



INDUSTRIAL CONTROL COMMUNICATIONS, INC.

M-Bus Master Driver Manual



TABLE OF CONTENTS

1 M-Bus Master	2
1.1 Overview	2
1.2 Connections	2
1.3 Master Settings	2
1.4 Service Object Settings	3
1.4.1 <i>Fixed Data Structure Settings</i>	4
1.4.2 <i>Variable Data Structure Settings</i>	4
1.4.2.1 Data Information Block (DIB) Configuration.....	4
1.4.2.2 Value Information Block (VIB) Configuration	5
1.4.2.3 Functions	5
1.5 Diagnostics Object	5

1 M-Bus Master

1.1 Overview

This driver supports the M-Bus (or Meter-Bus) master protocol. Some notes of interest are:

- All devices are addressed using primary addresses.
- Supports baud rates between 300...38400 baud.
- Supports all M-Bus modes, including mode 2 and the fixed data structure.
- Auto-detect feature for decoding M-Bus data.
- Supported commands include REQ_UD2 and SND_UD.
- Optional manual entry of the Data Information Block (DIB) and Value Information Block (VIB) provides support for custom requests and commands.

1.2 Connections

This section describes the typical connections used for a Millennium Series gateway.

Because M-Bus does not natively use an RS-485 physical layer, an RS-485 to M-Bus level converter must be used in conjunction with the gateway to allow the gateway to communicate on the M-Bus network. Figure 1 shows an example of the proper connection of the gateway to an M-Bus network.



Figure 1: M-Bus Connection Diagram

1.3 Master Settings

Baud Rate

Select the desired network baud rate.

Parity

Fixed at Even Parity.



Timeout

Sets the time in milliseconds that the driver will wait for a response from a device after sending a request.

Scan Rate

This is the time in milliseconds the driver will wait between sending requests. This is a useful feature for certain devices or infrastructure components (such as radio modems) that may not be capable of sustaining the maximum packet rates that the driver is capable of producing. The start time for this delay is taken with respect to the moment at which the driver is capable of sending the next packet (due to either reception or timeout of the previous request). If no additional time is required, setting this field to 0 instructs the driver to send its next request packet as soon as possible.

1.4 Service Object Settings

The M-Bus master driver uses service objects to describe what services the driver should perform. Each service object targets one data element which is defined by the Data Information Block (DIB) and Value Information Block (VIB). To read data (when enabled), the driver sends a REQ_UD2 request to the target device and searches the response for DIB and VIB matches to defined service objects. When data in the database changes where a service object is mapped, the driver will generate a SND_UD command to the target device for that service object attempting to write the value (when enabled.) Two different types of service objects are supported: variable data structure (read and write) and fixed data structure (read only).

Description

This 32-character (max) field is strictly for user reference: it is not used at any time by the driver.

Destination Address

Indicates the destination address or primary address of the remote slave device on the network that will be accessed by this service object. Enter a value of 0...250, 254, or 255. Note that address 254 is defined by M-Bus as the reply broadcast address and 255 is defined as the broadcast address. If the broadcast address is used (not available on fixed data structure service objects), the "Request Data (REQ_UD2)" function must be disabled, as attempts to read a service object targeting destination address 255 will invariably time out.

Database Address

Defines the database address where the data of this service object will be mapped. The configuration studio will not allow entry of a starting database address that will cause the service object to run past the end of the database. The highest valid database address, therefore, depends on the targeted data type.

Data Type

Specifies how the value associated with this service object will be stored in the database. This defines how many bytes will be allocated, whether the value should be treated as signed or unsigned, and whether the value should be interpreted as an integer or a floating point number. Select the desired data type from this dropdown menu.



Multiplier

The amount that associated network values are scaled by prior to being stored into the database or after being retrieved from the database. Upon retrieval from the database, raw data is multiplied by the multiplier to produce a network value (to be sent to the device). Similarly, network values (read from the device) are divided by the multiplier before being stored into the database.

1.4.1 Fixed Data Structure Settings

Data Coding

Select the data coding to search for in the response packet.

Function Coding

Select the function coding to search for in the response packet.

Units and Multiplier

Select the units and multiplier to search for in the response packet.

Request Data (REQ_UD2)

As fixed data structure service objects are read-only, the REQ_UD2 function is always enabled.

1.4.2 Variable Data Structure Settings

1.4.2.1 Data Information Block (DIB) Configuration

Manually Enter Bytes

This checkbox toggles between the ability to manually enter hex bytes for the DIB, versus configuring the DIB using the provided dropdowns and edit boxes.

DIB Bytes (in hex)

Defines the specific hex bytes to search for in the DIB.

Data Field

Only available when the “Manually Enter Bytes” checkbox is disabled. Select the desired data type to search for in the data field of the DIB. Selecting “No Data / Auto Detect” allows the driver to automatically detect the DIB based on a matching VIB field.

Function Field

Only available when the “Manually Enter Bytes” checkbox is disabled. Select the desired function to search for in the function field of the DIB.

LVAR

Only available when “Variable Length” is selected in the “Data Field” dropdown. Enter the desired value to search for in the LVAR byte.

Storage Number

Only available when the “Manually Enter Bytes” checkbox is disabled. Enter the storage number to search for in the DIB.



Tariff

Only available when the “Manually Enter Bytes” checkbox is disabled. Enter the tariff to search for in the DIB.

Subunit

Only available when the “Manually Enter Bytes” checkbox is disabled. Enter the subunit (or unit) to search for in the DIB.

1.4.2.2 Value Information Block (VIB) Configuration

Manually Enter Bytes

This checkbox toggles between the ability to manually enter hex bytes for the VIB, versus configuring the VIB using the provided dropdown.

VIB Bytes (in hex)

Defines the specific hex bytes to search for in the VIB.

Unit and Multiplier

Only available when the “Manually Enter Bytes” checkbox is disabled. Select the desired units and multiplier to search for in the VIB.

1.4.2.3 Functions

Request Data (REQ_UD2)

Check to enable reading: the service object will continuously read from the slave unless a pending write exists.

Send Data (SND_UD)

Check to enable writing: when values encompassed by this service object change in the internal database, these changes will be written down to the targeted slave.

1.5 Diagnostics Object

Each service object can optionally include a diagnostics object for debugging and diagnostics.

Diagnostics Database Address

Enter the database address at which to store the diagnostics information.



INDUSTRIAL CONTROL COMMUNICATIONS, INC.

1600 Aspen Commons, Suite 210
Middleton, WI USA 53562-4720
Tel: [608] 831-1255 Fax: [608] 831-2045

<http://www.iccdesigns.com>

Printed in U.S.A