

Clarification Request

Request from: "Horst Hannappel" <Horst.Hannappel@mbs-software.de>

References: TODO-revise this line as appropriate:

BTL Test Plan 5.0.final, 135.1-2007 test 7.3.1.9

BTL Test Plan 9.0.final, 135.1-2009 test 7.3.1.9

Stage: ☒Request, ☐Listed, ☐Analysis, ☒Resolved

Actions necessitated: ☐Checklist/Test Plan change, ☒BTL Specified Tests change,

☐SSPC Interpretation required, ☐Implementation Guidelines change,

Date of BTL-WG Response: _____

☐All actions necessitated have been completed

Background:

135.1-2007 - 7.3.1.9 - Binary Object Elapsed Active Time Tests

135.1-2009 - 7.3.1.9 - Binary Object Elapsed Active Time Tests

7.3.1.9 Binary Object Elapsed Active Time Tests

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

BACnet Reference Clauses: 12.6.17, 12.6.18, 12.7.17, 12.7.18, 12.8.15, and 12.8.16.

Purpose: To verify that the properties of binary objects that collectively track active time function properly. If the Elapsed_Active_Time and Time_Of_Active_Time_Reset properties are not supported then this test shall be omitted. This test applies to Binary Input, Binary Output, and Binary Value objects.

Test Concept: The Present_Value of the binary object being tested is set to INACTIVE. The Elapsed_Active_Time property is checked to verify that it does not accumulate time while the object is in an INACTIVE state. The Present_Value is then set to ACTIVE. The Elapsed_Active_Time property is checked to verify that it is accumulating time while the object is in an ACTIVE state. The Present_Value is then set to INACTIVE and the Elapsed_Active_Time is reset. The Time_Of_Active_Time_Reset property is checked to verify that it has been updated.

Configuration Requirements: The object being tested shall be configured such that the Present_Value and Elapsed_Active_Time properties are writable or another means of changing these properties shall be provided.

Test Steps:

1. IF (Present_Value is writable) THEN
WRITE Present_Value = INACTIVE
VERIFY Present_Value = INACTIVE
ELSE
MAKE (Present_Value = INACTIVE)
2. TRANSMIT ReadProperty-Request,
'Object Identifier' = (the object being tested),
'Property Identifier' = Elapsed_Active_Time
3. RECEIVE ReadProperty-ACK,
'Object Identifier' = (the object being tested),
'Property Identifier' = Elapsed_Active_Time,
'Property Value' = (the elapsed active time, T_{ELAPSED} in seconds)
4. WAIT (1 minute)
5. TRANSMIT ReadProperty-Request,
'Object Identifier' = (the object being tested),
'Property Identifier' = Elapsed_Active_Time

6. RECEIVE ReadProperty-ACK,
 'Object Identifier' = (the object being tested),
 'Property Identifier' = Elapsed_Active_Time,
 'Property Value' = (the same $T_{ELAPSED}$ as step 3)
 7. IF (Present_Value is writable) THEN
 WRITE Present_Value = ACTIVE
 VERIFY Present_Value = ACTIVE
 ELSE
 MAKE (Present_Value = ACTIVE)
 8. WAIT (Internal Processing Fail Time + 30 seconds)
 9. TRANSMIT ReadProperty-Request,
 'Object Identifier' = (the object being tested),
 'Property Identifier' = Elapsed_Active_Time
 10. RECEIVE ReadProperty-ACK,
 'Object Identifier' = (the object being tested),
 'Property Identifier' = Elapsed_Active_Time,
 'Property Value' = $(T: (T_{ELAPSED} + 30) \square T \square (T_{ELAPSED} + 30 + \text{Internal Processing Fail Time}))$
 11. IF (Present_Value is writable) THEN
 WRITE Present_Value = INACTIVE
 VERIFY Present_Value = INACTIVE
 ELSE
 MAKE (Present_Value = INACTIVE)
 12. IF (Elapsed_Active_Time is writable) THEN
 WRITE Elapsed_Active_Time = 0
 VERIFY Elapsed_Active_Time = 0
 ELSE
 MAKE (Elapsed_Active_Time = 0)
 13. TRANSMIT ReadProperty-Request,
 'Object Identifier' = (the IUT's Device object),
 'Property Identifier' = Local_Date
 14. RECEIVE ReadProperty-ACK,
 'Object Identifier' = (the IUT's Device object),
 'Property Identifier' = Local Date,
 'Property Value' = (the current local date, D)
 15. TRANSMIT ReadProperty-Request,
 'Object Identifier' = (the IUT's Device object),
 'Property Identifier' = Local_Time
 16. RECEIVE ReadProperty-ACK,
 'Object Identifier' = (the IUT's Device object),
 'Property Identifier' = Local_Time,
 'Property Value' = (the current local time, T_{LOC})
 17. TRANSMIT ReadProperty-Request,
 'Object Identifier' = (the object being tested),
 'Property Identifier' = Time_Of_Active_Time_Reset
 18. RECEIVE ReadProperty-ACK,
 'Object Identifier' = (the object being tested),
 'Property Identifier' = Present_Value,
 'Property Value' = (a date and time such that the date = D and the time is approximately T_{LOC})

The check in step 10 is not correct. For one it does take into account only one processing failtime; it should be two failtimes due to the verify in step 7 and the explicit wait in step 8. Then due to runtime behaviour of the WriteProperty and ReadProperty Requests especially when the test is performed with a router between IUT and TD the test can fail even when the IUT is performing correctly. The test should be changed to allow more tolerance in the time-check.

Question:

Will the test be corrected by applying another failtime and more tolerance in step 10?

Response:

No. The test will be changed to:

Swap step 11 to precede step 9. Then check that the 'Property Value' is less than or equal to Telapse plus the amount of time that has passed between beginning of step 7 and the read of Elapsed_Active_Time in what is now step 11, and that it is greater than or equal to Telapse + 30.