TEST SPECIFICATION

FIELDCOMM GROUP™

Connecting the World of Process Automation

HART® COMMUNICATION PROTOCOL

Slave Token-Passing Data Link Layer Test Specification

HCF_TEST-001, FCG TT20001
Revision 5.0

Release Date: 03 February 2020

Release Date: 03 February 2020

Document Distribution / Maintenance Control / Document Approval
To obtain information concerning document distribution control, maintenance control and document approval, please contact the FieldComm Group at the address shown below.

Copyright © 1995, 1997, 1999, 2001, 2004, 2009, 2018, 2020 FieldComm Group
This document contains copyrighted material and may not be reproduced in any fashion without the written permission of the FieldComm Group.

Trademark Information
HART® and WirelessHART® are registered trademarks of the FieldComm Group, Austin, Texas, USA. Any use of the term HART or WirelessHART hereafter in this document, or in any document referenced by this document, implies the registered trademark. All other trademarks used in this or referenced documents are trademarks of their respective companies. For more information contact the FieldComm Group Staff at the address below.

FIELDCOMM GROUP™
Connecting the World of Process Automation

FieldComm Group
9430 Research Boulevard
Suite 1-120
Austin, TX 78759, USA

Voice: +1 512-792-2300
FAX: 1-512-792-2310

http://www.fieldcommgroup.org

Use of imperatives in HART Specifications
The key words (imperatives) "must", "required", "shall", "should", "recommended", "may", and "optional" when used in this document are to be interpreted as follows:

Must	**Must**, **Shall**, or **Required** denotes an absolute mandatory requirement. For example, "All HART Field Devices must implement all Universal Commands"
Should	**Should** or **Recommended** indicates a requirement that, given good cause/reason, can be ignored. However, the consequences of ignoring the requirement must be fully understood and well justified before doing so.
May	**May** or **Optional** identifies a requirement that is completely optional and can be supported at the discretion of the implementation. May can be used to identify optional Host Application or Master functionality and, when this is the case, does not imply the function is optional in Field Devices.

Intellectual Property Rights
The FieldComm Group does not knowingly use or incorporate any information or data into the HART Specifications which the FieldComm Group does not own or have lawful rights to use. Should the FieldComm Group receive any notification regarding the existence of any conflicting Private IPR, the FieldComm Group will review the disclosure and either (a) determine there is no conflict; (b) resolve the conflict with the IPR owner; or (c) modify this specification to remove the conflicting requirement. In no case does the FieldComm Group encourage implementers to infringe on any individual's or organization's IPR.

Table of Contents

Preface 6
Introduction 7
1. Scope 9
1.1 Features Tested 9
1.2 Features Not Tested 9
2. References 10
2.1 The HART Communication Protocol Specifications 10
2.2 Other FieldComm Group Documents 10
2.3 Related Documents 10
3. Definitions 11
4. Symbols/Abbreviations 12
5. Approach 12
5.1 Testing Sequence 12
5.2 Test Classification 13
5.3 Conventions 16
6. Test Definitions 18
6.1 DLL001 FSK Preamble Check 18
6.2 DLL002 Delimiter Check 22
6.3 DLL003 Frame Expansion Check 25
6.4 DLL004 Short Frame Check 26
6.5 DLL005 Master Address Bit Check 27
6.6 DLL006 Burst Mode Bit Check 28
6.7 DLL007 Long Frame Address Check 29
6.8 DLL008 (Reserved) 30
6.9 DLL009 Incorrect Byte Count Check 31
6.10 DLL010 Vertical Parity Check 32
6.11 DLL011 Framing Error Check 34
6.12 DLL012 Check Byte Test 35
6.13 DLL013 FSK Gap Receive Timeout Test 36
6.14 DLL014 Long Message Test 38
6.15 DLL015 Start Of Message In Data Field Check 39
6.16 DLL016 Preamble Check For BACK Frames 40
6.17 DLL017 Preamble Check For ACK Frames 41
6.18 DLL018 Gap Errors in ACK Frames Check 42
6.19 DLL019 Gap Check For BACK Frames 44
6.20 DLL020 Dribble Byte Check For ACK Frames 45
6.21 DLL021 Dribble Byte Test For BACK Frames 46
6.22 DLL022 Test Host Address Bit For BACK Frames 47
6.23 DLL023 Test Burst Mode Bit Of Burst-Mode Slave Frames 49
6.24 DLL024 Test Slave Responds Within STO 50

6.25 DLL025 Burst Hold During Master Preamble 52
6.26 DLL026 Test Burst Response Time After a DUT ACK 54
6.27 DLL027 Test Response Time Between Consecutive Bursts 56
6.28 DLL028 BACK Timing with STXs Errors 57
6.29 DLL029 Burst Mode Timeout On Other Slave 59
6.30 DLL030 Burst After Response From Other Slave 61
6.31 DLL031 (Reserved) 62
6.32 DLL032 Read Unique Identifier (Command 0) 63
6.33 DLL033 Write Polling Address (Command 6) 65
6.34 DLL035 Write Number Of Response Preambles (Command 59) 73
6.35 DLL039 Slave Time-Out Stress Test 76
6.36 DLL040 Unique Address Test 79
6.37 DLL041 Framing Successive Messages 81
6.38 DLL042 Command Number Expansion 82

ANNEX A. Common Test Macros and Definitions 83
A1 CheckBurstCommands ( ) 83
A2 CheckDeviceAlive ( ) 84
A3 CheckReadyForBurst ( ) 85
A4 CheckSlaveSTO (command) 86
A5 CyclePower ( ) 87
A6 FindNextDeviceVariable (dVar) 88
A7 IdentifyDevice ( ) 90
A8 TestValidFrame ( ) 90
A9 VerifyNotWriteProtected ( ) 91
A10 VerifyResponseAndByteCount (rc, bc) 91

ANNEX B. Failure Point Cross Reference 92

ANNEX C. Revision History 94
C1 Changes from Revision 4.0 to 5.0 94
C2 Changes from Revision 3.1 to 4.0 94
C3 Changes from Revision 3.0 to 3.1 94
C4 Changes from Revision 2.1 to 3.0 95
C5 Changes from Revision 2.0 to 2.1 96
C6 Changes from Revision 1.2 to 2.0 97
C7 Changes from Revision 1.1 to 1.2 97
C8 Changes from Revision 1.0 to 1.1 98

Index to Tables

Table 1 Test Execution Sequence 12
Table 2 Frame Detection And Recognition Tests 13
Table 3 Frame Generation Tests 13
Table 4 Basic Bus Arbitration Tests 14
Table 5 Data Link Layer Services Tests 14
Table 6 Burst Mode Frame Generation 15
Table 7 Burst Mode Bus Arbitration Tests 15
Table 8 Common Response Codes 16
Table 9 Preamble Test Case Summary 18
Table 10 Frame Expansion Test Vectors 25
Table 11 STX's for Receive Gap Tests 36
Table 12 Embedded Message Test Cases 39
Table 13 Non-Existent Field Device Test Vectors 59
Table 14 Command 0 Pass / Fail Criteria 63
Table 15 Burst Mode Command Requirements 83