

E1.20 Review 4.2

Approved PR4 comment resolutions - CP/2023-1012r1

Comment Number: 1

Commentor: PN

Section: General

Previous Review Comment Number:

Decision: Reject

Comment:

In future please add draft PID values to allow trial implementations and interoperability testing of them. This has been done in the past and successfully flagged up issues which don't really become apparent until you try and implement it. Unless I've been particularly unobservant there doesn't appear to be a public review diff at all for this cycle (either as a past commenter or on the website), but in general they're useful for everyone not just previous commenters. While the standard continues to improve I think it would benefit from another public review after it's gone into a bugfixes only state where no brand new PIDs(!) or major changes are made.

TG Comment:

The diff has always been produced.

Comment Number: 2

Commentor: PN

Section: Various

Previous Review Comment Number:

Decision: Accept

Comment:

The document seems to be inconsistent about what NSC packets are called (or at least how it's capitalised). In the overview alone we've got both ""NULL START CODE"" and ""NULL START Code"", contrasted with ""Alternate START Code (ASC)"". Later on in 10.11.7 we complete the set with ""Null Start Code""! Personally I think it should be Null Start Code, or maybe NULL START CODE, but please could we make it consistent throughout?

TG Comment:

DMX512 standard uses NULL START Code and Alternate START CODE and so that have been implemented in this document.

Comment Number: 3

Commentor: PN

Section: Various

Previous Review Comment Number:

Decision: Accept

Comment:

As with NSC but for Break, there is variable/inconsistent case used.

TG Comment:

The TG has attempted to resolve all the stated inconsistencies

Comment Number: 4

Commentor: PW

Section: Table of Contents

Previous Review Comment Number:

Decision: Accept

Comment:

7.5 Discovery Unique Branch – typo in PID Name

TG Comment:

The ToC had not auto-refreshed. This has been fixed.

Comment Number: 5

Commentor: PW

Section: 3.2.5 Discovery Mark Time

Previous Review Comment Number:

Decision: Accept

Comment:

This requirement needs to be a SHALL. In Plugfest after Plugfest I continue to see issues with Discovery when responders fail to implement this. It may be aggravated by the use of 3/3.6V transceiver parts

TG Comment:

The TG agrees with the commentor and changes have been implemented.

Comment Number: 6

Commentor: PW

Section: 6.2.7.1 Port ID

Previous Review Comment Number:

Decision: Accept

Comment:

Second sentence: "Responders may respond ..." should become "Responders shall respond ..." The current wording still leaves ambiguity. Do we expect Responders NOT to respond at all, or to essentially ignore the PortID field and respond anyway? It is important to clarify this, as many early CONTROLLERS failed to set this field correctly (ie sent a zero, and so Responders have been backed into "non-compliance").

TG Comment:

The TG agrees with the commentor and changes have been implemented.

Comment Number: 7

Commentor: PN

Section: 6.2.8.1 Controller Flags / Message Count field for Controller Generated Messages

Previous Review Comment Number: 22

Decision: Accept in principle

Comment:

Previous comment 22. ""Actioned in document. Any undefined controller flags set – NR_FORMAT_ERROR."" I don't see this change in the document out for review?

TG Comment:

Corrected.

Comment Number: 8

Commentor: PW

Section: 6.2.8.1 Controller Flags

Previous Review Comment Number:

Decision:

Comment:

What ERROR message should a Responder use if it receives bits set in this field and has NOT previously been asked to verify support for this field using the CONTROLLER_FLAG_SUPPORT PID ? I suggest FORMAT Error – but whatever the consensus, it needs to be defined here.

TG Comment:

The TG agrees and has added text: ""Responders receiving a request with a Controller Flag field set, for which the Responder did not declare support using the CONTROLLER_FLAG_SUPPORT Parameter Message, shall respond with a NACK Response with NACK Reason Code of NR_DATA_OUT_OF_RANGE.""

Comment Number: 9

Commentor: PW

Section: 6.2.10.1 Command Class

Previous Review Comment Number:

Decision: Accept

Comment:

Bottom of page 24 "Responders shall not respond to non-discovery commands sent using Broadcast Addressing in order ..."

This implies it is allowable to respond to a Broadcast DISCOVERY_COMMAND class commands which is NOT generally true. It is only permitted to reply to a Broadcast Discovery Unique Branch DISCOVERY_COMMAND.

It is NOT permitted to reply to a Broadcast Discovery MUTE/UNMUTE DISCOVERY_COMMAND.

Suggested replacement text :

Responders shall not respond to Broadcast GET_COMMAND or SET_COMMAND messages, in order to prevent collisions.

Responders shall not respond to Broadcast DISCOVERY_COMMAND messages EXCEPT for the case of DISCOVERY_UNIQUE_BRANCH messages.

second last paragraph "...shall ignore and ..."

Ignore what ??

TG Comment:

Suggestions have been added.

Comment Number: 10

Commentor: PW

Section: 6.2.10.2.1

Previous Review Comment Number:

Decision: Accept

Comment:

Minimum Required Paramater Support ... the "Required" column of Table A-3 is ambiguous. There are in fact two columns labelled "Required".

TG Comment:

Changed to plural.

Comment Number: 11

Commentor: PW

Section: 6.3.3 Acknowledge Timer

Previous Review Comment Number:

Decision: Accept

Comment:

The example shows that a device requires approximately 60 seconds to process the request. However the example shows "80 seconds later" that the controller sends a GET_QUEUED_MESSAGE. There is no explanation as to why this is 80 seconds, or how it is related to the earlier ACK_TIMER response. The example would make more sense if it was worded "At least 60 seconds later (i.e after the returned ACK_TIMER value)"

TG Comment:

The TG changed text to: ""At least 60 seconds later, the controller ...""

"Comment Number: 12

Commentor: PW

Section: 6.3.5

Previous Review Comment Number:

Decision: Accept in part

Comment:

Acknowledge Timer Hi res

No mention that this may only be used if

- (i) The Responder has declared support via the CONTROLLER_FLAGS_SUPPORT PID
- (ii) The Controller has actually asked the Responder for its supported flags
- (iii) The Controller has set the requisite bits in the Message Count/Controller Flags field.

Please add text along these lines

TG Comment:

The TG changed text to: ""The RESPONSE_TYPE_ACK_TIMER_HI_RES response type shall only be used when the Controller flag HiResAckTimerSupport is set in the controller packet and the responder has implemented support for Acknowledge Timer Hi Res. ""

Comment Number: 13

Commentor: PN

Section: 7.5 Discovery Unique Branch Message

Previous Review Comment Number: 81

Decision: Accept

Comment:

Previous comment 81. This doesn't seem to have actually been changed, but only in the table of contents, the typo is still present. Does that need regenerating or similar?

TG Comment:

Table has been regenerated.

Comment Number: 14

Commentor: PW

Section: 7.7 Discovery Algorithm

Previous Review Comment Number:

Decision: Accept in part

Comment:

Whilst I welcome the addition of Fig7-2, it is to my mind a little misleading/simplistic.

- (1) It fails to show the need to UNMUTE to start.
- (2) The return from proxy device-no goes to "lower=upper", should it not return to the "Search Range" box ?
- (3) One box shows "Send MUTE cmd" (there is such a cmd), but another box shows "Send DISC cmd" (there is no such command, it is Discovery Unique Branch (DUB))
- (4) My reading of this flowchart is that it would STOP at the bottom of the first branch where upper=lower.

TG Comment:

Drawing has been redrafted to resolve.

"Comment Number: 15

Commentor: PW

Section: 8.3 Response Messages

Previous Review Comment Number:

Decision: Withdrawn

Comment:

Comments from 6.3.3 apply here as well.

TG Comment:

Withdrawn.

Comment Number: 16

Commentor: PW

Section: 9.2.3 Required Sub-Device Messages

Previous Review Comment Number:

Decision: Accept in principle

Comment:

Mention of the SUPPORTED_PARAMETERS_ENHANCED belongs here.

TG Comment:

The TG edited text to include SUPPORTED_PARAMETERS_ENHANCED.

Comment Number: 17

Commentor: PN

Section: 10.1.1

Previous Review Comment Number:

Decision: Reject

Comment:

I think it would be good to just explicitly clarify that the length restrictions are bytes not characters and the impact this may have on string lengths given this might trip people up around Unicode, it's sort of been alluded to with the Thai example. It did catch me out at first, especially as a Latin character was being converted to two bytes of UTF-8.

TG Comment:

The TG feels the text is already explicit.

Comment Number: 18

Commentor: PW

Section: 10.1.1 Unicode

Previous Review Comment Number:

Decision: Accept

Comment:

The first sentence says All responders shall support ASCII character encoding.

The next paragraph refers to responses that only support Unicode, which seems to contradict the first sentence. (and the second sentence, which refers to “additionally” support Unicode).

TG Comment:

The error has been corrected.

Comment Number: 19

Commentor: PW

Section: 10.3.1 QUEUED_MESSAGE

Previous Review Comment Number:

Decision: Accept in part

Comment:

Controller GET example "The Status Type Requested is one of"

The forward reference to Table 10-1 for "as allowed" is still not clear, that table is a list of allowed response types.

It would be much clearer to have an additional table of allowed Status Type requests, and place it right here in section 10.3.1.

STATUS_NONE, STATUS_ERROR, STATUS_WARNING, STATUS_ADVISORY or STATUS_GET_LAST

The example on page 53 "Controller sends GET:QM (No QM pending) is separate to the previous example. If possible, please format so that GET and response are on the same page.

It needs a new introduction text :

"In the event that a controller sends a GET:QM and there are no pending messages, the responder shall return any relevant Status Messages, in the format described in section 10.3.2"

I feel it would be better if the example actually showed what would be returned in the event there were also NO Status Messages, as the current example shown makes use of fields that have not yet been described, as they are in 10.3.2.

TG Comment:

The TG reworked this section. Regarding first point: ""In the following example we illustrate that when a controller sends a GET:QUEUED_MESSAGE and there are no pending messages, the responder returns any outstanding Status Messages, in the format described in section 10.3.2."" See document for changes related to second point.

Comment Number: 20

Commentor: PW

Section: 10.3.2 Status Messages

Previous Review Comment Number:

Decision: Accept

Comment:

The text “The Status Type of STATUS_NONE” ... belongs in section 10.3.1 under GET:QM, as it was always the intention that GET:QM was the mandated PID, and thus it is GET:QM that should be used when a controller wants to establish whether a device is present.

Whilst it could be used with STATUS_MESSAGES, the likely response from many responders will be NACK_UNSUPPORTED_PID.

TG Comment:

The TG moved the paragraph to 10.3.1. The Status Type of STATUS_NONE shall be used when a controller wants to establish whether a device is present on the network without retrieving any Status Message data from the device.

Comment Number: 21

Commentor: PW

Section: 10.3.2.2 Status Type

Previous Review Comment Number:

Decision: Accept

Comment:

The Note re earlier versions not allowing STATUS_NONE applies to 10.3.1 and should be moved there, associated with the new table of allowed request types. This section would then logically explain the response types, as embedded in the packed response.

TG Comment:

The TG moved the note to just below the above insertion.

Comment Number: 22

Commentor: PN

Section: 10.4.2 Get Parameter Description

Previous Review Comment Number:

Decision: Accept in principle

Comment:

""Data Type defines the size of the data entries in the PD of the message for this PID. For example: unsigned 8-bit byte versus signed 16-bit word. For historical reasons, this PID cannot be used with integer values greater than 32 bits. Table A-15 enumerates the field codes.""the table then includes a bits column and data types including DS_UINT32 (32 bit), DS_INT64 (64 bit) and DS_ipv6 (128 bit). I assume the signed and unsigned integers are included in this exception, but please could we have a column to clarify exactly which values are allowed and disallowed to avoid any confusion. Or headings in the table or something.

I'm assuming this is because of the min/max/default fields?

TG Comment:

TG changed text as follows: Data Type defines the size of the data entries in the PD of the message for this PID. For example: unsigned 8-bit byte versus signed 16-bit word. For historical reasons, this PID cannot be used with integer values greater than 32 bits.

Note: For a list of all field codes see Table A-15 enumerates the field codes.

Comment Number: 23

Commentor: PN

Section: 10.4.2 Get Parameter Description

Previous Review Comment Number:

Decision: Accept in principle

Comment:

Min/Max/Default have ""This field has no meaning for non-integer Data Types."" I think this used to be the bit that mentioned DS_BIT_FIELD, given the potential confusion about whether or not that's an integer, adding it back, or adding the table sections as mentioned above would be beneficial I believe. I'm assuming DS_ENUMERATION is also allowed to use these fields, otherwise you've got a potentially huge search space to find all the available enum values.

TG Comment:

The TG has modified the text to call out the explicit data types for which this is valid.

Comment Number: 24

Commentor: PN

Section: 10.4.2 Get Parameter Description

Previous Review Comment Number:

Decision: Accept

Comment:

""The meaning of Type is dependent upon Data Type. When Data Type is set to DS_ENUMERATION this field is used to access a Helper PID."" it doesn't tell you what to do when it's not DS_ENUMERATION; the old standard had ""This field no longer has any meaning and should be filled with 0x00 in the response. Controllers should ignore the contents of this field."" which I think should be re-included for the other cases.

TG Comment:

The text has been updated.

Comment Number: 25

Commentor: PN

Section: 10.4.2 Get Parameter Description

Previous Review Comment Number:

Decision: Accept in principle

Comment:

IMHO ""DS_ENUMERATION"" has been shoehorned in and needs a lot of work to tidy it up and make it possible to implement it accurately. I note at least the following:

""Table A-15: Data Type Defines"" has it as 32 bits which I think means the enum could be 32 bits, but the PDL of the get and set for HELPER_PID implies its only 8 bits. [See Newman 10.4.3 Helper_PID comments.]

TG Comment:

The TG has reworked this. See comment 27.

Comment Number: 26

Commentor: PN

Section: 10.4.2 Get Parameter Description

Previous Review Comment Number: 72

Decision: Accept

Comment:

Previous comment 72. This doesn't seem to have actually been changed, the full stops to show the abbreviations aren't present.

TG Comment:

Error has been corrected.

Comment Number: 27

Commentor: PN

Section: 10.4.3 Get Helper PID

Previous Review Comment Number:

Decision: Accept in principle

Comment:

I think this PID could be better named, in terms of the namespace of all defines it's a rather generic name and unclear what it does. Something like ENUM_DESCRIPTION (or definitely something ending in _DESCRIPTION) would better match the other PIDs that already exist in the standard. Also the presence of the abbreviation PID in the name is unusual when it's not currently referencing another PID (compared to the PACKED_PID_* PIDs).

Also the section title of the PID ""Get Helper PID"" doesn't seem to actually define what it does ""get a descriptive text label for a given manufacturer specific enumerated value""

TG Comment:

The TG has evolved a more elegant way to get enumeration text and over a 32-bit range.

Comment Number: 28

Commentor: PN

Section: 10.4.3 Helper PID

Previous Review Comment Number:

Decision: Accept in principle

Comment:

The HELPER_PID only lets you pass in the enumeration number requested, not a PID value, which implies you can only use it for one enumeration, but that restriction isn't listed anywhere. Or at least enumerations have to be unique within that manufacturer?

There are no details in HELPER_PID to explain how to use or implement it. Is it actually defining a template PID behaviour that you implement in the manufacturer specific PID 0x8001 (or whatever the type field has)? This isn't explained anywhere if so?

There is no define for the HELPER_PID value listed, current or to be assigned!

""E.g. PID # Requested =0x8000, Data Type = DS_ENUMERATION, Type = 0x01. The Helper PID = 0x8001.""I'm assuming this means the type field has an offset from the manufacturer specific PID requested to the PID value for the helper PID, adding the word ""offset""would make this much clearer. Presumably this means the Type field must be 0x01 or greater when used with DS_ENUMERATION? Should 0x00 be explicitly be excluded in this case, or does it imply no Helper PID? Are there any other special values (e.g. 0xFF)?

FWIW the comment resolution document with the original design partially clarifies some of this, but it needs firming up and transferring into the standard.

TG Comment:

The TG has reworked this. See comment 27.

Comment Number: 29

Commentor: PW

Section: 10.4.3 Helper PID

Previous Review Comment Number:

Decision: Accept in principle

Comment:

This section needs more explanation as to it's application.

I would like to be able to declare text for enumerated values, but the text is specific to each of my Manufacturer PIDS. So I think this PID is missing a field identifying the PID # my enumeration requests relate to.

TG Comment:

The TG has reworked this. See comment 27.

Comment Number: 30

Commentor: PN

Section: 10.4.4 Get Supported Parameters Enhanced

Previous Review Comment Number: 10

Decision: Accept

Comment:

Previous comment 10. ""For details of PACKED_PID_SUB see Section 10.4.8. For details of PACKED_PID_INDEX see Section 10.4.8."" , they're now ordered as expected thanks, but PACKED_PID_SUB is currently 10.4.7!

TG Comment:

Corrected.

Comment Number: 31

Commentor: PN

Section: 10.4.4 Get Supported Parameters Enhanced

Previous Review Comment Number: 69

Decision: Accept

Comment:

Previous comment 69. The Sub-device field in the response is still non-standard, it should be ""Copy of Controller SD"" so you can tell what the request was.

TG Comment:

Corrected.

Comment Number: 32

Commentor: PW

Section: 10.4.4 SUPPORTED_PARAMETERS_ENHANCED

Previous Review Comment Number:

Decision: Reject

Comment:

The document would read better if this appeared immediately after 10.4.1 (SUPPORTED_PARAMETERS)

I do NOT accept that this needs to be a mandated PID. This is an unacceptable overhead on existing responders that wish to maintain compliance with a basic RDM standard, especially given the expiry of the original E1.20 as an ANSI standard. It should ONLY be required if I wish to support the PACKED_PID commands.

TG Comment:

The decision on position in the document was to avoid renumbering sections that had been previously published. The mandate question was voted upon in the CPWG 9/2023 and the reject was approved.

Comment Number: 33

Commentor: PW

Section: 10.4.5 Controller Flag

Previous Review Comment Number:

Decision: Reject

Comment:

There is no table 6-3a.

TG Comment:

There is a Table 6-3a. The TG decision to name it such was to ensure that external documentation referencing tables in the document did not become stale.

Comment Number: 34

Commentor: PW

Section: 10.5.1 DEVICE_INFO

Previous Review Comment Number:

Decision: Accept in principle

Comment:

At the very least, the minor release version should be incremented from 0x00 to 0x01. This is a new version of the standard. Responders implementing in accordance with this version should have a clear manner to show that, especially since the previous version is no longer an ANSI standard. This is informative, so surely should not break any controllers.

TG Comment:

The TG has deprecated the field.

Comment Number: 35

Commentor: PN

Section: 10.6.3 Get/Set DMX512 Starting Address

Previous Review Comment Number: 87

Decision: Accept

Comment:

Previous comment 87. ""The document does address this scenario. 0xffff is returned. Group feels this is an unambiguous response."" The clarification requested was around SET (the response only refers to GET). Which NACK should it respond with?

NR_DATA_OUT_OF_RANGE seems sane to me too, but it's not currently spelt out in the standard.

TG Comment:

Implemented.

"Comment Number: 36

Commentor: PN

Section: 10.6.3 Get/Set DMX512 Starting Address

Previous Review Comment Number:

Decision: Accept

Comment:

Please use consistent field names across parameter data, between here and DEVICE_INFO. Changing both to DMX512 Start Address seems ideal. See also <https://github.com/ssilverman/rdm-schema/issues/23> and <https://github.com/ssilverman/rdm-schema/pull/24/files>

TG Comment:

The document used both DMX512 Start Address and DMX512 Starting Address. The former is now used throughout.

Comment Number: 37

Commentor: PN

Section: 10.6.4 Get Slot Info

Previous Review Comment Number: 46

Decision: Accept

Comment:

Previous comment 46. In ""Example Response for SLOT_INFO – A Moving Head"" slot 8 still features the old index rather than the new quantum define.

TG Comment:

Corrected text.

Comment Number: 38

Commentor: PN

Section: 10.7.2: Get/Set Sensor

Previous Review Comment Number: 13

Decision: Accept

Comment:

Previous comment 13. ""When a sensor is reset, the values for Lowest Detected Value, Highest Detected Value and Recorded Value shall all be set to the Present Value.""please add some words similar to ""for each of those values which are supported""to the end of that sentence.

TG Comment:

TG added commentor's text.

Comment Number: 39

Commentor: PN

Section: 10.11.1 Get/Set Identify Device

Previous Review Comment Number: 79

Decision: Accept

Comment:

Previous comment 79. This doesn't seem to have actually been changed, the typo is still present.

TG Comment:

Corrected.

Comment Number: 40

Commentor: PW

Section: 10.11.4 Self Test

Previous Review Comment Number:

Decision: Accept

Comment:

Have I missed something – or does this version still require controller trial and error to determine how many self tests are supported ? A SELF_TEST capability PID would allow reporting of which self test is currently active out of “N” supported. Also should state if Self tests are numbered contiguously or otherwise.

TG Comment:

The TG resolved this by the addition of new PID: SELFTEST_ENHANCED.

Comment Number: 41

Commentor: PN

Section: 10.5.1 Get Device Info

Previous Review Comment Number:

Decision: Yes

Comment:

Please use consistent field names across parameter data, between here and DMX_PERSONALITY. Changing both to Current DMX512 Personality seems ideal. See also <https://github.com/ssilverman/rdm-schema/issues/23> and <https://github.com/ssilverman/rdm-schema/pull/24/files>

TG Comment:

Comment Number: 42

Commentor: PN

Section: Appendix A Table A-3

Previous Review Comment Number: 53 / 67

Decision: Accept

Comment:

Table A-3: RDM Categories/Parameter ID Defines Previous comments 53 and 67. For the comment field for QUEUED_MESSAGE and LAMP_ON_MODE there is a formatting inconsistency which is still present.

TG Comment:

Font sizes are correct in Table A-3

Comment Number: 43

Commentor: PW

Section: Appendix A Table A-3

Previous Review Comment Number:

Decision: Accept

Comment:

CONTROLLER_FLAG_SUPPORT It is stated that support is only required if Controller Flags implemented – but it really relate to whether the Responder is supporting Hi Res ACK_TIMER and or Uincode.

TG Comment:

The TG addressed this edit in other comment resolutions.

Comment Number: 44

Commentor: PN

Section: Appendix A Table A-6

Previous Review Comment Number: 103

Decision:

Comment:

Table A-6: Product Detail Defines Previous comment 103. Doesn't this input category just duplicate the PRODUCT_CATEGORY_MONITOR options? Also doesn't this duplicate info that can already be fetched via the existing sensor PIDs? If it was kept, shouldn't the equivalent set of options from the category at least be mapped across? Or PRODUCT_DETAIL_INPUT dropped given details are inherently not generic?

TG Comment:

The comment is comparing table A-5 and A-6 and there is no required relationship between them. The TG feels this is doing no harm.

Comment Number: 45

Commentor: PW

Section: Appendix A Table A-8

Previous Review Comment Number:

Decision: Accept

Comment:

Formatting issue/page overflow

TG Comment:

Table has been corrected to carry headings across page break.

Comment Number: 46

Commentor: PW

Section: Appendix A Table A-9

Previous Review Comment Number:

Decision: Reject

Comment:

Incorrect font size for Table A-9 under Lamp On Mode.

TG Comment:

Font sizes are correct in Table A-9

Comment Number: 47

Commentor: PN

Section: Appendix A Table A-12

Previous Review Comment Number:

Decision: Accept in principle

Comment:

Table A-12: Sensor Type Defines This draft of the standard adds ""SENS_TANK_LEVEL"", but that's a sensor usage not a type, we've got ""SENS_VOLUME"" already, we could possibly add a new ""SENS_DEPTH"" although we've already got ""SENS_LENGTH"" which is pretty similar or ""SENS_PRESSURE"" if it's being measured from below. Otherwise I'd need SENS_CYLINDER_LEVEL if it's CO2 in a bottle. The sensor name should be what defines what it's measuring.

TG Comment:

The TG deleted SENS_TANK_LEVEL

Comment Number: 48

Commentor: PN

Section: Appendix A Table A-13

Previous Review Comment Number: 91

Decision: Accept

Comment:

Table A-13: Unit Defines. Previous comment 91. UNITS_VOLTS* are still inconsistently plural, they should be singular too.

TG Comment:

Typo's corrected.

Comment Number: 50

Commentor: PN

Section: Appendix A Table A-13

Previous Review Comment Number: 96

Decision: Reject

Comment:

Table A-13: Unit Defines. Previous comment 96. ""AC / DC are not units."" so why are they in the units defines then, rather than just the base UNITS_VOLT/UNITS_AMPERE as suggested?

TG Comment:

The TG decided that whilst the commenter has a valid point, the problem dates back to the original standard and changing enums now will cause too many problems. Additionally, adding a VOLTS will be no benefit as it simply gives more ambiguous options.

Comment Number: 51

Commentor: PN

Section: Appendix A Table A-13

Previous Review Comment Number: 15

Decision: Reject

Comment:

Previous comment 15. Possibly I'm flogging a dead horse here, but the group rejected it with ""The group feels that the protocol does the job of data transfer. This is an UI / implementation issue."" I'd argue that with the existing unit and prefix defines, the protocol does data transfer and UI, it tells me to render something as 0.01 milli seconds or 5 tera bytes. It even includes some special cases for when to override it ""When a prefix is used with SENS_MASS, note that the UNIT is kilogram. The prefix PREFIX_MILLI is used to denote grams."" , so converting 3,600 kilo seconds to 1,000 hours on my controller display is the only time I'd be ignoring what the responder told me to do and not just rendering it directly. Even if I wanted to I've only got the range and normal values to give me any heuristics as to whether I should, when the responder could just tell me instead and I wouldn't need to guess.

TG Comment:

The TG felt the previous comment resolution stands.

"Comment Number: 52

Commentor: PN

Section: Appendix A Table A-14

Previous Review Comment Number: 59

Decision: Accept in part

Comment:

Section Table A-14: Unit Prefix - Type: Clarification. Previous comment 59. ""It doesn't break it."" Please define what a picobyte is then (i.e. 10^{-12} th of a byte)!

TG Comment:

The TG decided added clarification:

Certain combinations of Prefix and Unit may be ambiguous or non-sensical and their use is discouraged. Example picoByte.

"Comment Number: 53

Commentor: PN

Section: Appendix A Table A-14

Previous Review Comment Number: 60b

Decision: Accept in part

Comment:

Section Table A-14: Unit Prefix - Type: Improvement. Previous comment 60b. Since the various SENS_LOG_RATIO* sensor types have been added, there's no equivalent notes at the bottom of the table clarifying ""The prefix PREFIX_DECA is used to denote bels."" for the other units for example Bel Volt.

TG Comment:

The TG decided that SENS_LOG_RATIO_METER should be deleted. The others have been added to the caveat below table A-14.

"Comment Number: 54

Commentor: PN

Section: Appendix A Table A-15

Previous Review Comment Number: 61

Decision: Accept

Comment:

Table A-15: Data Type Defines. Related previous comment 61. The table features defines for ""DS_Ipv4"" and ""DS_Ipv6"" defined as ""Ipv4 Address"" and ""Ipv6 Address"". The defines should be all in upper case, the descriptions should be IPv4/IPv6. Also neither is ""defined"" in the document with a nominative reference.

TG Comment:

Typo's corrected and normative references added as requested.

Comment Number: 55

Commentor: PW

Section: Appendix A Table A-15

Previous Review Comment Number:

Decision: Accept

Comment:

Formatting issue/page overflow

TG Comment:

Table has been corrected to carry headings across page break.

Comment Number: 56

Commentor: PN

Section: Appendix A Table A-17

Previous Review Comment Number:

Decision:

Comment:

Table A-17: Response NACK Reason Code Defines. The table has ""The Ipv4 Address provided is invalid."" it should be capitalised as IPv4.

TG Comment:

Typo's corrected.

Comment Number: 57

Commentor: PN

Section: Appendix C Table C-1

Previous Review Comment Number: 17

Decision: Accept

Comment:

Previous comment 17. ""Slot Index Type"" still only appears in Appendix C not in the main document, please be more consistent. Although I will concede that ""DMX512 Slot Indexes"" is used in SLOT_DESCRIPTION and DEFAULT_SLOT_VALUE

TG Comment:

Corrected.

Comment Number: 58

Commentor: PN

Section: Appendix C Table C-2

Previous Review Comment Number: 102

Decision: Reject

Comment:

Table C-2: Slot Label ID Definitions Previous comment 102. ""The point is very valid but this should be fixed in E1.37-5."" if someone wants to expose these as DMX channels, then E1.37-5 won't be relevant, although I appreciate most of them are probably more relevant as RDM PIDs, but adding SD_FREQUENCY (independent of strobe) and SD_PALETTE_SELECTION seem reasonable requests to me

TG Comment:

The TG felt that there is already a manufacturer specific area that deals with these needs.