HART Field Communications Protocol Specification

HART®
COMMUNICATION FOUNDATION

Block Data Transfer Specification

HCF_SPEC-190, Revision 1.0

Release Date: 18 April, 2001

Release Date: 18 April, 2001

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

Attention: Foundation Director
HART Communication Foundation
9390 Research Boulevard
Suite I-350
Austin, TX 78759, USA

Voice: (512) 794-0369
FAX: (512) 794-3904

http://www.hartcomm.org

Table of Contents

Table of Figures

Index to Tables

Preface

This preface is included for informational purposes only.

It has long been the intention of the HART Protocol to support block data transfers. However, details for using block data transfer services have remained incomplete and the use of commands 111 and 112 has been prohibited. This specification completes the block transfer mechanism introduced with HART Rev. 5 in 1989.

Since the data transfer concept was first added to HART, field device capabilities have dramatically increased. Today there is a diverse range of HART instruments and there are many sophisticated multivariable devices. While features of individual products vary, many devices are capable of:

- Storing a log of process activities for later retrieval (data logging)
- Recording errors to allow easier diagnosis of plant problems, and
- Streaming measurement data to profile device performance.

Completing the data transfer commands adds significant, valuable capabilities that support today's increasingly sophisticated field devices.

Introduction

This introduction is included for informational purposes only.

Generally, the HART Protocol provides only acknowledged, connectionless service[1]. In effect all Protocol transaction are autonomous, stand-alone and provide a receipt back to the master confirming the transaction. This provides simple, low overhead communication well suited to the process industries. The *Block Data Transfer Specification* extends the Protocol to offer an (optional) connection oriented reliable stream service.

Traditionally the HART Protocol is considered to support only three layers of the ISO/OSI 7-layer model for communications protocols (see Figure 1). Despite this there are elements of the intervening 4 layers found in the Protocol Specifications. For example, network layer capabilities can be found in the *Command Summary Specification*.

	OSI Layer	Function	HART
7	Application	Provides the User with Network Capable Applications	Command Oriented. Predefined Data Types and Application Procedures
6	Presentation	Converts Application Data Between Network and Local Machine Formats	
5	Session	Connection Management Services for Applications	
4	Transport	Provides Network Independent, Transparent Message Transfer	
3	Network	End to End Routing of Packets. Resolving Network Addresses	
2	Data Link	Establishes Data Packet Structure, Framing, Error Detection, Bus Arbitration	A Binary, Byte Oriented, Token Passing, Master/ Slave Protocol.
1	Physical	Mechanical / Electrical Connection. Transmits Raw Bit Stream	Simultaneous Analog & Digital Signaling. Normal 4-20mA Copper Wiring

Figure 1. OSI 7-Layer Model

[1] Halsall, F. *Data Communications, Computer Networks and Open Systems*. Third Edition. Addison Wesley. 1992 pp 238-239

The Block Data Transfer mechanism is best classified as a Transport Layer service. The data transfer mechanism is like a pipeline. It establishes a "connection" between the host and the slave device and guarantees the transfer of a stream of data. The mechanism is designed to maximize the utilization of the HART communication bandwidth while performing the transfer.

To access this capability in a field device the host establishes a connection to a specific "Port" in the slave using Command 111. A host opens a Port (Command 111), creating a pipeline to the device, and transfers data to or from the field device (Command 112). Each port provides functionality that is fully defined in "Port Specification" sub-documents that are included with this specification. A list of Ports and their function are summarized in the Section 9.

The HART Protocol includes robust error checking, and the Data Link Layer ensures that a given command will be delivered to the slave device. A master receives a status word indicating whether the command was successful or not. Masters then perform retries to deliver commands when communications error occur. Since HART commands are normally stand alone, implementing the retry mechanism is relatively simple.

However, when transferring a block of data, retries become more complex. Since the data is transferred as a stream, the progress of the data transfer must be tracked. This specification addresses the more stringent requirements of block data transfer (see Table 1).

Table 1. Data Transfer Features

Requirement	Feature
1. Maximize Data Throughput	The maximum number bytes to be transferred in a transaction is negotiated when the Port is opened. The more data transferred each transaction the more efficient the block transfer becomes.
2. Slaves and Masters have differing communication buffer space	Data is transferred in blocks as large as both the master and slave support.
3. Transfer must be reliable. No data may be duplicated or lost.	Byte counters are used to track the data transferred. Acknowledgment indicates which byte count is next expected by the recipient.
4. Must be flexible to support differing application needs	Transfer is based on a virtual connection to a Port. Different Ports support different, well defined functions. A block of Ports are allocated for device specific requirements.
5. Transfer must be synchronous	Function codes are used to open, close, or reset a Port. The Port is not closed until the master and the slave both agree the transfer is complete.

1. SCOPE

The *Block Data Transfer Specification* defines a reliable, stream oriented, Transport Layer service for the HART Protocol. The operation of the block transfer is specified as well as the services supplied to the HART Application Layer. Block transfer is accomplished via a virtual connection to a "Port". Each Port provides specific functionality that defines the content of the data stream.

In addition to specifying the operation of the block transfer, the content and functionality of the Ports are specified. Each Port Specification is a separate sub-document that is incorporated (by reference) into this specification.

This specification contains the rules that must be followed when defining a new Port Specification and when performing modifications to this specification. Among these rules are requirements to publish proposed Port Specifications to allow review and comment by the HCF membership. These proposed Port Specifications vary in maturity from Concept to Final. As part of this Port Specification development process a proposed Port Specification cannot be included into the *Block Data Transfer Specification* until proven in device implementations. As a result, incorporation into HART compatible products should only occur after a "Final" Port Specification is published. Furthermore, the HCF staff must be notified of any products using the Port Specification prior to its inclusion into the *Block Data Transfer Specification*. See Section 9.3 for more information.

2. REFERENCES

2.1 The HART Field Communications Protocol Specifications

These documents published by the HART Communication Foundation are referenced throughout this specification:

HART Field Communication Protocol Specification. HCF_SPEC-12

Data Link Layer Specification . HCF_SPEC-81

Command Summary Specification. HCF_SPEC-99

2.2 Port Specifications

The Block Data Transfer Specification includes the set of documents that define the functions of the Ports used for data transfer. This version of the specification includes the documents identified in Table 2.

Table 2. Port Specification Documents

Document Title	Rev.	Number
No Port Specifications are included in this, initial revision of the Block Transfer Specification.	x.x	HCF_SPEC-190.x

2.3 Related Documents

The following text describes communication specification techniques used in this document including Service Access Points (SAPs) and time sequence diagrams.

Halsall, F. *Data Communications, Computer Networks and Open Systems*. Third Edition. Addison Wesley. 1992

3. DEFINITIONS

Terms used in this document and defined in *HART Field Communications Protocol Specification* include: Data Link Layer, Delayed Response, Delayed Response Mechanism, Busy, DR_CONFLICT, DR_DEAD, DR_INITIATE, DR_RUNNING, Master, Request Data Bytes, Response Data Bytes, Response Message, Slave.

Burst-Mode Device This is a device which provides a digital response carrying measurement or other data, at regular intervals, without the data being specifically requested, i.e. this device normally functions as an independently broadcasting device. A Burst-Mode Device is defined to be a Slave Device with burst mode capability (hence the use of the word "mode" in describing the device type). When such a mode is enabled, the Slave Device is said to "be in burst mode".

Device A Field Device or a Master Device.

Field Device A Slave or Burst-Mode Device.

Host One of (possibly) several applications that can be executed sequentially or simultaneously on a **Master.**

Master Device	A Master Device is responsible for initiating, controlling and terminating transactions with a slave device or a burst mode device. A distinction is made between master devices into Primary Masters and Secondary Masters in order to allow the simultaneous use of two Master Devices on a HART communication link. The same protocol rules are followed by a Primary Master and a Secondary Master except for customizing time-outs that differentiate between them.
Port Specification	The definition of the functionality available from a field device that supports a specific Block Transfer Port number.
Slave Device	This is a device which accepts or provides a digital message carrying measurements or other data, but only when specifically requested, i.e. this device always functions as a slave in a master/slave relationship.
Stream	A reliable, virtual connection between two Application Layer functions. A stream delivers exactly the same sequence of data bytes to the destination device that the sender communicates to the source device transport layer.
Time Sequence Diagram	A diagram used to illustrate the interrelationship between the Protocol services. The protocol layer of interest and the lower, intervening layers are treated as a "black box". The internal workings of these layers are not shown on this diagram. The time sequence diagram shows the interactions between the service primitives over time. Sometimes referred to as a Message Sequence Diagram.
Transport Layer	The Transport Layer provides reliable data transfer between two devices. Communication is transparent in that detailed low level knowledge of the communication is not required.

4. SYMBOLS/ABBREVIATIONS

HCF	**H**ART **C**ommunication **F**oundation
LSB	**L**east **S**ignificant **B**yte. The LSB is always the last byte transmitted over a HART data link.
MSB	**M**ost **S**ignificant **B**yte. The MSB is always the first byte transmitted over a HART data link.
PDU	**P**rotocol **D**ata **U**nit. This refers to the format and content of a message exchanged between corresponding layers in two field devices (e.g. the Data Link Layer in one device with the Data Link Layer in another device).
SAP	**S**ervice **A**ccess **P**oint. The services supplied by one communication protocol layer and used by another protocol layer. These encapsulate the functionality supported by a protocol layer.
WG	**W**orking **G**roup.

5. DATA FORMAT

In command specifications, the following key words are used to refer to the data formats. For more information about these formats, see the *Command Summary Specification*.

Bits	Each individual bit in the byte has a specific meaning. Only values specified by the command may be used. Bit 0 is the least significant bit.
Enum	An enumerated value. Only values specified in the *Common Tables Specification* may be used.
Float	An IEEE 754 single precision floating point number. The exponent is transmitted first followed by the most significant mantissa byte.
Packed	A string consisting of 6-bit alpha-numeric characters that are a subset of the ASCII character set. This allows four characters to be packed into three bytes. Packed ASCII strings are padded out with space (0x20) characters.
Unsigned-*nn*	An unsigned integer where *nn* indicates the number of bits in this integer. Multi-byte integers are transmitted MSB — LSB.

6. DATA TRANSFER SERVICES

In this section, Block Data Transfer is specified from a "black box" point of view. The requirements are specified by defining the required Service Access Points (SAPs) and providing a time sequence diagram[2] indicating the order in which the SAPs may be used. See the *Data Link Layer Specification* for more information on the service specification methodology.

Block data transfer SAPs are used to support applications (e.g., uploads/downloads) that require a reliable stream service between (master and slave) devices. Because of the master-slave relationship of the HART Data Link Layer, operation of the transfer service is initiated and primarily controlled by the master device. There are eleven (11) Service Access Points (SAPs) defined to support block transfer. They are segregated into 3 functional groups:

- Connection Establishment SAPs that are used to open the stream between devices;
- Data Transport SAPs that perform the actual data communication; and
- Connection Termination SAPs that are used to close or, in case of an error, abort a connection.

In the definition of the SAPs, certain parameters are optional and may not be present in all invocations. Such parameters are distinguished by enclosing them within square brackets ("[","]") in the primitive definitions. The sequence diagram describing these primitives is shown in Figure 2.

All SAPs described here must be supported by any device supporting Block Transfer unless otherwise stated. The mapping of these SAPs into an implementation is entirely a local matter and is in no way restricted by this specification.

6.1 Connection Establishment SAPs

`CONNECT.request (slave_address, port):` This service is used by a master to open a connection to the slave at the specified **slave_address**. The function requested by the master is indicated by the **port** (see Section 11.1).

`CONNECT.indicate (master_address, port):` This service indicates to the slave that a master is requesting a connection to the function indicated by the **port**.

`CONNECT.response (session_handle, port):` This service is used by the slave to indicate the results of a previous connection request via the `CONNECT.indicate`. This service is only used if the connection is successfully opened. Unsuccessful connection attempts result in a `DISCONNECT.request`. The **session_handle** is provided by the Application Layer to allow future data exchange via the established connection.

[2] Halsall, F. *Data Communications, Computer Networks and Open Systems*. Third Edition. Addison Wesley. 1992 pp 526-528

`CONNECT.confirm (session_handle):` This service is used by the master to indicate the successful result of a previous `CONNECT.request`. The **session_handle** is provided by the Transport Layer to allow future data exchange via the established connection.

Note: The coding of the session_handle is a local only reference shared between the local Application Layer and Transport Layer.

6.2 Data Transport SAPs

`TRANSFER.request(session_handle, [data]):` This SAP is used by device (master or slave) to request **data** transfer to the correspondent device. The Transport Layer is responsible for partitioning the data into segments, as needed, and communicating each segment as part of Command 111 transaction.

`TRANSFER.indicate(session_handle, [data]):` This SAP delivers **data** to a receiving device via the (previously opened) stream connection. The Transport Layer must insure that only error free data is delivered. The number of `TRANSFER.indicate` SAPs invoked as the result of a transfer.request is implementation specific.

`TRANSFER.confirm(session_handle):` This SAP is returned indicating the successful completion of the previous `TRANSFER.request`.

Note: A fatal Transport Layer error may result in a `DISCONNECT.indicate`. Disposition of the error free data delivered is not governed by this specification. See the definition of the port (function) for specific requirements.

6.3 Connection Termination SAPs

`DISCONNECT.request(session_handle, status):` This SAP is used by a device to initiate the termination of a connection. Termination may be due to normal completion of data transfer or an abort due to an error condition. The **status** provides additional information regarding the `DISCONNECT.request`.

`DISCONNECT.indicate(session_handle, status):` This SAP is returned, indicating that the connection indicated by the **session_handle** is being terminated. The **status** provides additional information regarding connection termination. This SAP may be returned in response to a unsuccessful `CONNECT.request`.

`DISCONNECT.response(session_handle, status):` A device Application Layer uses this SAP to acknowledge the termination of the connection indicated by the **session_handle**.

`DISCONNECT.confirm(session_handle):` This SAP is returned, indicating the successful completion of the previous `DISCONNECT.request`.

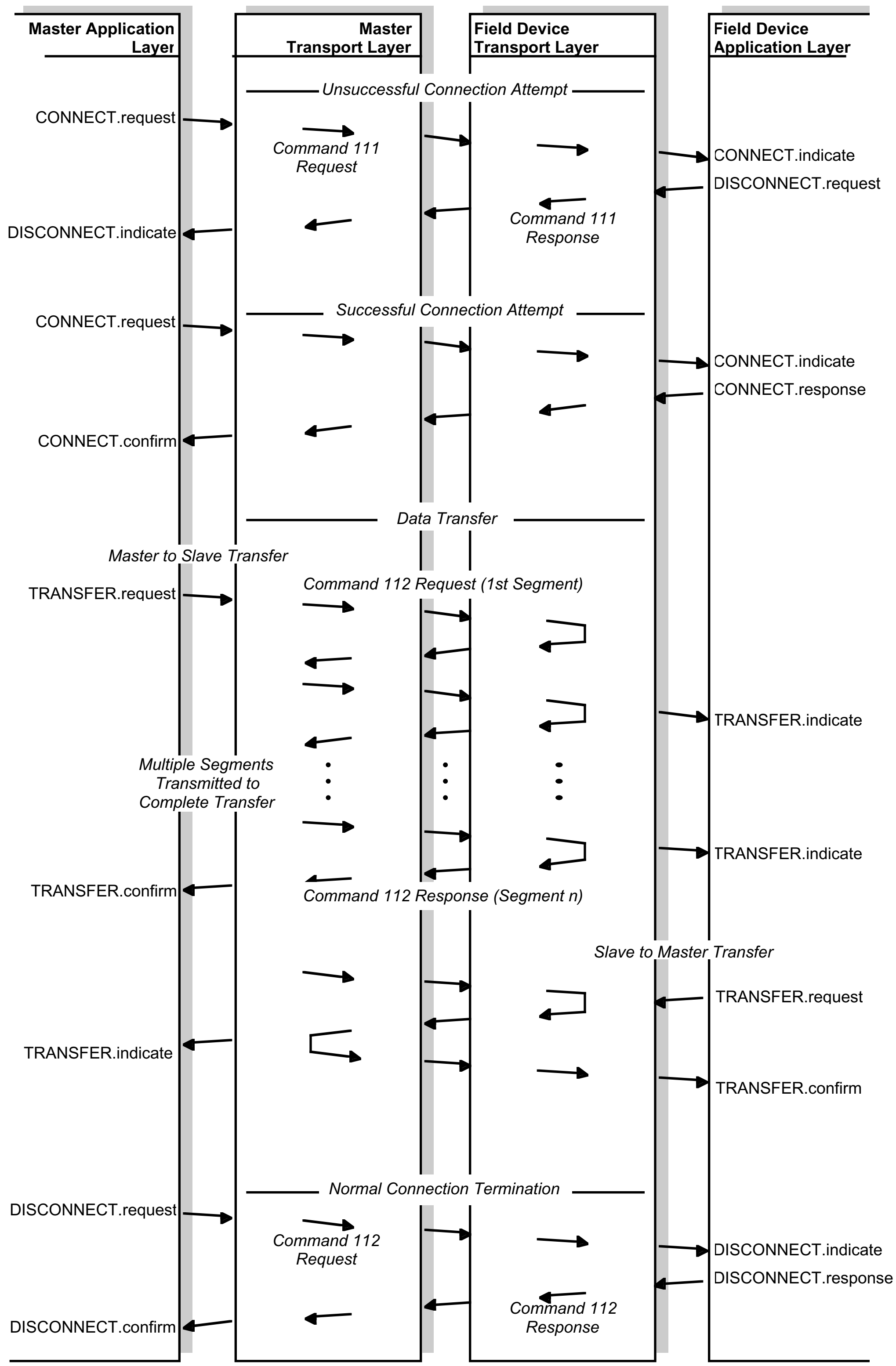

Figure 2. Transfer Service Sequence

7. DATA TRANSFER LIFE-CYCLE

While the previous section described block data transfer from a black-box perspective, this section specifies the life-cycle of a port connection and data transfer in detail.

Data transfer is accomplished by providing a reliable, stream-oriented data transfer path between devices. Block data is buffered and segmented to utilize the largest data packets supported by the intervening Data Link Layers. The life-cycle of a virtual connection between devices consist of three phases:

- **Opening the Connection:** In this phase the connection is negotiated (using Command 111) setting the maximum data that can be transferred in a single Command 112 transaction and, if successful, the connection opened. Depending on conditions and work load in the Field Device the connection may be refused.
- **Transferring the Data:** Once a connection is open, data transfer can begin. This consists of a series of Command 112 transactions. Each device is responsible for buffering the data and or breaking the stream into segments as needed to service the Application Layer needs.
- **Closing the Connection:** A connection can be closed normally, as indicated in the Port Specification, or the connection may be aborted. Generally, either device can initiate the closing of the port using the Function Code field in Command 112. The connection is not closed until both devices acknowledge its closure.

In effect, the previous section specified the interaction between the Application Layer and the Transport Layer, while this section defines the interaction between the (master and slave) device Transport Layer in terms of the actual command transactions employed.

7.1 Opening a Connection

Command 111 is used by a master to open a virtual connection to a Port located in the slave device. Opening the connection consists of the following steps:

- The master issues a Command 111 request including: the Port to be opened; the maximum amount of data the master can transfer per transaction; and the starting byte counter value.
- The slave's Command 111 response includes: the data from the master request; the slave's starting byte counter value; and response code information indicating the successful or failed execution of the master's request. In its response the slave may reduce the maximum data transfer size if necessary.

The starting byte counter is a semi-random number used to synchronize the start of the data stream. In other words, no device may depend on any specific value being returned by the other end of the connection as the starting byte counter. Once the stream is open, the byte counters track the progress of data transfers. They ensure, in case of communication errors or message retries, that no data is lost or duplicated.

7.2 Transferring Data

Command 112 is used to perform the actual data transfer. The amount of data transferred is indicated in the request and response. Each transaction includes:

- A Command 112 request from the master including: the master byte counter; the slave byte counter; the number of bytes transferred from the master to the slave; and the block data to be transferred. The master byte counter indicates the position in the stream of the first byte of master data included in this transaction. The slave byte counter indicates the position in the stream of the last data byte successfully received from the slave.
- The slave's reply is similar and includes: confirmation of the data received from the master; the starting position and quantity of slave data; the slave data. The master byte counter indicates the position in the stream of the last data byte successfully received from the master. The slave byte counter indicates the position in the stream of the first byte of slave data included in this transaction.

The byte counters are incremented by both devices as bytes are successfully transferred. This insures data is not lost or duplicated.

While a full duplex connection is supported, in practice, data transfer is predominantly either from the slave to the master or from the master to the slave.

7.3 Closing a Connection

Either device may initiate the termination of a connection. The connection may be terminated normally or due to a fatal error. The exact termination sequence depends on whether the slave or master initiates termination. The following sequence is used if the master initiates closure:

- The master includes the "Ready to Close" Function Code in its Command 111 request.
- If the slave has not completed data reception or transmission successfully, it may answer with the Function Code "Transfer Data". The master should complete the necessary retries before trying again to close the port.
- If the slave receives the data portion of Command 111 correctly, it responds with the Function Code "Port Closed".

This sequence allows for a retry on even the last block of data transferred. The slave transfer is similar. The slave issues the "Ready to Close" function code in its Command 111 response. The rest of the closure sequence is the same as when the master initiates termination.

In addition, Command 111 can be used to reset a Port if it is inadvertently left open or if there is fatal communication problems. For example, reset can be used when a master is physically disconnected from a field device while the port is still open.

8. BLOCK TRANSFER COMMANDS

8.1 Command 111 Transfer Service Control

This command opens virtual connect to the field device. The service to be provided is defined by the Port. See Section 11.2 a for definition of the Ports and the service they provide. When opening a Port the master indicates the maximum message size it supports. The slave can agree to this message length or decrease the message length. The starting point for the reference byte counters is established as well. The byte counters are not reset each time the port is opened.

This function can be used to reset a Port that is not functioning correctly.

Request Data Bytes

Byte	Format	Description
0	Enum	Port (See Section 11.2)
1	Enum	Function Code. Defines the action to be taken (See Section 11.1).
2	Unsigned-8	Maximum segment length supported by the master.
3-4	Unsigned-16	The starting master byte counter

Response Data Bytes

Byte	Format	Description
0	Enum	Port (See Section 11.2)
1	Enum	Function Code. Defines the action to be taken (See Section 11.1).
2	Unsigned-8	The maximum segment length supported by the slave device. The value returned by the slave cannot be larger then the corresponding value in the request.
3-4	Unsigned-16	The starting Master byte counter
5-6	Unsigned-16	The starting Field Device byte counter

Command-Specific Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1		Undefined
2	Error	Invalid Selection - Bad Function Code
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7		Undefined
8	Warning	Segment Length Changed
9	Error	Port not Found
10	Error	Port in Use.
11	Error	Maximum Ports In Use
12	Error	Segment Length Too Small
13 - 15		Undefined
16	Error	Access Restricted
17 -127		Undefined

8.2 Command 112 Block Transfer

This command passes data between a field device and a master via a port previously opened using Command 111. Data in the request is transferred to the slave. If received correctly, the slave increments its copy of the master byte counter by the number of bytes received. In its response, the slave indicates the master byte count of the next data byte it expects to receive, thus acknowledging successful data reception.

Data is transferred from the slave to the master in the response message using a slave byte counter to track data transfer progress. The number of bytes transferred each message can vary or even be zero providing the number of bytes does not exceed the maximum negotiated when the port is opened using Command 111.

When the data transfer is complete, the slave and master must agree to terminate the connection and close the Port.

Request Data Bytes

Byte	Format	Description
0	Enum	Port (See Section 11.2)
1	Enum	Function Code (See Section 11.1)
2	Unsigned-8	Number of bytes transferred in this message (i.e., *n*).
3-4	Unsigned-16	The master byte count corresponding to the first data byte in this message.
5-6	Unsigned-16	The slave byte count of the data bye the master expects to next receive from the slave device.
7-*n*	---	(Optional) Data to the Slave

Response Data Bytes

Byte	Format	Description
0	Enum	Port (See Section 11.2)
1	Enum	Function Code (See Section 11.1)
2	Unsigned-8	Number of bytes transferred in this message (i.e., *n*).
3-4	Unsigned-16	The Master byte count corresponding to the first data byte in this message.
5-6	Unsigned-16	The Field Device byte count of the data bye the master expects to next receive from the slave device.
7-*n*	---	(Optional) Data to the Master

Command-Specific Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1		Undefined
2	Error	Invalid Selection: Bad Function Code
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7-8		Undefined
9	Error	Port not Open
10 - 15		Undefined
16	Error	Access Restricted
17-31		Undefined
32	Error	Busy (A DR Could Not Be Started)
33	Error	DR Initiated
34	Error	DR Running
35	Error	DR Dead
36	Error	DR Conflict
37 - 127		Undefined

9. SPECIFICATION CONTROL

The *Block Data Transfer Specification* is designed to allow the addition of new Port Specifications . Since these Port Specifications may have a narrow scope that is not applicable or of interest to all devices, change control procedures used by other specifications do not apply to the Port Specifications. In other words, balloting of a new or revised Port Specification is not practical.

This section defines procedures and requirements that govern the addition of an individual Port Specification to this specification, and the modification of an approved Port Specification. These requirements are designed to:

- Ensure the technical excellence of each Port Specification,
- Provide a fair, open and objective basis for the development of Port Specifications,
- Alignment of the Port Specifications with Protocol requirements and practices, and
- Allow access and participation by all interested parties.

Any changes to this specification other then additions or changes to a specific Port Specification must follow the normal change procedures as defined in *HART Smart Communications Protocol Specification*.

The initial release of the *Block Data Transfer Specification* must be balloted and follow the normal process for HART Specifications as defined in *HART Smart Communications Protocol Specification*.

Note: These procedures intentionally and explicitly do not establish a fixed maximum time period that shall be considered "reasonable". A premium is placed on consensus and efforts to achieve it. These procedures deliberately defer the swift execution in favor of providing the flexibility to reach a genuine technical consensus.

9.1 Revision Numbers

As per normal Protocol requirements The *Block Data Transfer Specification* has a major and minor revision number. As a specification is changed, the major and minor revision numbers are incremented as follows:

- The addition of a new Port Specification shall be considered a functional change to this Specification. The major revision number must be incremented and the minor revision number reset to zero.
- The modification of a Port Specification shall be considered a non-functional change to this specification and the minor revision number must be incremented.
- Any other change to this specification must follow the revision numbering and change control procedures found in the *HART Smart Communications Protocol Specification*.

9.2 Port Specification Maturity Levels

Proposed Port Specifications evolve through a series of maturity levels before incorporation into this specification. These levels allow for the fair, open, and orderly development of a Port Specification . In addition, technical excellence increases at each level, ensuring the integrity of the *Block Data Transfer Specification*.

Table 3. Port Specification Maturity Levels

Conceptual	Proposals for the addition of a new Port Specification may be submitted by any HCF member company or the HCF staff. A Port Specification Concept must define, at least, the scope and provide a brief description of the Port Specification . Once the concept is established, an HCF technical working group must be formed to produce a draft Port Specification.
Draft	A Draft Port Specification is a complete specification meeting all the appropriate Protocol requirements. A Draft Port Specification should have all design and technical decisions resolved. Significant review by interested parties should be complete and a strong consensus formed.
Preliminary	A Preliminary Port Specification is a Draft that has been selected for promotion by the Port Specification Committee. Once a Port Specification reaches Preliminary, a final request for comments is solicited from the HCF membership.
Final	If no significant controversy arises from the final comments, a Port Specification may be promoted to Final by the Port Specification Committee. Once implemented successfully in real devices, the Final Port Specification will be incorporated into the next revision of the *Block Data Transfer Specification*.

The proposed Port Specification is given a document number in the form HCF_SPEC-190.x, where x is proposed Port number. Document revision numbering shall use the form 'y.z', where 'y' indicates a major revision number and 'z' a minor revision. The initial value for any major revision (i.e., new specification document) is 1. For any major revision, the minor revision number is reset to 0.

As a Port Specification is changed, the major and minor revision numbers are incremented as follows:

- The major revision number increments for functional changes that add or modify a Port Specification capabilities.
- The minor revision number increments for non-functional changes.

Every publication of a revised Port Specification must have a different revision number.

9.3 Port Specification Process

Proposals for the addition of a new Port Specification or the modification of an existing Port Specification may be submitted by any HCF member company or the HCF staff. These proposals are forwarded to the Port Specification Committee:

- Proposals for a new Port Specification must include the Conceptual Port Specification.
- Proposed modifications must state the Port Specification affected, changes envisioned and the benefits the changes would provide. An updated Draft Port Specification should be included with the proposal.
- Proposals for the advancement of a Port Specification (e.g., when advancing a specification from Draft to Preliminary) must include the completed specification.

The Port Specification Committee must evaluate all proposals in a timely fashion and may reject the specification, modify the specification, forward the specification to a technical working group, or accept the proposed Port Specification. The decisions and rationale for all Port Specification Committee actions shall be published in the Committee's Meeting Minutes.

9.3.1 Conceptual and Draft Port Specifications

Once the Conceptual Port Specification is accepted by the Port Specification Committee it is given a document number and published to the HCF membership along with an invitation to participate in developing a Port Specification. The resulting participants form an HCF technical working group to develop the specification. The working group must have meetings and discussions and develop the Draft Port Specification. Meeting announcements shall be made sufficiently in advance to allow HCF member participation. Meeting minutes shall be published within four weeks after the meeting. Each revised Draft Port Specification shall be published to allow for HCF member review. Once the Draft Port Specification is complete and a consensus achieved, it is forwarded for review by the Port Specification Committee.

9.3.2 Preliminary Port Specification

Once a Draft Port Specification has been accepted by the Port Specification Committee, it is elevated to Preliminary status. The Preliminary Port Specification must be published to allow comment by the HCF membership. The Port Specification Committee shall notify the HCF membership that the Preliminary Port Specification is ready for incorporation into the Port Specification and issue a Last-Call for comments. The comment period must last at least 30 days. All comments are reviewed and, if possible, incorporated into the Preliminary Port Specification. Serious flaws indicated in the comments may cause the Port Specification Committee to reject the specification and return it to Draft status for further revision by the working group.

9.3.3 Final Port Specification

If the comments indicate that consensus has been achieved, the Port Specification Committee may elevate the Port Specification to Final. The Final Port Specification shall be published to the HCF membership. Incorporation into HART compatible products should only occur after a Final Port Specification is published. The HCF staff must be notified of any products using the Port Specification prior to its inclusion into the *Block Data Transfer Specification*. These notifications shall be treated as confidential by HCF Staff and any pending changes to the Port Specification will be forwarded to affected developers. Once at least two field devices and one host application supports a Port Specification interoperability should be tested. Any flaws in the Final Port Specification shall be forwarded to the Port Specification Committee. Upon successful completion of the interoperability testing, the Final Port Specification will be scheduled for incorporation into the next revision of the *Block Data Transfer Specification*.

9.4 Port Specification Modifications

Once a Port Specification is incorporated into the *Block Data Transfer Specification,* modifications or enhancements may occasionally be justified. Modifications are classified by the Port Specification Committee as functional or non-functional using the following guidelines:

1. Additions to Tables found in the Port Specification should be classified as non-functional.
2. Additions or changes to a Port Specification should be classified as a functional change.
3. Once a Final Port Specification has been incorporated into the *Block Data Transfer Specification*, any changes must be backward compatible.

A Port Specification containing functional changes must return to the Draft level and progress through the entire process defined in Section 9.3. Changes determined as non-functional may, at the discretion of the Port Specification Committee, be immediately incorporated into the next revision of the *Block Data Transfer Specification* .

9.5 The Port Specification Committee

The Executive Committee of the HCF shall appoint the Port Specification Committee to control changes to this specification. The Port Specification Committee shall consist of three members. Each member should have a minimum of 5 years work experience with the Data Link and Application Layer of the Protocol. Since this group is responsible for the proposed Port Specification's technical excellence, experience in only one area of the protocol is not sufficient. The Port Specification Committee shall:

- Review all proposed changes to the *Block Data Transfer Specification* ;
- Ensure the *Block Data Transfer Specification* change procedures are followed;
- Determine Port Specification advancement;
- Document Committee actions; and
- Verify that a Port Specification adheres to all requirements of the HART Protocol.

The Chair must keep accurate and complete records of all meetings, design decisions and discussions. The Chair is responsible for publishing meeting minutes. All e-mail and other correspondence must be archived and available to the HCF membership. Port Specification Committee actions may not be implemented prior to the publication of the meeting minutes documenting the action being published to the HCF membership.

9.6 Appeals

The *Block Data Transfer Specification* control process is designed in such a way that compromises can be made, and genuine consensus achieved while ensuring the interoperability, value and technical excellence of each individual Port Specification. However, there are times when even the most reasonable and knowledgeable people are unable to agree. This section specifies the procedures that shall be followed for disputes that cannot be resolved through Working Group actions that normally reach consensus. These disputes must be resolved through open review and discussion.

9.6.1 Working Group Disputes

Any person who disagrees with a Working Group recommendation shall always first discuss the matter with the Working Group chair, who may involve other members of the Working Group (or the Working Group as a whole) in the discussion.

If resolution cannot be achieved in this manner, then any of the parties may request the assistance of the Port Specification Committee. The Port Specification Committee shall attempt to resolve the dispute.

If the disagreement cannot be resolved with the assistance of the Port Specification Committee then the matter may be appealed to the HCF Executive Committee. The Executive Committee shall review the situation and resolve it in a manner of its own choosing. Executive Committee decisions are final.

9.6.2 Port Specification Committee Disputes

The Port Specification Committee is responsible for controlling changes to the *Block Data Transfer Specification*. Disagreements with Port Specification Committee actions must follow the procedures outlined in this section.

If a person disagrees with the action taken by the Port Specification Committee, that person should first discuss the issue with the Port Specification Committee Chair. If the complaint is not resolved then the Port Specification Committee should re-examine the action taken, along with input from the complainant, and determine whether any further action is needed. The results of this review must be published in the Committee’s meeting minutes.

If this does not resolve the dispute, then an appeal may be made to the Executive Committee. The Executive Committee shall review the situation and resolve it in a manner of its own choosing. Executive Committee decisions are final.

9.6.3 Appeals Procedure

All appeals must include a detailed and specific description of the issues that are in dispute.

All appeals must be initiated within two months of publishing the Meeting Minutes, publishing the Working Group or Port Specification Committee action, or decision being challenged.

At all stages of the appeals process, the Working Group or committee may define the procedures they will follow in making their decision.

10. REQUIREMENTS FOR PORT SPECIFICATIONS

The Block Data Transfer mechanism can be used for a very wide variety of applications limited only by the speed of HART communications and the capabilities and resources of field devices. This section provides high level requirements for the development and content of a Port Specification. In addition, record keeping and reporting requirements for a Port Specification WG are included.

10.1 Port Specification Specification Content

All Port Specification Specifications must use the outline in Table 4.

10.2 Port Specification WG Responsibilities

Port Specification development must be an open and fair process. As a result, the WG Chair must keep accurate and complete records of all meetings, design decisions and discussions. All E-Mail and other correspondence must be archived and available to the HCF membership. The WG Chair is responsible for publishing meeting minutes and each revision of the Port Specification (see Section 9.3).

Table 4. Port Specification Specification Outline

	Section	Content
	Introduction	This section is not generally considered to be part of the Port Specification. However, this section must be included and introduce the Port Specification, its objectives and its operation.
1	**Scope**	This section defines the Port Specification and what features of the corresponding class of process connections are covered.
2	**References**	References to other applicable documents. Generally, these may include industry related documents or other standards.
3	**Definitions, Acronyms and, Symbols**	Any terms that are used in the Port Specification specification are defined. Diagrams should be used whenever possible to clarify definitions and reduce ambiguity.
4	**Overview**	The Overview adds to the scope statement to provide context for the command definitions that follow. The intended operation of the Port Specification is described here. Diagrams should be included to clarify the intent of the Port Specification. Design decisions should be documented.
5	**Data Stream**	This section is mandatory and defines the content of the data stream provided by the port. This stream may be analogous to a file (e.g., device parameterization) or continuous (e.g., a stream of process-related data).
6	**Services Access Points**	This (optional) section defines any services provided by devices utilizing this port. In some cases it may be desirable to include this section to provide a "black box" model of the port's functionality.
7	**Port Operation**	This (optional) section defines the operation of the Port. In some cases it may be desirable to include this section to provide detailed specification of port operation. This may include state machines and other details (as needed)

11. TABLES

Two Tables are required to support data transfer.

11.1 Data Transfer Function Codes

This table replaces the existing Common Table 12 and defines the function codes to control the Port.

Code	Description
0	Transfer Data
1	Open Port
2	Reset Port
3	Ready to Close
4	Port Closed
250	Not Used
251	None
252	Unknown
253	Special

11.2 Port Assignments

This table replaces the existing Common Table 13 and will be used to identify the Ports recognized by the Protocol. The Interoperability WG controls additions to this table.

Port	Description
0-1	**Reserved**
2	**Read Device Configuration File**. This port allows the entire configuration of a field device to be read.
3	**Write Device Configuration File**.
4	**Read All Device Variables**. This allows all of the Device Variables to be read as a block.
5	**Read Error Log**. Some sophisticated devices log errors as they occur. The log file can assist in trouble shooting.
6	**Read Data Log**. Some (e.g. flow) devices log data to record process information over time.
7	**PV Stream**. This allows a stream of PV's to be read as a block. This effectively allows a higher sampling rate compared to using individual Command 1 transactions thus reducing aliasing of the digital readings collected.
8	**Local Panel**. Provides a virtual connection to a device's local panel.
128-239	**Device Specific**
240-255	**Reserved**

ANNEX A REVISION HISTORY

A.1 Revision 1.0

Initial Revision.