

BACnet Errata
Addendum *bj* to ANSI/ASHRAE STANDARD 135-2016
A Data Communication Protocol for Building Automation and Control Networks

July 6, 2020

This document lists all known errata to Addendum *bj* to ANSI/ASHRAE 135-2016 as of the above date. Each entry is cited first by clause, then page number, except where an erratum covers more than one clause. The addendum as published is 135_2016_bi_20180618.pdf.

Changes to fix the addendum are indicated through the use of *italics*, while deletions are indicated by strikethrough.

1) **Clause YY.2.7**, p.37. Figures YY-6 and YY-7 should not include Data Options parameters

...

BVLC Function	X'00'	BVLC-Result
Control Flags	X'09' <u>X'08'</u>	Control octet = B'00001001' <u>B'00001000'</u> . Originating Virtual Address is present. Destination Virtual Address and Destination Options are absent. Data Options are present
Message ID	X'B5EC'	The Message ID from the BVLC message to which this is a result.
Originating Virtual Address	X'927BF71A96A2'	6 octets unicast Originating Virtual Address
Data Options	X'01'	Header Marker = B'00000001': Last Data Option, not required to understand, and Header Data absent Secure Path Header Option (1)
Payload	X'01'	Result for BVLC Function = Encapsulated-NPDU
	X'01'	Result Code = NAK
	X'BF'	Error Header Marker of header that caused the error.
	X'0007'	Error Class = COMMUNICATION
	X'0111'	Error Code = Proprietary error code (= 273)
	X'556E6DC3B6676C69 6368657220436F646521'	Error Details = "Unmöglicher Code!"

Figure YY-6. Example BVLC-Result Message with 'Error Details'

...

BVLC Function	X'00'	BVLC-Result
Control Flags	X'09' <u>X'08'</u>	Control octet = B'00001001' <u>B'00001000'</u> . Originating Virtual Address is present. Destination Virtual Address and Destination Options are absent. Data Options are present
Message ID	X'B5EC'	The Message ID from the BVLC message to which this is a result.
Originating Virtual Address	X'927BF71A96A2'	6 octets unicast Originating Virtual Address
Data Options	X'01'	Header Marker = B'00000001': Last Data Option, not required to understand, and Header Data absent Secure Path Header Option (1)
Payload	X'01'	Result for BVLC Function = Encapsulated-NPDU
	X'01'	Result Code = NAK
	X'3F'	Error Header Marker of header that caused the error.
	X'0007'	Error Class = COMMUNICATION
	X'0117'	Error Code = Proprietary error code (= 279)

Figure YY-7. Example BVLC-Result Message without 'Error Details'

2) **Clause YY.6.2, YY.6.2.2, YY.6.2.3**, p.47-50. Rename “connection wait timeout”, “connection wait timer” and “disconnection wait timeout”

[Modify **YY.6.2**, p. 48]

...

While waiting for a Connect-Request, or for a response to a Connect-Request, a ~~connection wait timer~~ connect wait timer shall be applied using the ~~connection wait timeout~~ connect wait timeout. On expiration of the ~~connection wait timer~~ connect wait timer, the peer shall close the WebSocket connection if any and enter the IDLE state.

The ~~connection wait timeout~~ connect wait timeout shall be configurable. The BACnet/SC node shall support a minimum range of 5..300 seconds. The recommended default value is 10 seconds.

[Modify **YY.6.2.2**, p. 49]

...

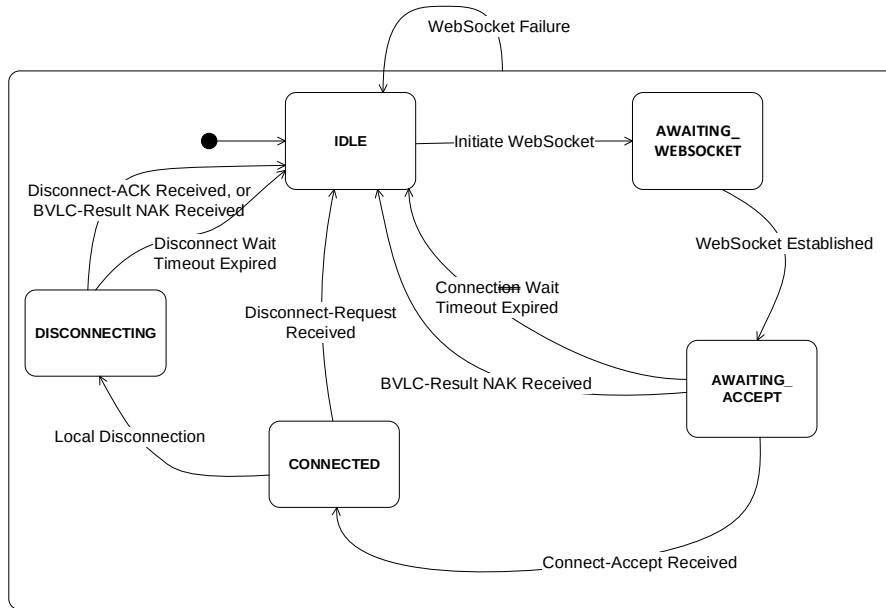


Figure YY-11. BACnet/SC Connection Initiating Peer Connection State Machine

...

In state **AWAITING_WEBSOCKET**

WebSocket established

On establishment of the WebSocket connection, send a Connect-Request, start the ~~connection wait timer~~connect wait timer, and enter the AWAITING_ACCEPT state.

In state **AWAITING_ACCEPT**

...

~~Connection wait timeout~~Connect wait timeout expired

On expiration of the ~~connection wait timer~~connect wait timer, close the WebSocket connection and enter the IDLE state.

...

[Modify YY.6.2.3, p. 49]

...

In state **IDLE**

Accepting a WebSocket

On accepting a WebSocket connection, start the ~~connection wait timer~~connect wait timer and enter the AWAITING_REQUEST state.

In state **AWAITING_REQUEST**

...

~~Connection wait timeout~~Connect wait timeout expired

On expiration of the ~~connection wait timeout~~connect wait timeout, close the WebSocket connection and enter the IDLE state.

...

In state **DISCONNECTING**

...

~~Disconnect Wait Timeout~~ Disconnect Wait Timeout expired

On expiration of a disconnect wait timeout, close the WebSocket connection and enter the IDLE state.

3) **Clause YY.5.3.1**, p 44. Inappropriate phrasing.

YY.5.3.1 Hub Connections

...

All BVLC message types related to the hub connection shall be initiated and sent to connection peers ~~as of Clause Error! Reference source not found.~~ ~~See Clause Error! Reference source not found.~~ ~~For indication of the~~ The VMAC address provided by the hub in the Connect-Accept message shall be the VMAC address of the BACnet/SC node of the network port in which the hub function resides ~~shall be used~~. Note that the VMAC is provided to the hub connector to allow a client to know the device which is hosting the hub function. It is otherwise unused by this protocol.

The ~~hub function's URI on which it accepts~~ ~~URI of where the BACnet/SC hub function accepts BACnet/SC connections as~~ hub connections is a WebSocket URI identified by the "wss" scheme.

...

4) **Clause YY.5.4**, p.45. Confusing language.

YY.5.4 Hub Connector for the BACnet/SC Hub Function

The hub connector of every BACnet/SC node shall support connecting to ~~the BACnet/SC hub function as the primary hub function or the failover hub function~~ a primary hub function and to a failover hub function. See Clause **Error! Reference source not found.**

...

5) **Clause YY.6**, p. 47. Missing “to” in the second paragraph.

YY.6 BACnet/SC Connections

...

The connection peer initiating the WebSocket connection is referred to as the initiating peer. The connection peer accepting the WebSocket connection is referred to as the accepting peer.

...

6) **Clause YY.6.2**, p. 47. The term “the peer” in the second paragraph is confusing.

YY.6.2 BACnet/SC Connection Establishment and Termination

...

While waiting for a Connect-Request, or for a response to a Connect-Request, a connection wait timer shall be applied using the connection wait timeout. On expiration of the connection wait timer, the initiating or accepting peer shall close the WebSocket connection ~~if any~~ and enter the IDLE state.

...