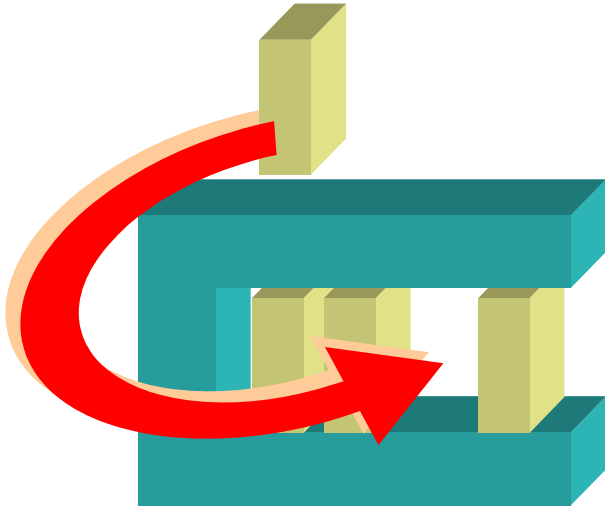


SIEMENS



MM8000 MP4.xx Modbus Control Units

Add-on module

Installation and Configuration
Guide

Data and design subject to change without notice. / Supply subject to availability.

© 2009 Copyright by
Siemens Switzerland Ltd

We reserve all rights in this document and in the subject thereof. By acceptance of the document the recipient acknowledges these rights and undertakes not to publish the document nor the subject thereof in full or in part, nor to make them available to any third party without our prior express written authorization, nor to use it for any purpose other than for which it was delivered to him.

About this document.....	1
1 Introduction	3
1.1 Modbus protocol.....	3
1.2 Modbus subsystems	3
1.3 Modbus add-on module	3
1.4 Version supported	3
2 Installation	4
2.1 Distribution package	4
2.1.1 Installation checklist	5
2.2 Software installation	6
2.2.1 Requirements.....	6
2.2.2 Software License.....	6
2.2.3 Modbus add-on installation	6
2.2.3.1 Installation on MM8000 MP4.10	6
2.2.3.2 Installation on MM8000 MP4.15 and later	7
2.2.3.3 Multiple add-on's installation	9
2.2.4 Modbus add-on uninstall.....	9
2.3 Modbus model customisations.....	9
2.3.1 Modbus Customisation tool.....	10
2.3.1.1 User Interface.....	10
2.3.1.2 Configuration fields	10
2.3.1.3 Main menu and toolbar	13
2.3.1.4 Configuration procedure	15
2.3.2 Control unit icon	16
2.4 CSV import file	17
2.4.1 CSV file structure	17
2.5 Communication network	18
2.5.1 NK82xx firmware download	18
3 Configuration.....	21
3.1 Configuration checklist.....	21
3.2 Configuration procedure	22
3.3 Configuration backup and restore.....	25
Annex A – Modbus meta-models in Poseidon	26
Annex B - Communication parameter settings	27
Annex C – Priority Flag.....	28
Annex D – Compatibility specifications for Modbus units.....	29

About this document

Purpose of this document

This manual is a guide to the installation, configuration, and operations for the MM8000 Management Stations that includes control units supporting the Modbus protocol. It presents the MM8000 add-on module for this type of subsystems.

Scope

This document applies to the MM8000 Management Station MP4.10 and higher.

Target readers

This documentation is intended for the following users:

- Project Managers
- Project Engineers
- Commissioning Personnel

Individuals performing the operations described in this manual are expected to have prior expertise and training in the field of security, at least a moderate level of familiarity with the Siemens Building Technologies product line, and experience with the installation, configuration, and commissioning of security management systems.

Also, a good knowledge of the Modbus standard, as applied to the specific implementation, is required for properly performing the configuration steps.

Reference documents

The **DMS8000 Documentation Resource Information Guide** document assembles in one place important information regarding documentation resources. It contains the following:

- Comprehensive definitions of the target audiences for FS DMS documents
- Training program information including the Siemens intranet link
- A complete list of all available DMS8000 documents
- Instructions for how to obtain a document via the Siemens intranet using the STEP Documentation Repository System
- A map of relevant documents for each target audience group
- Customer Support links & resources
- A glossary containing definitions of all terms and acronyms used in DMS8000 documentation

To access the **DMS8000 Documentation Resource Information Guide** (STEP #A6V10089056), go to the link and follow the instructions below:

<https://workspace.sbt.siemens.com/content/00001123/default.aspx>



1. Click on the **STEP WEB Client** image:
2. Choose **04 Fire -3F** from the **Product Segment** box and select **Activate filter**.
3. Select **All** in the **Documents** section of the **Quick Search** page and then select **Advanced Search**.
4. Enter the document number in the **Brochure No.** field (e.g. A6V10089056) and press **Enter**.

Operational and safety regulations



Before beginning work on the MM8000 Management Station, you must have read and understood the Operational and Safety Regulations included in the following documents:

- A6V10062425 - DMS8000 Connectivity Configuration Guide
- A6V10062413 - MM8000 Installation, Configuration and Commissioning
- A6V10062437 - NK8000 Installation, Configuration and Commissioning

Liability disclaimer for damage or injuries

Before products are delivered, they are tested to ensure they function correctly when used properly. Siemens disclaims all liability for damage or injuries caused by the incorrect application of the instructions, or the disregard of danger advisories. This disclaimer applies in particular to personal injuries or damage caused by:

- Improper and/or incorrect use.
- Disregard of safety instructions in the documentation or on the product.
- Poor maintenance or a lack of maintenance.

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcome.

Modification index

Document index	Date	Notes
A6V10067800_a_en	06.2009	Corresponds with version MP4.20 of the MM8000 software
A6V10067800_a_en	10.2008	Corresponds with version MP4.10 of the MM8000 software
A6V10067800_a_en	04.2007	Corresponds with version MP3.20-01 of the MM8000 software

1 Introduction

1.1 Modbus protocol

Modbus is a very popular protocol applied in process-control and industrial installations to exchange messages between computers and control units.

The Modbus standard defines both data-link and application layers of the protocol. Modbus is designed to model the control unit conditions into a set of binary data that can be easily mapped into the process image memory of central stations.

→ For more information about Modbus, please refer to: www.Modbus-IDA.org

1.2 Modbus subsystems

In this document, control unit that can communicate via Modbus are generally called **Modbus subsystems**.

Each unit type applies the Modbus standard according to a specific model, including a point and command map.

1.3 Modbus add-on module



Although a list of predefined models are available (see Annex B on p.27), it must be made clear that in general the Modbus add-on module for MM8000 **is not a ready-to-use solution** and is intended as an open tool-kit that allows for integrating Modbus subsystems via NK82xx. Each subsystem requires in fact that a specific Modbus model is defined and configured using the MM8000 tools.

Therefore, the Modbus integration will typically require a number of contacts and technical verifications with the HQ support that may require some weeks. Project planning must be defined accordingly.

In fact, depending on the modelling solutions, multiple applications of the Modbus protocol may exist for the various control units. A customisation tool is provided in the MM8000 add-on, along with the communication and configuration software, in order to adapt each system to the actual installed units.

1.4 Version supported

The reference Modbus version is 1.1a.



Note that Modbus protocol is supported with very specific restrictions. Refer to Annex D (see p.22) for the detailed technical specifications. Before planning for integrating a control unit in MM8000, make sure that it can match the MM8000 compatibility specifications.

2 Installation

2.1 Distribution package

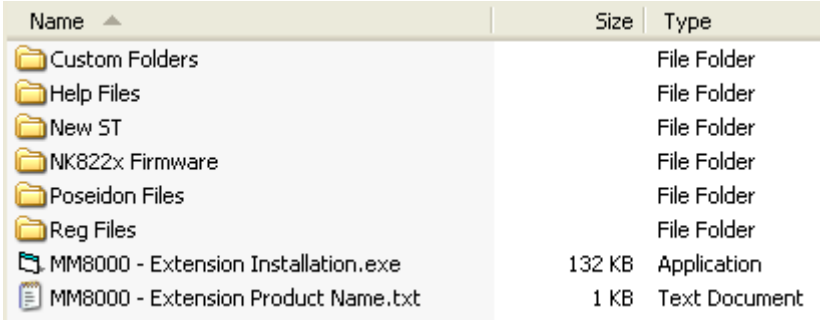
The M8000 software for the Modbus support is distributed as an add-on package, to be installed on the stations including the Composer tool (client-only and FEP stations are therefore excluded) after the standard MM8000 Setup.

The package is named: **MM8000 MPx.xx.xx - System Extension N.07 (ModBus V.1.21)** and is made up by an installation kit of a few files.

Installation kit

The installation kit includes (Fig. 1):

- A **Custom Folders** folder, containing the Modbus customisation tool for the specific application.
- The new **Help files**, describing the Modbus configuration procedures.
- The new Composer Subsystem Tool (**New ST**) for the Modbus models.
- The **NK82xx firmware**, i.e. an additional DLL module(s) supporting the Modbus protocol.
- The **Poseidon files**, including the definitions of the Modbus data structures.
- The registry file (**Reg Files**) folder, containing a command for registering the add-on package.
- The installation utility: the **MM8000 – Extension Installation.exe** program.
- The extension name text file; e.g.: **MM8000 – Extension Product Name**.



Name	Size	Type
Custom Folders		File Folder
Help Files		File Folder
New ST		File Folder
NK822x Firmware		File Folder
Poseidon Files		File Folder
Reg Files		File Folder
MM8000 - Extension Installation.exe	132 KB	Application
MM8000 - Extension Product Name.txt	1 KB	Text Document

Fig. 1 Installation kit

2.1.1 Installation checklist

ITEMS NEEDED FOR THE INSTALLATION

- The MM8000 Setup CD
- The add-on installation kit (MM8000 MPx.yy - System Extension N.zz ...)
- The MM8000 hardware key (dongle)
- The MM8000 license PAK code (or the REG file that contains it)

INSTALLATION CHECKLIST

- 1. Install the MM8000 Software and the required license key and PAK
 → See MM8000 Release Notes
 → See MM8000 Installation, Configuration and Commissioning
- 2. Install the NK8000 units (NK82xx)
 → See NK8000 Installation, Configuration and Commissioning
- 3. On the station(s) with configuration capability (Composer tool),
 install the Modbus add-on p. 6
- 4. Customise the Modbus model according to the modelling solution p. 9
- 5. Update the NK8000 firmware p. 18

2.2 Software installation

2.2.1 Requirements

The support for Modbus does not add any special requirements to the standard MM8000 setup. Therefore, software and hardware requirements are the same as for the base MM8000 software, as described in the document no. A6V10062413, MM8000 Installation, Configuration and Commissioning manual.

As far the NK8000 network is concerned, the requirements are described in the document no. A6V10062437, NK8000 Installation, Configuration and Commissioning manual.

The MM8000 software must be properly installed before the add-on can be installed. For more information on the MM8000 installation, please see the mentioned A6V10062413 document.

2.2.1.1 MM8000 Version compatibility

This Modbus add-on package is designed to work with MM8000 MP4.10 and higher. Contact FSP-DMS support to verify the compatibility with other versions.

2.2.2 Software License

An additional license is required to run the Modbus module. On top of the base MM8000 license codes, a specific PAK is therefore needed.

Therefore, the required license includes:

- WW8000 Composer (project configuration and download): Composer License or Service key.
- NS8210 driver: NK8000 connections, indicating the number of NK82xx units. This license is required for enabling the network driver and the NK82xx units communicating with the Modbus control units.
- MM8000 core, no. of subsystems. This license should include the number of Modbus control units.
- MM8000 core, no. of devices. This license should include the number of Modbus physical objects (detectors, auxiliary and control outputs).
- Modbus add-on license.
→ Check detailed sales policy for your country

Other licenses, covering more MM8000 options, may or may not be used and they are not related to the Modbus support.

2.2.3 Modbus add-on installation

The following are the installation procedures for the Modbus add-on module.



Depending on the MM8000 version a different installation procedure is foreseen. Please select the appropriate procedure.

2.2.3.1 Installation on MM8000 MP4.10

1. Copying files

The add-on installation is quite simple. The installation kit includes the **MM8000 – ST extension Installation.exe** utility (Fig. 2 below). Just run this program and the add-on files will be copied onto the local hard disk in the appropriate folders.

Name	Size	Type
Custom Command Timeout		File Folder
Default Project		File Folder
Help Files		File Folder
New ST		File Folder
NK822x Firmware		File Folder
Poseidon Files		File Folder
MM8000 - ST extension Installation.exe	56 KB	Application
MM8000 - ST extension Product Name.txt	1 KB	Text Document

Fig. 2 Starting the add-on installation

2. Installing the Subsystem Tool I

Composer requires that the tools are installed using a specific procedure. Therefore, a small utility is also launched in order to add the Modbus Subsystem Tool (ST) in to the Composer tool set.

The utility shows as illustrated in Fig. 3. Click **Install** to proceed.

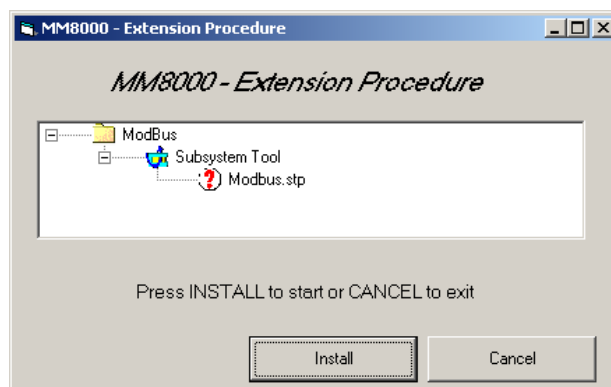


Fig. 3 Installing the Modbus Subsystem Tool

In few seconds, the tool is installed in Composer. The name of the new tool appears in the list (Fig. 4). At this point, click Exit to quit.

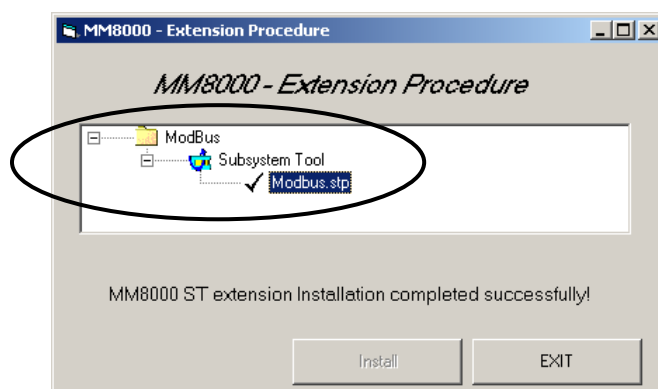


Fig. 4 Closing the installation tool

2.2.3.2 Installation on MM8000 MP4.15 and later

A specific application, the Add-on manager, is provided in MM8000 MP4.15 and later and allows installing and updating the add-on modules. The application sup-

ports the installation of add-ons developed for all MM8000 versions starting from MP3.20.

The following describes the installation procedure.

1. Start Add-on Manager.

From the Windows Start menu, select the following:

Start → DMS8000 → Tools → Add-on Manager

The Add-on Manager window appears (Fig. 5).

2. In the list of add-ons, select the name of the module that you wish to install

-- OR --

click the **Browse ...** button, locate the installation files of the add-on module and select the text file (**Extension Product Name.txt**) in the root folder (Fig. 6).

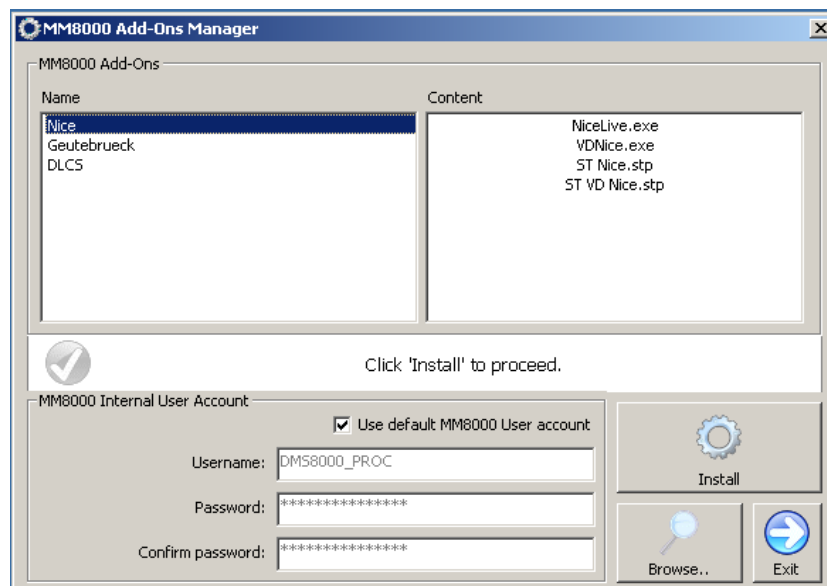


Fig. 5 Add-on Manager Window

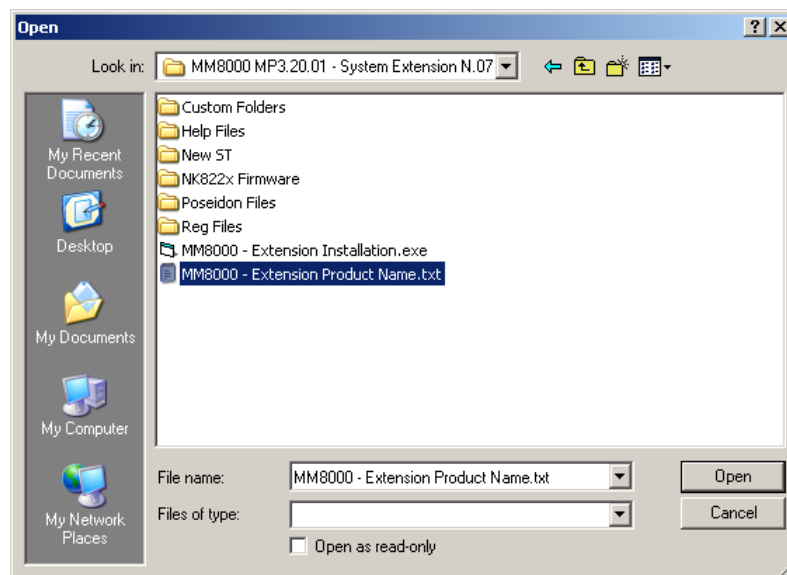


Fig. 6 Selecting the MM8000 Extension Product Name file

3. If your MM8000 installation includes a customized MM8000 internal account, then deselect the checkbox **Use default MM8000 User account** and specify the customized username and password.

4. Click **Install** (or **Update** if a previous version is detected).

2.2.3.3 Multiple add-on's installation

In general, it is possible to install multiple add-on packages and benefit of their combined functionalities. However, specific incompatibilities might exist. Please refer to the documentation of the other add-on modules.

2.2.4 Modbus add-on uninstall

The Modbus add-on module cannot be uninstalled.

2.3 Modbus model customisations

The Modbus standard defines a general message structure which can be applied to any data communication involving a set of multi-state variables.

In MM8000 applications, Modbus can be used using a 4-step approach:

1. Meta-model definitions

A general set of meta-model definitions have been define in the web-based Poseidon modelling environment (refer to FS-DMS document 2004-0217-00-Eng). The meta-models, typically organised per discipline (e.g. fire units meta-model), comprise a superset of all the applicable objects.



The Poseidon's meta-models are the base for any further steps. If the available meta-models are somehow not sufficient for a specific application, please contact the customer support in order to define new requirements.

When modifying or creating meta-models in Poseidon, please be aware of the specific requirements discussed in Annex A.

2. Modbus customisation tool

The *meta-models* can be applied in the Modbus customisation tool, which is included in the Modbus add-on package, whose task is to define applicable Composer models for the Modbus compatible units, i.e. creating *Modbus subsystem models*.

The composer models can make use of the meta-model objects, freely allocate them in the Modbus subsystem model structure and finally link the objects states to the actual Modbus messages. The result of this task consists of data files for Composer and NK8000 devices.

3. CVS import file

Instances of the subsystem model can be prepared in a CSV file to import into Composer for mapping the corresponding states (or Modbus messages) of a real control unit.

4. Composer

In Composer, once the model data file is available, the **Modbus subsystem model** can be used in the MM8000 projects to create a new type of node. The CSV file allows for importing the entire unit structure into the Composer project.

Note that this document does not discuss the step 1, which requires a specific training. Instead, step 2 is presented in the next section, whereas the structure of the CSV import file (step 3) is illustrated in 2.4 at page 17. Step 4 refers to a typical Composer configuration that is illustrated in section 3 at page 21.

Depending on the specific installation, the Modbus subsystem model may be already available for your specific requirements. In this case, the entire step 2 can be skipped.

2.3.1 Modbus Customisation tool

The tool is a Windows program named **MBUtility.exe**, which is available in the folder:

<Installation folder>\DMS8000\Composer\Configuration Data\OTD_Files\
Note: <Installation folder> is usually **C:\Program Files**

2.3.1.1 User Interface

Starting the program, the window shown in Fig. 7 appears.

The tool interface is organised in three parts:

- On the top, an area where name and type of the selected model are shown; also, some general configuration flags are present here.
- In the middle, the objects list is defined and the states of each object can be associated to Modbus messages.
- In the lower part of the window, the control commands can be associated to equivalent Modbus commands.

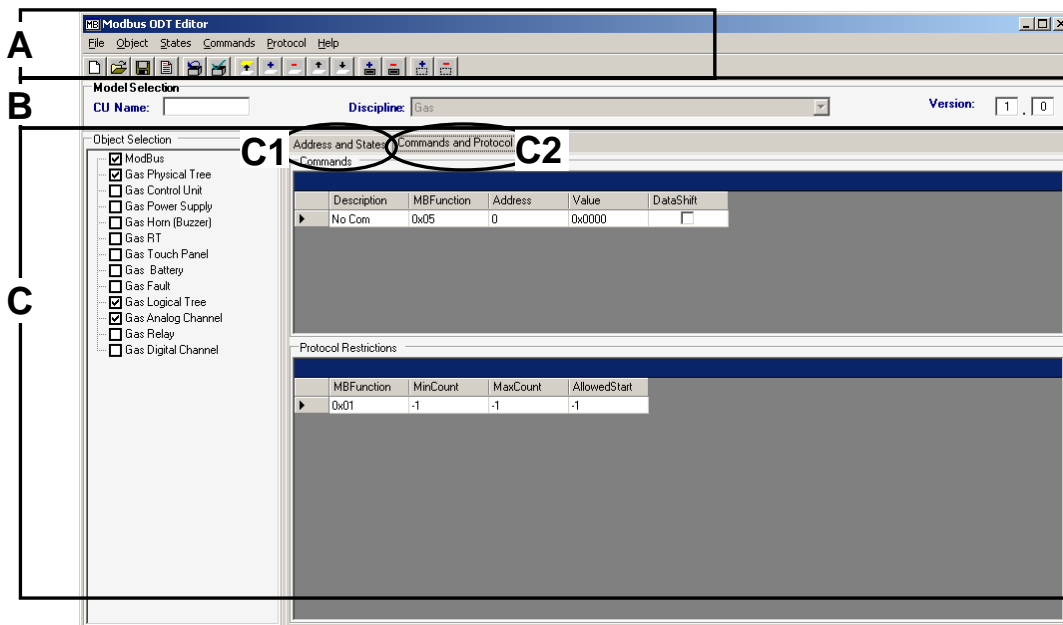


Fig. 7 Modbus tool interface

- A Menu and Toolbar
- B Model Selection
- C Working Area: Object Selection list and Object configuration: Address and States (C1 tab), Commands and general Protocol option (C2 tab)

2.3.1.2 Configuration fields

Model Selection

CU name: name of the model currently opened: a free text.

Discipline: application discipline, which defines the meta-model that is applied and the correspondent superset of object list. Note that, after you define a specific object to be part of the model, the discipline option can no longer be modified.

Version: field available to store a model version number; no check is executed on this value.

Object Selection

This is the large white space on the left part of the window. It lists the meta-model objects available for the given discipline; any subset of this list can be actually applied in the model.

Address and States tab

Referring to the object type currently selected in the Object Selection list, this tab contains the definitions of the possible object states. The default state (usually the **quiet** state) is shown with a yellow background.

Address From / To: object address range (0 to 9999) according to the Modbus data structure. If only one address is used, enter the same value in both **From** and **To** fields.

ModBus Function: reading code in the Modbus function definition. This field defines the type of data to be acquired. The following relations apply: 01=Coils; 02=Discrete inputs; 03=Holding Registers; 04=Input Registers.

Note that the Address and Function fields are typically specified in Modbus product documentation in a conventional format based on a 5-digit structure including both a code (digit 1: 1 means Function 02, 2 means Function 01, 3 means Function 04, 4 means Function 03) and a 1 to 9999 address (digit 2-5, base-1 value, whereas the Address to enter in the tool is base-0 and therefore has to be decremented by 1).

The table below illustrates some examples.



Typical addressing scheme (10001-49999)	Corresponding Address (0-9999)	Corresponding Function (01-04)
20010	0009	01
10005	0004	02
40002	0001	03
30001	0000	04

Mandatory: check this field if the object must be present in all the configurations. This also means that all the defined instances of the object (from/to range) must be defined in the CSV import file.

Import Parent: enabling this option results in the parent/child relationship being handled in the CSV import procedure. Refer to section 2.4 at page 17.

State list: enter here the information about the available states: **Description**, **Subaddress1**, **Mask1** and **Value1**, (**Subaddress2**, **Mask2** and **Value2**).

Priority: If this flag is unmarked then the first state found as valid (true) is considered as the current object state. If this flag is marked, then in case more than one object state is valid, then the last state found as valid is considered as the current object state (see Annex C for a more detailed explanation).

DataShift, and applicable **Commands** (up to 4).

Subaddress is used in two ways:

- If **DataShift** is not set, it can be a mask (specified as a bit position from 1 to 16) for addressing a single bit in a 16-bit wide register (“-1” means no mask).
- If **DataShift** is set, it is an offset from the original address position.

Mask and **Value** allows identifying one of more bits in the addressed 16-bit word. This allows creating more data objects out of a single Modbus address. For any given mask, the corresponding state is set if the selected bits match the given value.

Single bit masks (e.g. 0x0001, assuming the data bit is in the least significant position) should be used for Modbus Functions that address individual bits (function 1 and 2).

Modbus Functions 3 and 4 handle an entire 16-bit register and any mask can be used. For example, two values can be extracted from a register using the masks 0xFF00 (higher byte) and 0x00FF (lower byte).

In unmasked registry values (mask = 0xFFFF), individual bits can also be identified at CVS file level (see Address2 field in section 2.4.1 at page 17).

If necessary, a second set of information (**Subaddress2**, **Mask2** and **Value2**) can be specified and the resulting value is *ANDed* with the first.

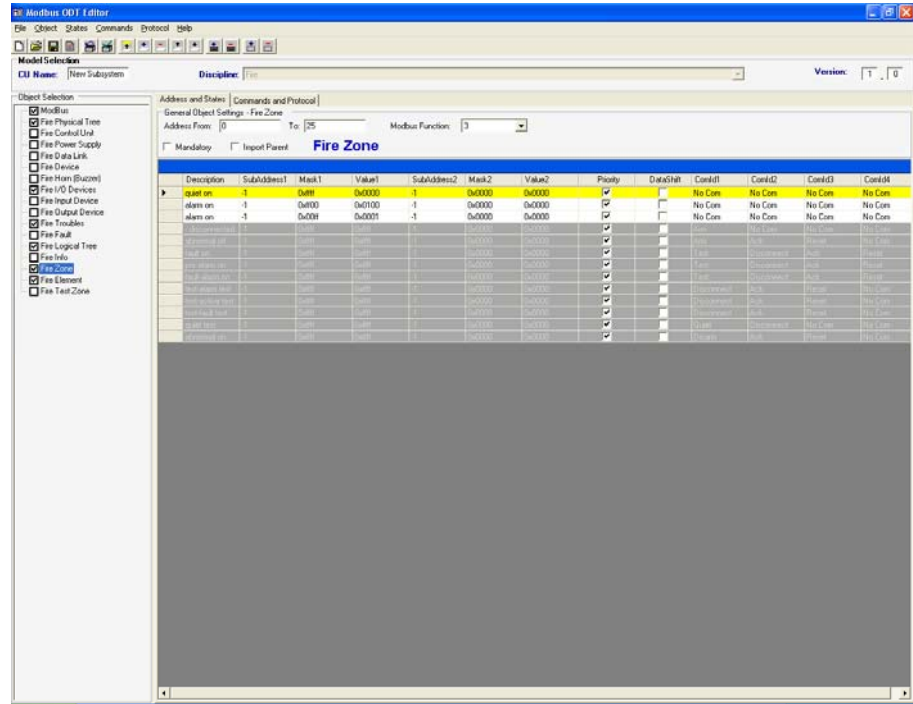


Fig. 8 States setting

Commands and Protocol tab

Commands configuration: enter here the information about the available commands: **MetaCom** Description, **MBFunction** (typically = 05 for coils and = 06 for Registers), **Address**, **Value**, **DataShift**.

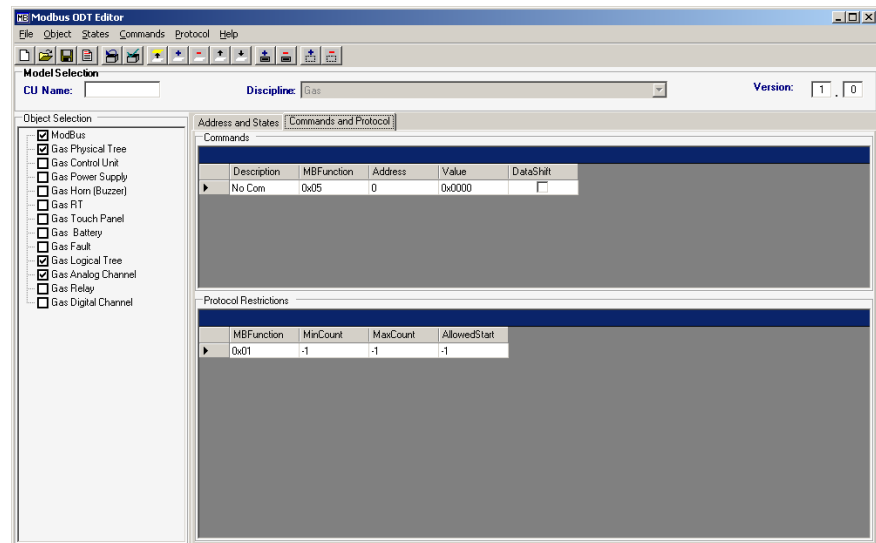


Fig. 9 Commands settings

Protocol Restrictions

MFunction, MinCount, MaxCount, AllowedStart.

These fields can restrict the range of the data polling for the specified Modbus function (MFunction), indicating a min/max extension (MinCount / MaxCount) and/or the starting address (AllowedStart).

2.3.1.3 Main menu and toolbar

The tool menu and the associated shortcut toolbar include the following main groups of items:

- File: creating/opening/closing model configurations
- Object: save/undo modifications
- States: adding/removing/... states of the selected object
- Commands: adding/removing object commands
- Protocol: adding/removing protocol addressing restrictions
- Help: showing about/help page

File menu

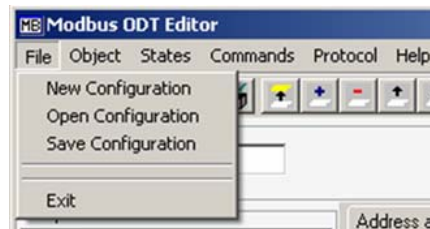





Fig. 10 File menu

New Configuration (): clear the configuration for defining a new model

A new model can be added by simply typing its name in the **CU name** field, filling-in the information and then saving it.

Open Configuration (): open one of the existing models, which are stored in .MUB files, located in the same folder as the tool program.

Save Configuration (): save the model into the .MUB file on disk

Exit: close the tool program

Object menu



Fig. 11 Object menu


Update: Save modifications

Restore: Undo modifications since latest Update


States menu





Fig. 12 States menu

Add (): Add a new state in the list

Remove (): Remove from the list the selected state

Up (): Move up the selected state (higher priority)


Down (): Move down the selected state (lower priority)


Set Default (): Set the currently select state as default state

Commands menu



Fig. 13 Command menu


Add (): Add a new command in the list


Remove (): Remove from the list the selected command

Protocol menu



Fig. 14 Protocol menu

Add restriction (): Add a new restriction in the addressing range

Remove restriction (): Remove the selected restriction

Help menu

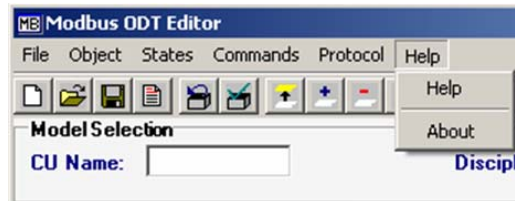


Fig. 15 Help menu

Help: show the help pages

About: show general information about the program

2.3.1.4 Configuration procedure

In order to configure a Modbus model, you should proceed as follows:

Model Preparation

1. Get the detailed Modbus point and command list, including the possible point states, as well as any required protocol restrictions.
2. Study the possible relationships between Modbus data and the objects available in one of the Poseidon meta-models (see Annex A at page 26).

Model Configuration

1. Open the Modbus Tool **MBUtility**.
2. Define a new model entering the name in the **Description** field.
3. Select the **Discipline**, thus setting the meta-model.
4. Enter the version number, e.g. **1.0**.
5. In the **Object Selection** list, select the applicable object types.
Note that the objects named **ModBus**, **Physical Tree** and **Logical Tree** cannot be associated to Modbus messages as they are only used for collecting other objects into a higher level structure: global, physical and logical level, respectively.
6. In the **Configuration** tab, select the Modbus address range⁽¹⁾ (**From/To**) and the reading code (ModBus Function) for the selected object.
7. Define Object states: add the required states according to the possible conditions of the object.
8. For each state, define the specific data set (one or more bits) in the Modbus messages using the fields: **SubAddress1**, **Mask1**, and **Value1**. If necessary, add another data set (**SubAddress2**, **Mask2**, and **Value2**) to be used in AND with the first set.
9. Repeat steps 5 to 8 for all the necessary object types.
10. In the **Control unit** tab, define Commands: enter the required commands indicating the corresponding Modbus Function (**MBFunction**) and related parameters (**Address**, **Value**, **DataShift**).
11. Save the model (menu **File**→**Save**).

When a new model is defined, new files (*.MUB and *.BIN) are available for handling the new Modbus unit in Composer and NK82xx, respectively.

¹⁾ Use the Modbus protocol address (base 0)

Model Installation

At this point, Composer is ready to operate with the new Modbus model, whereas the NK82xx devices require a download, as described in the section 2.4.



Be aware that any subsequent change in the Modbus model is taken into account by Composer in any further configuration step.

2.3.2 Control unit icon

A specific icon can be defined for representing the Modbus control unit in the Composer (and MM8000) tree. The configuration tool provides for a default icon, which is named after the Description name of the new Modbus model.

A customised file can replace the standard ICO file. We recommend the following format:

- ICO file; resolution 48x48: colour depth 32

2.4 CSV import file

The CSV file is meant for describing the data structure of a specific control unit, based on a configuration model prepared with the Modbus model customisation tool (see 2.3 at page 9). This file can be prepared using a manual procedure or an automated file processing functions and then quickly imported into the Composer project. Typically, a separate and specialised file is required for each Modbus control unit.

2.4.1 CSV file structure

General

The CSV format (Comma Separated Value) is a popular standard for exchanging table-oriented data (e.g. spreadsheet) between applications. CSV files can be easily imported and exported by Office application such as Excel™ and Access™.

The format is based on text files and includes one line per record. Each line contains the data fields separated by a “,” (comma) character.



Warning: depending on the regional setting of Windows, the separation character may differ from the standard (comma). In order to prevent problems in the Composer import, find out which character used in your PC and set it in Composer during configuration.

Record structure

As illustrated by the very first line of the file (column titles), the CSV record structure for the Modbus import file includes the following fields:

- Address (mandatory): numeric field indicating the Modbus data address.
- Address2 (optional): mask to address a reduced number of bits a 16-bit wide register. Note that this must be consistent with the masking defined in the Modbus configuration (see Mask field at page 11).
- Parent (mandatory for points with **Import Parent** set in the model): control unit ID of the parent point.
- Original (mandatory for Parent points): control unit ID of the parent point. This field is also copied in the technical text and can be a text string if the parent link is not used.
- Description (mandatory): free text field for customer name (max 64 characters).
- Remarks (optional): free text field for comments and notes.
- Function (mandatory): reading function (data type) according to Modbus data model; range 1-4: 1=Coils; 2=Discrete inputs; 3=Holding Registers; 4=Input Registers.

Example of CSV data

Address	Address2	Parent	Original	Description	Remarks	Function
14	0x00FF		201	Fire Zone 1		3
14	0xFF00		202	Fire Zone 2		3
15	1	201	203	5m fire		3
15	2	201	204	10m fire		3
15	3	201	205	50m fire		3
15	4	201	206	100m fire		3
17	1		0	Connection OK		3
18			0	Errors		3

2.5 Communication network

The Modbus is connected to the MM8000 system by means of the NK8000 network and namely via the NK82xx units.

In order to communicate with the Modbus, the NK82xx units should however be equipped with a new firmware that is included in the installation package as an additional component (DLL) to be added to the standard firmware file set.

The software installation procedure provides to copy the firmware files (a compressed ZIP archive) in the **NK82xx – Firmware** folder of the MM8000. From there, the files can be downloaded to the NK82xx units using standard Composer commands. The required procedure is described here below.

2.5.1 NK82xx firmware download

The following are the download procedures for the NK82xx firmware supporting the Modbus communication protocol.

Note: It is assumed that the NK82xx are physically installed, powered on, and communicating over the network. For more information about the NK8000 installation, please see the document no. A6V10062425, DMS8000 Network, Fire and Intrusion Connectivity Configuration Guide. More advanced technical issues are also discussed in the document no. A6V10062437, NK8000 Installation, Configuration and Commissioning.

Also, you should have available the Composer project that includes the NK8000 network and all the NK82xx units.

1. Verifying the connection with NK82xx

The NK82xx download requires that the TCP/IP connection between the host PC and the NK82xx is working properly. In the Windows Command Prompt window, you can check easily this connection using the **Ping** command:

```
ping n.n.n.n
```

with **n.n.n.n** being the IP address of the NK82xx unit, e.g. 168.123.8.76.

If the IP connection is good, the message text looks like the ones in Fig. 16, i.e.:

```
Reply from n.n.n.n: bytes=... time ... TTL=...
```

If the IP connection is not working for any reason, different messages may appear (Request timed out, Destination net unreachable, etc.) In these cases, verify the network settings and cabling and try again.

```
C:\>ping 192.168.8.76

Pinging 192.168.8.76 with 32 bytes of data:

Reply from 192.168.8.76: bytes=32 time<10ms TTL=60
Reply from 192.168.8.76: bytes=32 time<10ms TTL=60
Reply from 192.168.8.76: bytes=32 time<10ms TTL=60
Reply from 192.168.8.76: bytes=32 time<10ms TTL=60

Ping statistics for 192.168.8.76:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Fig. 16 Checking IP connection

2. Start Composer and open the project that includes the Modbus units.

3. Expand the **Channel collection** folder in:

Supervision System Settings → MM8000 System → Physical configuration → Station (or FEP) → Channel collection.

4. Select the **NS8210 driver** node and then the **Download** tab (Fig. 17).

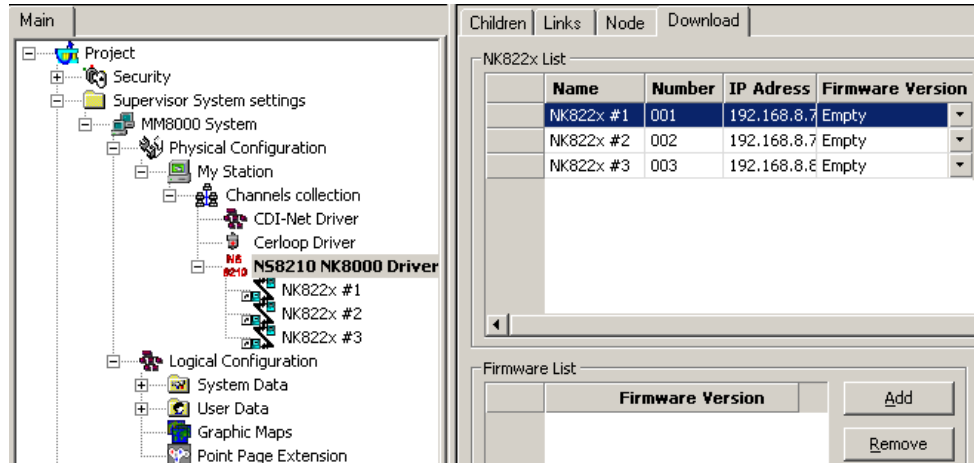


Fig. 17 Download tab

5. Select all the branches (NK82xx) in the list located in the upper part of the form.

Note: In order to select multiple branches, keep the CTRL key pressed while you make your selections.

See the following Fig. 18.

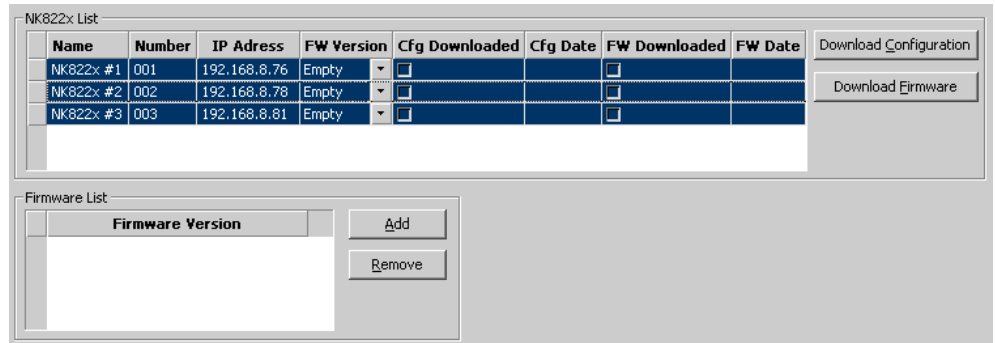


Fig. 18 Selecting the NK82xx

6. Add the new firmware version:

- Click **Add**, then browse and locate the additional firmware file in:
 <MM8000 installation folder>\NK82xx – Firmware
- Then, click the file **Nk82xx_ModBus__x.xx-xx_07.zip**
- And finally click **Open** (Fig. 19).

In a few moments, the new firmware shows in the Firmware List (Fig. 20).

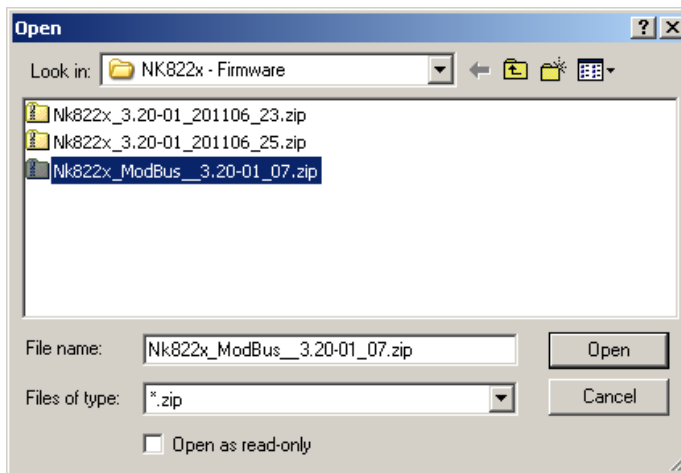


Fig. 19 Opening the new NK82xx firmware files

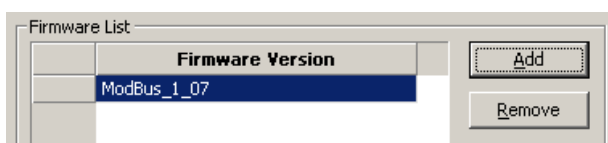


Fig. 20 New NK82xx firmware in the Firmware List



Note that the **_07** suffix in the firmware name indicates the add-on index (07 for the Modbus).

7. Select the new firmware version in the Firmware List.
8. Click the button **Download Firmware**.
The download procedure starts. The new firmware is downloaded to the NK82xx units via FTP (File Transfer Protocol) services over the network.
9. Ensure that you have successfully completed all downloads.
Verify that the **FW Downloaded** check boxes contain 'X's.
10. That completes the NK82xx firmware download.

Note that the NK82xx configuration will also need to be downloaded after having configured the Modbus units in Composer (see pag.24).

Downloading multiple firmware



In case multiple add-on packages have been installed, then all the associated firmware files can be safely downloaded in sequence (each of them being a single additional DLL) as long as the base NK82xx version is the same (e.g. ...4.10...). Please check the firmware version in the firmware list (Fig. 19) before selecting the file to download. Also, get informed about latest compatibility issues in the most recent NK8000 Release Notes.

3 Configuration

3.1 Configuration checklist

Verify that you have satisfied the items needed in the first checklist before proceeding to the configuration checklist that follows.

ITEMS NEEDED FOR CONFIGURATION

- The number and model (e.g.:) of Modbus-compatible units
- The local address (0 to 255) of each unit.
- The CSV file of each units
- The exact connection to the NK8000 unit (NK82xx).
- Plug-ins needed:
 - Plug-in #356401
 This is installed during the installation procedure.

CONFIGURING A Modbus SUBSYSTEM

- 1. Add the folder(s) required for identifying the location of the Modbus in the project structure tree. p. 22
- 2. Add the Modbus control unit node to the new folder p. 22
- 3. Set the Local Address p. 22
- 4. Import the CSV file p. 23
- 5. Link the Modbus to the communication network p. 23
- 6. Repeat steps above for all the Modbus units in the project
- 7. Download the configuration p. 24


3.2 Configuration procedure

The following are the configuration procedures for the Modbus control unit:

Adding the folder for the Modbus system

1. Open the Composer project.
2. Create a folder for the control unit.

Adding the Modbus node

1. Select the new folder.
2. Click the Modbus icon  in the icon bar or in the **Insert** menu (see Fig. 21) and select the type, i.e. one of the customised models currently available.

The new node is added to the project structure.

By default, the node will be named **Panel #1** You can customise the text by clicking once on the name, typing a new name, and pressing **Enter**.

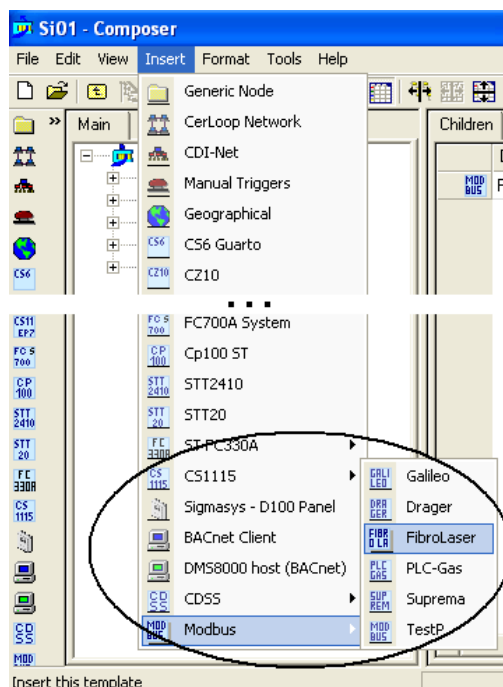


Fig. 21 Adding a Modbus subsystem

Setting the Modbus Local Address

1. Select the Modbus node.
2. Click the Node tab (see Fig. 22).
The Node form page shows.
3. In the Node tab form, you can find:
 - Description text: the node name you also have on the project structure tree.
 - Local Address: the Modbus address, which is fixed to 1.
 - Communication settings: parity bit, data bits, and stop bits (baud rate is instead set in the link node to the NK882x port)
4. Set the communication options.

Note: for detailed information about local address and communication settings on the tested Modbus units, refer to Annex B at page 27.

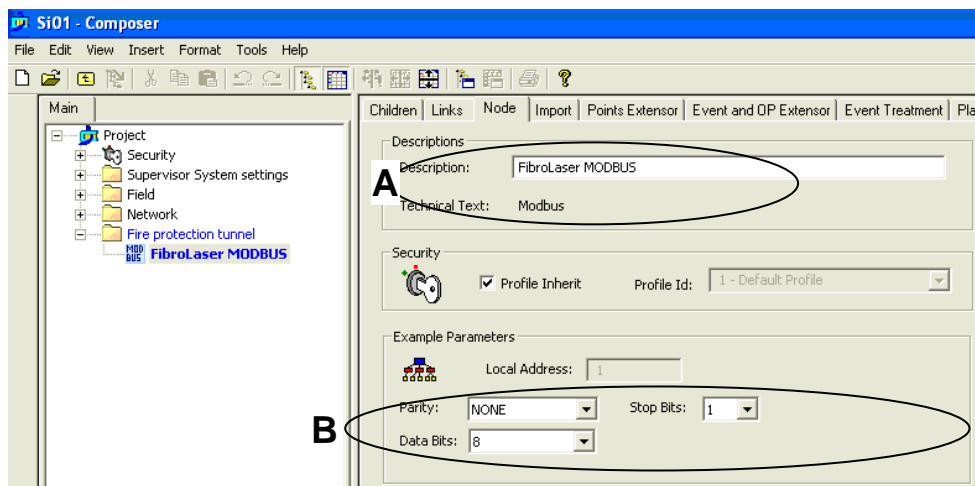


Fig. 22 Setting Modbus mode (subtype) and the Unit Address

- A Description
 B Communication setting: parity bit, data bits, stop bits.

Importing the CSV configuration file

For the tested Modbus units sample, CSV files can be found in the folder:
<Installation folder>\DMS8000\Composer\Configuration Data\OTD_Files

Note: <Installation folder> is usually **C:\Program Files**.

The sample CSV files have to be adapted according to the real configuration of the control unit. For Modbus units where no sample CSV files are available, the CSV file have to be created manually as described in section 2.4 at page 16.

The specific configuration of the Modbus subsystem is built up importing the CSV file (see section 2.4 at page 17). Proceed as follows:

1. Select the Modbus unit node.
2. Select **Tools**→**Import** in the Composer menu.

After a confirmation request, the software presents a browsing window to search for the files to import.

3. Using standard Windows file browser, do the following:
 - In the file system, locate the CSV file.
 - Select the file and click **Open**.

In a few moments, the unit structure is imported.





CSV files can be re-imported after a configuration change. Note the behaviour of the import procedure:

- New objects in CSV (not configured in Composer) are added to the configuration.
- Old objects in CSV (already configured in Composer) are updated.
- Removed objects in CSV (configured in Composer) are removed.

Linking the Modbus to the Communication network

1. Open NK82xx sub-folders
 Expand the NK8000 network folders until you reach the node that represents the NK882x COM port that is physically connected to the Modbus.
2. Select the Modbus node.
3. Drag and drop the Modbus node to the network COM port (see Fig. 23).

Note: Composer helps you recognise the valid link by displaying a shortcut link icon  instead of the no-link icon .

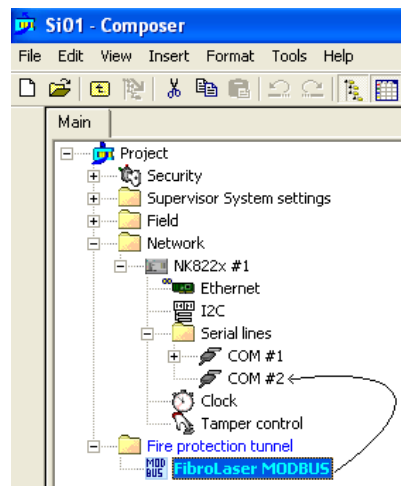


Fig. 23 Linking Modbus to the NK8000 network

When the link is established, a new node appears on the structure tree, and its properties can be seen on the Link tab of both the connected nodes.

Setting the communication baud rate

1. Select the new link node just below the NK82xx serial line node.
2. In the **Node** tab, set the communication baud rate, which must match the the corresponding setting in the control unit.

Note: for detailed information about communication settings on the tested Modbus units, refer to Annex B at page 27.

Download the MM8000 configuration

Before operating with the new MM8000 configuration, you need to download it. In Composer, the download command is available in the Tools menu. The preparation to the download is discussed in the document no. 006799, MM8000 Installation, Configuration and Commissioning.

Downloading the NK82xx configuration

After any modifications on the Modbus units, a new configuration download is required for the NK82xx devices.



Warning: the NK82xx units handle the Modbus messages interpretation for MM8000. In order to do so, the NK82xx needs to be downloaded with the updated configuration of the management stations, even after a minor modification to the subsystem structures (e.g.: importing an updated CSV file including new objects). Depending on the specific configuration change, failing to download the NK82xx units may affect the correct behaviour of the telegram interpretation and result in missing event signalling. In general, we recommend including an NK82xx download after any change in the configuration.

The download procedure can be started in two ways:

1. In the **Download** tab of the **NS8210 driver** node:
 - Select:
 - Supervision System Settings → MM8000 System → Physical configuration → Station (or FEP) → Channel collection → NS8210 driver**
 - see Fig. 17 above

- In the list that shows up, select the NK82xx units.
Note: In order to select multiple branches, keep the CTRL key pressed while you make your selections.
 - Click the button **Download Configuration**.
 - Ensure that you have successfully completed all downloads.
2. In the **NS82xx** node (select all units one after the other):
- Right click the node.
 - In the menu, click **Node commands** → **Download file CNF**.

3.3 Configuration backup and restore

The standard Backup and Restore functions in Composer provide for saving the Modbus customized models (refer to section 2.3.1 at page 10) along with all the project data.

Note that the entire folder **OTD_Files** is actually restored in
<installation folder>\Composer\Configuration Data\
and this also includes the model customisation tool (MBUtility.exe.exe).

Note: <Installation folder> is usually **C:\Program Files**

Annex A – Modbus meta-models in Poseidon

Supported disciplines

- Fire
- Gas
- Flooding
- Jail surveillance

Technical notes



The following notes are meant for engineers familiar with the Poseidon modelling environment and Composer add-on tools (Visual Builder).

The definition of meta-models designed to Modbus applications must include some specific configuration options that provide the required flexibility of use.

More specifically:

1. Set all Virtual Properties (or Common Properties, CP) as **not visible**.
In the Modbus customisation tool, the ones really needed can then be applied.
However, there is an important exception to this rule and it concerns the properties that must be shown when in quiet (normal) state. In this case, for example if you want to always show the **on-duty** (set) state of an object, then the visible attribute should be set.
2. In the Visual Builder tool, reset the Access attribute to **None** for all properties.
In the Modbus customisation tool, a specific access mode can then be applied case by case.

Annex B - Communication parameter settings

The table here below illustrates the recommended parameter settings for some tested integration solutions.

Modbus-compatible control unit	Baud rate	Bit settings (Parity, Data, Stop)	Local Address
Dräger Regard	4800 ⁽¹⁾	N ⁽¹⁾ , 8, 1	1
MSA Auer Suprema	9600 ⁽¹⁾	E, 8, 1	1
Sensitron Galileo	9600 ⁽¹⁾	N, 8, 1	1
FibroLaser II	19200 ⁽²⁾	N, 8, 1	1
Schneider Modicon PLC	19200 ⁽¹⁾	N, 8, 1 ⁽¹⁾	1
Siemens S700 PLC	19200 ⁽¹⁾	N, 8, 1 ⁽¹⁾	1

²⁾ Default value; configurable in the control unit.

Annex C – Priority Flag

The priority flag is used to indicate the normal mode of state calculation, namely:

1. The conversion algorithm calculates the states one after the other as written for the Object in the ODT.
2. Once a match is found (current value after mask equal state value in the ODT) the state is chosen for the object
3. Nevertheless, the algorithm continues to find other matches, if a new match is found the new match is selected as the object state.
4. The last state that was determined is the state that is being reported to the Meta Library
5. The priority name comes from the fact that each state has priority on the ones listed above him in the list (practically the line number is the priority rank)

When not indicating for a certain state, the priority flag it is used to indicate that the priority mechanism is not used on this state, i.e. if this state matches then the algorithm stops seeking for other possible state matches and report this state.

The normal way is to calculate using the priority flag. For example, we would put the states in the following order in the ODT:

Priority	MASK	VALUE	STATE
X	0x0002	2	Quiet
X	0x0004	4	Fault
X	0x0008	8	Alarm

So even it both the normal and the alarm bits are set we would receive alarm. This actually immunizes us for cases in which additional bits are set beside the most significant state.

Let's say that we have an element with 9 bits used to calculate the state, but the normal state is set when none of the bits is set. We can add in the top a line with mask 0xFFFF (or the mask of all the bits participating) and set the value to zero, no priority and the immediately when checking this line the algorithm can stop the calculation and save 9 or more calculation iterations:

Priority	MASK	VALUE	STATE
X	0xFFFF	0	Quiet
X	0x0004	4	Fault UnAck
X	0x0008	8	Fault Ack
X	0x3F40	0x3F20	State 3 Active

So usually we use the priority Flag, but in some cases, such as the PLC default state for object we use for the first state the non priority option.

Annex D – Compatibility specifications for Modbus units

MM8000 systems can integrate Modbus subsystems via NK82xx units. The following Modbus characteristics are supported:

- Modbus serial protocol (no Modbus TCP/IP) in RTU mode (no ASCII)
- NK82xx is the protocol master and connects one unit per line (no cluster)
- Serial line RS232 or RS485. Parameter settings:
 - Baud rate: 150-115,200
 - Data Bits: 7-8
 - Parity Bit: none, even, odd
 - Stop Bits: 1 or 2
- Modbus local address: 1 (not configurable)
- Modbus supported function codes:
 - Read: 01, 02, 03, 04
 - Write: 05, 06
- Modbus address range: 0-9999 (one address sub-range per object type)
- Address masks :
 - Mask for single bits of a 16-bit register
 - Mask for low and high byte of a 16-bit register
- Commands :
 - Address: Register address or Register address + Offset (data-shift)
 - Values: Fix values only (cannot include address offset)

Siemens Switzerland Ltd
Building Technologies Group
International Headquarters
Fire Safety & Security Products
Gubelstrasse 22
CH-6301 Zug
Tel +41 41 724 24 24
Fax +41 41 724 35 22
www.sbt.siemens.com