**IEEE P802.15**

**Wireless Personal Area Networks**

|  |  |
| --- | --- |
| Project | IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs) |
| Title | Resolution to Status comments in 802.15.4md |
| Date Submitted |  July, 17 2019  |
| Source | [Tero Kivinen][] | Voice: []Fax: []E-mail: [kivinen@iki.fi] |
| Re: | LB158 comments |
| Abstract | Proposed resolution how to resolve lots of comments related to status. |
| Purpose | Solve the issue.  |
| 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. |

# Description of Problem

There are several cases where Status is used inconsistently. Firstly there are few MAC commands, and IEs where there is field called Status. Those can be confusing especially in descriptions of the related MLME-calls. Then there is some cases where over the air status is mixed with the local status information. Sometimes this is done in very confusing manner by mapping over the air errors to some local existing error codes..

# Solution

Firstly change all Status fields in MAC commands and IEs to have some kind of prefix.

Secondly separate the remote operation status from the local MLME operation status, by adding another parameter to MLME-primitives.

# Changes

## MLME related status issues



MLME-ASSOCIATE.response, and MLME-ASSOCIATE.confirm uses Status inconsitently.

### MLME-ASSOCIATE.response

MLME-ASSOCIATE.response uses Status when it is actually giving out the Association Status field value for the Associate Respond Command. Rename the Status in .response to AssociationStatus, and change references to correct table.

CID-255 Resolution: Accept, i.e., Change Status to AssociationStatus.

CID-260 Resolution: Revised. Proposed resolution:

In Table 8-6 rename Status to AssociationStatus, change Valid range to "As defined in Table 7-55", and change Description to "The association status of the association attempt as defined in 7.5.3".

CID-257 Resolution: Revised. Proposed resolution:

If the AssocationStatus parameter~~field~~ of MLME-ASSOCIATE.response primitive is set to Fast assocation successful, then association response shall be sent to the device requesting fast association directly.

### MLME-ASSOCIATE.confirm

The MLME-ASSOCIATE.confirm conbines Status field that tells the result of the local operation with the actual association response over the air from the other end. This means we cannot get proper error information from the other end to the next higher layer. Make separate parameter for AssociationStatus, which contains the Assocation Status field value from the Assocation Response command.

CID-259 Resoltion Accept, i.e., Add AssociationStatus parameter to this primitive immediately after the AssocShortAddress paramater.

CID-261 Resolution Accept, i.e., Add AssociationStatus parameter to this primitive with Type "Enumeration", Valid Range "As defined in Table 7-55", and Description of "The association status of the association attempt from association request command as defined in 7.5.3". Add this immediately after the AssocShortAddress paramater.

CID-262 Resolution Accept, i.e., duplicate of CID-261.

## Association command and Status field



Figure 7-111 – Association Response command Content:

There is only Association Status field no Status field.

The text in Section 6.4.1, and 6.4.3 refers it directly as Status field, instead of Assocation Status field. Also CID-255 changed the name of the Status parameter to AssocationStatus to reduce confusion, so do the change here to.

CID-100 Resolution: Accept, i.e., Change “Status field” to “Assocation Status field”.

CID-101 Resolution: Accept, i.e., Change “Status field” to “Assocation Status field”.

CID-106 Resolution: Accept, i.e., Change “Status parameter” to “AssociationStatus parameter”.

CID-107 Resolution: Accept, i.e., Change “Status field” to “Assocation Status field”.

CID-108 Resolution: Accept, i.e., Change “Status field” to “Assocation Status field”.

## Rename Status field to Dsme Gts Status field



DSME GTS Response command has field called Status. This is confusing. See figure 7-124:

Proposal is to rename that field to “Dsme Gts Status field”.

CID-212: Resolution: Accept, i.e., change “Status field” to “Dsme Gts Status field” twice on row 7.

CID-213: Resolution: Accept, i.e., change “Status” to “Dsme Gts Status”.

CID-214: Resolution: Accept, i.e., change title from “Status field values” to “Dsme Gts Status field values”.

CID-215: Resolution: Accept, i.e., change “Status Field” to “Dsme Gts Status field”. The 7.5.16 DMSE GTS Notify command contains DSME GTS Management field which is defined in Figure 7-124, and this text refers to that subfield.

CID-125 – CID-128: Resolution: Accept, i.e., Change Status field" to "Dsme Gts Status field".

## DMSE Status field

The problem is that the MLME-DSME-GTS.response and confirm mixes status from the MAC command coming from the other end and local issues (like CHANNEL\_ACCESS\_FAILURE). The solution is to separate those two functions to separate parameters for primitives.

### MLME-DSME-GTS.response

CID-286: Resolution: Revise. Proposed Resolution: The section number is wrong, it should be 8.2.20.3 instead of 8.2.20.4. Change “Status” to “DsmeGtsStatus” on section 8.2.20.3 page 343 line 24. This is upper layer sending status of dsme operation to MAC, which then encodes it to DSME GTS response command.

CID-288: Resolution: Revised. Proposed Resolution: In Table 8-56 rename “Status” to “DsmeGtsStatus”, change type from Enumeration to Integer, and change Valid range to “As specified in Table 7-58”.

CID-289: Resolution: Accept, i.e., Change text "The Status field of the command shall be set to zero if the Status parameter value is SUCCESS. The Status field shall be set to one if the Status parameter value is DENIED. The Status field shall be set to two if the Status parameter value is INVALID\_PARAMETER" to "The Dsme Gts Status field of the command shall be set to the value DsmeGtsStatus parameter".

### MLME-DSME-GTS.confirm

CID-287: Resolution: Accept, i.e., Add DsmeGtsStatus parameter to primitive in the location where Status now, and move the Status to the end.

CID-295: Resolution: Revised, Proposed Resulution: Add DsmeGtsStatus, with type of integer, valid range “as specified in Table 7-58”, and with description of “The Dsme Gts Status field of DSME GTS Response command”. Change the Description of “Status” to “The status of the request”. Remove DENIED and INVALID\_PARAMETER from the list of Valid Range of Status.

CID-291: Resolution: Revised. Proposed Resolution: Change “If the value of Status field in the command is zero (SUCCESS)” to “If the value of the Status parameter is SUCCESS and the DsmGtsStatus parameter is zero (APPROVED)”.

CID-294: Resolution: Accept, i.e., Replace the text "The Status field of the command shall be set to zero if the Status parameter value is SUCCESS. The Status field shall be set to one if the Status parameter value is DENIED. The Status field shall be set to two if the Status paramter valie is INVALID\_PARAMETER." with "The DsmeGtsStatus parameter shall be set to have value of the Dsme Gts Status field of the command."

CID-297: Resolution: Accept, i.e., Remove text "If the value of the Status field in the command is zero, the Status in this pirmitive shall be set to SUCCESS. If the value of the Status field is one, the Status paramter shall be set to DENIED. If the value of the Status field is two, the Status parameter shall be set to INVALID\_PARAMETER."

## Have separate status parameters for local and remote end



As we renamed the “Status” to “Dsme Gts Status” in DSME GTS Response command and separated “Status” to “Status” and “DsmeGtsStatus” we need to change this text to match.

CID-120: Resolution: Accept, i.e., fix “reponse” to “response”, change “Status parameter” to “DsmeGtsStatus Parameter”, and change “SUCCESS” to “APPROVED”.

CID-121: Resolution: Accept, i.e., change “Status” (field) to “Dsme Gts Status” (field).

CID-122: Resolution: Accept, i.e., change “Status field” to “Dsme Gts Status field”.

CID-123: Resolution: Accept, i.e., change “Status field” to “Dsme Gts Status field”.

CID-124: Resolution: Revised. Proposed resolution: Change text “On receipt of MLME-DSME-GTS.reponse primitive with Status parameter value of DENIED or INVALID\_PARAMETER, the device shall send a DSME GTS Response command to requesting device.” to “On receipt of MLME-DSME-GTS.response primitive with DmseGtsStatus parameter value other than APPROVED, the device shall send a DSME GTS Response command to requesting device.”

CID-129: Resolution: Revised. Proposed resolution: Change text “with a Status of SUCCESS,” to “with a Status parameter of SUCCESS, DsmeGtsStatus parameter copied from the Dsme Gts Status field of the DSME GTS Response command,”.

## Generic DMSE related issues



CID-290: Resolution: Revised. Proposed Resolution: This is already taken care of by change done in CID-288: In Table 8-56 rename “Status” to “DsmeGtsStatus”, change type from Enumeration to Integer, and change Valid range to “As specified in Table 7-58”.

CID-292: Resolution: Revised. Proposed Resolution: Change text “the device shall check the Status field of the command” with “the device shall check the Status and DsmeGtsStatus parameters”.

CID-263: Resolution: Revised. Proposed Resolution: Change text “set to SUCCESS” with “set to SUCCESS, and the AssocationStatus parameter will indicate status of the assocation”. And remove sentence “Otherwise, the Status parameter will be set to indicate the type of failure.”

## Generic Status issues



CID-296: Resolution: Revised. Proposed resolution: Remove newline between “CHANNEL\_ACCESS\_” and “FAILURE”. After that remove it completely as we are removing generic errors in Status lists…

CID-270: Resolution: Accept, i.e., Add text “– INVALID\_INDEX– The index inside the hierarchical values in PIBAttribute is out of range.” after line about INVALID\_PARAMETER.

CID-272: Resolution: Revised. Proposed resolution: Add paragraph before line 5 on the page 314 saying:

If the MLME receives the MLME-SCAN.request primitive with invalid or incorrect bits set in the ScanChannels bitmap, it will not perform the scan and the Status parameter will be set to BAD\_CHANNEL

CID-251: Resolution Revised. Proposed resolution: Add following paragraph after line 10 on page 282:

List of generic security error is given below, and any MLME or MCSP confirm primitive may return them inside the Status parameter even when these errors are not listed in the Valid range column of the Status parameter.

– COUNTER\_ERROR– Returned when sending frame, if the frame counter has maximum value. Also returned when receiving frame where frame counter is smaller than what is received before.

– IMPROPER\_KEY\_TYPE– Returned when the incoming security policy checking notices that key used to protect the frame was not the one that was configured in the security policy.

– IMPROPER\_SECURITY\_LEVEL– Returned when the incoming security policy checking noticed that security level of the incoming frame is not allowed by security policy.

– SECURITY\_ERROR– Returned when unsecuring of the frame fails in the incoming security process, for example because the MIC is incorrect.

– UNAVAILABLE\_KEY– Returned when outgoing security process cannot find key requested by the MLME primitive, or when the incoming security process cannot find key indicated in the frame.

– UNSUPPORTED\_LEGACY– Returned when secured frame is received with Frame Version field set to zero.

– UNSUPPORTED\_SECURITY– Returned when security is requested for outgoing frame, or when secured frame is received and the security is not enabled in the device. Also received if the Auxiliary Security Header has security level of zero in it.

–

The text already covers generic errors: INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK. Add very short description and remove them from the MLME-primitives.

Add better description explaining FRAME\_TOO\_LONG, TRANSACTION\_EXPIRED, TRANSACTION\_OVERFLOW, and NO\_DATA after line 20 on page 282:

List of generic errors is given below, and any MLME or MCSP confirm primitive may return them inside the Status parameter even when these errors are not listed in the Valid range column of the Status parameter.

– INVALID\_PARAMETER– Some of the parameters are not supported or are out of range.

– CHANNEL\_ACCESS\_FAILURE– CSMA-CA algorithm fails.

– NO\_ACK– NO acknowledgement is received when it is expected.

– FRAME\_TOO\_LONG– The length of the frame exceeds the maximum size it can be. This might be because frame got expanded because IEs are added, or because of security processing. It is also returned if requested transaction is too large to fit in the CAP or GTS.

– TRANSACTION\_EXPIRED– Returned when response is expected but it not received within the expected time, or when the critical transaction is not sent out in time.

– TRANSACTION\_OVERFLOW– Returned when there is no capacity to store another transaction.

–

Note, that especially TRANSACTION\_EXPIRED, and TRANSACTION\_OVERFLOW has some text in standard which is bit vague, and might require fixing, or removal. NO\_DATA is not really generic error message, so I did not move it here.

–

Remove references to generic errors in the MLME primitives:

Table 8-7 MLME-ASSOCIATE.confirm:

* Remove COUNTER\_ERROR, IMPROPER\_KEY\_TYPE, IMPROPER\_SECURITY\_LEVEL, SECURITY\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_LEGACY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK, FRAME\_TOO\_LONG.
* Change to “SUCCESS, NO\_DATA, Also see 8.2.2”

Table 8-10 MLME-DISASSOCIATE.confirm:

* Remove COUNTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK, FRAME\_TOO\_LONG, TRANSACTION\_EXPIRED, TRANSACTION\_OVERFLOW.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-12 PANDescriptor:

* Remove COUNTER\_ERROR, IMPROPER\_KEY\_TYPE, IMPROPER\_SECURITY\_LEVEL, SECURITY\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_LEGACY, UNSUPPORTED\_SECURITY.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-13 MLME-COMM-STATUS.indication:

* Remove COUNTER\_ERROR, IMPROPER\_KEY\_TYPE, IMPROPER\_SECURITY\_LEVEL, SECURITY\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_LEGACY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK, FRAME\_TOO\_LONG, TRANSACTION\_EXPIRED, TRANSACTION\_OVERFLOW.
* Change to “SUCCESS, IMPROPER\_IE\_SECURITY, Also see 8.2.2”

Table 8-17 MLME-GET.confirm and Table 8-19 MLME-SET.confirm:

* Leave as it is.

Table 8-21 MLME-GTS.confirm:

* Remove COUNTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, FRAME\_TOO\_LONG.
* Change to “SUCCESS, DENIED, NO\_SHORT\_ADDRESS, NO\_DATA, Also see 8.2.2”

Table 8-28 MLME-RX-ENABLE.confirm:

* Remove INVALID\_PARAMETER.
* Change to “SUCCESS, PAST\_TIME, ON\_TIME\_TOO\_LONG, RANGING\_NOT\_SUPPORTED, Also see 8.2.2”

Table 8-30 MLME-SCAN.confirm:

* Remove COUNTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, FRAME\_TOO\_LONG.
* Change to “SUCCESS, LIMIT\_REACHED, NO\_BEACON, SCAN\_IN\_PROGRESS, BAD\_CHANNEL, Also see 8.2.2”

Table 8-32 MLME-START.confirm:

* Remove COUNTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, FRAME\_TOO\_LONG.
* Change to “SUCCESS, NO\_SHORT\_ADDRESS, SUPERFRAME\_OVERLAP, TRACKING\_OFF, Also see 8.2.2”

Table 8-36 MLME-POLL.confirm:

* Remove CONTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK, FRAME\_TOO\_LONG.
* Change to “SUCCESS, NO\_DATA, Also see 8.2.2”

Table 8-43 MLME-BEACON.confirm:

* Remove INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, FRAME\_TOO\_LONG.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-46 MLME-SET-SLOTFRAME.confirm:

* Remove INVALID\_PARAMETER.
* Change to “SUCCESS, SLOTFRAME\_NOT\_FOUND, MAX\_SLOTFRAMES\_EXCEEDED, Also see 8.2.2”

Table 8-48 MLME-SET-LINK.confirm:

* Remove INVALID\_PARAMETER.
* Change to “SUCCESS, UNKNOWN\_LINK, MAX\_LINKS\_EXCEEDED, Also see 8.2.2”

Table 8-52 MLME-KEEP-ALIVE.confirm:

* Remove INVALID\_PARAMETER.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-57 MLME-DSME-GTS.confirm:

* Remove INVALID\_PARAMETER, NO\_ACK, CHANNEL\_ACCESS\_FAILURE.
* Change to “SUCCESS, NO\_DATA, Also see 8.2.2”

Table 8-60 MLME-DSME-LINK-REPORT.confirm:

* Remove CHANNEL\_ACCESS\_FAILURE, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-63 MLME-PHY-OP-SWITCH.confirm:

* Remove CONTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK, FRAME\_TOO\_LONG, TRANSACTION\_EXPIRED, TRANSACTION\_OVERFLOW.
* Change to “SUCCESS, UNSUPPORTED\_FEATURE, Also see 8.2.2”

Table 8-67 MLME-DBS.confirm:

* Remove UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-70 MLME-DA.confirm:

* What is the meaning of FAILURE error code?

Table 8-74 MLME-RIT-RES.confirm:

* Remove INVALID\_PARAMETER, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-77 MLME-SRM-REPORT.confirm:

* Remove CHANNEL\_ACCESS\_F AILURE, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-80 MLME-SRM-INFORMATION.confirm:

* Remove CHANNEL\_ACCESS\_FAILUR E, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-85 MLME-SRM-RES.confirm:

* Remove CHANNEL\_ACCESS\_FAI LURE, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-86 MLME-SRM-REQ.confirm:

* Remove CHANNEL\_ACCESS\_ FAILURE, NO\_ACK.
* Change to “SUCCESS, Also see 8.2.2”

Table 8-89 MCPS-DATA.confirm:

* Remove COUNTER\_ERROR, UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, NO\_ACK, FRAME\_TOO\_LONG, TRANSACTION\_EXPIRED, TRANSACTION\_OVERFLOW.
* Change to “SUCCESS, INVALID\_ADDRESS, INVALID\_GTS, UNSUPPORTED\_FEATURE, UNSUPPORTED\_PRF, UNSUPPORTED\_RANGING, UNSUPPORTED\_PSR, UNSUPPORTED\_DATARATE, UNSUPPORTED\_LEIP, ACK\_RCVD\_NODSN\_NOSA, Also see 8.2.2”

Table F.5 MLME-TRLE-MANAGEMENT.confirm:

* Remove UNAVAILABLE\_KEY, UNSUPPORTED\_SECURITY, INVALID\_PARAMETER, CHANNEL\_ACCESS\_FAILURE, FRAME\_TOO\_LONG.
* Change to “SUCCESS, SLOT\_FULL, RELAY\_FULL, NOT\_FOUND, NOT\_CONFIRMED, Also see 8.2.2”

## MCSP-DATA related issues



The MCSP-DATA.confirm have lots of error codes which are not described anywhere. Some of those are obvious some are not so. Add text on page 386 after line 8:

If some parameters to the MCSP-DATA.request are not supported the MAC indicates that calling MCSP-DATA.confirm with following error codes:

– UNSUPPORTED\_DATARATE– The DataRate requested is not supported.

– UNSUPPORTED\_LEIP– The LocationEnhancingInforationPostamble or LocationEnhancingInformationPostambleLength parameter values are not supported.

– UNSUPPORTED\_PRF– The UwbPrf parameter value is not supported.

– UNSUPPORTED\_PSR– The UwbPreambleSymbolRepetitions value is not supported.

This Resolves CID-356, CID-357, CID-358, CID-359 with Resolution of Revised, and Proposed change as above

CID-360 is still left to be resolved, as there is no suitable text for it yet: We need to add something like:

– UNSUPPORTED\_RANGING– XXX description of error.

But I do not know when this error is returned, is it that there is something wrong in the ranging parameters, or is that Ranging is not supported at all, or something else.

## MLME-SOUNDING related status issues



CID-266: Resolution: Accept, i.e., Replace SOUNDING\_NOT\_SUPPORTED With UNSUPPORTED\_ATTRIBUTE.

CID-277: Resolution: Revised. Proposed change: Already taken care of by CID-266, i.e., replace SOUNDING\_NOT\_SUPPORTED with UNSUPPORTED\_ATTRIBUTE.

CID-278: Resolution: Reject. UNSUPPORTED\_ATTRIBUTE is quite acceptable error for case where calibrate is not supported. The same error is used also for MLME-SOUNDING. There is no need to add separate error saying SOUNDING\_NOT\_SUPPORTED.

CID-279: Resolution: Reject. UNSUPPORTED\_ATTRIBUTE is quite acceptable error for case where calibrate is not supported. The same error is used also for MLME-SOUNDING. There is no need to add separate error saying SOUNDING\_NOT\_SUPPORTED.