ID: QA56925 | Access Levels: Everyone

HART protocol and the SPECTRUM Controls module

Document ID	QA56925
Published Date	03/04/2020

Summary

HART protocol and the SPECTRUM Controls module

Question

Question

How does Spectrum Controls utilize HART protocol in their specialty I/O modules?

Answer

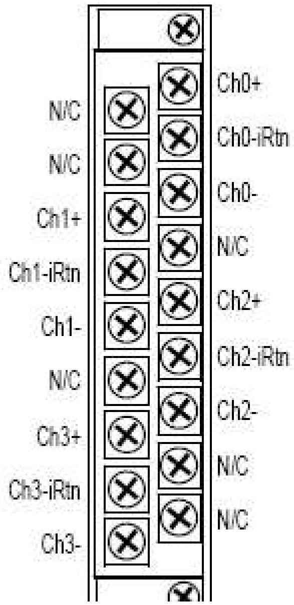
Hart Protocol and the Spectrum Controls Module

The IF4IH is a four channel isolated module that allows each channel to be configured independently for either current, voltage, or current with HART communication. The

1. Current, Voltage, Current with HART Communications

The terminal block layout is shown below:

Figure 2-1



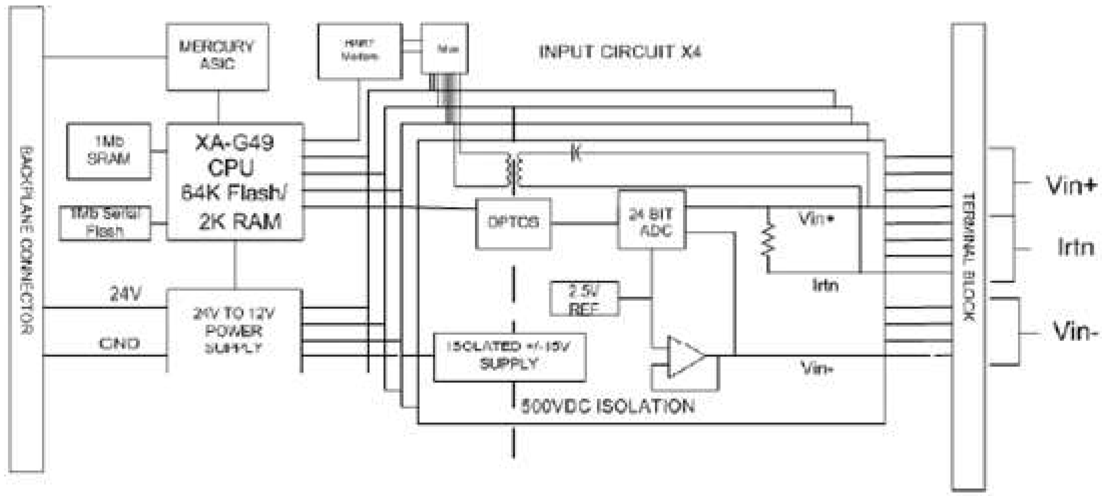
Note: A jumper must be installed on the terminal block between CH- and CH-iRtn for all current input ranges.

When the module receives the input from an analog device, the module's circuitry multiplexes the input into an A/D converter. The convert reads the signal and

process and therefore does not directly affect the analog update time.

See the block diagram below.

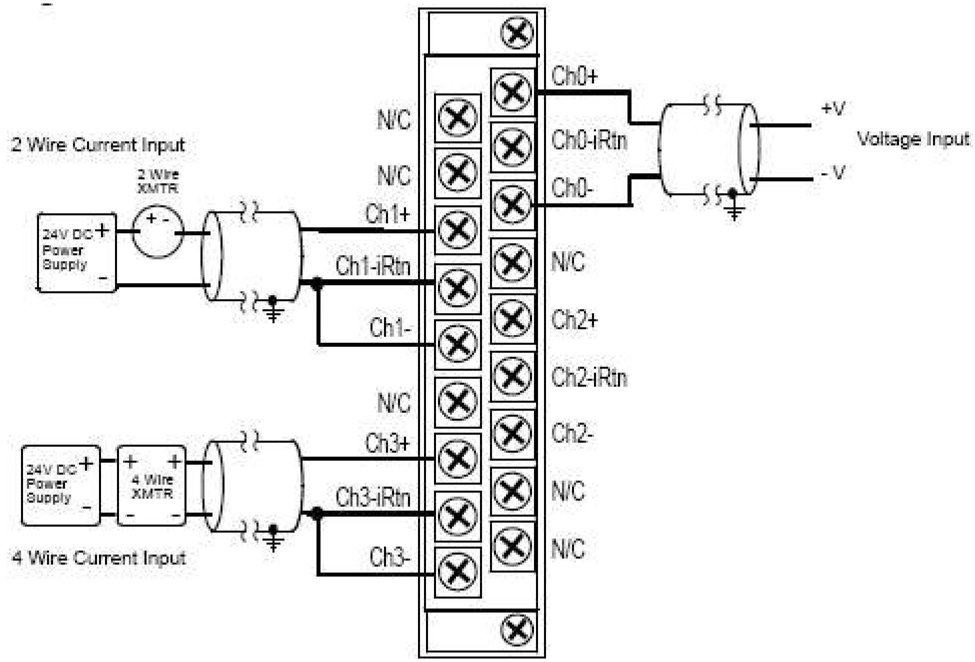
Figure 1-2



2. Onboard HART (FSK modem)
3. Maximum power supply distance rating = 8

3.7.6 Wiring Diagram

Figure 3-7

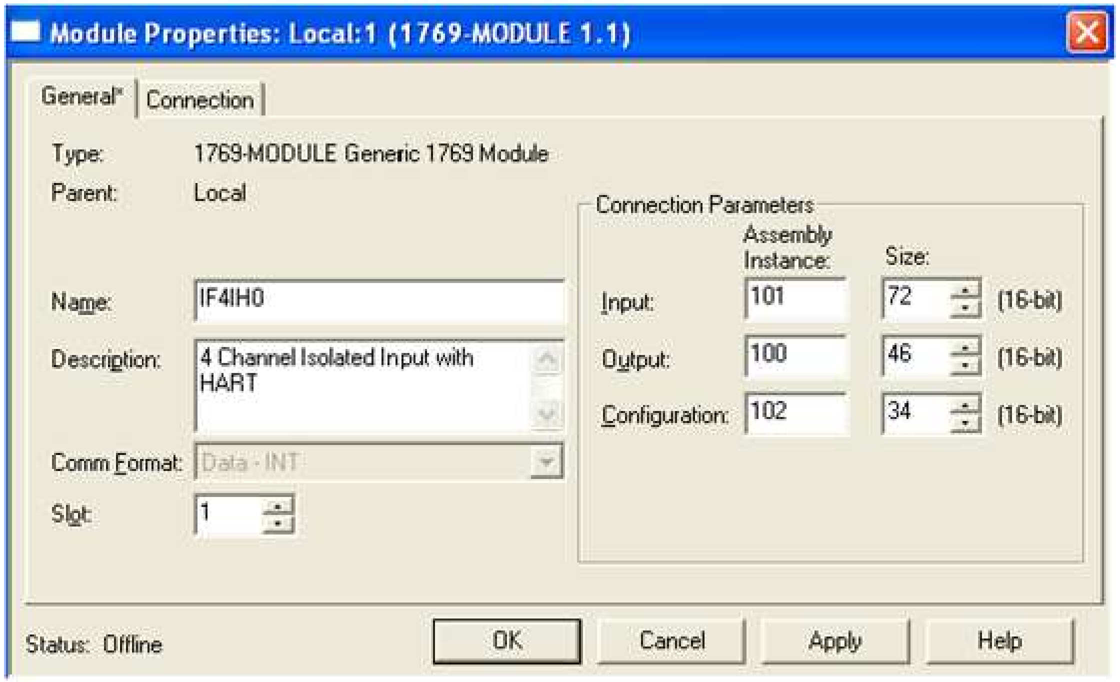


4. All current connections must have a jumper between Ch- to Ch(x)-iRtn
5. Factory calibrated
6. Generic profile screen (RSLogix 5000)

Table 4-1 (Generic Profile Parameters)

1769 I/O Module	Comm Format	Parameter	Assembly Instance	Size (16-Bit)
IF4IH	Data-INT	Input Output Config	101 100 102	72 46 34

Figure 4-6

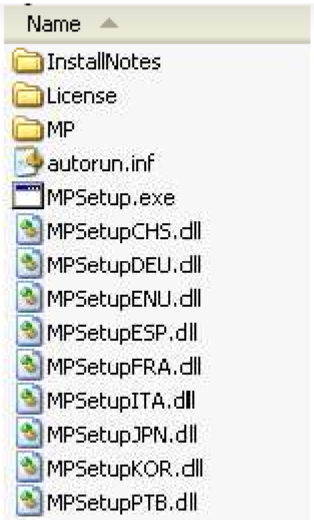


7. Addon profile

4.2.1 Installing the Add-On profile

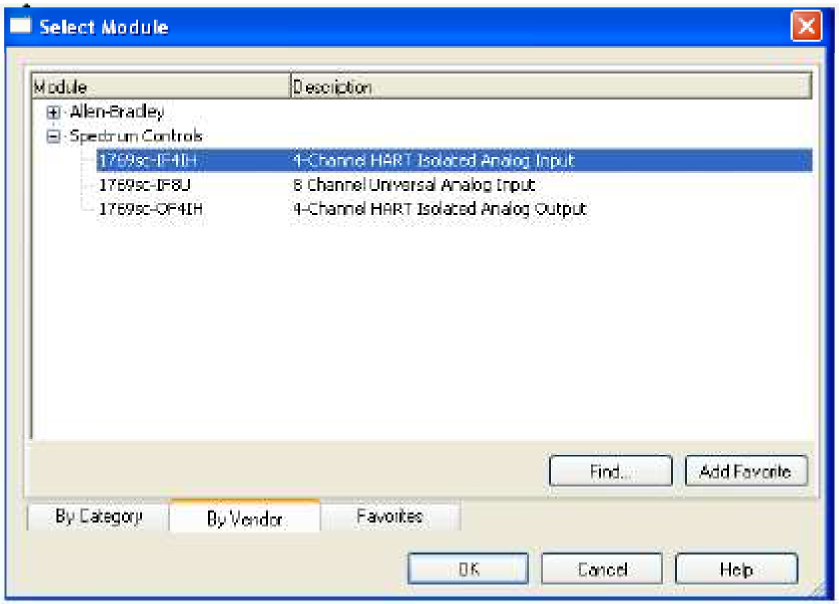
1. Download the zipped file from the Spectrum Controls website and unzip the file.

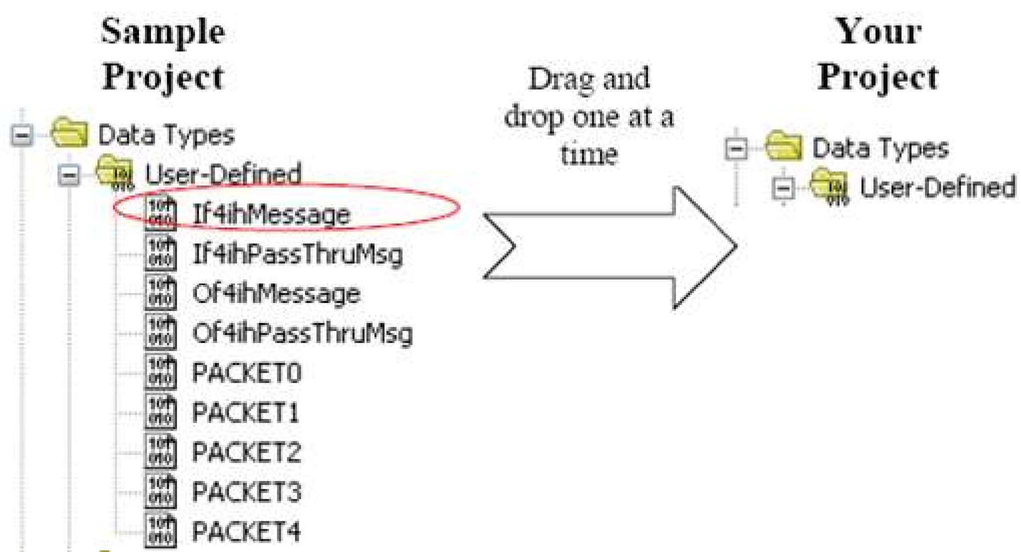
   http://www.spectrumcontrols.com/pdfs/aboi/SC_1769sc-IF4IH_DTM_1.0.0.3_Setup.zip



8. Cycle power to the computer to allow RSLogix 5000 an opportunity to register the new profile to it’s registry.

   When the dialog screen opens, select the "By Vendor" tab and expand the Spectrum Controls folder.


Select Module
Module
Description
Allen-Bradley
Spectrum Controls
1769sc-IF4IH
4-Channel HART Isolated Analog Input
1769sc-IF8U
8 Channel Universal Analog Input
1769sc-OF4IH
4-Channel HART Isolated Analog Output
Find...
Add Favorite
By Category
By Vendor
Favorites
OK
Cancel
Help


Sample Project
Data Types
User-Defined
If4ihMessage
If4ihPassThruMsg
Of4ihMessage
Of4ihPassThruMsg
PACKET0
PACKET1
PACKET2
PACKET3
PACKET4
Drag and drop one at a time
Your Project
Data Types
User-Defined

If4ihMessage	This data type defines the structure for tags used to send messages to and from the module using the paging scheme.
If4ihpass-thruMsg	Defines the structure for tags used to send HART pass-thru messages to and from the module.
Packet0	Defines the data structure for HART packet 0. HART packet zero contains device information for the connect HART device.
Packet1	Defines the data structure for HART packet 1. HART packet 1 is used to display the four dynamic variables for the selected HART device.
Packet2	Defines the data structure for HART packet 2. HART packet 2 is used to display the slot variable for the connected HART device.
Packet3	Defines the data structure for HART packet 3. HART packet 3 displays the ASCII message for the connected HART device.
Packet4	Defines the data structure for HART packet 4. HART packet 4 contains the extended status for the connected HART device.

10. Input data file

The input data file allows you to access module input data for use in the control program, via word and bit access. The data table structure is shown in the table below.

Table 6-1 (Module Input Image)

Word/Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Analog Input Data Channel 0															
1	Analog Input Data Channel 1															
2	Analog Input Data Channel 2															
3	Analog Input Data Channel 3															
4	Time Stamp Value															
5	OS3	OS2	OS1	OS0	Not Used								S3	S2	S1	S0
6	L3	H3	U3	O3	L2	H2	U2	O2	L1	H1	U1	O1	L0	H0	U0	O0
7	Pad (16 bit alignment)															
8..27	HART Packet Data															
28	Message Slave Control															
29	Message Response Size															
30..49	Message Response Buffer															
50..71	Reserved															

Words 0,1,2,3 are the real analog input values

Word 4 is a free running 15 bit clock 0-32767

6.3.3 General Status Bits S0 to S3 (Word 5)

Bits S0 through S3 of word 5 contain the general status information for channels 0 through 3, respectively. If set (1), this bit indicates an error (over- or under-range, low or high alarm, or channel data not valid). The data not valid condition is described below.

6.3.4 Out of Service Status Bits OS0 to OS3 (Word 5)

Bits SO0 through SO3 of word 0 indicate whether the associate channel is *out of service* (i.e. automatic HART acquisition is suspended).

NOTE: A channel that is placed out-of-service (i.e. Suspended) will

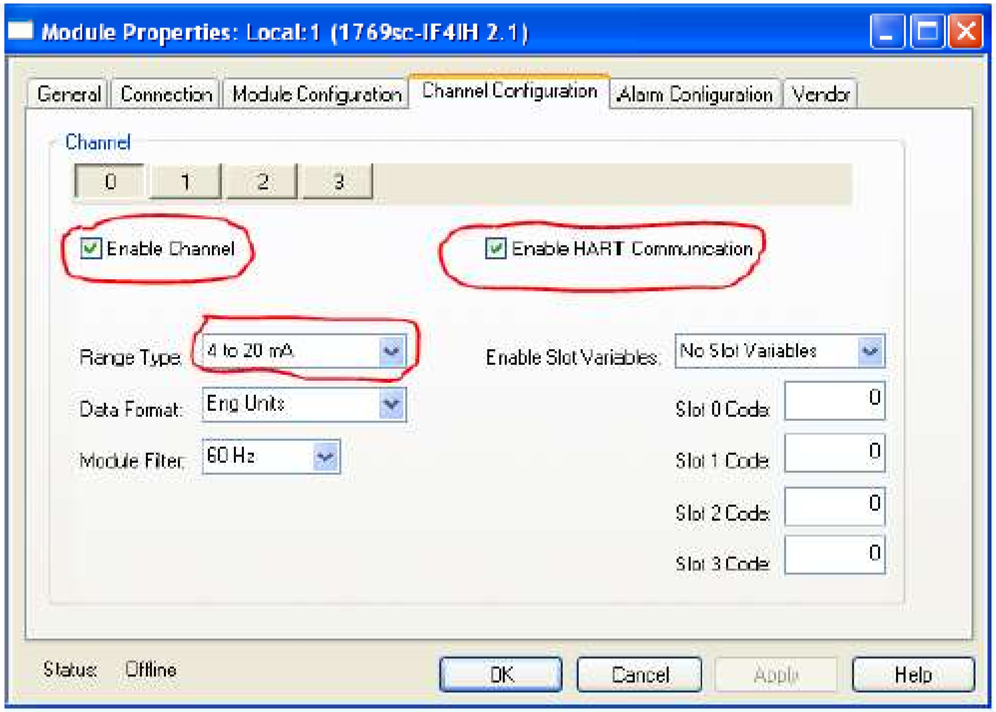
Over-range bits for channels 0 through 3 are contained in word 6, even-numbered bits. They apply to all input types. When set (1), the over-range flag bit indicates an input signal that is at the maximum of its normal operating range for the represented channel or sensor. The module automatically resets (0) the bit when the data value falls below the maximum for that range.

NOTE: If a channel is configured for a voltage type input and an open-circuit condition is present, the over-range flag bit will be set to indicate the open circuit condition and the associated channel data word will display the full-scale value.

6.3.6 Under-Range Flag Bits U0 to U3 (Word 6)

Under-range bits for channels 0 through 3 are contained in word 6, off-numbered bits. They apply to all input types. When set (1), the under-range flag bit indicates an input signal that is at the minimum of its normal operating range for the represented channel or sensor. The module automatically resets (0) the bit when the under-range condition is cleared and the data value is within the normal operating range.

NOTE: If a channel is configured for a current type input and an open-circuit condition is present, the under-range flag bit will be set to indicate the open circuit condition and the associated channel data word will display the minimum scale value.



11. Add-on profile layout

7.2.2 Auto Acquisition

When a channel is configured for HART, the module will automatically search and establish a connection to any HART field device wired to the channel. Once the module establishes a connection, it will begin to acquire HART data, including device specific codes (i.e., Manufacturer ID, serial number, etc.), the four dynamic variables, extended device status, slot variables (if enabled) , and any stored ASCII message descriptor that may be present. The HART data retrieved automatically by the module is then displayed in the input image (If4ih0Input.HartData) and is accessible by ladder logic. The HART data will update, on average, every 3.5 seconds if all four channels are enabled for HART. The module initiates the connection by sending a string of HART commands to the field device. Please see figure below.

12. a) Manufacturer ID
    b) serial number

Packet 0

If4ih0Packet0[X,0].ManufacturerID	SINT	DEC	HART device Manufacturer ID
If4ih0Packet0[X,0].DeviceType	SINT	DEC	HART device type code
If4ih0Packet0[X,0].NumPreambles	SINT	DEC	Minimum number of preambles the device requires.
If4ih0Packet0[X,0].UniversalCmdCode	SINT	DEC	HART Universal command set 5.0
If4ih0Packet0[X,0].XmitterRev	SINT	DEC	HART Transmitter specific revision
If4ih0Packet0[X,0].SwRev	SINT	DEC	HART device software revision number

Packet 1

If4ih0Packet1[X,0].HartCommStatus	SINT	BIN	HART communication status byte. Refer to appendix D for more details.
If4ih0Packet1[X,0].HartDevStatus	SINT	BIN	HART device status byte. Refer to appendix D for more details.
If4ih0Packet1[X,0].HartPV	REAL	FLOAT	HART Primary Variable
If4ih0Packet1[X,0].HartSV	REAL	FLOAT	HART Secondary Variable
If4ih0Packet1[X,0].HartTV	REAL	FLOAT	HART Tertiary Variable
If4ih0Packet1[X,0].HartFV	REAL	FLOAT	HART Fourth Variable
If4ih0Packet1[X,0].HartPVUnits	SINT	DEC	HART Primary Variable units code
If4ih0Packet1[X,0].HartSVUnits	SINT	DEC	HART Secondary Variable units code
If4ih0Packet1[X,0].HartTVUnits	SINT	DEC	HART Tertiary Variable units code
If4ih0Packet1[X,0].HartFVUnits	SINT	DEC	HART Fourth Variable units code
If4ih0Packet1[X,0].PV_Assignment	SINT	DEC	HART Primary Variable code
If4ih0Packet1[X,0].SV_Assignment	SINT	DEC	HART Secondary Variable code
If4ih0Packet1[X,0].TV_Assignment	SINT	DEC	HART Tertiary Variable code
If4ih0Packet1[X,0].FV_Assignment	SINT	DEC	HART Fourth Variable code
If4ih0Packet1[X,0].RangeLow	REAL	FLOAT	Low transmitter range for analog signal in engineering units
If4ih0Packet1[X,0].RangeHi	REAL	FLOAT	High transmitter range for analog signal in engineering units
If4ih0Packet1[X,0].Pad	SINT[4]	DEC	Packet pad (32 bit alignment)

[1] X represents the module channel number (0 to 3)

Packet 2

Tag Name	Data Type	Style	Description
If4ih0Packet2	Packet2[4,1]	NA	Two dimensional array containing packet 2 data for all 4 channels.
If4ih0Packet2[X,0][1]	Packet2	NA	Packet 2 data for channel X
If4ih0Packet2[X,0].HartChannelID	INT	BIN	Bits 0 to 3: Channel number (0 – 3). Bit 4: Searching/Initializing HART device Bit 5: HART communication failure or device not found Bit 6: Pass-through message pending (ready) Bit 7: Unused (0) Bits 8 to 10: Packet ID Bit 11 through 15: Unused
If4ih0Packet2[X,0].Slot0Data	REAL	Float	Variable for slot 0
If4ih0Packet2[X,0].Slot1Data	REAL	Float	Variable for slot 1
If4ih0Packet2[X,0].Slot2Data	REAL	Float	Variable for slot 2
If4ih0Packet2[X,0].Slot3Data	REAL	Float	Variable for slot 3
If4ih0Packet2[X,0].Slot0Units	SINT	DEC	Slot 0 units code
If4ih0Packet2[X,0].Slot1Units	SINT	DEC	Slot 1 units code
If4ih0Packet2[X,0].Slot2Units	SINT	DEC	Slot 2 units code
If4ih0Packet2[X,0].Slot3Units	SINT	DEC	Slot 3 units code
If4ih0Packet2[X,0].Slot0Assignment	SINT	DEC	Slot 0 variable code
If4ih0Packet2[X,0].Slot1Assignment	SINT	DEC	Slot 1 variable code
If4ih0Packet2[X,0].Slot2Assignment	SINT	DEC	Slot 2 variable code
If4ih0Packet2[X,0].Slot3Assignment	SINT	DEC	Slot 3 variable code
If4ih0Packet2[X,0].Pad	SINT[12]	DEC	Packet pad

Packet 3

Tag Name	Data Type	Style	Description
If4ih0Packet3	Packet3[4,1]	NA	Two dimensional array containing packet 3 data for all 4 channels.
If4ih0Packet3[X,0][1]	Packet3	NA	Packet 3 data for channel X
If4ih0Packet3[X,0].HartChannelID	INT	BIN	Bits 0 to 3: Channel number (0 – 3). Bit 4: Searching/Initializing HART device Bit 5: HART communication failure or device not found Bit 6: Pass-through message pending (ready)

Packet 4

Table 7-5 (HART Packet 4)

Tag Name	Data Type	Style	Description
If4ih0Packet4	Packet4[4,1]	NA	Two dimensional array containing packet 4 data for all 4 channels.
If4ih0Packet4[X,0].HartChannelID	INT	BIN	Bits 0 to 3: Channel number (0 – 3). Bit 4: Searching/Initializing HART device Bit 5: HART communication failure or device not found Bit 6: Pass-through message pending (ready) Bit 7: Unused (0) Bits 8 to 10: Packet ID Bit 11 through 15: Unused
If4ih0Packet4[X,0].Date	SINT[3]	DEC	Stored date in the field device
If4ih0Packet4[X,0].FinalAssemblyNumber	SINT[3]	DEC	The final assembly number is used for identifying the materials and electronics that comprise the field device.
If4ih0Packet4[X,0].ExtendedStatus	SINT[24]	DEC	The extended status returned by HART command 48
If4ih0Packet4[X,0].Pad	SINT[3]	DEC	Pad 32 bit alignment

[1] X represents the module channel number (0 to 3)

Controller tag input table

Table 7-7 (Module Input Table)

Word/Bit[1]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Analog Input Data Channel 0															
1	Analog Input Data Channel 1															
2	Analog Input Data Channel 2															
3	Analog Input Data Channel 3															
4	Time Stamp Value															
5	OS3	OS2	OS1	OS0	Not Used								S3	S2	S1	S0
6	L3	H3	U3	O3	L2	H2	U2	O2	L1	H1	U1	O1	L0	H0	U0	O0
7	Pad (16 bit alignment)															
8..27	HART Packet Data															

HART protocol specifies a message structure as follows:

Figure 7-11 (HART Message Structure)

Preamble	Start Character	Address	Command	Byte Count	Status	Data	Checksum

Note: The HART protocol supports two different formats, long and short frame. Older HART instruments (up to HART revision 4) used a short frame format. In this format, the address of the slave device is either 0, for non-multidrop devices using the 4-20mA current signal, or 1-15 for multidrop devices.

Preamble

The preamble consists of three or more hexadecimal FF characters (all 1s) allowing the receiving modem to get its frequency-detection circuits synchronized to the signal after any pause in transmission.

NOTE: The preamble does not need to be included in the HART message when using the module specific pass-thru command. The pass-thru command already includes the preamble.

Start Character

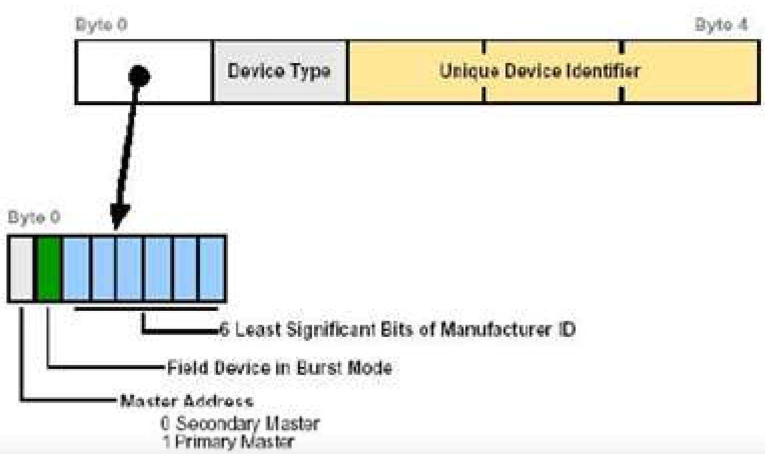
The start character in a HART message has various values, indicating which frame format is being used, the source of the message, and whether a field device is in burst mode. The possible definitions are shown in the table below.

Slave to master	06 (Hex)	86 (Hex)
Burst mode from slave	01 (Hex)	81 (Hex)

Address

The address field contains both the host and field device addresses for the message. These may be contained in a single byte (short frame format) or in five bytes (long frame format). Since the module presently only supports the long frame form, we will omit the discussion of the short frame form. In either format, the single-bit address of the master is the most significant. Only two masters are allowed for example, a control system and a hand-held communicator. The most significant bit of the address field differentiates these two hosts. Primary masters such as the IF4IH use address 1, and secondary masters such as handhelds use address 0. Please see figure below.

Figure 7-12 (Long Frame Address)



reply.

There are three categories of commands: universal commands, which all HART devices must implement; common practice commands, which should be used if the particular function is provided; and device specific commands, which are for function more or less unique to a particular slave device.

Byte Count

The byte count portion of the message contains an integer value representing the number of bytes that form the remainder of this message excluding the checksum. In other words, the byte count determines the length of the data and status.

Status

Status is included only in reply messages from a slave. It consists of two bytes of bit-coded information. The first byte indicates communication errors, if any. Otherwise if communication was good, this byte may indicate the status of the received command such as a busy device, or a command not recognized. The second status byte indicates the operational state of the slave device. A properly operating slave device will have both status bytes set to logical zero. The meaning of the individual status bits can be found in Appendex D.

Data

This portion of the HART message contains the data, if any, for the command. Not all commands or responses contain data. For those that do, up to 25 bytes can be included. Data may be in the form of unsigned integers, floating point numbers, or ASCII character strings. The number of bytes of data and the data format used for further check on transmission integrity, beyond the parity check on the 8 bits of each individual byte.

HART protocol command set

0	Read unique identifier	none	Byte 0 "254" (expansion) Byte 1 manufacturer identification code Byte 2 mfr's device type code Byte 3 number of preambles Byte 4 universal command revision Byte 5 transmitter-specific cmd revision Byte 6 software revision Byte 7 hardware revision (H) Byte 8 device function flags (B) Byte 9-11 device ID number
1	Read primary variable	none	Byte 0 PV units code Byte 1-4 primary variable (F)
2	Read current and percent of range	none	Byte 0-3 current (mA) (F) Byte 4-7 percent of range (F)
3	Read current and four (predefined) dynamic variables	none	Byte 0-3 current (mA) (F) Byte 4 PV units code Byte 5-8 primary variable (F) Byte 9 SV units code Byte 10-13 second variable (F) Byte 14 TV units code Byte 15-18 third variable (F) Byte 19 FV units code Byte 20-23 fourth variable (F)
6	Write polling addr.	Byte 0 polling address	as in command
11	Read unique ident. associated with tag	Byte 0-5 tag (A)	as command 0
12	Read message	none	Byte 0-23 message (A)
13	Read tag, descriptor, date	none	Byte 0-5 tag (A) Byte 6-17 descriptor (A) Byte 18-20 date (D)
14	Read PV sensor information	none	Byte 0-2 sensor serial number Byte 3 units code for sensor limits and min span Byte 4-7 upper sensor limit (F) Byte 8-11 lower sensor limit (F) Byte 12-15 minimum span (F)
15	Read output information	none	Byte 0 alarm select code Byte 1 transfer function code Byte 2 PV/range units code Byte 3-6 upper range value (F) Byte 7-10 lower range v (F) Byte 11-14 damping value (sec) (F) Byte 15 write-protect code Byte 16 private-label distributor code
16	Read final assembly number	none	Byte 0-2 final assembly number
17	Write message	Byte 0-23 message (A)	
18	Write tag,	Byte 0-5 tag (A)	as in command

COMMAND NUMBER AND FUNCTION		DATA IN COMMAND	(TYPE)		DATA IN REPLY	(TYPE)
33 Read transmitter variables	Byte 0	transm. var. code for slot 0		Byte 0	transmitter var. code for slot 0	
	Byte 1	transm. var. code for slot 1		Byte 1	units code for slot 0	
	Byte 2	transm. var. code for slot 2		Byte 2-5	variable for slot 0	(F)
	Byte 3	transm. var. code for slot 3		Byte 6	transmitter var. code for slot 1	
	Byte 7	units code for slot 1				
		(truncated after last requested code)		Byte 8-11	variable for slot 1	(F)
	Byte 12	transmitter var. code for slot 2				
	Byte 13	units code for slot 2				
	Byte 14-17	variable for slot 2				(F)
	Byte 18	transmitter var. code for slot 3				
	Byte 19	units code for slot 3				
	Byte 20-23	variable for slot 3 (truncated after last requested variable)				(F)
34 Write damping value	Byte 0-3	damping value (sec)	(F)		as in command	
35 Write range values	Byte 0	range units code			as in command	
	Byte 1-4	upper range value	(F)			
	Byte 5-8	lower range value	(F)			
36 Set upper range value (= push SPAN button)		none			none	
37 Set lower range value (= push ZERO button)		none			none	
38 Reset "configuration changed" flag		none			none	
39 EEPROM control	Byte 0	EEPROM control code (0 = burn EEPROM, 1 = copy EEPROM to RAM)			as in command	

40 Enter/exit fixed current mode	Byte 0-3	current (mA) (F) (0 = exit the mode)		as in command
41 Perform transmitter self test		none		none
42 Perform master reset		none		none
43 Set PV zero		none		none
44 Write PV units	Byte 0	PV units code		as in command
45 Trim DAC zero	Byte 0-3	measured current (mA) (F)		as in command
46 Trim DAC gain	Byte 0-3	measured current (mA) (F)		as in command
47 Write transfer function	Byte 0	transfer function code		as in command
48 Read additional transmitter status		none	Byte 0-24	additional status
49 Write PV sensor serial number	Byte 0-2	sensor serial number		as in command
50 Read dynamic variable assignments		none	Byte 0 Byte 1 Byte 2 Byte 3	PV transmitter variable code SV transmitter variable code TV transmitter variable code FV transmitter variable code

COMMAND NUMBER AND FUNCTION		DATA IN COMMAND (TYPE)		DATA IN REPLY (TYPE)
51 Write dynamic variable assignments	Byte 0 Byte 1 Byte 2 Byte 3	PV transmitter variable code SV transmitter variable code TV transmitter variable code FV transmitter variable code		as in command
52 Set transmitter variable zero	Byte 0	transmitter variable code		as in command
53 Write transmitter variable units	Byte 0 Byte 1	transmitter variable code transmitter var. units code		as in command
54 Read transmitter variable information	Byte 0	transmitter variable code	Byte 0 Byte 1-3 Byte 4 Byte 5-8 Byte 9-12 Byte 13-16	transmitter variable code transm. var. sensor serial number transm. var. limits units code transm. variable upper limit (F) transm. variable lower limit (F) transm. var. damping value (sec)(F)
55 Write transmitter variable damping value	Byte 0 Byte 1-4	transmitter variable code transmitter variable damping value (sec (F)		as in command
56 Write transmitter variable sensor serial number	Byte 0 Byte 1-3	transmitter variable code transmitter variable sensor serial number		as in command

109 Burst mode control	Byte 0 burst mode control code (0 = exit, 1 = enter)	as in command
110 Read all dynamic variables	none	Byte 0 PV units code Byte 1-4 PV value (F) Byte 5 SV units code Byte 6-9 SV value (F) Byte 10 TV units code Byte 11-14 TV value (F) Byte 15 FV units code Byte 16-19 FV value (F)

Data types:

A ASCII string (packed 4 characters per 3 bytes)
B Bit-mapped flags (bit 0 = multisensor device; bit 1 = EEPROM control required)
D Date (day, month, year—1900)
F Floating point (4 bytes IEEE 754)
H Integers xxxxx yyy (xxxxx = hardware rev., yyy = physical signaling code)
Unmarked items are 8-, 16- or 24-bit integers

Attachments

NON-INFRINGEMENT. Note that certain jurisdictions do not countenance the exclusion of implied warranties; thus, this disclaimer may not apply to you.