ANALYT MESSTECHNIK



COMMUNICATION GUIDE

358 Series IO-Link

Wiring, process data, variables and commands for reading and controlling 358 Series flow and pressure devices from an IO-Link master.

AMTC-CG-0005

Rev. 1 · October 2026

About this guide

This guide explains how to use a 358 Series device with IO-Link: how to wire it to an IO-Link master, which process data it exchanges, which variables you can read and write, and which commands it accepts. It covers the mass flow meters (M) and controllers (MC), the pressure gauges (P, EP) and controllers (PC, EPC), and the liquid flow meters (L) and controllers (LC).

Each command states the firmware version that introduced it: 10v18.0 for every command of this guide. Some variables and commands exist only on certain devices, for example controllers or mass flow devices; this is stated with each of them.

If you are connecting a device for the first time, start with Connecting to an IO-Link master (p. 7). If you only need indices and command IDs, go straight to the Command and register reference (p. 12).

Most devices with IO-Link also have an M8 connector for RS-232 or RS-485 serial communication. The serial commands are described in the *358 Series serial interface (RS-232/RS-485) — Communication guide* (AMTC-CG-0002).

Conventions in this guide:

- Index and subindex numbers, command IDs and values are decimal. Hexadecimal values are written with the prefix `0x`.
- Status names and values you send or read exactly as shown are set in monospace type, for example `SUCCESS`.
- Keys, menu entries and screen names on the device display are printed like this: **`MENU`** › **`SETUP`** › **`IO-Link`**.

Abbreviations

Abbreviation	Meaning
C/Q	IO-Link data line
FOPDT	First order plus dead time (a model of the control response)
IODD	Device description file of an IO-Link device
L+, L–	24 V power supply of the IO-Link port, positive and negative
LED	Light-emitting diode
NaN	Not a number (IEEE 754 invalid float value)
P24, N24	Isolated valve power supply of a class B port, positive and negative
PD2I, PDF	The two closed-loop control algorithms of the controllers
PLC	Programmable logic controller
ro, rw	Read only, read and write (access of a variable)

Safety notices in this guide

NOTICE

Practices not related to personal injury — damage to the product, the process or the sample.

Note
Information that helps understanding; never a hazard.

Contact and support

ANALYT-MTC Messtechnik GmbH

German Headquarters

Klosterrunsstraße 18, 79379 Müllheim
Germany

+49 7631 5545

info@analyt-mtc.de

French Subsidiary

175 Avenue d'Alsace, 68000 Colmar
France

+33 3 89 41 47 78

info@analyt-mtc.fr

www.analyt-mtc.com

Before you call

Have the following at hand so that Analyt-MTC can help you quickly:

- the model and serial number of the device (on its label);
- whether the device and the master port are class A or class B;
- the variable index or command ID you used, the argument you sent and the status or event code you received.

Document information

Title	358 Series IO-Link — Communication guide
Document number	AMTC-CG-0005
Revision	Rev. 1 — October 2026
Applies to	358 Series flow and pressure meters and controllers with IO-Link
Language	English
Related documents	358 Series Modbus RTU and Modbus TCP/IP — Communication guide (AMTC-CG-0001); 358 Series serial interface (RS-232/RS-485) — Communication guide (AMTC-CG-0002); 358 Series L Liquid Flow Meters — Operating manual (AMTC-MN-0002); 358 Series MC Mass Flow Controllers — Operating manual (AMTC-MN-0005); 358 Series P Pressure Transducers — Operating manual (AMTC-MN-0006); 358 Series PC Pressure Controllers — Operating manual (AMTC-MN-0007); 358 Series Dual-Valve Pressure Controllers — Operating manual (AMTC-MN-0001)

Revision history

Rev.	Date	Change
1	October 2026	First issue.

Information in this guide is subject to change without notice. © 2026 ANALYT-MTC Messtechnik GmbH. All rights reserved.

Contents

1 **Interface and wiring** 5
- Connectors and status LED 5
- Class A and class B devices and masters 5
- Pin-outs 6
- Connecting to an IO-Link master 7
- IO-Link menu on the display 8

2 **Protocol conventions** 10
- Process data and variables 10
- Conventions used in this guide 10
- Process data 10
- Access and stored settings 11

3 **Command and register reference** 12
- Standard variables 12
- Device status and events 12
- Variables 13
- Commands 17
  - Command function 18
  - Data readings 18
  - Control 19
  - Control response analysis 24
  - Control optimisation 28
  - Administrative 31

4 **Examples** 36
- How to read the examples 36
- Example 1: tare the flow reading 36
- Example 2: repeat the same command 36
- Example 3: show the temperature of a mass flow meter in °F 36
- Example 4: create a mix of 50 % nitrogen and 50 % argon 37
- Example 5: limit the set point ramp to 1 % of full scale per second 37

5 **Error handling** 38
- Symptoms and remedies 38

A **Statistics** 40

B **Engineering units** 42

C **Data sources** 45

D **Gas numbers** 46

CHAPTER 1

Interface and wiring

Connectors and status LED

A device with IO-Link has one M12 port and one M8 port (Figure 1). EP and EPC devices have a 5-pin M12 port and no M8 port.

The M12 port is the IO-Link port: class A with 4 pins, or class B with 5 pins. A class B port uses an isolated power supply for the valve, most often on devices with large valves.

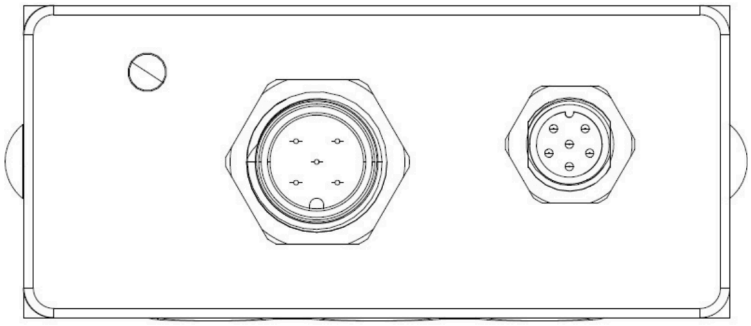


Figure 1. Top of the device: M12 IO-Link port (centre) and M8 port (right).

The LED on the top of the device shows its status. The colours and their meanings follow the NAMUR NE 107 recommendation, and each matches a value of the *Device Status* standard variable (Standard variables (p. 12)).

1-1 Status LED

LED colour	Status	Device status	Description
Green	Normal operation	**0**	Normal function
Red	Failure	**4**	The device has detected a sensor or other malfunction.
Orange	Check function	**3**	The device is in a mode other than normal operation.
Yellow	Out of specification	**2**	At least one reading is outside its specified measuring range or environmental conditions.
Blue	Maintenance required	**1**	Maintenance required

Class A and class B devices and masters

IO-Link devices and masters are either class A or class B.

- A **class A master** supplies enough power for class A devices: meters and controllers with small valves.
- A **class B master** also provides an isolated power supply on pin 5. Class B devices, for example controllers with large valves, need a class B master.

A class B master also supports class A devices. It can be the better choice even for controllers with small valves, which a class A master could drive, because it keeps the valve power on a separate, isolated supply.

Note
Many class A masters supply enough current for controllers with large valves. A master shares its power among all its devices, however, so this can reduce the number of IO-Link devices that one master can support.

Pin-outs

M12 IO-Link connector

The IO-Link connector of most devices has the pin-out below.

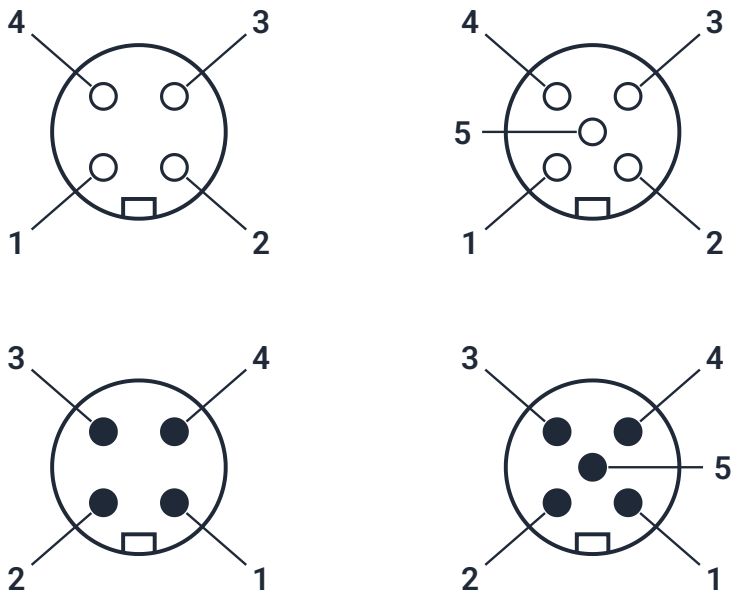


Figure 2. M12 connectors: class A, 4 pins (left) and class B, 5 pins (right). Top row: male connector on the device; bottom row: female connector on the cable.

1-2 4-pin M12, class A

Pin	Function
1	24 V power supply (L+)
2	Not connected
3	24 V power supply (L−)
4	IO-Link data (C/Q)

1-3 5-pin M12, class B

Pin	Function
1	24 V power supply (L+)
2	Valve power supply (P24)
3	24 V power supply (L−)
4	IO-Link data (C/Q)
5	Valve power supply (N24)

EP and EPC devices have a 5-pin M12 connector with this pin-out:

1-4 5-pin M12, EP and EPC devices

Pin	Function
1	24 V power supply (L+)
2	MFG 1, reserved (compatible with class A or class B)
3	24 V power supply (L−)

Pin	Function
4	IO-Link data (C/Q)
5	MFG 2, reserved (compatible with class A or class B)

M8 factory connector

Most devices also have a factory-fitted M8 connector, for applications that use RS-232 or RS-485 serial communication or need an analogue interface.

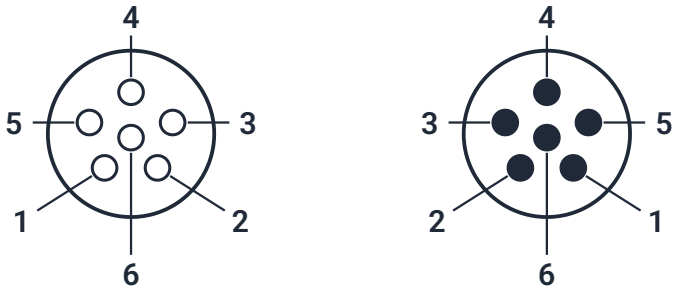


Figure 3. 6-pin M8 connector: male connector on the device (left), female connector on the cable (right).

NOTICE

Power applied through the M12 and the M8 connector at the same time can damage the device. Connect power to only one of the two connectors.

1-5 6-pin M8

Pin	Function
1	Power in: powers the device; see the specification sheet of the device for details. Read the notice above.
2	Analogue out: 0–5 V DC output signal. Optional: 1–5 V DC, 0–10 V DC, 4–20 mA.
3	Ground: common ground for power, digital communication, analogue signals and alarms.
4	RS-232 Tx: sends RS-232 (Tx) or RS-485 A (−) signals from the device.
5	RS-232 Rx: receives RS-232 (Rx) or RS-485 B (+) signals to change the settings of the device.
6	Meters: ground to tare. Controllers: analogue in, 0–5 V DC input defining the set point. Optional: 1–5 V DC, 0–10 V DC or 4–20 mA.

Connecting to an IO-Link master

The IO-Link master is the bridge between the IO-Link devices and the rest of the process automation network. You need an IO-Link master of the right class and the device description file (IODD) of the device. To connect a device:

1. Provide a class A or class B IO-Link master (Class A and class B devices and masters (p. 5)).

2. Connect the device or devices to the IO-Link master, and the master to the network, as shown in Figure 4.

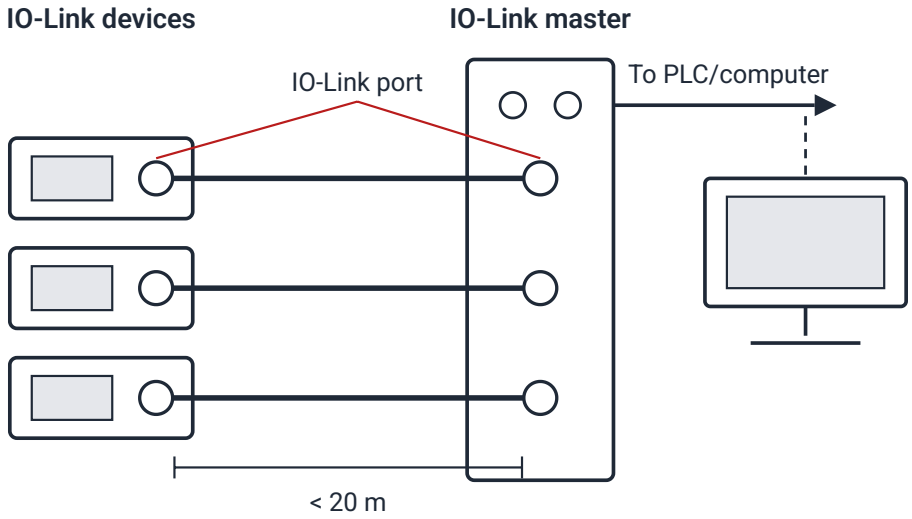


Figure 4. IO-Link devices connected to the IO-Link ports of a master; cable length less than 20 m.

3. Load the device description file into the master. Some master software finds and loads the right file by itself when it detects the device, if an Internet connection is available. Otherwise, use the device description file supplied with the instrument. The file can be read with the IODD Viewer.
4. Check that the master communicates with the device:
   - Some IO-Link masters have an indicator light that shows a valid device. The documentation of the master explains its colours and modes.
   - Open the master application. When the device and the master communicate, information from the device appears and updates in the master application.

The device is connected when its information updates in the master application. If no connection can be established, contact Analyt-MTC (About this guide (p. 2)).

IO-Link menu on the display

A device with a display has an extra IO-Link menu that the standard device menus do not include. Choose **`MENU`** › **`SETUP`** › **`IO-Link`**. The menu shows the process data format, *Floating Point* or *Integer* (Process data (p. 10)), and leads to the screens below.

- **IO-Link Status** shows whether the device is class A or class B, the state of the IO-Link connection, and whether process data (cyclic data) is being exchanged with the master.
- **Process Data** shows how many times the device has updated the process data readings. Press **`CLEAR`** to set the count to 0.
- **Received Commands** lists the ten most recent commands written to the device, the most recent at the top (Commands (p. 17)). **`HOLD`** stops the list from updating; it is then replaced by **`UPDATE`**, which makes the list refresh again with new commands. **`DETAIL`** opens the selected command and shows the command ID written to the device, the argument, the numerical status of the command and the value it returned, if any.

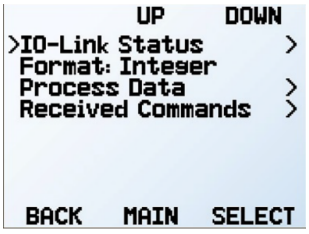


Figure 5. IO-Link menu.

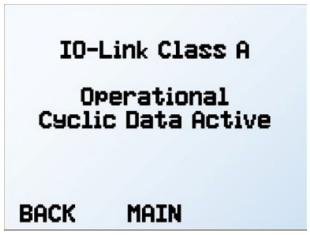


Figure 6. `IO-Link Status` screen.

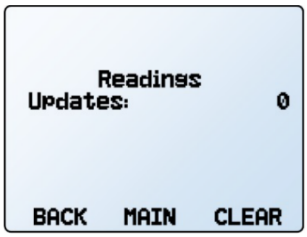


Figure 7. `Process Data` screen.

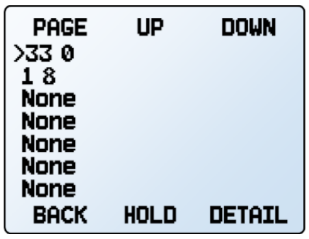


Figure 8. `Received Commands` screen.

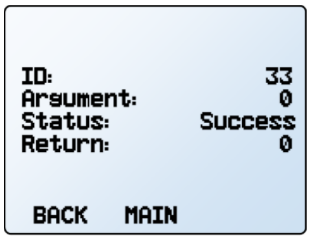


Figure 9. Details of a received command.

CHAPTER 2

Protocol conventions

Process data and variables

A 358 Series device exchanges two kinds of data with an IO-Link system: cyclic **process data**, which carries the readings and the set point, and acyclic **variables**, which you read or write by index to obtain information, change settings and send commands. The device follows the IO-Link Common Profile; the standard variables of that profile are listed in Chapter 3 (p. 12).

Conventions used in this guide

A variable is identified by its **index**; a record variable holds several values, each identified by a **subindex** and placed at a **bit offset** in the record. The tables give the data type of each value: `UInteger8`, `UInteger16`, `UInteger32` (or `Unsigned32`), `Integer32`, `Float32` or `String6`. Access is `ro` (read only) or `rw` (read and write).

Process data

Float32 and Integer32 formats

The device sends its process data either as floating-point values (Float32) or as integer values (Integer32). Only one format is active at a time. Change it on the display (IO-Link menu on the display (p. 8)) or with the Process Data Format (p. 14) variable (index 66). The content of the process data is the same in both formats; only the presentation of the readings differs.

- **Float32:** an invalid value reads as `0xFFFFFFFF`, a floating-point NaN.
- **Integer32:** an invalid reading reads as −2147483648. To obtain the value in its engineering units, multiply the integer by 10^{-n}, where n is the number of decimal places given in subindex 9 of the variable of that reading (Process data information (p. 14)).

Process data in

The input process data are the values that the device sends to the master. Which reading each subindex carries depends on the device series:

2-1 Process data in

Sub-index	Bit offset	Variable index	M	MC	P, EP	PC, EPC	L	LC
1	0	67	Pressure	Pressure	Pressure	Pressure	Pressure	Pressure
2	32	68	Secondary pressure	2nd pressure	2nd pressure	2nd pressure	2nd pressure	2nd pressure
3	64	69	Temperature	Temperature	Bar. pressure	Bar. pressure	Bar. pressure	Bar. pressure
4	96	70	Volumetric flow	Vol. flow	Temperature	Temperature	Temperature	Temperature
5	128	71	Mass flow	Mass flow	N/A	Set point	Vol. flow	Vol. flow
6	160	72	Totaliser 1	Totaliser 1	N/A	Valve drive	Totaliser 1	Totaliser 1
7	192	73	Barometric pressure	Set point	N/A	N/A	Totaliser 2	Set point

Sub-index	Bit offset	Variable index	M	MC	P, EP	PC, EPC	L	LC
8	224	74	Humidity	Valve drive	N/A	N/A	N/A	Valve drive

Bar. pressure: barometric pressure. Vol. flow: volumetric flow. N/A: not available on this series.

Process data out

Output process data exist on controllers only. They carry the requested set point, four octets long, as an Integer32 or a Float32 value according to the active format.

Communication timeout behaviour

A master can disconnect from the device, on purpose or because communication is interrupted. A meter simply continues measuring. When a controller detects a disconnection or invalid output process data, it does one of two things, which you select on the display under **`MENU`** › **`CONTROL`** › **`Setpoint Setup`** › **`On Comm Timeout`**:

- **`Maintain Last Setpt`**: it keeps controlling to the last set point received;
- **`Set Zero Setpt`**: it forces the set point to zero until the connection is re-established.

You can also set this behaviour with the Communication Timeout Behavior (p. 15) variable (index 81).

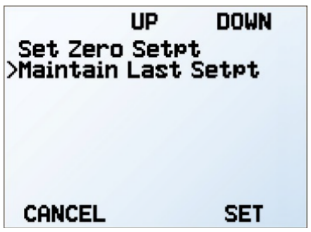


Figure 10. Options for the communication timeout.

Access and stored settings

The device implements the standard variables *Device Access Locks* (index 12) and *Data Storage Index* (index 3) of the IO-Link Common Profile (Standard variables (p. 12)); the IO-Link Interface and System Specification describes their use. To stop settings being changed on the display, use the Lock/unlock display (p. 34) command.

Some settings are kept across power cycles, as stated with each command. Restore factory settings (p. 35) returns all settings and the configuration to their values when the device was last at the factory and removes all calibrations made by third parties.

CHAPTER 3

Command and register reference

Every variable a 358 Series device offers over IO-Link and every command it accepts. Not every variable exists on every device; the devices are named with each variable and command.

Standard variables

As part of the IO-Link Common Profile, the device implements the standard variables below. The IO-Link Interface and System Specification and the IODD Common Profile specification describe them.

3-1 Standard variables

Index	Variable name
0	Direct Parameter Page 1
2	System Command
3	Data Storage Index
12	Device Access Locks
13	Profile Characteristic
14	PDInput Descriptor
15	PDOutput Descriptor
16	Vendor Name
18	Product Name
19	Product ID
21	Serial Number
22	Hardware Revision
23	Firmware Revision
24	Application Specific Tag
25	Function Tag
26	Location Tag
36	Device Status (Device status and events (p. 12))
37	Detailed Device Status (Device status and events (p. 12))

Device status and events

The *Detailed Device Status* standard variable reports the events that are pending. The status column gives the value of *Device Status* that goes with each event; the values are those of the status LED (Table 1-1).

3-2 Event codes

Event code	Type	Status	Name	Description
6145	Warning	2	Comms Error	A timeout was detected on a communication protocol.
6147	Error	4	ADC Error	There is an internal hardware error, and the device requires service.
6148	Warning	2	Mass Flow Out of Range	The current mass flow is larger than the overrange value.

Event code	Type	Status	Name	Description
6149	Warning	2	Volumetric Flow Out of Range	The current volumetric flow is larger than the overrange value.
6151	Warning	2	Abs Pressure Out of Range	The current absolute primary pressure is outside the device specifications.
6152	Warning	2	Gauge Pressure Out of Range	The current gauge primary pressure is outside the device specifications.
6153	Warning	2	Diff Pressure Out of Range	The current differential primary pressure is outside the device specifications.
6154	Warning	2	Abs Secondary Pressure Out of Range	The current absolute secondary pressure is outside the device specifications.
6155	Warning	2	Gauge Secondary Pressure Out of Range	The current gauge secondary pressure is outside the device specifications.
6156	Warning	2	Diff Secondary Pressure Out of Range	The current differential secondary pressure is outside the device specifications.
6157	Warning	2	Temperature Out of Range	The current temperature is outside the device specifications.
6159	Warning	2	Totaliser 1 Missed Flow	Flow was out of range while totaliser 1 was active; some volume may be missing.
6160	Warning	2	Totaliser 1 Overflow	Totaliser 1 has exceeded the maximum volume.
6161	Warning	2	Totaliser 2 Missed Flow	Flow was out of range while totaliser 2 was active; some volume may be missing.
6162	Warning	2	Totaliser 2 Overflow	Totaliser 2 has exceeded the maximum volume.
6166	Warning	3	Control Hold	Control is set to hold; closed-loop control is not active.
6167	Warning	3	Control Optimization Active	Control optimisation is in progress; set points you send are not followed.
6168	Warning	3	Over Pressure Limit	If configured, the pressure has exceeded the pressure limit.

Variables

The device offers the acyclic variables below for information and configuration. Not every variable is present on every device.

3-3 Variable overview

Index	Variable	Type	Access	Devices
66	Process Data Format	UInteger8	rw	All devices
67–74	Process Data Information	Record	ro	All devices
75	Gas Number	UInteger16	rw	Mass flow devices
76	Closed Loop Control Variable	UInteger16	rw	Controllers
77	P Gain	UInteger32	rw	Controllers
78	I Gain	UInteger32	rw	Controllers
79	D Gain	UInteger32	rw	Controllers
80	Closed Loop Control Algorithm	UInteger8	rw	Controllers
81	Communication Timeout Behavior	UInteger8	rw	Controllers
82	Flow Averaging Time Constant	UInteger16	rw	Flow devices
83	Primary Pressure Averaging Time Constant	UInteger16	rw	All devices
84	Secondary Pressure Averaging Time Constant	UInteger16	rw	All devices

Index	Variable	Type	Access	Devices
85	Totaliser 1 Flow Type	UInteger16	rw	Flow devices
86	Totaliser 1 Flow Accumulation Mode	UInteger8	rw	Flow devices
87	Totaliser 1 Limit Mode	UInteger8	rw	Flow devices
88	Totaliser 2 Flow Type	UInteger16	rw	Liquid flow meters
89	Totaliser 2 Flow Accumulation Mode	UInteger8	rw	Liquid flow meters
90	Totaliser 2 Limit Mode	UInteger8	rw	Liquid flow meters
91	Asynchronous Command Request	Record	rw	All devices
92	Asynchronous Command Status	Record	ro	All devices
93	Gas Mix Data	Record	rw	Mass flow devices

Process Data Format

Index 66 · All devices

Type **UInteger8** · Access **rw**

The format of the cyclic process data: 0 for floating point, 1 for integer (Float32 and Integer32 formats (p. 10)).

Process Data Information

Index 67–74 · All devices

Type **Record** · Access **ro**

Describes the values in the input process data (Process data in (p. 10)): one variable per value, as floating-point or integer values according to the process data format. Table 3-4 shows which value each index describes; Table 3-5 the data each of these variables holds.

3-4 Process data value described by each index

Index	M	MC	P, EP	PC, EPC	L	LC
67	Pressure	Pressure	Pressure	Pressure	Pressure	Pressure
68	Secondary pressure	2nd pressure	2nd pressure	2nd pressure	2nd pressure	2nd pressure
69	Temperature	Temperature	Barometric pressure	Bar. pressure	Bar. pressure	Bar. pressure
70	Volumetric flow	Volumetric flow	Temperature	Temperature	Temperature	Temperature
71	Mass flow	Mass flow		Set point	Volumetric flow	Volumetric flow
72	Totaliser 1	Totaliser 1		Valve drive	Totaliser 1	Totaliser 1
73	Barometric pressure	Set point			Totaliser 2	Set point
74	Humidity	Valve drive				Valve drive

3-5 Content of each process data information variable

Subindex	Bit offset	Type	Description
1	0	UInteger16	Value type: the reading reported in this process data subindex (Appendix A (p. 40)).
2	16	UInteger16	Value source (Appendix C (p. 45)).
3	32	Float32	Minimum of the value, without overrange.
4	64	Float32	Maximum of the value, without overrange.
5	96	Integer32	Minimum of the value, without overrange. Multiply by 10^{-n}, n being the value of subindex 9, for the value in engineering units.
6	128	Integer32	Maximum of the value, without overrange. Multiply by 10^{-n}, n being the value of subindex 9, for the value in engineering units.

Subindex	Bit offset	Type	Description
7	160	UInteger8	Engineering units of the value (Appendix B (p. 42)).
8	168	String6	Label of the engineering units, US-ASCII encoded.
9	216	UInteger8	Number of digits after the decimal point of the current value, the minimum and the maximum.

Gas Number

Index 75

Mass flow devices

`Type` **`UInteger16`** · `Access` **`rw`**

The number of the gas used to calculate the mass flow (Appendix D (p. 46)).

Closed Loop Control Variable

Index 76 Controllers

`Type` **`UInteger16`** · `Access` **`rw`**

The statistic number of the set point that drives closed-loop control. Valid values are the set point numbers of Appendix A (p. 40).

P Gain, I Gain, D Gain

Index 77, 78, 79

Controllers

`Type` **`UInteger32`** · `Access` **`rw`**

The proportional (index 77), integral (index 78) and derivative (index 79) gain of closed-loop control. Valid values are 0–65535.

Closed Loop Control Algorithm

Index 80 Controllers

`Type` **`UInteger8`** · `Access` **`rw`**

The algorithm used for closed-loop control.

Value	Description
1	PD/PDF
2	PD2I

Communication Timeout Behavior

Index 81 Controllers

`Type` **`UInteger8`** · `Access` **`rw`**

How a controller responds when it detects a disconnection of the IO-Link master or invalid output process data (Communication timeout behaviour (p. 11)). The same setting is available on the display.

Value	Description
0	Force the set point to zero until the connection is re-established.
1	Continue controlling to the last set point received.

Averaging time constants

Index 82, 83, 84

`Type` **`UInteger16`** · `Access` **`rw`**

The time constant, in milliseconds, of the geometric running average applied to the flow (index 82, flow devices), to the primary pressure (index 83) and to the secondary pressure (index 84). Valid values are 0–9999.

Totaliser settings

Index 85–90

`Flow type:` **`UInteger16`** · `Accumulation mode, limit mode:` **`UInteger8`** · `Access` **`rw`**

Totaliser 1 (indices 85, 86 and 87, flow devices) and totaliser 2 (indices 88, 89 and 90, liquid flow meters) are set the same way:

- **Flow type** (index 85 or 88): 0 disables the totaliser; otherwise one of the flow statistic numbers of Appendix A (p. 40).
- **Flow accumulation mode** (index 86 or 89): how flow is accumulated into the totaliser.
- **Limit mode** (index 87 or 90): what happens when the totaliser reaches its limit.

Value	Flow accumulation mode (index 86, 89)
0	Accumulate only positive flow.
1	Accumulate only negative flow.
2	Add positive and subtract negative flow.
3	Accumulate positive flow, resetting after flow ceases.

Value	Limit mode (index 87, 90)
0	Hold the value at the maximum limit and do not generate an event.
1	Set the value to zero and do not generate an event.
2	Hold the value at the maximum limit and generate an event.
3	Set the value to zero and generate an event.

Asynchronous Command Request

Index 91 · All devices

`Type` **`Record`** · `Access` **`rw`**

Commands the device to perform an action such as a tare, a gas change or locking the display (Commands (p. 17)).

Subindex	Bit offset	Type	Description
1	0	Unsigned32	ID of the command
2	32	Integer32	Argument of the command

Asynchronous Command Status

Index 92 · All devices

`Type` **`Record`** · `Access` **`ro`**

Information on the command being executed or the last command completed (Commands (p. 17)).

Subindex	Bit offset	Type	Description
1	0	Unsigned32	ID of the current or last completed command
2	32	Integer32	Argument of the current or last completed command
3	64	Unsigned32	Status of the current or last completed command, as a number (Table 3-7)
4	96	Integer32	Value returned by the command

Gas Mix Data

Index 93

Mass flow devices

Type **Record** · Access **rw**

The constituents of a gas mix. With this variable the device can store a custom mix of up to 5 gases and report its mass flow accurately. Write the gas numbers (Appendix D (p. 46)), each followed by its percentage of the mix. A percentage count of 1 is 0.01 %: 5000 is 50 %. With fewer than 5 gases, write 0 to the remaining subindices. The percentages must add up to 100 %, otherwise creating the mix fails. Then run the Create/update gas mix (p. 32) command.

Subindex	Bit offset	Type	Description
1	0	UInteger16	Gas number of gas 1
2	16	UInteger16	Percentage of gas 1; 1 count = 0.01 %
3	32	UInteger16	Gas number of gas 2
4	48	UInteger16	Percentage of gas 2; 1 count = 0.01 %
5	64	UInteger16	Gas number of gas 3
6	80	UInteger16	Percentage of gas 3; 1 count = 0.01 %
7	96	UInteger16	Gas number of gas 4
8	112	UInteger16	Percentage of gas 4; 1 count = 0.01 %
9	128	UInteger16	Gas number of gas 5
10	144	UInteger16	Percentage of gas 5; 1 count = 0.01 %

Note

The gas mix tells the device how to calculate the mass flow; it does not make the device mix gases. Mix the gases in the wanted proportions before they reach the device.

Commands

To send a command, write its ID and its argument to the Asynchronous Command Request (p. 16) variable, index 91: the ID at bit offset 0, the argument at bit offset 32. For example, to tare the flow, write command ID 33 at offset 0 and the tare time in milliseconds at offset 32.

3-6 Command request

Byte offset	Type	Description
0	Unsigned32	ID of the command
4	Integer32	Argument of the command

After a command, the device sets a status and reports it as a number in the Asynchronous Command Status (p. 16) variable, index 92. Some commands also return a value, described with each command; a command that defines no return value returns 0.

3-7 Command status values

Value	Status	Description
0	SUCCESS	The last command completed successfully.
1	IN_PROGRESS	A command is being executed.
2	INVALID_ID	The ID of the last command is invalid.
3	INVALID_ARGUMENT	The argument of the last command is invalid.

Value	Status	Description
4	`UNSUPPORTED`	The device does not support the last command.
5	`INVALID_MIX_IDX`	The requested gas mix number is invalid.
6	`INVALID_MIX_GAS`	A gas used in the mix does not exist on the device.
7	`INVALID_MIX_PCT`	The gas fractions of the mix do not add up to 100 %.

Note
The device does not run the same command twice in a row. To repeat a command, send No operation (p. 18) between the two. Sending No operation before or after each command prevents a command from repeating unintentionally.

Command function

No operation

All devices

`Command ID 0`

Does nothing. It separates two identical commands (same ID and argument), and can be sent before every command or at any time.

`Argument` Ignored.

Data readings

The type, source, minimum and maximum, units and decimal places of each reading are in variables 67–74 (Process Data Information (p. 14)).

Set reading engineering units

`10v18.0` All devices

`Command ID 65300 + offset`

Changes the engineering units of a reading. Add the offset of the reading (table below) to 65300: for example, 65302 changes the units of the temperature of an M device, or of the barometric pressure of a P device. Readings are not all independent: changing the units of the pressure also changes those of the secondary pressure and the barometric pressure.

`Argument` The value of the units wanted, from Appendix B (p. 42).

Response: `SUCCESS` if the units are changed, with the value of the new units as return value; `UNSUPPORTED` if the offset in the command ID does not exist on the device; `INVALID_ARGUMENT` if the units are not valid for the reading.

Offset	M	MC	P, EP	PC, EPC	L	LC
0	Pressure	Pressure	Pressure	Pressure	Pressure	Pressure
1	Secondary pressure	Secondary pressure	Secondary pressure	Secondary pressure	Secondary pressure	Secondary pressure
2	Temperature	Temperature	Barometric pressure	Barometric pressure	Barometric pressure	Barometric pressure
3	Volumetric flow	Volumetric flow	Temperature	Temperature	Temperature	Temperature
4	Mass flow	Mass flow		Set point	Volumetric flow	Volumetric flow
5	Totaliser 1	Totaliser 1		Valve drive	Totaliser 1	Totaliser 1
6	Barometric pressure	Set point			Totaliser 2	Set point

Offset	M	MC	P, EP	PC, EPC	L	LC
7	Humidity	Valve drive				Valve drive

Control

Commands for controllers only: set point, valves, control loop and totaliser batches.

Set power-up set point

`10v18.0` Controllers

`Command ID` **12**

Stores the current set point; the controller uses it immediately the next time it is powered.

`Argument` Ignored.

Response: `SUCCESS`; the current set point is saved as the power-up set point.

Set point maximum ramp (saved)

`10v18.0` Controllers

`Command ID` **65546**

Sets the maximum ramp rate of the set point and keeps it across power cycles. Do not send this command more often than every few minutes.

`Argument` A negative value queries the current maximum ramp rate; 0 disables it. To set a rate, take the change in percent of full scale per millisecond and multiply it by 10 000 000. For example, 1 % of full scale per second is 0.001 % per millisecond, which gives 10 000. More values in the table below.

Value	Maximum ramp rate
1000000000	100 % of full scale every millisecond
10000000	1 % of full scale every millisecond
1000000	100 % of full scale every second
16667	100 % of full scale every minute
10000	1 % of full scale every second
278	100 % of full scale every hour
167	1 % of full scale every minute
28	10 % of full scale every hour

Response: `SUCCESS` when the command is completed, with the current maximum ramp rate as return value; multiply it by 0.0000001 for the rate in percent of full scale per millisecond.

Set point maximum ramp (temporary)

10v18.0 Controllers

`Command ID` **65547**

Sets the maximum ramp rate of the set point without saving it: the rate is lost when the device loses power.

`Argument` As for Set point maximum ramp (saved) (p. 19): a negative value queries, 0 disables, other values from the table there.

Response: `SUCCESS` when the command is completed, with the current maximum ramp rate as return value; multiply it by 0.0000001 for the rate in percent of full scale per millisecond.

Hold valve(s)

10v18.0 Controllers

`Command ID` **6**

Pauses the valve or valves of the controller and stops any further control of the process, or cancels a hold.

`Argument` A value from the table below.

Value	Description
0	Cancel the valve hold and resume normal closed-loop control.
1	Hold all valves closed.
2	Hold the valves at their current positions.
3	Exhaust: close the upstream valve and fully open the downstream valve. Dual-valve controllers only.

Response: `SUCCESS` if the valves are set to the requested mode; `UNSUPPORTED` if the device does not support the mode; `INVALID_ARGUMENT` if the argument is not valid.

Set active valve

10v18.0 MCT controllers

`Command ID` **15**

Only on MCT stream-switching controllers: selects the valve that actively controls the process.

`Argument` A value from the table below.

Value	Description
0	Upstream valve, or the only valve
1	Downstream valve, or auxiliary valve wired in the downstream location
2	First auxiliary valve
3	Second auxiliary valve
65535	Query the active valve

Response: `SUCCESS` if the active valve changes; the return value is the active control valve (0 to 3) for argument 65535, otherwise 0. `UNSUPPORTED` if the device is not an MCT controller; `INVALID_ARGUMENT` if the value is not valid.

Set proportional closed-loop control gain

10v18.0 Controllers

```
Command ID 8
```

Sets the proportional gain (P gain) to the value of the argument.

`Argument` 0–65535.

Response: `SUCCESS` when the command is completed and the gain is set.

Set derivative closed-loop control gain

10v18.0 Controllers

```
Command ID 9
```

Sets the derivative gain (D gain) to the value of the argument.

`Argument` 0–65535.

Response: `SUCCESS` when the command is completed and the gain is set.

Set integral closed-loop control gain

10v18.0 Controllers

```
Command ID 10
```

Sets the integral gain (I gain) to the value of the argument.

`Argument` 0–65535.

Response: `SUCCESS` when the command is completed and the gain is set.

Set loop control variable

10v18.0 Controllers

```
Command ID 11
```

Changes the statistic the controller controls: a mass flow controller can, for example, control pressure or volumetric flow instead.

`Argument` A value from the table below.

Value	Description
0	Mass flow
1	Volumetric flow
2	Differential pressure
3	Absolute pressure
4	Gauge pressure
…	Any set point value of Appendix A (p. 40)

Response: `SUCCESS` if the loop control variable changes; `INVALID_ARGUMENT` if the statistic is not available on the device (for example mass flow on a pressure controller).

Set loop control algorithm

10v18.0 Controllers

`Command ID` **13**

Selects the PDF or the PD2I algorithm for the controller. Only one algorithm is active at a time.

`Argument` A value from the table below.

Value	Description
1	PDF closed-loop control algorithm
2	PD2I closed-loop control algorithm

Response: `SUCCESS` if the algorithm is selected; `INVALID_ARGUMENT` if the value is not a valid algorithm.

Read closed-loop gain

10v18.0 Controllers

`Command ID` **14**

Returns the current value of a gain of the control algorithm.

`Argument` A value from the table below.

Value	Description
0	Proportional gain (P gain) for PDF or PD2I
1	Derivative gain (D gain) for PDF or PD2I
2	Integral gain (I gain) for PD2I

Response: `SUCCESS` if the argument is valid, with the gain value (0–65535) as return value; `INVALID_ARGUMENT` if the gain is not a valid choice for the algorithm.

Set inverse pressure control

10v18.0 Controllers

`Command ID` **16**

Selects normal pressure control or inverse pressure control (usually back pressure), and whether the mode is kept across power cycles.

`Argument` A value from the table below.

Value	Description
0	Normal pressure control, not kept across power cycles
1	Inverse pressure control (usually back pressure), not kept across power cycles
3	Normal pressure control, kept across power cycles
4	Inverse pressure control (usually back pressure), kept across power cycles

Response: `SUCCESS` if the pressure control mode changes; `INVALID_ARGUMENT` if the argument is not available on the device or has no pressure control mode.

Query totaliser batch (integer)

10v18.0 Flow controllers

`Command ID` **65543**

Returns the batch size of one of the two totalisers.

`Argument` 1 for totaliser 1, 2 for totaliser 2.

Response: `SUCCESS` if the totaliser is enabled, with the batch size as return value; multiply it by 10^{-n}, n being the number of decimal places of the totaliser, for the volume in its engineering units. The return value is 0 if batching is disabled. `INVALID_ARGUMENT` if the totaliser is not enabled.

Query totaliser batch (float)

10v18.0 Flow controllers

`Command ID` **65540**

Returns the batch size of one of the two totalisers.

`Argument` 1 for totaliser 1, 2 for totaliser 2.

Response: `SUCCESS` if the totaliser is enabled, with the batch size as a Float32 return value in the engineering units of the totaliser. The return value is 0 if batching is disabled. `INVALID_ARGUMENT` if the totaliser is not enabled.

Set totaliser batch 1 (integer)

10v18.0 Flow controllers

`Command ID` **65544**

Sets the batch size of totaliser 1.

`Argument` The batch size multiplied by 10^{n}, n being the number of decimal places of the totaliser; 0 disables batching.

Response: `SUCCESS` if totaliser 1 is enabled and the batch size is valid, with the batch size as return value; `UNSUPPORTED` if totaliser 1 is not enabled; `INVALID_ARGUMENT` if the batch is larger than totaliser 1 can hold.

Set totaliser batch 1 (float)

10v18.0 Flow controllers

`Command ID` **65541**

Sets the batch size of totaliser 1.

`Argument` The batch size as a Float32 value in the engineering units of totaliser 1; 0 disables batching.

Response: `SUCCESS` if totaliser 1 is enabled and the batch size is valid, with the batch volume as return value; `UNSUPPORTED` if totaliser 1 is not enabled; `INVALID_ARGUMENT` if the batch is larger than totaliser 1 can hold.

Set totaliser batch 2 (integer)

10v18.0 Flow controllers

`Command ID` **65545**

Sets the batch size of totaliser 2.

`Argument` The batch size multiplied by 10^n, n being the number of decimal places of the totaliser; 0 disables batching.

Response: `SUCCESS` if totaliser 2 is enabled and the batch size is valid, with the batch size as return value; `UNSUPPORTED` if totaliser 2 is not enabled; `INVALID_ARGUMENT` if the batch is larger than totaliser 2 can hold.

Set totaliser batch 2 (float)

10v18.0 Flow controllers

`Command ID` **65542**

Sets the batch size of totaliser 2.

`Argument` The batch size as a Float32 value in the engineering units of totaliser 2; 0 disables batching.

Response: `SUCCESS` if totaliser 2 is enabled and the batch size is valid, with the batch volume as return value; `UNSUPPORTED` if totaliser 2 is not enabled; `INVALID_ARGUMENT` if the batch is larger than totaliser 2 can hold.

Control gain gas compensation

10v18.0

Mass flow controllers

`Command ID` **62**

Sets or queries whether gas compensation is enabled for the closed-loop control gains.

`Argument` A mode from the table below.

Value	Description
0	Do not change the current setting; query only.
1	Disable gas compensation of the closed-loop gains.
2	Enable gas compensation of the closed-loop gains.
3	Enable gas compensation of the closed-loop gains until the user changes a closed-loop control gain.

Response: `SUCCESS` if the device supports gas compensation of the control gains, with the current mode as return value; `UNSUPPORTED` if the device is not a controller or does not support setting the active gas properties; `INVALID_ARGUMENT` if the mode is not a valid value.

Control response analysis

A set point response check changes the set point to a test value, collects data for a set time and calculates the response of the control loop.

Check set point response: test set point (set point counts)

10v18.0 Controllers

`Command ID` **63**

Sets or queries the value to which the set point changes during the check.

`Argument` The test set point in set point counts, 0–65534; 65535 queries it.

Response: `SUCCESS` if the test set point was set or queried, with the test value in set point counts as return value; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the test set point is outside 0–65535.

Check set point response: test set point (integer)

10v18.0 Controllers

`Command ID` **65565**

Sets or queries the value to which the set point changes during the check.

`Argument` The test set point in set point engineering units, multiplied by 10^{n}, n being the number of decimal places of the set point; −2147483648 queries it.

Response: `SUCCESS` if the test set point was set or queried, with the test value as return value; multiply it by 10^{-n} for the value in set point engineering units. `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the test set point is outside the valid range.

Check set point response: test set point (float)

10v18.0 Controllers

`Command ID` **65564**

Sets or queries the value to which the set point changes during the check.

`Argument` The test set point as a Float32 value in set point engineering units; −3.40282346639e+38 (`0xFF7FFFFF`) queries it.

Response: `SUCCESS` if the test set point was set or queried, with the test value as a Float32 return value in set point engineering units; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the test set point is outside the valid range.

Check set point response: collection time

10v18.0 Controllers

`Command ID` **64**

Sets or queries how long the check collects data after the set point change.

`Argument` The collection time in seconds, 1–65535; 0 queries it.

Response: `SUCCESS` if the collection time was set or queried, with the collection time in seconds as return value; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the collection time is outside 0–65535.

Check set point response: run status

10v18.0 Controllers

`Command ID` **66**

Queries the state of the set point response check, and can start or abort it.

`Argument` An action from the table below.

Value	Action
0	Query the state of the check only; no action.

Value	Action
1	Start a check. The device changes the set point (overriding an analogue set point) and collects data for the set time; the set point returns to normal when data collection is complete.
2	Abort the check in progress. If no check is running, nothing happens.

Response: `SUCCESS` if the action, if any, was executed, with the state of the running or last completed check as return value (table below); `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the action is not a listed value.

Return value	State
0	The last check completed successfully and the results are calculated.
1	No check has run since power-up.
2	The check could not start because control gain tuning or another check is active.
3	A check is running.
4	The user aborted data collection early; the returned values are all zero.
5	Data collection stopped early because of oscillation; the returned values are all zero.

Check set point response: results notes

`10v18.0` Controllers

`Command ID` **67**

Returns notes about unusual circumstances during the last completed check that may affect the accuracy of the results.

`Argument` Always 0.

Response: `SUCCESS`, with the notes as return value: the sum of the values below that apply. `UNSUPPORTED` if the device is not a controller.

Value	Note
0	No notes; the process data showed typical patterns.
+1	The collection time was too short for reliable results; the values may change with a longer collection time.
+2	The process variable oscillated significantly. Control was not stable and the results are not reliable.
+4	The process variable moved little compared with the noise in the system, so the results may not be reliable. This happens most often when the set point change is too small, or when the process variable oscillated in a way the oscillation detection did not catch.
+8	A closed valve opened during data collection. The results are correct, but performance will differ significantly from cases that only move a valve that is already open.
+16	A valve opened as far as it could during data collection. The results are correct, but performance is probably limited by the size of the valve, not by the configuration of the control loop.

Check set point response: overshoot (set point counts)

10v18.0 Controllers

`Command ID` **68**

Returns the overshoot measured in the last completed check.

`Argument` Always 0.

Response: `SUCCESS`, with the overshoot in set point counts as return value; `UNSUPPORTED` if the device is not a controller.

Check set point response: overshoot (integer)

10v18.0 Controllers

`Command ID` **65567**

Returns the overshoot measured in the last completed check.

`Argument` Always 0.

Response: `SUCCESS`, with the overshoot as return value; multiply it by 10^{-n}, n being the number of decimal places of the set point, for the value in set point engineering units. `UNSUPPORTED` if the device is not a controller.

Check set point response: overshoot (float)

10v18.0 Controllers

`Command ID` **65566**

Returns the overshoot measured in the last completed check.

`Argument` Always 0.

Response: `SUCCESS`, with the overshoot as a Float32 return value in set point engineering units; `UNSUPPORTED` if the device is not a controller.

Check set point response: timing results

10v18.0 Controllers

`Command ID` **69**

Returns a timing measured in the last completed check.

`Argument` The timing to query, from the table below.

Value	Argument: timing queried	Return value
0	Time constant	The closed-loop time constant (T63) measured in the check, in milliseconds; for use in a first order plus dead time (FOPDT) model of the device.
1	Dead time	The dead time or delay of the system measured in the check, in milliseconds; for use in a FOPDT model of the device.
2	Rise time	The time the closed-loop response took to go from 10 % to 90 % of the step, in milliseconds.
3	Bandwidth	The estimated bandwidth of the closed-loop response, calculated from the rise time; multiply by 0.01 for the bandwidth in hertz.

Response: `SUCCESS`, with the requested timing as return value; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the value is not in the table.

Control optimisation

Control optimisation searches for closed-loop control gains that give the response you set.

Control optimisation: maximum flow (set point counts)

10v18.0 · Flow controllers

> Command ID **70**

Sets or queries the highest flow that control optimisation lets through on purpose, in set point counts.

`Argument` The maximum flow in set point counts, 1–64000; 0 sets the limit to full-scale flow; 65535 queries it.

Response: `SUCCESS` if the maximum flow was set or queried, with the current maximum flow in set point counts as return value; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the maximum flow is outside 0–64000 and not 65535, or below 20 % of full-scale flow.

Control optimisation: maximum flow (integer)

10v18.0 · Flow controllers

> Command ID **65569**

Sets or queries the highest flow that control optimisation lets through on purpose, as an integer.

`Argument` The maximum flow in set point engineering units multiplied by 10^{n}, n being the number of decimal places of the set point; 0 sets the limit to full-scale flow; a value below 0 queries it.

Response: `SUCCESS` if the maximum flow was set or queried, with the current maximum flow as return value; multiply it by 10^{-n} for the value in set point engineering units. `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the maximum flow is below 20 % of full-scale flow and not 0.

Control optimisation: maximum flow (float)

10v18.0 · Flow controllers

> Command ID **65568**

Sets or queries the highest flow that control optimisation lets through on purpose, as a floating-point value.

`Argument` The maximum flow as a Float32 value in set point engineering units; 0 sets the limit to full-scale flow; a value below 0 queries it.

Response: `SUCCESS` if the maximum flow was set or queried, with the current maximum flow as a Float32 return value in set point engineering units; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the maximum flow is below 20 % of full-scale flow and not 0.

Control optimisation: speed mode

10v18.0 Flow controllers

> Command ID **71**

Sets or queries the speed mode of control optimisation.

`Argument` A speed mode from the table below.

Value	Speed mode
0	Query the speed mode.
1	Most versatile use, with the slowest response.
2	Versatility before speed of response.
3	Speed of response before versatility.
4	Fastest response (smallest time constant of the control loop response); a small overshoot is allowed.

Response: `SUCCESS` if the speed mode was set or queried, with the current speed mode as return value; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the speed mode is not in the table.

Control optimisation: goal time

10v18.0 Flow controllers

> Command ID **72**

Sets or queries the response time that control optimisation aims for. After optimisation, the control response has a time constant close to this value at the tested set points, if the system can support it. If the requested time constant is smaller than the system can reasonably achieve, optimisation finds the gains with the shortest time constant. If one of the gains reaches 1 during optimisation and cannot become smaller to lengthen the time constant, the resulting time constant may be faster than requested.

`Argument` The time constant of the control response wanted, in milliseconds; 0 uses the fastest possible time constant (as the fastest speed mode); 65535 queries it.

Response: `SUCCESS` if the goal time was set or queried, with the current goal time in milliseconds as return value (0 while a speed mode is active); `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the goal time is negative.

Control optimisation: control algorithm

10v18.0 Flow controllers

`Command ID` **73**

Sets or queries the control algorithm for which control optimisation determines the gains.

`Argument` A value from the table below.

Value	Algorithm
0	Query only; do not change the configured mode.
1	Control optimisation chooses the most suitable algorithm for the device configuration and the optimisation parameters. The algorithm chosen for a given situation and configuration may change between firmware versions.
2	Use the PDF algorithm. Speed modes are 2 to 3 times slower than with the other values.
3	Use the PD2I algorithm.

Response: `SUCCESS` if the algorithm was set or queried, with the current value from the table as return value; `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the value is not in the table.

Control optimisation: run state

10v18.0 Flow controllers

`Command ID` **74**

Queries the state of control optimisation, and can take an action.

`Argument` An action from the table below.

Action	Description
0	Query the state only; no action.
1	Abort any control optimisation in progress.
2	Start a new control optimisation with the current settings. An optimisation in progress is aborted first.
3	As action 2, but pause the optimisation at each possible point, closing the valve or valves.
4	Ask the optimisation to pause and close the valves at the next possible point. A pause is most often used to let a closed volume that filled during optimisation bleed off to more normal pressure. If no optimisation is paused, nothing happens.
5	Resume a paused optimisation.

Response: `SUCCESS` if the action, if any, was performed, with the current activity of control optimisation as return value (table below); `UNSUPPORTED` if the device is not a controller; `INVALID_ARGUMENT` if the action is not in the table.

Mode	Activity
0	Control optimisation is not active.
1	Control optimisation is driving the valves of the device.
2	Control optimisation is active but paused. The valves stay closed until this command is sent with the resume action.
3	An abort was requested, but control optimisation is still active.

Control optimisation: result

10v18.0 Flow controllers

`Command ID 75`

Returns the result of the last completed control optimisation.

`Argument` Always 0.

Response: `SUCCESS`, with the result as return value (table below); `UNSUPPORTED` if the device is not a controller.

Result	Description
0	A control optimisation is in progress.
1	The last control optimisation completed successfully.
2	The last control optimisation was aborted by the user or because of an error.
3	An unexpected state occurred during the last control optimisation.
4	The maximum flow allowed for optimisation is too small.
5	A valve requested for control optimisation is not configured for closed-loop control.
6	No closed-loop control was configured at the time of the last control optimisation.
7	There was not enough flow to complete the last control optimisation. Check the inlet and outlet pressures.
8	The last control optimisation found no gains with acceptable performance.

Administrative

Gas selection and gas mixes, taring, totaliser reset, humidity correction, display control and factory settings.

Set gas

10v18.0

Mass flow devices

`Command ID 1`

Selects the gas the device measures. Set the gas so that the device reads the mass flow of the process gas accurately.

`Argument` The number of the gas (Appendix D (p. 46)).

Response: `SUCCESS` if the gas has changed; `INVALID_ARGUMENT` if the gas number does not exist.

Create/update gas mix

10v18.0

Mass flow devices

> Command ID **2**

Writes a new gas mix to the gas list of the device. Write the constituents to the Gas Mix Data (p. 16) variable (index 93) first.

`Argument` The gas number of the mix: 0, or 236–255. 0 writes the mix to the first free number, starting from 255 downwards; if no number is free, the command fails. The number of an existing mix overwrites that mix.

Response: `SUCCESS` if the gas has changed, with the number of the mix created or updated as return value; `INVALID_MIX_IDX` if the number is outside 236–255; `INVALID_MIX_GAS` if a gas of the mix does not exist on the device; `INVALID_MIX_PCT` if the percentages do not add up to 100 %.

Delete gas mix

10v18.0

Mass flow devices

> Command ID **3**

Deletes a custom gas mix from the device.

`Argument` The gas number of the mix to delete.

Response: `SUCCESS` if the mix was deleted; `INVALID_MIX_IDX` if the mix does not exist.

Perform tare

10v18.0 All devices

> Command ID **4**

Takes the current reading as zero. Gauge pressure tares need the sensor open to the atmosphere; differential pressure tares need a common pressure to measure from; absolute pressure tares need a barometer. Flow tares need no flow through the process, best at process pressure or as close to it as possible.

`Argument` A value from the table below.

Value	Description
0	Tare gauge or differential pressure
1	Tare absolute pressure
2	Tare mass and/or volumetric flow

Response: `SUCCESS` if the tare is done; `UNSUPPORTED` if the tare does not apply (for example a mass flow tare on a pressure device).

Perform pressure sensor tare

10v18.0 All devices

`Command ID` **31**

Takes the current gauge or differential pressure reading as zero; an absolute pressure tare uses the zero reading of the barometer. Gauge pressure tares need the sensor open to the atmosphere; differential pressure tares need a common pressure to measure from; absolute pressure tares need a barometer.

`Argument` The tare time in milliseconds, 0–32767; 0 uses the default of 256 milliseconds.

Response: `SUCCESS` if the tare is done; `UNSUPPORTED` if the sensor cannot be tared, usually an absolute pressure tare without a barometer; `INVALID_ARGUMENT` if the argument is below 0 or above 32767.

Perform secondary pressure sensor tare

10v18.0 All devices

`Command ID` **32**

Takes the current reading of the secondary pressure sensor as zero; an absolute pressure tare uses the zero reading of the barometer. Not all devices have a secondary pressure sensor. The same conditions apply as for the Perform pressure sensor tare (p. 33).

`Argument` The tare time in milliseconds, 0–32767; 0 uses the default of 256 milliseconds.

Response: `SUCCESS` if the tare is done; `UNSUPPORTED` if the sensor cannot be tared, usually an absolute pressure tare without a barometer, or no secondary pressure sensor; `INVALID_ARGUMENT` if the argument is below 0 or above 32767.

Perform flow tare

10v18.0 Flow devices

`Command ID` **33**

Takes the current flow reading as zero, for volumetric and mass flow.

`Argument` The tare time in milliseconds, 0–32767; 0 uses the default of 256 milliseconds.

Response: `SUCCESS` if the tare is done; `UNSUPPORTED` if the tare is not possible, usually because the device does not read volumetric or mass flow; `INVALID_ARGUMENT` if the argument is below 0 or above 32767.

Reset totaliser

10v18.0 Flow devices

`Command ID` **5**

Clears the current count of totaliser 1.

`Argument` Not used.

Response: `SUCCESS` when the reset is complete.

Set relative humidity percentage

`10v18.0`

Devices with humidity sensor

`Command ID` **24**

Sets the relative humidity used for gas corrections.

`Argument` 0–10000; 1 count is 0.01 % humidity: 100 for 1 %, 10000 for 100 %.

Response: `SUCCESS` if the relative humidity is changed; `UNSUPPORTED` if the relative humidity cannot be set on the device; `INVALID_ARGUMENT` if the argument is outside 0–10000.

Set relative humidity reference temperature

`10v18.0`

Devices with humidity sensor

`Command ID` **25**

Sets the reference temperature, in degrees Celsius, of the relative humidity used for gas corrections. It should match the relative humidity percentage.

`Argument` −3000 to 10000, for −30 °C to 100 °C; 1 count is 0.01 °C.

Response: `SUCCESS` if the reference temperature is changed; `UNSUPPORTED` if the relative humidity cannot be set on the device; `INVALID_ARGUMENT` if the argument is outside −3000 to 10000.

Read configuration checksum

`10v18.0` All devices

`Command ID` **17**

Calculates and returns a checksum of the calibration and configuration of the device when the command starts. The calculation can take 300 milliseconds; a timeout of more than 500 milliseconds is recommended. Every setting kept across power cycles is included, and so are values that may or may not be kept. Because the calibration is included, two devices with identical configurations can have different checksums; recalibration can change the checksum, and a firmware change may or may not change it. Set any parameter that changes in normal operation, such as the set point, to a known value before you read the checksum.

`Argument` Always 0.

Response: `SUCCESS` when the checksum is complete and the argument is 0, with the checksum of the whole configuration (0–65535) as return value; `INVALID_ARGUMENT` if the argument is not 0.

Lock/unlock display

`10v18.0` All devices

`Command ID` **7**

Locks or unlocks the display on the front of the device. A locked device still responds to button presses, but settings cannot be changed on the display.

`Argument` 0 unlocks the display; any other value locks it.

Response: `SUCCESS` if the display is locked or unlocked as requested.

Flash display

`10v18.0` All devices

> Command ID **20**

Makes the backlight of the display flash, for a set time or indefinitely.

`Argument` 1–65534 flashes the backlight for that number of seconds; 0 stops the flashing; 65535 flashes it indefinitely.

Response: `SUCCESS` if the backlight flashes or stops as requested; `UNSUPPORTED` if no display is connected to the device.

Restore factory settings

`10v18.0` All devices

> Command ID **26**

NOTICE

A factory restore removes every calibration made by third parties and returns all settings and the configuration to their values when the device was last at the factory. Use this command only when you troubleshoot a problem together with Analyt-MTC support.

Returns all settings and the configuration of the device to their values when it was last at the factory. Power-cycle the device after the restore.

`Argument` 49374, to confirm the factory restore.

Response: `SUCCESS` if the factory restore is complete; it may be readable only briefly before the restore clears the status. `UNSUPPORTED` if an error occurs during the restore; `INVALID_ARGUMENT` if the argument is not 49374.

CHAPTER 4

Examples

Five common tasks, shown as the variable writes and reads your IO-Link master performs. Use them to check your own configuration.

How to read the examples

Each example lists the values written to or read from a variable, by index and subindex. All values are decimal. How the master writes a record variable depends on the master and its software; the content of the write is the same in every case.

Example 1: tare the flow reading

Command 33 takes the current flow reading as zero. Only do this with no flow through the device (see Perform tare (p. 32)). An argument of 0 uses the default tare time of 256 milliseconds.

Write: index 91, Asynchronous Command Request

Subindex	Value	Meaning
1	33	Command ID: perform flow tare
2	0	Argument: default tare time

Then read index 92, Asynchronous Command Status (p. 16). While the tare runs, subindex 3 holds 1 (IN_PROGRESS); when subindex 1 holds 33 and subindex 3 holds 0 (SUCCESS), the tare is done.

Example 2: repeat the same command

The device does not run the same command twice in a row. To tare the flow a second time, write No operation (p. 18) to index 91 between the two tares:

Three writes to index 91

Write	Subindex 1	Subindex 2	Meaning
1	33	0	Perform flow tare
2	0	0	No operation
3	33	0	Perform flow tare again

Example 3: show the temperature of a mass flow meter in °F

On an M device the temperature has offset 2 in the set reading engineering units (p. 18) table, so the command ID is 65300 + 2 = 65302. The value for degree Fahrenheit is 3 (Table B-7).

Write: index 91, Asynchronous Command Request

Subindex	Value	Meaning
1	65302	Command ID: units of the reading at offset 2
2	3	Argument: °F

On success, index 92 holds status 0 (SUCCESS) in subindex 3 and the new units, 3, in subindex 4.

Example 4: create a mix of 50 % nitrogen and 50 % argon

Write the constituents to index 93, Gas Mix Data (p. 16). Nitrogen is gas number 8 and argon gas number 1 (Appendix D (p. 46)); 5000 counts are 50 %. The unused subindices are 0.

Write: index 93, Gas Mix Data

Subindex	Value	Meaning
1	8	Gas 1: nitrogen
2	5000	Gas 1: 50 %
3	1	Gas 2: argon
4	5000	Gas 2: 50 %
5–10	0	Not used

Then write command 2, Create/update gas mix (p. 32), with argument 0 to index 91: the device stores the mix under the first free gas number from 255 downward. Subindex 4 of index 92 returns the number of the mix. Select the mix with Set gas (p. 31) (command 1) and that number as argument.

Example 5: limit the set point ramp to 1 % of full scale per second

1 % of full scale per second is 0.001 % per millisecond; multiplied by 10 000 000 this gives an argument of 10000. Command 65546 keeps the setting across power cycles; send it only occasionally (Set point maximum ramp (saved) (p. 19)).

Write: index 91, Asynchronous Command Request

Subindex	Value	Meaning
1	65546	Command ID: maximum ramp, saved
2	10000	Argument: 1 % of full scale per second

On success, index 92 holds status 0 (SUCCESS) in subindex 3 and the current maximum ramp rate, 10000, in subindex 4: the ramp is now limited. Multiply the value by 0.0000001 for the rate in percent of full scale per millisecond.

CHAPTER 5

Error handling

Common symptoms, their likely causes and what to do. If none of these fits, contact Analyt-MTC support with the information listed in About this guide (p. 2).

Symptoms and remedies

Symptom	Cause and remedy
CONNECTION AND STATUS	
The master does not recognise the device	Load the device description file supplied with the instrument into the master or its configuration software (Connecting to an IO-Link master (p. 7)). Check the cable between the M12 connector of the device and the IO-Link port of the master.
A controller with large valves does not work on the master	Class B devices need a class B master, which supplies the isolated valve power on pin 5 (Class A and class B devices and masters (p. 5)).
Status LED red	Device status 4, failure: the device has detected a sensor or other malfunction. Read the *Detailed Device Status* (Device status and events (p. 12)); event 6147, *ADC Error*, means the device requires service.
Status LED yellow	Device status 2, out of specification: at least one reading is outside its measuring range or environmental conditions. The pending event names the reading (events 6148–6157), or a totaliser that missed flow or overflowed (events 6159–6162).
Status LED orange	Device status 3, check function: control is on hold (event 6166), control optimisation is running (event 6167) or the pressure limit is exceeded (event 6168). Cancel a valve hold with Hold valve(s) (p. 20), argument 0.
A controller goes to zero when the master disconnects	The communication timeout behaviour is set to force the set point to zero. To keep the last set point instead, set index 81 to 1 (Communication timeout behaviour (p. 11)).
VALUES	
A reading is `0xFFFFFFFF` or −2147483648	The value is invalid: NaN in the Float32 format, −2147483648 in the Integer32 format (Float32 and Integer32 formats (p. 10)).
Integer readings are too large by a power of ten	Integer32 values are scaled. Multiply by 10^{-n}, n being subindex 9 of the variable of that reading (Process Data Information (p. 14)).
The master shows values in the wrong format	Only one format is active at a time. Check index 66, Process Data Format (p. 14): 0 for floating point, 1 for integer.
Changing the units of the pressure also changed other readings	Readings are not all independent: the units of the pressure also apply to the secondary pressure and the barometric pressure (Set reading engineering units (p. 18)).
COMMANDS	
A repeated command does nothing	The device does not run the same command twice in a row. Send command 0, No operation (p. 18), between two identical commands.
Command status `INVALID_ID` (2)	The command ID is not valid. Check subindex 1 of index 91 against the Commands (p. 17).
Command status `INVALID_ARGUMENT` (3)	The argument is not allowed for this command. Check subindex 2 of index 91 against the arguments of the command.
Command status `UNSUPPORTED` (4)	The device does not support the command or the requested feature: for example, exhaust works only on dual-valve controllers, a set active valve only on MCT controllers, and a tare only if it applies to the device (no mass flow tare on a pressure device).
Command status `INVALID_MIX_IDX`, `INVALID_MIX_GAS` or `INVALID_MIX_PCT` after a mix	The mix number must be 0 or 236–255 (`INVALID_MIX_IDX`, 5), every gas of the mix must exist on the device (`INVALID_MIX_GAS`, 6) and the percentages must add up to 100 %, that is 10000 counts (`INVALID_MIX_PCT`, 7). See Gas Mix Data (p. 16).
Creating a mix fails although the data is correct	All gas numbers from 236 to 255 are in use. Delete a mix you no longer need with Delete gas mix (p. 32), or give the number of a mix you want to overwrite.

Symptom	Cause and remedy
Set points sent by the master are not followed	Control optimisation is in progress (event 6167). Wait for it to finish, or abort it with Control optimisation: run state (p. 30), action 1.

APPENDIX A

Statistics

Statistics are the readings and measurements a device provides. Their values are returned in subindex 1 of the process data information (p. 14) variables and are used by the closed loop control variable (p. 15), the totaliser flow types (Totaliser settings (p. 16)) and the set loop control variable (p. 21) command. Some statistics exist only on certain devices and configurations; ask Analyt-MTC about a statistic and how it works with your device.

A-1 Flow statistics

Flow statistic	Value	Notes
Mass flow	**5**	Current mass flow
Mass flow set point	**37**	Set point for mass flow
Total mass	**9**	Totalised mass
Total volume	**8**	Totalised volume, referenced to flow conditions
Volumetric flow	**4**	Volumetric flow, referenced to flow conditions
Volumetric flow set point	**36**	Set point for volumetric flow, referenced to flow conditions

A-2 Pressure statistics

Pressure statistic	Value	Notes
Pressure, absolute	**2**	Current absolute pressure
Pressure, absolute set point	**34**	Set point for absolute pressure
Pressure, barometric	**15**	Barometer reading
Pressure, differential	**7**	Current differential pressure
Pressure, differential set point	**39**	Set point for differential pressure
Pressure, gauge	**6**	Current gauge pressure
Pressure, gauge set point	**38**	Set point for gauge pressure
Pressure, second absolute	**344**	Absolute pressure of the second sensor, on devices with a second pressure sensor
Pressure, second absolute set point	**345**	Set point for the second absolute pressure
Pressure, second differential	**360**	Differential pressure of the second sensor, on devices with a second pressure sensor
Pressure, second differential set point	**361**	Set point for the second differential pressure
Pressure, second gauge	**352**	Gauge pressure of the second sensor, on devices with a pressure sensor in a second location
Pressure, second gauge set point	**353**	Set point for the second gauge pressure

A-3 Other statistics

Other statistic	Value	Notes
None	**1**	No statistic; usually an empty slot
Relative humidity	**25**	The fraction of complete saturation the device is using; with water vapour, this is the relative humidity
Temperature, stream	**3**	Current stream temperature
Valve drive	**13**	Valve drive signal

Other statistic	Value	Notes
Valve drive set point	**45**	The set point drives the selected valve directly

APPENDIX B

Engineering units

Commands and variables that take or return a unit use the values below; subindex 7 of the process data information (p. 14) variables returns them too. Which table applies depends on the statistic: for example, use table B-3 for a totaliser that counts standard or normal volume. To change the units of a reading, use set reading engineering units (p. 18).

B-1 Standard and normal flow units

Value	Label	Description
0		Unit not specified; use default values.
1	`---`	Unknown unit; no conversions are performed to other units. If calibrated with different units, the value is displayed in those units.
2	`SµL/m`	Standard microlitre per minute
3	`SmL/s`	Standard millilitre per second
4	`SmL/m`	Standard millilitre per minute
5	`SmL/h`	Standard millilitre per hour
6	`SL/s`	Standard litre per second
7	`SLPM`	Standard litre per minute
8	`SL/h`	Standard litre per hour
11	`SCCS`	Standard cubic centimetre per second
12	`SCCM`	Standard cubic centimetre per minute
13	Scm^3/h	Standard cubic centimetre per hour
14	Sm^3/m	Standard cubic metre per minute
15	Sm^3/h	Standard cubic metre per hour
16	Sm^3/d	Standard cubic metre per day
17	Sin^3/m	Standard cubic inch per minute
18	`SCFM`	Standard cubic foot per minute
19	`SCFH`	Standard cubic foot per hour
21	`SCFD`	Standard cubic foot per day
20	`kSCFM`	1000 standard cubic feet per minute
32	`NµL/m`	Normal microlitre per minute
33	`NmL/s`	Normal millilitre per second
34	`NmL/m`	Normal millilitre per minute
35	`NmL/h`	Normal millilitre per hour
36	`NL/s`	Normal litre per second
37	`NLPM`	Normal litre per minute
38	`NL/h`	Normal litre per hour
41	`NCCS`	Normal cubic centimetre per second
42	`NCCM`	Normal cubic centimetre per minute
43	Ncm^3/h	Normal cubic centimetre per hour
44	Nm^3/m	Normal cubic metre per minute
45	Nm^3/h	Normal cubic metre per hour
46	Nm^3/d	Normal cubic metre per day
62	`Count`	Set point count, 0–64000
63	`%`	Percent of the full scale

B-2 True mass flow units

Value	Label	Description
64	mg/s	Milligram per second
65	mg/m	Milligram per minute
66	g/s	Gram per second
67	g/m	Gram per minute
68	g/h	Gram per hour
69	kg/m	Kilogram per minute
70	kg/h	Kilogram per hour
71	oz/s	Ounce per second
72	oz/m	Ounce per minute
73	lb/m	Pound per minute
74	lb/h	Pound per hour

B-3 Total standard and normal volume units

Value	Label	Description
0		Unit not specified; use default values.
1	---	Unknown unit; no conversions are performed to other units. If calibrated with different units, the value is displayed in those units.
2	SµL	Standard microlitre
3	SmL	Standard millilitre
4	SL	Standard litre
6	Scm^3	Standard cubic centimetre
7	Sm^3	Standard cubic metre
8	Sin^3	Standard cubic inch

Value	Label	Description
9	Sft^3	Standard cubic foot
10	$kSft^3$	1000 standard cubic feet
32	NµL	Normal microlitre
33	NmL	Normal millilitre
34	NL	Normal litre
36	Ncm^3	Normal cubic centimetre
37	Nm^3	Normal cubic metre

B-4 Volumetric flow units

Value	Label	Description
0		Unit not specified; use default values.
1	`---`	Unknown unit; no conversions are performed to other units. If calibrated with different units, the value is displayed in those units.
2	`µL/m`	Microlitre per minute
3	`mL/s`	Millilitre per second
4	`mL/m`	Millilitre per minute
5	`mL/h`	Millilitre per hour
6	`L/s`	Litre per second
7	`LPM`	Litre per minute
8	`L/h`	Litre per hour
9	`US GPM`	US gallon per minute
10	`US GPH`	US gallon per hour
11	`CCS`	Cubic centimetre per second
12	`CCM`	Cubic centimetre per minute
13	cm^3/h	Cubic centimetre per hour
14	m^3/m	Cubic metre per minute
15	m^3/h	Cubic metre per hour
16	m^3/d	Cubic metre per day
17	in^3/m	Cubic inch per minute
18	`CFM`	Cubic foot per minute
19	`CFH`	Cubic foot per hour
21	`CFD`	Cubic foot per day
62	`count`	Set point count, 0–64000
63	`%`	Percent of full scale

B-5 Total volume units

Value	Label	Description
0		Unit not specified; use default values.
1	---	Unknown unit; no conversions are performed to other units. If calibrated with different units, the value is displayed in those units.
2	µL	Microlitre
3	mL	Millilitre
4	L	Litre
5	US GAL	US gallon
6	cm^3	Cubic centimetre
7	m^3	Cubic metre
8	in^3	Cubic inch
9	ft^3	Cubic foot
61	µP	Micropoise, a measure of viscosity; no conversions are performed to or from other units

B-6 Pressure units

Value	Label	Description
0		Unit not specified; use default values.
1	---	Unknown unit; no conversions are performed to other units. If calibrated with different units, the value is displayed in those units.
2	Pa	Pascal
3	hPa	Hectopascal
4	kPa	Kilopascal
5	MPa	Megapascal
6	mbar	Millibar
7	bar	Bar
8	g/cm^2	Gram force per square centimetre
9	kg/cm^2	Kilogram-force per square centimetre
10	PSI	Pound-force per square inch
11	PSF	Pound-force per square foot
12	mTorr	Millitorr
13	torr	Torr
14	mmHg	Millimetre of mercury at 0 °C
15	inHg	Inches of mercury at 0 °C
16	mmH_2O	Millimetre of water at 4 °C (NIST conventional)
17	mmH_2O	Millimetre of water at 60 °F
18	cmH_2O	Centimetre of water at 4 °C (NIST conventional)
19	cmH_2O	Centimetre of water at 60 °F
20	inH_2O	Inch of water at 4 °C (NIST conventional)
21	inH_2O	Inch of water at 60 °F
22	atm	Atmosphere (absolute pressure only)

Value	Label	Description
61	V	Volt; no conversions are performed to or from other units (intended only for log-linear absolute pressure sensors)
62	count	Set point count, 0–64000
63	%	Percent of full scale

B-7 Temperature units

Value	Label	Description
0		Unit not specified; use default values.
1	---	Unknown unit; no conversions are performed to other units. If calibrated with different units, the value is displayed in those units.
2	°C	Degree Celsius
3	°F	Degree Fahrenheit
4	°K	Kelvin
5	°Ra	Degree Rankine

APPENDIX C

Data sources

Subindex 2 of the process data information (p. 14) variables returns one of these values.

C-1 Data sources

Data source	Value	Notes
Instant display	**1**	Current readings, smoothed for the display
Instant serial	**2**	Current readings, smoothed for serial values
Totaliser 2	**25**	Readings taken from the second totaliser

APPENDIX D

Gas numbers

Gas numbers are used by the set gas (p. 31) command, the Gas Mix Data (p. 16) variable and the gas mix commands, and are read in the Gas Number (p. 15) variable (index 75). Numbers 236 to 255 are reserved for your own gas mixes (create/update gas mix (p. 32)).

#	Short name	Long name
0	Air	Air (Clean Dry)
1	Ar	Argon
2	CH_4	Methane
3	CO	Carbon Monoxide
4	CO2	Carbon Dioxide
5	C_2H_6	Ethane
6	H_2	Hydrogen
7	He	Helium
8	N_2	Nitrogen
9	N_2O	Nitrous Oxide
10	Ne	Neon
11	O_2	Oxygen
12	C_3H_8	Propane
13	nC_4H_{10}	Normal Butane
14	C_2H_2	Acetylene
15	C_2H_4	Ethylene (Ethene)
16	iC_4H_{10}	Isobutane
17	Kr	Krypton
18	Xe	Xenon
19	SF_6	Sulfur Hexafluoride[1]
20	C-25	25 % CO_2, 75 % Ar
21	C-10	10 % CO_2, 90 % Ar
22	C-8	8 % CO_2, 92 % Ar
23	C-2	2 % CO_2, 98 % Ar
24	C-75	75 % CO_2, 25 % Ar
25	He-25	25 % He, 75 % Ar
26	He-75	75 % He, 25 % Ar
27	A1025	90 % He, 7.5 % Ar, 2.5 % CO_2
28	Star29	Stargon CS (90 % Ar, 8 % CO_2, 2 % O_2)
29	P-5	5 % CH_4, 95 % Ar
30	NO	Nitric Oxide[2]
31	NF_3	Nitrogen Trifluoride[2]
32	NH_3	Ammonia[2]
33	Cl_2	Chlorine[2]
34	H_2S	Hydrogen Sulfide[2]
35	SO_2	Sulfur Dioxide[2]
36	C_3H_6	Propylene[2]
80	1Buten	1-Butylene[2]
81	cButen	Cis-Butene (cis-2-Butene)[2]
82	iButen	Isobutene[2]
83	tButen	Trans-2-Butene[2]
84	COS	Carbonyl Sulfide[2]
85	DME	Dimethylether (C_2H_6O)[2]
86	SiH_4	Silane[2]
100	R-11	Trichlorofluoro-methane (CCl_3F)[2,3]
101	R-115	Chloropentafluoro-ethane (C_2ClF_5)[2,3]
102	R-116	Hexafluoroethane (C_2F_6)[2]
103	R-124	Chlorotetrafluoro-ethane (C_2HClF_4)[2,3]
104	R-125	Pentafluoroethane (CF_3CHF_2)[2,3]
105	R-134A	Tetrafluoroethane (CH_2FCF_3)[2,3]
106	R-14	Tetrafluoromethane (CF_4)[2]
107	R-142b	Chlorodifluoroethane (CH_3CClF_2)[2,3]
108	R-143a	Trifluoroethane ($C_2H_3F_3$)[2,3]
109	R-152a	Difluoroethane ($C_2H_4F_2$)[2]
110	R-22	Difluoromonochloro-methane ($CHClF_2$)[2,3]
111	R-23	Trifluoromethane (CHF_3)[2,3]
112	R-32	Difluoromethane (CH_2F_2)[2,3]
113	R-318	Octafluorocyclobutane (C_4F_8)[2]
114	R-404A	44 % R-125, 4 % R-134A, 52 % R-143A[2,3]
115	R-407C	23 % R-32, 25 % R-125, 52 % R-134A[2,3]
116	R-410A	50 % R-32, 50 % R-125[2,3]
117	R-507A	50 % R-125, 50 % R-143A[2,3]
140	C-15	15 % CO_2, 85 % Ar
141	C-20	20 % CO_2, 80 % Ar
142	C-50	50 % CO_2, 50 % Ar
143	He-50	50 % He, 50 % Ar
144	He-90	90 % He, 10 % Ar
145	Bio5M	5 % CH_4, 95 % CO_2
146	Bio10M	10 % CH_4, 90 % CO_2
147	Bio15M	15 % CH_4, 85 % CO_2
148	Bio20M	20 % CH_4, 80 % CO_2
149	Bio25M	25 % CH_4, 75 % CO_2
150	Bio30M	30 % CH_4, 70 % CO_2
151	Bio35M	35 % CH_4, 65 % CO_2
152	Bio40M	40 % CH_4, 60 % CO_2
153	Bio45M	45 % CH_4, 55 % CO_2
154	Bio50M	50 % CH_4, 50 % CO_2
155	Bio55M	55 % CH_4, 45 % CO_2
156	Bio60M	60 % CH_4, 40 % CO_2
157	Bio65M	65 % CH_4, 35 % CO_2
158	Bio70M	70 % CH_4, 30 % CO_2
159	Bio75M	75 % CH_4, 25 % CO_2
160	Bio80M	80 % CH_4, 20 % CO_2
161	Bio85M	85 % CH_4, 15 % CO_2
162	Bio90M	90 % CH_4, 10 % CO_2
163	Bio95M	95 % CH_4, 5 % CO_2
164	EAN-32	32 % O_2, 68 % N_2
165	EAN-36	36 % O_2, 64 % N_2
166	EAN-40	40 % O_2, 60 % N_2
167	HeOx20	20 % O_2, 80 % He
168	HeOx21	21 % O_2, 79 % He
169	HeOx30	30 % O_2, 70 % He
170	HeOx40	40 % O_2, 60 % He
171	HeOx50	50 % O_2, 50 % He
172	HeOx60	60 % O_2, 40 % He
173	HeOx80	80 % O_2, 20 % He

#	Short name	Long name
174	HeOx99	99 % O_2, 1 % He
175	EA-40	Enriched air, 40 % O_2
176	EA-60	Enriched air, 60 % O_2
177	EA-80	Enriched air, 80 % O_2
178	Metab	Metabolic Exhalant (16 % O_2, 78.04 % N_2, 5 % CO_2, 0.96 % Ar)
179	LG-4.5	4.5 % CO_2, 13.5 % N_2, 82 % He
180	LG-6	6 % CO_2, 14 % N_2, 80 % He
181	LG-7	7 % CO_2, 14 % N_2, 79 % He
182	LG-9	9 % CO_2, 15 % N_2, 76 % He
183	HeNe-9	9 % Ne, 91 % He
184	LG-9.4	9.4 % CO_2, 19.25 % N_2, 71.35 % He
185	SynG-1	40 % H_2, 29 % CO, 20 % CO_2, 11 % CH_4
186	SynG-2	64 % H_2, 28 % CO, 1 % CO_2, 7 % CH_4
187	SynG-3	70 % H_2, 4 % CO, 25 % CO_2, 1 % CH_4
188	SynG-4	83 % H_2, 14 % CO, 3 % CH_4
189	NatG-1	93 % CH_4, 3 % C_2H_6, 1 % C_3H_8, 2 % N_2, 1 % CO_2
190	NatG-2	95 % CH_4, 3 % C_2H_6, 1 % N_2, 1 % CO_2
191	NatG-3	95.2 % CH_4, 2.5 % C_2H_6, 0.2 % C_3H_8, 0.1 % C_4H_{10}, 1.3 % N_2, 0.7 % CO_2
192	CoalG	50 % H_2, 35 % CH_4, 10 % CO, 5 % C_2H_4
193	Endo	75 % H_2, 25 % N_2
194	HHO	66.67 % H_2, 33.33 % O_2
195	HD-5	LPG: 96.1 % C_3H_8, 1.5 % C_2H_6, 0.4 % C_3H_6, 1.9 % n-C_4H_{10}
196	HD-10	LPG: 85 % C_3H_8, 10 % C_3H_6, 5 % n-C_4H_{10}
197	OCG-89	89 % O_2, 7 % N_2, 4 % Ar
198	OCG-93	93 % O_2, 3 % N_2, 4 % Ar
199	OCG-95	95 % O_2, 1 % N_2, 4 % Ar
200	FG-1	2.5 % O_2, 10.8 % CO_2, 85.7 % N_2, 1 % Ar
201	FG-2	2.9 % O_2, 14 % CO_2, 82.1 % N_2, 1 % Ar
202	FG-3	3.7 % O_2, 15 % CO_2, 80.3 % N_2, 1 % Ar
203	FG-4	7 % O_2, 12 % CO_2, 80 % N_2, 1 % Ar
204	FG-5	10 % O_2, 9.5 % CO_2, 79.5 % N_2, 1 % Ar
205	FG-6	13 % O_2, 7 % CO_2, 79 % N_2, 1 % Ar
206	P-10	10 % CH_4, 90 % Ar
210	D-2	Deuterium

[1] Sulfur hexafluoride is a highly potent greenhouse gas monitored under the Kyoto Protocol.

[2] Corrosion-resistant devices only.

[3] Under the Montreal Protocol and its Kigali Amendment, the production and consumption of these ozone-depleting substances is being, or has been, phased out. Make sure you comply with these regulations before using these gases; the same applies to R-113, R-123 and R-141b.

ANALYT-MTC Messtechnik GmbH

Sales, service and calibration

German Headquarters
Klosterrunsstraße 18, 79379 Müllheim
Germany
+49 7631 5545
info@analyt-mtc.de

French Subsidiary
175 Avenue d'Alsace, 68000 Colmar
France
+33 3 89 41 47 78
info@analyt-mtc.fr

www.analyt-mtc.com

AMTC-CG-0005 · Rev. 1 · 2026-10

© 2026 ANALYT-MTC Messtechnik GmbH. All rights reserved. Subject to technical changes without notice.