

ISO 16845-1 2016

Document review

Ondrej Ille

ondrej.ille@gmail.com

Introduction

The aim of this document is to summarize issues found during in depth review of ISO16845-1 2016. Following types of issues are described in this document:

Typos – Not critical, however, they should not be present in official ISO documents.

Ambiguities – Missing information, parameters which may cause that document will be understood ambiguously during implementation of test sequences.

Semantic errors – Errors in description of test sequence which may cause verification of wrong functionality.

Errors in this document are ordered by test number. *Quotations of [1] are written with blue italic.*

The information provided in this document were gathered based on interest in CAN technology with no guarantee provided that observations in this document are correct, giving no warranty or liability for any possible usage of information shown in this document.

I declare that I provide this document in a good belief that it will improve future versions of ISO16845-1 2016 with no demand for payment/fee/reward in exchange for providing this document.

List of Issues

7.1.7 Minimum time for bus idle after protocol exception handling

Category: Ambiguity

Clearly, the purpose of this test is to verify that IUT is able to receive frames after detecting protocol exception. According to [1], protocol exception state shall not be exited by any implementation before detecting Idle condition (11 consecutive recessive bits).

To verify such time, LT must send such minimal sequence of bits to DUT, and verify that DUT leaves Protocol exception state (by checking proper reception of consecutive frame).

Elementary tests:

#1 The second frame starts after the third intermission bit + 1 bit time after the first frame.

#2 The second frame starts after the third intermission bit of the first frame.

#3 The second frame starts after the second intermission bit of the first frame followed by a third frame starts after the third intermission bit of the previous frame

Execution:

The LT send a frame with non-nominal bit in control field causing protocol exception behavior. The LT send a valid classical frame according to elementary test cases.

Test description is an missing important fact that LT must also send ACK bit low (even if it transmits a frame!).

If ACK bit is sent Dominant, duration of recessive sequence is following:

ACK Delimiter + End of frame + Intermission (3/3/2 bits for each elementary test) + Additional bit (only in #1).

The duration of Recessive sequence is: #1 = 13 bits, #2 = 12 bits, #3 = 11 bits. Therefore elementary test 3 verifies shortest possible recessive sequence needed to come out of protocol exception state.

If ACK bit is sent Recessive, duration of Recessive sequence includes ACK bit, CRC delimiter, and possibly last bits of CRC which could be Recessive. We see that without transmitting Dominant ACK by LT, leaving Protocol exception state of DUT is not properly verified. If e.g. DUT will leave protocol exception state after 13 recessive bits (which contradicts 10.9.4 of [1]), all elementary tests will pass and DUT with error will pass the test.

Solution: Clearly state that ACK bit of first frame in each frame shall be sent Dominant by LT.

Example: The LT send a frame with non-nominal bit in control field causing protocol exception behaviour. ACK bit of this frame shall be sent dominant.

7.2.8 Form error at fixed stuff bit in FD frames

Category: Semantic error / Ambiguity

There are 22 elementary tests to perform.

Tests to perform on recessive stuff bits:

#1 $DLC \leq 10 \rightarrow$ CRC (17) field – (6 bits)

#2 $DLC > 10 \rightarrow$ CRC (21) field – (7 bits)

Tests to perform on dominant stuff bits:

#3 $DLC \leq 10 \rightarrow$ CRC (17) field – (6 bits)

#4 DLC > 10 - > CRC (21) field - (7 bits)

Test states that there shall be 22 elementary tests. However, there are $6+7+6+7=26$ stuff bits to corrupt. It is not clear which stuff bits shall be tested and which not.

Solution: Change number of Elementary tests to 26, therefore it is clear that each possible combination of Dominant/Recessive stuff bit at each position shall be tested.

**7.8.3.1 Synchronization when $e > 0$ and $e \leq SJW(D)$ at ESI bit position –
7.8.6.3 Synchronization when $e < 0$ and $|e| > SJW(N)$ at “ACK” bit position**

Category: Ambiguity

All tests in this range clearly intend to verify various synchronization mechanisms in Data bit-rate. This is obvious from phase error calculations which use Data-bit rate bit timings (e.g. $SJW(D)$). Non-of the tests however, requires $BRS = 1$. In such case, if frames are generated by LT pseudo-randomly (as it is common in modern digital verification environments), it may happen that LT will send frames which do not switch bit-rate, and therefore not verify the functionality it intended to verify.

Solution: Clearly state in Test variables that $BRS = 1$ for all of these tests.

7.8.4.1 Synchronization when $e > 0$ and $e > SJW(D)$ at ESI bit position

Category: Typo

It is believed that “{“ at the end of elementary test description is a typo:

$e \in \{[SJW(D) + 1], [NTQ(D) - Phase_Seg2(D) - 1]\}$

Instead “}“ shall be used.

7.8.8.2 Glitch filtering test on negative phase error at “DATA” field

Category: Error

Description of Test states:

Elementary tests:

There is one elementary test to perform for at least 1 bit rate configuration.

#1 The LT reduce the length of a DATA bit by one $TQ(D)$ and the LT force the second TQ of this dominant stuff bit to recessive.

Execution:

The LT sends a frame according to elementary test cases.

Additionally, the Phase_Seg2(D) of this dominant stuff bit shall be forced to recessive.

The bit shall be sampled as dominant.

Clearly, the intent of this test is to verify, that there is only one synchronization between two data bits in data bit-rate. If a bit before dominant stuff bit is shortened by 1 TQ and then second TQ of Dominant stuff bit is forced to recessive, then there shall be only negative re-synchronization with Phase Error = -1. Glitch on second Time Quanta of dominant stuff bit shall be ignored. If it is not ignored by IUT, then additional Positive Re-synchronization will shift IUTs sample point to PH2 part of Dominant bit which is forced to recessive. This will cause stuff error.

In summary: If IUT re-synchronizes twice, it shall detect stuff error and send error frame. No error frames shall be sent in this test, therefore error frame means a bug in IUT.

However, the formulation of description is not correct. The formulation seems to describe that the same bit whose Ph2 is forced to recessive (dominant stuff bit), shall be also shortened by 1 TQ. Modifying the bit in such way means, that there are not two synchronisation edges between sample points of these two bits, therefore the scenario is not properly tested.

In context of 7.8.8.1 (test principle is the same), I believe that formulation of this test is wrong.

Solution: State that bit before stuff bit shall be shortened by 1 TQ.

Example: #1 The LT reduces length of recessive data bit just before dominant stuff bit by 1 TQ. Then LT forces second TQ of this Dominant stuff to recessive.

8.5.11 Bus-off recovery

Category: Semantic error

Test description states:

The IUT shall not transmit the frame before the end of the profiles sent by the LT according to elementary test cases and shall send it before the end of the TIMEOUT.

According to [1] IUT can have ability to re-integrate back after becoming Bus-off if it receives restart request from upper layer. It also states, that if it re-integrates it shall be no sooner than 128 times idle condition. However, it does not specify any upper limit on the time after it re-integrates back (only lower limit), therefore it should not be strictly obligatory to require that IUT will retransmit the frame till some TIMEOUT.

Also, if IUT goes bus-off, then restart request must usually be issued by upper layer. If such request is not given by SW/LT, test will fail, despite IUT being compliant to [1]. This is also not described in the test.

Solution 1: Remove “*and shall send it before the end of the TIMEOUT.*” part of the sentence as IUT is not obligatory to retransmit frame after reintegration (in fact, it might be intentional to require interaction with upper layer in case of going bus off).

Solution 2: If TIMEOUT is kept, then add clause indicating that LT shall issue restart request when IUT goes bus-off, since IUT is not obligated to start reintegrating automatically.

8.6.5 TEC increment when overload flag is followed by dominant bits

Category: Typo

Test description states:

*The IUT's TEC value shall be increased by 8 on each eighth dominant bit after the **error** Flag.*

Since this test invokes overload frame transmission, the above description is clearly wrong.

Solution: Correct the response to “overload”.

Example: *The IUT's TEC value shall be increased by 8 on each eighth dominant bit after the **overload** Flag.*

8.6.19 TEC non-increment on stuff error during arbitration

Category: Ambiguity

Test description says:

Response:

The IUT's TEC value shall be equal to the set-up value.

This is strictly speaking correct. Since IUT detects stuff error in arbitration field, it shall not increment its TEC. However, IUT will most likely retransmit the frame (if not in one shot mode). In such case, TEC will eventually decrement by 1 after successful retransmission. Therefore it might be good to clarify that TEC shall be checked after error frame transmission and before IUTs retransmission.

Solution: Clarify when should TEC be checked.

Example: *IUT's TEC value shall be equal to set-up value after transmission of Error frame. If IUT retransmits the frame, its TEC will eventually be decremented by 1 upon successful retransmission.*

8.6.21 TEC non-increment after arbitration lost and error

Category: Semantic Error

Test description says:

Execution:

The LT causes the IUT to transmit a frame.

The LT sends a frame with higher ID priority to cause the IUT to lose arbitration according to elementary test cases.

The LT receives the repeated frame without error.

Response:

The IUT's TEC value shall be unchanged equal to setup value.

In this test, IUT retransmits frame after it lost arbitration and LT receives this frame. Due to this fact, IUT will lower its TEC by 1, therefore final value of TEC will be equal to setup value minus one.

Solution: Correct the expected response.

Example: *The IUT's TEC value shall be unchanged, equal to setup value minus one.*

8.7.9 Synchronization after sample point while sending a dominant bit

Category: Semantic Error

Test description states:

Response:

The IUT sends an error flag and the next edge sent by the IUT occurs 6 bit times + $[Phase_Seg2(N) - SJW(N)]$ after the recessive to dominant edge applied by the LT after the sample point of the dominant bit.

In this test, IUT will execute re-synchronization with negative phase error 1 TQ after sample point (at TQ where there is dominant to recessive edge). This re-synchronization will be limited by SJW. Therefore, overall length of PH2 will be: $PH2(N) - SJW(N)$. However, the edge occurs 1 TQ after sample point, so the next edge (after the end of Error flag) shall come from IUT at:

$(PH2 - SJW - 1)$ TQs + 6 Bit Times

after the Recessive to Dominant edge by LT. Note the “-1”.

Solution: Fix the expected time when next edge shall arrive

Example: *The IUT sends an error flag and the next edge sent by the IUT occurs 6 bit times +*

[Phase_Seg2(N) – SJW(N) - 1] after the recessive to dominant edge applied by the LT after the sample point of the dominant bit.

8.8.4.2 Glitch filtering test on negative phase error within FD frames at “DATA” field position

Category: Semantic Error

Execution:

The LT causes the IUT to transmit a frame. The LT forces the last TQ of Phase_Seg2(D) of a recessive bit to dominant. The LT forces a following recessive bit to dominant from sync-segment up to Sync_Seg(D) + Prop_Seg(D) + Phase_Seg1(D) – 1TQ(D).

The aim of this test is to verify that if there are two synchronization edges within one bit time, only first one is used for synchronization, and the second one is ignored. If Last TQ of a recessive bit is forced to dominant, and next bit is forced to Dominant from SYNC, then forcing of these two bits will create one continuous sequence of dominant value. Therefore, there will not be two re-synchronization edges and this test will not verify what it intends to verify. To achieve the intended behavior, second bit must be forced from first TQ of PROP segment. Then last TQ of first bit will be dominant, SYNC will of second bit will be recessive and PROP till 1TQ before SP of second bit will be dominant, giving two Dominant to Recessive re-synchronization edges.

Solution: Correct the duration of force in second bit.

Example: *The LT forces a following recessive bit to dominant from **PROP**-segment up to Sync_Seg(D) + Prop_Seg(D) + Phase_Seg1(D) – 1TQ(D).*

Revision history

Revision	Date	Author
0.1	23.4.2021	Ondrej Ille
0.2	24.6.2021	Ondrej Ille

References

[1] ISO 11898-1 2015 Road vehicles — Controller area network (CAN), Part 1: Data link layer and physical signalling