ADDENDA

ANSI/ASHRAE Addendum n to
ANSI/ASHRAE Standard 135.1-2009

Method of Test for Conformance to BACnet®

Approved by the ASHRAE Standards Committee on January 21, 2012; by the ASHRAE Board of Directors on January 25, 2012; and by the American National Standards Institute on January 26, 2012.

This addendum was approved by a Standing Standard Project Committee (SSPC) for which the Standards Committee has established a documented program for regular publication of addenda or revisions, including procedures for timely, documented, consensus action on requests for change to any part of the standard. The change submittal form, instructions, and deadlines may be obtained in electronic form from the ASHRAE Web site (www.ashrae.org) or in paper form from the Manager of Standards.

The latest edition of an ASHRAE Standard may be purchased on the ASHRAE Web site (www.ashrae.org) or from ASHRAE Customer Service, 1791 Tullie Circle, NE, Atlanta, GA 30329-2305. E-mail: orders@ashrae.org. Fax: 404-321-5478. Telephone: 404-636-8400 (worldwide), or toll free 1-800-527-4723 (for orders in US and Canada). For reprint permission, go to www.ashrae.org/permissions.

 ISSN 1041-2336

Approved American National Standard
ANSI
www.ansi.org

SPECIAL NOTE

This American National Standard (ANS) is a national voluntary consensus standard developed under the auspices of ASHRAE. *Consensus* is defined by the American National Standards Institute (ANSI), of which ASHRAE is a member and which has approved this standard as an ANS, as "substantial agreement reached by directly and materially affected interest categories. This signifies the concurrence of more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that an effort be made toward their resolution." Compliance with this standard is voluntary until and unless a legal jurisdiction makes compliance mandatory through legislation.

ASHRAE obtains consensus through participation of its national and international members, associated societies, and public review.

ASHRAE Standards are prepared by a Project Committee appointed specifically for the purpose of writing the Standard. The Project Committee Chair and Vice-Chair must be members of ASHRAE; while other committee members may or may not be ASHRAE members, all must be technically qualified in the subject area of the Standard. Every effort is made to balance the concerned interests on all Project Committees.

The Manager of Standards of ASHRAE should be contacted for:

a. interpretation of the contents of this Standard,
b. participation in the next review of the Standard,
c. offering constructive criticism for improving the Standard, or
d. permission to reprint portions of the Standard.

DISCLAIMER

ASHRAE uses its best efforts to promulgate Standards and Guidelines for the benefit of the public in light of available information and accepted industry practices. However, ASHRAE does not guarantee, certify, or assure the safety or performance of any products, components, or systems tested, installed, or operated in accordance with ASHRAE's Standards or Guidelines or that any tests conducted under its Standards or Guidelines will be nonhazardous or free from risk.

ASHRAE INDUSTRIAL ADVERTISING POLICY ON STANDARDS

ASHRAE Standards and Guidelines are established to assist industry and the public by offering a uniform method of testing for rating purposes, by suggesting safe practices in designing and installing equipment, by providing proper definitions of this equipment, and by providing other information that may serve to guide the industry. The creation of ASHRAE Standards and Guidelines is determined by the need for them, and conformance to them is completely voluntary.

In referring to this Standard or Guideline and in marking of equipment and in advertising, no claim shall be made, either stated or implied, that the product has been approved by ASHRAE.

[This foreword and the "rationales" on the following pages are not part of this standard. They are merely informative and do not contain requirements necessary for conformance to the standard.]

FOREWORD

The purpose of this addendum is to present a proposed change for publication. These modifications are the result of change proposals made pursuant to the ASHRAE continuous maintenance procedures and of deliberations within Standing Standard Project Committee 135. The proposed changes are summarized below.

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135.1-2009 and Addenda is indicated through the use of *italics*, while deletions are indicated by ~~strikethrough~~. Where entirely new subclauses are proposed to be added, plain type is used throughout. Only this new and deleted text is open to comment at this time. All other material in this addendum is provided for context only and is not open for public review comment except as it relates to the proposed changes.

135.1-2009n-1. Restrict The "Non-Documented" Test To Standard Object Types.

Rationale

Restrict the test to standard object types.

[Change **Clause 7.1.X** in **Addendum *i* to 135.1-2009**, p. 58]

7.1.X Non-documented Property Test

Purpose: To verify that all properties contained in every *standard* object are documented in the EPICS.

Test Concept: For each *standard* object in the EPICS database, attempt to read each standard property that the EPICS does not document as being part of the object.

Test Steps:

1. REPEAT X = (a tester selected set of *standard* objects) DO {
   REPEAT Y = (0 through 511) DO {
   IF (the property Y is not in the EPICS for object X) THEN
   TRANSMIT ReadProperty-Request,
   'Object Identifier' = X,
   'Property Identifier' = Y
   RECEIVE BACnet-Error-PDU,
   Error Class =PROPERTY,
   Error Code =UNKNOWN_PROPERTY
   }
   }

Notes to Tester: The objects selected by the tester should include one instance of each supported *standard* object type. Where some instances of an object type differ in the set of supported properties, the allowable value ranges for a property, or the writability of a property, then one instance of each variant of that object type should be selected.

135.1-2009n-2. Add Router Binding Test.

Rationale

This test is added to allow for routing binding via the broadcasting on the local network a confirmed service destined for a device on a different network.

This also clarifies that the RECEIVE statement should take this allowance into account.

[Change **Clause 6.2.7**, p. 28]

6.2.7 RECEIVE Statement

The RECEIVE procedure is used to define a message from the IUT.

<receive statement> ::= RECEIVE (<packet desc> | '(' <packet desc> ')' ['|' '(' <packet desc> ')'] ...)

The <pdu specification> parameter is the same as used in the TRANSMIT statement. If unspecified, the SOURCE defaults to IUT and DESTINATION defaults to TD. *Note: When DESTINATION refers to a device on a remote network, DA is allowed to be a local MAC broadcast.*

...

[Add new **Clause 10.Y.X** under "**10.Y Route Binding Tests"** in **Addendum *g* to 135.1-2009,** p. 28]

10.Y.X Reuse Router Binding Found By Broadcast

Purpose: This test verifies that a device which uses a local broadcast of a confirmed-request PDU (and thus in which the data-expecting-reply (DER) of the NPDU control octet is set) to route to a device on a remote network uses the address information from the response for subsequent requests to that device.

Test Concept: Configure the IUT so that it will use a local network broadcast of a confirmed-request PDU (i.e., the data-expecting-reply of the NPCI octet is set) in order to locate a router on the local network towards the specified remote device. TD impersonates a local router, responding with a PDU that appears to have been routed from RD. The IUT communicates with RD again but does not use a local network broadcast because it does not need to, having retained the address binding to TD which is impersonating the local router.

Configuration Requirements: The IUT shall be configured to request information from a device on a remote network, RD, using a BACnetAddress (network number and MAC address), and to use a MAC broadcast of a confirmed request PDU in order to locate a router to that network, on the local network, in order to communicate to that device. If the IUT cannot be configured in this fashion, then this test shall be skipped.

BACnet Reference Clause: 6.5.3.

Test Steps:

1. MAKE (IUT initiate a request to RD)
2. RECEIVE
   SA = IUT,
   DA = MAC BROADCAST,
   DNET = (network number of RD),
   DADR = (MAC address of RD),
   APDU Type = Confirmed-Request
3. TRANSMIT
   SA = TD,

DA = IUT,
SNET = (network number of RD),
SADR = (MAC address of RD),
APDU Type = Simple-ACK | Complex-ACK | Error-PDU | Reject-PDU

4. MAKE (IUT initiate a request to RD)
5. RECEIVE
   SA = IUT,
   DA = TD,
   DNET = (network address of RD),
   DADR = (MAC address of RD),
   APDU Type = Confirmed-Request
6. TRANSMIT BACnet-Simple-ACK-PDU

135.1-2009n-3. Update Priority_For_Writing Tests.

Rationale

These changes allow the IUT to leave out the Priority parameter when the target property is known to be not commandable.

[Change Clause **7.3.2.23.Y.3** in **Addendum *g* to 135.1-2009**, p. 64]

Note to Tester: The Priority parameter for the WriteProperty-Request may be left out if the Schedule is configured with a value of 16 in its Priority_For_Writing property *or if the target property is a standard property of a standard object for which commandability is not an option*. The test shall pass regardless of the order in which the IUT generates the WriteProperty-Requests in steps 2 and 5.

[Change Clause **7.3.2.23.Y.4** in **Addendum *g* to 135.1-2009**, p. 65]

Note to Tester: The Priority parameter for the WriteProperty-Request may be left out if the Schedule is configured with a value of 16 in its Priority_For_Writing property *or if the target property is a standard property of a standard object for which commandability is not an option*. The test shall pass or fail regardless of the order in which the IUT generates the WriteProperty-Requests in steps 2 and 5.

[Change Clause **7.3.2.23.8**, p. 93]

7.3.2.23.8 List_Of_Object_Property_Reference External Test

Dependencies: ReadProperty Service Execution Tests, 9.18; TimeSynchronization Service Execution Tests, 9.30.

BACnet Reference Clause: 12.24.10.

Purpose: To verify that the Schedule object writes to object properties contained in a device other than the IUT.

Test Concept: The Schedule object is configured to write to a property of another object in the same device and a property of an object in the TD. The IUT's clock is then set to a time between a pair of scheduled write operations, and verification of the first write operation's data value is performed. The time is advanced to the second time, the Schedule object's Present_Value is checked, and verifications of the write operation are performed. If the IUT does not support writes to object properties contained in a device other than the IUT, then this test shall not be performed.

Configuration Requirements: The TD is configured to indicate that it supports the WriteProperty-Request service but not WritePropertyMultiple-Request. The IUT is configured with a Schedule object containing a List_Of_Object_Property_References property that references a property of an object contained in the TD. The Schedule object is configured with either a Weekly_Schedule or an active Exception_Schedule, during a period where Effective_Period is active, with at least two consecutive entries with distinguishable values in the List of BACnetTimeValues, and with no Exception_Schedules at a higher priority. D1 represents the date and time of the first of these two BACnetTimeValues, with corresponding value V1, while D2 and V2 (a value distinguishable from V1) represent the second BACnetTimeValue. A time Dt is defined to occur between D1 and D2.

Test Steps:

1. (TRANSMIT TimeSynchronization-Request, 'Time' = D_t) | MAKE (the local date and time = D_t)
2. WAIT **Schedule Evaluation Fail Time**
3. VERIFY Present_Value = V_1
4. (TRANSMIT TimeSynchronization-Request, 'Time' = D_2) | MAKE (the local date and time = D_2)
5. BEFORE **Schedule Evaluation Fail Time**
   RECEIVE WriteProperty-Request,
   'Object Identifier' = (the referenced object in the TD),

'Property Identifier' = (the referenced property in the TD),
'Property Value' = V_2,
'Priority' = (the value of the Schedule object's Priority_For_Writing property)

6. WAIT **Schedule Evaluation Fail Time**
7. VERIFY Present_Value = V_2

Note to Tester: The Priority parameter for the WriteProperty-Request may be left out if the Schedule is configured with a value of 16 in its Priority_For_Writing property or if the target property is a standard property of a standard object for which commandability is not an option.

135.1-2009n-4. Make Trend Log Tests Generic.

Rationale

Make the Trend Log tests generic so they may be applied to other logging objects.

[Change **Clause 7.3.2.24.1** in **Addendum g to 135.1-2009**, p. 38]
[Note to reviewer: because the published Addendum *g* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *g*. Instead, text to be removed from and inserted into Addendum *g* is shown separately]

[remove existing text]
7.3.2.24.1 Log_Enable Test
[replace with new text]
7.3.2.24.1 Enable Test
[end new text]

Dependencies: ReadProperty Service Execution Tests, 9.18; WriteProperty Service Execution Tests, 9.22.

BACnet Reference Clause: 12.25.5.

[remove existing text]
Purpose: To verify that the Log_Enable property enables and disables the logging of data by the Trend Log object.
[replace with new text]
Purpose: To verify that the ~~Log_~~Enable property enables and disables the logging of data by the ~~Trend Log~~ *logging* object.
[end new text]

[remove existing text]
Test Concept: The Trend Log is configured to acquire data by each means (polling and COV subscription) available to the implementation. Log_Enable is enabled and the collection of one or more records in the Log_Buffer is confirmed. Log_Enable is then disabled and non-collection of records is confirmed.
[replace with new text]
Test Concept: The ~~Trend Log~~ *logging object* is configured to acquire data by each means ~~(polling and COV subscription)~~ available to the implementation. ~~Log_~~Enable is enabled and the collection of one or more records in the Log_Buffer is confirmed. ~~Log_~~Enable is then disabled and non-collection of records is confirmed.
[end new text]

[remove existing text]
The COV increment used is either the Client_COV_Increment property of the Trend Log or the COV_Increment property of the monitored object, depending on the configuration of the Trend Log object being tested.
[end remove text]

Configuration Requirements: Start_Time, if present, shall be configured with a date and time preceding the beginning of the test. Stop_Time, if ~~present shall~~ *present, shall* be configured with a time that will occur after the completion of the test. Stop_When_Full, if configurable, shall be set to FALSE.

Test Steps:

~~1. WRITE Log_Enable = FALSE~~
~~2. WRITE Record_Count = 0~~
~~3. WAIT **Internal Processing Fail Time**~~
~~4 TRANSMIT ReadProperty Request,~~
~~'Object Identifier' = (the object being tested),~~

~~'Property Identifier' = Total_Record_Count~~
~~5 RECEIVE ReadProperty-ACK,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~'Property Value' = (any valid value, X)~~
~~6. WRITE Log_Enable = TRUE~~
~~7. WAIT **Internal Processing Fail Time**~~
~~8. IF (COV subscription in use) THEN~~
~~MAKE (monitored value change more than Client_COV_Increment)~~
~~ELSE~~
~~WAIT (Log_Interval)~~
~~9. WAIT (**Notification Fail Time + Internal Processing Fail Time**)~~
~~10. VERIFY Total_Record_Count > (value X returned in step 5)~~
~~11. WRITE Log_Enable = FALSE~~
~~12. WAIT **Internal Processing Fail Time**~~
~~13. TRANSMIT ReadProperty-Request,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~14. RECEIVE ReadProperty-ACK,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~'Property Value' = (any valid value, X)~~
~~15. IF (COV subscription in use) THEN~~
~~MAKE (monitored value change more than Client_COV_Increment)~~
~~ELSE~~
~~WAIT (Log_Interval)~~
~~16. WAIT (**Notification Fail Time + Internal Processing Fail Time**)~~
~~17. VERIFY Total_Record_Count = (value X returned in step 14)~~

1. READ I = Log_Interval
[remove existing text]
2. WRITE Log_Enable = FALSE
[replace with new text]
2. WRITE Enable = FALSE
[end new text]
3. WRITE Record_Count = 0
*4. WAIT **Internal Processing Fail Time***
[remove existing text]
5. WRITE Log_Enable = TRUE
[replace with new text]
5. WRITE Enable = TRUE
[end new text]
6. READ X = Total_Record_Count
[remove existing text]
7. IF (I = 0) THEN
MAKE (monitored value change more than the COV increment)
ELSE
WAIT (I)
[replace with new text]
7. MAKE (IUT collect another record)
[end new text]
*8. WAIT (**Notification Fail Time** + **Internal Processing Fail Time**)*
9. VERIFY Total_Record_Count > X
[remove existing text]
10. WRITE Log_Enable = FALSE
[replace with new text]

10. WRITE Enable = FALSE
[end new text]
11. READ Y = Total_Record_Count
[remove existing text]
12. IF (I = 0) THEN
MAKE (monitored value change more than the COV increment)
ELSE
WAIT (I)
[replace with new text]
12. MAKE (IUT collect another record)
[end new text]
*13. WAIT (**Notification Fail Time** + **Internal Processing Fail Time**)*
14. VERIFY Total_Record_Count = Y

[add new text]
Note to Tester: For each MAKE (IUT collect another record), perform the following actions:

IF (Event Log Object) THEN
MAKE (Event Log Object collect another record)
ELSE
IF (COV subscription in use) THEN
MAKE (monitored value change sufficient to generate another record)
ELSE IF (interval or period logging is in use) THEN
WAIT (Log_Interval)
ELSE
MAKE (Trend Log or Trend Log Multiple Object collect another record)
[end new text]

[Change **Clause 7.3.2.24.3** in **Addendum *g* to 135.1-2009**, p. 39]
[Note to reviewer: because the published Addendum *g* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *g*. Instead, text to be removed from and inserted into Addendum *g* is shown separately]

7.3.2.24.3 Stop_Time Test

Dependencies: ReadProperty Service Execution Tests, 9.18; WriteProperty Service Execution Tests, 9.22.

BACnet Reference Clause: 12.25.7.

Purpose: To verify that logging is disabled at the time specified by Stop_Time.

[remove existing text]
Test Concept: The Trend Log is configured to acquire data by each means (polling and COV subscription) available to the implementation. The test is begun at some time prior to the time specified in Start_Time and collection of records is confirmed. Non-collection of records after the time specified by Stop_Time is then confirmed.
[replace with new text]
Test Concept: The ~~Trend Log~~ *logging object* is configured to acquire data by each means ~~(polling and COV subscription)~~ available to the implementation. The test is begun at some time prior to the time specified in Start_Time and collection of records is confirmed. Non-collection of records after the time specified by Stop_Time is then confirmed.
[end new text]

Configuration Requirements: Stop_Time shall be configured with a date and time such that steps 1 through 9 will be concluded before that time. Start_Time, if present, shall be configured with date and time preceding the initiation of the test. Stop_When_Full, if configurable, shall be set to FALSE.

Test Steps:

[remove existing text]
1. WRITE Log_Enable = FALSE
[replace with new text]
1. WRITE ~~Log_~~Enable = FALSE
[end new text]
2. WAIT **Internal Processing Fail Time**
3. WRITE Record_Count = 0
[remove existing text]
4. WRITE Log_Enable = TRUE
[replace with new text]
4. WRITE Enable = TRUE
[end new text]
[the following test steps have been renumbered appropriately]
~~4. TRANSMIT ReadProperty-Request,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~5. RECEIVE ReadProperty-ACK,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~'Property Value' = (any valid value, X)~~
~~6. WRITE Log_Enable = TRUE~~
[remove existing text]
5. READ X = Record_Count
[replace with new text]
5. READ X = Total_Record_Count
[end new text]
6. WAIT **Internal Processing Fail Time**
[remove existing text]
7. IF (COV subscription in use) THEN
MAKE (monitored value change more than Client_COV_Increment)
ELSE
WAIT (Log_Interval)
[replace with new text]
~~7. IF (COV subscription in use) THEN~~
~~MAKE (monitored value change more than Client_COV_Increment)~~
~~ELSE~~
~~WAIT (Log_Interval)~~
7. MAKE (IUT collect another record)
[end new text]
8. WAIT (**Notification Fail Time** + **Internal Processing Fail Time**)
9. VERIFY Total_Record_Count > ~~(value X returned in step 5)~~*X*
10. WHILE (IUT clock is earlier than Stop_Time) DO {}
11. WAIT (**Notification Fail Time** + **Internal Processing Fail Time**)
~~13. TRANSMIT ReadProperty-Request,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~14. RECEIVE ReadProperty-ACK,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~'Property Value' = (any valid value, X)~~
12. READ X = Total_Record_Count
[remove existing text]
13. IF (COV subscription in use) THEN
MAKE (monitored value change more than Client_COV_Increment)

ELSE
WAIT (Log_Interval)

[replace with new text]

~~13. IF (COV subscription in use) THEN~~
~~MAKE (monitored value change more than Client_COV_Increment)~~
~~ELSE~~
~~WAIT (Log_Interval)~~

13. MAKE (IUT collect another record)

[end new text]

14. WAIT (**Notification Fail Time** + **Internal Processing Fail Time**)
15. VERIFY Total_Record_Count = ~~(value X returned in step 14)~~*X*

[add new text]

Note to Tester: For each MAKE (IUT collect another record), perform the following actions:

IF (Event Log Object) THEN
MAKE (Event Log Object collect another record)
ELSE
IF (COV subscription in use) THEN
MAKE (monitored value change sufficient to generate another record)
ELSE IF (interval or period logging is in use) THEN
WAIT (Log_Interval)
ELSE
MAKE (Trend Log or Trend Log Multiple Object collect another record)

[end new text]

[Change **Clause 7.3.2.24.9** in **Addendum *g* to 135.1-2009**, p. 43]

[Note to reviewer: because the published Addendum *g* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to the Addendum *g*. Instead, text to be removed from and inserted into Addendum *g* is shown separately]

7.3.2.24.9 Total_Record_Count Test

Dependencies: ReadProperty Service Execution Tests, 9.18; WriteProperty Service Execution Tests, 9.22.

BACnet Reference Clause: 12.25.16.

Purpose: To verify that the Total_Record_Count property increments for each record added to the Log_Buffer, even after Buffer_Size records have been added. (Note: it is not reasonable to test for the requirement of BACnet Clause 12.23.16 that the value wrap from 2^{32}-1 to 0; even if a record was collected every 100^{th} of a second it could take more than 497 days to complete the test.)

[remove existing text]

Test Concept: The Trend Log is configured to acquire data by whatever means. Record_Count is set to zero and Total_Record_Count is read. Collection of data proceeds until Record_Count changes, collection is halted and Total_Record_Count is checked that it has incremented by Record_Count. If, for whatever reason, the IUT cannot be configured such that the TD is able to halt collection before Buffer_Size records are collected this test shall not be performed.

[replace with new text]

Test Concept: The ~~Trend Log~~ *logging object* is configured to acquire data by whatever means. Record_Count is set to zero and Total_Record_Count is read. Collection of data proceeds until Record_Count changes, collection is halted and Total_Record_Count is checked that it has incremented by Record_Count. If, for whatever reason, the IUT cannot be configured such that the TD is able to halt collection before Buffer_Size records are collected this test shall not be performed.

[end new text]

[remove existing text]

Configuration Requirements: Start_Time, if present, shall be configured with a date and time preceding the beginning of the test. Stop_Time, if present, shall be configured with the latest possible date and time in order that it occur after the end of the test. Log_Enable shall be set to FALSE.

[replace with new text]

Configuration Requirements: Start_Time, if present, shall be configured with a date and time preceding the beginning of the test. Stop_Time, if present, shall be configured with the latest possible date and time in order that it occur after the end of the test. ~~Log_~~Enable shall be set to FALSE.

[end new text]

Test Steps:

1. WRITE Record_Count = 0
2. WAIT **Internal Processing Fail Time**

~~3. TRANSMIT ReadProperty-Request,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count~~
~~4. RECEIVE ReadProperty-ACK,~~
~~'Object Identifier' = (the object being tested),~~
~~'Property Identifier' = Total_Record_Count,~~
~~'Property Value' = (any valid value, X)~~

3. READ X = Total_Record_Count
4. READ Y = Record_Count

[remove existing text]

5. WRITE Log_Enable = TRUE

[replace with new text]

5. WRITE ~~Log_~~Enable = TRUE

[end new text]

6. WHILE (Record_Count = ~~0~~*Y + 1*) DO { }

[remove existing text]

7. WRITE Log_Enable = FALSE

[replace with new text]

7. WRITE ~~Log_~~Enable = FALSE

[end new text]

8. WAIT **Internal Processing Fail Time**
9. ~~IF (Record_Count = Buffer_Size) THEN~~
~~ERROR "Buffer full; cannot verify Total_Record_Count value."~~
~~ELSE {~~
~~IF (Total_Record_Count != Record_Count + (value X returned in step 4)) THEN~~
IF (Total_Record_Count - X != Record_Count - Y) THEN
ERROR "Total_Record_Count has incorrect value."
~~}~~

135.1-2009n-5. Bring Attention To Change In Length Of BACnetLogStatus.

Rationale

A note is added to the test bringing the tester's attention to the length change in the BACnetLogStatus production and to make the test generic for all logging objects.

[Change **Clause 7.3.2.24.X1** in **Addendum *i* to 135.1-2009**, p. 20]
[Note to reviewer: because the published Addendum *i* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *i*. Instead, text to be removed from and inserted into Addendum *i* is shown separately]

7.3.2.24.X1 Log-Status Test

Dependencies: ReadRange Service Execution Tests, 9.21; WriteProperty Service Execution Tests, 9.22.

BACnet Reference Clause: 12.25.14.

Purpose: To verify proper logging of log-disabled and buffer-purged events.

Test Concept: The buffer is cleared. Then the Enable property is changed and it is verified that the Record_Count property is changed and it is verified that the status entry is made correctly in the Log_Buffer. The Record_Count is also set to zero while the Enable property is FALSE and it is verified that the buffer-purged event is recorded into the Log_Buffer.

[remove existing text]
Test Configuration: The Trend Log, O1, is configured to acquire data by whatever means available. Configure the logging such that the entire test may be run without the trend buffer overflowing.
[replace with new text]
Test Configuration: The ~~Trend Log~~*logging object*, O1, is configured to acquire data by whatever means available. Configure the logging such that the entire test may be run without the trend buffer overflowing.
[end new text]

Test Steps:

1. WRITE Enable = FALSE
2. WRITE Record_Count = 0
3. VERIFY Record_Count = 1
4. TRANSMIT ReadRange
   - 'Object Identifier' = O1,
   - 'Property Identifier' = Log_Buffer,
   - 'Reference Index' = 1,
   - 'Count' = 1
5. RECEIVE ReadRange-Ack
   - 'Object Identifier' = O1,
   - 'Property Identifier' = Log_Buffer,
   - 'Result Flags' = (True, True, False),
   - 'Item Count' = 1
   - 'Item Data' = ((a buffer purged record))
6. WRITE Enable = TRUE
7. WRITE Enable = FALSE
8. TRANSMIT ReadRange
   - 'Object Identifier' = O1,
   - 'Property Identifier' = Log_Buffer,
   - 'Reference Index' = 1,

‘Count’ = 2

9. RECEIVE ReadRangeAck
   - ‘Object Identifier’ = O1,
   - ‘Property Identifier’ = Log_Buffer,
   - ‘Result Flags’ = (True, False, False),
   - ‘Item Count’ = 2
   - ‘Item Data’ = ((a buffer purged record), (a log-enable record))
10. TRANSMIT ReadRange
   - ‘Object Identifier’ = O1,
   - ‘Property Identifier’ = Log_Buffer,
   - ‘Reference Time’ = (2154-12-31, 23:59:59.99),
   - ‘Count’ = -1
11. RECEIVE ReadRangeAck
   - ‘Object Identifier’ = O1,
   - ‘Property Identifier’ = Log_Buffer,
   - ‘Result Flags’ = (False, True, False),
   - ‘Item Count’ = 1
   - ‘Item Data’ = ((a log-disable record))

[add new text]

Notes to Tester: When the IUT's Protocol_Revision < 7, the length of BACnetLogStatus shall be 2; otherwise, it shall be 3.

[end new text]

135.1-2009n-6. Clarify That "Ignore Remote Packets" Test is Not for Use with Intervening Router.

Rationale

Clarify that the new test 10.X.1 cannot be applied in a test setup where there is a router between the TD and the IUT.

[Change **Clause 10.X.1** in **Addendum *g* to 135.1-2009**, p. 22]
[Note to reviewer: because the published Addendum *g* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *g*. Instead, text to be removed from and inserted into Addendum *g* is shown separately]

10.X.1 Ignore Remote packets

BACnet Reference Clause: 6.5.2.1, 6.5.4

Purpose: This test case verifies that the non-router IUT will quietly accept and discard packets destined for remote networks.

Test Concept: The TD transmits both broadcast and directed requests to the IUT with DNET (not equal to x'FFFF') and DADR in the Network Layer header. The IUT is required to silently drop the requests because it is not a router.

Test Steps:
1. TRANSMIT
   DA = BROADCAST,
   SA = TD,
   DNET = DNET3,
   DADR= BROADCAST,
   Hop Count = 255,
   BACnet-Unconfirmed-Request-PDU,
   'Service Choice' = who-Is
2. WAIT **Internal Processing Fail Time**
3. CHECK (that the IUT did not send an I-Am)
4. TRANSMIT
   DA = IUT,
   SA = TD,
   DNET = DNET3,
   DADR = IUT,
   Hop Count = 255,
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = O2 (any BACnet standard object in IUT),
   'Property Identifier' = P2 (any required property of the specified object)
5. WAIT **Internal Processing Fail Time**
6. CHECK (that the IUT did not send a response to the ReadProperty)

[add new text]
Notes to Tester: Ensure that the packets transmitted in Step 1 and Step 4 will actually reach the IUT by transmitting them on the local network of the IUT. It is impossible to perform this test through a router.
[end new text]

135.1-2009n-7. Modify B/IP Test For NAT Operation.

Rationale

The BBMD tests are modified to take allow for testing IUTs operating in NAT mode.

[Change **Clause 14.2** in **Addendum *e* to 135.1-2009**, p. 3]
[Note to reviewer: because the published Addendum *e* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *e*. Instead, text to be removed from and inserted into Addendum *e* is shown separately]

14.2 ~~Non-~~**BBMD B/IP** ~~device~~ **Device with a Server Application**

This group of tests verifies that a *BBMD* B/IP device ~~that is not a BBMD~~ *with a server application* will correctly process NPDU's conveyed in the NPDU portion of Forwarded-NPDU, Original-Broadcast-NPDU and Original-Unicast-NPDU messages.

[remove existing text]
~~Configuration Requirements: Before this group of tests is performed, the IUT shall be configured so that the BBMD option is off. The IUT shall be made to go through its startup procedure. A server application shall be running.~~
Configuration Requirements: A server application shall be running in the IUT. For one-hop distribution tests the Internet Routers in Figure 14-1 must be configured to forward directed broadcasts.
[replace with new text]
~~Configuration Requirements: Before this group of tests is performed, the IUT shall be configured so that the BBMD option is off. The IUT shall be made to go through its startup procedure. A server application shall be running.~~
Configuration Requirements: A server application shall be running in the IUT. For one-hop distribution tests, the Internet Routers in Figure 14-1 must be configured to forward directed broadcasts. For two-hop distribution tests utilizing Internet Routers providing Network Address Translation (NAT), the IUT must be configured for NAT operation. In addition, the Internet Routers must be configured to port-forward the UDP port in use by the B/IP network.
[end new text]

Notes to Tester: Figure 14-1 shows the logical network configuration for tests 14.2 – 14.7. The complete network is not required for the tests, so long as the IUT can receive packets formed as though they arrived from the specified device. The role of the TD when executing the TRANSMIT statement in each test is specified. The TD must also monitor the IUT's subnet throughout all tests and, RECEIVE shall mean from the IUT's subnet. To accomplish this, the TD may be multi-homed or another TD can be used to monitor the IUT's subnet.

…

[Change **Clause 14.2.1.2** in **Addendum *e* to 135.1-2009**, p. 5]
[Note to reviewer: because the published Addendum *e* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *e*. Instead, text to be removed from and inserted into Addendum *e* is shown separately]

14.2.1.2 Execute Forwarded-NPDU (Two-hop Distribution)

Configuration Requirements: The IUT shall be configured with a BDT that contains:

B/IP Address	*Broadcast Distribution Mask*
IUT	*255.255.255.255*
BBMD1	*255.255.255.255*

[add new text]

When the IUT is configured for NAT, the Originating-Device in Forwarded-NPDUs that originate at the IUT, OD, is equal to the Global IP Address and Port of the IUT's Internet Router. When the IUT is not configured for NAT operation, OD is equal to the IUT.

[end new text]

Test Steps:

1. *TRANSMIT*
   DA = IUT,
   SA = BBMD1,
   Forwarded-NPDU,
   Originating-Device = BBMD1,
   NPDU = Who-Is
2. *RECEIVE*
   DA = Local IP Broadcast,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = BBMD1,
   NPDU = Who-Is
3. *RECEIVE*
   DA = Local IP Broadcast,
   SA = IUT,
   Original-Broadcast-NPDU,
   NPDU = I-Am
4. *RECEIVE*
   DA = BBMD1,
   SA = IUT,
   Forwarded-NPDU,

[remove existing text]

Originating-Device = IUT,

[replace with new text]

Originating-Device = OD,

[end new text]

NPDU = I-Am

Notes to Tester: The order of the messages transmitted by the IUT is not significant.

[Change **Clause 14.2.2.2** in **Addendum *e* to 135.1-2009**, p. 7]

[Note to reviewer: because the published Addendum *e* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *e*. Instead, text to be removed from and inserted into Addendum *e* is shown separately]

14.2.2.2 Execute Original-Broadcast-NPDU (Two-hop Distribution)

Configuration Requirements: The IUT shall be configured with a BDT that contains:

B/IP Address	*Broadcast Distribution Mask*
IUT	*255.255.255.255*
BBMD1	*255.255.255.255*

[add new text]

When the IUT is configured for NAT, the Originating-Device in Forwarded-NPDUs that originate at the IUT, OD, is equal to the Global IP Address and Port of the IUT's Internet Router. When the IUT is not configured for NAT operation, OD is equal to the IUT.

[end new text]

Test Steps:

1. *TRANSMIT*
   DA = Local IP Broadcast,
   SA = D1,
   Original-Broadcast-NPDU,
   NPDU = Who-Is
2. *RECEIVE*
   DA = BBMD1,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = D1,
   NPDU = Who-Is
3. *RECEIVE*
   DA = Local IP Broadcast,
   SA = IUT,
   Original-Broadcast-NPDU,
   NPDU = I-Am
4. *RECEIVE*
   DA=BBMD1,
   SA=IUT,
   Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

Notes to Tester: The order of the messages transmitted by the IUT is not significant.

[Change **Clause 14.7.1.2** in **Addendum *e* to 135.1-2009**, p. 23]
[Note to reviewer: because the published Addendum *e* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *e*. Instead, text to be removed from and inserted into Addendum *e* is shown separately]

14.7.1.2 Broadcast Message from Directly Connected IP Subnet (Two-hop Distribution)

Configuration Requirements: The BDT shall contain the following three entries:

B/IP Address	*Broadcast Distribution Mask*
IUT	*255.255.255.255*
BBMD1	*255.255.255.255*
BBMD2	*255.255.255.255*

[add new text]
When the IUT is configured for NAT, the Originating-Device in Forwarded-NPDUs that originate at the IUT, OD, is equal to the Global IP Address and Port of the IUT's Internet Router. When the IUT is not configured for NAT operation, OD is equal to the IUT.
[end new text]

Steps 2-5 are the distribution of the Who-Is request to the devices considered to be members of the BACnet network; steps 6-10 are the distribution of the I-Am response from the local application.

Test Steps:

1. *TRANSMIT*
   DA = Local IP Broadcast,
   SA = D1,
   Original-Broadcast-NPDU,
   NPDU = Who-Is
2. *RECEIVE*
   DA = BBMD1,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = D1,
   NPDU = Who-Is
3. *RECEIVE*
   DA = BBMD2,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = D1,
   NPDU = Who-Is
4. *RECEIVE*
   DA = FD1,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = D1,
   NPDU = Who-Is
5. *RECEIVE*
   DA = FD2,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = D1,
   NPDU = Who-Is
6. *RECEIVE*
   DA = Local IP Broadcast,
   SA = IUT,
   Original-Broadcast-NPDU,
   NPDU = I-Am
7. *RECEIVE*
   DA = BBMD1,
   SA = IUT,
   Forwarded-NPDU,

[remove existing text]

   Originating-Device = IUT,

[replace with new text]

   Originating-Device = OD,

[end new text]

   NPDU = I-Am
8. *RECEIVE*
   DA = BBMD2,
   SA = IUT,
   Forwarded-NPDU,

[remove existing text]

   Originating-Device = IUT,

[replace with new text]

   Originating-Device = OD,

[end new text]

NPDU = I-Am

9. *RECEIVE*
   DA = FD1,
   SA = IUT,
   Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

10. *RECEIVE*
    DA = FD2,
    SA = IUT,
    Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

Notes to Tester: The order of the messages transmitted by the IUT is not significant.

[Change **Clause 14.7.2.2** in **Addendum *e* to 135.1-2009**, p. 26]
[Note to reviewer: because the published Addendum *e* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to Addendum *e*. Instead, text to be removed from and inserted into Addendum *e* is shown separately]

14.7.2.2 Broadcast Message Forwarded by a Peer BBMD (Two-hop Distribution)

Configuration Requirements: The BDT shall be configured as in test 14.7.1.2

[add new text]
When the IUT is configured for NAT, the Originating-Device in Forwarded-NPDUs that originate at the IUT, OD, is equal to the Global IP Address and Port of the IUT's Internet Router. When the IUT is not configured for NAT operation, OD is equal to the IUT.
[end new text]

Steps 2-4 are the distribution of the Who-Is request to the devices considered to be members of the BACnet network; steps 5-9 are the distribution of the I-Am response from the local application.

Test Steps:

1. *TRANSMIT*
   DA = IUT,
   SA = BBMD1,
   Forwarded-NPDU,
   Originating-Device = D2,
   NPDU = Who-Is
2. *RECEIVE*
   DA = Local IP Broadcast,

SA = IUT,
Forwarded-NPDU,
Originating-Device = D2,
NPDU = Who-Is

3. *RECEIVE*
DA = FD1,
SA = IUT,
Forwarded-NPDU,
Originating-Device = D2,
NPDU = Who-Is
4. *RECEIVE*
DA = FD2,
SA = IUT,
Forwarded-NPDU,
Originating-Device = D2,
NPDU = Who-Is
5. *RECEIVE*
DA = Local IP Broadcast,
SA = IUT,
Original-Broadcast-NPDU,
NPDU = I-Am
6. *RECEIVE*
DA = BBMD1,
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

7. *RECEIVE*
DA = BBMD2,
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

8. *RECEIVE*
DA = FD1,
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

9. *RECEIVE*
DA = FD2,
SA = IUT,
Forwarded-NPDU,

[remove existing text]

Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

Notes to Tester: The order of the messages transmitted by the IUT is not significant.

[Change **Clause 14.7.3.2** in **Addendum *e* to 135.1-2009**, p. 30]
[Note to reviewer: because the published Addendum *e* text already has italics and strikeouts to indicate changes to standard 135.1-2009, that markup format cannot be used to show changes to the Addendum *e*. Instead, text to be removed from and inserted into Addendum *e* is shown separately]

14.7.3.2 Broadcast Message from a Foreign Device (Two-hop Distribution)

Configuration Requirements: The BDT and FDT shall be configured as in test 14.7.1.2.

[add new text]
When the IUT is configured for NAT, the Originating-Device in Forwarded-NPDUs that originate at the IUT, OD, is equal to the Global IP Address and Port of the IUT's Internet Router. When the IUT is not configured for NAT operation, OD is equal to the IUT.
[end new text]

Steps 2-5 are the distribution of the Who-Is request to the devices considered to be members of the BACnet network; steps 6-10 are the distribution of the I-Am response from the local application.

Test Steps:

1. *TRANSMIT*
   DA = IUT,
   SA = FD1,
   Distribute-Broadcast-To-Network,
   NPDU = Who-Is
2. *RECEIVE*
   DA = Local IP Broadcast,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = FD1,
   NPDU = Who-Is
3. *RECEIVE*
   DA = BBMD1,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = FD1,
   NPDU = Who-Is
4. *RECEIVE*
   DA = BBMD2,
   SA = IUT,
   Forwarded-NPDU,
   Originating-Device = FD1,
   NPDU = Who-Is
5. *RECEIVE*
   DA = FD2,

SA = IUT,
Forwarded-NPDU,
Originating-Device = FD1,
NPDU = Who-Is

6. *RECEIVE*
DA = Local IP Broadcast,
SA = IUT,
Original-Broadcast-NPDU,
NPDU = I-Am
7. *RECEIVE*
DA = BBMD1,
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

8. *RECEIVE DA = BBMD2,*
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

9. *RECEIVE*
DA = FD1,
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

10. *RECEIVE*
DA = FD2,
SA = IUT,
Forwarded-NPDU,

[remove existing text]
Originating-Device = IUT,
[replace with new text]
Originating-Device = OD,
[end new text]
NPDU = I-Am

Notes to Tester: The order of the messages transmitted by the IUT is not significant.

POLICY STATEMENT DEFINING ASHRAE'S CONCERN FOR THE ENVIRONMENTAL IMPACT OF ITS ACTIVITIES

ASHRAE is concerned with the impact of its members' activities on both the indoor and outdoor environment. ASHRAE's members will strive to minimize any possible deleterious effect on the indoor and outdoor environment of the systems and components in their responsibility while maximizing the beneficial effects these systems provide, consistent with accepted standards and the practical state of the art.

ASHRAE's short-range goal is to ensure that the systems and components within its scope do not impact the indoor and outdoor environment to a greater extent than specified by the standards and guidelines as established by itself and other responsible bodies.

As an ongoing goal, ASHRAE will, through its Standards Committee and extensive technical committee structure, continue to generate up-to-date standards and guidelines where appropriate and adopt, recommend, and promote those new and revised standards developed by other responsible organizations.

Through its *Handbook*, appropriate chapters will contain up-to-date standards and design considerations as the material is systematically revised.

ASHRAE will take the lead with respect to dissemination of environmental information of its primary interest and will seek out and disseminate information from other responsible organizations that is pertinent, as guides to updating standards and guidelines.

The effects of the design and selection of equipment and systems will be considered within the scope of the system's intended use and expected misuse. The disposal of hazardous materials, if any, will also be considered.

ASHRAE's primary concern for environmental impact will be at the site where equipment within ASHRAE's scope operates. However, energy source selection and the possible environmental impact due to the energy source and energy transportation will be considered where possible. Recommendations concerning energy source selection should be made by its members.