MBEDIT – MODBUS MAP EDITOR

USER GUIDE

Revision 1.0

While this information is presented in good faith and believed to be accurate, RAP Systems does not guarantee satisfactory results from reliance upon such information. *Nothing contained herein is to be construed as a warranty or guarantee, express or implied, regarding the performance, merchantability, fitness or any other matter with respect to the products*, nor as a recommendation to use any product or process in conflict with any patent. RAP Systems reserves the right, without notice, to alter or improve the designs or specifications of the products described herein.

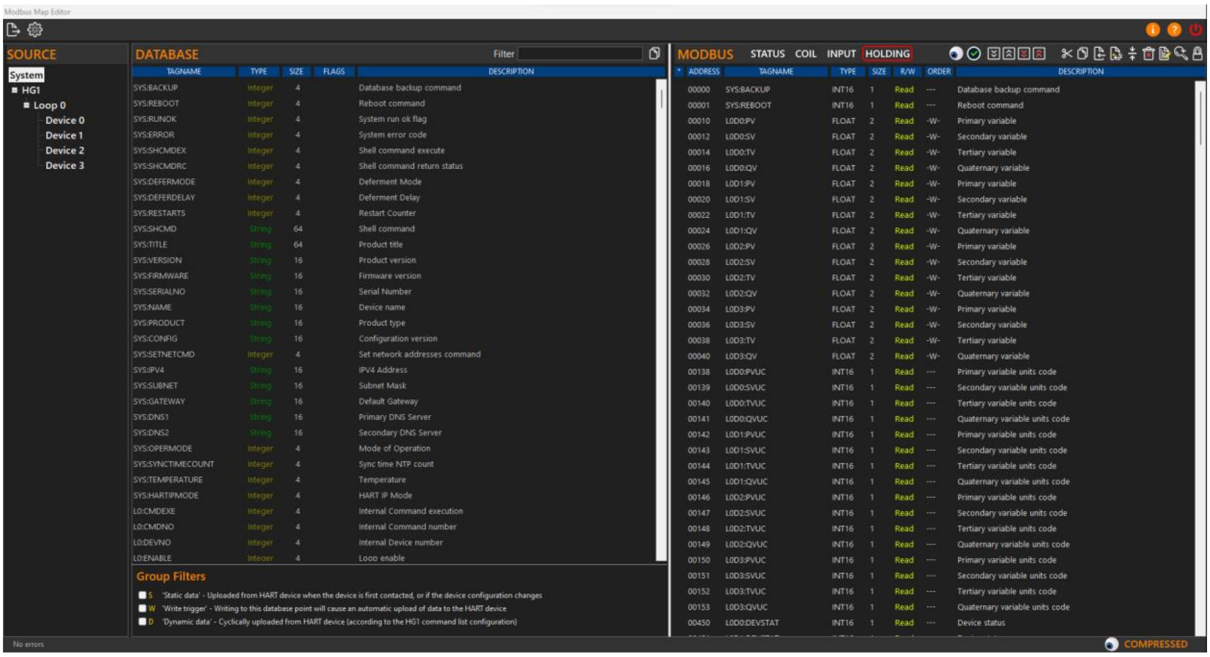
1 CONTENTS

2 OVERVIEW

2.1 INTRODUCTION

mbEdit is a windows utility used to edit Modbus register maps used in RAP systems products. It is installed as part of the software suite for the product.

The editor provides a single window interface with three panels from which all operations are performed, as shown below:



TAGNAME	TYPE	SIZE	FLAGS	DESCRIPTION
SYS:BACKUP	Integer	4		Database backup command
SYS:REBOOT	Integer	4		Reboot command
SYS:RUNOK	Integer	4		System run ok flag
SYS:ERROR	Integer	4		System error code
SYS:SHCMDEX	Integer	4		Shell command execute
SYS:SHCMDRC	Integer	4		Shell command return status
SYS:DEFERMODE	Integer	4		Deferment Mode
SYS:DEFERDELAY	Integer	4		Deferment Delay
SYS:RESTARTS	Integer	4		Restart Counter
SYS:SHCMD	String	64		Shell command
SYS:TITLE	String	64		Product title
SYS:VERSION	String	16		Product version
SYS:FIRMWARE	String	16		Firmware version
SYS:SERIALNO	String	16		Serial Number
SYS:NAME	String	16		Device name
SYS:PRODUCT	String	16		Product type
SYS:CONFIG	String	16		Configuration version
SYS:SETNETCMD	Integer	4		Set network addresses command
SYS:IPV4	String	16		IPV4 Address
SYS:SUBNET	String	16		Subnet Mask
SYS:GATEWAY	String	16		Default Gateway
SYS:DNS1	String	16		Primary DNS Server
SYS:DNS2	String	16		Secondary DNS Server
SYS:OPERMODE	Integer	4		Mode of Operation
SYS:SYNCTIMECOUNT	Integer	4		Sync time NTP count
SYS:TEMPERATURE	Integer	4		Temperature
SYS:HARTIPMODE	Integer	4		HART IP Mode
L0:CMDEXE	Integer	4		Internal Command execution
L0:CMDNO	Integer	4		Internal Command number
L0:DEVNO	Integer	4		Internal Device number
L0:ENABLE	Integer	4		Loop enable

Group Filters

- □ S 'Static data' - Uploaded from HART device when the device is first contacted, or if the device configuration changes
- □ W 'Write trigger' - Writing to this database point will cause an automatic upload of data to the HART device
- □ D 'Dynamic data' - Cyclically uploaded from HART device (according to the HG1 command list configuration)

ADDRESS	TAGNAME	TYPE	SIZE	R/W	ORDER	DESCRIPTION
00000	SYS:BACKUP	INT16	1	Read	---	Database backup command
00001	SYS:REBOOT	INT16	1	Read	---	Reboot command
00010	L0D0:PV	FLOAT	2	Read	-W-	Primary variable
00012	L0D0:SV	FLOAT	2	Read	-W-	Secondary variable
00014	L0D0:TV	FLOAT	2	Read	-W-	Tertiary variable
00016	L0D0:QV	FLOAT	2	Read	-W-	Quaternary variable
00018	L0D1:PV	FLOAT	2	Read	-W-	Primary variable
00020	L0D1:SV	FLOAT	2	Read	-W-	Secondary variable
00022	L0D1:TV	FLOAT	2	Read	-W-	Tertiary variable
00024	L0D1:QV	FLOAT	2	Read	-W-	Quaternary variable
00026	L0D2:PV	FLOAT	2	Read	-W-	Primary variable
00028	L0D2:SV	FLOAT	2	Read	-W-	Secondary variable
00030	L0D2:TV	FLOAT	2	Read	-W-	Tertiary variable
00032	L0D2:QV	FLOAT	2	Read	-W-	Quaternary variable
00034	L0D3:PV	FLOAT	2	Read	-W-	Primary variable
00036	L0D3:SV	FLOAT	2	Read	-W-	Secondary variable
00038	L0D3:TV	FLOAT	2	Read	-W-	Tertiary variable
00040	L0D3:QV	FLOAT	2	Read	-W-	Quaternary variable
00138	L0D0:PVUC	INT16	1	Read	---	Primary variable units code
00139	L0D0:SVUC	INT16	1	Read	---	Secondary variable units code
00140	L0D0:TVUC	INT16	1	Read	---	Tertiary variable units code
00141	L0D0:QVUC	INT16	1	Read	---	Quaternary variable units code
00142	L0D1:PVUC	INT16	1	Read	---	Primary variable units code
00143	L0D1:SVUC	INT16	1	Read	---	Secondary variable units code
00144	L0D1:TVUC	INT16	1	Read	---	Tertiary variable units code
00145	L0D1:QVUC	INT16	1	Read	---	Quaternary variable units code
00146	L0D2:PVUC	INT16	1	Read	---	Primary variable units code
00147	L0D2:SVUC	INT16	1	Read	---	Secondary variable units code
00148	L0D2:TVUC	INT16	1	Read	---	Tertiary variable units code
00149	L0D2:QVUC	INT16	1	Read	---	Quaternary variable units code
00150	L0D3:PVUC	INT16	1	Read	---	Primary variable units code
00151	L0D3:SVUC	INT16	1	Read	---	Secondary variable units code
00152	L0D3:TVUC	INT16	1	Read	---	Tertiary variable units code
00153	L0D3:QVUC	INT16	1	Read	---	Quaternary variable units code
00450	L0D0:DEVSTAT	INT16	1	Read	---	Device status

The right panel entitled 'MODBUS' displays a table of registers in the current Modbus map, new registers can be added, existing registers can be modified or deleted. The central panel called 'DATABASE' lists the available database objects that can be added to the Modbus map. The left most panel 'SOURCE' is a tree structure that is used to select which objects are visible in the 'DATABASE' table, this makes finding the desired database objects easier. The process of creating a Modbus map involves selecting database object from the DATABASE panel and copying them into Modbus registers in the MODBUS panel. Various tools are available the toolbars above the panels to facilitate this process.

The remainder of this document describes the use of mbEdit in configuring Modbus maps for the HG1 hart gateway, however the general principle of operation applies to its use with all RAP systems products.

3 USING THE EDITOR

3.1 STARTING MBEDIT

mbEdit will be automatically opened when you select the **Edit Register Map** tool from the Modbus menu of HGxConfigure. Any changes you make to the Modbus map will be saved to the HG1 project when you close the editor.

On opening, the editor will display (in the DATABASE panel) the available database objects from the HG1 (which will depend on the current HG1 configuration) and the current content of the map in the MODBUS panel.

3.2 THE SOURCE PANEL

The HG1 database contains many objects, some relate to the HG1 itself (such as SYS:RESTART – the number of times the HG1 has restarted), some relate to the instrument loop (such as L0:MAXDEV – the number of devices on the loop) and some relate to instruments on the loop (such as L0D0:PV – the primary variable for device 0). To focus on specific database objects, you can use the SOURCE tree structure to display objects you are interesting in. For example, selecting 'DEVICE 0' from the source tree will display database objects related to HART device 0. The source tree selection works as follows:

System: All database object for the entire system.

HG1: Objects related to the HG1 device itself.

LOOP 0: Objects related to the physical loop interface.

Device n: Objects related to attached HART device number 'n'.

3.3 THE DATABASE PANEL

The database panel displays a table of database objects available for inclusion in the Modbus map. Objects from this panel can be copied and pasted to the Modbus register panel to its right.

Each database object is shown on a single line and occupies several columns, these are:

TAGNAME: This is the name of the data base object which is formed of two parts <origin>:<content> e.g. SYS:VERSION The <origin> can be used to determine source of the data for the object

SYS: The datapoint is related to the HG1 device itself

L0: The datapoint is related to the current loop interface 0

L0Dx: The datapoint is related to instrument number 'x' on the loop. For example L0D4:PV is the primary variable for device 4 on loop 0

TYPE: The type of the data, this may be integer, float, or string (text message).

SIZE: This size (in bytes) of the database object

FLAGS: Any flags associated with the database object – the meaning of the flags are device specific and can be seen in the Group Filters section below to the database panel.

DESCRIPTION: A description of the database object.

3.3.1 FILTERING

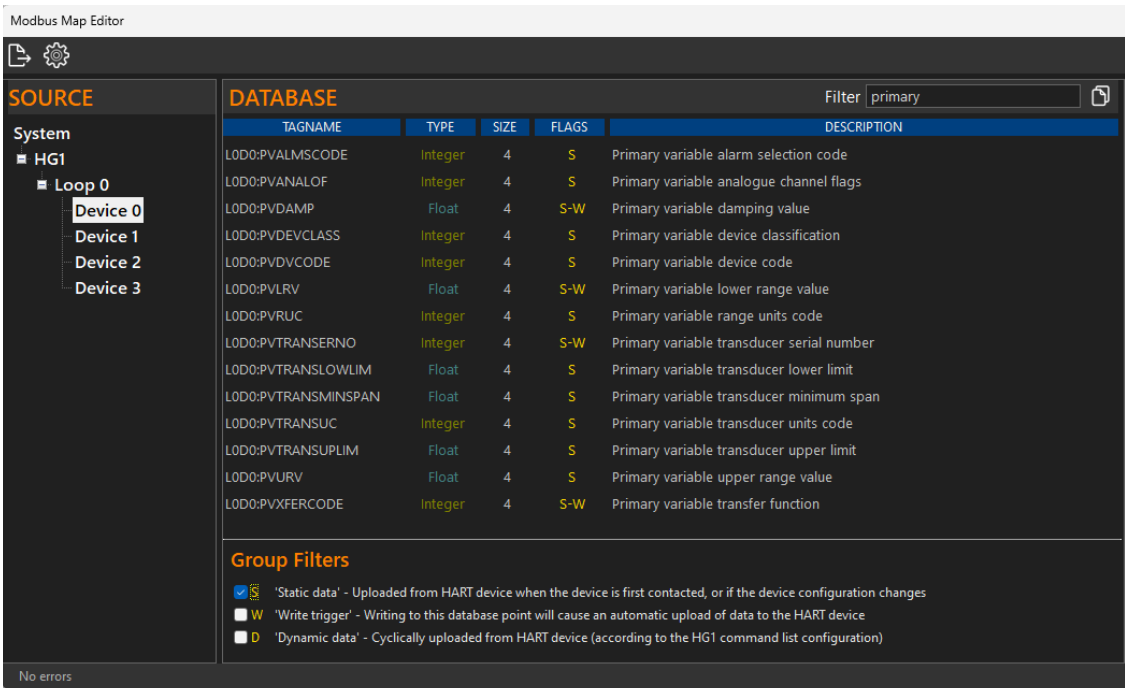
The table of database objects can be large, to ease the task of finding objects of interest two mechanisms for filtering the table are provided**.**

Tagname & Description Filter: The panel header contains a filter in which you can enter text. Only database objects whose tagname or description contain the text in the filter box are displayed (not case sensitive). For example, to find objects related to the primary variable you may enter “primary” or “PV” into the filter box. Clearing the box will result in all objects being displayed.

Group Filters: Further selection of objects can be made by ticking any of the checkboxes next to the group filters shown below the database panel. These groups are device specific; their meaning is displayed next to the group filter checkboxes.

Filters are applied in the following order: First the selection from the source tree, then the tagname filter and finally the group filter.

The example below shows how to find all database objects related to the primary variable for device 0 that are static, i.e., are part of the device configuration and don’t change with the measure variable. First the ‘Device 0’ is selected in the source tree on the left, then ‘Primary’ is entered into the tagname filter, and finally the ‘static data’ group flag is checked.



Filter primary

TAGNAME	TYPE	SIZE	FLAGS	DESCRIPTION
L0D0:PVALMSCODE	Integer	4	S	Primary variable alarm selection code
L0D0:PVANALOF	Integer	4	S	Primary variable analogue channel flags
L0D0:PVDAMP	Float	4	S-W	Primary variable damping value
L0D0:PVDEVCLASS	Integer	4	S	Primary variable device classification
L0D0:PVDVCODE	Integer	4	S	Primary variable device code
L0D0:PVLRV	Float	4	S-W	Primary variable lower range value
L0D0:PVRUC	Integer	4	S	Primary variable range units code
L0D0:PVTRANSERNO	Integer	4	S-W	Primary variable transducer serial number
L0D0:PVTRANSLOWLIM	Float	4	S	Primary variable transducer lower limit
L0D0:PVTRANSMINSPAN	Float	4	S	Primary variable transducer minimum span
L0D0:PVTRANSUC	Integer	4	S	Primary variable transducer units code
L0D0:PVTRANSUPLIM	Float	4	S	Primary variable transducer upper limit
L0D0:PVURV	Float	4	S	Primary variable upper range value
L0D0:PVXFERCODE	Integer	4	S-W	Primary variable transfer function

Group Filters

- ☑ S 'Static data' - Uploaded from HART device when the device is first contacted, or if the device configuration changes
- ☐ W 'Write trigger' - Writing to this database point will cause an automatic upload of data to the HART device
- ☐ D 'Dynamic data' - Cyclically uploaded from HART device (according to the HG1 command list configuration)

3.3.2 SELECTING

1. Single Object Selection:

To select an individual object from the table, simply move your cursor over the object and click once with your mouse. This action highlights the chosen object, indicating it is selected.

2. Multiple Objects Selection Using the CTRL Key:

If you wish to select multiple objects one by one, hold down the 'CTRL' key on your keyboard. While holding 'CTRL', click on each object you want to select. Each click will add the object to your selection group.

Remember, you must keep the 'CTRL' key pressed down while selecting each additional object.

3. Selecting Multiple Objects by Dragging:

To select a group of objects by dragging, place your cursor at one corner of the desired group of objects.

Press and hold down the left mouse button. Then, drag your mouse diagonally across the objects. A selection box will appear, encompassing the objects within its area.

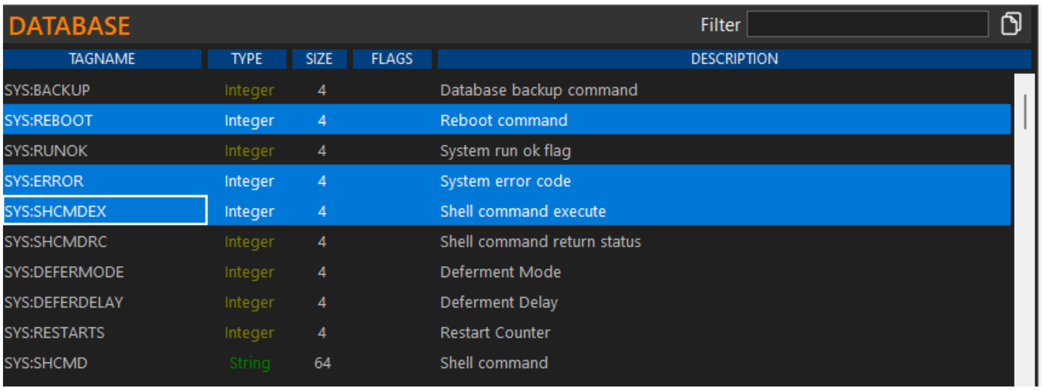
Release the mouse button once all the desired objects are within the selection box. This action selects all objects within the box's area.

4. Selecting all objects

To select all the object in the table, press ‘CTRL’ key and the letter ‘a’.

Note:

If you accidentally select an unwanted object, simply use the 'CTRL' key method to deselect it by clicking on the object while holding down the 'CTRL' key.



TAGNAME	TYPE	SIZE	FLAGS	DESCRIPTION
SYS:BACKUP	Integer	4		Database backup command
SYS:REBOOT	Integer	4		Reboot command
SYS:RUNOK	Integer	4		System run ok flag
SYS:ERROR	Integer	4		System error code
SYS:SHCMDEX	Integer	4		Shell command execute
SYS:SHCMDRC	Integer	4		Shell command return status
SYS:DEFERMODE	Integer	4		Deferment Mode
SYS:DEFERDELAY	Integer	4		Deferment Delay
SYS:RESTARTS	Integer	4		Restart Counter
SYS:SHCMD	String	64		Shell command

Having selected the items you want, you can copy them to the clipboard pressing CTRL-c, or by pressing the right mouse button and selecting ‘copy’ from the drop-down menu. Items copied to the clipboard can be pasted into the Modbus map – see next section.

3.4 THE MODBUS PANEL

The Modbus panel displays the registers of current Modbus map. Registers are displaying ascending order of address, with low address at the top and higher address further down. Each row in the table represents one register and is comprised of several columns:

ADDRESS: This is the address of the register. When viewing the map in 'compact view' (see later for description of view types) for database objects that occupy multiple registers, such as floating-point values, this is the address of the first register.

TAGNAME: This is the name of the database object associated with the register.

TYPE: This column defined how the database object will be encoded in the Modbus register(s). This may be one of:

INT16: A sixteen-bit signed integer (requires 1 Modbus register)

INT32: A 32-bit signed integer (requires 2 Modbus registers)

FLOAT: A 32-bit floating-point number (requires 2 Modbus registers)

DOUBLE: A 64-bit floating-point number (requires 4 Modbus registers)

STRING: An array of characters (requires 1 register for every 2 characters in the string)

Note: the type of the Modbus register can be different from the type of the associated database object.

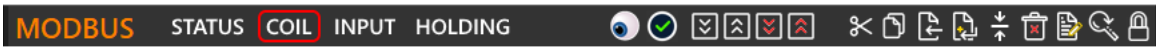
SIZE: The number of Modbus registers used to represent the database object

R/W: Read / Write flag. Most registers will be defined as read as they are intended to be used to read data from the HG1. However some objects in the database can be written as well as being read (such as L0D0:PVDAMP – Primary Variable Damping Value). In cases where you want to use a Modbus register to write to a database point, you should set the write flag using the register editor. Note that if you want to both read and right to a database point you should define two registers pointing to the same database object, one for read and one for write.

ORDER: This column displays the byte ordering applied to the object. See the section on byte ordering for mor details.

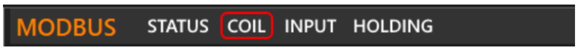
DESCRIPTION: A description of the contents of the register.

3.4.1 TOOLBAR



The Modbus panel includes a toolbar that allows for editing and moving of registers within the map. These tools are also available on a drop-down menu opened by left clicking anywhere in the Database panel. Most tools have an equivalent keyboard shortcut which can be seen in the drop-down menu.

Register sections:



Modbus registers are arranged into four sections: Status Registers, Coil registers, Input Registers and Holding Registers. You select which section is displayed using the four text buttons on the Modbus panel tool bar. The selected section is highlighted with a red border.

View style

Register information can be view in three different ways, and you can toggle through these different views using the toggle button. The in-use view style (Compressed, Expanded, All) is shown on the right-hand side of the bottom status bar.

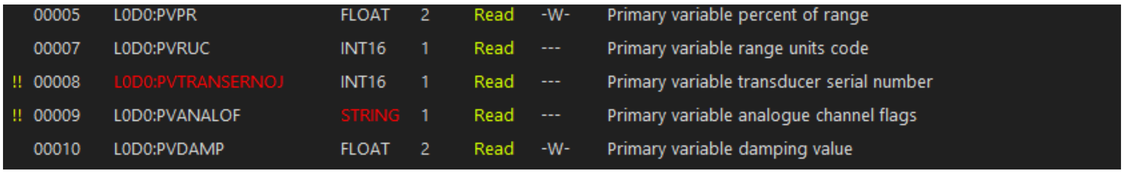
Compressed: This style shows only registers that have been assigned a database object, however for database objects that require more than on Modbus register (i.e. a 32-bit floating point value that occupies 2 consecutive 16-bit Modbus registers) only the starting register is shown.

Expanded: This style shows registers that have been assigned a database object, including all registers that are part of a multi-register object.

All: This view shows all registers, both configure and unconfigured. This view makes it easier to visualise how database objects are spread within the Modbus map.

Delete Registers with Errors

This is a convenience tool used to delete any registers that have errors. Register with errors are preceded by a double exclamation mark, and red text is used for the field that is in error.



```
   00005  L0D0:PVPR           FLOAT   2  Read  -W-  Primary variable percent of range
   00007  L0D0:PVRUC          INT16   1  Read  ---  Primary variable range units code
!! 00008  L0D0:PVTRANSERNOJ   INT16   1  Read  ---  Primary variable transducer serial number
!! 00009  L0D0:PVANALOF       STRING  1  Read  ---  Primary variable analogue channel flags
   00010  L0D0:PVDAMP        FLOAT   2  Read  -W-  Primary variable damping value
```

Errors are caused by:

Illegal database tagname: the tagname in the register definition does not match any of the available database object from the Database panel.

Illegal Type: the specified register type is not valid for the selected database object (e.g., a database object of type float cannot be represented in the Modbus register as a string).

Illegal Size: the number of registers configured is not compatible with the size of the database object (e.g. a database string object of size 8 characters needs 4 registers to represent it – 2 characters per register).

Note, you can correct errors by double clicking on the register to open the register editor and editing the register details.

Shift registers down

This tool is used to increment the addresses of selected registers by 1, effectively moving then 'down' the register map (given that lowest register address is displayed at the top). This shift down is only executed if there are empty registers to shift into.

Shift registers up

This tool is used to decrement the addresses of selected registers by 1, effectively moving then 'up' the register map (given that lowest register address is displayed at the top). This shift up is only executed if there are empty registers to shift into.

Shove registers down

This tool is used to increment the addresses of selected registers by 1, effectively moving then 'down' the register map (given that lowest register address is displayed at the top). Unlike the shift function, this tool shoves lower registers out of the way (moves them down) to make room for the selected registers.

Shove registers up

This tool is used to decrement the addresses of selected registers by 1, effectively moving then 'up' the register map (given that lowest register address is displayed at the top). Unlike the shift function, this tool shoves preceding registers out of the way (moves them up) to make room for the selected registers.

Cut

Cuts the selected registers. This action copies the selected registers to the clipboard and deletes them from the register map. You can paste the cut registers using the paste or append tool.

Copy

Copies the selected registers to the clipboard. You can paste the cut registers using the paste or append tool.

Paste

Inserts the contents of the clipboard into the register map. The clipboard contents will be inserted starting at the register address immediately following the one currently selected. If there are insufficient free addresses to fit the contents of the clipboard, then registers will be moved down as needed to make room.

Append

Inserts the contents of the clipboard into the register map. The clipboard contents will be inserted starting at the first free register address after the register currently selected. If there are insufficient free addresses to fit the contents of the clipboard, then registers will be moved down as needed to make room.

Compress

Moves the selected registers together into one continuous block, removing any unused register address between them. Gap are closed by moving lower register addresses upwards (i.e. from higher to lower address).

Delete

Deletes the selected registers.

Edit

Opens an editor to edit aspects of the selected register.

Search & Replace

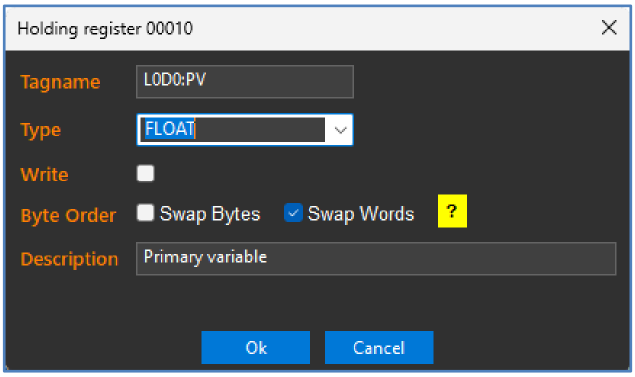
Performs a search and replace function of the database name for selected register. This tool allows you to replace selected text within the database names or all selected registers. For instance, if you have selected a group of registers whose database names contains L0D0, you can use this tool to replace the L0D0 portion of the database names with L0D1.

Lock / Unlock

This tool locks or unlocks a register. Locked registers cannot be deleted, moved, or edited. Locked registers are identified by a red 'L' system preceding their address.

3.4.2 EDITING

Details of a Modbus register can be edited by either double clicking on a register or using the editor tool from the toolbar. This will open a dialog box in which you can modify aspects of the register definition.



Holding register 00010

Tagname: L0D0:PV

Type: FLOAT

Write: □

Byte Order: □ Swap Bytes ☑ Swap Words

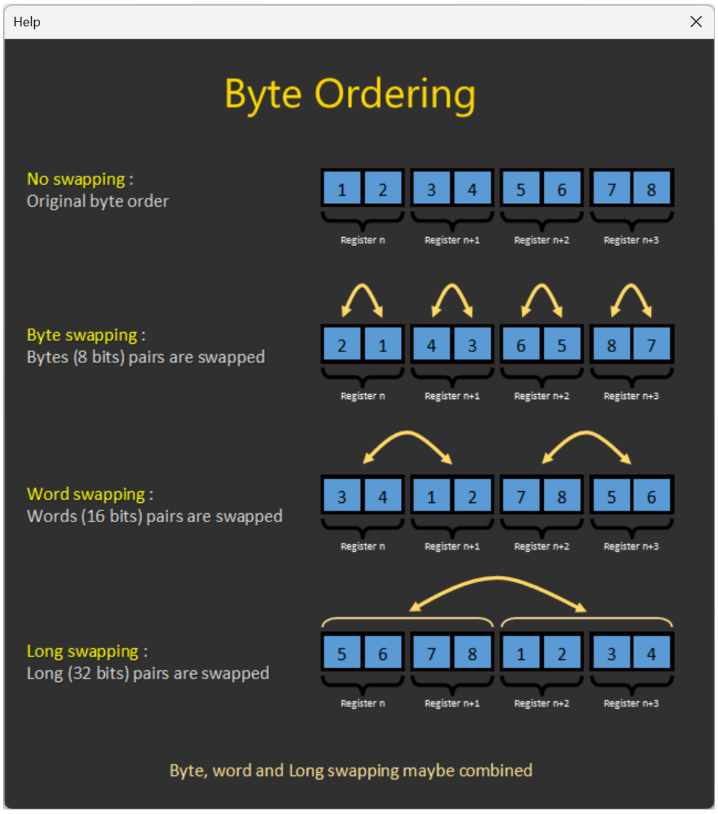
Description: Primary variable

Using this editor, you can manually enter a database name, set the desired data type, mark the register as being for read or write and specify the byte ordering and description. You can use this editor on existing registers, or by double clicking on an unused empty register, you can create a new one.

3.4.3 BYTE ORDERING

Different Modbus master device (PLCS, DCS or SCADA systems) often require specific ordering of the bytes returned in the Modbus data. This ordering can be specified in the Modbus editor both at a global level or on a register-by-register basis. The global settings are configured using the settings to on the system toolbar and are applied to any object copied into the map from the database panel. You can choose to modify the byte ordering of any specific register by using the register editor described above.

The meaning or the byte ordering settings are defined as:



Help

Byte Ordering

No swapping : Original byte order

1 2 | 3 4 | 5 6 | 7 8

Byte swapping : Bytes (8 bits) pairs are swapped

2 1 | 4 3 | 6 5 | 8 7

Word swapping : Words (16 bits) pairs are swapped

3 4 | 1 2 | 7 8 | 5 6

Long swapping : Long (32 bits) pairs are swapped

5 6 | 7 8 | 1 2 | 3 4

Byte, word and Long swapping maybe combined

3.5 SYSTEM TOOLBAR

Additional tools are available from the system toolbar:

Export Modbus map

This tool is used to save the Modbus map to a text file. You can choose to save the file either as a csv file suitable for opening with excel, or as a plain text file. Choose the output file type by using ‘save as type’ drop down menu from within the Export dialog.

Settings

Used to set the default byte ordering for data in the Modbus map. Note that byte ordering can be set on an individual register basis by using the Edit tool on any selected register.

About

Displays information about mbEdit such as version numbers.

Help

Provides access to mbEdit user documentation.

Exit

Prompts user to save changes and closes the program.