

# Library Description



## **MBUS\_649\_01.lib** **WAGO M-Bus Library**

Version 1.0.3

© 2019 by WAGO Kontakttechnik GmbH & Co. KG  
All rights reserved.

### **WAGO Kontakttechnik GmbH & Co. KG**

Hansastraße 27  
D-32423 Minden

Phone: +49 (0) 571/8 87 – 0  
Fax: +49 (0) 571/8 87 – 1 69

Email: [info@wago.com](mailto:info@wago.com)

Online: <http://www.wago.com>

### **Technical Support**

Phone: +49 (0) 571/8 87 – 4 45 55  
Fax: +49 (0) 571/8 87 – 84 45 55

Email: [support@wago.com](mailto:support@wago.com)

Every conceivable measure has been taken to ensure the accuracy and completeness of this documentation. However, as errors can never be fully excluded, we always appreciate any information or suggestions for improving the documentation.

We wish to point out that the software and hardware terms, as well as the trademarks of companies used and/or mentioned in the present document are generally protected by trademark or patent.

# Information about This Documentation

## Copyright

This documentation, including all figures and illustrations contained therein, is subject to copyright protection. Any use of this documentation that infringes upon the copyright provisions stipulated herein is prohibited. Reproduction, translation, electronic and photo-technical filing/archiving (e.g., photocopying), as well as any amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden, Germany. Non-observance will entail the right of claims for damages.

## Number Notation

Table 1: Number Notation






Number System	Example	Comment
Decimal	100	Normal notation
Hexadecimal	0x64	C notation
Binary	'100' '0110.0100'	In quotation marks, nibble separated by a period

## Font Conventions

Table 2: Font Conventions

Font Type	Explanation
<i>italic</i>	Names of paths and files are displayed in italics, e.g.: <i>C:\Programs\WAGO-I/O-CHECK</i>
<b>Menu</b>	Menu options are displayed in bold, e.g.: <b>Save</b>
<b>&gt;</b>	A “greater than” symbol between two names denotes the selection of a menu option, e.g.: <b>File &gt; New</b>
<b>Input</b>	Designation of input or optional fields are displayed in bold, e.g.: <b>Start of measurement range</b>
“Value”	Input or selection values are displayed in quotation marks, e.g.: Enter the value “4 mA” under <b>Start of measurement range</b> .
<b>[Button]</b>	Button labels within the dialogs are bold and enclosed in square brackets, e.g.: <b>[Input]</b>
<b>[Keys]</b>	Key labels on the keyboard are displayed in bold and enclosed in square brackets, e.g.: <b>[F5]</b>

## Symbols

<b>DANGER</b>	<b>Warning against personal injury!</b>
	Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.
<b>DANGER</b>	<b>Do not work on components while energized!</b>
	Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.
<b>WARNING</b>	<b>Warning against personal injury!</b>
	Indicates a moderate-risk, potentially hazardous situation which, if not avoided, could result in death or serious injury.
<b>CAUTION</b>	<b>Warning against personal injury!</b>
	Indicates a low-risk, potentially hazardous situation which, if not avoided, may result in minor or moderate injury.
<b>NOTICE</b>	<b>Warning: Damage to property!</b>
	Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.
<b>ESD (Electrostatic Discharge)</b>	<b>Warning: Damage to property caused by electrostatic discharge (ESD)!</b>
	Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.
<b>Note</b>	<b>Important note!</b>
	Indicates a potential malfunction which will not result in damage to property, however, if not avoided.
<b>Information</b>	<b>Additional Information</b>
	Refers to additional information which is not an integral part of this documentation (e.g., the Internet).

## **Legal Principles Subject to Change**

WAGO Kontakttechnik GmbH & Co. KG reserves the right to make any alterations or modifications that serve to increase the efficiency of technical progress. WAGO Kontakttechnik GmbH & Co. KG owns all rights arising from granting patents or from the legal protection of utility patents. Third-party products are always mentioned without any reference to patent rights. Thus, the existence of such rights cannot be excluded.

## **Personnel Qualification**

The use of the product described in this document is exclusively geared to specialists having qualifications in PLC programming, electrical specialists or persons instructed by electrical specialists who are also familiar with the appropriate current standards.

Moreover, the persons cited here must also be familiar with all of the products cited in this document, along with the operating instructions. They must also be capable of correctly predicting any hazards which may not arise until the products are combined.

WAGO Kontakttechnik GmbH & Co. KG assumes no liability resulting from improper action and damage to WAGO products and third-party products due to non-observance of the information contained in this document.

## **Limitation of Liability**

This documentation describes the use of various hardware and software components in specific example applications. The components may represent products or parts of products from different manufacturers. The respective operating instructions from the manufacturers apply exclusively with regard to intended and safe use of the products. The manufacturers of the respective products are solely responsible for the contents of these instructions.

The sample applications described in this documentation represent concepts, that is, technically feasible applications. Whether these concepts can actually be implemented depends on various guidelines. For example, different versions of the hardware or software components can require different handling than that described here. Therefore, the descriptions contained in this documentation do not form the basis for assertion of a certain product characteristic.

Responsibility for safe use of a specific software or hardware configuration lies with the party that produces or operates the configuration. This also applies when one of the concepts described in this document was used for implementation of the configuration.

WAGO Kontakttechnik GmbH & Co. KG assumes no liability for the realization of these concepts.

## Table of Contents

<b>Table of Contents.....</b>	<b>6</b>
<b>1 Communication Module.....</b>	<b>7</b>
1.1 M-Bus Master via M-Bus Module (753-649) (FbMbusMaster_753_649)	7
<b>2 Meters.....</b>	<b>9</b>
2.1 Electricity meter (FbMbusElectricity).....	9
2.2 Heat meter (FbMbusHeat).....	11
2.3 Multi Telegram Meter (FbMbusMultiTel).....	13
2.4 Water meter (FbMbusWater).....	15
<b>3 User Data .....</b>	<b>17</b>
3.1 General Data Set (FbMbusGeneral).....	17
3.2 M-Bus Raw Data (FbMbusRawData).....	19
<b>4 Transmit Data .....</b>	<b>21</b>
4.1 Transmitting M-Bus Data (FbMbusSendData).....	21
<b>5 M-Bus Module Tunnel Protocol (753-649) .....</b>	<b>24</b>
5.1 Requesting Module Statistics (FbMbusStatistic).....	24
5.2 Requesting Module Status (FbMbusStatus).....	26
<b>6 Tools .....</b>	<b>28</b>
6.1 Measured Value Conversion in REAL (FbUnitConverter).....	28
6.2 Address conversion from decimal to hexadecimal (FuSecondaryAddress)	29
<b>7 Data Structure.....</b>	<b>30</b>
7.1 M-Bus Info (typMbusInfo).....	30
7.2 M-Bus Record (typMbusRecord).....	31
7.3 M-Bus Units (eMbusUnit).....	32
<b>8 Appendix.....</b>	<b>33</b>
8.1 Feedback (bFeedback).....	33



# 1 Communication Module

## 1.1 M-Bus Master via M-Bus Module (753-649) (FbMbusMaster\_753\_649)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusMaster_753_649		
Type:	Function	Function block	X Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
uiBaudrate	UINT	Baud rate of the M-Bus meter: 300, 2400, 9600 Default: 2400 (the common M-Bus baud rate)	
tTimeout	TIME	Timeout for M-Bus communication Default: t#3s	
xReset	BOOL	Rising edge: Master module is reset.	
Output parameter:	Data type:	Comment:	
bFeedback	BYTE	Response byte (see appendix)	
Graphical illustration:			
<div><div>FbMbusMaster_753_649</div><div><div>bModule_753_649</div><div>bFeedback</div><div>uiBaudrate</div><div>tTimeout</div><div>xReset</div></div></div>			

**Function description:**

The **FbMbusMaster753\_649** function block is used as the interface to the M-Bus Master module (753-649). All other function blocks communicate with the M-Bus Master module via this function block. This function block may be used only once per installed module.

The corresponding M-Bus module is specified at the “**bModule\_753\_649**” input. The value of this input cannot be changed at runtime.

The master module can be reinitialized by operating the “**xReset**” input. The output “**bFeedback**” outputs a numeric code with the response.

**Note:**

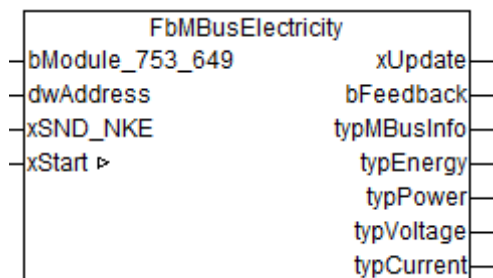
- The “FbMbusMaster753\_649” function block must be called in the program sequence before all other M-Bus function blocks.
- All M-Bus function blocks that communicate with this master must be called up in the same program task.



## 2 Meters

### 2.1 Electricity meter (FbMbusElectricity)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusElectricity		
Type:	Function	Function block	X   Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the meter readout; automatically reset after readout is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New telegram available	
bFeedback	BYTE	Response byte (see appendix)	
typMbusInfo	typMbusInfo	Fixed meter data (header in the data telegram with ID, manufacturer, medium, etc.)	
typEnergy	typMbus Record	Meter reading for energy consumed (value, power of 10, unit)	
typPower	typMbus Record	Current heat output (value, power of 10, unit)	
typVoltage	typMbus Record	Voltage (value, power of 10, unit)	
typCurrent	typMbus Record	Current (value, power of 10, unit)	

**Graphical illustration:****Function description:**

The function block is used to read and decode the data of an M-Bus electricity meter.

The “**bModule\_753\_649**” input defines the M-Bus module with which this function block must communicate.

The “**dwAddress**” input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

If “**xSND\_NKE**” is set, an M-Bus scaling command is sent to the meter before reading out the data. The command ensures in the case of meters with sequential telegrams, the first (most important) telegram is sent as a reply to the next query.

The “**xStart**” variable starts the meter readout. If the readout is successful, the function block resets the variable.

If the readout fails, the readout is started again. This operation repeats three times by default. The “**g\_MBUS\_MAX\_REPEAT**” global variable can be used to assign how many times the operation repeats.

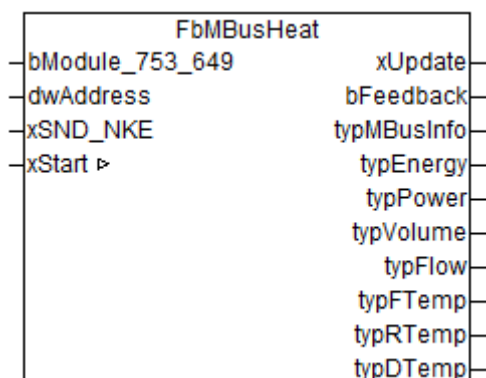
If a valid telegram is successfully queried, the “**xUpdate**” output is triggered. The output “**bFeedback**” outputs a numeric code with the response.

**Notes:**

To convert the meter readings to the required unit and a REAL type value, the user can use the function described further below. (See 6.1, page 28)

## 2.2 Heat meter (FbMbusHeat)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusHeat		
Type:	Function	Function block	X   Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the meter readout; automatically reset after readout is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New telegram available	
bFeedback	BYTE	Response byte (see appendix)	
typMbusInfo	typMbusInfo	Fixed meter data (header in the data telegram with ID, manufacturer, medium, etc.)	
typEnergy	typMbus Record	Meter reading for energy consumed (value, power of 10, unit)	
typPower	typMbus Record	Current heat output (value, power of 10, unit)	
typVolume	typMbus Record	Metered water volume (value, power of 10, unit)	
typFlow	typMbus Record	Current flow rate (value, power of 10, unit)	
typFTemp	typMbus Record	Supply temperature (value, power of 10, unit)	
typRTemp	typMbus Record	Return air temperature (value, power of 10, unit)	
typDTemp	typMbus Record	Differential temperature (value, power of 10, unit)	

**Graphical illustration:****Function description:**

The function block is used to read and decode the data of an M-Bus heat meter.

The “**bModule\_753\_649**” input defines the M-Bus module with which this function block must communicate.

The “**dwAddress**” input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

If “**xSND\_NKE**” is set, an M-Bus scaling command is sent to the meter before reading out the data. The command ensures in the case of meters with sequential telegrams, the first (most important) telegram is sent as a reply to the next query.

The “**xStart**” variable starts the meter readout. If the readout is successful, the function block resets the variable.

If the readout fails, the readout is started again. This operation repeats three times by default. The “**g\_MBUS\_MAX\_REPEAT**” global variable can be used to assign how many times the operation repeats.

If a valid telegram is successfully queried, the “**xUpdate**” output is triggered. The output “**bFeedback**” outputs a numeric code with the response.

**Notes:**

To convert the meter readings to the required unit and a REAL type value, the user can use the function described further below. (See 6.1, page 28)

## 2.3 Multi Telegram Meter (FbMBusMultiTel)

WAGO-I/O-PRO 32 Library Elements			
Category:		Building technology	
Name:		FbMBusMultiTel	
Type:	Function	Function block <b>X</b>	Program
Name of library:		MBUS_649_01.lib	
Libraries used:		WagoLibMBX_01.lib WagoLibKBUS.lib	
Applicable to:		See Release Note	
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the meter readout; automatically reset after readout is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New telegram available	
bFeedback	BYTE	Response byte (see appendix)	
typMbusInfo	typMbusInfo	Fixed meter data (header in the data telegram with ID, manufacturer, medium, etc.)	
aMbusRecord	ARRAY [1..MBUS_MAX _DATARECOR DS] OF typMbusRecor d	Array with meter values (value, power of 10, unit)	
bNumberOfRecords	BYTE	Number of meter values	
Graphical illustration:			
<div><div>FbMBusMultiTel</div><div><div>bModule_753_649</div><div>dwAddress</div><div>xSND_NKE</div><div>xStart ▶</div></div><div><div>xUpdate</div><div>bFeedback</div><div>typMbusInfo</div><div>aMbusRecord</div><div>bNumberOfRecords</div></div></div>			

**Function description:**

The function block is used to read and decode the data of an M-Bus multi telegram meter.

The **“bModule\_753\_649”** input defines the M-Bus module with which this function block must communicate.

The **“dwAddress”** input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

If **“xSND\_NKE”** is set, an M-Bus scaling command is sent to the meter before reading out the data. The command ensures in the case of meters with sequential telegrams, the first (most important) telegram is sent as a reply to the next query.

The **“xStart”** variable starts the meter readout. If the readout is successful, the function block resets the variable.

If the readout fails, the readout is started again. This operation repeats three times by default. The **“g\_MBUS\_MAX\_REPEAT”** global variable can be used to assign how many times the operation repeats.

If a valid telegram is successfully queried, the **“xUpdate”** output is triggered. The output **“bFeedback”** outputs a numeric code with the response.

**Notes:**

To convert the meter readings to the required unit and a REAL type value, the user can use the function described further below. (See 6.1, page 28)

## 2.4 Water meter (FbMBusWater)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMBusWater		
Type:	Function	Function block <b>X</b>	Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the meter readout. Automatically reset after readout is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New telegram available	
bFeedback	BYTE	Response byte (see appendix)	
typMbusInfo	typMbusInfo	Fixed meter data (header in the data telegram with ID, manufacturer, medium, etc.)	
typVolume	typMbus Record	Metered water volume (value, power of 10, unit)	
typFlow	typMbus Record	Current flow rate (value, power of 10, unit)	
Graphical illustration:			
<div><div>FbMBusWater</div><div><div>bModule_753_649</div><div>dwAddress</div><div>xSND_NKE</div><div>xStart</div></div><div><div>xUpdate</div><div>bFeedback</div><div>typMbusInfo</div><div>typVolume</div><div>typFlow</div></div></div>			



**Function description:**

The function block is used to read and decode the data of an M-Bus water meter.

The **“bModule\_753\_649”** input defines the M-Bus module with which this function block must communicate.

The **“dwAddress”** input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

If **“xSND\_NKE”** is set, an M-Bus scaling command is sent to the meter before reading out the data. The command ensures in the case of meters with sequential telegrams, the first (most important) telegram is sent as a reply to the next query.

The **“xStart”** variable starts the meter readout. If the readout is successful, the function block resets the variable.

If the readout fails, the readout is started again. This operation repeats three times by default. The **“g\_MBUS\_MAX\_REPEAT”** global variable can be used to assign how many times the operation repeats.

If a valid telegram is successfully queried, the **“xUpdate”** output is triggered. The output **“bFeedback”** outputs a numeric code with the response.

**Notes:**

To convert the meter readings to the required unit and a REAL type value, the user can use the function described further below. (See 6.1, page 28)

## 3 User Data

### 3.1 General Data Set (FbMbusGeneral)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusGeneral		
Type:	Function	Function block	X Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the meter readout; automatically reset after readout is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New telegram available	
bFeedback	BYTE	Response byte (see appendix)	
typMbusInfo	typMbusInfo	Fixed meter data (header in the data telegram with ID, manufacturer, medium, etc.)	
typMbusRecord	typMbusRecord	Meter reading (value, power of 10, unit)	
Graphical illustration:			
<div><div>FbMbusGeneral</div><div><div>bModule_753_649</div><div>dwAddress</div><div>xSND_NKE</div><div>xStart</div></div><div><div>xUpdate</div><div>bFeedback</div><div>typMbusInfo</div><div>typMbusRecord</div></div></div>			

**Function description:**

The function block is used to read and decode the data of an M-Bus meter and returns exactly one meter reading.

The **“bModule\_753\_649”** input defines the M-Bus module with which this function block must communicate.

The **“dwAddress”** input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

If **“xSND\_NKE”** is set, an M-Bus scaling command is sent to the meter before reading out the data. The command ensures in the case of meters with sequential telegrams, the first (most important) telegram is sent as a reply to the next query.

The **“xStart”** variable starts the meter readout. If the readout is successful, the function block resets the variable.

If the readout fails, the readout is started again. This operation repeats three times by default. The **“g\_MBUS\_MAX\_REPEAT”** global variable can be used to assign how many times the operation repeats.

If a valid telegram is successfully queried, the **“xUpdate”** output is triggered. The output **“bFeedback”** outputs a numeric code with the response.

**Notes:**

To convert the meter readings to the required unit and a REAL type value, the user can use the function described further below. (See 6.1, page 28)

## 3.2 M-Bus Raw Data (FbMbusRawData)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusRawData		
Type:	Function	Function block <b>X</b>	Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the meter readout; automatically reset after readout is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New telegram available	
bFeedback	BYTE	Response byte (see appendix)	
abData	ARRAY [0..MBUS_MAX_MBUSDATA] OF BYTE	Raw data	
uiLength	UINT	Length of the raw data	
Graphical illustration:			
<div><div>FbMbusRawData</div><div><div>bModule_753_649</div><div>dwAddress</div><div>xSND_NKE</div><div>xStart ▸</div><div>xUpdate</div><div>bFeedback</div><div>abData</div><div>uiLength</div></div></div>			

**Function description:**

The function block is used to read and decode the data of an M-Bus meter and returns the raw data in an **“abData”** byte array. The length of the raw data is output at the **“uiLength”** output.

The **“bModule\_753\_649”** input defines the M-Bus module with which this function block must communicate.

The **“dwAddress”** input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

If **“xSND\_NKE”** is set, an M-Bus scaling command is sent to the meter before reading out the data. The command ensures in the case of meters with sequential telegrams, the first (most important) telegram is sent as a reply to the next query.

The **“xStart”** variable starts the meter readout. If the readout is successful, the function block resets the variable.

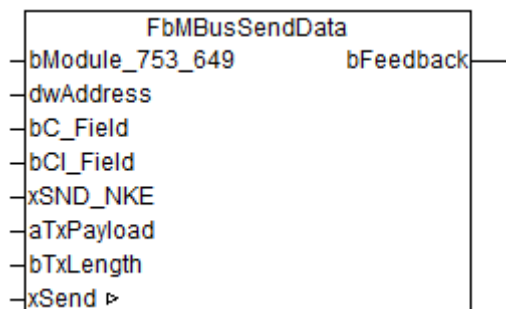
If the readout fails, the readout is started again. This operation repeats three times by default. The **“g\_MBUS\_MAX\_REPEAT”** global variable can be used to assign how many times the operation repeats.

If a valid telegram is successfully queried, the **“xUpdate”** output is triggered. The output **“bFeedback”** outputs a numeric code with the response.

## 4 Transmit Data

### 4.1 Transmitting M-Bus Data (FbMbusSendData)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMBusSendData		
Type:	Function	Function block	X    Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
dwAddress	DWORD	M-Bus primary address (>255 sec. addr.)	
bC_Field	BYTE	M-Bus C field Default: 16#53 (Transmit data)	
bCI_Field	BYTE	M-Bus CL field (16#53 for transmitting data)	
xSND_NKE	BOOL	M-Bus scaling command beforehand	
aTxPayload	ARRAY [0..MBUS_MAX_MBUSDATA] OF BYTE	M-Bus transmit data without frame (from CI field)	
bTxLength	BYTE	Length of the transmit data	
Input/output parameters:	Data type:	Comment:	
xSend	BOOL	Variable to start transmitting; automatically reset after transmission is complete.	
Output parameter:	Data type:	Comment:	
bFeedback	BYTE	Response byte (see appendix)	

**Graphical illustration:****Function description:**

The function block is used to transmit data to an M-Bus meter.

The “**bModule\_753\_649**” input defines the M-Bus module with which this function block must communicate.

If “**xSND\_NKE**” is set, an additional M-Bus scaling command is sent to the addressed meter before the actual data are sent.

The “**xSend**” variable starts the transmission of data to the M-Bus meter. This variable is automatically reset when transmission is complete.

If the transmission fails, the transmission is repeated. This operation repeats three times by default. The “**g\_MBUS\_MAX\_REPEAT**” global variable can be used to assign how many times the operation repeats.

**Telegram:**

A data set to be sent appears as follows:

Start	Field	Field	Start	Field	Field	Field	User	CS	Stop
68h	L	L	68h	C	O	CI	Data	LSB	16h

The function block automatically adds a start character (68h), L field, checksum (CS) and stop character (16h).

**C field:**

The “**bC\_Field**” field is the C field of the M-Bus telegram to be transmitted.

*The C field has the value 16#53, for example, for “send data to slave” (or 16#73 when FCB is set).*

The value of 16#52 (or 16#72 when FCB is set) is used to select a meter with the help of its 8-digit ID (secondary address) on the primary address 253.

**A field:**

The “**dwAddress**” input variable is preset with the primary address (<256) or the secondary address (ID) (>=256) of the meter.

The meter addressed by means of “dwAddress” acknowledges the received data record with the individual character 16#E5. If all M-Bus meters on the bus are to receive the data set, then the broadcast address 255 can be used without acknowledgement. In this case, the M-Bus Master Module automatically ensures that no telegrams are repeated and that there is no unnecessary waiting time due to timeouts.



**CI field:**

The **“bCI\_Field”** field is the CI field of the M-Bus telegram to be transmitted.

The value of 16#51 signals to the addressed meter that data is to be sent to it from the master. *If the value is 16#50, the addressed meter executes the “Application Reset” command.*

**User data:**

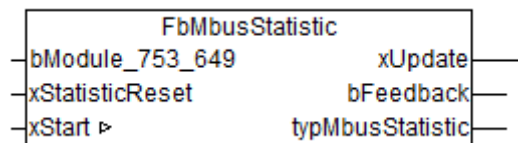
The M-Bus data to be transmitted is made available in the **“axTxPayload”** input array (User Data). The length of the data is specified at the **“bTxLength”** input.

The output **“bFeedback”** outputs a numeric code with the response.

## 5 M-Bus Module Tunnel Protocol (753-649)

### 5.1 Requesting Module Statistics (FbMbusStatistic)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusStatistic		
Type:	Function	Function block	X    Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
xStatisticReset	BOOL	TRUE: Statistics reset requested	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the module request; automatically reset after request is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New request successful	
bFeedback	BYTE	Response byte (see appendix)	
typMbusStatistic	typMbusStatistic	Requested values	
.wTxFrame	WORD	Number of M-Bus frames sent	
.wRxFrame	WORD	Number of fully received M-Bus frames	
.wRejectedTxFrame	WORD	Number of rejected M-Bus send requests because M-Bus not ready	
.dwSumRxFrame	DWORD	Number of total bytes received	
.dwSumFullRxFrame	DWORD	Number of bytes received that belong to the fully received M-Bus frames	
.wResponseTime	WORD	Number of M-Bus response timeouts Unsigned 16-bit integer value. Byte 1 is MS byte, byte 2 is LS byte.	
.wParityError	WORD	Number of M-Bus parity errors	
.wCollision	WORD	Number of M-Bus collisions	
.wOverload	WORD	Number of M-Bus overloads	
.wShortCircuit	WORD	Number of M-Bus short circuits	
.wOverheat	WORD	Number of internal terminal temperatures too high	
.wTimeoutFEC	WORD	Number of FEC status interface timeouts	

**Graphical illustration:****Function description:**

The function block is used to request statistics from the M-Bus module (753-649).

The "**bModule\_753\_649**" input defines the M-Bus module with which this function block must communicate.

The "**xStart**" variable starts the module request. If the request is successful, the function block resets the variable.

If the "**xStatisticReset**" variable is set to TRUE before the request, the module statistics are reset.

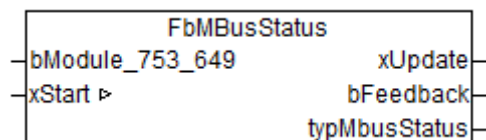
If the request fails, the request is started again. This operation repeats three times by default. The "**g\_MBUS\_MAX\_REPEAT**" global variable can be used to assign how many times the operation repeats.

If the request is successful, the "**xUpdate**" output is triggered. The output "**bFeedback**" outputs a numeric code with the response.

## 5.2 Requesting Module Status (FbMbusStatus)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbMbusStatus		
Type:	Function	Function block	X    Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
bModule_753_649	BYTE	Specifies which M-Bus Multi-Master module is to be addressed at the controller. Default: 1 Area: 1-MBUS_MAX_MODULES	
Input/output parameters:	Data type:	Comment:	
xStart	BOOL	Variable to start the module request; automatically reset after request is complete.	
Output parameter:	Data type:	Comment:	
xUpdate	BOOL	New request successful	
bFeedback	BYTE	Response byte (see appendix)	
typMbusStatus	typMbusStatus	Requested values	
.bBaudrate	BYTE	Baud rate of the module interface: 16#00: 300 baud 16#01: 2400 baud 16#02: 9600 baud	
.xShortCircuit	BOOL	M-Bus short circuit	
.xCollision	BOOL	M-Bus collision	
.xOverheat	BOOL	Overtemperature in the module	
.xFailedFEC	BOOL	Failure field side	
.xInvalidVoltage	BOOL	Field supply voltage invalid	
.xOverload	BOOL	Overload at the M-Bus	
.xMbusReady	BOOL	M-Bus ready	
.xMbusVoltageOn	BOOL	M-Bus voltage on	
.bCableQuality	BYTE	Cable quality: Measurement of the capacity of the connected M-Bus cabling; no specification in the SI unit farad! Value range = 0 ... 255 %	
.rTemperature	REAL	Temperature in the module (raw value) Signed 16-bit integer raw value [-32768 ... 32767] of the module temperature; byte 1 is MS byte, byte 2 is LS byte. To determine the temperature, the raw value must be divided by 100. Example: 9 = (16-bit raw value) / 100 □ [-327.68°c ... 327.67°c]	

.uiMarkVoltage	UINT	Mark operating voltage: Unsigned 16-bit integer measured value of the Mark operating voltage in Mv. Byte 1 is MS byte, byte 2 is LS byte. Example: 0x9168 □ 37224mV
.uiSpaceVoltage	UINT	Space operating voltage: Unsigned 16-bit integer measured value of the Space operating voltage in Mv. Byte 1 is MS byte, byte 2 is LS byte. Example: 0x5BF4 □ 23540mV
.uiPlusVoltageToGround	UINT	“M-Bus+” voltage based on Ground Unsigned 16-bit integer measured value of the “M-Bus+” voltage in Mv. Byte 1 is MS byte, byte 2 is LS byte. Example: 0x8E24 □ 36388mV
.uiMinusVoltageToGround	UINT	“M-Bus-” voltage based on Ground Unsigned 16-bit integer measured value of the “M-Bus-” voltage in Mv. Byte 1 is MS byte, byte 2 is LS byte. Example: 0x0094 □ 148mV
.bVersionFEC	BYTE	FEC version: Version number of the frontend controller firmware

**Graphical illustration:****Function description:**

The function block is used to request the status from the M-Bus module (753-649).

The “**bModule\_753\_649**” input defines the M-Bus module with which this function block must communicate.

The “**xStart**” variable starts the module request. If the request is successful, the function block resets the variable.

If the request fails, the request is started again. This operation repeats three times by default. The “**g\_MBUS\_MAX\_REPEAT**” global variable can be used to assign how many times the operation repeats.

If the request is successful, the “**xUpdate**” output is triggered. The output “**bFeedback**” outputs a numeric code with the response.

## 6 Tools

### 6.1 Measured Value Conversion in REAL (FbUnitConverter)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FbUnitConverter		
Type:	Function	Function block <b>X</b>	Program
Name of library:	MBUS_649_01.lib		
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib		
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
typMbus Record	typMbus Record	M-Bus data records (value, power of 10, unit)	
eUnit	eMbusUnit	Target unit	
Output parameter:	Data type:	Comment:	
rValue	REAL	Meter reading for target unit	
xError	BOOL	Error when converting to the target unit	
Graphical illustration:			
<div><div>FbUnitConverter</div><div><div>typMbus_Record ▶</div><div>eUnit ▶</div></div><div><div>rValue</div><div>xError</div></div></div>			
Function description:			
<p>The function block is used to convert an M-Bus “<b>typMBRecord</b>” data set to a required “<b>eUnit</b>” target unit and a meter value of the REAL type “<b>rValue</b>”. “<b>xError</b>” indicates whether the conversion into the target unit has been performed without error.</p>			
<u>Note:</u>			
<p>Please note that the converted meter values may have rounding errors due to the REAL format. It is not possible to convert input values outside the range of <math>\pm 3\,999\,999\,999</math></p>			

## 6.2 Address conversion from decimal to hexadecimal (FuSecondaryAddress)

WAGO-I/O-PRO 32 Library Elements			
Category:	Building technology		
Name:	FuSecondaryAddress		
Type:	Function X	Function block	Program
Name of library:	MBUS_649_01.lib		
Libraries used:			
Applicable to:	See Release Note		
Input parameter:	Data type:	Comment:	
dwAddress	DWORD	Adresse in decimal	
Output parameter:	Data type:	Comment:	
	DWORD	Adresse in hexadezimal	
Graphical illustration:			
<div><div><div>FuUnitConverter</div><div><div>typMbusRecord ▶</div><div>eUnit ▶</div></div></div><div></div></div>			
Function description:			
<p>The function <b>FuSecondaryAddress</b> is used to convert an M-Bus address. The decimal address on input <b>dwAddress</b> (e.g. from visualization with %s) is converted to the hexadecimal secondary address of the MBus device.</p> <p>Example:</p> <p><b>dwAddress</b> : 1234</p> <p><b>FuSecondaryAddress</b> : 16#1234</p> <p>If <b>dwAddress</b> is lower than 255, the address is interpreted as primary address and outputs unconverted.</p>			



## 7.1 M-Bus Info (typMBusInfo)

WAGO-I/O-PRO 32 Library Elements		
Category:	Building technology	
Name:	typMbusInfo	
Type:	Data type <input checked="" type="checkbox"/>	Enumeration <input type="checkbox"/>
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib	
Applicable to:	See Release Note	
Declaration:		
TYPE typMbusInfo: STRUCT bPAdr		

## 7.2 M-Bus Record (typMbusRecord)

WAGO-I/O-PRO 32 Library Elements		
Category:	Building technology	
Name:	typMbusRecord	
Type:	Data type <input checked="" type="checkbox"/>	Enumeration <input type="checkbox"/>
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib	
Applicable to:	See Release Note	
Declaration:		
TYPE typMbusRecord: STRUCT sValue                  : STRING[21]; sPof10                  : STRING[3]; sUnit                   : STRING[7]; bSubdevice              : BYTE; bTariff                 : BYTE; bStorage                : BYTE; bFunctionNo             : BYTE; sDescriptor             : STRING[23]; eEnum_Unit              : eMbusUnit; END_STRUCT END_TYPE		

## 7.3 M-Bus Units (eMBusUnit)

WAGO-I/O-PRO 32 Library Elements		
Category:	Building technology	
Name:	eMBusUnit	
Type:	Data type <input type="checkbox"/>	Enumeration <input checked="" type="checkbox"/>
Libraries used:	WagoLibMBX_01.lib WagoLibKBUS.lib	
Applicable to:	See Release Note	
Declaration:		
TYPE eMBusUnit : (MBus_miW, MBus_W, MBus_kW, MBus_MW, MBus_GW, MBus_miJps, MBus_Jps, MBus_kJps, MBus_MJps, MBus_GJps, MBus_miJpmin, MBus_Jpmin, MBus_kJpmin, MBus_MJpmin, MBus_GJpmin, MBus_miJph, MBus_Jph, MBus_kJph, MBus_MJph, MBus_GJph, MBus_miWs, MBus_Ws, MBus_kWs, MBus_MWs, MBus_GWs, MBus_miJ, MBus_J, MBus_kJ, MBus_MJ, MBus_GJ, MBus_miWmin, MBus_Wmin, MBus_kWmin, MBus_MWmin, MBus_GWmin, MBus_miWh, MBus_Wh, MBus_kWh, MBus_MWh, MBus_GWh, MBus_mil, MBus_l, MBus_m3, MBus_milps, MBus_lps, MBus_m3ps, MBus_milpmin, MBus_lpmin, MBus_m3pmin, MBus_milph, MBus_lph, MBus_m3ph, MBus_g, MBus_kg, MBus_gps, MBus_kgps, MBus_gpmin, MBus_kgpmin, MBus_gph, MBus_kgph, MBus_mibar, MBus_bar, MBus_kbar, MBus_miC, MBus_C, MBus_miK, MBus_K, MBus_HCA, Mbus_none, MBus_V, MBus_miA, MBus_A, MBus_pRH, MBus_Hz); END_TYPE		

## 8 Appendix

### 8.1 Feedback (bFeedback)

Table 1: Feedback

Feedback	ID Number (hex)	ID Number (dez)
MBUS_OK	16#00	0
MBUS_INFO_APP_BUSY	16#01	1
MBUS_INFO_APP_ACTIVE_SEND	16#02	2
MBUS_INFO_APP_WAITING	16#03	4
MBUS_ERROR_NULL	16#FF	255
MBUS_ERROR_KBUS	16#B0	176
MBUS_ERROR_MBX2ERROR	16#B1	177
MBUS_ERROR_MODULE_IDENTIFICATION	16#B2	178
MBUS_ERROR_APP_INVALIDPARAM	16#81	129
MBUS_ERROR_APP_TIMEOUT Extend the timeout time, Task cycle must be between 30-80ms.	16#82	130
MBUS_ERROR_APP_CRC	16#83	131
MBUS_ERROR_APP_FRAME	16#84	132
MBUS_ERROR_APP_UNANALYSABLE Wrong function block, use FbMbusMultiTel. Extend the timeout time, Task cycle must be between 30-80ms.	16#85	133
MBUS_ERROR_APP_INVALID_TELEGRAM Extend the timeout time, Task cycle must be between 30-80ms.	16#86	134
MBUS_ERROR_MSG_RSP_TIMEOUT Wrong address. Or secondary address not in Hex, use 16#xxxxxx.	16#E1	225
MBUS_ERROR_MSG_PARITY	16#E2	226
MBUS_ERROR_MSG_COLLISION	16#E3	227
MBUS_ERROR_MSG_REJECTED	16#E4	228
MBUS_ERROR_RSP_SHORTCIRCUIT Short circuit on MBus-fieldbus.	16#C1	193
MBUS_ERROR_RSP_COLLISION Telegram collision.	16#C2	194

---

MBUS_ERROR_RSP_OVERHEAT MBus-Module overheat.	16#C3	195
MBUS_ERROR_RSP_FEC MBus field side failed, 24V power supply not available.	16#C4	196
MBUS_ERROR_RSP_FVOLT Voltage not allowed	16#C5	197
MBUS_ERROR_RSP_OVERLOAD Telegram overload	16#C6	198
MBUS_ERROR_RSP_NOTREADY MBUS not ready	16#C7	199
MBUS_ERROR_RSP_BUSVOLTOFF MBus-Module: 24V power supply not available.	16#C8	200

MBUS_ERROR_RSP_BUSVOLTOFF Mbus voltage off	16#C8
---	-------

# WE! INNOVATE!

WAGO Kontakttechnik GmbH & Co. KG  
PO box 2880 • D-32385 Minden  
Hansastraße 27 • D-32423 Minden  
Phone: +49 (0) 571/8 87 – 0  
Fax: +49 (0) 571/8 87 – 1 69  
Email: [info@wago.com](mailto:info@wago.com)  
Web: <http://www.wago.com>

