ISO 16845-2:2018 (E)

Road vehicles — Controller area network (CAN) conformance test plan — Part 2: High-speed medium access unit — Conformance test plan

Contents

Foreword

Introduction

1 Scope

2 Normative references

3 Terms and definitions

4 Symbols and abbreviated terms

5 Global overview

5.1 OSI conformance test method
5.2 General organization
5.3 Test case organization
5.3.1 Overview
5.3.2 Setup state
5.3.2.1 General
5.3.2.2 Default setup
5.3.2.3 Default state
5.3.3 Test state
5.3.4 Test frame definition for protocol related test cases
5.3.4.1 Elements of CAN test frames
5.3.4.2 IUT configuration and default parameters
5.3.4.3 Sync frame sequence
5.3.5 Hierarchical structure of tests
5.3.5.1 Overview
5.3.5.2 Test group structure
5.3.5.3 Test case structure
5.3.6 Elementary tests
5.3.7 Applicable test cases for IUTs with enhanced voltage biasing

6 Test type 1, static test cases

7 Test type 2, normal CAN communication acceptance

7.1 Test class 1, valid frame format
7.1.1 ID test in CBFF messages
7.1.2 ID test in CEFF messages
7.1.3 WUP element of WUF
7.1.4 WUF element of another valid frame — CBFF message
7.1.5 WUF element of another valid frame — CEFF message
7.1.6 Acceptance of no nominal "SRR" in CEFF message
7.1.7 Absent bus idle after data frame
7.1.8 Stuff acceptance test 1
7.1.9 Stuff acceptance test 2
7.1.10 Acceptance of Sync Sequence
7.1.11 Idle detection after CAN FD frame (FD tolerant implementation only)
7.2 Test class 2, error detection
7.2.1 Stuff error test 1
7.2.2 Stuff error test 2
7.2.3 CRC error test
7.2.4 Form error in data frame

7.3 Test class 3, error frame management
7.3.1 Absent bus idle after error scenario
7.3.2 Active error condition during ignored frames after switching on the bias
7.3.3 Passive error condition during ignored frames after switching on the bias
7.4 Test class 4, CAN bit decoding
7.4.1 Correct sampling of the 10th bit after the last dominant edge causing resync
7.4.2 Correct sampling of the 10th bit after the last dominant edge after hard sync
7.4.3 IUT robustness against dominant bit extensions
7.4.4 IUT robustness against dominant bit shortening
7.4.5 Correct sampling after bit deformation and hard sync
7.4.6 No frame constant bit deformation due to loss of arbitration or ringing effects
7.4.7 Glitch filtering test in idle state
7.4.8 Glitch filtering test after FD format frame after IFS and EOF (FD tolerant implementation only)
7.4.9 Glitch filtering test in CAN FD data phase (FD tolerant implementation only)
7.4.10 Bit (glitch) detection test in CAN FD data phase (FD tolerant implementation only)
7.4.11 Clock tolerance test
7.4.12 Not constant network timing due to loss of arbitration

8 Test type 3, WUF evaluation

8.1 Test class 1, CAN message ID filter test
8.1.1 Message filter / CBFF – test 1
8.1.2 Message filter / CBFF – test 2
8.1.3 Message filter / CBFF – test 3
8.1.4 Message filter / CBFF – test 4
8.1.5 Message filter / CEFF – test 1
8.1.6 Message filter / CEFF – test 2
8.1.7 Message filter / CEFF – test 3
8.1.8 Message filter / CEFF – test 4
8.2 Test class 2, CAN message data filter test
8.2.1 Message data filter – matching data field
8.3 Test class 3, CAN message DLC filter tests
8.3.1 Message DLC filter test
8.4 Test class 4, optional data mask bit tests
8.4.1 Message filter / CBFF – test 1 while DLC matching condition disabled
8.4.2 Message filter / CBFF – test 2 while DLC matching condition disabled
8.4.3 Message filter / CBFF – test 3 while DLC matching condition disabled
8.4.4 Message filter / CBFF – test 4 while DLC matching condition disabled
8.4.5 Message filter / CEFF – test 1 while DLC matching condition disabled
8.4.6 Message filter / CEFF – test 2 while DLC matching condition disabled
8.4.7 Message filter / CEFF – test 3 while DLC matching condition disabled
8.4.8 Message filter / CEFF – test 4 while DLC matching condition disabled
8.4.9 Acceptance of frames independent of the DLC while DLC matching condition disabled
8.4.10 Acceptance of remote frames independent of the DLC while DLC matching condition disabled
8.5 Test class 5, non-acceptance of remote frames
8.5.1 Non-acceptance of remote frames

9 Test type 4, FEC management

9.1 General
9.2 Test class 1, valid frame format
9.2.1 FEC decrement on valid frame presence
9.2.2 FEC no increment on form error in error delimiter
9.2.3 FEC no increment on sixth bit of error delimiter
9.2.4 FEC no increment on ACK error
9.2.5 FEC no increment on form error in ACK delimiter
9.2.6 FEC no increment on form error in EOF field
9.2.7 FEC no increment on glitches
9.2.8 FEC no increment on classical CAN frames with not nominal "FDF, r0"
9.2.9 FEC no increment on CAN FD frames (FD tolerant implementation only)
9.3 Test class 2, error detection
9.3.1 FEC increment on form error in CRC delimiter
9.3.2 FEC increment on stuff error
9.3.3 FEC increment on CRC error

9.3.4 FEC incremented once when active error flag length is 13 bit
9.3.5 FEC incremented once when active error flag is longer than 13 bit
9.4 Test class 3, HS-PMA handling
9.4.1 FEC reset after expiration of tSILENCE
9.4.2 FEC reset on enabling selective wake-up function
9.4.3 FEC no reset during change from normal to low-power mode (optional)
9.4.4 FEC evaluation direct after WUP presence

10 Test type 5, HS-PMA implementation

10.1 Test class 1, WUP
10.1.1 Wake-up after valid WUP
10.1.2 No wake-up after invalid WUP
10.1.3 No wake-up after expiration of optional timer tWake
10.1.4 Reset of the optional timer tWake
10.1.5 No wake-up due to not stabilized recessive bus state
10.2 Test class 2, low-power mode operation
10.2.1 Reset of the timer tSILENCE
10.2.2 Expiration of the timer tSILENCE AND implementation in low-power mode
10.2.3 Biasing independency from VCC availability
10.2.4 Transmitter in low-power mode
10.2.5 Wake-up independency from VCC availability

Page count: 87