RS485 Supplemental Manual

QUANTIM® QMC Series
Mass Flow Controllers & Meters

BROOKS® INSTRUMENT
Beyond Measure

Section 1 Introduction

Introduction 7

Section 2 Device Configuration and Wiring

Device Configuration 11
Wiring 11

Section 3 Message Protocol Structure

Message Protocol Structure 12
Addressing Concept 12
Character Coding 12
Message Format 13
Message Structure 13
Preamble Characters 13
Start Character 14
Address Characters 14
Command Character 16
Byte Count Character 16
Status Characters 17
Data Characters 18
8-Bit Unsigned Integer Format 18
24-Bit Unsigned Integer Format 18
IEEE 754 Floating Point Format 18
ASCII Data Format 18
Packed-ASCII (6-Bit ASCII) Data Format 19
Checksum Characters 20

Section 4 Master/Slave Communications

Master/Slave Communications 21
RS-485 Line Handling 21
Establishing Communications with a Device 22
Example of Using Command #11 23
Alarm Configuration and Monitoring 25
Error Handling 25
Examples 26
Reading Flow Rate 26
Sending the Setpoint 27

Section 5 General Transmitter Information

Referenced Documents 29
Unit Conversions 29
Flow Rate Conversions 29
Temperature Conversions 29

Section 6 Universal Command Specifications

Universal Command Specifications....30
Command #0 Read Unique Identifier....30
Command #0 Specific Response Codes....31
Command #1 Read Primary Variable....31
Command #1 Specific Response Codes....31
Command #2 Read Primary Variable Current and Percentage of Rate....31
Command #2 Specific Response Codes....32
Command #3 Read Current and all Dynamic Variables....32
Command #3 Specific Response Codes....33
Command #6 Write Polling Address....33
Command #6 Specific Response Codes....34
Command #11 Read Unique Identifier Associated with Tag....34
Command #11 Specific Response Codes....35
Command #12 Read Message....35
Command #12 Specific Response Codes....36
Command #13 Read Tag, Descriptor, Date....36
Command #13 Specific Response Codes....37
Command #14 Read Primary Variable Sensor Information....37
Command #14 Specific Response Codes....38
Command #15 Read Output Information....38
Command #15 Specific Response Codes....39
Command #16 Read Final Assembly Number....39
Command #16 Specific Response Codes....39
Command #17 Write Message....40
Command #17 Specific Response Codes....41
Command #18 Write Tag, Descriptor, Date....41
Command #18 Specific Response Codes....42
Command #19 Write Final Assembly Number....42
Command #19 Specific Response Codes....43

Section 7 Common Practice Command Specifications

Common Practice Command Specifications....44
Command #33 Read Device Variables....44
Command #33 Response Codes....45
Command #35 Write Primary Range Values....45
Command #35 Response Codes....46
Command #37 Set Primary Variable Lower Range Value....46
Command #37 Specific Response Codes....47
Command #38 Reset Configuration Changed Flag....47
Command #38 Specific Response Codes....47
Command #44 Write Primary Variable Units....47
Command #44 Response Codes....48
Command #48 Read Additional Transmitter Status....48
Command #48 Specific Response Codes....48
Command #54 Read Device Variable Information....49
Command #54 Response Codes....49
Command #59 Write Number of Response Preambles....50
Command #59 Specific Response Codes....50
Command #60 Read Analog Channel and Percent of Range....51

Command #60 Response Codes 51
Command #66 Enter/Exit Fixed Analog Channel Mode 52
Command #66 Response Codes 52
Command #67 Trim Analog Channel Zero 52
Command #67 Response Codes 53
Command #68 Trim Analog Channel Gain 53
Command #68 Response Codes 54
Command #110 Read All Dynamic Variables 54
Command #110 Response Codes 55

Section 8 Transmitter Specific Command Specifications

Device/Transmitter Specific Command Specifications 56
Command #131 Read Brooks Serial Number 56
Command #131 Specific Response Codes 56
Command #132 Read Model Number 57
Command #132 Specific Response Codes 57
Command #134 Read Software Revisions 57
Command #134 Specific Response Codes 58
Command #135 Read Sensor Software Rev 58
Command #136 Read Bootloader Rev 58
Command #141 Perform Mass Flow Zero 59
Command #142 Perform Reset 59
Command #150 Read Totalizer 59
Command #151 Clear Totalizers 59
Command #152 Read On Time 60
Command #153 Read Total On Time 60
Command #154 Read Calibration Timeout 60
Command #155 Write Calibration Timeout 60
Command #156 Write Calibration Performed 61
Command #157 Read Overhaul Timeout 61
Command #158 Write Overhaul Timeout 61
Command #159 Write Overhaul Performed 61
Command #160 Read Hart Dynamic Variable 62
Command #161 Write Hart Dynamic Variable Units 62
Command #162 Read Dynamic Hart Variable Assignment 62
Command #163 Write Dynamic Hart Variable Assignment 62
Command #170 Read Control Mode 63
Command #171 Write Control Mode 63
Command #172 Read Setpoint and Percent of Range 63
Command #173 Write Setpoint 64
Command #176 Read Valve Override and Valve Drive 64
Command #177 Write Valve Override 64
Command #178 Read Setpoint Current 64
Command #180 Read Controller Values 65
Command #181 Write Controller Values 65
Command #190 Read Error Code 65
Command #192 Read Event Status Words 66
Command #200 Read Analog Output Alarm Behavior 66
Command #201 Write Analog Output Alarm Behavior 67
Command #202 Read Flow Alarm Values 67

Command #203 Write Flow Alarm Values....67
Command #204 Read Density Alarm Values....68
Command #205 Write Density Alarm Values....68
Command #206 Read Temperature Alarm Values....68
Command #207 Write Temperature Alarm Values....69
Command #208 Read Slug Alarm Values....69
Command #209 Write Slug Alarm Values....70
Command #210 Read Control Alarm Values....70
Command #211 Write Control Alarm Values....70

Section 9 Transmitter Specific Tables

Transmitter Specific Tables....71
Device Type Codes....71
Flow Rate Unit and Reference Codes....71
Density Unit Codes....72
Temperature Unit Codes....72
Write Protect Codes....72
Physical Signalling Codes....72
Device Variable Codes....72
Flag Assignments....73
Valve Override Codes....73
Totalizer Unit Codes....73

Warranty, Local Sales/Service Contact Information

Warranty, Local Sales/Service Contact Information.... Back Cover

Tables

1-1 Command Summary....8
2-1 D-Connector Communication Pins....11
3-1 Start Character Codings (Hexadecimal)....14
3-2 Status Byte Coding....17
3-3 Packed-ASCII Codes....19
4-1 Converting Tag Name to Packed-ASCII....24

Figures

2-1 RS-485 Multidrop Interconnection DMF/C and PC....11
3-1 Single Character Bit Sequence....13
3-2 HART Message Structure....13
3-3 Start Character Settings....14
3-4 Short Frame Address Character....15
3-5 Long Frame Address Character....15
3-6 Packed-ASCII Construction....19
4-1 Typical Message Exchange Using RS-485 Communications....21
4-2 Command #11 Response to Long Frame Address....23
4-3 Command #11 Master Request....24

4-4 Command #11 Response Message 24
4-5 Extracting the Long Address 25
4-6 Reading Flow Rate Example 27
4-7 Writing Setpoint Example 28

Introduction

The Brooks® Digital Communication RS485 S-Protocol provides a reliable, transaction oriented service between a master device, such as a Personal Computer, and one or more Brooks® S-Protocol compatible Mass Flow Meters and Controllers. The protocol is designed to allow a centralized controller to acquire measurement data from a Mass Flow device and, in case of Mass Flow Controllers, send setpoint values.

The Brooks QUANTIM QMC Series RS485 S-Protocol devices support digital communications as defined by this manual. This protocol is based on the HART® Communication Foundation (HCF) protocol. Brooks QUANTIM QMC Series RS485 S-Protocol devices support all the Universal Commands and many of the Common Practice commands as defined by the HCF. However, conformance to the HCF specifications is neither claimed nor implied.

The following Device Variables are supported:
1 - Mass Flow
2 - Density
3 - Volumetric Flow
4 - Temperature
5 - Valve
6 - Setpoint

Many of the HCF defined protocol commands are based on 4 transmitter variables, referred to as Primary, Secondary, Tertiary, and Quanternary Variable. The assignment of transmitter values to these variables is done at order placement per table below.

The variable assignments can be adjusted by the user using the Brooks Expert Support Tool (BEST) software.

Primary Variable (PV)	Mass Flow or Volumetric Flow
Secondary Variable (SV)	Density or Temperature
Tertiary Variable (TV)	Mass Flow or Volumetric Flow, which ever is not selected as the PV
Quaternary Variable (QV)	Density or Temperature, which ever is not selected as the SV

The only physical layer supported by the QUANTIM QMC Series RS485 S-Protocol devices is RS485 (See Section 2). The HART Communication Foundation FSK physical layer (Bell-202 modem) is NOT supported by the Brooks S-Protocol devices. Therefore, the commonly available HART “Hand Held Configurators” are NOT compatible with Brooks S-Protocol devices.

This document is intended to give a user the means to implement the protocol structure into their own control system in order to establish communication between the control system and the Brooks QUANTIM QMC Series RS485 S-Protocol devices. It does not cover the non-communication functionality of the Brooks QUANTIM QMC Series S-Protocol Mass Flow Meters and Controllers. For this description please refer to Installation and Operation Manual for your specific device.

The remaining sections of this document are summarized below:

- **Section 2 – Device Configuration and Wiring** defines how to properly configure and wire Brooks QUANTIM QMC Series S-Protocol devices for digital communications.
- **Section 3 – Message Protocol Structure** describes the HART message protocol.
- **Section 4 – Master/Slave Communications** describes the requirements of the Master in the HART protocol.
- **Section 5 – General Transmitter Information** defines transmitter specific information such as communication response times and units conversions.
- **Section 6 – Universal Commands** defines the message formats for all supported universal commands.
- **Section 7 – Common Practice Commands** defines the message formats for all supported common practice commands.
- **Section 8 – Transmitter Specific Commands** defines the message formats for all supported transmitter specific commands.
- **Section 9 – Transmitter Specific Tables** defines the meanings of various codes utilized by individual commands.

Table 1-1 provides a summary of S-Protocol commands available in the Brooks S-Protocol devices. This manual provides details that apply specifically to the Brooks QUANTIM QMC Series RS485 S- Protocol products:

Table 1-1 Command Summary

Cmd	Description	QMC
0	Read unique identifier	X
1	Read primary variable	X
2	Read primary variable current and percent of range	X
3	Read all dynamic variables and current	X
6	Write polling address	X
11	Read unique identifier associated with tag	X
12	Read message	X
13	Read tag, descriptor, date	X
14	Read primary variable sensor information	X
15	Read output information	X
16	Read final assembly number	X
17	Writer message	X
18	Write tag, descriptor, date	X
19	Write final assembly number	X
33	Read device variables	X
35	Write primary range values	X
37	Set primary variable lower range value	X
38	Reset configuration changed flag	X
44	Write primary variable range units	X
48	Read additional transmitter status	X

54	Read Device Variable information	X
59	Write number of response preambles	X
60	Read analog channel and percent of range	X
66	Enter/exit fixed analog channel mode	X
67	Trim analog channel zero	X
68	Trim analog channel gain	X
110	Read all dynamic variables	X
131	Read Serial Number	X
132	Read Model Number	X
134	Read Firmware Rev	X
135	Read Sensor Software Rev	X
136	Read Bootloader Rev	X
141	Perform Mass Flow Zero	X
142	Perform Reset	X
150	Read Totalizer	X
151	Clear Totalizers	X
152	Read On Time	X
153	Read Total On Time	X
154	Read Calibration Timeout	X
155	Write Calibration Timeout	X
156	Write Calibration Performed	X
157	Read Overhaul Timeout	X
158	Write Overhaul Timeout	X
159	Write Overhaul Performed	X
160	Read Hart Dynamic Variable	X
161	Write Hart Dynamic Variable Units	X
162	Read Dynamic Hart Variable Assignment	X
163	Write Dynamic Hart Variable Assignment	X
170	Read Control Mode	X
171	Write Control Mode	X
172	Read Setpoint and percent of range	X
173	Write Setpoint	X
176	Read Valve Override and Valve Drive	X
177	Write Valve Override	X
178	Read Setpoint Current	X
180	Read Controller Values	X
181	Write Controller Values	X
190	Read Error Code	X
192	Read Event Status Words	X
200	Read Analog Output Alarm Behavior	X
201	Write Analog Output Alarm Behavior	X
202	Read Flow Alarm Values	X
203	Write Flow Alarm Values	X
204	Read Density Alarm Values	X
205	Write Density Alarm Values	X

206	Read Temperature Alarm Values	X
207	Write Temperature Alarm Values	X
208	Read Slug Alarm Values	X
209	Write Slug Alarm Values	X
210	Read Control Alarm Values	X
211	Write Control Alarm Values	X

Device Configuration

The RS485 S-Protocol communications interface is a selectable option on all Brooks Quantim QMC Series devices. No hardware configuration is required.

All devices are shipped with the communication data rate set to 19200 baud.

⚠ WARNING

Before operating the device, ensure all fluid connections have been properly tightened and, where applicable, all electrical connections have been properly terminated.

Wiring

The RS485 communications interface is a multidrop connection making it possible to connect up to 32 devices to a computer on a single multidrop line as shown Figure 2-1. Most Computers are NOT equipped with RS485 ports. In order to connect an RS485 to a computer, you will need an RS485 to USB or RS-232C converter. Figure 2-1 shows the connection of three Brooks QUANTIM QMC Series S-Protocol devices via an RS485 bus utilizing an RS485 to USB or RS-232C converter to the USB or RS232 serial port of a typical computer. The RS485 bus requires two matching resistors of 120 Ohm, one at the end of the bus and one at the beginning, near the converter. Note that a control line from the PC to the converter is necessary to control the data direction of the RS485 buffers. The RTS (“Request To Send”) line is shown in Figure 2-1 because this line is used to control data direction in many of the commercially available converters. The actual line used depends on the converter selected.

Table 2-1 - D-Connector Communication Pins

D-Connector Pin Number	RS485
Pin #14	B (inverted driver side)
Pin #15	A (non-inverted driver side)

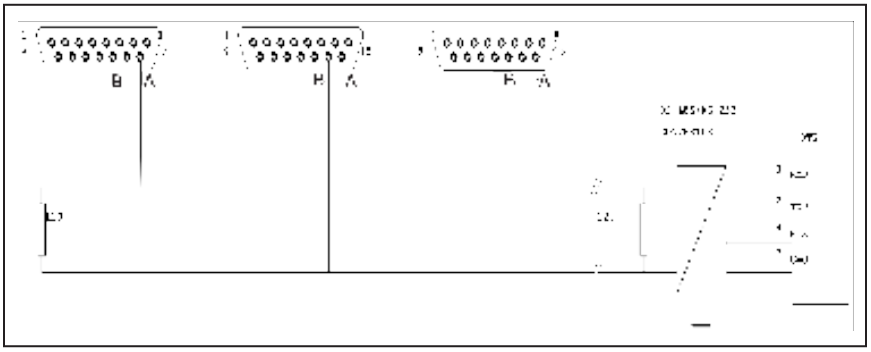

Figure 2-1 - RS485 Multidrop Interconnection

Message Protocol Structure

HART is a “master-slave” protocol: each message transaction is originated by the master (central) station, whereas the slave (field) device only replies when it receives a command message addressed to it. The reply from the slave device will acknowledge that the command has been received and it may contain the data requested by the master.

Brooks QUANTIM QMC Series S-Protocol devices do not guarantee the timing required to support multiple masters communicating simultaneously to slave devices as defined by the HART Communications Foundation. Brooks QUANTIM QMC Series S-Protocol devices do not support Burst Mode.

Addressing Concept

HART utilizes two possible addressing modes: short frame addressing and long frame addressing. The short frame addressing uses a one byte address of which the least significant nibble (four bits) is used to indicate the slave address. Because slave address 0 is reserved as a broadcast address, this provides the possibility to attach up to 15 different field devices and one master device on one multidrop bus. The long frame addressing mode uses 5 bytes (40 bits) as an address of which 38 bits are used to indicate the slave device. The slave address is built up from the manufacturer code (1 byte), the device type code (1 byte) and a device identification number (3 bytes). Long Frame Addressing allows a master to communicate with up to 16,777,215 devices on a wide area network (RS485 has a limit of 32 devices per daisy chain). Details on addressing are explained on page 15.

Character Coding

HART messages are coded as a series of 8-bit characters or bytes. These are transmitted serially, using a conventional UART (Universal Asynchronous Receiver/ Transmitter). As in normal RS-232C and other asynchronous communication links, a start bit, a parity bit and a stop bit are added to each byte. These allow the receiving UART to identify the start of each character and to detect bit errors due to electrical noise or other interference. A HART character is built up from:

1 Start bit - 0 bit
8 Databits
1 Odd parity bit
1 Stop bit - 1 bit

This sequence is summarized in Figure 3-1. Since HART is an asynchronous protocol, successive characters may be separated by idle periods (logical 1 level), but the idle period must not exceed 1 character time.

Figure 3-1 Single Character Bit Sequence

Message Format

Message Structure

HART specifies a message structure which is given in Figure 3-2 below.

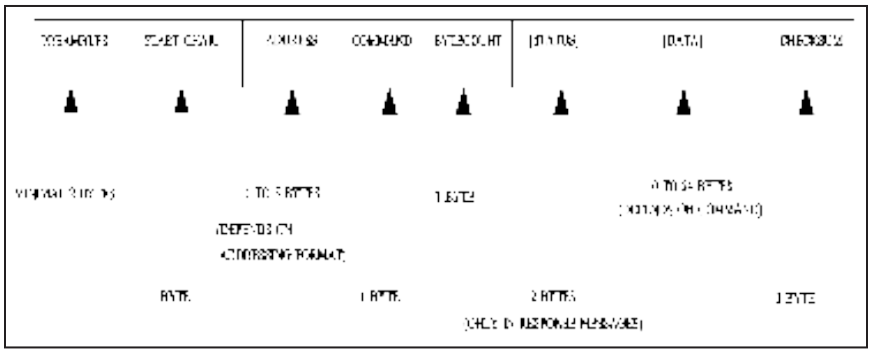

Figure 3-2 HART Message Structure

This structure is used for both the request (master to slave) and the response (slave to master) messages. The status part and the data part are shown in square brackets, because their occurrence in the message depends on the type of message (response or request message) and the command number. The individual items are explained below.

Preamble Characters

Every message, whether from a master or a slave device, is preceded by a specified number of hexadecimal FF characters (databyte with all 1's). These characters, called preamble characters, are used in the message-detect pattern together with the start character. The preamble characters are used to synchronize the field device. The Brooks QUANTIM QMC Series S-Protocol devices require at least 2 preamble characters in order to be able to proceed in the message detection with the start of message character. A master should send a minimum of 5 preamble characters in order to guarantee that slave device receives the required 2 preamble characters.

Start Character

The start character or delimiter is a one byte code used to detect the type of frame (type of message) being transmitted and the type of addressing being used. The most significant bit indicates the addressing mode used: 0 for short frame and 1 for long frame addressing, whereas the three least significant bits indicate the frame type of the message: 010 indicates a Start-Of-Text character and 110 indicates an Acknowledge character. The Start-Of-Text character is used to indicate a message from the master to a slave device whereas the Acknowledge character is used to indicate the response messages from slave devices to the master. The rest of the bits in the character are all zeros. See Figure 3-3 and Table 3-1 below.

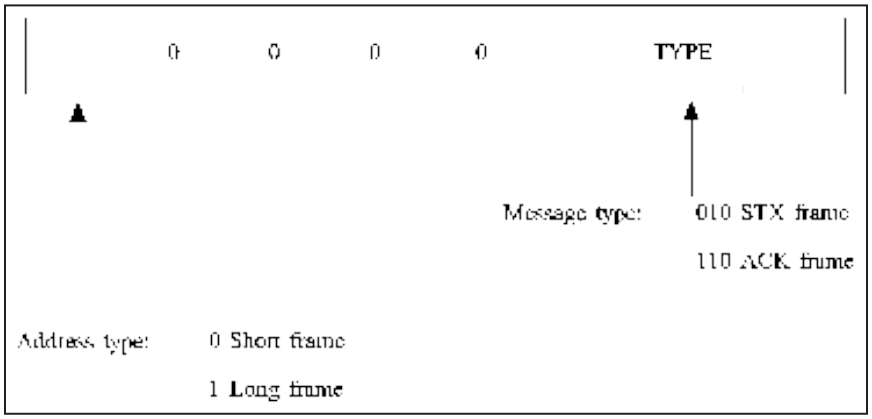

Figure 3-3 Start Character Settings

Table 3-1 Start Character Codings (Hexadecimal)

	Short frame	Long frame
Master to slave (STX)	02	82
Slave to master (ACK)	06	86
Address field length	1 byte	5 bytes

Address Characters

The address field contains both the master and the field device addresses for the message. These may be contained in a single byte (short frame format) or in five bytes (long frame format). In either format, the most significant bit is usually the single-bit address of the master device taking part in the message transaction (either sending a command or receiving a reply from a slave device). Since only two masters are allowed only one bit is needed for the master address. This bit will be 1 if it indicates the primary master system, and 0 if it indicates the secondary master system. The rest of the address field is determined by the frame format.

Figure 3-4 below shows the address character in the short frame format. The 4 least significant bits are the slave address, which can be used as a polling address.

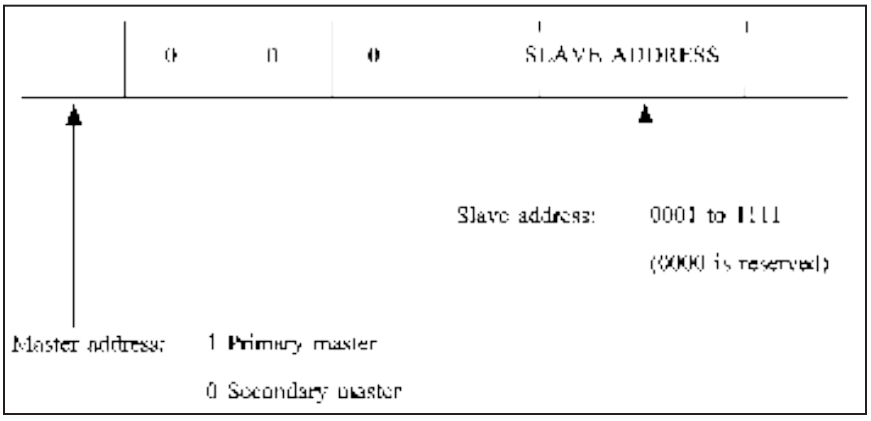

Figure 3-4 Short Frame Address Character

In the long frame format the slave device address is represented by a 38-bit number. The structure of the address is given in Figure 3-5 below.

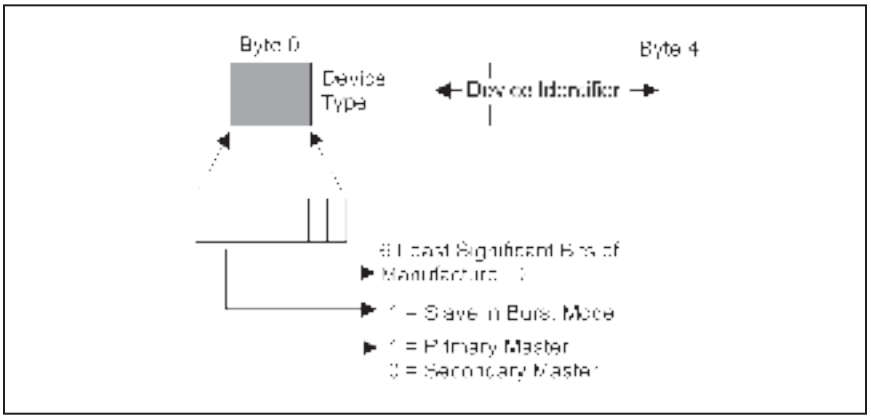

Figure 3-5 Long Frame Address Characters

In the long frame format the slave address part of the five address characters is build up from three sources: The 6 bits of the first byte of the slave address part represent the manufacturers code. In case of devices made by Brooks Instrument this is the number 10 (decimal). The manufacturer number is a number which is stored in the device by the manufacturer and which can not be changed by the user.

The second byte in the address is the device type code. This code indicates the type of the device addressed. The device type code will be 4 for all Brooks QUANTIM QMC Series S-Protocol devices. The device type code is a number which is stored in the Brooks QUANTIM QMC Series S-Protocol devices by the manufacturer and which can not be changed by the user.

The last three bytes form a 24-bit unique identification number. As the name implies, this value must be unique to each Brooks QUANTIM QMC Series S-Protocol device on a network. For legacy products this value was derived from the serial number of the device, however for the QUANTIM QMC Series this value is a random value.

A special case occurs when all bits of the slave address part are set to 0. A message with this type of address, called a broadcast address, will be accepted by all slave devices attached to the bus. A slave device will always respond to a message with the broadcast address unless the message contains additional information in the data portion of the message that allows the slave device to determine that the message is not addressed to that device. Brooks QUANTIM QMC Series S-Protocol devices support only one such command, Command #11. This type of addressing can be used to address devices of which the manufacturer and the device type codes and the unique identification number are not available to the host system and with which this information can still be retrieved from the unknown device. Command #11 data contains a Tag Name. Only a slave device with the specified Tag Name will respond to Command #11 even if the address in the message is the broadcast address. The Tag Name is an 8 character field which is equal to the last 8 digits of the device's serial number. See Establishing Communications with a Device (page 23) for a detailed description of the use of Command #11.

Command Character

The command character is a 1 byte unsigned integer in the range from 0 to 255 (decimal), which indicates the action the slave device has to perform. A larger range of commands is theoretically possible by using the expansion code or 254 (decimal) followed by a second byte. This feature however is not implemented by the Brooks QUANTIM QMC Series S-protocol devices.

Three types of commands are available to the user: the 'Universal Commands', the 'Common-Practice Commands' and the 'Transmitter-Specific Commands'. The Universal Commands are implemented by all field devices utilizing the HART protocol. Refer to Section 6 for descriptions of all available universal commands. The Common-Practice Commands can be implemented by all devices. These commands perform tasks which are often common to most devices. Refer to Section 7 for descriptions of all implemented Common-Practice Commands. The last category, Transmitter-Specific Commands are a number of commands, ranging from 128 to 250 which are specific to the type of device. Refer to Section 8 for descriptions of all available Transmitter-Specific Commands. The commands #251 to #255 are reserved.

Byte Count Character

The byte count character is a 1 byte unsigned integer indicating the number of bytes which will form the remainder of the message. This number includes the two status bytes (only if the message is a response message) and the bytes in the data part. It does NOT include the checksum byte. The byte count character is used by the receiving device to identify the checksum byte and to determine when the message is finished.

Status Characters

Status Characters consists of two bytes, which contain bit-coded information about communications errors, command errors, and device status as defined in Table 3-2. Only response messages from the slave device to the master device will contain status characters.

Table 3-2 Status Byte Coding

	First Byte	Second Byte
Communication errors	Bit 7 1 = Communication error Bit 6 Parity error (hex C0) Bit 5 Overrun error (hex A0) Bit 4 Framing error (hex 90) Bit 3 Checksum error (hex 88) Bit 2 Reserved (hex 84) Bit 1 Rx Buffer Overflow (hex 82) Bit 0 Undefined	Bit 7 0 Bit 6 Bit 5 Bit 4 All 0 Bit 3 Bit 2 Bit 1 Bit 0
Command errors	Bit 7 0 = Communication error Bit 6 to 0 (not bit-mapped): 0 Non command specific error 1 Undefined 2 Invalid selection 3 Passed parameter too large 4 Passed parameter too small 5 Incorrect byte count 6 Transmitter specific command error 7 IIn write-protect mode 8-15 Command specific errors 16 Access restricted 32 Device is busy 64 Commanded not implemented	Bit 7 Device Malfunction Bit 6 Configuration Changed Bit 5 Cold Start Bit 4 More Status available. Use Command # 48 to get more information Bit 3 Primary variable analog output fixed Bit 2 Primary variable analog output saturated Bit 1 Non primary variable out of range Bit 0 Primary variable out of range

If the communication failed (i.e. the slave received distorted information) the first byte indicates the receiver error(s) of the slave device. The second byte will then be 0. If communication did not fail, the first byte will give command execution information, whereas the second byte will give information on the status of the device. The command specific errors 8 - 15 are errors which can have a different meaning for different commands. Refer to the Sections 6, 7 and 8 for more information.

Data Characters

For the commands that contain data, the data field may contain up to a maximum of 24 8-bit data bytes. The data can appear in a number of formats described in the following sections.

8-Bit Unsigned Integer Format

This format can be used to transfer codes (e.g unit codes), indexes (e.g analog output numbers) and raw data. If a parameter, represented by an 8-bit unsigned integer in a command data part is not implemented, codes like 250, “Not Used” or 0 will be used.

24-Bit Unsigned Integer Format

This format can be used to transfer large integer data numbers (e.g. the valve values).

IEEE 754 Floating Point Format

This format is based on the IEEE 754 single precision floating point standard:

S EEEEEEE E MMMMMMM MMMMMMMM MMMMMMMM
byte # 0 byte # 1 byte # 2 byte # 3

Where: S - Sign of mantissa (1 = negative)
E - Exponent; Biased by 127 in two’s complement format
M - Mantissa; 23 least significant bits, fractional portion

The value of a parameter described in the above format can thus be found by:

$$\text{Value} = S\ 1.M * 2^{(E - 127)}$$

This format is also used in most personal computers.

The floating point parameters not used by a device will be filled with 7F A0 00 00 (hexadecimal) or ‘Not-A-Number’.

ASCII Data Format

Some of the alphanumeric data passed by the protocol is transmitted to and from the devices in the ASCII format. Refer to any ASCII Code table for the alphanumeric code assignments.

Packed-ASCII (6-bit ASCII) Data Format

Some of the alphanumeric data passed by the protocol is transmitted to and from the devices in the Packed-ASCII format. Packed-ASCII is a subset of ASCII (See Table 3-3) produced by removing the two most significant bits from each ASCII character. This allows four Packed-ASCII to be placed in the space of three ASCII characters. Typically four Packed- ASCII strings are even multiples of three bytes. Figure 3-6 illustrates the byte sequence.

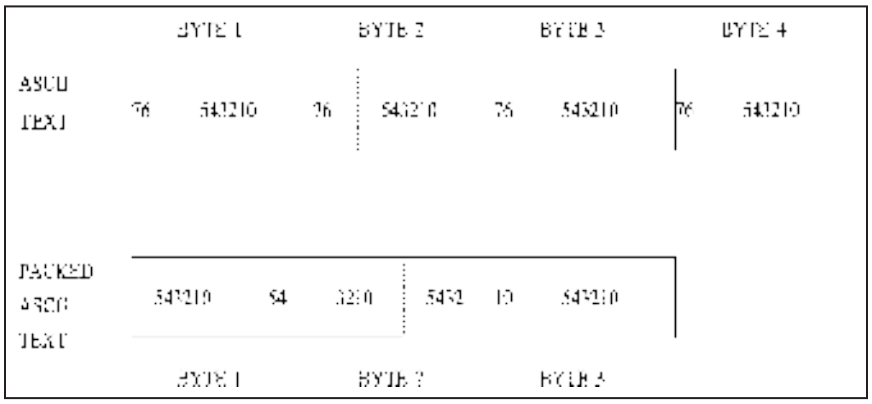

Figure 3-6 Packed-ASCII Construction

Construction of Packed-ASCII:

a. Remove bit #7 and bit #6 from each ASCII character.
b. Pack four 6-bit ASCII bytes into three bytes.

Reconstruction of ASCII characters:

a. Unpack the four 6-bit ASCII characters into four bytes.
b. Place the complement of bit #5 of each unpacked 6-bit ASCII character into bit #6.
c. Set bit #7 of each unpacked ASCII to zero.

Table 3-3 Packed-ASCII Codes

Char	Code	Char	Code	Char	Code	Char	Code
@	00	P	10	(space)	20	0	30
A	01	Q	11	!	21	1	31
B	02	R	12	“	22	2	32
C	03	S	13	#	23	3	33
D	04	T	14	$	24	4	34
E	05	U	15	%	25	5	35
F	06	V	16	&	26	6	36
G	07	W	17	‘	27	7	37
H	08	X	18	(	28	8	38
I	09	Y	19	)	29	9	39
J	0A	Z	1A	*	2A	:	3A
K	0B	[	1B	+	2B	;	3B

L	0C	\	1C	,	2C	<	3C
M	0D	]	1D	-	2D	=	3D
N	0E	^	1E	.	2E	>	3E
O	0F	_	1F	/	2F	?	3F

Checksum Characters

The checksum byte contains the ‘exclusive-or’ (‘longitudinal parity’) of all the characters preceding it in the message starting with the start character. It provides a further check on transmission integrity, beyond the one provided by the parity check on each individual byte. The exclusive-or of all the message bytes (including the start character, excluding the checksum byte) and the checksum byte itself should read exactly zero.

Master/Slave Communications

Section 3 of this manual defined the S-Protocol message structure in detail. Section 4 of this manual will describe how to utilize the S-Protocol message structure to perform master slave communications with a Brooks QUANTIM QMC Series S-protocol device. This section focuses on RS485 line handling, establishing communications with a device, error recovery, and timing. Sections 6, 7, and 8 of this manual define all S-Protocol commands available in Brooks QUANTIM QMC Series S-protocol devices. This section will conclude with examples of typical communications sequences.

Master devices initiate all communications on a Master/Slave communications network. Master devices are typically a computer of some kind but other devices such as PLC's can also operate as a Master device.

Slave devices only respond to messages initiated by a Master. Brooks QUANTIM QMC Series S-Protocol devices are always Slaves on the communications network.

RS485 Line Handling

The physical communications layer used by Brooks QUANTIM QMC Series S-Protocol devices is RS485. On an RS485 physical communications layer, all data is transmitted and received using differential signals on a single pair of wires. Since both the Master and the Slave devices use the same pair of wires to transmit their data, care must be taken to ensure that only one device has its transmitter enabled at any point in time.

Figure 4-1 shows a typical message exchange using RS485. Notice that the Master's transmitter is enabled only during the Master Request message and the Slave's transmitter is enabled only during the Slave Response message. At all other times, the transmitters on the Master and all Slaves connected to the network must be in their high impedance state, leaving the network "Un-Driven."

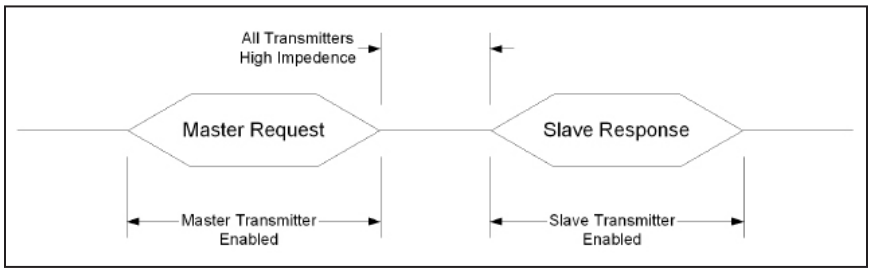

Figure 4-1 Typical Message Exchange Using RS485 Communications

It is the user's responsibility to guarantee that the Master's transmitter is enabled only during the Master Request message. Control of the Master's transmitter is dependent upon the hardware used by the Master. If an RS232 to RS485 converter is used, the most common control is the RTS signal on the RS232 interface as shown in Figure 2-1

(See page 12). Refer to the user manual for your hardware to determine the proper control method required in your system.

Timing the enabling/disabling of the transmitter is very important. The transmitter must be enabled before the first bit of the first character is transmitted and must be disabled only after the last bit of the last character is transmitted. Additionally, all transmitters have some finite turn-on/turn-off delays which may be affected by the wire length and wire quality of your network. The S-Protocol message structure attempts to minimize these affects by requiring all messages to have at least 5 preamble characters while only 2 are required for the receiving device to detect a valid message (See page 14). This allows up to 3 lost characters due to turn-on/turn-off delays.

Disabling a transmitter at the proper time is frequently a difficult task. Many UARTS/systems do not provide an indication when the last byte of a message is completely transmitted. It is more likely that an indication is provided when the last byte of a message is starting to be transmitted. Since the last byte of an S-Protocol message is the checksum byte for the message, it is critical that the transmitter remain enabled until the last byte is completely transmitted. One solution is to transmit an extra character at the end of a message (typically 0x00) and then disable the transmitter when the indication is received that the extra character is starting to be transmitted. However, the transmitter cannot be enabled too long after a message is complete. Slave devices will begin transmitting a response as soon as 5 msec after the reception of an error free request message.

High data rates increase the importance of disabling the transmitter quickly. At 19200 baud, one character time is 0.57 msec. Thus, the 3 lost character “cushion” represents only 1.72 msec. Lower data rates provide a longer “cushion” and thus is a possible solution if disabling the transmitter in a timely manner proves difficult. Another solution is to increase the number of preamble characters transmitted by the Master and/or the slave.

Establishing Communications with a Device

In order for a Master to establish communications with a Brooks QUANTIM QMC Series S-Protocol device, the Master must know the address of the Brooks device. The S-Protocol supports both Short Frame Addressing and Long Frame Addressing as defined on page 13.

Short Frame Addressing allows a master to communicate with up to 15 devices. Each device on the network must have a unique Polling Address with a value of 1–15.

Long Frame Addressing allows a master to communicate with up to 16,777,215 devices on a wide area network (RS485 has a limit of 32 devices per daisy chain). Each device is pre-programmed at the factory with a unique long address. Using the process described below, the Master can obtain the long address from the device by knowing only the

device Tag Name. The Tag Name is pre-programmed at the factory and is printed on the device's calibration sheet.

The following procedure can be performed online in order to obtain a device's long address:

1. Send Command #11 (See page 35) using Long Frame Addressing and an address of 0. In the data section of Command #11, use the device's Tag Name to identify the device. Command #11 requires that the Tag Name be transmitted in Packed-ASCII format as defined on page 20.
2. Extract the Manufacturer ID, Manufacturer's Device ID, and Device ID Number from the response and construct the Long Address Frame as shown in Figure 4-2.

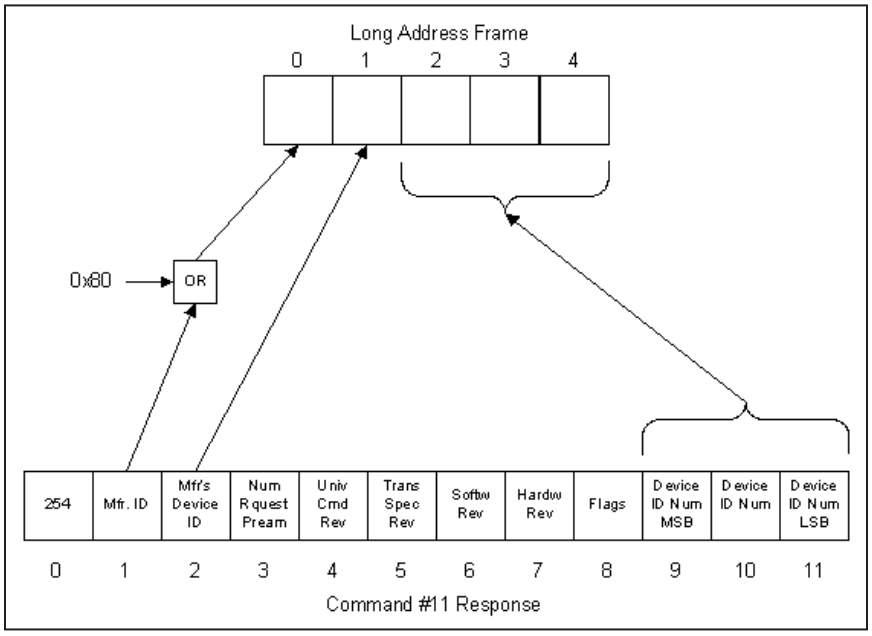

Figure 4-2 Command #11 Response to Long Frame Address

Example of Using Command #11

Command #11 reads the unique identifier from a device whose Tag Name is specified in the Command #11 request from the Master. Tag Names are strings of up to 8 characters which are limited to the reduced ASCII set defined in Table 3-3. A Tag Name consists of the last 8 digits of the device's serial number. Table 4-1 is an example of converting an 8 character Tag Name to 6 bytes in the Packed-ASCII format. In this example, the Tag Name of the device will be "MFC-1234".

Table 4-1 Converting Tag Name to Packed ASCII

	Representation							
Tag Name	MFC-1234							
Characters	M	F	C	-	1	2	3	4
8- bit ASCII (hex)	4D	46	43	2D	31	32	33	34
Bit 7 & 8 removed:								
6 bit ASCII (hex)	0D	06	03	2D	31	32	33	34
6 bit ASCII (binary)	001101	000110	000011	101101	110001	110010	110011	110100
Packed (binary)	00110100 0110 0000 11101101 11000111 00101100 11110100							
Packed (hex)	34 60 ED C7 2C F4							

Figure 4-3 shows the request message for Command #11 sent by the Master to the Brooks QUANTIM QMC Series S-Protocol device whose Tag Name is MFC-1234.

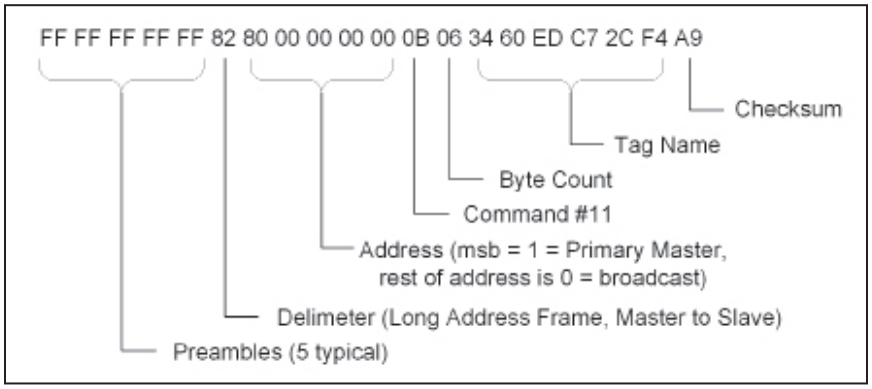

Figure 4-3 Command #11 Master Request

A possible Response Message from a Brooks QUANTIM QMC Series S-Protocol device is shown in Figure 4-4.

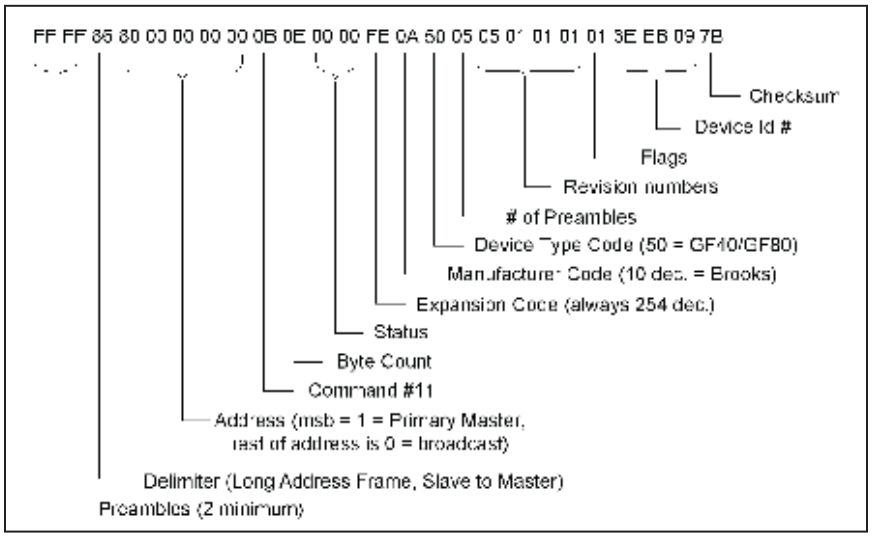

Figure 4-4 Command #11 Response Message

From the response, the long address can be extracted as shown in Figure 4-5.

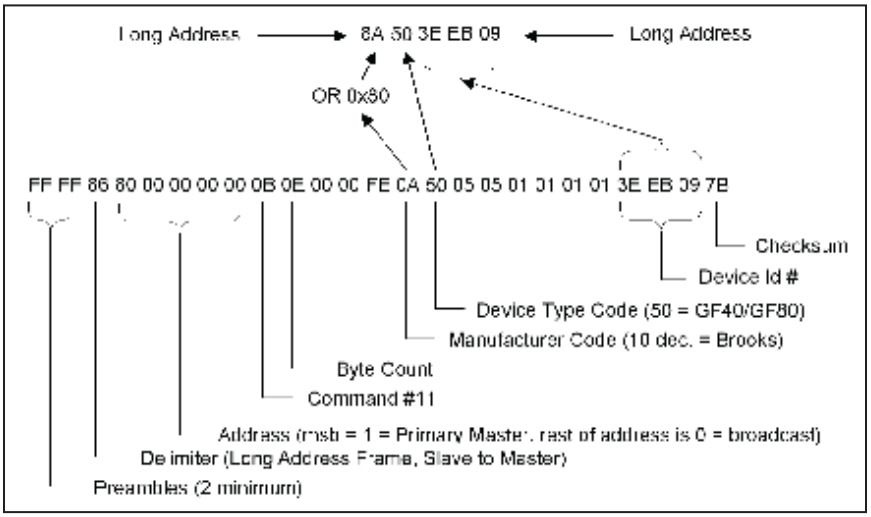

Figure 4-5 Extracting the Long Address

Alarm Configuration and Monitoring

Brooks QUANTIM QMC Series S-Protocol devices monitor for various alarm conditions such as Flow Rate, Temperature, Density, and Diagnostics. To determine which alarms conditions have been detected, use Command #48. However, it is not necessary to constantly poll Command #48 to determine when an alarm condition has been detected. All slave response messages contain a 2 byte status. If an alarm condition has been detected, then bit 4 of the second status byte will indicate “More Status Available”. Then Command #48 can be used to determine the alarm condition(s) that has been detected.

To configure which alarm conditions are monitored and reported by the device, refer to Commands 190, 192, 202, 203, 204, 205, 206, 207, 208, 209, 210, and 211 in Section 7, also Table 9-15.

Error Handling

In all communications networks, communications errors can and will occur. Both the Master and the Slave devices must be able to properly handle errors in order to maintain a operating network. When a Brooks QUANTIM QMC Series S-protocol device detects a communications error, one of two results may occur. It may respond with an error code, or it may not respond at all to the request. The result depends upon the type of error that was detected, and where in the message the error was detected. It is important that the Master handles the situation correctly.

There are two basic type of errors defined by the S-Protocol: Communications Errors and Command Response errors. The type of error can be determined by examining the Status Code returned by the slave device (See page 17). Command Response errors are

typically the result of a programming error in the Master and should not normally occur in a mature system. The main focus of this section will be Communication Errors.

Communications Errors are frequently the result of external environment issues, faulty wiring, etc. In a properly designed network, Communications Errors should be rare. A Communications Error can occur in either the Master to Slave Request or the Slave to Master response. If the error occurs in a Master to Slave request, one of two results may occur. It may respond with an error code, or it may not respond at all to the request. The result depends upon the type of error that was detected, and where in the message the error was detected. It is the responsibility of the Master device to check all Slave to Master responses for errors including message frame formatting, longitudinal parity, and vertical parity.

Regardless of the type of error and when or where it was detected, the normal way to handle a Communications Error is to simply retry the message. Typically, a master would attempt to retry a message at least twice to allow any external disturbance to clear. In the event that the retries are unsuccessful, then the Master device must handle the situation in a manner consistent with the requirements of the system. Typical responses to such an error are: Taking the device off-line so that the remainder of the network is not affected; Notifying an operator; Triggering a system alarm; etc.

A Master device must allow sufficient time for a Slave to respond before attempting to retry the message. The Master should wait 40 msec before retrying the message. As long as communications errors are infrequent, this retry delay time should not affect system performance.

Examples

The following 2 examples show the most typical messages used by a Master when communicating to a Brooks QUANTIM QMC Series S-Protocol device: Reading Flow Rate and Sending the Setpoint. These examples will use the Long Addressing Frame with the long address established in the example on page 24. The calibrated full scale of the device used in these examples is 1.0 liters per minute.

Reading Flow Rate

The flow rate of the device can be read using any of the following commands:

- Command #1 – Read Primary Variable
- Command #2 – Read Primary Variable Current and Percent of Range
- Command #3 – Read Current and All Dynamic Variables

This example will use Command #1 to read the Flow Rate of the device. This command returns the flow rate in the unit of measure as configured in the device. The units can be changed using Command #161, Dynamic

Variable Units.

In the example shown in Figure 4-6, the device returns a flow of 0.8502 liters/min.

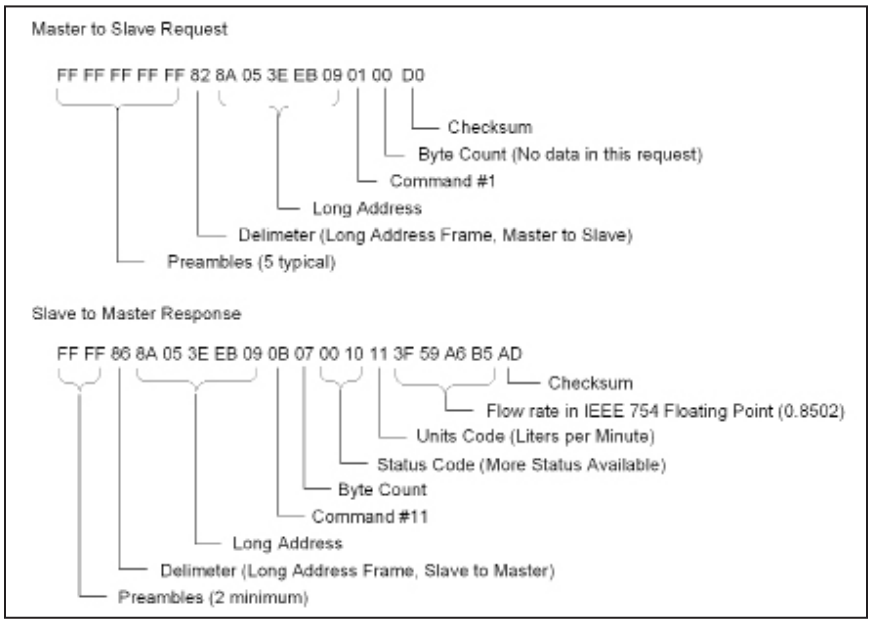

Figure 4-6 Reading Flow Rate Example

Sending the Setpoint

The Setpoint can be controlled via the network using Command #173. In the example shown in Figure 4-7, the setpoint is set to 85% of full scale.

If Setpoint is controlled via an analog input, then Setpoint can be read using Command #172.

Command 171 allows the user to change the control model from analog to digital and from digital to analog. The device will always revert to it's original mode, analog or digital when power is cycled.

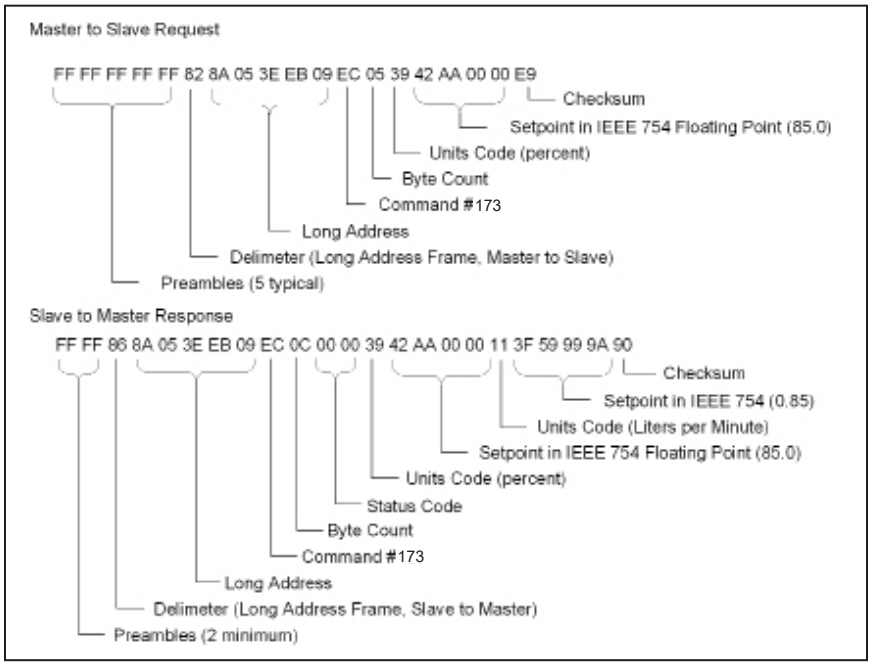

Figure 4-7 Writing Setpoint Example

Referenced Documents

The following HART documents were referenced in order to implement the protocol:

- Data Link Layer Specification Rev.
  HCF_SPEC-81 Rev 7.1
- Command Summary Information Rev.
  HCF_SPEC-99 Rev 7.1
- Command-Specific Response Code Defs. Rev.
  HCF_SPEC-307 Rev 4.1
- Universal Command Specification Rev.
  HCF_SPEC-127 Rev 5.2
- Common-Practice Command Specification Rev.
  HCF_SPEC-151 Rev 7.1
- Common Tables Rev.
  HCF_SPEC-183 Rev 11.0

Unit Conversions

Flow Rate Conversions

All flow values involved in the exchange of data during communication are converted to/from the user specified flow units. A list of supported flow units is provided in on page 71. The user can change the flow units to be used for all flow rate conversions with Command #161. Note: The Device Variable equals 1 for Flow. See Device Variable codes (page 72) for more details.

Temperature Conversions

All temperature values involved in the exchange of data during communication are converted to/from the user specified temperature units. A list of supported temperature units is provided on page 72. The user can change the temperature units to be used for all temperature conversions with Command #161. Note: The Device Variable equals 4 for Temperature. See Device Variable codes (page 72) for more details.

Universal Command Specifications

This section addresses Universal Commands available in the QUANTIM QMC Series S-Protocol devices. These universal commands are per HCF_SPEC-127, FCG TS20127 Revision 7.2.

Command #0 Read Unique Identifier

Command used to retrieve the expanded device-type codes, revision levels and the device identification number from the specified device. The device type code will always be returned in the expanded three byte format (i.e. “254”, manufacturer identification code, manufacturers device type code). The combination of the manufacturer identification code, manufacturer’s device type code and device identification code make up the unique identifier for the extended frame format of the data link layer.

Request data bytes:

NONE

Response data bytes:

254	MFR. ID	MFR's DEVICE TYPE	NUMBER RQUEST PREAM	UNIV. CMD. REV.	TRANS. SPEC. REV.	SOFTW REV.	HARDW REV.	FLAGS
#0	#1	#2	#3	#4	#5	#6	#7	#8

DEVICE ID NUM MSB	DEVICE ID NUM	DEVICE ID NUM
#9	#10	#11

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Device type code for “expansion”. Contains the code “254” (decimal).
1	8-bit unsigned integer	Manufacturer identification code. (Always 10).
2	8-bit unsigned integer	Manufacturers device type code. Refer to Device type codes, page 68.
3	8-bit unsigned integer	Number of response preamble characters required for the request message from the master to the slave.
4	8-bit unsigned integer	Universal command revision level implemented by this device.
5	8-bit unsigned integer	Transmitter specific command revision level implemented by this device.
6	8-bit unsigned integer	Software revision level of the device.
7	8-bit unsigned integer	Hardware revision level of the electronics in the device.
8	8-bit unsigned integer	Flags. Refer to Flag assignments, page 69.
9-11	24-bit unsigned integer	Device identification number.

Command #0
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #1
Read Primary Variable

Read the primary variable. The primary variable is the mass flow rate or volumetric flow rate of the device expressed in the selected flow units. See Command #161 for information on setting Flow Units.

Note: assignment of mass flow or volumetric flow to the Primary Variable is done at order placement. The variable assignments can be adjusted by the user using the Brooks Expert Support Tool (BEST) software or via HART command 163. See Section 1.

Request data bytes:

NONE

Response data bytes:

SEL. PV UNIT	PV MSB	PV	PV	PV LSB
#0	#1	#2	#3	#4

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Primary variable unit code. Refer to page 71, Flow rate unit and reference codes.
1-4	32-bit floating point, IEEE 754 format	Primary variable: flow rate.

Command #1
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #2
Read Primary Variable Current and Percent Range

Read the primary variable, as current or voltage and as a percent of the primary variable range. For Brooks QUANTIM QMC Series S-protocol devices, the current field reports current in mAmps. The current always matches the analog output unless alarm conditions and set values are present. Percent of range always follows the primary variable, even if the current is in an alarm condition or set to a value. Also, the percent

of range is not limited to values between 0% and 100%, but tracks the primary variable to the sensor limits.

Request data bytes:

NONE

Response data bytes:

CURRENT/ VOLTAGE MSB	CURRENT/ VOLTAGE	CURRENT/ VOLTAGE	CURRENT/ VOLTAGE LSB	PV % RANGE MSB	PV % RANGE
#0	#1	#2	#3	#4	#5

PV % RANGE	PV % RANGE LSB
#6	#7

Data Byte #	Type	Remarks
0-3	32-bit floating point, IEEE 754 format	Analog output current or voltage [milliamperes or volts].
4-7	32-bit floating point, IEEE 754 format	Primary variable: [% of range]

Command #2
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #3
Read Current and All Dynamic Variables

Read the current and the dynamic variables. The current field reports current in mAmps. The current always matches the analog output current of the device including alarm conditions and set values. For the QUANTIM QMC Series S-Protocol devices, the dynamic variable assignments are done at order placement. The variable assignments can be adjusted by the user using the Brooks Expert Support Tool (BEST) software. See Section 1. Note all four dynamic variables are always returned.

Request data bytes:

NONE

Response data bytes:

CURRENT/ VOLTAGE MSB	CURRENT/ VOLTAGE	CURRENT/ VOLTAGE	CURRENT/ VOLTAGE LSB	PRIMARY VAR. UNITS	PRIMARY VAR.
#0	#1	#2	#3	#4	#5

PRIMARY VAR.	PRIMARY VAR.	PRIMARY VAR. LSB	SECOND. VAR. UNITS	SECOND. VAR. MSB	SECOND. VAR
#6	#7	#8	#9	#10	#11

SECOND. VAR	SECOND. VAR LSB	TV UNITS	TV MSB	TV	TV	TV LSB
#12	#13	#14	#15	#16	#17	#18

Data Byte #	Type	Remarks
0-3	32-bit floating point, IEEE 754 format	Analog output current/ voltage [milliamperes or volts].
4	8-bit unsigned integer code	Primary variable unit
5-8	32-bit floating point, IEEE 754 format	Primary variable
9	8-bit unsigned integer	Secondary variable unit code.
10-13	32-bit floating point, IEEE 754 format	Secondary variable.
14	8-bit unsigned integer	Tertiary variable unit
15-18	32-bit floating point IEEE 754 format	Tertiary variable

Command #3
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #6
Write Polling Address

This command writes the Polling Address (Short Frame Address) to the field device.

Request data bytes:

POLLING ADDRESS
#0

Data Byte #	Type	Remarks

0	8-bit unsigned integer	Polling Address: 0-15 16-255 Undefined

Response data bytes:

POLLING ADDRESS
#0

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Polling Address: 0-15 16-255 Undefined

Command #6
Specific Response Codes

0 No command-specific errors
1 Undefined
2 Invalid selection
3-4 Undefined
5 Incorrect bytecount
6 Undefined
7 In write protect mode
8-15 Undefined
16 Access restricted
17-127 Undefined

Command #11
Read Unique Identifier Associated with Tag

This command returns the expanded device-type codes, revision levels and the device identification number of a device containing the requested tag. It will be executed when either the appropriate long address or the broadcast long address, “00000” is received. The address field in the response message of this command always contains the address received in the request message. This command is unique in that no response is made unless the tag matches that of the device.

Request data bytes:

TAG	TAG	TAG	TAG	TAG	TAG
#0	#1	#2	#3	#4	#5

Data Byte #	Type	Remarks
0	6(8-bit) byte packed ASCII	Device tag number

Response data bytes:

254	MFR. ID	MFR's DE-VICE TYPE	NUMBER RQUEST PREAM	UNIV. CMD. REV.	TRANS. SPEC. REV.
#0	#1	#2	#3	#4	#5

SOFTW REV.	HARDW REV.	FLAGS	DEVICE ID NUM MSB	DEVICE ID NUM	DEVICE ID NUM
#6	#7	#8	#9	#10	#11

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Device type code for “expansion”. Contains the code “254” (decimal).
1	8-bit unsigned integer	Manufacturer identification code. (Always 10).
2	8-bit unsigned integer	Manufacturers device type code. Refer to Device type codes, page 68.
3	8-bit unsigned integer	Number of response preamble characters required for the request message from the master to the slave.
4	8-bit unsigned integer	Universal command revision level implemented by this device.
5	8-bit unsigned integer	Transmitter specific command revision level implemented by this device.
6	8-bit unsigned integer	Software revision level of the device.
7	8-bit unsigned integer	Hardware revision level of the electronics in the device.
8	8-bit unsigned integer	Flags. Refer to Flag assignments, page 69.
9-11	24-bit unsigned integer	Device identification number.

Command #11
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #12
Read Message

Read the 32 Character Message String contained within the device. The message string is a 32 character storage area that the user may use for any application related function desired. The message string is not used by the device.

Request data bytes:

NONE

Response data bytes:

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#0	#1	#2	#3	#4	#5

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#6	#7	#8	#9	#10	#11

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#12	#13	#14	#15	#16	#17

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#18	#19	#20	#21	#22	#23

Data Byte #	Type	Remarks
0-23	24(8-bit) byte packed ASCII	32 character message string.

Command #12
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #13
Read Tag, Descriptor, Date

Read the tag, descriptor and date contained within the device. The tag name is used to identify the device (See Command #11). The description and date fields can be utilized for any application specific function desired. The description and date fields are not used by the device.

Request data bytes:

NONE

Response data bytes:

TAG	TAG	TAG	TAG	TAG	TAG
#0	#1	#2	#3	#4	#5

DESCRIPT.	DESCRIPT.	DESCRIPT.	DESCRIPT.	DESCRIPT.	DESCRIPT.
#6	#7	#8	#9	#10	#11

DESCRIPT.	DESCRIPT.	DESCRIPT.	DESCRIPT.	DESCRIPT.	DESCRIPT.
#12	#13	#14	#15	#16	#17

DATE DAY	DATE MONTH	DATE YEAR
#18	#19	#20

Data Byte #	Type	Remarks
0-5	6(8-bit) byte packed ASCII	Device tag name.
6-17	12(8-bit) byte packed ASCII	Device descriptor. (16 character string)
18-20	3(8-bit) byte packed ASCII	Date. Respectively day, month, year - 1900.

Command #13
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect bytecount
6-127 Undefined

Command #14
Read Primary Variable Sensor Information

This command is intended to read primary variable sensor information.

Request data bytes:

NONE

Response data bytes:

SENSOR SERIAL NUM MSB	SENSOR SERIAL NUM	SENSOR SERIAL NUM LSB	LIMITS UNITS CODE	UPPER SENSOR LIMIT MSB	UPPER SENSOR LIMIT
#0	#1	#2	#3	#4	#5

UPPER SENSOR LIMIT	UPPER SENSOR LIMIT LSB	LOWER SENSOR LIMIT MSB	LOWER SENSOR LIMIT	LOWER SENSOR LIMIT	LOWER SENSOR LIMIT LSB
#6	#7	#8	#9	#10	#11

MIN SPAN MSB	MIN SPAN	MIN SPAN	MIN SPAN LSB
#12	#13	#14	#15

Data Byte #	Type	Remarks
0-2	24-bit unsigned integer	Sensor serial number.
3	8-bit unsigned integer	Sensor limits/minimum span unit code.
4-7	32-bit floating point, IEEE 754 format	Upper sensor limit.
8-11	32-bit floating point, IEEE 754 format	Lower sensor limit.
12-15	32-bit floating point, IEEE 754 format	Minimum span.

Command #14
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect bytecount
6-127 Undefined

Command #15
Read Output Information

This command is intended to read the alarm selection code, transfer function, primary variable/range unit code, upper range value, lower range value, damping value (applied to the sensor, not the output), write protect code and private label distributor.

Request data bytes:

NONE

Response data bytes:

ALARM SELECT CODE	TRANSF. FUNCT. CODE	PV / RANGE UNITS CODE	UPPER RANGE MSB	UPPER RANGE	UPPER RANGE
#0	#1	#2	#3	#4	#5

UPPER RANGE MSB	LOWER RANGE MSB	LOWER RANGE	LOWER RANGE	LOWER VALUE MSB	DAMPING VALUE
#6	#7	#8	#9	#10	#11

DAMPING VALUE	DAMPING VALUE	DAMPING VALUE LSB	WRITE PRO-TECT CODE	PVT LABEL DIST
#12	#13	#14	#15	#16

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Alarm select code. Not implemented for the Brooks QUANTIM® QMC Series S-Protocol devices, the integer returned is a “Not-Used” or “250” (decimal).
1	8-bit unsigned integer	Transfer function code. Always returns LINEAR (0)
2	8-bit unsigned integer	Primary variable upper and lower range unit code.
3-6	32-bit floating point, IEEE 754 format	Upper range value.
7-10	32-bit floating point, IEEE 754 format	Lower range value.
11-14	32-bit floating point, IEEE 754 format	Damping value. (Always 0.0)
15	8-bit unsigned integer	Write protect code. Not supported, returns Not Used (250 dec)
16	8-bit unsigned integer	Private label distributor. Returns Hart code for Brooks Instrument (10dec)

Command #15
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect bytecount
6-127 Undefined

Command #16
Read Final Assembly Number

This command is used to read the final assembly number associated with the device.

Request data bytes:

NONE

Response data bytes:

FINAL ASS. NUM MSB	FINAL ASS. NUM	FINAL ASS. NUM LSB
#0	#1	#2

Data Byte #	Type	Remarks
0 - 2	24-bit unsigned integer	Final assembly number.

Command #16
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect bytecount
6-127 Undefined

Command #17
Write Message

Write a 32 Character Message String into the device. See Command #12 for more information about the message string.

Request data bytes:

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#0	#1	#2	#3	#4	#5

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#6	#7	#8	#9	#10	#11

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#12	#13	#14	#15	#16	#17

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#18	#19	#20	#21	#22	#23

Data Byte #	Type	Remarks
0 - 23	24 (8-bit) byte packed ASCII	32 Character message string.

Response data bytes:

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#0	#1	#2	#3	#4	#5

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#6	#7	#8	#9	#10	#11

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#12	#13	#14	#15	#16	#17

MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE	MESSAGE
#18	#19	#20	#21	#22	#23

Data Byte #	Type	Remarks
0 - 23	24 (8-bit) byte packed ASCII	32 Character message string.

Command #17
Specific Response Codes

0	No command-specific errors
1-4	Undefined
5	Incorrect bytecount
6-127	Undefined
7	In write protect mode
8-127	Undefined

Command #18
Write Tag, Descriptor, Date

Write the tag, descriptor and date into the device. See Command #13 for more information.

Request data bytes:

TAG	TAG	TAG	TAG	TAG	TAG
#0	#1	#2	#3	#4	#5

DESCR.	DESCR.	DESCR.	DESCR.	DESCR.	DESCR.
#6	#7	#8	#9	#10	#11

DESCR.	DESCR.	DESCR.	DESCR.	DESCR.	DESCR.
#12	#13	#14	#15	#16	#17

DATE DAY	DATE MONTH	DATE YEAR
#18	#19	#20

Data Byte #	Type	Remarks
0-5	6 (8-bit) byte packed ASCII	Device tag number.
6-17	12 (8-bit) byte packed ASCII	Device descriptor. (16 character string)
18-20	3 (8-bit) unsigned integers	Date. Respectively day, month, year - 1900.

Response data bytes:

TAG	TAG	TAG	TAG	TAG	TAG
#0	#1	#2	#3	#4	#5

DESCR.	DESCR.	DESCR.	DESCR.	DESCR.	DESCR.
#6	#7	#8	#9	#10	#11

DESCR.	DESCR.	DESCR.	DESCR.	DESCR.	DESCR.
#12	#13	#14	#15	#16	#17

DATE DAY	DATE MONTH	DATE YEAR
#18	#19	#20

Data Byte #	Type	Remarks
0-5	6 (8-bit) byte packed ASCII	Device tag number.
6-17	12 (8-bit) byte packed ASCII	Device descriptor. (16 character string)
18-20	3 (8-bit) unsigned integers	Date. Respectively day, month, year - 1900.

Command #18
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect bytecount
6-127 Undefined
7 In write protect mode
8-127 Undefined

Command #19
Write Final Assembly Number

Write the final assembly number into the device.

Request data bytes:

DATE DAY	DATE MONTH	DATE YEAR
#18	#19	#20

Data Byte #	Type	Remarks
0-2	24-bit unsigned integer	Final assembly number.

Response data bytes:

FINAL ASS. NUM MSB	FINAL ASS. NUM	FINAL ASS. NUM LSB
#0	#1	#2

Data Byte #	Type	Remarks
0-2	24-bit unsigned integer	Final assembly number.

Command #19
Specific Response Codes

0	No command-specific errors
1-4	Undefined
5	Incorrect bytecount
6-127	Undefined
7	In write protect mode
8-127	Undefined

Common Practice Command Specifications

This section addresses Common Practice Commands available in the QUANTIM QMC Series S-Protocol devices. These common practice commands are per HCF_SPEC-151, FCG TS20151 Revision 12.0.

Command #33 Read Device Variables

This command allows a Master to request the value of up to four Device Variables. In other words, a Master may request only 1, 2, 3 or 4 Device Variables. Each slot will accept any Device Variable supported by the device. The Field Device must answer these Master requests without returning Response Code 5, Too Few Data Bytes Received. If the Field Device receives 1, 2 or 3 Request Data Bytes it must return only the corresponding number of Device Variables (See Table 7).

Table 7. Command 33 Response Based on Number of Device Variables Requested

No. of Device Variables Requested	No. of Request Data Bytes	No. of Response Data Bytes
1	1	6
2	2	12
3	3	18
4	4	24

Other command requirements include:

- When a Device Variable requested is not supported in the Field Device, then the corresponding Value must be set to “0x7F, 0xA0, 0x00, 0x00”, and the Units Code must be set to “250”, Not Used.
- This command is capable of Burst Mode Operation and is configured with Command 107, Write Burst Mode Device Variables.
- If Command 33 is supported then Command 54 must also be supported.

Request Data Bytes

Byte	Format	Description
0	Unsigned-8	Slot 0: Device Variable Code (see Device Variable Codes Table in appropriate device-specific document)
1	Unsigned-8	Slot 1: Device Variable Code (see Device Variable Codes Table in appropriate device-specific document)
2	Unsigned-8	Slot 2: Device Variable Code (see Device Variable Codes Table in appropriate device-specific document)
3	Unsigned-8	Slot 3: Device Variable Code (see Device Variable Codes Table in appropriate device-specific document)

When a Field Device receives 1, 2, or 3 request data bytes it must answer the Master request without returning Response Code 5, Too Few Data Bytes Received.

Response Data Bytes

Byte	Format	Description
0	Unsigned-8	Slot0: Device Variable Code (see Device Variable Code Table in appropriate device-specific document)
1	Enum	Slot0: Units Code (refer to Common Tables Specification)
2 - 5	Float	Slot0: Device Variable Value
6	Unsigned-8	Slot1: Device Variable Code (see Device Variable Code Table in appropriate device-specific document)
7	Enum	Slot1: Units Code (refer to Common Tables Specification)
8 - 11	Float	Slot1: Device Variable Value
12	Unsigned-8	Slot2: Device Variable Code (see Device Variable Code Table in appropriate device-specific document)
13	Enum	Slot2: Units Code (refer to Common Tables Specification)
14 - 17	Float	Slot2: Device Variable Value
18	Unsigned-8	Slot3: Device Variable Code (see Device Variable Code Table in appropriate device-specific document)
19	Enum	Slot3: Units Code (refer to Common Tables Specification)
20 - 23	Float	Slot3: Device Variable Value

Command #33
Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1		Undefined
2	Error	Invalid Selection[40]
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7		Undefined
8	Warning	Update Failure
9 - 15		Undefined
16	Error	Access Restricted
17 -127		Undefined

Command #35
Write Primary Range Values

Defines the relationship between the Loop Current 4.00 and 20.0mA points and the Primary Variable value. The Upper Range Value of the Primary Variable is independent of the Lower Range Value. Most devices allow the Upper Range Value of the Primary Variable to be lower than its Lower Range Value, enabling the device to be operated with reverse action. The device-specific document will indicate if this capability has not been implemented. The Primary Variable Range Units received with this command do not affect the Primary Variable Units of the device. The range values will be returned in the same units as received. For a transmitter, the Range Values allow the Primary Variable value to be converted to a percent for transmission via the Loop Current. For an actuator, the Range Values allow the Loop Current to be converted to a percent for use by the actuator (e.g., to use as the actuator setpoint).

Request Data Bytes

Byte	Format	Description
0	Unsigned-8	PV Upper and Lower Range Values Units Code (refer to Common Tables Specification)
1 - 4	Float	PV Upper Range Value
5 - 8	Float	PV Lower Range Value

Response Data Bytes

Byte	Format	Description
0	Unsigned-8	PV Upper and Lower Range Values Units Code
1 - 4	Float	PV Upper Range Value
5 - 8	Float	PV Lower Range Value

Note: The value returned in the response data bytes reflects the rounded or truncated value actually used by the device.

Command #35 Response Codes

Code	Class	Description
0	Success	No Command Specific Errors
1		Undefined
2	Error	Invalid Selection
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7	Error	In Write Protect Mode
8	Warning	Set To Nearest Possible Value (Upper or Lower Range Pushed)
9	Error	Lower Range Value Too High
10	Error	Lower Range Value Too Low
11	Error	Upper Range Value Too High
12	Error	Upper Range Value Too Low
13	Error	Upper and Lower Range Values Out Of Limits
14	Warning	Span Too Small (Device Accuracy May Be Impaired)
15		Undefined
16	Error	Access Restricted
17		Undefined
18	Error	Invalid Units Code
19 - 28		Undefined
29	Error	Invalid Span
30 - 31		Undefined
32	Error	Busy
33 - 127		Undefined

Command #37 Set Primary Variable Lower Range Value

This command generates a sensor zero action for the sensor associated with the Primary Variable, the same function as pushing the zero button on the analog device.

When the Primary Variable is flow, user must take action to insure that there is no flow thru the device when this command is used.

When the Primary Variable is pressure, user must take action to insure that pressure at the sensor is 0 psia when this command is used.

The command will return an error response code 9, “Applied process too high,” if flow output is greater than 2% when the command is received.

Request data bytes:

NONE

Response data bytes:

NONE

Command #37
Specific Response Codes

0	No command-specific errors
1-4	Undefined
5	Incorrect bytecount
6	Undefined
7	In write protect mode
8	Undefined
9	Applied process too high
10-127	Undefined

Command #38
Reset Configuration Changed Flag

Resets the configuration changed response code, bit #6 of the transmitter status byte. Secondary master devices, address ‘0’ should not issue this command. Primary master devices, address ‘1’, should only issue this command after the configuration changed response code has been detected and acted upon.

Request data bytes:

NONE

Response data bytes:

NONE

Command #38
Specific Response Codes

0	No command-specific errors
1-4	Undefined
5	Incorrect bytecount
6	Undefined
7	In write protect mode
8-15	Undefined
16	Access restricted
17-127	Undefined

Command #44
Write Primary Variable Units

Selects the units in which the Primary Variable and its range will be returned. This command also selects the units for transducer limits and minimum span.

Request Data Bytes

Byte	Format	Description
0	Enum	PV Units Code (refer to Common Tables Specification)

Response Data Bytes

Byte	Format	Description
0	Enum	PV Units Code (refer to Common Tables Specification)

Note: The value returned in the response data bytes reflects the value actually used by the device.

Command #44
Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1		Undefined
2	Error	Invalid Selection
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7	Error	In Write Protect Mode
8 - 15		Undefined
16	Error	Access Restricted
17 - 31		Undefined
32	Error	Busy
33 - 127		Undefined

Command #48
Read Additional Transmitter Status

This command is used to retrieve additional transmitter status information.

Request data bytes:

NONE

Response data bytes:

ADD. STATUS BYTE#0	ADD. STATUS BYTE#1	ADD. STATUS BYTE#2	ADD. STATUS BYTE#3
#0	#1	#2	#3

Refer to page 68 for a definition of the Additional Status Bytes.

Command #48
Specific Response Codes

0 No command-specific errors
1-4 Undefined
5 Incorrect byte count
6-127 Undefined

Command #54
Read Device Variable Information

Responds with the transducer serial number, the Limits, Damping Value, and Minimum Span of the selected Device Variable along with the corresponding engineering units. The engineering units returned by this command must be the same as the Device Variable's engineering units.

The device must update the Device Variable at least once in the interval indicated by the Acquisition Period.

Request data bytes:

Byte	Format	Description
0	Unsigned-8	Device Variable Code (see Device Variable Codes Table in appropriate device-specific document)

Response data bytes:

Byte	Format	Description
0	Unsigned-8	Device Variable Code (see Device Variable Codes Table in appropriate device-specific document)
1 - 3	Unsigned-24	Device Variable Transducer Serial Number[48]
4	Enum	Device Variable Limit/Minimum Span Units Code (refer to Common Tables Specification)
5 - 8	Float	Device Variable Upper Transducer Limit
9 - 12	Float	Device Variable Lower Transducer Limit
13 - 16	Float	Device Variable Damping Value
17 - 20	Float	Device Variable Minimum Span
21	Enum	Device Variable Classification[49] (see Common Table 31, Device Variable Classification Codes)
22	Enum	Device Variable Family[50] (see Common Table 20, Device Variable Family Codes)
23-26	Time	Acquisition Period. The Acquisition Period indicates the maximum period between Device Variable update[51]
27	Bits	Device Variable Properties (see Common Table 55 Device Variable Property Codes)

Note:

[48] The Transducer Serial Number will be set to zero when it does not apply to the selected Device Variable. The other parameters will be set to "0x7F, 0xA0, 0x00, 0x00" or "250" (Not Used) when they are not applicable.

[49] If the Device Variable Classification is not supported by this Device Variable then the Field Device must return "0" (Not

[50] If the Device Variable Family is not supported by this Device Variable then the Field Device must return "250" (Not Used) and the least significant bits of Device Variable Status must be set to 0 (See the Command Summary Specification). Yet Implemented).

[51] Acquisition Period must return 0xFFFF FFFF if Device Variable is not calculated by the Field Device (i.e., the Device Variable is a setpoint or remote sensor value).

Command #54
Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1		Undefined
2	Error	Invalid Selection
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7 - 15		Undefined
16	Error	Access Restricted
17-31		Undefined
32	Error	Busy
33-127		Undefined

Command #59
Write Number of Response Preambles

Set the minimum number of preambles to be sent by a device before the start of a response packet. This number includes the two preambles contained in the start of message. The value can vary from 2 to 15.

Request data bytes:

#0

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Number of response preambles to be sent with the response message from slave to master.

Response data bytes:

#0

Data Byte #	Type	Remarks
0	8-bit unsigned integer	Number of response preambles to be sent with the response message from slave to master.

Command #59
Specific Response Codes

0	No command-specific errors
1-2	Undefined
3	Passed parameter too large
4	Passed parameter too small
5	Incorrect bytecount
6	Undefined
7	In write protect mode
8-15	Undefined
16	Access restricted
17	Undefined

Command# 60
Read Analog Channel and Percent of Range

Read the Analog Level and Percent of Range of the selected Analog Channel. The Analog Level always matches the associated physical Analog Channel of the device, including alarm conditions and set values. The Analog Level always matches the value that can be measured by an externally connected reference meter.

Percent of Range (Transmitters): Percent of Range always follows the associated Device Variable value, including alarm conditions and set values. The Upper and Lower Range Values maps the Dynamic Variable value to the Percent of Range. Percent of Range is not limited to values between 0% and 100%, but tracks the Device Variable to the Transducer Limits. When the Device Variable value reaches the Upper (100%) or Lower (0%) Range Value the Analog Channel Level must correspond to the Upper or Lower Endpoint signal level respectively.

Percent of Range (Actuators): Percent of Range always follows the Analog Level even if is set to a value. The Upper and Lower Range Values maps the Analog Level to the Percent of Range. As a result the Percent of Range is not limited to values between 0% and 100%, but tracks the Analog Level to Transducer Limits when they are defined.

Request data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)

Response data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Analog Channel Level
6 - 9	Float	Analog Channel Percent of Range (units of percent)

Command #60
Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1		Undefined
2	Error	Invalid Selection
3 - 4		Undefined
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7		Undefined
8	Warning	Update Failure
9 - 15		Undefined
16	Error	Access Restricted
17 -127		Undefined

Command #66
Enter/Exit Fixed Analog Channel Mode

The device's Analog Channel level is fixed to the value received. The value returned in the response data bytes reflects the rounded or truncated value actually used by the device. A level containing "0x7F, 0xA0, 0x00, 0x00", with any Units Code exits the Fixed Analog Channel Mode. Fixed Analog Channel Mode is also exited when power is removed from device or upon performing a Device Reset.

Request data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Fixed Analog Channel Level

Response data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Fixed Analog Channel Level

Note: The value returned in the response data bytes reflects the rounded or truncated value actually used by the device.

Command #66
Response Codes

Code	Class	Description
0	Success	No Command Specific Errors
1 - 2		Undefined
3	Error	Passed Parameter Too Large
4	Error	Passed Parameter Too Small
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7	Error	In Write Protect Mode
8 - 10		Undefined
11	Error	In Multidrop Mode
12	Error	Invalid Units Code
13 - 14		Undefined
15	Error	Invalid Analog Channel Code Number
16	Error	Access Restricted
17 - 31		Undefined
32	Error	Busy
33 - 127		Undefined

Command #67
Trim Analog Channel Zero

This command trims the zero or lower endpoint value of the selected Analog Channel so that the Analog Channel value matches the connected meter reading. The value sent with the command may be rounded or truncated by the device. The Response Data Bytes contain the value from the request as used by the device.

Use Command 66, Enter/Exit Fixed Analog Channel Mode, to set the

Analog Channel exactly to the lower endpoint value before using this command. Response Code 9, Not in Proper Analog Channel Mode, will be returned if the Fixed Analog Channel Mode has not been entered or the Analog Channel is not set exactly to the lower endpoint value.

Request data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Externally Measured Analog Channel Level

Response data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Actual Analog Channel Level

Note: The value returned in the response data bytes reflects the rounded or truncated value actually used by the device.

Command #67
Response Codes

Code	Class	Description
0	Success	No Command Specific Errors
1 - 2		Undefined
3	Error	Passed Parameter Too Large
4	Error	Passed Parameter Too Small
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7	Error	In Write Protect Mode
8		Undefined
9	Error	Not In Proper Analog Channel Mode
10		Undefined
11	Error	In Multidrop Mode
12	Error	Invalid Units Code
13 - 14		Undefined
15	Error	Invalid Analog Channel Code Number
16	Error	Access Restricted
17 - 31		Undefined
32	Error	Busy
33 - 127		Undefined

Command #68
Trim Analog Channel Gain

This command trims the gain or upper endpoint value of the selected Analog Channel so that the Analog Channel value matches the connected meter reading. The value that is sent with the command may be rounded or truncated by the device. The response data bytes contain the value from the request as used by the device.

Use Command 66, Enter/Exit Fixed Analog Channel Mode, to Set the Analog Channel exactly to the upper endpoint value before using this command. Response Code 9, Not In Proper Analog Channel Mode, will be returned if the Fixed Analog Channel Mode has not been entered or

the Analog Channel is not set exactly to the upper endpoint value.

Request data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Externally Measured Analog Channel Level

Response data bytes:

Byte	Format	Description
0	Unsigned-8	Analog Channel Number Code (see Analog Channel Number Codes Table in appropriate device-specific document)
1	Enum	Analog Channel Units Code (refer to Common Tables Specification)
2 - 5	Float	Externally Measured Analog Channel Level

Note: The value returned in the response data bytes reflects the rounded or truncated value actually used by the device.

Command #68
Response Codes

Code	Class	Description
0	Success	No Command Specific Errors
1 - 2		Undefined
3	Error	Passed Parameter Too Large
4	Error	Passed Parameter Too Small
5	Error	Too Few Data Bytes Received
6	Error	Device-Specific Command Error
7	Error	In Write Protect Mode
8		Undefined
9	Error	Not In Proper Analog Channel Mode
10		Undefined
11	Error	In Multidrop Mode
12	Error	Invalid Units Code
13 - 14		Undefined
15	Error	Invalid Analog Channel Code Number
16	Error	Access Restricted
17 - 31		Undefined
32	Error	Busy
33 - 127		Undefined

Command #110
Read All Dynamic Variables

Read up to four predefined Dynamic Variables. The Secondary, Tertiary, and Quaternary Variables are defined by each device type.

Request data bytes:

Byte	Format	Description
None		

Response data bytes:

Byte	Format	Description
0	Enum	Primary Variable Units Code (refer to Common Tables Specification)
1-4	Float	Primary Variable
5	Enum	Secondary Variable Units Code (refer to Common Tables Specification)
6-9	Float	Secondary Variable
10	Enum	Tertiary Variable Units Code (refer to Common Tables Specification)
11-14	Float	Tertiary Variable
15	Enum	Quaternary Variable Units Code (refer to Common Tables Specification)
16-19	Float	Quaternary Variable

Note: Response Data Bytes are truncates after last variable supported by each device type.

Command #110
Response Codes

Code	Class	Description
0	Success	No Command-Specific Errors
1 - 5		Undefined
6	Error	Device-Specific Command Error
8	Warning	Update Failure
9-15		Undefined
16	Error	Access Restricted
17- 27		Undefined

Device/Transmitter Specific Command Specifications

This section addresses Device Specific Commands available in the QUANTIM QMC Series S-Protocol devices.

Command #131 Read Brooks Serial Number

Read the Brooks order number from the device's memory. The Brooks order number is a 24-byte packed ASCII string (resulting in 32 total unpacked ASCII characters) indicating the serial number of the device. The number can be used for traceability purposes.

Request data bytes:

None

Response data bytes:

BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER
#0	#1	#2	#3	#4	#5

BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER
#6	#7	#8	#9	#10	#11

BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER
#12	#13	#14	#15	#16	#17

BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER	BROOKS SERIAL NUMBER
#18	#19	#20	#21	#22	#23

Data Byte #	Type	Remarks
0-23	24 (8-bit) packed ASCII	Brooks Serial Number

Command #131 Specific Response Codes

0 No command-specific errors
1-127 Undefined

Command #132
Read Model Number

Read the device Model number from the device's memory. The device Model number is a 24-byte packed ASCII string (resulting in 32 total unpacked ASCII characters).

Request data bytes:

None

Response data bytes:

DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER
#0	#1	#2	#3	#4	#5

DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER
#6	#7	#8	#9	#10	#11

DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER
#12	#13	#14	#15	#16	#17

DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER	DEVICEMODEL NUMBER
#18	#19	#20	#21	#22	#23

Data Byte #	Type	Remarks
0-23	24 (8-bit) packed ASCII	Brooks Serial Number

Command #132
Specific Response Codes

0 No command-specific errors
1-127 Undefined

Command #134
Read Software Revisions

Read the software revision from the device as an ASCII string of up to 8 characters. If the firmware revision string is less than 8 characters, the remaining bytes wil be 0.

Request data bytes:

None

Response data bytes:

FIRMWARE REVISION	FIRMWARE REVISION	FIRMWARE REVISION	FIRMWARE REVISION	FIRMWARE REVISION	FIRMWARE REVISION
#0	#1	#2	#3	#4	#5

FIRMWARE REVISION	FIRMWARE REVISION
#6	#7

Data Byte #	Type	Remarks
0-7	8 (8-bit) ASCII text	Firmware revision

Command #134
Specific Response Codes

0 No command-specific errors
1-127 Undefined

Command #135
Read Sensor Software Rev

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
6	Packed Hart String	Unpacked Format xx.yy.zz

Command #136
Read Bootloader Rev

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
6	Packed Hart String	Unpacked Format xx.yy.zz

Command #141
Perform Mass Flow Zero

Request data bytes:

NONE

Response data bytes:

NONE

Command #142
Perform Reset

Request data bytes:

Data Byte #	Type	Remarks
0	Reset Code	1-Soft Reset, 2 Hard Reset

Response data bytes:

NONE

Command #150
Read Totalizer

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Floating Point	Totalizer Value

Command #151
Clear Totalizers

Request data bytes:

NONE

Response data bytes:

NONE

Command #152
Read On Time

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
15	Packed Hart String	Unpacked Format xxxyxxxdxxhxxmxxs

Command #153
Read Total On Time

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
15	Packed Hart String	Unpacked Format xxxyxxxdxxhxxmxxs

Command #154
Read Calibration Timeout

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0-3		Long as Seconds

Command #155
Write Calibration Timeout

Request data bytes:

Data Byte #	Type	Remarks
0-3		Long as Seconds

Response data bytes:

NONE

Command #156
Write Calibration Performed

Request data bytes:

NONE

Response data bytes:

NONE

Command #157
Read Overhaul Timeout

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0-3		Long as Seconds

Command #158
Write Overhaul Timeout

Request data bytes:

Data Byte #	Type	Remarks
0-3		Long as Seconds

Response data bytes:

NONE

Command #159
Write Overhaul Performed

Request data bytes:

NONE

Response data bytes:

NONE

Command #160
Read Hart Dynamic Variable

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	Value

Command #161
Write Hart Dynamic Variable Units

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable
1		Units

Response data bytes:

NONE

Command #162
Read Dynamic Hart Variable Assignment

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable

Response data bytes:

Data Byte #	Type	Remarks
0	Device	Variable

Command #163
Write Dynamic Hart Variable Assignment

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable
1	Device	Variable

Response data bytes:

NONE

Command #170
Read Control Mode

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0		Control Mode

Command #171
Write Control Mode

Request data bytes:

Data Byte #	Type	Remarks
0		Control Mode

Response data bytes:

NONE

Command #172
Read Setpoint and Percent of Range

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	Value
5-8	Float	Percent Value

Command #173
Write Setpoint

Request data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	Value

Response data bytes:

NONE

Command #176
Read Valve Override and Valve Drive

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0		Override state
1-4	Float	Valve Percent of drive. This does not directly correlate to valve position!

Command #177
Write Valve Override

Request data bytes:

Data Byte #	Type	Remarks
0		Override state

Response data bytes:

NONE

Command #178
Read Setpoint Current

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0-3	Float	Current in mA

Command #180
Read Controller Values

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0-3	Float	PGain
4-7	Float	IGain
8-11	Float	IGain2
12-15	Float	Dgain
16-19	Float	Dgain2

Command #181
Write Controller Values

Request data bytes:

Data Byte #	Type	Remarks
0-3	Float	PGain
4-7	Float	IGain
8-11	Float	IGain2
12-15	Float	Dgain
16-19	Float	Dgain2

Response data bytes:

NONE

Command #190
Read Error Code

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
1	Fail	EXT_FLASH_FAIL
1	Fail	INTERNAL_ERROR
2	Fail	TEMP_SENS_FAIL
3	Error	FLOW_SENS_FAIL
4	Fail	FLOW_UPDATE_FAIL
5	Error	CONFIG
6	Fail	NETWORK_FAIL
11	Error	ATTR_DATA_LOST

12	Error	ZEROING
15	Error	FLOW_ALARM_LO
15	Error	FLOW_ALARM_HI
15	Warn	FLOW_WARN_LO
15	Warn	FLOW_WARN_HI
16	Error	DENSITY_ALARM_LO
16	Error	DENSITY_ALARM_HI
16	Warn	DENSITY_WARN_LO
16	Warn	DENSITY_WARN_HI
17	Error	SLUG_ALARM
18	Error	TEMP_ALARM_LO
18	Error	TEMP_ALARM_HI
18	Warn	TEMP_WARN_LO
18	Warn	TEMP_WARN_HI
19	Error	CNTRL_ALARM
19	Warn	CNTRL_WARN
21	Warn	CALIBRATE_TIME
22	Warn	OVERHAUL_TIME
24	Warn	VOLTAGE_LEVEL_LO
24	Warn	VOLTAGE_LEVEL_HI
25	Warn	WARMUP

Command #192 Read Event Status Words

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0-3	Long Word1	For Service
4-7	Long Word2	For Service

Command #200 Read Analog Output Alarm Behavior

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable

Response data bytes:

NONE

Data Byte #	Type	Remarks
0		Behavior

Command #201
Write Analog Output Alarm Behavior

Request data bytes:

Data Byte #	Type	Remarks
0	Dynamic	Variable
1		Behavior

Response data bytes:

NONE

Command #202
Read Flow Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	High Value
5-8	Float	Low Value
9-12	Float	Hysteresis
13		Contact Action

Command #203
Write Flow Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm
1		Unit
2-5	Float	High Value
6-9	Float	Low Value
10-13	Float	Hysteresis
15		Contact Action

Response data bytes:

NONE

Command #204
Read Density Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	High Value
5-8	Float	Low Value
9-12	Float	Hysteresis
13		Contact Action

Command #205
Write Density Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm
1		Unit
2-5	Float	High Value
6-9	Float	Low Value
10-13	Float	Hysteresis
15		Contact Action

Response data bytes:

NONE

Command #206
Read Temperature Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm

Response data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm
1-4	Float	High Value
5-8	Float	Low Value
9-12	Float	Hysteresis
13		Contact Action

Command #207
Write Temperature Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm
1		Unit
2-5	Float	High Value
6-9	Float	Low Value
10-13	Float	Hysteresis
15		Contact Action

Response data bytes:

NONE

Command #208
Read Slug Alarm Values

Request data bytes:

NONE

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	High Value
5-8	Float	Low Value
9-12	Float	Hysteresis
13		Contact Action

Command #209
Write Slug Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	High Value
5-8	Float	Low Value
9-12	Float	Hysteresis
13		Contact Action

Response data bytes:

NONE

Command #210
Read Control Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm

Response data bytes:

Data Byte #	Type	Remarks
0		Units
1-4	Float	Error Band
5-8	Float	Hysteresis
9	Float	Contact Action

Command #211
Write Control Alarm Values

Request data bytes:

Data Byte #	Type	Remarks
0	Severity	0-Warn, 1-Alarm
1		Unit
2-5	Float	Error Band
6-9	Float	Hysteresis
10		Contact Action

Response data bytes:

NONE

Transmitter Specific Tables

This Section lists all transmitter specific codes as used by the Brooks QUANTIM QMC Series S-Protocol devices. The codes are commonly 8-bit unsigned integers, ranging from 0 to 255. In a number of cases these code tables are subsets of existing “Common Tables” provided by the HART communication specification.

Device Type Codes

The Device type code for all Brooks QUANTIM QMC Series S-Protocol devices is 4.

Flow Rate Unit and Reference Codes

The flow rate unit codes are covered by two tables: one for mass/ gravimetric flow units and one for volumetric flow units.

Code	Flow Rate Unit (Mass)
70	g/sec
71	g/min
72	g/hour (Native device variable units)
73	kg/sec
74	kg/min
75	kg/hour
76	kg/day
80	lb/sec
81	lb/min
82	lb/hour
83	lb/day

Code	Flow Rate Unit (Volume)
15	cubic feet/min
17	liters/min
19	cubic meters/hr
30	imp gal/hr
130	cubic feet/hr
131	cubic meters/min
136	gal/hr
138	liters/hr
240	cc/hr
241	cc/min
242	ml/hr (Native device variable units)
243	ml/min

Density Unit Codes

The density units are always referenced at 273.15 Kelvin and 1013.33 mBar (‘normal’ conditions).

Code	Density Unit
91	Grams/cubic centimetre
92	Kilograms/cubic meters
93	Pounds/Gallons
94	Pounds/cubic feet
95	Grams/milliliter
96	Kilograms/litre

Temperature Unit Codes

Code	Temperature Unit
32	Degrees Celsius
33	Degrees Fahrenheit
35	Kelvin

Write Protect Codes

Write Protect Codes

Code	Material
0	Not write protected
2..249	Undefined
250..255	Reserved

Physical Signalling Codes

The physical signalling codes indicate the physical layer that can be used for communication.

Physical Signalling Codes

Code	Physical Signalling Code
0	RS485
1..249	Undefined
250..255	Reserved

Device Variable Codes

Definition of the transmitter variable codes.

Transmitter Variable Codes

Code	Variable
1	Mass Flow
2	Density
3	Volumetric Flow
4	Temperature
5	Valve

6	Setpoint

Flag Assignments

The flag assignments indicate implementation facts of the device.

Flag Assignments

Bit	Indication
#0	Multisensor device
#1	Undefined
#2	Undefined
#3	Undefined
#4	Undefined
#5	Undefined
#6	Undefined
#7	Reserved

Valve Override Codes

Note: These codes are all ‘Undefined’ for the meter models.

Valve Override Codes

Code	Valve Override Selection
0	Valve override off (normal opera-tion)
1	Valve override close
2	Valve override open
3	Hold last valve

Totalizer Unit Codes

Totalizer Unit Codes

Code	Totalizer Unit
40	Gallons
42	Imp. Gallons
43	Cubic Meters
60	Grams
61	KiloGrams
63	Pounds
112	Cubic Feet
167	Normal Liters
168	Standard Cubic Feet
241	Cubic Centimeters

LIMITED WARRANTY

Visit www.BrooksInstrument.com for the terms and conditions of our limited warranty.

BROOKS SERVICE AND SUPPORT

Brooks is committed to assuring all of our customers receive the ideal flow solution for their application, along with outstanding service and support to back it up. We operate first class repair facilities located around the world to provide rapid response and support. Each location utilizes primary standard calibration equipment to ensure accuracy and reliability for repairs and recalibration and is certified by our local Weights and Measures Authorities and traceable to the relevant International Standards.

Visit www.BrooksInstrument.com to locate the service location nearest to you.

START-UP SERVICE AND IN-SITU CALIBRATION

Brooks Instrument can provide start-up service prior to operation when required.
For some process applications, where ISO-9001 Quality Certification is important, it is mandatory to verify and/or (re)calibrate the products periodically. In many cases this service can be provided under in-situ conditions, and the results will be traceable to the relevant international quality standards.

SEMINARS AND TRAINING

Brooks Instrument can provide seminars and dedicated training to engineers, end users and maintenance persons.

Please contact your nearest sales representative for more details.

Due to Brooks Instrument's commitment to continuous improvement of our products, all specifications are subject to change without notice.

X-DPT-RS485-QMC-eng/541B189AAG/2022-12

Global Headquarters
Brooks Instrument
407 West Vine Street
Hatfield, PA
19440-0903 USA

Toll-Free (USA): 888-554-FLOW
T: 215-362-3500
BrooksAM@BrooksInstrument.com

A list of all Brooks Instrument locations and contact details can be found at www.BrooksInstrument.com

BROOKS® INSTRUMENT
Beyond Measure