Installation and Operation Manual
X-DPT-DeviceNet-Digital-MFC-eng
Part Number 541B066AAG
July, 2010

DeviceNet™ Supplemental Manual for Brooks Instrument Delta Class MFC's/MFM's

Brooks® Delta Series MFC/MFM's

BROOKS® INSTRUMENT

Brooks® Delta Class DeviceNet MFC's/MFM's

Dear Customer,

We appreciate this opportunity to service your flow measurement and control requirements with a Brooks Instrument device. Every day, flow customers all over the world turn to Brooks Instrument for solutions to their gas and liquid low-flow applications. Brooks provides an array of flow measurement and control products for various industries from biopharmaceuticals, oil and gas, fuel cell research and chemicals, to medical devices, analytical instrumentation, semiconductor manufacturing, and more.

The Brooks product you have just received is of the highest quality available, offering superior performance, reliability and value to the user. It is designed with the ever changing process conditions, accuracy requirements and hostile process environments in mind to provide you with a lifetime of dependable service.

We recommend that you read this manual in its entirety. Should you require any additional information concerning Brooks products and services, please contact your local Brooks Sales and Service Office listed on the back cover of this manual or visit www.BrooksInstrument.com

Yours sincerely,
Brooks Instrument

Figures

Tables

THIS PAGE WAS INTENTIONALLY LEFT BLANK

1 Introduction

This manual is a supplement to the Delta Series Mass Flow Controller Operation Manual. It is assumed that the owner of this Delta Series MFC is thoroughly familiar with the theory and operation of a Delta Series MFC/MFM. If not, it is recommended that the owner read the Operations Manual first before continuing with this supplement.

Many applications of Flow Meters/Controllers and Pressure devices are moving to increasing use of automation. Automation comes in many forms: PLC's (Programmable Logic Controllers such as Allen-Bradley's SLC500), DCS's (Distributed Control Systems, such as Emerson's Delta V), and
PC-based solutions (National Instrument's Labview™). Digital communications from these varied automation systems and the devices they measure and control are a very effective means of not only accomplishing more effective and rapid system integration, but also providing greatly improved system diagnostics and maintainability. DeviceNet™ is an open digital protocol capable of high speeds and easy system connectivity. Brooks Instrument has several of its devices available on this popular networking standard, and is a member of ODVA™(Open DeviceNet Vendors Association), the governing standard body for DeviceNet.

THIS PAGE WAS INTENTIONALLY LEFT BLANK

2.1 Background & Assumptions

This manual assumes a basic knowledge and understanding of DeviceNet (it's topology and its method of logically accessing the data or parameters contained within a device). This manual also assumes basic knowledge and understanding regarding the operation of Mass Flow Controllers or Mass Flow Meters. This manual is not intended to be a replacement to the ODVA™ (Open DeviceNet Vendors Association) specification, which is still the authoritative definition and description of DeviceNet communications. It is recommended, but not required for the purposes of this manual, that the user obtain a copy of the DeviceNet specification from ODVA™ (www.odva.org).

This manual does not make any assumptions about any particular manufacturer of equipment or custom software used by the user to communicate with the Brooks device, but assumes the user has thorough understanding of such equipment and any configuration software. Application Notes and FAQ's are available at the Brooks Instrument web site (www.brooksintrument.com), which may contain specific information about the interfacing of other vendors equipment and the Delta Series Mass Flow Controller or Meter.

2.2 Notations

This section details notations and conventions used throughout the manual. It is recommended that the reader become very familiar with these conventions. Hypertext links are used in the manual to assist in navigating. This manual is best viewed in its electronic form and can be obtained from the Brooks Customer Service Department (215) 362-3700 in Adobe® Acrobat® PDF format. A glossary is provided for reference in Section: 8 Glossary to aid in reviewing and/or to define any unfamiliar terms.

Numbers

Numeric values used throughout this manual will be clearly denoted as to the base numeric system it represents. All hexadecimal number (base 16) will be prefixed with a 0x, like 0xA4. All binary numbers (base 2) will be suffixed with a b, like 1001b. All other numbers not annotated this way will be assumed decimal (base 10).

EPATHs

EPATH's will be donated within brackets for example [0x31, 1, 3], which represents, left to right, the Class ID (hexadecimal), Instance ID (decimal) and Attribute ID (decimal).

THIS PAGE WAS INTENTIONALLY LEFT BLANK

3 Quick Start

This section assumes the owner of the Delta Series device has a fully operational and trouble-free communications network with appropriate power supplies as defined in the DeviceNet specification. This section also assumes that there exists, connected to the network, a "master" type of device with Poll I/O scanning capability (generally this is the most common application for a DeviceNet network).

3.1 Step 1: Set Baud Rate and MAC ID

Two sets of switches are provided for setting the communication baud rate and for setting the MAC ID of the device. The default MAC-ID is 63. The default Baud Rate is 125K. Set these switches on the MFC(s) to the desired values.

3.1.1 Baud Rate

The switch labeled "RATE" sets the baud rate of the MFC. Possible values along with their corresponding label are (see figure below): "1" = 125K baud, "2" = 250K baud, and "5" = 500K baud, "P" = Software programmable where DeviceNet communications may be used to set the baud rate to one of the above values. The out-of-box default setting is 125K baud.

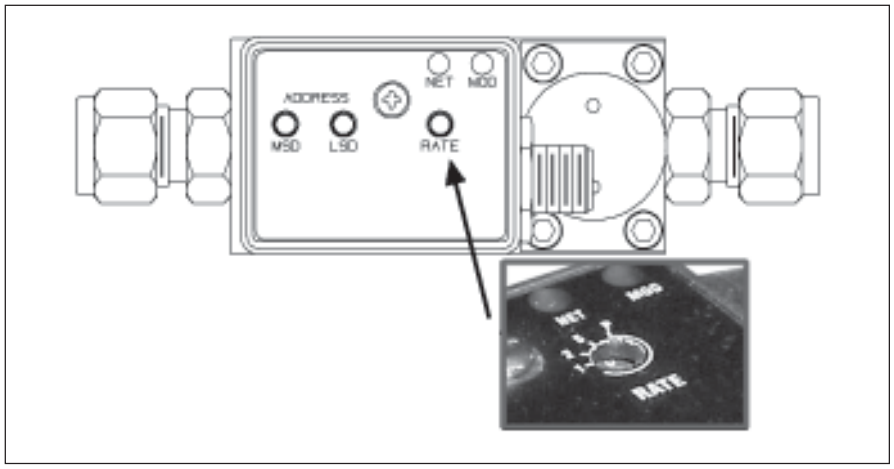


Figure 3-1 Baud Rate Switch

3.1.2 MAC ID

Two switches labeled with "ADDRESS" are used to configure the MAC ID of the device. MAC ID stands for Media Access Control Identifier and is used to set the unique address of the device on the network. The possible range of addresses is 00 to 63.

The switch labeled "MSD" is used to set the most significant digit of the address: 00, 10, 20, 30…etc. all the way up to 60. If "MSD" is set in the range labeled as "P", it means the MAC ID of the device may be software programmable through the network.

The switch labeled "LSD' sets the least significant digit of the MAC ID. The out-of-box default setting for the MAC ID is 63.

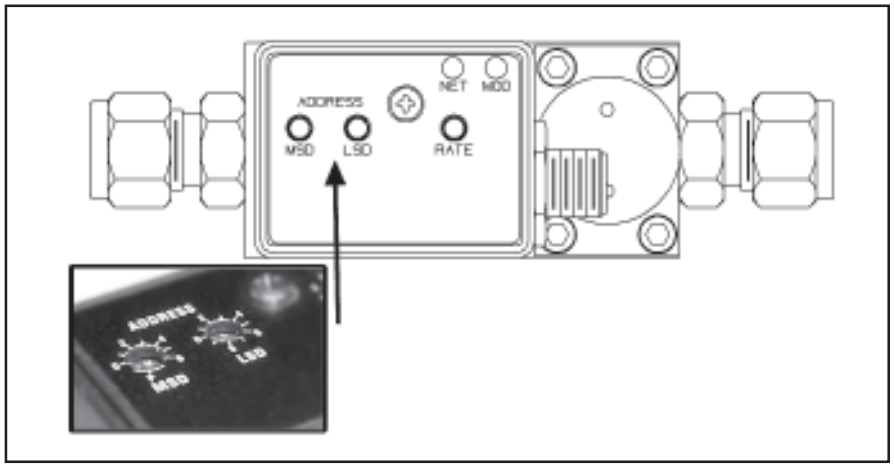


Figure 3-2 MAC ID Switches

Note:

Any changes made to the switches, either by externally setting them or through software, will not take effect until the device has been power cycled.

3.2 Step 2: Configure Scanner

This section contains information that will be needed to configure the scanning device (e.g. PLC, PC, DCS). You will need to consult the documentation for these systems for proper configuration to the device configuration information below.

The Delta Series MFC/MFM comes from the factory with a default configuration as defined by the MFC/MFM Device Profile. The table below lists attributes that are of the most interest to owners of this type of device and the factory configured default values. For more information on all the supported attributes in the device, see Section 5 Detail Configuration.

Note:

If you ordered your Delta Series MFC/MFM and requested that the Factory pre-configure the device per your own custom specification, the following tables may not apply.

Table 3-1 I/O Data Description Table (default):

	Input Assembly	Output Assembly	
Assembly Instance	2	7	
Data Size	3 bytes	2 bytes	
Assembly Type	Input	Output	
Definition of Bytes			
Byte 0	**Status***	Byte 0	**Setpoint*****
Byte 1	**Flow****	Byte 1	
Byte 2			

***Status:**
An 8-bit Bitfield whose bits have the following definition

Bit	Description
0	Common Alarms
1	Device Profile Specific Alarms
2	Manufacturer Specific Alarms
3	0
4	Common Warnings
5	Device Profile Specific Warnings
6	Manufacturer Specific Warnings
7	1

****Flow:**
A 16-bit signed integer (-32768 to 32767) to represent flow in data units of counts[1]. The numeric range definitions for Flow are:

*****Setpoint:**
A 16-bit signed integer (-32768 to 32767) to represent the setpoint value in data units of counts[1]. The numeric range definitions for Setpoint are:

Table 3-2 Counts Range Definition

Counts Range	< 0	0 - 23405	23406 - 25745	25746 - 32767
Flow	Reverse Flow	0% to 100% of Full Scale	100+% to 110% of Full Scale	110+% to 140% of Full Scale
Counts Range	< 0	0 – 23405	23406 - 25745	25746 - 32767
Setpoint	Not Valid. The minimum value for setpoint will clamped to 0% of Full Scale	0% to 100%of Full Scale	100+% to 110% of Full Scale	Not Valid. The maximum value for setpoint will be clamped to 110% of Full Scale

[1] The Data Units Counts is a dimensionless unit that defines a range of numbers to represent a quantity. See Configuration Section for more information on Data Units counts.

3.3 Step 3: Plumb to the Device

Ensure gas flow paths are provided to the inlet and outlet connectors of the MFC. Ensure pressures are per the specifications stated on the purchase order for the device (or the calibration data sheet shipped with the device).

3.4 Step 4: Connect Device to the Network

Connect the device to the network by attaching the network cable to the DeviceNet Micro connector on top of the device.
If power is already been applied to the network, go to Step 6.

3.4.1 Interface Connector

Power and network signals are interfaced to the MFC through the standard 5-pin "micro" connector on the device. This connector is specified in the DeviceNet Specifications and is defined as a male, sealed, micro-style connector. The figure below illustrates the electrical connections to the device.

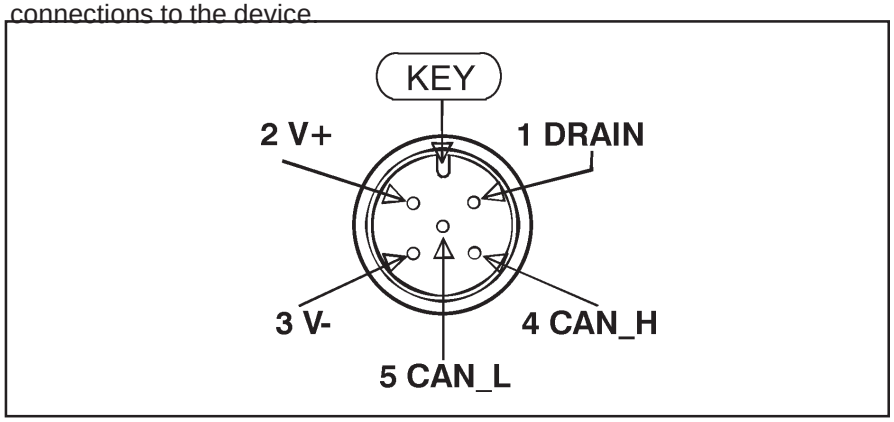


Figure 3-3 Interface Connector

3.5 Step 5: Apply Power to the Network

A DeviceNet network is powered by a (nominal) +24 VDC supply. Ensure that your network is properly powered per the ODVA™ specifications.

3.6 Step 6: LED Verification

Two LED's are provided to indicate network status and module status, labeled "NET" and "MOD" respectively. Appendix D provides a table of flash codes for each of the LED's

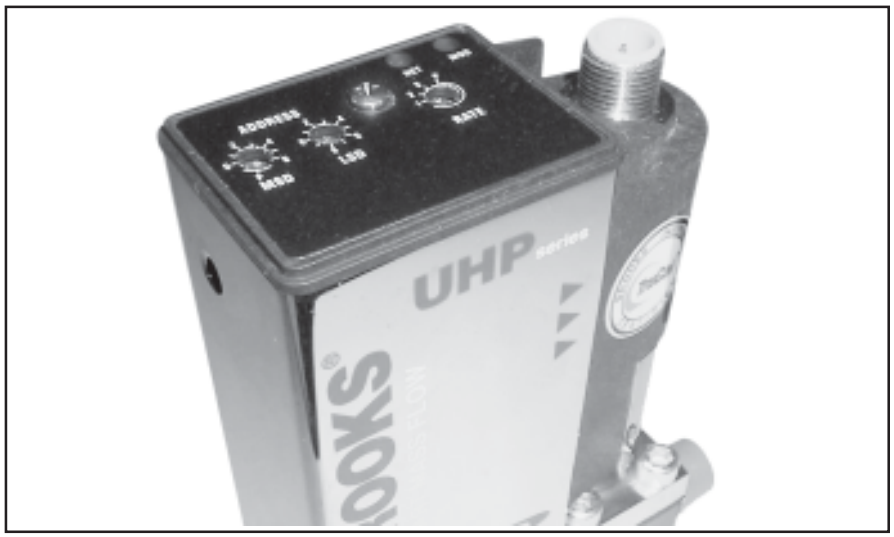


Figure 3-4 View Showing Top of MFC/MFM Can.

When power is applied to the device, each LED will flash red, then green a minimum of one time. This is a LED verification test. If this does not occur, go to the Troubleshooting section.

Next the MOD LED ("module" Status) will then begin to flash red/green indicating that the device is self-testing. When all internal tests have passed, the MOD LED will turn solid green. If this does not occur, go to the Troubleshooting section.

The NET LED ("Network" Status) should flash green, indicating that the device MAC ID and Baud Rate are configured properly and the device recognizes the network. It is now capable of establishing "connections." Proceed to Step 7.

If the NET LED is solid green, an active connection has now been established to the device. Proceed to Step 7.

If the NET LED is indicating any other status than those described above, consult Appendix D for LED flash codes or go to the Troubleshooting section.

3.7 Step 7: Begin Scanning the Network

Enable the scanner to begin scanning the network. If the NET LED on the MFC goes to solid green, the scanner has connected to the MFC and is now actively polling the MFC. Each poll request to the MFC contains a setpoint value, and each poll response from the MFC contains Status and Flow information.

If the NET LED does not maintain a solid green status while the scanner is actively scanning the network, consult Appendix D for LED flash codes or go to the Troubleshooting section.

4.1 Commonly Configured Attributes

The Delta Class MFC/MFM supports many different configurable attributes. The out-of-box defaults meet the needs of a great majority of applications, but some applications may require the device to report more information or behave differently than is configured with default settings, such as valve position, safe mode, or flow and/or setpoint as real numbers in engineering units etc.

This section covers the more common attributes that are configured to meet the unique needs of applications. The terms "attribute" and "parameters" can be used interchangeably and ultimately refer to the same data item within the MFC device. The term "parameter" is widely used within the EDS paradigm whereas "attribute" is used within the DeviceNet specification.

The tables will make reference to both the EDS Parameter name (if the configuration software utilizes the EDS sheet) and the EPATH descriptor (class-instance-attribute) for those who are writing custom or have other types of configuration interfaces.

The following table lists attributes/parameters that will be discussed in this section.

Table 4-1 Commonly Configured Attributes/Parameters:

Attribute	EDS Parameter	EPATH	Default Value	Semantics
Default Polled I/O Conn. Produce Assembly Instance	Poll Prod Assy	[0x5-1-100]	2	Uses Assembly Instance 2 to Produce Data onto the network
Default Polled I/O Consume Assembly Instance	Poll Cons Assy	[0x5-1-101]	7	Uses Assembly Instance 7 to Consume Data from the network
Sensor Data Type	Flow Sensor Data Type	[0x31-1-3]	0xC3	Flow will be reported as a 16-bit signed integer
Actuator Data Type	Valve Data Type	[0x32-1-3]	0xC3	Valve position will be reported as a 16-bit signed integer
Setpoint Data Units	Flow Control Data Type	[0x33-1-3]	0xC3	Setpoint will be a 16-bit signed integer.
Sensor Data Units	Flow Sensor Data Units	[0x31-1-4]	0x1001	See Counts Description See Counts Range Values
Actuator Data Units	Valve Data Units	[0x32-1-4]	0x1001	See Counts Description See Counts Range Values
Setpoint Data Units	Flow Control Data Units	[0x33-1-4]	0x1001	See Counts Description See Counts Range Values
Sensor Safe State	Flow Sensor Safe State	[0x31-1-35]	0	The sensor will report a value of 0 when device is in its Safe State
Actuator Safe State	Valve Safe State	[0x32-1-21]	0	The valve will close when device is in its Safe State
Selected Gas Calibration	Flow Sensor Gas Cal Instance	[0x31-1-35]	1	The instance of the Gas Calibration Object used to linearize the Flow Sensor

4.1.1 I/O Assemblies

Assemblies are a collection of attributes and are an integral part of the I/O data exchange. It is the Assembly that defines what data and how much data is exchanged between the Master and the device in an I/O type of connection. Assemblies are grouped into two categories: Input and Output

From a device perspective, an Input Assembly is the data PRODUCED from the device ONTO the network (or can also be viewed as data "INPUT" to the Master FROM the device). Again, from a device perspective, an Output Assembly is the data CONSUMED from the network INTO the device (or can also be viewed as the data "OUTPUT" from the Master TO the device).

A selection of pre-defined Assemblies is listed in Appendix A. Once the desired Assemblies have been configured in the device, corresponding changes in configuration will probably be required from the Master side of connection.

Note:

For I/O Connections to be established between Master and Slave device, the I/O data sizes for both Input/Output must match. Selection of an Assembly in the MFC device automatically sets the I/O data sizes and no further configuration by the customer is required.

The table below references the parameters/attributes associated with I/O Assembly configuration.

Table 4-2 Attributes to Configure Default Assemblies

EDS Parameter	EPATH	Configures Connection Type	Valid Assembly Type	Data Size (Bytes)	Out-of Box Default
Poll Prod Assy	[0x5-1-100]	Polled	Input	3	2
Poll Cons Assy	[0x5-1-101]	Polled	Output	2	7
Not Supported	[0x5-1-102]	Cyclic	Input	3	2
Not Supported	[0x5-1-104]	Change-of-State	Input	3	2

Note:

After configuring these attributes, the MFC device must be power cycled for the changes to take affect.

The "Inferred" Data Type for Assembly Objects

It is difficult to talk about I/O Assemblies without talking about Data Type since both are tied very closely together (For more detailed explanation on Data Type, see section below "Data Type").

The Assembly Object is a collection of parameters/attributes. The purpose being that one can move a grouping of data to many attributes at one time as opposed to using individual messages to get or set each attribute. A difficulty occurs when one or more of the attributes in the Assembly have a changeable or dynamic data type. For example, the attribute Flow in the S-Analog Sensor object has data type that is dependent on the value of attribute Data Type within the same object.

The Assembly objects defined in the MFC device profile are considered static, meaning that the number of bytes defined in the Assembly is the number of bytes that will be transferred, no more, no less.

Example:
S-Analog Sensor object's Data Type (Flow Sensor Data Type parameter) is configured to REAL and Assembly object Instance 1 is used to retrieve the flow data. The size of data type REAL is 4 bytes and therefore Flow will be reported as 4 bytes, but Assembly Instance 1 reports flow as 2 bytes. A data size mismatch/conflict has occurred. When a data size mismatch occurs, the Master device usually indicates this condition with an error.

The solution is to force the Data Type attribute (Flow Sensor Data Type parameter) to match the "inferred" data type of the Assembly instance. A mechanism is in place such that if an Assembly object has been assigned to an I/O Connection AND the I/O Connection becomes active (transitions to the Established state), the attribute/parameter that defines the data type for the particular object will be forced automatically to the inferred data type of the Assembly.

Example:
In the example above, attribute Data Type (Flow Sensor Data Type parameter) of the S-Analog Sensor Object will be changed to INT (same as the Assembly), enforcing the data size match between the Flow data and the Assembly instance.

NOTE:

While the I/O Connection is active, the Data Type parameter/attribute will have "get only" accessibility, effectively locking out any means of modifying the attribute, including the establishment of another I/O Connection, until the original I/O Connection is terminated.

4.1.2 Data Type

The Data Type of an attribute defines the way a data value is represented in the MFC and transmitted across the network, such as integer and floating point. Most of the attributes defined in the MFC have a static data type, meaning that the data type for that attribute is defined once and never changes.

Several of the key attributes in the MFC have a dynamic data type that is defined and configurable via a companion attribute. Flow in the S-Analog Sensor object (Flow Sensor Parameter) has a companion attribute to define Flow's data type. Some other key attributes with dynamic data type are Setpoint and Valve Position. It may be desirable when changing Data Units (see Data Units section below) to change the Data Type also. For example, if Flow and Setpoint are configured to represent Data Units "Percent", then Data Type INT would limit the values to whole numbers from 0 to 100. Therefore, the smallest increment of flow/setpoint would be 1. However, if Data Type was sent to REAL, Flow/Setpoint could now be floating point numbers between 0.00000 to 100.000.

The following table lists objects and their attributes associated with configuring data type.

Table 4-3 Configurable Data Type Attributes

EDS Parameter	EPATH	Applicable Units Table	Out-of Box Default
Flow Sensor Data Type	[0x31-1-3]	Appendix B - Data Type Definitions	INT
Value Data Type	[0x32-1-3]	Appendix B - Data Type Definitions	INT
Flow Control Data Type	[0x33-1-3]	Appendix B - Data Type Definitions	INT

Note:

The two data types supported by the Delta Series MFC for the dynamic data typed attributes are INT and REAL. An INT is a number that is represented by whole numbers ranging from –32768 to 32767. A REAL is a floating-point number ranging from –3.8E34 to 3.8E34.

4.1.3 Data Units

The Delta Series MFC is capable of reporting flow and accepting setpoints in values associated to engineering units. This can simplify user interpretation of information from the device by letting the device perform the calculations necessary to interpret the flow signal from its internal sensor based upon information in the selected calibration. Though some may be more familiar with Counts or Percent from using Analog type MFC's, the interpretation of Counts (see section Counts below and Range Definition in Quick Start) can vary.

The following table lists objects and their attributes associated with data units. Data Units attributes have a data type of UINT.

Table 4-4 Configurable Data Units Attributes

EDS Parameter	EPATH	Applicable Units Table	Out-of Box Default
Flow Sensor Data Units	[0x31-1-4]	Appendix C: Volumetric Flow Data Units Table Appendix C: Mass Flow Data Units Table	Counts
Valve Data Units	[0x32-1-4]	Appendix C: Volumetric Flow Data Units Table Appendix C: Mass Flow Data Units Table	Counts
Flow Control Data Units	[0x33-1-4]	Appendix C: Volumetric Flow Data Units Table Appendix C: Mass Flow Data Units Table	Count

Counts
Counts define a range from 0 to 100% of Flow. The problem with this unit is defining what is 100%. In Delta Series, the range of numbers available to define 0-100% is -32768 to 32767. Delta forces 0% to be 0 Counts, and defaults 100% to 23405 Counts. This 23405 number allows for a flow signal to be indicated as high as 140% of full scale (32767 = 140%). If the user desires to change this upper range number, a service is provide in the S-Analog Sensor object (Set Full Scale Counts) to change this value.

Note:

If the value 32767 is used for 100%, be aware that any flow value that exceeds 100% of full scale cannot be indicated. This is a result of the limitation of a 2 byte signed integer number for which Counts is associated with.

4.1.4 Safe State (i.e. Safe Mode)

The Safe State or Safe Mode of the devices means that the device has transitioned to a state that is considered "safe" whereby it will shut down the normal controller process and place mechanical and sensing mechanisms in the device to a safe condition (e.g.. The actuator may be forced closed). The device will ignore input stimuli, such as setpoint or actuator override. In most cases, the safe mode of the device is equivalent to the operational mode when the device is not powered. Example, an MFC with a normally closed actuator would not be able to flow gas in an un-powered state because the actuator would physically be closed and the sensor signal indicating flow would be zero. However, this is not the case with normally open valves. The safe mode defaults for normally open devices are the same as the defaults for normally closed devices.

The State Machine and Safe Mode
All products in the Delta Series product line employ an internal State Machine to govern the operation of the device. The state machine is illustrated with supporting diagrams and transition tables in Appendix F. Within each state, the device can enter a mode that defines an operational behavior. One mode in particular is the Safe Mode (a.k.a. the Safe State).

The device will be in Safe Mode when the device is in any state other than the Executing State. The Executing State can be entered in two ways: first, the establishment and subsequent exchange of I/O data through an I/O connection, and second, a directive to the S-Device Supervisor Object using explicit messaging. Likewise, the Executing State will be exited by closing all I/O connections to the device , an I/O Connection timeout (i.e. EPR Timeout), or by a directive to the S-Device Supervisor Object. In addition, a transition out of the Executing State will occur internally by the device if a severe fault condition is detected.

The Safe Modes of the MFC/MFM device is described in the below tables. The label "default" indicates the out-of-the box default defined by the MFC device profile.

Valve Safe Mode
Attribute Safe State in S-Analog Actuator Object [0x32-1-21], or EDS Parameter Group Valve. These states apply to both Normally Closes and Normally Open Valves

Value	State
0	Closed(default)
1	Open
2	Hold
3	Safe State

Flow Sensor Safe Mode
Attribute Safe State in S-Analog Sensor Object [0x31-1-25], or EDS Parameter Group Flow Sensor

Value	State
0	Zero(default)
1	Full Scale
2	Hold Last Value
3	Use Safe Value
100	Track

4.1.5 Gas Calibration Selection

If the MFC/MFM contains multiple calibrations, the selection of a particular calibration can be configured in attribute "Gas Calibration Object Instance" of the S-Analog Sensor Object [0x31-1-35], Parameter Flow Sensor Gas Cal Instance in the Flow Sensor group of parameters.

To obtain a list of all calibrations in the device, service 0x4B, "Get All Instances", of the S-Gas Calibration Object [0x34] can be invoked which will return the list.

4.2 EDS Parameter Configuration

This section describes the parameters defined in the EDS file. These parameters are grouped logically in the following categories:

Group Name	Description
Flow Sensor	These parameters are associated with configuring the flow sensor. The *attributes* associated with these parameters are found in the S-Analog Sensor Object
Valve[1]	These parameters are associated with configuring the Valve. The *attributes* associated with these parameters are found in the S-Analog Actuator Object
Flow Control[1]	These parameters are associated with configuring the Flow Control behavior. The *attributes* associated with these parameters are found in the S-Single Stage Controller Object. These parameters select the I/O Data Assembly to be used with the Polled I/O Connection.
Polled Connection	The *attributes* associated with these parameters are found in the Connection Object – Explicit Connection Instance.
Device Info	These are informational parameters. The *attributes* associated with these parameters are found in the S-Device Supervisor Object.
Softstart Control[1]	These parameters configure the softstart control (ramping) feature of the device. The *attributes* associated with these parameters are found in the S-Single Stage Controller Object.
PID Control[1]	These parameters are used to tune the control performance of the device. The *attributes* associated with these parameters are found in the S-Single Stage Controller Object.

4.2.1 Flow Sensor

Table 4-5 Flow Sensor Configuration Values

EDS Parameter	Values	Description	EPATH
Flow Sensor Data Type[2,3,4]	Appendix B - Data Type Definitions	The data type associated with reporting flow	[0x31-1-3]
Flow Sensor Data Units[1]	See "Volumetric Units Table" and "Mass Units Table" in Appendix C -Data Units	The engineering units associated with reporting flow	[0x31-1-4]
Flow Sensor Safe State	See Note: Flow Sensor Safe State below	Configures the behavior of the flow sensor when the device is in Safe Mode	[0x31-1-25]
Flow Sensor Gas Cal Instance	1 up to the number of calibrations in the device	Sets the calibration curve used by the sensor to linearize the flow sensor values	[0x31-1-35]

1. Available for Mass Flow Controllers Only
2. This parameter is only configurable when the device is in the Idle State (i.e. when the device is not being actively scanned/polled)
3. This parameter will automatically be set based upon the inferred data type of the selected Assembly used for I/O (i.e. if the Assembly that is selected to report flow is INT, then this attribute will be automatically set to INT once I/O data exchange begins)
4. The device only supports values INT and REAL. All others will be rejected.

4.2.1.1 Note: Flow Sensor Safe State

Table 4-6 Flow Sensor Safe State Values

Value	State	Description
0	Zero (default)	The sensor value will be 0
1	Full Scale	The sensor value will equal Full Scale
2	Hold Last Value	The sensor value will hold to the most recent sensor reading just prior to entering the safe state
3	Use Safe Value	The sensor value will equal Safe Value attribute of the S-Analog Sensor Object
100	Track	The sensor value will continue to update normally as if in the Executing State

4.2.2 Valve

Table 4-7 Valve Values

Parameter	Values	Description	Path
Valve Data Type[1,2]	Appendix B - Data Type Definitions	The data type associated with reporting valve position.	[0x32-1-3]
Valve Data Units[1]	See the Actuator Units Table Appendix C - Data Units	The engineering units associated with reporting valve position.	[0x32-1-4]
Valve Safe State	See Note: Valve Safe State below	Configures the behavior of the valve when the device is in Safe Mode	[0x32-1-21]
Valve Override[3]	See Note: Valve Override below	Overrides the controller by commanding the valve to one of the selectable values in the table below.	[0x32-1-5]

4.2.2.1 Note: Valve Safe State

The following table outlines valid safe states for the valve. This table applies for normally closed and normally open valves.

Table 4-8 Valve Safe State Values

Value	State	Description
0	Closed	The valve will be driven closed and the valve position will be 0% or 0 Counts
1	Open	The valve will be driven open and the valve position will be 100% or 32767 Counts
2	Hold Last Value	The valve will be driven to the last updated value just prior to the entering of the safe state.
3	Use Safe Value	The valve will be driven to the value configured in the Safe Value attribute of the S-Analog Actuator

1. This parameter is only configurable when the device is in the Idle State (i.e. when the device is not being actively scanned/polled)
2. This parameter will automatically be set based upon the inferred data type of the selected Assembly used for I/O (i.e. if an Assembly is selected that reports valve position, and the inferred data type of the assembly is INT, then this attribute will be automatically set to INT once I/O data exchange begins)
3. This Override action only occurs when the device is in the Executing State.

4.2.2.2 Note: Valve Override

The following table outlines the valid valve override types

Table 4-9 Valve Override Values

Value	Sate	Description
0	Normal	Valve is under normal operational control
1	Closed	Valve is driven fully closed
2	Open	Valve is driven fully open
3	Hold	Valve is held to last updated value prior to assertion of override
4	Safe State	Valve is driven to the condition specified by Safe State

4.2.3 Flow Control

Table 4-10A Flow Control Values

Parameter	Values	Description	Path
Flow Control Data Type[3,4]	Appendix B - Data Type Definitions	The data type associated with the value of Setpoint.	[0x33-1-3]
Flow Control Data Units[1]	See the Actuator Units Table Appendix C - Data Units	The engineering units associated with Setpoint.	[0x33-1-4]
Flow Control Constant Time Ramp Rate	0 to 86400000 msec	Sets the time the controller will takes to move from the current position to the new setpoint position	[0x33-1-19]
Flow Control Constant Slope Ramp Rate	0.0 to 3.4e38	Sets the slope (percent change per second) that the controller will use to move from the current position to the new setpoint position	[0x33-1-100]
Flow Control Derivative Gain	-3.4e38 to 3.4e38	The value of the Derivative gain of the PID controller	[0x33-1-101]
Flow Control Integral Gain	-3.4e38 to 3.4e38	The value of the Integral gain of the PID controller	[0x33-1-102]
Flow Control Proportional Gain	-3.4e38 to 3.4e38	The value of the Proportional gain of the PID controller	[0x33-1-103]

4.2.4 Polled Connection

Table 4-10B Flow Control Values

Parameter	Values	Description	Path
Poll Prod Assy[5]	See "Summary Table of Input Assemblies" - Appendix A - Assembly Object Detail	This parameter selects the Assembly that the Polled I/O Connection will use to produce data	[0x5-2-100]
Poll Cons Assy[2]	See "Summary Table of Output Assemblies" - Appendix A - Assembly Object Detail	This parameter selects the Assembly that the Polled I/O Connection will use to receive data	[0x5-2-101]

1. Available for Mass Flow Controllers Only
2. This Override action only occurs when the device is in the Executing State
3. This parameter is only configurable when the device is in the Idle State (i.e. when the device is not being actively scanned/polled)
4. This parameter will automatically be set based upon the inferred data type of the selected Assembly used for I/O (i.e. if an Assembly is selected that contains Setpoint as one of its data members, and the inferred data type of the assembly is INT, then this attribute will be automatically set to INT once I/O data exchange begins)
5. A power cycle or reset is required for the value of this attribute to take affect.

4.2.5 Device Info

All values associated with this group are Read Only

Table 4-11 Device Info

Parameter	Values	Description	Path
Device Type	"MFC" or "MFM"	The SEMI Standard specified name which identifies the Device Model used by this device	[0x30-1-3]
SEMI Standard Rev. Level	" E54-0997"	The revision level of the SEMI S/A Network Standard to which this device complies	[0x30-1-4]
Mfr. Name	"Brooks Instrument"	The name of the manufacturer of the device	[0x30-1-5]
Mfr. Model Number		The model number of the device assigned by Brooks Instrument	[0x30-1-6]
Software Rev.		The firmware revision contained in the device	[0x30-1-7]
Hardware Rev.		The hardware revision of the electronics contained in the device	[0x30-1-8]
Mfr. Serial Number		The serial number assigned to the device by Brooks Instrument	[0x30-1-9]
Device Config		Any additional configuration information related to the device	[0x30-1-10]

4.2.6 Soft Start (Ramp) Control

These parameters are repeated for convenience from the Flow Control Group above

Table 4-12 SoftStart Ramp Control

Parameter	Values	Description	Path
Flow Control Constant Time Ramp Rate	0 to 86400000 msec	See Flow Control Group	[0x33-1-19]
Flow Control Constant Slope Ramp Rate	0.0 to 3.4e38	See Flow Control Group	[0x33-1-100]

4.2.7 PID Control

These parameters are repeated for convenience from the Flow Control Group above

Table 4-13 PID Control

Parameter	Values	Description	Path
Flow Control Derivative Gain	-3.4e38 to 3.4e38	See Flow Control Group	[0x33-1-101]
Flow Control Integral Gain	-3.4e38 to 3.4e38	See Flow Control Group	[0x33-1-102]
Flow Control Proportional Gain	-3.4e38 to 3.4e38	See Flow Control Group	[0x33-1-103]

THIS PAGE WAS INTENTIONALLY LEFT BLANK

5 Detail Configuration

This section details all of the Classes, Instances, Attributes and Services supported by the Delta Series MFC/MFM. Differences between the MFC and MFM type devices are noted as exceptions in each sub-section.

Note:

This section is recommended for advance users of DeviceNet and Brooks MFC/MFM products.

The classes detailed in the following sections can categorized into following functional groups to indicate what aspect of the device is being configured:

Communications

Classes in this category define how the device communicates on DeviceNet. A great majority of the information that is accessible is very well defined in the DeviceNet specification and therefore does not require much custom configuration. If the user of this device would like to do custom configuration in this category, it is strongly recommended that the user have a thorough understanding of these classes and DeviceNet protocols. More detail can be found about these classes in the DeviceNet specification.

Data Flow

Classes in this category define how data is moved to and from Application classes to the Communication classes. Again, much of what is detailed in this category is provided for informational purposes and is very limited in its ability to be customized

Application

This category of classes defines how the device is to function. In this case, the attributes in these classes configure the behavior of an MFC/MFM. Of all the categories listed here, this is the one that is the most customizable.

The table below lists the classes accessible in the Delta Series MFC/MFM and their associated category. It also indicates if the class is present in either of the MFC or MFM device types.

Table 5-1 Accessible Classes

Class	Category	No Of Instances	MFC	MFM
Identity Object [0x1]	Communications	1	Y	Y
Message Router Object [0x2]	Data Flow	1	Y	Y
DeviceNet Object [0x3]	Communications	1	Y	Y
Assembly Object [0x4]	Data Flow	19	Y	Y
Connection Object [0x5]	Communications	1..4	Y	Y
Acknowledge Handler Object [0x2B]	Communications	1	Y	Y
S-Device Supervisor Object [0x30]	Application	1	Y	Y
S-Analog Sensor Object [0x31]	Application	1	Y	Y
S-Analog Actuator Object [0x32]	Application	1	Y	N/A
S-Single Stage Controller Object [0x33]	Application	1	Y	N/A
S-Gas Calibration Object [0x34]	Application	1..n	Y	Y

In the ODVA™ DeviceNet specification, Instance 0 of both attributes and services are referred to as Class Level attributes and services. Instance 1 and higher are referred to as Instance Level attributes and services. This document will refer to all Levels by their instance number to avoid possible confusion.

The following details the meaning of the table heading names:

Attribute ID: The ID number of the attribute

Name: The ODVA™ DeviceNet Specification label for the attribute

Data Type: The ODVA™ DeviceNet Data Type for this attribute. See Appendix B for the definition of each data type.

Access Rule: "Get" means that the value of this attribute is "Read Only". "Set" means that the value of this attribute can be read and/or written.

NV: "NV" = The value of the attribute is stored in NV memory and its value will be retained after a power cycle. "V" = The value of the attribute is in volatile memory and its value will be returned to default after a power cycle.

Description: A brief description of the meaning of the attribute.

Notes: Any additional notations of importance about the attribute. These notes will be found in the same section as the table.

5.1 Identity Object [0x1]

The Identity Object contains informational attributes that uniquely describe the device.
Example: The use of attributes Vendor ID, Device Type, Product Code, and Serial Number together uniquely describe this device from any other device.

5.1.1 Attributes

Table 5-2 Identity Object [0x1] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
1	Vendor ID	UINT	Get	NV	ID Number assigned to vendor by ODVA™	Brooks Instrument ID = 246
2	Device Type	UINT	Get	NV	Numeric identifier indicating the ODVA™ Device Profile implemented by the device.	Device Type = 26 See 'Note: Device Type' Below
3	Product Code	UINT	Get	NV	Identification of a particular product of an individual vendor.	MFC = 1 MFM = 2
4	**Revision** Major Revision Major Revision	STRUCT of: USINT USINT	Get	NV	Revision of the device the Identity Object represents	See 'Note: Revision' Below
5	Status	WORD	Get	V	Summary status of the device.	See 'Note: Status' Below
6	Serial Number	UDINT	Get	NV	A unique identifier	See 'Note: Serial Number' Below
7	Product Name	Product Name	Get	NV	A human readable product name label for this device.	

5.1.1.1 Note: Device Type

Not to be confused with the S-Device Supervisor Object's attribute Device Type that is a string value defined in the Device Profile implemented by this device.

5.1.1.2 Note: Revision

The ODVA™ specification defines Major Revision as a significant change to the fit, form, or function of the product. Minor Revision is defined as changes that do not affect user configuration choices such as bug fixes, hardware component change, labeling change, etc.

5.1.1.3 Note: Status

The S-Device Supervisor object has an attribute called Device Status that operates in parallel to this attribute and provides additional device information that is not available using this attribute. It is recommended that users use the S-Device Supervisor Device Status attribute for determining the status of the device. The table below defines the valid values that can be returned when reading the Identity Object Status attribute. Note that the Status attribute values are Bit oriented.

Table 5-3 Note: Identity Object [0x1] Status

Bit	Description
0	Set to 1 when the device has been allocated.
1 – 8	Reserved, always zero
9	Minor Recoverable Fault, S-Device Supervisor object Device Status Attribute = Abort
10	Major Recoverable Fault, S-Device Supervisor object Device Status Attribute = Critical Fault
11-15	Reserved, always zero

5.1.1.4 Note: Serial Number

This Serial Number attribute differs from the S-Device Supervisor Serial Number attribute whereby the Identity Object attribute is strictly a numeric value that is guaranteed to be unique by the manufacturer across all of the manufacturer's DeviceNet products. The S-Device Supervisor attribute Serial Number is a string value that should represent the manufacturers method of defining serial numbers for its products.

5.1.2 Services

Table 5-4 Identity Object [0x1] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 SERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single See 'Service Reset' Details Below
0x05	Reset	Resets the DeviceNet interface of the device.	

5.1.3 Service Details

5.1.3.1 Service Reset

Table 5-5 Identity Object [0x1] Service Reset

Arguments

Parameter Name	Data Type	Required	Parameter Value	Semantics
Type	USINT	N	0	Emulate as closely as possible cycling power on the item the *Identity Object* represents. This value is the default if this parameter is omitted (**default**).

5.1.3.2 Service Reset

Table 5-6 Identity Object [0x1] Service Reset

Response

Parameter Name	Data Type	Required	Parameter Value	
NO RESPONSE DATA				

5.2 Message Router Object [0x2]

The Message Router is responsible for tracking the accessibility of the Classes and Instances via the DeviceNet network. Functionally (transparent to the user), the object routes explicit messages to the various objects in the device. What can be obtained directly from this class is a list of objects that the device supports.

5.2.1 Attributes

Table 5-7 Message Router Object [0x2] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
1	Object List	STRUCT of:	Get	NV	A List of supported objects	
	Number	UINT			The number of supported classes in the class array.	
	Classes	ARRAY OF UNIT			A list of Class ID codes	

5.2.2 Services

Table 5-8 Message Router Object [0x2] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 SERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single

5.2.3 Service Details

NONE

5.3 DeviceNet Object [0x3]

The DeviceNet Object is responsible for maintaining and managing the DeviceNet communications interface for the device. Information concerning the MAC ID, Baud Rate of the device can be found here.

This object also is responsible for managing the pre-defined Master/Slave Connection Set. Through this object, services are provided to allocate and release connections associated with the connection set.

5.3.1 Attributes

Table 5-9 DeviceNet Object [0x3] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
1	Revision	UINT	Get	NV	Revision of the DeviceNet Object Class Definition upon which the implementation is based. Range 1– 65535	

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
1	MAC ID	USINT	Cond.	NV	The node address of this device	See 'Note: MAC ID' below
2	Baud Rate	USINT	Cond.	NV	The communication data rate	See 'Note: Baud Rate' below
3	BOI	BOOL	Get	V	Bus Off Interrupt	See 'Note: BOI' below
5	**Allocation Information**	STRUCT of:	Get	V		
	Allocation Choice Byte	BYTE			The current Master/Slave Connection set Connections that have been allocated	See 'Note: Allocation Information' below
	Master's MAC IDAC	USINT			The node address of the current Master	
8	ID Switch Value	USINT	Get	V	Actual value of the Node Address switches	
9	Baud Rate Switch Value	USINT	Get	V	Actual value of the Baud Rate switch	

5.3.1.1 Note: MAC ID

The valid range of values for this attribute is 0 to 63. Setting the MAC ID switches on top of the device can configure the node address of this device. Reading this attribute will return the node address as configured with the switches. However, if switch "MSD" is placed in the range labeled "P", the node address can be configured by setting this attribute using the Set Attribute Single service. In either case, once the MAC ID has been configured, the value is retained in non-volatile memory, and the device must be power cycled, or a Reset service must be sent to the Identity Object for the changes to take affect. If the device cannot determine what its MAC ID should be, the MAC ID defaults to 63.

NOTE:

If the "MSD" switch is not in the range labeled "P", the Set Attribute Single service will return the error code 0x0E ("Attribute not settable").

5.3.1.2 Note: Baud Rate

The valid values for this attribute are:

Value	Meaning
0	125 K Baud
1	250 K Baud
2	500 K Baud

Setting the Baud Rate switch on top of the device can configure the communication's data rate of this device. Reading this attribute will return the node address as configured with the switches. However, if the Baud Rate switch is placed in the range labeled "P", the data rate can be figured by setting this attribute using the Set Attribute Single service. In either case, once the Baud Rate has been configured, the value is retained in non-volatile memory, and the device must be power cycled, or a Reset service must be sent to the Identity Object for the changes to take affect. If the device cannot determine what its Baud Rate should be, the device defaults to 125K.

NOTE:

If the Baud Rate switch is not in the range labeled "P", the Set Attribute Single service will return the error code 0x0E ("Attribute not settable").

5.3.1.3 Note: BOI

This attribute configures the action that the device will take in processing the Bus Off Interrupt. Valid values are:

Table 5-10 DeviceNet Object [0x3] BOI

Value	Meaning
0	Hold the CAN chip in its bus-off (reset) state upon detection of a bus-off indication.
1	If possible, fully reset the CAN chip and continue communicating upon detection of a bus-off indication

5.3.1.4 Note: Allocation Information

Part of the information contained in this attribute indicates the Master/Slave Connections that are currently allocated and active. The table below details the bit mapping of this BYTE descriptor.

Table 5-11 DeviceNet Object [0x3] Attributes

Bit	7	6	5	4	3	2	1	0
Description	0	Acknowledge Suppression	Cyclic	Change of State	0	0	Polled	Explicit Message

5.3.2 Services

Table 5-12 DeviceNet Object [0x3] Services

Instance 0

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single

Instance 1

Service Code	Service Name	Service Description Details	
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute	Appendix E: Set Attribute Single
0x4B	Allocate Master/Slave Connection	Requests the instantiation of a Master/Slave Connection	Allocate Master/Slave Connection
0x4C	Release Master/Slave Connection	Releases (deletes) a Master/Slave Connection	Release Master/Slave Connection

5.3.3 Service Details

5.3.3.1 Service Allocate Master/Slave Connection

Table 5-13 DeviceNet Object [0x3] Service Allocate Master/Slave Connection

Arguments

Parameter Name	Data Type	Required	Parameter Value	Semantics
Allocation Choice Byte	BYTE	Y	See Allocation Choice Byte Below	Selects the Master/Slave connection to be allocated (opened) in the slave device.
Allocator's MAC ID	USINT	Y	0 to 63	The MAC ID of the Master Device issuing this request

Allocation Choice Byte

Bit	7	6	5	4	3	2	1	0
Description	0	Acknowledge Suppression	Cyclic	Change of State	0	0	Polled	Explicit Message

Response

Parameter NameType	Data	Required Value	Parameter	Semantics
Message Body Format	BYTE	Y		Sets the format of the Explicit Message body to be used between the Client (Master) and the slave device. This value can be ignored by the Client

5.3.3.2 Service Release Master/Slave Connection

Table 5-14 DeviceNet Object [0x3] Service Release Master/Slave Connection

Arguments

Parameter Name	Data Type	Required	Parameter Value	Semantics
Release Choice Byte	BYTE	Y	See Release Choice Byte Below	Selects the Master/Slave connection to be release (closed) in the slave device.

Release Choice Byte

Bit	7	6	5	4	3	2	1	0
Description	0	Acknowledge Suppression	Cyclic	Change of State	0	0	Polled	Explicit Message

Response

Parameter Name	Data Type	Required	Parameter Value	Description
NO RESPONSE DATA				

5.3.3.3 Rules Governing Master/Slave Connections

When allocating Master/Slave connections using the Allocate service, the entire allocation request must be valid or else the allocation is rejected.

When allocating any of the Master/Slave I/O Connections, the Explicit Connection must have already been allocated first. OR, if the Explicit Connection has not been allocated, it can be allocated in same request as the I/O Connections.

A Master/Slave Connection that is already allocated cannot be allocated again unless it is released. The allocation of a Connection that has already been allocated AND not released will be considered an invalid request.

Multiple Master/Slave Connections can be requested in the same allocation request.

The Master/Slave Cyclic and Change of State I/O Connections are mutually exclusive and therefore cannot exist at the same time. These two connections, although different in behavior, are considered the same Connection instance.

Once a Master has made a successful allocation request to a slave, no other Master may make allocation requests to the slave until all Master/Slave connections are released.

Any Master may release Master/Slave connections in any Slave.

Allocation of the Master/Slave Cyclic or Change of State I/O Connection will automatically allocate the Poll I/O Connection if the Poll I/O Connection is not already allocated.

5.4 Assembly Object [0x4]

The Assembly Object contains a list of attributes that data can be written to (sink) and read from (source) via the Data Buffer attribute contained in this object. The Assembly Object is generally assigned as the endpoint of an I/O Connection object (assigned via the Path attributes in the Connection Object). In this way, large amounts of data can be moved to and from the network quickly.

5.4.1 Attributes

Table 5-15 Assembly Object [0x4] Attributes

Instance 0

Attrib ID	Name	Data Type	AccessRule	NV	Description	Notes
1	Revision	UINT	Get	NV	Revision of this Object	Default = 2

Instance 1..n

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
1	Number of Members in List	UINT	Get	NV	Lists the number of members in this Assembly instance	
2	Member List	ARRAY of STRUCT:	Get	NV	The member list is an array of DeviceNet paths	
	Member Data Description	UINT			Size of member data	Size in bits
	Member Path Size	UINT			Size of member path (in bytes)	
	Member Path	EPATH			Path to the attribute this member points to	
3	Data Buffer	ARRAY of BYTE	Set		The data buffer to sink or source the member(s) data	The size of this attribute can be determined by adding together the Member Data Description for each member in the Assembly

5.4.2 Services

Table 5-16 Assembly Object [0x4] Services

Instance 0

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single

5.4.3 Service Details

NONE

5.5 Rules Governing Master/Slave Connections [0x5]

The Connection Object configures the characteristics and behavior of a logical connection in the device. A Connection is allocated (opened) in the device by the Master via the Allocate M/S Connection Request service in the DeviceNet Object. Configuration of the Connection Object attributes for the requested connection is done automatically in the device and generally need no further modification except for setting the Expected Packet Rate attribute (Attribute 9). However, it is within this object that selection of Input/Output assemblies may be configured. There are two ways to perform this configuration:

1. Modification of Attribute 100 through 104 in the M/S Explicit Connection

   OR

2. Modification of the Produce and Consume Path attributes (Attribute 14 and 15 respectively) in the I/O Connection Objects

5.5.1 Attributes

Table 5-17 Rules for Governing Master/Slave Connections [0x5] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1 - M/S Explicit Connection

Attrib ID	Name	Data Type	Access Rule	Description	Notes
1	State	USINT	Get	State of Object	Note: 'State' below Default: 3 = Established State
2	Instance Type	USINT	Get	Indicates type of Connection	Default: 0 = Explicit Connection
3	Transport Class & Trigger	USINT	Get	Defines behavior of Connection Placed in CAN Identifier	Note: 'Transport Class & Trigger' below Default: 0x83
4	Produced Connection ID	UINT	Get	Field when the Connection transmits	
5	Consume Connection ID	UINT	Get	CAN Identifier Field value that denotes message to be received	
6	Initial Comm Characteristics	BYTE	Get	Defines the Message Group(s) across which productions and consumptions associated with this Connection occur	Note: 'Initial Comm Characteristics' below Default: 0x21
7	Produced Connection Size	UINT	Get	Maximum number of bytes transmitted across this Connection	Default: 0xFFFF
8	Consumed Connection Size	UINT	Get	Maximum number of bytes received across this Connection	Default: 0xFFFF
9	Expected Packet Rate (EPR)	UINT	Set	Defines timing associated with this Connection in milliseconds	Default: 2500 msec
12	Watchdog Timeout Action	USINT	Set	Specifies how the Connection will handle inactivity/watchdog timeouts	Note: 'Watchdog Timeout Action' below Default: 1 = Auto Delete

Instance 1 (M/s Explicit Connection Continued)

Attrib ID	Name	Data Type	Access Rule	Description	Notes
13	Produce Connection Path Length	UINT	Get	Number of bytes in the Produced Connection Path attribute	Default: 0
14	Produce Connection Path	EPATH	Get	Specifies the application object data to be produced by this Connection	Default: Empty
15	Consume Connection Path Length	UINT	Get	Number of bytes in the Consume Connection Path attribute	Default: 0
16	Consume Connection Path	EPATH	Get	Specifies the application object to receive data by this Connection	Default: Empty
100	Default Polled I/O Conn. Produce Assembly Instance	UINT	Set	Specifies the default Assembly Object instance to be set in the Produce Connection Path when the M/S Polled I/O Connection Object is instantiated	The value of this attribute is stored in Non-Volatile memory
101	Default Polled I/O Conn. Consume Assembly Instance	UINT	Set	Specifies the default Assembly Object instance to be set in the Consume Connection Path when the M/S Polled I/O Connection Object is instantiated	The value of this attribute is stored in Non-Volatile memory
102	Default Cyclic I/O Conn. Produce Assembly Instance	UINT	Set	Specifies the default Assembly Object instance to be set in the Produce Connection Path when the M/S Cyclic I/O Connection Object is instantiated	The value of this attribute is stored in Non-Volatile memory
104	Default COS I/O Conn. Produce Assembly Instance	UINT	Set	Specifies the default Assembly Object instance to be set in the Produce Connection Path when the M/S Change of State I/O Connection Object is instantiated	The value of this attribute is stored in Non-Volatile memory

Table 5-17 Rules for Governing Master/Slave Connections [0x5] Attributes (continued)

Instance 2 - M/S Polled I/O Connection

Attrib ID	Name	Data Type	Access Rule	Description	Notes
1	State	USINT	Get	State of Object	Note: 'State' below Default: 1 = Configuring State
2	Instance Type	USINT	Get	Indicates type of Connection	Default: 1 = I/O Connection
3	Transport Class & Trigger	USINT	Get	Defines behavior of Connection	Note: 'Transport Class & Trigger' below Default: 0x82
4	Produced Connection ID	UINT	Get	Placed in CAN Identifier Field when the Connection transmits	
5	Consume Connection ID	UINT	Get	CAN Identifier Field value that denotes message to be received	
6	Initial Comm Characteristics	BYTE	Get	Defines the Message Group(s) across which productions and consumptions associated with this Connection occur	Note: 'Initial Comm Characteristics' below Default: 0x01
7	Produced Connection Size	UINT	Get	Maximum number of bytes transmitted across this Connection	Defined by the size of the Application Data object pointed to by Produce Connection Path attribute
8	Consumed Connection Size	UINT	Get	Maximum number of bytes received across this Connection	Defined by the size of the Application Data object pointed to by Produce Connection Path attribute
9	Expected Packet Rate (EPR)	UINT	Set	Defines timing associated with this Connection	Must be configured. Once this value is set, the Polled I/O Connection transitions to the Established State
12	Watchdog Timeout Action	USINT	Set	Specifies how the Connection will handle inactivity/watchdog timeouts	Note: 'Watchdog Timeout Action' below Default: 0 = Timed Out Stat
13	Produce Connection Path Length	UINT	Get	Number of bytes in the Produced Connection Path attribute	Default: 6
14	Produce Connection Path	EPATH	Conditional	Specifies the application object data to be produced by this Connection Number of bytes in the Consume	Default: As defined by Attribute 100 in the M/S Explicit Connection instance.This attribute is settable only in the Configuring State
15	Consume Connection Path Length	UINT	Get	Connection Path attribute	Default: 6
16	Consume Connection Path	EPATH	Conditional	Specifies the application object to receive data by this Connection	As defined by Attribute 101 in the M/S Explicit Connection instance. This attribute is settable only in the Configuring State

Table 5-17 Rules for Governing Master/Slave Connections [0x5] Attributes (continued)

Instance 4 - M/S COS/Cyclic I/O Connection (Acknowledged)

The Change of State and Cyclic connection types are mutually exclusive. When allocating this type of connection, only one or the other may be selected.

The selection of Acknowledged versus Unacknowledged is also mutually exclusive. See Allocating Connections in the DeviceNet Object.

Attrib ID	Name	Data Type	Access Rule	Description	Notes
1	State	USINT	Get	State of Object	Note: 'State' below Default: 1 = Configuring State
2	Instance Type	USINT	Get	Indicates type of Connection	Default: 1 = I/O Connection
3	Transport Class & Trigger	USINT	Get	Defines behavior of Connection	Note: 'Transport Class & Trigger' below Default: Cyclic Type = 0x02 COS Type = 0x12
4	Produced Connection ID	UINT	Get	Placed in CAN Identifier Field when the Connection transmits	
5	Consume Connection ID	UINT	Get	CAN Identifier Field value that denotes message to be received	
6	Initial Comm Characteristics	BYTE	Get	Defines the Message Group(s) across which productions and consumptions associated with this Connection occur	Note: 'Initial Comm Characteristics' below Default: 0x01
7	Produced Connection Size	UINT	Get	Maximum number of bytes transmitted across this Connection	Defined by the size of the Application Data object pointed to by Produce Connection Path attribute
8	Consumed Connection Size	UINT	Get	Maximum number of bytes received across this Connection	Default: 0
9	Expected Packet Rate (EPR)	UINT	Set	Defines timing associated with this Connection	Must be configured. Once this value is set, the Polled I/O Connection transitions to the Established State
12	Watchdog Timeout Action	USINT	Set	Specifies how the Connection will handle inactivity/watchdog timeouts	Note: 'Watchdog Timeout Action' below Default: 0 = Timed Out State
13	Produce Connection Path Length	UINT	Get	Number of bytes in the Produced Connection Path attribute	Default: 6
14	Produce Connection Path	EPATH	Cond [1]	Specifies the application object data to be produced by this Connection	Default: As defined by Attribute 102 in the M/S Explicit Connection instance. This attribute is settable only in the Configuring State
15	Consume Connection Path Length	UINT	Get	Number of bytes in the Consume Connection Path attribute	Default: 4
16	Consume Connection Path	EPATH	Get	Specifies the application object to receive data by this Connection	Default: 0x20 0x2B 0x24 0x1 This sets the path to instance 1 of the Acknowledge Handler Object
17	Production Inhibit Time	UINT	Set	Configures the minimum time between data productions	Default: 0

Note 1. This attribute is GET only when the Acknowledger Handler Object is active.

Table 5-17 Rules for Governing Master/Slave Connections [0x5] Attributes (continued)

Instance 4 - M/S COS/Cyclic Connection (Unacknowledged)

The Change of State and Cyclic connection types are mutually exclusive. When allocating this type of connection, only one or the other may be selected.

The selection of Acknowledged versus Unacknowledged is also mutually exclusive. See Allocating Connections in the DeviceNet Object.

Attrib ID	Name	Data Type	Access Rule	Description	Notes
1	State	USINT	Get	State of Object	Note: 'State' below Default: 1 = Configuring State
2	Instance Type	USINT	Get	Indicates type of Connection	Default: 1 = I/O Connection
3	Transport Class & Trigger	USINT	Get	Defines behavior of Connection	Note: 'Transport Class & Trigger' below Default: Cyclic Type = 0x00 COS Type = 0x10
4	Produced Connection ID	UINT	Get	Placed in CAN Identifier Field when the Connection transmits	
5	Consume Connection ID	UINT	Get	CAN Identifier Field value that denotes message to be received	
6	Initial Comm Characteristics	BYTE	Get	Defines the Message Group(s) across which productions and consumptions associated with this Connection occur	Note: 'Initial Comm Characteristics' below Default: 0x0F
7	Produced Connection Size	UINT	Get	Maximum number of bytes transmitted across this Connection	Defined by the size of the Application Data object pointed to by Produce Connection Path attribute
8	Consumed Connection Size	UINT	Get	Maximum number of bytes received across this Connection	Default: 0 Connection consumes no data
9	Expected Packet Rate (EPR)	UINT	Set	Defines timing associated with this Connection	Must be configured. Once this value is set, the Polled I/O Connection transitions to the Established State
12	Watchdog Timeout Action	USINT	Set	Specifies how the Connection will handle inactivity/watchdog timeouts	Note: 'Watchdog Timeout Action' below Default: 0 = Timed Out State
13	Produce Connection Path Length	UINT	Get	Number of bytes in the Produced Connection Path attribute	Default: 6
14	Produce Connection Path	EPATH	Cond [1]	Specifies the application object data to be produced by this Connection	Default: As defined by Attribute 104 in the M/S Explicit Connection instance. This attribute is settable only in the Configuring State
15	Consume Connection Path Length	UINT	Get	Number of bytes in the Consume Connection Path attribute	Default: 0
16	Consume Connection Path	EPATH	Get	Specifies the application object to receive data by this Connection	Default: Empty
17	Production Inhibit Time	UINT	Set	Configures the minimum time between data productions	Default: 0

Note 1. This attribute is GET only when the Acknowledger Handler Object is active.

5.5.1.1 State

Table 5-18 Rules for Governing Master/Slave Connections [0x5] State

Value	State	Description
0	Non-Existent	The Connection has not yet been instantiated
1	Configuring	The Connection has been instantiated but not yet properly configured. Upon completion of configuration the Apply service must be sent to the Connection
2	Waiting for Connection ID	The Connection is waiting exclusively for the Produce or Consume Connection ID's to be set
3	Established	The Connection has been properly configured and applied
4	Timed Out	The Inactivity/Watchdog has timed out and may have transitioned to this state if the Watchdog Timeout Action attribute has been configured to do so. See Attribute 12.
5	Deferred Delete	If the Connection Object is of Explicit Message Connection type (see Attribute 2) and the Inactivity/Watchdog has timed out, the Connection may transition to this state if the Watchdog Timeout Action attribute has been configured to do so. See Attribute 12.

5.5.1.2 Note: Transport Class & Trigger

Table 5-19 Rules for Governing Master/Slave Connections [0x5] Transport Class & Trigger

Bit Descriptions

Bit	7	6	5	4	3	2	1	0
Description	Direction	Production Trigger			Transport Class			

Direction

Value	Description
0	Client
1	Server

Production Trigger

Value	Description
0	Cyclic
1	Change of State
2	Application Object

Transport Class

Value	Description
0	Class 0
1	Reserved
2	Class 2
3	Class 3

5.5.1.3 Note: Initial Comm Characteristics

Table 5-20 Rules for Governing Master/Slave Connections [0x5] Initial Comm Characteristics

Bit Descriptions

Bit	7	6	5	4	3	2	1	0
Description	Initial Production Characteristics				Initial Consume Characteristics			

Initial Production Characteristics

Value	Description
0	Produce Across Message Group 1
1	Produce Across Message Group 2 (Destination)
2	Produce Across Message Group 2 (Source)
3	Produce Across Message Group 3

Initial Consume Characteristics

Value	Description
0	Consume Across Message Group 1
1	Consume Across Message Group 2 (Destination)
2	Consume Across Message Group 2 (Source)
3	Consume Across Message Group 3

5.5.1.4 Note: Watchdog Timeout Action

Table 5-21 Rules for Governing Master/Slave Connections [0x5] Watchdog Timeout Action

Value	Timeout Action	Description
0	Transition to Timed Out	The Connection transitions to the Timed Out state and remains in that state until Deleted or Reset. Invalid value for Explicit Connections
1	Auto Delete	The Connection Class automatically deletes the Connection if an Inactivity/Watchdog timeout occurs
2	Auto Reset	The Connection remains in the Established State and restarts the Inactivity/Watchdog timer
3	Deferred Delete	The Connection transitions to the Deferred Delete state if any child connection instances are in the Established state. If no child connection instances are in the Established state the connection is deleted. This value is invalid for I/O Messaging Connections.

5.5.2 Services

Table 5-22 Rules for Governing Master/Slave Connections [0x5] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 SERVICES ARE SUPPORTED		

Table 5-22 Rules for Governing Master/Slave Connections [0x5] Services (continued)

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single
0x5	Reset	Used to reset the Inactivity/Watchdog timer and transition the Connection back to the Established state.	No Arguments or Response Data

5.5.3 Service Details

NONE

5.6 Acknowledge Handler Object [0x2B]

The Acknowledge Handler Object is used in conjunction with the acknowledged COS/Cyclic I/O connection types. With acknowledged COS/Cyclic I/O, the Master acknowledges the receipt of data produce by the slave. The Acknowledge Handler is responsible for managing retries and timeouts if the acknowledgement message from the Master is not received.

5.6.1 Attributes

Table 5-23 Acknowledge Handler Object [0x2B] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	Description	Notes
1	Acknowledge Timer	UINT	Set	Time to wait for acknowledge before resending	
2	Retry Limit	USINT	Set	Number of Acknowledge Timeouts to wait before indicating to the producing connection of a timeout.	
3	COS Producing Connection Instance	UINT	Cond [1]	Connection Instance that contains the path of the producing I/O application object that will be notified of Acknowledge Handler events.	
4	Ack List Size	BYTE	Get	Maximum number of members in the Acknowledge List.	
5	Ack List	BYTE Array of UINT	Get	List of active connection instances which are receiving Acknowledges.	

Note 1. This attribute is GET only when the Acknowledger Handler Object is active.

5.6.2 Services

Table 5-24 Acknowledge Handler Object [0x2B] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 SERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single

5.6.3 Service Details

NONE

5.7 S-Device Supervisor Object [0x30]

The S-Device Supervisor Object oversees device operation and status. This object maintains an internal state machine that is used to govern the behavior of the Sensor, Actuator, and Controller objects. Also contained in this object are informational attributes extending those provided in the Identity Object.

5.7.1 Attributes

Table 5-25 Device Supervisor Object [0x30] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
3	Device Type	SHORT STRING	Get	NV	SEMI Standard specified name which identifies the Device Model within the *Hierarchy of Semiconductor Equipment Devices*	Max. 8 Characters 'MFC' or 'MFM'
4	SEMI Standard Revision Level	SHORT STRING	Get	NV	Specifies the revision level of the SEMI S/A Network Standard to which this device complies	'E54-0997'
5	Manufacturer's Name	SHORT STRING	Get	NV	The name of the manufacturer of the device.	Max. 20 characters 'Brooks Instrument'
6	Manufacturer's Model Number	SHORT STRING	Get	NV	The manufacturer specified model number for the device	
7	Software Revision Level	SHORT STRING	Get	NV	Revision level of the firmware in the device.	Note: 'Software Revision Level' below
8	Hardware Revision Level	SHORT STRING	Get	NV	Revision level of the hardware in the device.	
9	Manufacturer's Serial Number	SHORT STRING	Get	NV	Serial number of device assigned by the manufacturer	Max. 30 Characters
10	Device Configuration	SHORT STRING	Get	NV	Any additional manufacturer specific information about the device	Max. 50 characters 'N/A'
11	Device Status	USINT	Get	V	The current operational state of the state.	Note: 'Device Status'below
12	Exception Status	BYTE	Get	V	Summary status of the current warning and alarm status in the device	Note: 'Exception Status' below

Table 5-25 Device Supervisor Object [0x30] Attributes (continued)

Instance 1 (continued)

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
13	Exception Detail Alarm	STRUCT of:	Get	V	A complex structure made up of 3 levels of alarm exception details.	
	Common Exception Detail	STRUCT of:			Common detail bytes defined by the Supervisor Object	Note: 'Exception Detail Alarm - Common Exceptions' below
	Size	USINT				
	Detail	ARRAY of:				
	Detail n	BYTE				
	Device Exception Detail	STRUCT of:			Device detail bytes defined for MFC/ MFM device types.	Note: 'Exception Detail Alarm - Device Exceptions' below
	Size	USINT				
	Detail	ARRAY of:				
	Detail n	BYTE				
	Manufacturer Exception Detail	STRUCT of:			Manufacturer detail bytes as defined by Brooks Instrument	Note: 'Exception Detail Alarm - Manufacturer Exceptions' below
	Size	USINT				
	Detail	ARRAY of:				
	Detail n	BYTE				
14	Exception Detail Warning	STRUCT of:	Get	V	A complex structure made up of 3 levels of alarm exception details.	
	Common Exception Detail	STRUCT of:			Common detail bytes defined by the Supervisor Object	Note: 'Exception Detail Warning - Common Exceptions' below
	Size	USINT				
	Detail	ARRAY of:				
	Detail n	BYTE				
	Device Exception Detail	STRUCT of:			Device detail bytes defined for MFC/ MFM device types.	Note: 'Exception Detail Warning - Device Exceptions' below
	Size	USINT				
	Detail	ARRAY of:				
	Detail n	BYTE				
	Manufacturer Exception Detail	STRUCT of:			Manufacturer detail bytes as defined by Brooks Instrument	Note: 'Exception Detail Warning - Manufacturer Exceptions' below
	Size	USINT				
	Detail	ARRAY of:				
	Detail n	BYTE				
15	Alarm Enable	BOOL	Set	NV	Enables/Disables the Supervisor Object's processing of Alarm information	
16	Warning Enable	BOOL	Set	NV	Enables/Disables the Supervisor Object's processing of Warning information	
99	Subclass	UINT	Get	NV	Identifies a subset of additional instance attributes, services, and behaviors for the Supervisor Object	

5.7.1.1 Note: Software Revision Level

This attribute represents the current firmware revision running in the device. Brooks Instrument defines the format of this revision level The format of this attribute is comprised of two letters. The first letter indicates the major revision and the second is the minor revision. **Example:** AB

Any software revision that contains an X in the identifier is an experimental release. Releases such as these are strictly reserved for Beta evaluations and are not standard production release.

5.7.1.2 Note: Device Status

The value of this attribute represents the current operational state of the device. The values are defined in the following table. A State Event Matrix and State Event Diagram are outlines in Appendix F - State Machine

Table 5-26 Device Supervisor Object [0x30] Device Status

Attrib	Operational	Safe	Description
0	Undefined	Y	
1	Self Testing	Y	All object instances exist and have been initialized. Exception Status bits have been reset. The device is executing internal tests to determine if the device is qualified to begin normal monitoring and control operations.
2	Idle	Y	The device is ready for normal monitoring and control operations.
3	Self Test Exception	Y	The device has detected an exceptional condition during self-test. The details of the exception are stored in the exception detail attributes of the Supervisor Object.
4	Executing	N	The device is performing normal control and monitoring operations
5	Abort	Y	The device is idle, but is NOT ready for normal monitoring and control operations. A Recover Request must be initiated to reach a ready condition.
6	Critical Fault	Y	The Supervisor Object and device are in a fault state from which there is no recovery.

NOTE:

Safe State indicates the condition that the sensor, actuator, and controller will be in when the device is in a particular Operational State. The Safe State condition can be configured for each of these elements by setting the appropriate Safe State attribute in each of their corresponding objects (e.g. Analog Sensor Object, Analog Actuator Object, and Single Stage Controller Object) if supported.

5.7.1.3 Note: Exception Status

Exception Status for this device utilizes the Expanded Method outlined in the ODVA™ specification for the S-Device Supervisor Object. This is indicated by setting bit 7 in this attribute to 1.

The table below shows the format of bits in this attribute. The setting of any of these bits is obtained by taking the logical "OR" of the related exception detail bits. For example, if any of the bits in the Device Common Alarm Detail are set, bit 0 of this attribute will be set to 1.

Table 5-27 Device Supervisor Object [0x30] Exception Status

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	**Warning** Mfr. Specific	**Warning** Device Specific	**Warning** Device Common	0	**Alarm** Mfr. Specific	**Alarm** Device Specific	**Alarm** Device Common
	Note: Attribute 14.C	Note: Attribute 14.B	Note: Attribute 14.A		Note: Attribute 13.C	Note: Attribute13.B	Note: Attribute13.A

5.7.1.4 Note: Exception Detail Alarm – Common Exception Detail

Table 5-28 Device Supervisor Object [0x30] Common Exception Detail

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Detail Size	2							
Detail Byte 0	0	0	0	RAM	NV RAM	Flash Memory	0	Diagnostic
Detail Byte 1	0	0	0	0	0	0	0	0

5.7.1.5 Note: Exception Detail Alarm – Device Exception Detail

Table 5-29 Device Supervisor Object [0x30] Device Exception Detail

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Detail Size	1							
Detail Byte 0	0	0	Valve High	Valve Low	Flow Control	Flow High	Flow Low	0

5.7.1.6 Note: Exception Detail Alarm –Manufacturer Exception Detail

Table 5-30 Device Supervisor Object [0x30] Manufacturer Exception Detail

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Detail Size	0 (None Defined)							

5.7.1.7 Note: Exception Detail Warning – Common Exception Detail

Table 5-31 Device Supervisor Object [0x30] Common Exception Detail

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Detail Size	2							
Detail Byte 0	0	0	0	0	0	0	0	Diagnostic
Detail Byte 1	0	0	0	0	0	0	0	0

5.7.1.8 Note: Exception Detail Warning – Device Exception Detail

Table 5-32 Device Supervisor Object [0x30] Device Exception Detail

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Detail Size	1							
Detail Byte 0	0	0	Valve High	Valve Low	Flow Control	Flow High	Flow Low	0

5.7.1.9 Note: Exception Detail Warning – Manufacturer Exception Detail

Table 5-33 Device Supervisor Object [0x30] Manufacturer Exception Detail

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Detail Size	0 (None Defined)							

5.7.2 Services

Table 5-34 Device Supervisor Object [0x30] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 SERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single
0x05	Reset	Resets the device to the Self-Testing state	Acceptance of these services is dependent upon the current device operational state. See State Event Matrix or Diagram in Appendix F - State Machine. For details on Perform Diagnostic Service go to Perform Diagnostics in Service Details section
0x06	Start	Moves the device to the Executing State	
0x07	Stop	Moves the device to the Idle State	
0x4B	Abort	Moves the device to the Abort State	
0x4C	Recover	Moves the device out of the Abort State	
0x4D	Perform Diagnostics	Causes the device to perform a set of diagnostic routines.	

5.7.3 Service Details

5.7.3.1 Service Perform Diagnostics

Table 5-35 Device Supervisor Object [0x30] Service Perform Diagnostics

Arguments

Parameter Name	Data Type	Required	Parameter Value	Semantics
Test ID	USINT	N	See Test ID Table Below	The identifier of the diagnostic test to be performed. See Table Below

Test ID Table

Test ID	Description
0 (default)	Standard

Response

Parameter Name	Data Type	Required Value	Parameter	Description
NO RESPONSE DATA				

5.8 S-Analog Sensor Object [0x31]

The S-Analog Sensor Object is responsible for reporting sensor values. The sensor object can linearize the sensor values and convert measurements into engineering data units.

5.8.1 Attributes

Table 5-36 S-Analog Sensor Object [0x31] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
3	Device Type	USINT	Cond.	NV	Defines the Data Type of Flow[6] and other attributes in this object.	Default = INT See Note 'Data Type' below
4	Data Units	ENGUNITS	Cond.	NV	Defines the Engineering Units context of Flow[6] and other attributes in this object.	Default = Counts See Note 'Data Units' below
5	Reading Valid	BOOL	Get	V	Indicates that Flow[6] has a valid value.	0 = Invalid 1 = Valid Flow
6	Flow	Specified by Attrib. 3 & 4	Get	V	The amount of flow going through the sensor	This value is corrected, converted, and calibrated to report the actual value of flow.
7	Status	BYTE	Get	V	Alarm and Warning status for the S-Analog Sensor Object	See Note 'Status' below
10	Full Scale	Specified by Attrib. 3 & 4	Get	NV	The flow value that represents 100% of the currently selected calibration full scale	See Note 'Full Scale' below
11	Offset-A Data Type	USINT	Get	NV	Defines the data type for Offset-A[12]	Default = REAL
12	Offset-A	Specified by Attrib. 11	Set	NV	A value added to the sensor flow value in the process of determining Flow[6]	See Note 'Offset-A' below
25	Safe State	USINT	Set	NV	Specifies the behavior of value reported for Flow[6] when the device is in an Operational State other than Executing State	See Note 'Safe State' below
26	Safe Value	Specified by Attrib. 3 & 4	Set	NV	The value that is indicated by Flow[6] if the Safe State[25] is configured to "Use Safe Value"	
34	Produce Trigger Delta	Specified by Attrib. 3 & 4	Set	NV	The amount that Flow[6] must change by before a Change of State Production is triggered	See Note 'Produce Trigger Delta' below
35	Gas Calibration Object Instance	UINT	Set	NV	Configures which S-Gas Calibration Object instance is currently active for this object.	See Note 'Gas Calibration Object Instance' below
99	Subclass	UINT	Get	NV	Identifies a subset of additional instance attributes, services, and behaviors for the Sensor Object	The value of this attribute is 1 = Flow Diagnostics Subclass

5.8.1.1 Note: Data Type

The value of this attribute is limited to INT or REAL. This attribute can only be set when the device is not in the Operational State of Executing. If any of the attributes in this object use the attribute Data Type[3] to define their data type and the attribute is the end point of an active I/O connection, then the access rule for the attribute Data Type[3] becomes Get and thus cannot be set until the I/O connected becomes inactive. Additionally, the inferred data type of an Assembly object (see The "Inferred" Data Type) will modify this value if the Assembly object has been assigned to an active I/O connection.

Example 1: If the value of Flow[6] is exchanged via the Polled I/O connection, Data Type[3] cannot be modified until that Polled I/O Connection is terminated.

Example 2: If Assembly instance 1 is assigned to the Polled I/O connection produce path, the value of Data Type[3] will be set to INT when the Polled I/O connection transitions to the Active state. When the Polled I/O connection is terminated, the value of this attribute will remain INT. It will not return to the value prior to commissioning the Polled I/O connection.

5.8.1.2 Note: Data Units

The value of this attribute is limited to the values specified in the Volumetric Flow Units Table and Mass Flow Units table in Appendix C - Data Units. This attribute can only be set when the device is not in the Operational State of Executing. See Example 1 above in Note: Attribute 3.

5.8.1.3 Note: Status

The following table defines the bits in this status byte. The value of 1 = Set and 0 = Cleared.

Table 5-37 S-Analog Sensor Object [0x31] Status

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	Low Flow Warning	High Flow Warning	Low Flow Alarm	High Flow Alarm

5.8.1.4 Note: Full Scale

The value of this attribute will depend upon the values configured for attributes 4 and 35. The following table will describe the behavior of this attribute.

Table 5-38 S-Analog Sensor Object [0x31] Full Scale

Data Units	Full Scale Defintion
Counts	Default = 23405 for Full Scale[10] regardless of the selected calibration object full scale. This allows for an overshoot indication of 140% of the calibrated full scale (e.g. 32767 = 140%). To change this, utilize the Set Full Scale Counts service to set the desired 100% full scale. A good formula to use is: Full Scale Counts = (100% / Desired Percent Overshoot) X 32767 The value will always be 100.
Percent	Full Scale[10] will be determined by using the attribute Calibration Full Scale [0x34,1,6] of the S-Gas
Any Volumetric or Mass Flow Engineering Unit	Calibration Object as configured by the Selected Gas Calibration Object Instance[35] and converted to the configured Data Unit[4]. Example: If Instance 1 of an S-Gas Calibration Object has a Calibration Full Scale[6] of 100 Liters/sec, and the desired Data Units[4] in the Sensor Object is sccm, the value of Full Scale[10] will be 600000 (Note: Mass units will take into account the density when doing the conversion).

5.8.1.5 Note: Offset-A

The following formula is used to calculate the value of Flow[6].

(Raw Flow Signal + Offset-A[12]) x Full Scale [10]

Where Raw Flow Signal is a number such that 0.0 = Zero Flow and 1.0 is equal to 100% of Full Scale.

5.8.1.6 Note: Safe State

The following table outlines valid values for this attribute:

Table 5-39 S-Analog Sensor Object [0x31] *Safe State*

Value	State	Description
0	Zero (default)	The value of Flow[6] will be 0
1	Full Scale	The value of Flow[6] will equal Full Scale[10]
2	Hold Last Value	The value of Flow[6] will hold to the most recent sensor reading just prior to entering the safe state
3	Use Safe Value	The value of Flow[6] will equal Safe Value[26]
100	Track	The value of Flow[6] will continue to update normally as if in the Executing State

5.8.1.7 Note: Produce Trigger Delta

This attribute only affects the behavior of the COS I/O Connection. This attribute will set the threshold of when to create a COS trigger event. The following algorithm is employed to generate a trigger:

IF

Flow[6] >= Last COS Trigger Value + Product Trigger Delta[34]

THEN

COS Trigger Event

Last COS Trigger Value = Flow[6]

ENDIF

This algorithm is run once every two control loop times which is approximately once every 40 milliseconds. Therefore, COS Production will occur no faster than this. Also, the Production Inhibit Timer, if configured for the COS Connection, will throttle the frequency of I/O data produced.

5.8.1.8 Gas Calibration Object Instance

The value of this attribute is limited to the number of S-Gas Calibration Object instances configured in the device. The minimum value is 1, which is also the default value.

5.8.2 Services

Table 5-40 S-Analog Sensor Object [0x31] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 SERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single
0x4B	Zero Adjust	Indicates to the device that the current Flow Sensor value is zero flow.	
0x32	Set Full Scale Counts	Defines the Counts value that represents 100% of the calibration full scale	This only affects the way the value of Flow (Attrib. 6) is reported in relation to the Full Scale calibration when the Units (Attrib. 4) is set to Counts.

5.8.3 Service Details

5.8.3.1 Service Set Full Scale Counts

Table 5-41 S-Analog Sensor Object [0x31] Service Set Full Scale Counts

Arguments

Parameter Name	Data Type	Required	Parameter Value	Semantics
Full Scale Counts	INT	Y	-32768 to 32767	The Counts value that defines 100% of the calibration full scale. The value that is passed here in this argument will be reported in attribute Full Scale (Attrib. 10).

Response

Parameter Name	Data Type	Required Value	Parameter	Description
NO SUCCESS RESPONSE DATA				

5.9 S-Analog Actuator Object [0x32]

The S-Analog Actuator is responsible for management of the actuation device controlling the process.

5.9.1 Attributes

Table 5-42 S-Analog Actuator Object [0x32] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
3	Data Type	USINT	Cond.	NV	Defines the Data Type of Valve[6] and other attributes in this object.	Default = INT See Note 'Data Type' below
4	Data Units	ENGUNITS	Cond.	NV	Defines the Engineering Units context of Valve[6] and other attributes in this object.	Default = Counts See Note 'Data Units' below
5	Override	USINT	Set	V	Specifies a direct override of the physical actuator	See Note 'Override' below
6	Valve	Specified by Attrib. 3 & 4	Get	V	The value of the analog output signal used to drive the physical actuator	See Note 'Valve' below
7	Status	BYTE	Get	V	Alarm and Warning status for the S-Analog Actuator Object	See Note 'Status' below
21	Safe State	USINT	Set	NV	Specifies the behavior for the physical actuator in an Operational State other than Executing State	See Note 'Safe State' below
22	Safe Value	Specified by Attrib. 3 & 4	Set	NV	The analog output signal value that is indicated by Attrib. 6 if the Safe State[21] is configured to "Use Safe Value"	Default = 0
99	Subclass	UINT	Get	NV	Identifies a subset of additional instance attributes, services, and behaviors for the Actuator Object	The value of this attribute is 0

5.9.1.1 Note: Data Type

The value of this attribute is limited to INT or REAL. This attribute can only be set when the device is not in the Operational State of Executing. If any of the attributes in this object use the attribute Data Type[3] to define their data type and the attribute is the end point of an active I/O connection, then the access rule for the attribute Data Type[3] becomes Get and thus cannot be set until the I/O connected becomes inactive. Additionally, the inferred data type of an Assembly object (see The "Inferred" Data Type) will modify this value if the Assembly object has been assigned to an active I/O connection.

Example 1: If the value of Valve[6] is being exchanged via the Polled I/O connection, attribute 3 cannot be modified until that Polled I/O Connection is terminated.

Example 2: If Assembly instance 3 is assigned to the Polled I/O connection produce path, the value of attribute 3 will be set to INT when the Polled I/O connection transitions to the Active state. When the Polled I/O connection is terminated, the value of this attribute will remain INT. It will not return to the value prior to commissioning the Polled I/O connection.

5.9.1.2 Note: Data Units

The value of this attribute is limited to the values specified in the Actuator Units table in Appendix C - Data Units. This attribute can only be set when the device is not in the Operational State of Executing. See Example 1 above in Note: Attribute 3.

5.9.1.3 Note: Override

The following table outlines the valid actuator override types

Table 5-43 S-Analog Actuator Object [0x32] Override

Value	State	Description
0	Normal	Actuator is under normal operational control
1	Closed	Actuator is driven fully closed (Valve[6] = 0% or 0 Counts)
2	Open	Actuator is driven fully open (Valve[6] = 100% or 32767 Counts)
3	Hold	Actuator is held to last updated analog output signal prior to assertion of override
4	Safe State	Actuator is driven to the condition specified by the Safe State[21] attribute

5.9.1.4 Note: Valve

To interpret the value of this attribute, it is important to understand the following terms:

Operational Range:
This is the range that is reported by Valve[6]. The operational range of the actuator is full range that the actuator can be driven to move. This corresponds to Valve[6] values of 0 to 100% (0 to 32767 Counts).

Nominal Control Range:
The *nominal control* range is a set of values that the actuator is driven to that maps directly between 0 flow and full scale flow. This set of values is a sub-range within the large operational range of the actuator. Example, the nominal control range for a 0 to 100 sccm device flowing nitrogen could be as follows:

at 0 sccm Actuator = 20% (6553 counts)
at100 sccm, Actauator = 30% (9830 counts)..

Under normal operational control (no override), the actuator generally operates in the nominal control range. The upper end of the control range is not an absolute limit under normal control. The controller will drive the actuator to whatever value necessary to control flow. For example, if an restriction occurred upstream of the device resulting in reduced supply to the device, the controller will drive the actuator beyond the nominal control range to maintain control.

5.9.1.5 Note: Status

The following table defines the bits in this status byte. The value of 1 = Set and 0 = Cleared.

Table 5-44 S-Analog Actuator Object [0x32] Status

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	Low Flow Warning	High Flow Warning	Low Flow Alarm	High Flow Alarm

5.9.1.6 Note: Safe State

The following table outlines valid values for this attribute. This table applies for normally closed and normally open valves.

Table 5-45 S-Analog Actuator Object [0x32] Safe State

Value	State	Description
0	Closed	The actuator will be driven closed and the value of Valve[6] will be 0% or 0 Counts
1	Open	The actuator will be driven open and the value of Valve[6] will be 100% or 32767 Counts
2	Hold	The actuator will be driven to the last updated value of the analog output just prior to the entering of the safe state.
3	Last Value Use Safe Value	The actuator will be driven to the value configured in Safe Value[22] and Valve[6] = Safe Value[22]

5.9.2 Services

Table 5-46 S-Analog Actuator Object [0x32] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 ASERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single

5.9.3 Service Details

NONE

5.10 S-Single Stage Controller Object [0x33]

The S-Single Stage Controller object is responsible for closing the loop between the measured process variable (via the S-Analog Sensor) and the control variable (via the S-Aanalog Acuator).

5.10.1 Attributes

Table 5-47 S-Single Stage Controller Object [0x33] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
3	Data Type	USINT	Cond.	NV	Defines the Data Type of Setpoint[6] and other attributes in this object.	Default = INT See Note 'Data Type' below
4	Data Units	ENGUNITS	Cond.	NV	Defines the Engineering Units context of Setpoint[6] and other attributes in this object	Default = Counts See Note 'Data Units' below
6	Setpoint	Specified by Attrib. 3 & 4	Set	V	The amount of flow the device will control to.	
10	Status	BYTE	Get	V	Alarm and Warning status for the S-Single Stage Controller Object	See Note 'Status' below
19	Constant Time Ramp Rate	UDINT	Set	NV	The amount of time, in milliseconds, the controller will take to "ramp" flow from its current value to its final value as commanded in Setpoint[6]	The value of this attribute is 0
99	Subclass	UINT	Get	NV	Identifies a subset of additional instance attributes, services, and behaviors for the Sensor Object	Default = 0.0 (Disabled)
100	Constant Slope Ramp Rate	REAL	Set	NV	Defines the rate of change in flow per second the controller will maintain in ramping flow from its current value to its final value as commanded in Setpoint[6]	See Note 'Ramp Rates' below
101	Derivative Gain (Kd)	REAL	Set	NV	The gain factor for the derivative term of the PID control compensation	See Note 'PID Gains' below
102	Integral Gain (Ki)	REAL	Set	NV	The gain factor for the integral term of the PID control compensation	See Note 'PID Gains' below
103	Proportional Gain (Kp)	REAL	Set	NV	The gain factor for the proportional term of the PID control compensation	See Note 'PID Gains' below

5.10.1.1 Note: Data Type

The value of this attribute is limited to INT or REAL. This attribute can only be set when the device is not in the Operational State of Executing. If any of the attributes in this object use the attribute Data Type[3] to define their data type and the attribute is the end point of an active I/O connection, then the access rule for the attribute Data Type[3] becomes Get and thus cannot be set until the I/O connected becomes inactive. Additionally, the inferred data type of an Assembly object (see The "Inferred" Data Type) will modify this value if the Assembly object has been assigned to an active I/O connection.

Example 1: If the value of Setpoint[6] is exchanged via the Polled I/O connection, Data Type[3] cannot be modified until that Polled I/O Connection is terminated.

Example 2: If Assembly instance 7 is assigned to the Polled I/O connection consume path, the value of Data Type[3] will be set to INT when the Polled I/O connection transitions to the Active state. When the Polled I/O connection is terminated, the value of this attribute will remain INT. It will not return to the value prior to commissioning the Polled I/O connection.

5.10.1.2 Note: Data Units

The value of this attribute is limited to the values specified in the Volumetric Flow Units Table and Mass Flow Units table in Appendix C - Data Units. This attribute can only be set when the device is not in the Operational State of Executing. See Example 1 above in Note: Attribute 3.

5.10.1.3 Note: Status

The following table defines the bits in this status byte. The value of 1 = Set and 0 = Cleared.

Table 5-48 S-Single Stage Controller Object [0x33] Status

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	Controller Warning	Controller Warning

5.10.1.4 Note: Ramp Rates

Constant Time Ramp Rate[19] and Constant Slope Ramp Rate[100] functions are mutually exclusive. The setting of one of these attributes to enable a ramp function will disable the other ramp function by clearing its corresponding attribute. Example, if Constant Slope Ramp Rate[100] is set to 1.3, Constant Time Ramp Rate will automatically be cleared (set to 0), disabling the constant time ramp function. Setting both attributes to 0 disables all ramp functionality.

5.10.1.5 Note: PID Gains

The MFC Device utilizes a PID compensator in the control loop as a means of optimizing and tuning control. The attributes are scalar numbers used as multipliers for their respective P, I, and D terms.

5.10.2 Services

Table 5-49 S-Single Stage Controller Object [0x33] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 ASERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single

5.10.3 Service Details

NONE

5.11 S-Gas Calibration Object [0x34]

The S-Gas Calibration object defines characteristics associated with linearization/compensation of a gas flow sensor.

5.11.1 Attributes

Table 5-50 S-Gas Calibration Object [0x33] Attributes

Instance 0

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
NO INSTANCE 0 ATTRIBUTES ARE SUPPORTED						

Instance 1..n

Attrib ID	Name	Data Type	Access Rule	NV	Description	Notes
3	Gas Standard Number	UINT	Cond.	NV	The gas type number assigned to this gas The instance of the S-Analog Sensor	Default = 0, no gas type specified. See Note 'Gas Standard Number' below
4	Valid Sensor Instance	UINT	Cond.	NV	Object that this Gas Calibration Object valid	Default = 1
6	Full Scale	STRUCT of::	Get	V	The full-scale flow this object was calibrated to.	
	Amount	REAL			The amount of gas corresponding to full scale	
	Units	ENGUNITS			The engineering units assigned to the above	
10	Gas Correction Factor	REAL	Get	V	A simple scalar number used to correct the flow sensor reading	See Note 'Gas Correction Factor' below
95	Calibration Pressure	REAL	Get	NV	The gas pressure, in Pa, under which this calibration was performed	Default = 101325 Pa
96	Calibration Temperature	REAL	Get	NV	The gas temperature, in kelvin, under which this calibration was performed	Default = 273.15 K
99	Subclass	UINT	Set	NV	Identifies a subset of additional instance attributes, services, and behaviors for the S-Gas Calibration Object	The value of this attribute is 1 = Standard T & P subclass
100	Gas Correction Type	USINT	Set	NV	The Gas Correction Method this calibration utilizes	See Note 'Gas Correction Type' below
101	Customer Full Scale	REAL	Set	NV	A user configurable attribute to rescale the Full Scale value of this calibration object	See Note 'Customer Full Scale' below

5.11.1.1 Note: Gas Standard Number

The Gas Standard Number as defined by SEMI publication SEMI E52-0298, "Practice for Referencing Gases Used in Digital Mass Flow Controllers."

5.11.1.2 Note: Gas Correction Factor

This correction factor is used to correct for the sensor value if this calibration instance was calibrated using a calibration gas that is different from the gas this calibration instance represents.

Example: If this gas calibration instance represents a Hydrogen calibration, but the calibration was performed using Nitrogen, a gas correction factor is used to correct for property differences between Hydrogen and Nitrogen.

5.11.1.3 Note: Gas Correction Type

The Delta Series MFC/MFM supports 4 types of sensor correction methods to linearize the sensor signal. This attribute indicates the method being employed for this calibration instance.

Table 5-51 S-Gas Calibration Object [0x33] Gas Correction Type

Attribute Value	Correction Method	Description
0	None	No correction method employed. The flow value reported is a raw, non-linearized signal.
1	Empirical	
2	Linear Gas Factor	
3	Tru-Cal™	Employs the Brooks Instrument Tru-Cal™ method of sensor linearization

5.11.1.4 Note: Customer Full Scale

This attributes provides a means to rescale the Full Scale value of this calibration instance. By default, it will equal the Full Scale[6] attribute.

Note:

There is no limitation on changing this attribute. It is recommended that the customer consult with a Brooks Instrument service representative or engineer about their particular application before changing this attribute.

5.11.2 Services

Table 5-52 S-Gas Calibration Object [0x33] Services

Instance 0

Service Code	Service Name	Service Description
NO INSTANCE 0 ASERVICES ARE SUPPORTED		

Instance 1

Service Code	Service Name	Service Description	Details
0x0E	Get Attribute Single	Returns the contents of the specified attribute	Appendix E: Get Attribute Single
0x10	Set Attribute Single	Sets the contents of the specified attribute with the value passed with this service	Appendix E: Set Attribute Single
0x4B	Get All Instances	Returns all S-Gas Calibration Instance objects	Get All Instances

5.11.3.1 Service Details

Table 5-53 S-Gas Calibration Object [0x33] Service Details

Arguments

Parameter Name	Data Type	Required	Parameter Value	Description
NONE				

Response

Parameter Name	Data Type	Required	Parameter Value	Semantics
Size of List	UINT	Y		Specifies the number of elements in the following ARRAY
List of Gas	ARRAY of:	Y		Supported List. This list is an array of structures
Calibrations	STRUC of:			
Calibration Instance ID	UINT			S-Gas Calibration Object Instance ID
Gas Standard Number	UINT			Gas Standard Number
Sensor Instance ID	UINT			Valid Sensor Instance

6-1 Troubleshooting

Table 6-1 Troubleshooting

Problem	Possible Causes
Scanner is actively scanning the network but the NET LED is flashing green	• The MAC ID of the device is not programmed in the scan list of the scanner or does not match a MAC ID already programmed in the scanner
Scanner is actively scanning the network but the NET LED on the device alternates between solid green state for a short time, then flashing green state for a short time.	• Produce/Consume I/O data size mismatch. The I/O data sizes programmed in the scanner is different from the Produce/Consume I/O data size in the device
When power is applied to the device, network communications are disrupted	• The Power Supply for the network is undersized to handle the additional load, causing the power supply to "droop" • The baud rate setting on the device does not match the network baud rate.
When power is applied to the device, the NET LED goes solid red	• The MAC ID of the device is the same as the MAC ID of another device on the network. • The Baud Rate switch setting on the device does not match the network baud rate. • The device was forced off the bus because of communications error (a.k.a. Bus Off condition). If the problem persists, contact Brooks service.
When power is applied to the device, the NET LED remains off	• The device is the only device connected on the network. • The Baud Rate switch setting on the device does not match the network baud rate.
The NET LED is flashing red	• The I/O connection is timed out.
The device never comes out of Self-Test (MOD LED continually flashes red/green).	• Cycle power to the device. If problem persists, contact Brooks service.
The MOD LED is solid RED	• An internal fault in the device was detected. Cycle power to the device. If the problem persists, contact Brooks service.
When power is applied to the device, both LED's continually cycle through the power-on LED test (each LED will cycle red then green).	• An internal fault has occurred in the device on power-up. Contact Brooks service.
A setpoint value is being sent to the device but the MFC fails control flow (i.e. no actuator movement, low or no flow indication)	• Check Data Units for the Sensor Object and Controller Object. Make sure the Data Units for both match and that they configured to the desired values.

THIS PAGE WAS INTENTIONALLY LEFT BLANK

7-1 Appendix A - Assembly Object Detail

The following table details each of the Assembly Objects supported by the MFC. The DeviceNet specification defines Input and Output relative to the network (i.e. The data being PRODUCED from the device as an INPUT into the network or the data is being an CONSUMED by the device is an OUTPUT from the network).

7.1.1 Summary Table of Input Assemblies

Table 7-1 Summary Table of Input Assemblies

Instance ID	Data Description	Size(bytes)	Data Type
1	Flow	2	INT
2	Status, Flow	3	INT
3	Status, Flow, Valve	5	INT
4	Status, Flow, Setpoint	5	INT
5	Status, Flow, Setpoint, Valve	7	INT
6	Status, Flow, Setpoint, Actuator Override, Valve	8	INT
13	Flow	4	REAL
14	Status, Flow	5	REAL
15	Status, Flow, Valve	9	REAL
16	Status, Flow, Setpoint	9	REAL
17	Status, Flow, Setpoint, Valve	13	REAL
18	Status, Flow, Setpoint, Actuator Override, Valve	14	REAL

7.1.2 Summary Table of Output Assemblies

Table 7-2 Summary Table of Input Assemblies

Instance ID	Data Description	Size(bytes)	Data Type
7	Setpoint	2	INT
8	Actuator Override, Setpoint	3	INT
19	Setpoint	4	REAL
20	Actuator Override, Setpoint	5	REAL

7.1.3 Assembly Details

7.1.3.1 Input Assembly 1 - Detail

Table 7-3 Input Assembly 1 - Detail

Assembly Instance	1
Data Size	2 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	Flow	[0x31-1-6]
Byte 1		

7.1.3.2 Input Assembly 2 - Detail

Table 7-4 Input Assembly 2 - Detail

Assembly Instance	**2**
Data Size	**3 bytes**
Type	**Input**

Definition of Bytes	Description	EPATH
Byte 0	Status	[0x30-1-12]
Byte 1	Flow	[0x31-1-6]
Byte 2		

7.1.3.3 Input Assembly 3 - Detail

Table 7-5 Input Assembly 3 - Detail

Assembly Instance	**3**
Data Size	5 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3	**Valve**	**[0x32-1-6]**
Byte 4		

7.1.3.4 Input Assembly 4 - Detail

Table 7-6 Input Assembly 4 - Detail

Assembly Instance	**4**
Data Size	5 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3	**Setpoint**	**[0x33-1-6]**
Byte 4		

7.1.3.5 Input Assembly 5 - Detail

Table 7-7 Input Assembly 5 - Detail

Assembly Instance	**5**
Data Size	7 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1 Byte 2	**Flow**	**[0x31-1-6]**
Byte 3 Byte 4	**Setpoint**	**[0x33-1-6]**
Byte 5 Byte 6	**Valve**	**[0x32-1-6]**

7.1.3.6 Input Assembly 6 - Detail

Table 7-8 Input Assembly 6 - Detail

Assembly Instance	**6**
Data Size	8 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1 Byte 2	**Flow**	**[0x31-1-6]**
Byte 3 Byte 4	**Setpoint**	**[0x33-1-6]**
Byte 5	**Actuator Override**	**[0x32-1-5]**
Byte 6 Byte 7	**Actuator Drive Value**	**[0x32-1-6]**

7.1.3.7 Output Assembly 7 - Detail

Table 7-9 Input Assembly 7 - Detail

Assembly Instance	**7**
Data Size	2 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0 Byte 1	**Setpoint**	**[0x33-1-6]**

7.1.3.8 Output Assembly 8 - Detail

Table 7-10 Input Assembly 8 - Detail

Assembly Instance	**8**
Data Size	3 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Actuator Override**	**[0x32-1-5]**
Byte 1	**Setpoint**	**[0x33-1-6]**
Byte 2		

7.1.3.9 Input Assembly 13 - Detail

Table 7-11 Input Assembly 13 - Detail

Assembly Instance	**13**
Data Size	4 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Flow**	**[0x31-1-6]**
Byte 1		
Byte 2		
Byte 3		

7.1.3.10 Input Assembly 14 - Detail

Table 7-12 Input Assembly 14 - Detail

Assembly Instance	**14**
Data Size	5 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3		
Byte 4		

7.1.3.11 Input Assembly 15 - Detail

Table 7-13 Input Assembly 15 - Detail

Assembly Instance	**15**
Data Size	9 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3		
Byte 4		
Byte 5	**Valve**	**[0x32-1-6]**
Byte 6		
Byte 7		
Byte 8		

7.1.3.12 Input Assembly 16 - Detail

Table 7-14 Input Assembly 16 - Detail

Assembly Instance	**16**
Data Size	9 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3		
Byte 4		
Byte 5	**Setpoint**	**[0x33-1-6]**
Byte 6		
Byte 7		
Byte 8		

7.1.3.13 Input Assembly 17 - Detail

Table 7-15 Input Assembly 17 - Detail

Assembly Instance	**17**
Data Size	13 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3		
Byte 4		
Byte 5	**Setpoint**	**[0x33-1-6]**
Byte 6		
Byte 7		
Byte 8		
Byte 9	**Valve**	**[0x32-1-6]**
Byte 10		
Byte 11		
Byte 12		

7.1.3.14 Input Assembly 18 - Detail

Table 7-16 Input Assembly 18 - Detail

Assembly Instance	**18**
Data Size	14 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Status**	**[0x30-1-12]**
Byte 1	**Flow**	**[0x31-1-6]**
Byte 2		
Byte 3		
Byte 4		
Byte 5	**Setpoint**	**[0x33-1-6]**
Byte 6		
Byte 7		
Byte 8		
Byte 9	**Actuator Override**	**[0x32-1-5]**
Byte 10	**Actuator Drive Value**	**[0x32-1-6]**
Byte 11		
Byte 12		
Byte 13		

7.1.3.15 Output Assembly 19 - Detail

Table 7-17 Input Assembly 19 - Detail

Assembly Instance	**19**
Data Size	4 bytes
Type	Output

Definition of Bytes	Description	EPATH
Byte 0	**Setpoint**	**[0x33-1-6]**
Byte 1		
Byte 2		
Byte 3		

7.1.3.16 Output Assembly 20 - Detail

Table 7-18 Input Assembly 20- Detail

Assembly Instance	**20**
Data Size	5 bytes
Type	Input

Definition of Bytes	Description	EPATH
Byte 0	**Actuator Override**	**[0x32-1-5]**
Byte 1	**Setpoint**	**[0x33-1-6]**
Byte 2		
Byte 3		
Byte 4		

7.2 Appendix B - Data Type Definitions

The following table list DeviceNet data types used throughout this manual and in the DeviceNet specification. The column C/C++ Encoding is given as a comparative common example reference.

Table 7-19 DeviceNet Data Type Definitions

Data Type	Size (bytes)	Description	Range	C/C++ Keyword
BOOL	1	A true/false represented as 0 = false and 1 = true	0 and 1	bool
SINT	1	An 8-bit signed integer value	-128 to 127	char
USINT	1	An 8-bit unsigned integer value	0 to 255	unsigned char
INT	2	A 16-bit signed integer value	-32768 to 32767	short int
UINT	2	A 16-bit unsigned integer value	0 to 65535	unsigned short int
DINT	4	A 32-bit signed integer value	-2147483648 to 2147483647	int
UDINT	4	A 32-bit unsigned integer	0 to 4294967296	unsigned int
REAL	4	An IEEE single precision floating point number	-3.8E38 to 3.8E38	float
DREAL	8	An IEEE double precision floating point number		Long
ENGUNIT	1	An enumerated value representing an engineering unit of measure	4096 - 65535	N/A
BYTE	1	An 8-bit Bitfield	N/A	N/A
SHORT STRING	Up to 128 bytes	A character array where the first byte is the number of characters in the array, and the subsequent bytes contain the ASCII characters. This is not a NULL terminated string.	N/A	N/A

7.3 Appendix C - Data

Table 7-20 Data

Volumetric Flow Units

Description	Symbol	Units Code	
		Decimal	Hex
Counts		4097	0x1001
Percent	%	4103	0x1007
Standard Cubic Centimeter per Minute	SCCM	5120	0x1400
Standard Liter per Minute	SLM	5121	0x1401
Cubic Foot per Minute	CFM	5122	0x1402
Cubic Meter per Second	m^3/s	5125	0x1405
Liter per Second	L/s	5126	0x1406
Milliliter per Second	mL/s	5127	0x1407
Gallon per Second	GPS	5128	0x1408
Gallon per Minute	GPM	5129	0x1409
Gallon per Hour	GPH	5130	0x140A

Mass Flow Units

Description	Symbol	Units Code	
		Decimal	Hex
Counts		4097	0x1001
Percent	%	4103	0x1007
Kilogram per Second	kg/s	5124	0x1404
Pound per Second	lb/s	5131	0x140B
Pound per Minute	lb/min	5132	0x140C
Pound per Hour	lb/hr	5133	0x140D
Milligrams per Minute	mg/M	5134	0x140E
Grams per Minute	g/M	5135	0x140F
Kilograms per Hour	kg/H	5136	0x1410

Actuator Units

Description	Symbol	Units Code	
		Decimal	Hex
Counts		4097	0x1001
Percent	%	4103	0x1007

7.4 Appendix D - LED Flash Codes

The following table assumes power has been applied to the network and the device has been properly connected to the network.

Table 7-21 LED Flash Codes

"NET" LED Flash Codes

Flash Code	Description
Off	The device is the only node on the network
Flashing Green	Baud Rate and MAC ID are configured correctly. The device recognizes the network but no connection has been made to the device
Solid Green	A Connection has been established to the device
Flashing Red	An I/O Connection to the device has timed out
Solid Red	The Baud Rate Switch is set to a different baud rate than the network **OR** The MAC ID of the device is identical to another node on the network **OR** The device has gone Bus Off, indicating network communication errors.

"MOD" LED Flash Codes

Flash Code	Description
Flashing Red-Green	The device is in the Self-Test mode
Solid Green	All self-tests have passed. No faults have been detected
Flashing Red	A recoverable fault has been detected or the device has been commanded into the Abort state
Solid Red	An unrecoverable fault has occurred

7.5 Appendix E – Service Summary and Details

7.5.1 Get Attribute Single

7.5.1.1 SERVICE Parameters

Table 7-22 Service Parameters

Parameter Name	Data Type	Required	Description	Default
Attribute ID	USINT	Y	The attribute ID of the attribute to be read.	None

7.5.1.2 Success Response Data

Table 7-23 Success Response Data

Return Value	Data Type	Description
Attribute Value	The Data Type of the Attribute being read	

7.5.2 Set Attribute Single

7.5.2.1 SERVICE Parameters

Table 7-24 Service Parameters

Parameter Name	Data Type	Required	Description	Default
Attribute ID	USINT	Y	The attribute ID of the attribute to be read.	None
Attribute Value	(Equivalent to the data type of the Attribute)	Y	The value to which the attribute will be set	None

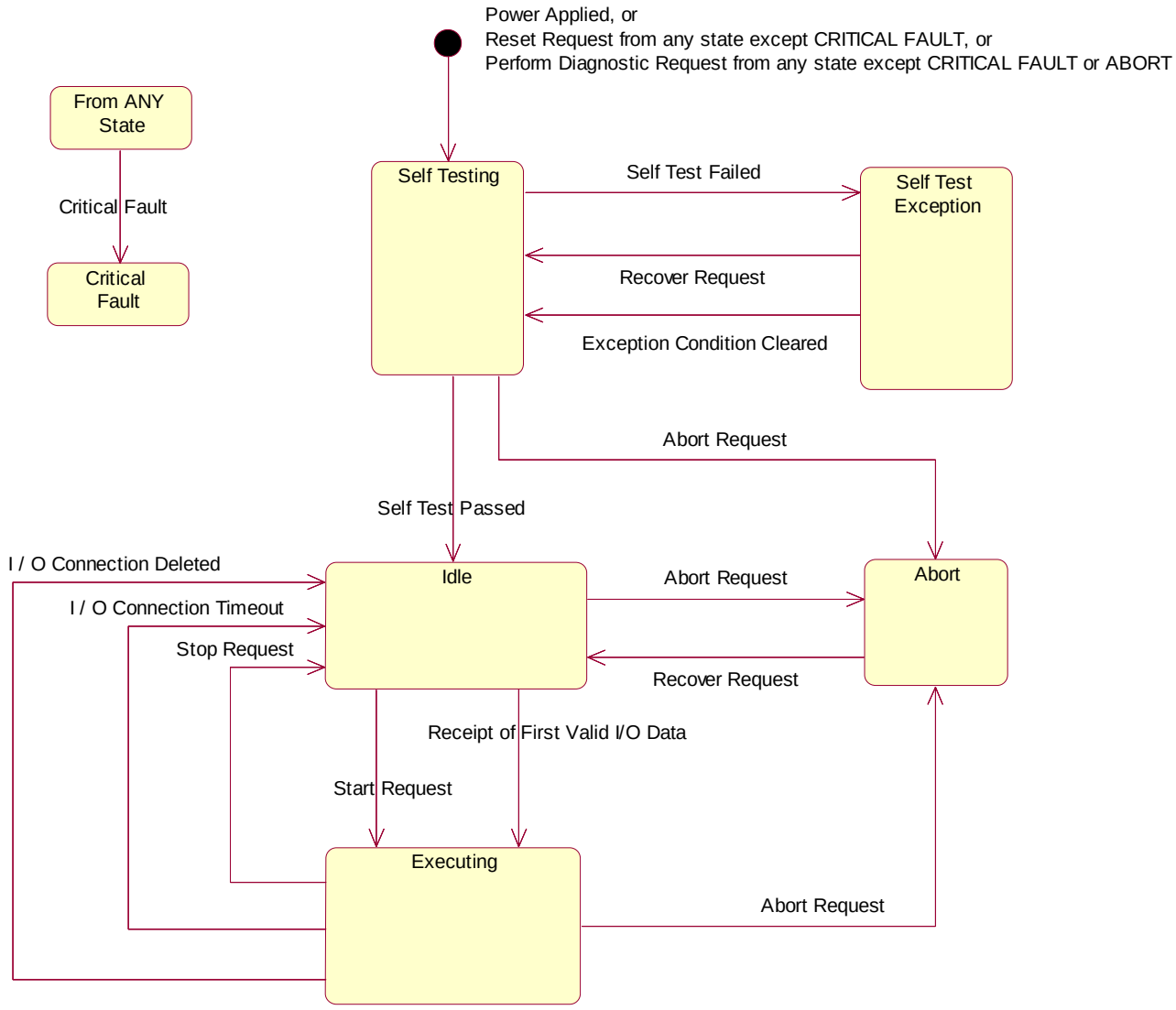
7.5.2.2 Success Response Data

Table 7-25 Success Response Data

Return Value	Data Type	Description
NO SUCCESS RESPONSE DATA		

7.6 Appendix F - State

7.6.1.1 Device Supervisor State Machine Diagram



This manual is a supplement to the Delta Series Mass Flow Controller This section is intended as a brief overview of DeviceNet terminology used throughout this manual.

Assembly

An Assembly is a Class that defines a collection of EPATH(s). This collection allows multiple attributes to be virtually accessed all at once. Each instance of an Assembly defines a unique set of EPATH(s).
Example: The movement of data in an I/O type Connection is defined by the Produce and Consume Path attributes discussed above (see EPATH). Only one EPATH can be assigned to each of these attributes, thus limiting the amount of data that can be moved through an I/O type Connection. The solution to move more data through the I/O type Connection is to assign the Produce and Consume EPATH attributes of the Connection to "point" to an Assembly, which in turn contains multiple EPATH's.

Attribute

A Parameter or Data Item that may be read or written and is used for the purpose of configuration or is used to obtain information.
Example: The attribute *Data Units* defines the engineering units flow will be reported in. The attribute *Value* indicates the current flow through the device. Attributes can be read/write or read only.

Baud Rate

The data rate, or speed, that the device will communicate. Setting the Baud Rate switch found on the top cover of the Delta Series MFC/MFM will configure this. The choices are 125k, 250k, and 500k.

COS

The Change-of-State (COS) I/O connection is an extension of the Cyclic I/O connection. The M/S COS I/O Connection establishes a data exchange between a Master and Slave devices whereby the device will produce a message when a change in value of the I/O data has been detected. At a minimum, the device will produce a message at a frequency equivalent to the EPR (see Cyclic) if no change in I/O data was detected.

Class

A logical collection of related Attributes that define a particular function and/or behavior.
Example: The class *S-Analog Sensor* contains information about configuring a sensor, the current status of the sensor, and/or the current value of what is being sensed.

Cyclic

The M/S Cyclic I/O Connection establishes a data exchange between a Master and Slave devices whereby the device produces an I/O message periodically at a frequency defined by the EPR for the connection.
Example: If the EPR for the Cyclic I/O Connection is set to 500 msec, the device will produce a message every 500 msec. If so configured, the Master can respond with an acknowledge message.

Connection

A logical link between two devices by which messages are transferred. A device can have 1 or more simultaneous Connections. In most typical applications, data is exchanged using 1 Explicit Connection and 1 I/O Connection. The DeviceNet specification has defined a pre-configured set of Connections referred to as the Master/Slave (M/S) Connection Set.

Device Profile

A specification that defines a set of DeviceNet objects that uniquely represents a particular device of that type or class. The device profile can further define attributes, services, assemblies, etc. that a device must support to be considered part of that type or class of device. These profiles are found in the ODVAä specification, Vol. 2. The Delta Series MFC/MFM conforms to the MFC device profile.

EDS

The Electronic Data Sheet (EDS) is a specially formatted text description for a device that describes the I/O characteristics and configurable parameters that are accessible via the DeviceNet network. EDS files can be read by configuration software used to configure DeviceNet networks (ex: RsNetworx from Allen-Bradley)

EPATH (Formerly known as PATH)

An EPATH is a unique identifier (sometimes referred to as a pointer) comprised of a Class ID, an Instance ID, and an Attribute ID. Some Classes have EPATH attributes that point to a particular data item. An example of this would be the Connection Class that contains two attributes, Produce Path and Consume Path. These attributes define where incoming data is sent to, and outgoing data comes from. I/O Connections make use of these two particular attributes.

Expected Packet Rate (EPR)

The EPR is an attribute in the Connection Class that defines the maximum amount of time (in msec) messages should be received by the Connection (implementation of this value is dependent upon the Connection type, Explicit or I/O but the behavior is the same in all Connection types). If the time between received messages for that connection exceeds the EPR, the Connection times out, which may result in the Connection being released by the device.

Explicit Connection

An Explicit Connection dictates a *request* and *response* exchange between two devices. The device sending the request must get a response from the device receiving the request message. Embedded in the Explicit Message is information about the Class, Instance, Attribute, Service, and any service data needed to process the message. As a result, processing of Explicit Messages generally takes longer than I/O messaging. This is why Explicit Connections are typically used for commissioning/configuration.

I/O Connection

I/O Connections are used for the exchange of data only. How a device processes the data and/or responds with data via an I/O Connection is defined within the Produce and Consume Path attributes of the I/O Connection instance (See EPATH and Assembly above).

Instance

An instance of a Class is a particular invocation of a Class (sometimes referred to as an Object). An Instance of a Class is *unique* in describing the behavior for a particular kind of object. Each instance of the class contains the same set of attributes defined by the class. The uniqueness of the instance is defined by the attribute values.
Example: Assume that a device contains two sensors, one to measure flow, the other to measure temperature. To access information about one or the other sensor, two Instances of the class *S-Analog Sensor* class would

need to exist. Each sensor would have the same attribute set because they were both created from the class S-Analog Sensor, but the values in each attribute set would be unique to the sensor to describe flow or to

Example: The class Connection contains information about configuring a Connection. With most kinds of DeviceNet devices, multiple Connections are established within a device to exchange information. In most cases one Explicit type Connection is created to configure the device, another I/O type Connection is created to transfer larger amounts of data. Each type of Connection is created from the Connection class. Each Connection has unique information in its Attribute set that defines the behavior of the Connection et.al. an Explicit type Connection or an I/O type Connection.

MAC ID

The Media Access Control Identifier, more commonly knows as the network address of the device. Setting the MAC ID switches found on the top cover of the Delta Series MFC/MFM will configure this address. The range of addresses is 0 to 63.

Master / Slave

The relationship between one controlling device (Master) and one or more subordinate devices (Slaves). This is a typical application of a DeviceNet network where one device has the capability to establish Connections with more than one device. The "Master" then sends to each device control data and receives from each "Slave" device status and/or operational data. An example would be a "Master" device sending a setpoint to one or more mass flow controllers (Slaves), and each controller would respond with flow data.

M/S Polled I/O Connection

The M/S Polled I/O Connection establishes a data exchange between a Master and Slave devices whereby the Master device sends an I/O message with 0 or more data to a Slave device, and the Slave device must respond with 0 or more data bytes.

Polling

A request / response method of exchanging data.

Safe State (Safe Mode)

A operational mode or state that is considered "safe" whereby the normal controller process is shut down and mechanical and sensing mechanisms are placed in a safe condition.

Service

A service is a pre-defined action that a Class provides. The most commonly used services are used to configure the device such as Get Attribute (0x0E) or Set Attribute (0x10). Other types of services may directly affect the behavior of a Class (or Object) such as Reset (0x5), Stop (0x6), or Start (0x7). There are many more services not listed here and each Class specifies which Services it supports.

Scanning

A technique of exchanging data with devices on a network whereby the Master device (scanner) polls each device on a network sequentially, continuing through the list of all devices connected to the network before starting the process again.

Brooks® Delta Class DeviceNet MFC's/MFM's

LIMITED WARRANTY

Seller warrants that the Goods manufactured by Seller will be free from defects in materials or workmanship under normal use and service and that the Software will execute the programming instructions provided by Seller until the expiration of the earlier of twelve (12) months from the date of initial installation or eighteen (18) months from the date of shipment by Seller. Products purchased by Seller from a third party for resale to Buyer ("Resale Products") shall carry only the warranty extended by the original manufacturer.
All replacements or repairs necessitated by inadequate preventive maintenance, or by normal wear and usage, or by fault of Buyer, or by unsuitable power sources or by attack or deterioration under unsuitable environmental conditions, or by abuse, accident, alteration, misuse, improper installation, modification, repair, storage or handling, or any other cause not the fault of Seller are not covered by this limited warranty, and shall be at Buyer's expense.
Goods repaired and parts replaced during the warranty period shall be in warranty for the remainder of the original warranty period or ninety (90) days, whichever is longer. This limited warranty is the only warranty made by Seller and can be amended only in a writing signed by an authorized representative of Seller.

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.

CUSTOMER SEMINARS AND TRAINING

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

Please contact your nearest sales representative for more details.

HELP DESK

In case you need technical assistance:

Americas	☎ 1 888 554 FLOW
Europe	☎ +31 (0) 318 549 290
Asia	☎ +81 (0) 3 5633 7100

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

TRADEMARKS
Adobe Acrobat ..... Adobe Systems, Inc.
Brooks ..... Brooks Instrument, LLC
DeviceNet ..... Open DeviceNet Vendors Association, Inc.
Labview ..... National Instrument, Inc.
ODVA ..... Open DeviceNet Vendors Association, Inc.
TruCal ..... Brooks Instrument, LLC

ISO 9001 QUALITY SYSTEM

Brooks Instrument
407 West Vine Street
P.O. Box 903
Hatfield, PA 19440-0903 USA
T (215) 362 3700
F (215) 362 3745
E-Mail BrooksAm@BrooksInstrument.com
www.BrooksInstrument.com

Brooks Instrument
Neonstraat 3
6718 WX Ede, Netherlands
T +31 (0) 318 549 300
F +31 (0) 318 549 309
E-Mail BrooksEu@BrooksInstrument.com

Brooks Instrument
1-4-4 Kitasuna Koto-Ku
Tokyo, 136-0073 Japan
T +81 (0) 3 5633 7100
F +81 (0) 3 5633 7101
E-Mail BrooksAs@BrooksInstrument.com

BROOKS® INSTRUMENT