSOCIATE.confirm primitive

|  |  |
| --- | --- |
| **Value** | **Description** |
| 128 | No response |
| 129 | Invalid Parameter |
| 130-255 | reserved |

Table 23: Status codes for local results of the association request.

* + - 1. Indication

MLME-ASSOCIATE.indication primitive is issued by the MLME of a coordinator in order to ask the DME about the association of a device from which it received an association request.

The parameters of the primitive are listed in Table 23.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| DeviceAddress | 48-bit MAC address | The full 48-bit address of the requesting device | The OWPAN ID for which the association was requested earlier.  |

Table 24: Parameters of the MLME-ASSOCIATE.indication primitive

* + - 1. Response

The MLME-ASSOCIATE.response primitive is used by a coordinator DME to respond to a MLME-ASSOCIATE.indication after deciding how to proceed with the requested association.

The parameters of the primitive are listed in Table 24.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| DeviceAddress | 48-bit MAC address | The full 48-bit address of the requesting device | The OWPAN ID for which the association was requested earlier.  |
| ShortAddress | 16-bit short address | [1, 65534] | The short address allocated to the device seeking association |
| Status | integer | Status IDs according to Table 8 | The result of the decision about the association request. |

Table 25: Parameters of the MLME-ASSOCIATE.response primitive

* + 1. MLME-DISASSOCIATE

The MLME-DISASSOCIATE primitive is invoked in order to disassociate a device from an OWPAN. The primitive may be invoked by a device or the OWPAN coordinator, as described in 5.4.6.

* + - 1. Request

The MLME-DISASSOCIATE.request indicates to the MLME to begin with the disassociation procedure as described in 5.4.6.

The parameters of the primitive are listed in Table 29.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| OwpanId | 48-bit MAC address | OWPAN ID which the device is associated with | The ID of the OWPAN to request disassociation from. |
| DeviceAddress | 48-bit MAC address | Device addresses | The 48-bit MAC address of the device requesting to be disassociated. |
| Reason | integer | Reason codes from Table 9 | The reason for disassociation |

Table 30: Parameters of the MLME-DISASSOCIATE.request primitive

* + - 1. Confirm

The MLME-DISASSOCIATE.confirm is issued by the device MLME to the DME in order to report the result of a preceding MLME-DISASSOCIATE.request.

The parameters of the primitive are listed in Table 30.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| OwpanId | 48-bit MAC address | OWPAN ID which the device is associated with | The ID of the OWPAN to request disassociation from. |
| DeviceAddress | 48-bit MAC address | Device addresses | The 48-bit MAC addressof the device requesting to be disassociated. |
| Reason | integer | Reason codes from Table 9 | The reason for disassociation |

Table 31: Parameters of the MLME-DISASSOCIATE.confirm primitive

* + - 1. Indication

The MLME-DISASSOCIATE.indication is invoked by the MAC to indicate the disassociation of a device from an OWPAN. It may be used by the MLME of a coordinator or participant device of an OWPAN.

The parameters of the primitive are listed in Table 31.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| OwpanId | 48-bit MAC address | OWPAN ID which the device is associated with | The ID of the OWPAN to request disassociation from. |
| DeviceAddress | 48-bit MAC address | Device addresses | The 48-bit MAC address of the device requesting to be disassociated. |
| Reason | integer | Reason codes from Table 9 | The reason for disassociation |

Table 32: Parameters of the MLME-DISASSOCIATE.indication primitive

* + 1. MLME-GET

The MLME-GET primitive allows the DME to obtain the value of certain readable MAC and PHY PIB attributes.

* + - 1. Request

Upon reception of a MLME-GET.request primitive, the MLME shall read the requested MAC or PHY PIB attribute from its information storage and return it via the corresponding confirm primitive.

The parameters of the primitive are listed in Table 32.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| AttributeId | integer | Valid attribute IDs as listed in Table 43Table 40. | The ID of the attribute to get. |

Table 33: Parameters of the MLME-GET.request primitive

* + - 1. Confirm

The MLME-GET.confirm primitive is issued by the MLME as a response to a preceding MLME-GET.request primitive.

The parameters of the primitive are listed in Table 33.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| Status | enumeration | SUCCESS,FAIL\_NON\_EXISTENT,FAIL\_OTHER | Indicates whether the preceding MLME-GET.request primitive was successful or not. |
| AttributeId | integer | Valid attribute IDs as listed in Table 40. | The ID of the attribute to get. |
| AttributeValue | variable | attribute-specific as defined in Table 43Table 40, | The value of the attribute to get. |

Table 34: Parameters of the MLME-GET.confirm primitive

* + 1. MLME-SET

The MLME-SET primitive allows the DME to modify the value of certain writable MAC and PHY PIB attributes.

* + - 1. Request

Upon reception of a MLME-GET.request primitive, the MLME shall set the requested MAC or PHY PIB attribute to have the value provided with the AttributeValue parameter.

The parameters of the primitive are listed in Table 34.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| AttributeId | integer | Valid attribute IDs as listed in Table 40. | The ID of the attribute to get. |
| AttributeValue | variable | Attribute-specific as defined in Table 40 | The value of the attribute to set. |

Table 35: Parameters of the MLME-SET.request primitive

If a PIB attribute is set by the coordinator in accordance with the OWPAN operation configuration, it shall not be writable via a device-local invocation of the MLME-SET.request primitive.

If setting a read-only attribute is attempted, the MLME shall respond in the corresponding confirm with the *Status* parameter set to FAIL\_READ\_ONLY.

* + - 1. Confirm

Through issuing the MLME-SET.confirm primitive, the MLME responds to a previous MLME-SET.request.

The parameters of the primitive are listed in Table 35.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| AttributeId | integer | Valid attribute IDs as listed in Table 40 | The ID of the attribute to set. |
| AttributeValue | variable | Attribute-specific as defined in Table 40Table 43 | The value of the attribute to set. |
| Status | enumeration | SUCCESS,FAIL\_READ\_ONLY,FAIL\_NON\_EXISTENT,FAIL\_OTHER | Indicates whether setting the PIB attribute was successful or not. |

Table 36: Parameters of the MLME-SET.confirm primitive

* + 1. MLME-SCAN

The MLME-SCAN primitive supports the DME in requesting the MLME to issue a scan for existing OWPANs.

* + - 1. Request

The MLME-SCAN.request is issued by the DME in order to initiate the scanning procedure as described in clause 5.4.1.

The parameters of the primitive are listed in Table 36.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| ScanDuration | integer milliseconds | [1, 65535] | Specifies for how long the device shall listen for incoming frames. |

Table 37: Parameters of the MLME-SCAN.request primitive

* + - 1. Confirm

The MLME-SCAN.confirm primitive is used by the MLME to report the results of a scan to the DME.

The parameters of the primitive are listed in Table 37.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| ResultList | A list of result entries  | Entries as specified in Table 38. | The set of observed OWPANs.  |
| Status | enumeration | SUCCESS,EMPTY,FAIL\_SCAN\_IN\_PROGRESSFAIL\_OTHER | The result of the association process. |

Table 38: Parameters of the MLME-SCAN.confirm primitive

The *ResultList* parameter shall contain a list in which every entry has the elements listed in Table 38.

|  |  |  |
| --- | --- | --- |
| **Detail** | **TYPE** | **Description** |
| OWPAN ID | 48-bit MAC address | The ID of the observed OWPAN |
| Electrical SNR |  | The SNR during the reception of the OWPAN’s frame |
|  |  |  |

Table 39: Scan result entry elements

* + 1. MLME-START

The MLME-START primitive is used to instruct a device MAC to serve as a coordinator and start operation of a new OWPAN.

* + - 1. Request

The MLME-START.request primitive is issued by the DME and received by the MLME and triggers the procedure to start an OWPAN.

The MLME-START.request primitive shall be confirmed by the MLME through a subsequent MLME-START.confirm primitive invocation.

The parameters of the primitive are listed in Table 39.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| Delay | integersuperframe slots | [0, 65535] | The number of superframe slots to wait before starting an OWPAN in the beacon-enabled mode. |

Table 40: Parameters of the MLME-START.request primitive

* + - 1. Confirm

The MLME-START.confirm primitive is issued by the coordinator MLME to report the result of the preceding request to start a new OWPAN.

The parameters of the primitive are listed in Table 40.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| Status | enumeration | SUCCESS,FAIL\_PARAMETER\_ERROR,FAIL\_OTHER | Whether the preceding MLME-START.request primitive was successful or failed. |

Table 41: Parameters of the MLME-START.confirm primitive

* + 1. MLME-STOP

The MLME-STOP primitive is issued by the DME of a coordinator in order to cease operation of a running OWPAN.

* + - 1. Request

The MLME-STOP.request primitive is issued by the DME of an active coordinator to the MLME in order to stop a running OWPAN.

The parameters of the primitive are listed in Table 41.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| Timeout | integermilliseconds | [1, 65535] | The time after which the OWPAN shall be stopped. If the given time has passes since invocation of the primitive, the MLME shall respond with the corresponding MLME-STOP.confirm primitive, indicating success or failure. |
| Force | Boolean | TRUE,FALSE | Whether to stop an OWPAN forcefully. If set to true, the coordinator MLME shall not wait for successfully disassociation of the associated devices. |

Table 42: Parameters of the MLME-STOP.request primitive

* + - 1. Confirm

The MLME-STOP.confirm primitive is issued by the MLME of a coordinator as a response to a preceding MLME-STOP.request primitive.

The parameters of the primitive are listed in Table 42.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Type** | **Value range** | **Parameter description** |
| Status | enumeration | SUCCESS,FORCED,FAILURE | The result of the preceding MLME-STOP.request primitive. |

Table 43: Parameters of the MLME-STOP.confirm primitive

* 1. MAC PIB Attributes

The MAC comprises variables and constants that define its behavior. In this standard, these are referred to as “PIB attributes”.

Table 43 and Table 44 list variable and constant PIB attributes. It provides the attribute name, a description and information about the constant or space of possible values and associated units. Some variables shall be readable and writable via the MLME-GET.request and MLME-SET.request respectively. Whether a variable can be read or written is indicated by a get for read or set for write in the get/set column. Attributes that are purely internal to the MAC are neither readable nor writable.

|  |
| --- |
| **Variable attributes** |
| **Name** | **ID** | **Description** | **get****/****set** | **Bits** | **Unit / Range** |
| **Association and OWPAN membership** |
| *macOwpanId* | 1 | The ID of the OWPAN with which the device is associated. | get | 48 | Valid 48-bit MAC addresses |
|  |  |  |  |  |  |
| *macAssociationTimeout* | 2 | The time after transmitting an association request to the coordinator after which an association response is expected | getset | 16 | integermilliseconds |
| *macSecurityType* | 3 | The security type used by the OWPAN. | get | 8 | IDs from XXX |
| *macDevShortAddress* | 4 | The short address assigned to the dev during association. | get | 16 | integer[1, 65534] |
| *macEdScanThreshold* | 5 | The threshold for energy detection during a passive scan. | getset | [?] | dBm optical power |
| *macDeviceTimeout* | 6 | The duration after which a coordinator assumes a device to be disassociated if it does not receive frames from that device. | getset | 16 | [1, 65545]milliseconds |
| **Beacon-enabled channel access** |
| *macBeaconNumber* | 7 | The number of the current superframe, embedded by the coordinator in the beacon frame. | get | 16 | integer[0, 65535] |
| *macNumSuperframeSlots* | 8 | The total number of superframe slots in a superframe | get | 16 | integer[1, 65535]superframe slots |
| *macCapMaxRetries* | 9 | The maximum retransmission attempts for CAP transmissions. | getset | 8 | integer[1, 255] |
| *macCapSlotLength* | 10 | The number of superframe slots that form a single CAP slot for the slotted ALOHA access in the CAP. | get | 8 | integer [1, 255] superframe slots |
| *macNumCapSlots* | 11 | The total number of CAP slots in the CAP. | get | 8 | integer[1, 255]cap slots |
| *macMaximumCapCw* | 12 | The maximum value for CW in the CAP. | getset | 8 | integer[1, 255]cap slots |

Table 44: Variable MAC PIB attributes

|  |
| --- |
| **Variable attributes (continued)** |
| **Name** | **ID** | **Description** | **get****/****set** | **Bits** | **Unit / Range** |
| **Protected Transmission** |
| *macRetransmitTimeout* | 13 | The duration after which an ACK is required for a transmitted frame. Upon expiration, a MPDU is typically retransmitted. | get | 16 | unsigned integer[1, 65535] µs |
| *macMaxFrameRetries* | 14 | The maximum number of attempted retransmissions, before the transmission of an MPDU is ultimately considered failed. | getset | 8 | TRUE (1),FALSE (0) |

Table 43: Variable MAC PIB attributes (continued)

|  |
| --- |
| **Constant attributes** |
| **Name** | **Description** | **Value** | **Unit** |
| *aSuperframeSlotDuration* | The duration of a single superframe slot. | **1** | µs |
| *aInitialCapCw* | The value to select as contention window for the first retransmission in the CAP. | **1** | CAP slots |
| *aMinCAPLength* | The minimum number of CAP slots in every superframe. | **2** | CAPslots |
| *aClockAccuracy* | The required accuracy of the device system-clock. | **20** | ppm |
| *aMinFragmentSize* | The minimum size of a MSDU fragment. | **64** | octets |
| *aProtectedWindow* | The maximum number of unacknowledged MPDUs to be in-flight. | **1024** | MPDUs |
| *aMaxAssocRetries* | The maximum number of retries for the association through random channel access. | **10** | retries |
| *aMac48Address* | The device’s 48-bit MAC address. | **valid address** | 48-bit MAC address |
| *aGtsTaifs* | The Turn Around Interframe Space used in the beacon-enabled mode. | **-** | - |

Table 45: Constant MAC PIB attributes

* 1. Capabilities

Capabilities formally indicate functionality that is supported, i.e. implemented, by a device. Each capability has a name and a numeric ID with a width of 16 bits. Some capabilities may require other capabilities to be implemented through the device. Capabilities are listen in Table 45.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **ID** | **Description** | **Required capabilities** |
| *capCoordinator* | 0 | The device supports acting as a coordinator. |  |
| *capHbPhy* | 1 | The device supports usage of the HB-PHY. | *capMultipleInputFeedback,**capEffectiveChannelFeedback,**capVariableElements* |
| *capMultiOfeEstimation* | 2 | The device supports orthogonal pilot channel estimation and feedback. |  |
| *capMcsRequest* | 3 | The device supports requesting a modulation and a coding scheme to be used for transmission towards it. |  |
| *capFullDuplex* | 4 | The device supports simultaneous transmission and reception. |  |
| *capBlockAcknowledgment* | 5 | The device supports the block acknowledgment mechanism. |  |
| *capVariableElements* | 6 | The device supports parsing and transmitting of MAC frames containing a variable number of elements. |  |
| *capSecurity* | 7 | The device supports security. |  |
| *capPmPhy* | 8 | The device supports usage of the PM-PHY. |  |
| *capHcm* | 9 | The device supports HCM-coded modulation for the payload. | *capPmPhy* |
| *capInterferenceDetection* | 10 | The device supports the interference detection procedure in clause 5.4.7. |  |
| ***reserved*** | 11-2040 |  |  |

Table 46: MAC capabilities