

EtherNet/IP™

Add-On Instructions
for Rockwell Logix 5000 controllers

EtherNet/IP™

Manual

English

IMPORTANT!

Read carefully before use.
Keep for future reference.



Bronkhorst®

Table of Contents

1 **Safety information..... 3**

1.1 General safety information 3

2 **General information 3**

2.1 Scope 3

2.2 About this document 3

2.3 Warranty 3

2.4 Ethernet/IP™ communication protocol 3

2.5 Declaration of Conformity 3

3 **Installation..... 4**

3.1 Bronkhorst® product information 4

3.2 EDS file 4

3.3 Selecting IO connection 4

3.4 Default Units 5

3.5 Acyclic messaging (get/set attribute single) 5

3.6 Add-On Instructions (AOI) 6

3.7 Using AOI 6

1 Safety information

1.1 General safety information

⚠ Please read this document and all other applicable documents carefully before installing and operating the product.

- All instructions must be observed. Failure to comply with instructions may result in material damage and personal injury.
- Before installation, take notice of the applicable product manual, which contains additional electrical requirements.

2 General information

2.1 Scope

This Add-On Instructions (AOI) provides general information and instructions on how Bronkhorst® products can be added to Rockwell Logix 5000 controllers.

2.2 About this document

Illustrations used

The illustrations in this document serve to provide general notices regarding correct operation. The illustrations are simplified representations of the actual situation.

Document validity

Documents are updated regularly. The most recent version is available online.

- ▶ Prior to work, check whether a newer version of this document is available on the Bronkhorst website.
- ▶ For download information, see the back cover of this manual.

Notices for the user

This document is intended for engineers and technicians responsible for integrating Bronkhorst products into an EtherNet/IP™ network. The reader is expected to be familiar with EtherNet/IP™ concepts and third-party software (such as Rockwell Studio 5000) used in their system.

- ▶ For tool-specific guidance, refer to the applicable Rockwell Automation documentation.

2.3 Warranty

- ▶ For information about the warranty, the general terms of delivery, please visit the Bronkhorst website: <https://bronkhorst.com/terms-and-conditions/>.

2.4 EtherNet/IP™ communication protocol

EtherNet/IP™ is an industrial network protocol that combines standard Ethernet with the Common Industrial Protocol (CIP) to enable real-time control and data exchange in automation systems. It uses TCP/IP and UDP/IP for reliable communication between devices, ensuring scalability and interoperability. For information about EtherNet/IP™:

- ▶ Visit the EtherNet/IP™ website at <https://www.odva.org/>

2.5 Declaration of Conformity

Bronkhorst® Mass Flow Meters/Controllers comply with the EtherNet/IP™ specifications and regulations as specified by ODVA Inc. The applicable declaration can be downloaded from the Bronkhorst website.

- ▶ For download information, see the back cover of this manual.

3 Installation

3.1 Bronkhorst® product information

All product descriptions, including LEDs and labels, are provided in the appropriate product documentation.

- ▶ See applicable product information in the online available manuals.
- ▶ For download information, see the back cover of this manual.

3.2 EDS file

The EDS file defines the available IO assemblies, data sizes, and device identity parameters of the Bronkhorst instrument. When installed in the engineering tool, these parameters are automatically recognized, reducing the need for manual configuration.

- ▶ Download the EDS file (zip file) from the Bronkhorst website.
- ▶ For download information, see the back cover of this manual.

The zip file contains two EDS files:

- Bronkhorst_FLEXI-FLOW_EtherNetIP.eds - for all FLEXI-FLOW Compact product types
- Bronkhorst_Meter_Controller_EtherNetIP.eds - for all other Bronkhorst instruments

3.2.1 FLEXI-FLOW Compact EtherNet/IP



When using the new Bronkhorst_FLEXI-FLOW_EtherNetIP.eds file with FLEXI-FLOW firmware version earlier than 1.10.0, electronic keying must be disabled.

The FLEXI-FLOW Compact series can be equipped with an integrated pressure sensor. Depending on the product type, two pressure channels (Pressure Inlet, Pressure Outlet) are available in addition to the flow channel. Product types without an integrated pressure sensor provide the flow channel only (» Tab. 1).

- ▶ Use the FLEXI-FLOW EDS file for all FLEXI-FLOW product types, regardless of the available channels.

Product type	Flow channel	Pressure channels
FF-Axxx	■	■
FF-Cx1, FF-Cx1D, FF-Cx1S, FF-Mx1, FF-Mx1D	■	■
FF-Cx0, FF-Cx0D, FF-Cx0S, FF-Mx0, FF-Mx0D, FF-Sxxx	■	--
FF-CxB, FF-MxB	■	--

Tab. 1. Available channels per FLEXI-FLOW Compact product type

3.3 Selecting IO connection

Four I/O connection types are available within the Ethernet/IP assemblies. The selected connection type determines which parameters are exchanged cyclically.

- ▶ Select the appropriate connection based on your data type requirements:
 - Basic_INT - Input: 3 SINT, Output: 2 SINT
 - Extended_INT - Input/Output: extended parameter set (integer)
 - Basic_REAL - Input/Output: basic parameter set (floating-point)
 - Extended_REAL - Input/Output: extended parameter set (floating-point)

Both the basic and extended assemblies are available in INT and REAL formats. The only difference between these options is the data type; the units remain identical. The default unit is counts (0–32000), where 32000 represents the full-scale value of the sensor.

3.4 Default Units

Units can be configured by setting the Data Unit (0x04) attribute of both the S-Analog Sensor and the S-Single Stage Controller classes. The value is stored in non-volatile memory and can be configured using any compatible EtherNet/IP configuration tool.

The different sensors of the FLEXI-FLOW must be configured separately by using different class instances (this applies only to the S-Analog Sensor and the S-Single Stage Controller classes):

Instance 0x01 = Flow

Instance 0x02 = Pressure Inlet

Instance 0x03 = Pressure Outlet

3.5 Acyclic messaging (get/set attribute single)

To read or write any value not available in the cyclic IO connection, CIP Generic messaging is used. For each value to be read or written, create a separate message of type CIP Generic using the Get Attribute Single or Set Attribute Single service.

The following Bronkhorst-specific parameters apply:

Parameter	Value	Description
Service	0x0E (Get) / 0x10 (Set)	Get or Set Attribute Single
Class	0x69	S-Analog Sensor / S-Single Stage Controller
Instance	0x01 / 0x02 / 0x03	Flow / Pressure Inlet / Pressure Outlet
Attribute	0x06	Counter value (example)
Attribute	0x04	Data Unit

Tab. 2. Bronkhorst-specific parameter



All values must be entered in hexadecimal.

For the full list of available classes, instances, and attributes, refer to the Bronkhorst EtherNet/IP™ manual.

- For download information, see the back cover of this manual.

3.6 Add-On Instructions (AOI)



The selected AOI must correspond exactly to the configured IO connection type (Basic or Extended, INT or REAL). A mismatch will result in incorrect data mapping and unpredictable instrument behaviour.

An AOI is provided for each assembly type.

- ▶ Import the appropriate AOI file into your Rockwell Logix 5000 project.
- ▶ Create input/output tags for each instrument.
- ▶ Select the AOI type that matches the IO connection configured for your instrument (Basic or Extended, INT or REAL).

The following AOI files are available:

- For Bronkhorst FLEXI-FLOW Compact
 - Bronkhorst_FF_Basic_INT_AOI.L5X
 - Bronkhorst_FF_Extended_INT_AOI.L5X
 - Bronkhorst_FF_Basic_REAL_AOI.L5X
 - Bronkhorst_FF_Extended_REAL_AOI.L5X
- For all other Bronkhorst instruments
 - Bronkhorst_Basic_INT_AOI.L5X
 - Bronkhorst_Extended_INT_AOI.L5X
 - Bronkhorst_Basic_REAL_AOI.L5X
 - Bronkhorst_Extended_REAL_AOI.L5X
- ▶ For each instrument, create a tag of the corresponding AOI data type for both input and output.

Tag data is accessible via the controller IO data (e.g. MFC03:I1.Data, MFC03:O1.Data).

IO data is exchanged as an array of SINT (signed short integer, 1 byte). Without the use of AOIs, this data is accessed as a raw byte array. The supplied AOI handles conversion to and from INT or REAL types internally, using the COP (copy) instruction, and maps the data to structured parameters.

3.7 Using AOI

The examples below show the Bronkhorst-specific tag assignments and data mappings. Controller-side implementation follows standard Rockwell Logix 5000 structured text conventions.

Example - Extended REAL (output to device):

```
MFC03_Output.Override      := 0;
MFC03_Output.Setpoint      := 16000.0;
MFC03_Output.Pressure_Inlet_Setpoint := 8000.0;
MFC03_Output.Pressure_Outlet_Setpoint := 24000.0;
```

```
Bronkhorst_FF_Extended_REAL_Output(MFC03_Output, MFC03:O1.Data);
```

Example - Extended REAL (input from device):

```
Bronkhorst_FF_Extended_REAL_Input(MFC03_Input, MFC03:I1.Data);
// Accessible values: MFC03_Input.Status, MFC03_Input.Measure,
//                   MFC03_Input.Setpoint, MFC03_Input.Pressure_Inlet,
//                   MFC03_Input.Pressure_Outlet
```

Example - Basic INT (manual data conversion):

```
// Controller input (device output)
Status := MFC01.I1.Data[0];
COP(MFC01.I1.Data[1], Measure, 1); // INT = 2 bytes

// Controller output (device input)
COP(Setpoint, MFC01.O1.Data[0], 2); // SINT = 1 byte, copy 2 bytes
```

Example - Basic REAL (manual data conversion):

```
// Controller input (device output)
Status := MFC01.I1.Data[0];
COP(MFC01.I1.Data[1], Flow, 1); // REAL = 4 bytes
COP(MFC01.I1.Data[5], Pressure_Inlet, 1);
COP(MFC01.I1.Data[9], Pressure_Outlet, 1);

// Controller output (device input)
COP(Flow, MFC01.O1.Data[0], 4);
COP(Pressure_Inlet, MFC01.O1.Data[4], 4);
COP(Pressure_Outlet, MFC01.O1.Data[8], 4);
```



9.17.234A • 2026-06



Service & Support



Contact



Product Info



Bronkhorst High-Tech B.V.
Nijverheidsstraat 1a
NL-7261 AK Ruurlo, The Netherlands

+31 573 45 88 00
info@bronkhorst.com
www.bronkhorst.com