



FIELD COMM GROUP™

*Connecting the World of
Process Automation*



COMMUNICATION PROTOCOL

**Token-Passing Data Link
Layer Test Specification**

**HCF_TEST-001, FCG TT20001
Revision 6.0**

Release Date: 20 November 2024

Release Date: 20 November 2024

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, 2023, 2024 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. Approach | 10 |
| 2.1 Testing Sequence | 10 |
| 2.2 Test Classification | 11 |
| 2.3 Conventions | 13 |
| 3. Test Definitions | 15 |
| 3.1 DLL001 FSK Preamble Check | 15 |
| 3.2 DLL002 Delimiter Check | 19 |
| 3.3 DLL003 Frame Expansion Check | 22 |
| 3.4 DLL004 Short Frame Check | 23 |
| 3.5 DLL005 Master Address Bit Check | 24 |
| 3.6 DLL006 Burst Mode Bit Check | 25 |
| 3.7 DLL007 Long Frame Address Check | 26 |
| 3.8 DLL009 Incorrect Byte Count Check | 27 |
| 3.9 DLL010 Vertical Parity Check | 28 |
| 3.10 DLL011 Framing Error Check | 30 |
| 3.11 DLL012 Check Byte Test | 31 |
| 3.12 DLL013 FSK Gap Receive Timeout Test | 32 |
| 3.13 DLL014 Long Message Test | 34 |
| 3.14 DLL015 Start Of Message In Data Field Check | 35 |
| 3.15 DLL016 Preamble Check For BACK Frames | 36 |
| 3.16 DLL017 Preamble Check For ACK Frames | 37 |
| 3.17 DLL018 Gap Errors in ACK Frames Check | 38 |
| 3.18 DLL019 Gap Check For BACK Frames | 40 |
| 3.19 DLL020 Dribble Byte Check For ACK Frames | 41 |
| 3.20 DLL021 Dribble Byte Test For BACK Frames | 42 |
| 3.21 DLL022 Test Host Address Bit For BACK Frames | 43 |
| 3.22 DLL023 Test Burst Mode Bit Of Burst-Mode Slave Frames | 47 |
| 3.23 DLL024 DUT Responds Within STO | 48 |
| 3.24 DLL025 Burst Hold During Master Preamble | 50 |
| 3.25 DLL026 Test Burst Response Time After a DUT ACK | 52 |
| 3.26 DLL027 Test Response Time Between Consecutive Bursts | 54 |
| 3.27 DLL028 BACK Timing with STXs Errors | 55 |
| 3.28 DLL029 Burst Mode Timeout On Other Slave | 57 |
| 3.29 DLL030 Burst After Response From Other Slave | 59 |
| 3.30 DLL032 Read Unique Identifier (Command 0) | 60 |
| 3.31 DLL035 Write Number Of Response Preambles (Command 59) | 62 |

| | | |
|----------|--|----|
| 3.32 | DLL039 DUT Time-Out Stress Test | 65 |
| 3.33 | DLL040 Unique Address Test | 68 |
| 3.34 | DLL041 Framing Successive Messages | 70 |
| 3.35 | DLL042 Command Number Expansion | 71 |
| ANNEX A. | Common Test Macros and Definitions | 72 |
| A1 | CheckBurstCommands () | 72 |
| A2 | CheckDeviceAlive () | 73 |
| A3 | CheckReadyForBurst () | 73 |
| A4 | CheckSlaveSTO (command) | 74 |
| A5 | CyclePower () | 75 |
| A6 | FindNextDeviceVariable (dVar) | 76 |
| A7 | IdentifyDevice () | 78 |
| A8 | TestValidFrame () | 78 |
| A9 | VerifyNotWriteProtected () | 79 |
| A10 | VerifyResponseAndByteCount (rc, bc) | 79 |
| ANNEX B. | Failure Point Cross Reference | 80 |
| ANNEX C. | References | 82 |
| C1 | The HART Communication Protocol Specifications | 82 |
| C2 | Other FieldComm Group Documents | 82 |
| C3 | Related Documents | 82 |
| ANNEX D. | Definitions | 83 |
| ANNEX E. | Symbols/Abbreviations | 84 |
| ANNEX F. | Revision History | 85 |
| F1 | Revision 6.0 | 85 |
| F2 | Revision 5.0 | 85 |
| F3 | Revision 4.0 | 85 |
| F4 | Revision 3.1 | 86 |
| F5 | Revision 3.0 | 87 |
| F6 | Revision 2.1 | 87 |
| F7 | Revision 2.0 | 88 |
| F8 | Revision 1.2 | 88 |
| F9 | Revision 1.1 | 89 |