

BSR/ASHRAE Addendum *m*
to ANSI/ASHRAE Standard 135-2004

Public Review Draft

ASHRAE® Standard

Proposed Addendum *m* to Standard 135-2004, *BACnet*®—A *Data Communication Protocol for Building Automation and Control Networks*

First Public Review (September 2007)
(Draft Shows Proposed Changes
to Current Standard)

This draft has been recommended for public review by the responsible project committee. To submit a comment on this proposed addendum, use the comment form and instructions provided with this draft. The draft is subject to modification until it is approved for publication by the responsible project committee, the ASHRAE Standards Committee, and the Board of Directors. Then it will be submitted to the American National Standards Institute Board of Standards Review (BSR) for approval. Until this time, the current edition of the standard (as modified by any published addenda on the ASHRAE web site) remains in effect. The current edition of any standard may be purchased from the ASHRAE Bookstore @ <http://www.ashrae.org> or by calling 404-636-8400 or 1-800-527-4723 (for orders in the U.S. or Canada).

This standard is under continuous maintenance. To propose a change to the current standard, use the change submittal form available on the ASHRAE web site @ <http://www.ashrae.org>.

The appearance of any technical data or editorial material in this public review document does not constitute endorsement, warranty, or guaranty by ASHRAE of any product, service, process, procedure, or design, and ASHRAE expressly disclaims such.

© September 21, 2007. This draft is covered under ASHRAE copyright. Permission to reproduce or redistribute all or any part of this document must be obtained from the ASHRAE Manager of Standards, 1791 Tullie Circle, NE, Atlanta, GA 30329-2305. Phone: 404-636-8400, Ext1125. Fax: 404-321-5478. E-mail: standards.section@ashrae.org.

AMERICAN SOCIETY OF HEATING,
REFRIGERATING AND AIR-CONDITIONING
ENGINEERS, INC.
1791 Tullie Circle, NE · Atlanta GA 30329-2305



[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 public review. 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.

- 135-2004m-1. Resolve Foreign Device registration grace period and remaining time ambiguities, p. 1.**
- 135-2004m-2. Improve Clause 5 FillWindow segment timeout constraints, p. 2.**
- 135-2004m-3. Clarify the Priority Filter parameter in the GetEventEnrollment service request, p. 4.**
- 135-2004m-4. Allow alarms to be re-acknowledged successfully, p. 5.**
- 135-2004m-5. Add requirements to Alarm and Event BIBBs, p. 6.**
- 135-2004m-6. Remove B-BC requirements for BIBBs without use cases, p. 8.**
- 135-2004m-7. Clarify that a device may support only the ReinitializeDevice restart choices, p. 9.**
- 135-2004m-8. Clarify DeviceCommunicationsControl and ReinitializeDevice interactions, p. 10.**
- 135-2004m-9. Define "object," p. 11.**
- 135-2004m-10. Add a Deadband property to the Loop object, p.12.**
- 135-2004m-11. Correct the TO-FAULT conditions in the Life Safety objects' Reliability properties, p. 14.**
- 135-2004m-12. Clarify the Trend Log's acquisition of Status_Flags, p.15.**

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135-2004 and Addenda is indicated through the use of *italics*, while deletions are indicated by ~~striketrough~~. Where entirely new subclauses are proposed to be added, plain type is used throughout. Only this new and deleted text is open to comment as 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.

135-2004m-1. Resolve Foreign Device registration grace period and remaining time ambiguities.

Rationale

When an Annex J Foreign Device registers with a BBMD, it supplies a period of time for which the registration should remain active. The BBMD adds a 30 second grace period to this time, after which it will purge the registrant's entry in the Foreign Device Table (FDT). The amount of time remaining for registrations can be read from the FDT, but it has not been clear whether this time included the 30 second grace period.

Addendum 135-2004m-1

[Change Annex **J.2.8.1**, pp. 567-568.]

J.2.8.1 Read-Foreign-Device-Table-Ack: Format

The Read-Foreign-Device-Table-Ack message consists of four fields:

BVLC Type: 1-octet X'81' BVLL for BACnet/IP
BVLC Function: 1-octet X'07' Read-Foreign-Device-Table-Ack
BVLC Length: 2-octets L Length L, in octets, of the BVLL message
List of FDT Entries: N*10-octets

N indicates the number of entries in the FDT whose contents are being returned. Each returned entry consists of the 6-octet B/IP address of the registrant; the 2-octet Time-to-Live value supplied at the time of registration; and a 2-octet value representing the number of seconds remaining before the BBMD will purge the registrant's FDT entry if no re-registration occurs. *The time remaining includes the 30 second grace period as defined in J.5.2.3.*

[Change Annex **J.5.2.1**, p. 571.]

J.5.2.1 Foreign Device Table

Each device that registers as a foreign device shall be placed in an entry in the BBMD's Foreign Device Table (FDT). Each entry shall consist of the 6-octet B/IP address of the registrant; the 2-octet Time-to-Live value supplied at the time of registration; and a 2-octet value representing the number of seconds remaining before the BBMD will purge the registrant's FDT entry if no re-registration occurs. ~~This value will~~ *The number of seconds remaining shall be initialized to the 2-octet Time-to-Live value supplied at the time of registration, registration plus 30 seconds (see J.5.2.3), with a maximum of 65,535.*

...

[Change Annex **J.5.2.3**, p. 572.]

J.5.2.3 Foreign Device Table Timer Operation

Upon receipt of a BVLL Register-Foreign-Device message, a BBMD shall start a timer with a value equal to the Time-to-Live parameter supplied plus a fixed grace period of 30 seconds. If, within the period during which the timer is active, another BVLL Register-Foreign-Device message from the same device is received, the timer shall be reset and restarted. If the time expires without the receipt of another BVLL Register-Foreign-Device message from the same foreign device, the FDT entry for this device shall be cleared.

~~Upon receipt of a BVLC Result message containing a result code of X'0000' indicating the successful completion of the registration, a foreign device shall start a timer with a value equal to the Time to Live parameter of the preceding Register-Foreign-Device message. At the expiration of the timer, the foreign device shall re-register with the BBMD by sending a BVLL Register Foreign-Device message.~~

135-2004*m*-2. Improve Clause 5 FillWindow segment timeout constraints.

Rationale

There is a potential for unnecessary segment timeouts, depending upon the implementation of the Clause 5.4.3 FillWindow function, caused by starting the SegmentTimer before calling FillWindow. Starting SegmentTimer after calling FillWindow may reduce this possibility.

Addendum 135-2004*m*-2

[Change Clause 5.4.4.2, p. 27.]

5.4.4.2 SEGMENTED_REQUEST

In the SEGMENTED_REQUEST state, the device waits for a BACnet-SegmentACK-PDU for one or more segments of a BACnet-Confirmed-Request-PDU.

DuplicateACK_Received

If a BACnet-SegmentACK-PDU whose 'server' parameter is TRUE is received from the network layer and InWindow('sequence-number' parameter of the BACnet-SegmentACK-PDU, InitialSequenceNumber) returns a value of FALSE,

then restart SegmentTimer and enter the SEGMENTED_REQUEST state to await an acknowledgment.

NewACK_Received

If a BACnet-SegmentACK-PDU whose 'server' parameter is TRUE is received from the network layer and InWindow('sequence-number' parameter of the BACnet-SegmentACK-PDU, InitialSequenceNumber) returns a value of TRUE and there is at least one segment remaining to send,

then set InitialSequenceNumber equal to the 'sequence-number' parameter of the BACnet-SegmentACK-PDU plus one, modulo 256; set ActualWindowSize equal to the 'actual-window-size' parameter of the BACnet-SegmentACKPDU; ~~restart SegmentTimer;~~ set SegmentRetryCount to zero; call FillWindow(InitialSequenceNumber) to transmit one or more BACnet-Confirmed-Request-PDUs containing the next ActualWindowSize segments of the message; *restart SegmentTimer;* and enter the SEGMENTED_REQUEST state to await an acknowledgment.

FinalACK_Received

If a BACnet-SegmentACK-PDU whose 'server' parameter is TRUE is received from the network layer and InWindow('sequence-number' parameter of the BACnet-SegmentACK-PDU, InitialSequenceNumber) returns a value of TRUE and there are no more segments to send,

then stop SegmentTimer; start RequestTimer; and enter the AWAIT_CONFIRMATION state to await a reply.

Timeout

If SegmentTimer becomes greater than T_{seg} and SegmentRetryCount is less than N_{retry} ,

then increment SegmentRetryCount; ~~restart SegmentTimer;~~ call FillWindow(InitialSequenceNumber) to retransmit one or more BACnet-Confirmed-Request-PDUs containing the next ActualWindowSize segments of the message; *restart SegmentTimer;* and enter the SEGMENTED_REQUEST state to await an acknowledgment.

FinalTimeout

If SegmentTimer becomes greater than T_{seg} and SegmentRetryCount is greater than or equal to N_{retry} , then stop SegmentTimer; send CONF_SERV.confirm(-) to the local application program; and enter the IDLE state.

[Change Clause **5.4.5.4**, pp. 35-36.]

5.4.5.4 SEGMENTED_RESPONSE

In the SEGMENTED_RESPONSE state, the device waits for a BACnet-SegmentACK-PDU for a segment or segments of a BACnet-ComplexACK-PDU.

DuplicateACK_Received

If a BACnet-SegmentACK-PDU whose 'server' parameter is FALSE is received from the network layer and InWindow('sequence-number' parameter of the BACnet-SegmentACK-PDU, InitialSequenceNumber) returns a value of FALSE,

then restart SegmentTimer and enter the SEGMENTED_RESPONSE state to await an acknowledgment or timeout.

NewACK_Received

If a BACnet-SegmentACK-PDU whose 'server' parameter is FALSE is received from the network layer and InWindow('sequence-number' parameter of the BACnet-SegmentACK-PDU, InitialSequenceNumber) returns a value of TRUE and there is at least one segment remaining to send,

then set InitialSequenceNumber equal to the 'sequence-number' parameter of the BACnet-SegmentACK-PDU plus one, modulo 256; set ActualWindowSize equal to the 'actual-window-size' parameter of the BACnet-SegmentACKPDU; ~~restart SegmentTimer;~~ set SegmentRetryCount to zero; call FillWindow(InitialSequenceNumber) to issue an N-UNITDATA.request with 'data_expecting_reply' = TRUE to transmit one or more BACnet-ComplexACK-PDUs containing the next ActualWindowSize segments of the message; *restart SegmentTimer*; and enter the SEGMENTED_RESPONSE state to await an acknowledgment.

FinalACK_Received

If a BACnet-SegmentACK-PDU whose 'server' parameter is FALSE is received from the network layer and InWindow('sequence-number' parameter of the BACnet-SegmentACK-PDU, InitialSequenceNumber) returns a value of TRUE and there are no more segments to send,

then stop SegmentTimer and enter the IDLE state.

Timeout

If SegmentTimer becomes greater than T_{seg} and SegmentRetryCount is less than Number_Of_APDU_Retries,

then increment SegmentRetryCount; ~~restart SegmentTimer;~~ call FillWindow(InitialSequenceNumber) to reissue an N-UNITDATA.request with 'data_expecting_reply' = TRUE to transmit one or more BACnet-ComplexACK-PDUs containing the next ActualWindowSize segments of the message; *restart SegmentTimer*; and enter the SEGMENTED_RESPONSE state to await an acknowledgment.

FinalTimeout

If SegmentTimer becomes greater than T_{seg} and SegmentRetryCount is greater than or equal to Number_Of_APDU_Retries,

then stop the SegmentTimer, and enter the IDLE state.

135-2004*m*-3. Clarify the Priority Filter parameter in the GetEventEnrollment service request.

Rationale

Since the entry in the GetEnrollmentSummary response indicates the current Event State of the event generating object, the Priority parameter should indicate the priority of the corresponding transition. For example, if the Event State of the returned event generating object is High_Limit, then the value of the Priority parameter must be the To-OffNormal priority from the related Notification Class object.

Addendum 135-2004*m*-3

[Change Clause **13.11.1.1.5**, p. 282.]

13.11.1.1.5 Priority Filter

This parameter consists of two parts, MinPriority and MaxPriority, each of datatype Unsigned8. It provides a means of restricting the summary to only those event-initiating objects that can generate event notifications with a Priority as specified by this parameter. The 'Priority Filter' parameter consists of two parts, MinPriority and MaxPriority. All event-initiating objects, such that $\text{MinPriority} \leq \text{Priority} \leq \text{MaxPriority}$, shall be included in the summary. *For the purpose of this filter, the Priority checked by the filter is the Priority associated with the most recent transition.* If 'Priority Filter' is omitted, all event-initiating objects shall be summarized without regard to their Priority.

135-2004*m*-4. Allow alarms to be re-acknowledged successfully.

Rationale

The language in Clause 13.5.2 is unclear on whether or not an acknowledgement should succeed or fail if the latest transition has already been acknowledged. This is changed so that the AcknowledgeAlarm service will succeed even if the transition was previously acknowledged.

Addendum 135-2004*m*-4

[Change Clause **13.5.2**, p. 270.]

13.5.2 Service Procedure

After verifying the validity of the request, the responding BACnet-user shall attempt to locate the specified object. If the object exists and if the 'Time Stamp' parameter matches the most recent time for the event being acknowledged, then the bit in the Acked_Transitions property of the object that corresponds to the value of the 'Event State Acknowledged' parameter ~~is shall be set~~ ~~acknowledged by changing the bit value to one~~~~1~~, and a 'Result(+)' primitive shall be issued, ~~and an event notification with a 'Notify Type' parameter equal to ACK_NOTIFICATION shall be issued.~~ Otherwise, a 'Result(-)' primitive shall be issued. ~~If the acknowledgment was successful, causing a 'Result(+)' to be issued, then an event notification, with a 'Notify Type' parameter equal to ACK_NOTIFICATION, shall also be issued. The~~ An acknowledgment notification shall use the same type of service (confirmed or unconfirmed) directed to the same recipients to which the original confirmed or unconfirmed event notification was sent.

135-2004m-5. Add requirements to Alarm and Event BIBBs.

Rationale
 Several of the Alarm and Event BIBBs need additional requirements to better reflect the use of the required services.

Addendum 135-2004m-5

[Change Annex **K.22**, p. 580.]

K.2.2 BIBB - Alarm and Event-Notification Internal-B (AE-N-I-B)

Device B generates notifications about alarms and other events.

BACnet Service	Initiate	Execute
ConfirmedEventNotification	x	
UnconfirmedEventNotification	x	

Devices claiming conformance to AE-N-I-B shall also support either Intrinsic or Algorithmic reporting. Any device that supports the generation of event notifications that require operator acknowledgment ~~must~~ shall support ~~AE-ACK-B, AE-ACK-B and AE-INFO-B~~. Any device that supports the generation of TO-FAULT or TO-OFFNORMAL event notifications shall support AE-INFO-B.

Devices that only support generation of life safety and/or buffer-ready notifications cannot claim support for this BIBB.

[Change Annex **K.2.3**, p. 580.]

K.2.3 BIBB - Alarm and Event-Notification External-B (AE-N-E-B)

Device B contains an Event Enrollment object that monitors values in another device. Device B is capable of generating event notifications for alarm conditions based on value(s) in another device. Devices conforming to this BIBB ~~must~~ shall conform to DS-RP-A, AE-N-I-B, and ~~must~~ shall support at least 1 Event Enrollment object with an Object_Property_Reference property that supports references to properties in objects contained in other devices. Any device that supports the generation of event notifications that require operator acknowledgment ~~must~~ shall support ~~AE-ACK-B, AE-ACK-B and AE-INFO-B~~. Any device that supports the generation of TO-FAULT or TO-OFFNORMAL event notifications shall support AE-INFO-B.

Devices that only support Event Enrollment objects that only support generation of life safety and/or buffer-ready notifications cannot claim support for this BIBB.

[Change Annex **K.2.13**, p. 581.]

[Note: Annex **K.2.13** was modified in Addendum 135-2004d-4.]

K.2.13 BIBB - Alarm and Event-LifeSafety-B (AE-LS-B)

Life safety device B is able to generate life safety notifications and is able to process silence and reset operations on its life safety objects.

BACnet Service	Initiate	Execute
LifeSafetyOperation		x
ConfirmedEventNotification	x	
UnconfirmedEventNotification	x	

Devices claiming conformance to AE-LS-B shall support at least one instance of a Life Safety Point or Life Safety Zone object *and shall be able to generate ConfirmedEventNotification and UnconfirmedEventNotification service requests describing CHANGE_OF_LIFE_SAFETY event transitions.*

Any device that supports the generation of event notifications that require operator acknowledgment ~~must~~ shall support ~~AE-ACK-B~~ AE-ACK-B and AE-INFO-B. Any device that supports the generation of TO-FAULT or TO-OFFNORMAL event notifications shall support AE-INFO-B.

135-2004m-6. Remove B-BC requirements for BIBBs without use cases.

Rationale
 The standard requires the DS-COVU-A,B and DM-DOB-A BIBBs to be supported by B-BC devices, but there are no interoperable/configurable use cases for these BIBBs.

Addendum 135-2004m-6

[Change Annex L.7, p. 593.]

L.7 Profiles of the Standard BACnet Devices

The following tables indicate which BIBBs must be supported by each device type for each interoperability area.

	B-OWS	B-BC	B-AAC	B-ASC	B-SA	B-SS
Data Sharing	DS-RP-A,B	DS-RP-A,B	DS-RP-B	DS-RP-B	DS-RP-B	DS-RP-B
	DS-RPM-A	DS-RPM-A,B	DS-RPM-B	DS-WP-B	DS-WP-B	
	DS-WP-A	DS-WP-A,B	DS-WP-B			
	DS-WPM-A	DS-WPM-B	DS-WPM-B			
		DS-COVU-A,B				

...

	B-OWS	B-BC	B-AAC	B-ASC	B-SA	B-SS
Device &	DM-DDB-A,B	DM-DDB-A,B	DM-DDB-B	DM-DDB-B		
Network Mgmt	DM-DOB-A,B	DM-DOB-A,B	DM-DOB-B	DM-DOB-B		
	DM-DCC-A	DM-DCC-B	DM-DCC-B	DM-DCC-B		
	DM-TS-A	DM-TS-B or DM-UTC-B	DM-TS-B or DM-UTC-B			
	DM-UTC-A					
	DM-RD-A	DM-RD-B	DM-RD-B			
	DM-BR-A	DM-BR-B				
	NM-CE-A	NM-CE-A				

135-2004m-7. Clarify that a device may support only the ReinitializeDevice restart choices.

Rationale
 ReinitializeDevice originally supported a set of restart choices. Subsequently device backup and restore capability was added, but the intent was not to require that ReinitializeDevice always support backup and restore.

Addendum 135-2004m-7

[Change Clause **16.4**, p. 330.]

16.4 ReinitializeDevice Service

The ReinitializeDevice service is used by a client BACnet-user to instruct a remote device to reboot itself (cold start), reset itself to some predefined initial state (warm start), or to control the backup or restore procedure. Resetting or rebooting a device is primarily initiated by a human operator for diagnostic purposes. Use of this service during the backup or restore procedure is usually initiated on behalf of the user by the device controlling the backup or restore. Due to the sensitive nature of this service, a password may be required from the responding BACnet-user prior to executing the service.

A BACnet device may support the ReinitializeDevice service by supporting only the restart choices COLDSTART and WARMSTART. Support for the backup and restore features of this service is claimed separately.

[Change Annex **K.5.15**, p. 586.]

K.5.15 BIBB - Device Management-ReinitializeDevice-A (DM-RD-A)

Devices claiming conformance to DM-RD-A shall be able to initiate ReinitializeDevice requests containing the Password parameter. ~~This shall be both for warm and cold start.~~ *Devices claiming conformance to DM-RD-A are only required to support the WARMSTART and COLDSTART service choices.*

[Change Annex **K.5.16**, p. 586.]

K.5.16 BIBB - Device Management- ReinitializeDevice-B (DM-RD-B)

The B device performs reinitialization requests from the A device. The optional password field shall be supported.

BACnet Service	Initiate	Execute
ReinitializeDevice		x

Devices claiming conformance to DM-RD-B are only required to support the WARMSTART and COLDSTART service choices.

[Change Annex **K.5.17**, p. 586.]

K.5.17 BIBB - Device Management- Backup and Restore-A (DM-BR-A)

Devices claiming conformance to DM-BR-A are required to support all service choices of the ReinitializeDevice service. In addition, devices ~~Devices~~ claiming conformance to DM-BR-A ~~must~~ shall support the device A capabilities as described in 19.1.

[Change Annex **K.5.18**, p. 586.]

K.5.18 BIBB - Device Management- Backup and Restore-B (DM-BR-B)

Devices claiming conformance to DM-BR-B are required to support all service choices of the ReinitializeDevice service. In addition, devices ~~Devices~~ claiming conformance to DM-BR-B ~~must~~ shall support the device B capabilities as described in 19.1. Once a Restore procedure has been initiated on the device, the Read_Only property of configuration File objects shall contain the value FALSE and the File_Size property of the configuration File objects shall be writable if the size of the configuration file can change based on the device's configuration.

135-2004*m*-8. Clarify DeviceCommunicationControl and ReinitializeDevice interactions.

Rationale

Devices with communications disabled via DeviceCommunicationControl should ignore ReinitializeDevice requests containing a 'Reinitialized State of Device' parameter value other than COLDSTART or WARMSTART.

Addendum 135-2004*m*-8

[Change Clause 16.1.2, p. 326.]

16.1.2 Service Procedure

After verifying the validity of the request, including the password, the responding BACnet-user shall respond with a 'Result(+)' service primitive. ~~and, if the 'Enable/Disable' parameter is DISABLE, discontinue responding to any subsequent messages except DeviceCommunicationControl and, if supported, ReinitializeDevice messages and discontinue initiating messages. Communication shall be disabled until either the 'Time Duration' has expired or a valid DeviceCommunicationControl (with 'Enable/Disable' = ENABLE) or, if supported, a valid ReinitializeDevice message is received.~~

If the responding BACnet-user does not have a clock and the 'Time Duration' parameter is not set to "indefinite," the APDU shall be ignored and a 'Result(-)' service primitive shall be issued. If the password is invalid or absent when one is required, the APDU shall be ignored and a 'Result(-)' response primitive shall be issued.

If a Result(+) is returned and the 'Enable/Disable' parameter is DISABLE, communication shall be disabled by:
discontinuing the initiation of messages and,
discontinuing responding to subsequent messages except for:
DeviceCommunicationControl requests and, if supported,
ReinitializeDevice requests with a 'Reinitialized State of Device' of WARMSTART or COLDSTART.

If a Result(+) is returned and the 'Enable/Disable' parameter is DISABLE_INITIATION, communication shall be disabled by:
discontinuing the initiation of messages except for I-Am requests, which shall be initiated only in response to Who-Is messages.

Communication shall remain disabled until either:
the time specified by 'Time Duration' has expired,
a valid DeviceCommunicationControl with 'Enable/Disable' = ENABLE is received or, if supported,
a valid ReinitializeDevice message with a 'Reinitialized State of Device' of WARMSTART or COLDSTART is received.

135-2004*m*-9. Define "object".

Rationale

The sentence "These properties are required if the object supports intrinsic reporting" was found to be unclear as to whether it referred a specific instance of an object type or whether it referred to all objects of that type.

Addendum 135-2004*m*-9

[Add new Clause **3.2.38**, p. 4, and renumber subsequent clauses.]

3.2.37 object type: a generic classification of data that is defined by a set of properties.

3.2.38 object: *a specific instance of an object type. While an object type is identified by a unique Object_Type property, an object is identified by its Object_Identifier property.*

135-2004*m*-10. Add a Deadband property to the Loop object.

Rationale
 The Loop object requires a Deadband property in order to support intrinsic notification using the FLOATING_LIMIT algorithm.

Addendum 135-2004*m*-10

[Modify Table 12-20, p. 206.]

Table 12-20. Properties of the Loop Object Type

Property Identifier	Property Datatype	Conformance Code
...
Error_Limit	REAL	O ⁵
<i>Deadband</i>	<i>REAL</i>	<i>O⁵</i>
Event_Enable	BACnetEventTransitionBits	O ⁵
...

[Add new clauses 12.17.32.1 and 12.17.32.2, p. 211.]

12.17.32.1 Conditions for Generating a TO-OFFNORMAL Event

A TO-OFFNORMAL event is generated under these conditions:

- (a) the Error must exceed the Error_Limit for a minimum period of time, specified in the Time_Delay property, and
- (b) the TO-OFFNORMAL flag must be set in the Event_Enable property,

where Error is defined as $AbsoluteValue(Setpoint - Controlled_Variable_Value)$. From the perspective of the FLOATING_LIMIT algorithm, the limits are calculated as:

$$High_Diff_Limit = Error_Limit$$

$$Low_Diff_Limit = -Error_Limit$$

12.17.32.2 Conditions for Generating a TO-NORMAL Event

Once the Error_Limit has been exceeded, a TO-NORMAL event is generated under these conditions:

- (a) the Error must fall below the Error_Limit minus the Deadband for a minimum period of time, specified in the Time_Delay property, and
- (b) the TO-NORMAL flag must be set in the Event_Enable property,

where Error is defined as $AbsoluteValue(Setpoint - Controlled_Variable_Value)$.

[Add new Clause 12.17.33, p. 211, and renumber subsequent clauses.]

12.17.33 Deadband

This property, of type REAL, shall specify a positive offset from the Error_Limit property, where the offset defines a band around the Error (defined as the difference between the Setpoint and the Controlled_Variable_Value properties) within the range of $(Error_Limit - Deadband)$ and $((-Error_Limit)+Deadband)$. The Error must remain within this range for a TO-NORMAL event to be generated under these conditions:

- (a) the Error must fall below the Error_Limit minus Deadband, and
- (b) the Error must exceed the —Error_Limit plus the Deadband, and
- (c) the Error must remain within this range for a minimum period of time, specified in the Time_Delay property, and
- (d) the TO-NORMAL flag must be set in the Event_Enable property.

This property is required if intrinsic reporting is supported by this object.

[Change **Annex C**, p.460.]

```
LOOP ::= SEQUENCE {  
    ...  
    error-limit [34] REAL OPTIONAL,  
    deadband [25] REAL OPTIONAL,  
    event-enable [35] BACnetEventTransitionBits OPTIONAL,  
    ...  
}
```

[Change **Annex D.17**, p.477.]

D.17 Example of a Loop Object

```
...  
Property: Error_Limit = 5.0  
Property: Deadband = 1.0  
Property: Event_Enable = {TRUE, TRUE, TRUE}  
...
```

135-2004*m*-11. Correct the TO-FAULT conditions in the Life Safety objects' Reliability properties.

Rationale

The language for the Reliability property of the Life Safety Point and Life Safety Zone objects is incorrect in its description of generating TO-FAULT events.

Addendum 135-2004*m*-11

[Change Clause **12.15.10.1**, page 196]

12.15.10.1 Conditions for Generating a TO-FAULT Event

A TO-FAULT event is generated under these conditions:

- ~~(a) the TO-FAULT flag must be enabled in the Event_Enable property, and~~
- ~~(b) the Present_Value must equal at least one of the values in the Fault_Values list.~~
- (a) the Reliability property becomes not equal to NO_FAULT_DETECTED, and*
- (b) the TO-FAULT flag is enabled in the Event_Enable property.*

[Change Clause **12.16.10.1**, page 202]

12.16.10.1 Conditions for Generating a TO-FAULT Event

A TO-FAULT event is generated under these conditions:

- ~~(a) the TO-FAULT flag must be enabled in the Event_Enable property, and~~
- ~~(b) the Present_Value must equal at least one of the values in the Fault_Values list.~~
- (a) the Reliability property becomes not equal to NO_FAULT_DETECTED, and*
- (b) the TO-FAULT flag is enabled in the Event_Enable property.*

135-2004*m*-12. Clarify the Trend Log's acquisition of Status_Flags.

Rationale

The language describing the acquisition of Status_Flags by the Trend Log buffer needs clarification that it is recommended that the Status_Flags property be read in the same service request by which the data value is acquired.

Addendum 135-2004*m*-12

[Change Clause 12.25, p.246]

12.25 Trend Log Object Type

A Trend Log object monitors a property of a referenced object and, when predefined conditions are met, saves ("logs") the value of the property and a timestamp in an internal buffer for subsequent retrieval. The data may be logged periodically or upon a change of value. Errors that prevent the acquisition of the data, as well as changes in the status or operation of the logging process itself, are also recorded. Each timestamped buffer entry is called a trend log "record."

The referenced object may reside in the same device as the Trend Log object or in an external device. The referenced property's value may be recorded upon COV subscription or periodic poll. *If the value of the monitored object's Status_Flags property is available, then it may optionally be recorded along with the value of the referenced property.* ~~Where status flags are available (such as when the COVNotification or ReadPropertyMultiple services are used), they are also acquired and saved with the data.~~

[Change Clause 12.25.14, p.249]

12.25.14 Log_Buffer

This property is a list of up to Buffer_Size timestamped records of datatype BACnetLogRecord, each of which conveys a recorded data value, an error related to data-collection, or status changes in the Trend Log object. Each record has data fields as follows:

Timestamp The local date and time when the record was collected.

LogDatum The data value read from the monitored object and property, an error encountered in an attempt to read a value, or a change in status or operation of the Trend Log object itself.

StatusFlags ~~The Status_Flags property of the monitored object, if present and available atomically associated with the LogDatum data value. If the Status_Flags property is not present or not available atomically associated with the data value, this item shall not be included in the log record. If this field is present in the log record, then it shall contain the value of the Status_Flags property of the monitored object. If the monitored object is in a different device than the Trend Log object, then it is recommended that the Status_Flags and the data value in the monitored object property be acquired together with a single service request, such as COVNotification or ReadPropertyMultiple.~~