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

ASHRAE ADDENDA

Method of Test for Conformance to BACnet®

Approved by the ASHRAE Standards Committee on January 29, 2011; by the ASHRAE Board of Directors on February 2, 2011; and by the American National Standards Institute on February 3, 2011.

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

American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.
1791 Tullie Circle NE, Atlanta, GA 30329
www.ashrae.org

SPECIAL NOTE

This American National Standard (ANS) is a national voluntary consensus standard developed under the auspices of the American Society of Heating, Refrigerating and Air-Conditioning Engineers (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

Addendum 135.1*g* to ANSI/ASHRAE Standard 135.1-2009 contains a number of changes to the current standard. 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 changes are summarized below.

In the following document, language 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 added, plain type is used throughout.

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135.1-2009*g*-1 Correct Test Step Indention.

Rationale
The test step 8 is incorrectly numbered and incorrectly indented.

[Change **Clause 8.2.7**, p 113]

8.2.7 Change of Value Notification from Loop Object Present_Value Property

Purpose: To verify that the IUT can initiate ConfirmedCOVNotification service requests conveying a change of the Present_Value property of a loop object.

Test Concept: A subscription for COV notifications is established. The Present_Value of the monitored object is changed by an amount less than the COV increment and it is verified that no COV notification is received. The Present_Value is then changed by an amount greater than the COV increment and a notification shall be received.

The Present_Value may be changed by placing the Loop Out_Of_Service and writing directly to the Present_Value. For implementations where this option is not ~~possible an~~ *possible, an* alternative trigger mechanism shall be provided to accomplish this task, such as changing the Setpoint or the Setpoint_Reference. All of these methods are equally acceptable.

The object identifier of the Loop object being tested is designated as L in the test steps below.

Test Steps:

1. TRANSMIT SubscribeCOV-Request,
   'Subscriber Process Identifier' = (any value > 0 chosen by the TD),
   'Monitored Object Identifier' = L,
   'Issue Confirmed Notifications' = TRUE,
   'Lifetime' = 0
2. RECEIVE BACnet-SimpleACK-PDU
3. RECEIVE ConfirmedCOVNotification-Request,
   'Subscriber Process Identifier' = (the same value used in step 1),
   'Initiating Device Identifier' = IUT,
   'Monitored Object Identifier' = L,
   'Time Remaining' = 0,
   'List of Values' = (the initial Present_Value, initial Status_Flags, initial Setpoint, and initial Controlled_Variable_Value)
4. TRANSMIT BACnet-SimpleACK-PDU
5. TRANSMIT ReadProperty-Request,
   'Object Identifier' = L,
   'Property Identifier' = COV_Increment
6. RECEIVE BACnet-ComplexACK-PDU,
   'Object Identifier' = L,
   'Property Identifier' = COV_Increment,
   'Property Value' = (a value "increment" that will be used below)
7. IF (Out_Of_Service is writable) THEN
   WRITE X, Out_Of_Service = TRUE
   ~~RECEIVE BACnet-SimpleACK-PDU~~
   BEFORE **Notification Fail Time**
   RECEIVE ConfirmedCOVNotification-Request,
   'Subscriber Process Identifier' = (the same value used in step 1),
   'Initiating Device Identifier' = IUT,
   'Monitored Object Identifier' = L,
   'Time Remaining' = 0,
   'List of Values' = (the initial Present_Value, new Status_Flags, initial Setpoint,

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

and initial Controlled_Variable_Value)

~~8.~~ TRANSMIT BACnet-SimpleACK-PDU

[the following steps have been renumbered]

8. IF (Present_Value is now writable) THEN
WRITE X, Present_Value = (any value that differs from "initial Present_Value" by less than "increment")
~~RECEIVE BACnet-SimpleACK-PDU~~
ELSE
MAKE (Present_Value = any value that differs from "initial Present_Value" by less than "increment")

9. WAIT **NotificationFailTime**

10. CHECK (verify that no COV notification was transmitted)

11. IF (Present_Value is now writable) THEN
WRITE X, Present_Value = (any value that differs from "initial Present_Value" by an amount greater than "increment")
~~RECEIVE BACnet-SimpleACK-PDU~~
ELSE
MAKE (Present_Value = any value that differs from "initial Present_Value" by an amount greater than "increment")

12. BEFORE **NotificationFailTime**
RECEIVE ConfirmedCOVNotification-Request,
'Subscriber Process Identifier' = (the same value used in step 1),
'Initiating Device Identifier' = IUT,
'Monitored Object Identifier' = L,
'Time Remaining' = 0,
'List of Values' = (the new Present_Value, new Status_Flags, initial Setpoint, and initial Controlled_Variable_Value)

13. TRANSMIT BACnet-SimpleACK-PDU

14. TRANSMIT SubscribeCOV-Request,
'Subscriber Process Identifier' = (the same value used in step 1),
'Monitored Object Identifier' = L

15. RECEIVE BACnet-SimpleACK-PDU

16. IF (Out_Of_Service is writable) THEN
WRITE L, Out_Of_Service = FALSE
~~RECEIVE BACnet-SimpleACK-PDU~~

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135.1-2009*g*-2 Remove Recipient Test.

Rationale
The Recipient property was previously removed from the Event Enrollment object. The corresponding test should be removed from 135.1.

[Change **Clause 9.4.4**, p. 206]

9.4.4 ConfirmedEventNotification Without a Notification Class Parameter

This test has been removed.
~~Test Steps:~~

~~1. TRANSMIT ConfirmedEventNotification-Request,~~
~~'Process Identifier' = (any valid process identifier),~~
~~'Initiating Device Identifier' = TD,~~
~~'Event Object Identifier' = (any Event Enrollment object),~~
~~'Time Stamp' = (current time using the DateTime format),~~
~~'Priority' = (any valid priority),~~
~~'Event Type' = (any standard event type),~~
~~'Notify Type' = ALARM | EVENT,~~
~~'AckRequired' = FALSE,~~
~~'From State' = NORMAL,~~
~~'To State' = (any non-normal state appropriate to the event type),~~
~~'Event Values' = (any values appropriate to the event type)~~
~~2. RECEIVE BACnet-SimpleACK-PDU~~
~~3. CHECK (for any vendor-defined observable actions)~~

135.1-2009*g*-3 Correcting Errors in Routing Tests.

Rationale
Errors have identified in a number of routing tests in ANSI/ASHRAE Standard 135.1-2009. The most common types of errors are the following:

- Hop Count field present in a message when it should not be present
- Use of the DESTINATION keyword instead of DA
- Use of the SOURCE keyword instead of SA

Note that DA and SA specify the MAC addresses that should be present in a packet. It is often important to specify the MAC addresses in routing tests.

[Change **Clause 10.2.2.3**, p. 316]

10.2.2.3 Forward I-Could-Be-Router-To-Network

BACnet Reference Clause: 6.6.3.4.

Purpose: To verify that the IUT will forward a received I-Could-Be-Router-To-Network message to the intended recipient.

Test Steps:

1. TRANSMIT PORT B,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = HR2-4,
   DNET = 1,
   DADR = D1A,
   Hop Count = 255,
   I-Could-Be-Router-To-Network,
   Network Number = 4,
   Performance Index = 6
2. RECEIVE PORT A,
   ~~DESTINATION~~ *DA* = D1A,
   ~~SOURCE~~ *SA* = IUT,
   SNET = 2,
   SADR = HR2-4,
   ~~Hop Count = (any integer x: 0 < x < 255),~~
   I-Could-Be-Router-To-Network,
   Network Number = 4,
   Performance Index = 6

[Change **Clause 10.2.2.7.2**, p. 324]

10.2.2.7.2 Unknown Network Layer Message Type

Purpose: To verify that the IUT will reject a network layer message ~~with an unknown message type~~ *directed to the IUT that contains an unknown message type in the range of message types reserved for use by ASHRAE.*

Test Steps:

1. TRANSMIT PORT A,
   DESTINATION = IUT,
   SOURCE = ~~D1A~~ *TD*,
   Message Type = (any value ~~from X'0A' to X'7F~~ *in the range reserved for use by ASHRAE that is undefined in the protocol revision claimed by the device*)

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

2. RECEIVE PORT A,
   DESTINATION = ~~D1A~~ *TD*,
   SOURCE = IUT,
   Reject-Message-To-Network,
   Reject Reason = 3 (unknown network layer message type),
   DNET = ~~1~~ *any value*

[Change **Clause 10.2.3.2**, pp. 325]

10.2.3.2 Route Message from a Local Device to a Local Device

Purpose: To verify that the IUT can route a unicast message from a local device on Network 1 to a device on Network 2.

Test Steps:

1. TRANSMIT PORT A,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = D1A,
   DNET = 2,
   DADR = D2C,
   Hop Count = 255,
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (any object identifier),
   'Property Identifier' = (any property of the specified object)
2. RECEIVE PORT B,
   ~~DESTINATION~~ *DA* = D2C,
   ~~SOURCE~~ *SA* = IUT,
   SNET = 1,
   SADR = D1A,
   ~~Hop Count = (any integer x: 0 < x < 255),~~
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (the object identifier used in step 1),
   'Property Identifier' = (the property identifier used in step 1)
3. TRANSMIT PORT B,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = D2C,
   DNET = 1,
   DADR = D1A,
   Hop Count = 255,
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (any object identifier),
   'Property Identifier' = (any property of the specified object)
4. RECEIVE PORT A,
   ~~DESTINATION~~ *DA* = D1A,
   ~~SOURCE~~ *SA* = IUT,
   SNET = 2,
   SADR = D2C,
   ~~Hop Count = (any integer x: 0 < x < 255),~~
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (the object identifier used in step ~~1~~ *3*),
   'Property Identifier' = (the property identifier used in step ~~1~~ *3*)

[Change **Clause 10.2.3.5**, p. 327]

10.2.3.5 Route Message from a Router to a Local Device

Purpose: To verify that the IUT can route a unicast message from a peer router to the destination device on a local network.

Test Steps:

1. TRANSMIT PORT A,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = R1-5,
   DNET = 2,
   DADR = D2C,
   SNET = 5,
   SADR = D5F,
   Hop Count = 254,
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (any object identifier),
   'Property Identifier' = (any property of the specified object)
2. RECEIVE PORT B,
   ~~DESTINATION~~ *DA* = D2C,
   ~~SOURCE~~ *SA* = IUT,
   SNET = 5,
   SADR = D5F,
   ~~Hop Count = (any integer x: 0 < x < 254),~~
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (the object identifier used in step 1),
   'Property Identifier' = (the property identifier used in step 1)

[Change **Clause 10.2.4.4**, p. 330]

10.2.4.4 Remote Broadcast from a Local Device to a Directly-Connected Network

Purpose: To verify that the IUT properly forwards remote broadcast messages that originate on a local network and are directed to another local network.

Test Steps:

1. TRANSMIT PORT B,
   ~~DESTINATION~~ *DA* = LOCAL BROADCAST,
   ~~SOURCE~~ *SA* = D2C,
   DNET = 1,
   DLEN = 0,
   Hop Count = 255,
   BACnet-Unconfirmed-Request-PDU,
   'Service Choice' = Who-Is
2. RECEIVE PORT A,
   ~~DESTINATION~~ *DA* = LOCAL BROADCAST,
   ~~SOURCE~~ *SA* = IUT,
   SNET = 2,
   SADR = D2C,
   ~~Hop Count = (any integer x: 0 < x < 255),~~

BACnet-Unconfirmed-Request-PDU,
'Service Choice' = Who-Is

[Change **Clause 10.2.4.6**, p. 331]

10.2.4.6 Remote Broadcast from a Remote Device to a Directly-Connected Network

Purpose: To verify that the IUT properly forwards remote broadcast messages that originate on a remote network and are directed to a directly-connected network.

Test Steps:

1. TRANSMIT PORT B,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = R2-3,
   DNET = 1,
   DLEN = 0,
   SNET = 3,
   SADR = D3D,
   Hop Count = 254,
   BACnet-Unconfirmed-Request-PDU,
   'Service Choice' = Who-Is
2. RECEIVE PORT A,
   ~~DESTINATION~~ *DA* = LOCAL BROADCAST,
   ~~SOURCE~~ *SA* = IUT,
   SNET = 3,
   SADR = D3D,
   ~~Hop Count = (any integer x,: 0 < x < 254),~~
   BACnet-Unconfirmed-Request-PDU,
   'Service Choice' = Who-Is

[Change **Clause 10.2.4.8**, p. 332]

10.2.4.8 Remote Broadcast that Should Be Ignored

Purpose: To verify that the IUT ignores broadcast messages ~~that are~~ intended for a remote network that is reachable through the same port ~~that~~ the message was received from.

Test Steps:

1. TRANSMIT PORT B,
   ~~DESTINATION~~ *DA* = LOCAL BROADCAST,
   ~~SOURCE~~ *SA* = D2C,
   DNET = 3,
   DLEN = ~~(any valid MAC address length)~~ *0*,
   ~~DADR = (any valid MAC address the agrees with DLEN),~~
   Hop Count = 255,
   BACnet-Unconfirmed-Request-PDU,
   'Service Choice' = Who-Is
   ~~ReadProperty-Request ,~~
   ~~'Object Identifier' = (any BACnet object),~~
   ~~'Property Identifier' = (any property of the specified object)~~
2. CHECK (verify that the IUT does not forward~~ed~~ this message)

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Change **Clause 10.2.6**, pp. 333]

10.2.6 Network Layer Priority

BACnet Reference Clauses: 6.1, 6.2.2, and 6.5.4.

Purpose: To verify that the IUT can process and forward messages with all network priorities.

Test Steps:

1. TRANSMIT PORT A,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = D1A,
   Priority = B'00',
   DNET = 2,
   DADR = D2C,
   Hop Count = 255,
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (any object identifier),
   'Property Identifier' = (any property of the specified object)
2. RECEIVE PORT B,
   ~~DESTINATION~~ *DA* = D2C,
   ~~SOURCE~~ *SA* = IUT,
   Priority = B'00',
   SNET = 1,
   SDR = D1A,
   ~~Hop Count = (any integer x: 0 < x < 255),~~
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (the object identifier used in step 1),
   'Property Identifier' = (the property identifier used in step 1)
3. TRANSMIT PORT A,
   ~~DESTINATION~~ *DA* = IUT,
   ~~SOURCE~~ *SA* = D1A,
   Priority = B'01',
   DNET = 2,
   DADR = D2C,
   Hop Count = 255,
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (any object identifier),
   'Property Identifier' = (any property of the specified object)
4. RECEIVE PORT B,
   ~~DESTINATION~~ *DA* = D2C,
   ~~SOURCE~~ *SA* = IUT,
   Priority = B'01',
   SNET = 1,
   SDR = D1A,
   ~~Hop Count = (any integer x: 0 < x < 255),~~
   BACnet-Confirmed-Request-PDU,
   'Service Choice' = ReadProperty-Request,
   'Object Identifier' = (the object identifier used in step 3),
   'Property Identifier' = (the property identifier used in step 3)
5. TRANSMIT PORT A,
   ~~DESTINATION~~ *DA* = IUT,

~~SOURCE~~ *SA* = D1A,
Priority = B'10',
DNET = 2,
DADR = D2C,
Hop Count = 255,
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = (any object identifier),
'Property Identifier' = (any property of the specified object)

6. RECEIVE PORT B,
~~DESTINATION~~ *DA* = D2C,
~~SOURCE~~ *SA* = IUT,
Priority = B'10',
SNET = 1,
SDR = D1A,
~~Hop Count = (any integer x: 0 < x < 255),~~
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = (the object identifier used in step 5),
'Property Identifier' = (the property identifier used in step 5)
7. TRANSMIT PORT A,
~~DESTINATION~~ *DA* = IUT,
~~SOURCE~~ *SA* = D1A,
Priority = B'11',
DNET = 2,
DADR = D2C,
Hop Count = 255,
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = (any object identifier),
'Property Identifier' = (any property of the specified object)
8. RECEIVE PORT B,
~~DESTINATION~~ *DA* = D2C,
~~SOURCE~~ *SA* = IUT,
Priority = B'11',
SNET = 1,
SDR = D1A,
~~Hop Count = (any integer x: 0 < x < 255),~~
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = (the object identifier used in step 7),
'Property Identifier' = (the property identifier used in step 7)

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135.1-2009*g*-4 Change the Ignore Process ID Test.

Rationale
Tests need to be updated as a result of changes made by Addendum 135-2004b-8.

[Delete **Clause 9.1.2.2,** Unsuccessful Alarm Acknowledgment of Confirmed Event Notifications Because the 'Acknowledging Process Identifier' is Invalid, p 196, renumbering subsequent clauses]

[Insert new **Clause 9.1.1.X1**, p.194, renumbering subsequent clauses]

9.1.1.X1 Successful Alarm Acknowledgment of Confirmed Event Notifications Using an Unknown 'Acknowledging Process Identifier' Parameter

Purpose: To verify the successful acknowledgment of an alarm signaled by a ConfirmedEventNotification when the acknowledgement contains a mismatched or unknown 'Acknowledging Process Identifier' parameter.

Test Concept: An alarm is triggered that causes the IUT to notify the TD and at least one other device. The TD acknowledges the alarm with a mismatched 'Acknowledging Process Identifier' (the Process Identifier associated with another recipient), or an unknown 'Acknowledging Process Identifier' (a Process Identifier not associated with any recipient), and verifies that the acknowledgment is properly noted by the IUT. This test should be performed twice, once with a mismatched Process Identifier and once with an unknown Process Identifier.

Configuration Requirements: The IUT shall be configured with at least one object, Object1, that can detect alarm conditions and send confirmed notifications. The Acked_Transitions property shall have the value (TRUE,TRUE,TRUE), indicating that all transitions have been acknowledged. The TD and at least one other BACnet device shall be recipients of the alarm notification and shall use different Process Identifiers.

This test is recommended for all BACnet devices that execute AcknowledgeAlarm but is required only for those that claim conformance to Protocol_Revision 5 or greater.

Test Steps:

1. VERIFY (Object1), Acked_Transitions = (TRUE,TRUE,TRUE)
2. MAKE (a change that triggers the detection of an alarm event in the IUT)
3. BEFORE **Notification-Fail-Time**
   RECEIVE ConfirmedEventNotification-Request,
   'Process Identifier' = (any Process ID),
   'Initiating Device Identifier' = IUT,
   'Event Object Identifier' = Object1,
   'Time Stamp' = (the current time or sequence number),
   'Notification Class' = (the Notification Class configured for this event),
   'Priority' = (the priority configured for this event),
   'Event Type' = (any valid event type),
   'Notify Type' = ALARM or EVENT,
   'AckRequired' = TRUE,
   'From State' = (any appropriate event state),
   'To State' = (any appropriate event state),
   'Event Values' = (values appropriate to the event type)
4. TRANSMIT BACnet-SimpleACK-PDU
5. RECEIVE
   DESTINATION = (at least one device other than the TD),
   SOURCE = IUT,
   ConfirmedEventNotification-Request,
   'Process Identifier' = (any Process ID),

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

'Initiating Device Identifier' = IUT,
'Event Object Identifier' = Object1,
'Time Stamp' = (the current time or sequence number),
'Notification Class' = (the notification class configured for this event),
'Priority' = (the priority configured for this event),
'Event Type' = (any valid event type),
'Notify Type' = ALARM | EVENT,
'AckRequired' = TRUE,
'From State' = (any appropriate event state),
'To State' = (any appropriate event state),
'Event Values' = (values appropriate to the event type)

6. TRANSMIT
   DESTINATION = IUT,
   SOURCE = (DESTINATION in step 5),
   BACnet-SimpleACK-PDU
7. VERIFY (Object1), Acked_Transitions = (one bit FALSE, the others TRUE)
8. TRANSMIT AcknowledgeAlarm-Request,
   'Acknowledging Process Identifier' = (Any mismatched or unknown value),
   'Event Object Identifier' = Object1,
   'Event State Acknowledged' = (the state specified in the 'To State' parameter of the notification),
   'Time Stamp' = (the timestamp conveyed in the notification),
   'Time of Acknowledgment' = (the current timestamp)
9. RECEIVE BACnet-SimpleACK-PDU
10. BEFORE **Notification-Fail-Time**
    RECEIVE ConfirmedEventNotification-Request,
    'Process Identifier' = (any Process ID),
    'Initiating Device Identifier' = IUT,
    'Event Object Identifier' = Object1,
    'Time Stamp' = (the current time or sequence number),
    'Notification Class' = (the Notification Class configured for this event),
    'Priority' = (the priority configured for this event),
    'Event Type' = (any valid event type),
    'Notify Type' = ACK_NOTIFICATION,
    'To State' = (any appropriate event state)
11. TRANSMIT BACnet-SimpleACK-PDU
12. RECEIVE
    DESTINATION = (at least one device other than the TD),
    SOURCE = IUT,
    ConfirmedEventNotification-Request,
    'Process Identifier' = (any Process ID),
    'Initiating Device Identifier' = IUT,
    'Event Object Identifier' = Object1,
    'Time Stamp' = (the current time or sequence number),
    'Notification Class' = (the notification class configured for this event),
    'Priority' = (the priority configured for this event),
    'Event Type' = (any valid event type),
    'Notify Type' = ACK_NOTIFICATION,
    'To State' = (any appropriate event state)
13. TRANSMIT
    DESTINATION = IUT,
    SOURCE = (DESTINATION in step 5),
    BACnet-SimpleACK-PDU
14. VERIFY (Object1), Acked_Transitions = (TRUE,TRUE,TRUE)

Notes to Tester: The ConfirmedEventNotification-Request messages can be received in either order.

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Insert new **Clause 9.1.1.X2**, p.194]

9.1.1.X2 Successful Alarm Acknowledgment of Unconfirmed Event Notifications Using an Unknown 'Acknowledging Process Identifier' Parameter

Purpose: To verify the successful acknowledgment of an alarm signaled by an UnconfirmedEventNotification when the acknowledgement contains a mismatched or unknown 'Acknowledging Process Identifier' parameter.

Test Concept: An alarm is triggered that causes the IUT to notify the TD and at least one other device. The TD acknowledges the alarm with a mismatched 'Acknowledging Process Identifier' (the Process Identifier associated with another recipient), or unknown (a Process Identifier not associated with any recipient), and verifies that the acknowledgment is properly noted by the IUT. This test should be performed twice, once with a mismatched Process Identifier and once with an unknown Process Identifier.

Configuration Requirements: The IUT shall be configured with at least one object, Object1, that can detect alarm conditions and send unconfirmed notifications. The Acked_Transitions property shall have the value (TRUE,TRUE,TRUE), indicating that all transitions have been acknowledged. The TD and at least one other BACnet device shall be recipients of the alarm notification, configured to receive different Process Identifiers.

This test is recommended for all BACnet devices that execute AcknowledgeAlarm but is required only for those that claim conformance to Protocol_Revision 5 or greater.

Test Steps:

1. VERIFY (Object1), Acked_Transitions = (TRUE,TRUE,TRUE)
2. MAKE (a change that triggers the detection of an alarm event in the IUT)
3. BEFORE **Notification-Fail-Time**

RECEIVE UnconfirmedEventNotification-Request,
'Process Identifier' = (any Process ID),
'Initiating Device Identifier' = IUT,
'Event Object Identifier' = Object1,
'Time Stamp' = (the current time or sequence number),
'Notification Class' = (the Notification Class configured for this event),
'Priority' = (the priority configured for this event),
'Event Type' = (any valid event type),
'Notify Type' = ALARM or EVENT,
'AckRequired' = TRUE,
'From State' = (any appropriate event state),
'To State' = (any appropriate event state),
'Event Values' = (values appropriate to the event type)
RECEIVE
DESTINATION = (at least one device other than the TD),
SOURCE = IUT,
UnconfirmedEventNotification-Request,
'Process Identifier' = (any Process ID),
'Initiating Device Identifier' = IUT,
'Event Object Identifier' = (the object detecting the alarm),
'Time Stamp' = (the current time or sequence number),
'Notification Class' = (the notification class configured for this event),
'Priority' = (the priority configured for this event),
'Event Type' = (any valid event type),
'Notify Type' = ALARM | EVENT,
'AckRequired' = TRUE,
'From State' = (any appropriate event state),
'To State' = (any appropriate event state),

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

'Event Values' = (values appropriate to the event type)

5. VERIFY (Object1), Acked_Transitions = (one bit FALSE, the others TRUE)
6. TRANSMIT AcknowledgeAlarm-Request,
   'Acknowledging Process Identifier' = (Any mismatched or unknown value),
   'Event Object Identifier' = Object1,
   'Event State Acknowledged' = (the state specified in the 'To State' parameter of the notification),
   'Time Stamp' = (the timestamp conveyed in the notification),
   'Time of Acknowledgment' = (the current timestamp)
7. RECEIVE BACnet-SimpleACK-PDU
8. BEFORE **Notification-Fail-Time**
   RECEIVE UnconfirmedEventNotification-Request,
   'Process Identifier' = (any Process ID),
   'Initiating Device Identifier' = IUT,
   'Event Object Identifier' = Object1,
   'Time Stamp' = (the current time or sequence number),
   'Notification Class' = (the Notification Class configured for this event),
   'Priority' = (the priority configured for this event),
   'Event Type' = (any valid event type),
   'Notify Type' = ACK_NOTIFICATION,
   'To State' = (any appropriate event state)
   RECEIVE
   DESTINATION = (at least one device other than the TD),
   SOURCE = IUT,
   UnconfirmedEventNotification-Request,
   'Process Identifier' = (any Process ID),
   'Initiating Device Identifier' = IUT,
   'Event Object Identifier' = Object1,
   'Time Stamp' = (the current time or sequence number),
   'Notification Class' = (the notification class configured for this event),
   'Priority' = (the priority configured for this event),
   'Event Type' = (any valid event type),
   'Notify Type' = ACK_NOTIFICATION,
   'To State' = (any appropriate event state)
9. VERIFY (Object1), Acked_Transitions = (TRUE,TRUE,TRUE)

Note to Tester: The UnconfirmedEventNotification-Request messages can be received in either order.

© 2011 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. (www.ashrae.org). For personal use only.
Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135.1-2009*g*-5 Add Max Info Frames Check.

Rationale
There is no test that explicitly tests max-info-frame functionality. This test verifies that devices do not hog the token for too long.

[Add new **Clause 12.1.1.9.X**, p.389]

12.1.1.9.X Max Info Frame Check

Dependencies: None

BACnet Reference Clauses: 9.5.3 and 9.5.6.5

Purpose: This check verifies that the MS/TP Master Node State Machine does not issue more than $\mathbf{N_{max\_info\_frames}}$ information frames between the time the IUT receives a Token and either the time it passes the Token or it initiates a Poll For Master. Unlike tests, checks are not constructed of test steps, but rather conditions that must hold true through the complete testing process. As such, checks are periodically verified during or after the execution of tests.

Configuration Recommendations: If the Max_Info_Frames property of the Device object is configurable, it is recommended that this property be set to its minimum setting for the performance of at least some tests involving the MS/TP port being tested.

Check conditions: Monitor the MS/TP LAN during operations where the IUT would be expected to issue a number of information frames; if the IUT emits more information frames than:

a) the configured value for Max_Info_Frames in the interval between receiving and passing the Token (with multiple masters on the LAN), or
b) the configured value for Max_Info_Frames in the interval between receiving the Token and issuing PFM (with multiple masters on the LAN), or
c) the configured value for Max_Info_Frames in the interval between any two consecutive Poll For Master frames except the interval between the issuance of a Poll For Master to (TS-1) modulo Max_Master and a Poll For Master to (TS+1) modulo Max_Master, (with the IUT as the only master on the LAN), or
d) 52 times the configured value for Max_Info_Frames in the interval between a Poll For Master frame issued to (TS-1) modulo Max_Master, and the subsequent Poll For Master frame to (TS+1) modulo Max_Master (with the IUT as the only master on the LAN),

then the IUT shall fail this check.

Note to Tester: The value 52 is used in d) because an error in the MS/TP state machine originally defined in Standard 135-1995 caused the Token to be passed 52 times between Poll For Master cycles, instead of 50 times.