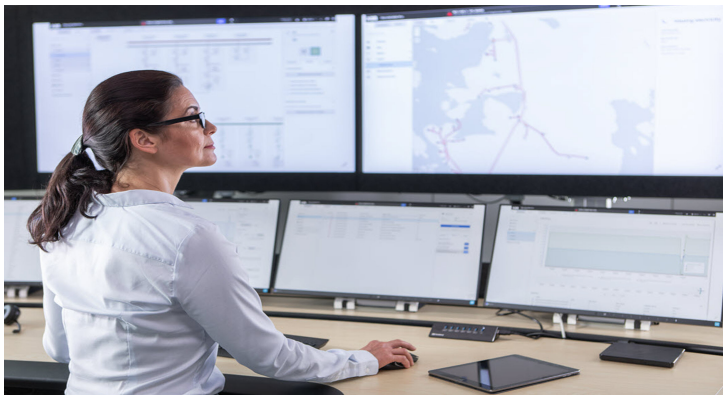
GRID AUTOMATION PRODUCTS

MicroSCADA X SYS600 10.2

Communication Gateway COM500i



Document ID: 1MRK 511 468-UEN
Issued: March 2021
Revision: A
Product version: 10.2

Table of contents

Section 1 Copyrights

The information in this document is subject to change without notice and should not be construed as a commitment by Hitachi Power Grids. Hitachi Power Grids assumes no responsibility for any errors that may appear in this document.

In no event shall Hitachi Power Grids be liable for direct, indirect, special, incidental or consequential damages of any nature or kind arising from the use of this document, nor shall Hitachi Power Grids be liable for incidental or consequential damages arising from the use of any software or hardware described in this document.

This document and parts thereof must not be reproduced or copied without written permission from Hitachi Power Grids, and the contents thereof must not be imparted to a third party nor used for any unauthorized purpose.

The software or hardware described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license.

Trademarks

ABB is a registered trademark of ABB Asea Brown Boveri Ltd. Manufactured by/for a Hitachi Power Grids company. All other brand or product names mentioned in this document may be trademarks or registered trademarks of their respective holders.

Guarantee

Please inquire about the terms of guarantee from your nearest Hitachi Power Grids representative.

Third Party Copyright Notices

List of Third Party Copyright notices are documented in "3rd party licenses.txt" and other locations mentioned in the file in SYS600 and DMS600 installation packages.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (https://www.openssl.org/). This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Section 2 Introduction

2.1 This manual

This manual provides thorough information on how to use the COM500*i* functionality to make a SYS600-based communication gateway. This manual describes how to install, configure and engineer COM500*i*. Detailed descriptions are given to provide the user with a deeper understanding of the functionality of COM500*i*.

This user's guide is divided into the following sections:

- Overview
  Provides general information about COM500*i*, its components and its main functions.
- Safety information
  Provides information about the prevention of hazards and how to create backups from the system.
- Instructions
  Provides an overview of COM500*i*. It also contains instructions for installation, configuration, engineering and upgrading.
- Technical description
  Contains descriptions of the functionality, design and configuration of COM500*i*. Detailed description is given to help the user understand how COM500*i* functions.

2.2 Use of symbols

This publication includes warning, caution and information symbols to point out safety-related or other important information. It also includes tips to point out useful hints to the user. The corresponding symbols should be interpreted as follows:

Warning icon indicates the presence of a hazard which could result in personal injury.

Caution icon indicates important information or a warning related to the concept discussed in the text. It might indicate the presence of a hazard, which could result in corruption of software or damage to equipment/property.

Information icon alerts the reader to relevant factors and conditions.

Tip icon indicates advice on, for example, how to design a project or how to use a certain function.

Although warning hazards are related to personal injury and caution hazards are associated with equipment or property damage, it is necessary to understand that under certain operational conditions, operation of damaged equipment may result in degraded process performance leading to personal injury or death. It is important that the user fully complies with all the warnings and cautionary notices.

2.3 Intended audience

This manual is intended for installation personnel, administrators and skilled operators to support installation of the software.

2.4 Related documents

The following is a list of documentation related to COM500*i*.

Name of the manual	Document ID
SYS600 10.2 System Configuration	1MRK 511 481-UEN
SYS600 10.2 System Objects	1MRK 511 482-UEN
SYS600 10.2 Application Objects	1MRK 511 467-UEN
SYS600 10.2 Communication Programming Interface (CPI)	1MRK 511 469-UEN
LIB 500 Operation Manual	1MRS755359
SYS600 10.2 DNP 3.0 Slave Protocol	1MRK 511 487-UEN
SYS600 10.2 DNP 3.0 Master Protocol	1MRK 511 486-UEN
SYS600 10.2 IEC 60870-5-101 Slave Protocol	1MRK 511 490-UEN
SYS600 10.2 IEC 60870-5-101 Master Protocol	1MRK 511 489-UEN
SYS600 10.2 IEC 60870-5-104 Slave Protocol	1MRK 511 493-UEN
SYS600 10.2 IEC 60870-5-104 Master Protocol	1MRK 511 492-UEN
SYS600 10.2 IEC 61850 Master Protocol (OPC)	1MRK 511 495-UEN
SYS600 10.2 Modbus Master Protocol	1MRK 511 497-UEN
SYS600 10.2 Modbus Slave Protocol	1MRK 511 498-UEN
SYS600 10.2 IEC 60870-5-103 Master Protocol	1MRK 511 491-UEN
SYS600 10.2 External OPC Data Access Client	1MRK 511 471-UEN
SYS600 10.2 OPC Server	1MRK 511 476-UEN
SYS600 10.2 IEC 61850 Server	1MRK 511 509-UEN
SYS600 10.2 IEC 60870-6 (ICCP) Protocol	1MRK 511 508-UEN

2.5 Document conventions

The following conventions are used for the presentation of material:

- The words in names of screen elements (for example, the title in the title bar of a dialog, the label for a field of a dialog box) are initially capitalized.
- Capital letters are used for file names.
- Capital letters are used for the name of a keyboard key if it is labeled on the keyboard. For example, press the CTRL key. Although the Enter and Shift keys are not labeled they are written in capital letters, e.g. press ENTER.
- Lowercase letters are used for the name of a keyboard key that is not labeled on the keyboard. For example, the space bar, comma key and so on.
- Press CTRL+C indicates that the user must hold down the CTRL key while pressing the C key (in this case, to copy a selected object).
- Press ALT E C indicates that the user presses and releases each key in sequence (in this case, to copy a selected object).
- The names of push and toggle buttons are boldfaced. For example, click **OK**.
- The names of menus and menu items are boldfaced. For example, the **File** menu.

    - The following convention is used for menu operations: **Menu Name/Menu Item/ Cascaded Menu Item**. For example: select **File/Open/New Project**.
    - The **Start** menu name always refers to the **Start** menu on the Windows Task Bar.
- System prompts/messages and user responses/input are shown in the Courier font. For example, if the user enters a value that is out of range, the following message is displayed: `Entered value is not valid`.
  The user may be told to enter the string MIF349 in a field. The string is shown as follows in the procedure: `MIF349`
- Variables are shown using lowercase letters: sequence name

2.6 Document revisions

Revision	Version number	Date	History
A	10.2	31.03.2021	New document for SYS600 10.2

Section 3 Overview

3.1 About this section

This section provides general information about COM500*i*, its components and its main functions.

3.2 COM500*i* as a communication gateway

COM500*i* is a communication gateway functionality included in SYS600. COM500*i* is the successor of COM500. The purpose of COM500*i* is to provide a gateway between process devices and up to eight upper level systems (NCC). The main tasks of COM500*i* are signal re-routing and protocol conversions. The use of COM500*i* as a gateway in a substation is illustrated in Figure 1.

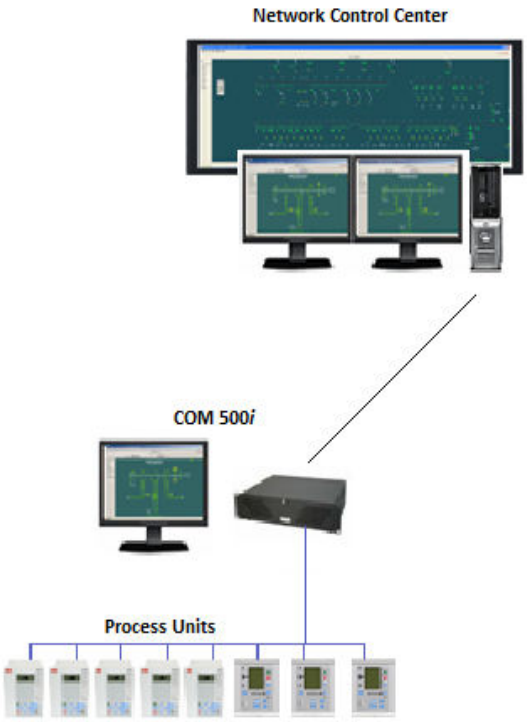


Figure 1: COM500i as a communication gateway

COM500*i* provides a variety of protocols for both process and upper level communication.

3.3 COM500*i* engineering process

The engineering process of COM500*i* contains the following steps:

1. Software installation
2. System configuration
3. Signal generation
4. Signal engineering

For more information on the engineering steps, see Section 4.

3.3.1 Signal engineering process

The signal engineering process contains the following configuration steps. Configuration is done by using Signal X-References tool, ICCP Configurator or using IET600 and IET Data Loader.

Signal X-References tool

Supported protocols: IEC 60870-5-101, IEC 60870-5-104, DNP 3.0, Modbus, CPI and RP570 protocols.

IEC 61850 protocol is configured using IET600 and ICCP protocol using ICCP Configurator. NCCs tab and Parameters tab can be used for IEC61850 and ICCP protocols in Signal X-References tool.

1. Add the NCCs to the Signal X-References tool and define the information related to it, such as the protocol to be used, and define the alarm groups to be used.
2. Check that all the indications and commands (input and output process objects) that are needed are shown in the tool. If they are not, add them.
3. Check that all the necessary attributes for indications and commands are correctly shown in the tool. If they are not, add the missing attributes or change their definitions.
4. Define to which NCCs COM500*i* should send the indications. Connect the indications to alarm groups, if needed.
5. Configure the commands to be received from the NCCs.
6. Adjust the system and application parameters.

For more information about the signal engineering process and Signal X-References tool, see Section 5.5.

IET600 and IET Data Loader

IET Data Loader and IET600 supports IEC 60870-5-101, IEC 60870-5-104, DNP 3.0 and IEC 61850 NCC protocols. See configuration steps from SYS600 IEC 61850 System Design documentation.

ICCP Configurator

ICCP Configurator is configuration tool for making ICCP configuration for client and server. When ICCP operates as server, it operates together with COM500i application. ICCP Configurator is started from SYS600 Tool Launcher, Application Configuration tab. See configurations steps from IEC 60870-6 (ICCP) Protocol documentation.

Section 4 Safety information

This section has information on the prevention of hazards and taking backups from the system.

4.1 Backup copies

4.1.1 Taking backup copies

We recommend taking backup copies before making any changes, especially ones that might have side effects. Software and data need to be copied to another place.

Backup copying makes it easier to restore the application software in case of disk crash or other severe failure where stored data is lost. It is therefore recommended that backup copies are taken regularly.

There should be at least two system backup copies and two application copies. A new backup is copied over the oldest backup. This way the latest version is always available, even if the backup procedure fails.

Detailed information on how to take backup copies should be delivered to the customer with the application.

4.1.2 System backup

Usually a system back up is taken after the application is made. It should be taken again when changes are made to the SYS600 system. This is required when the driver configuration or the network setup is changed.

4.1.3 Application backup

An application backup is also taken at the same time with the system backup, after the application is made. It should be taken again when changes are made to the application, for example, if pictures or databases are edited or new pictures are added.

4.2 Fatal errors

A fatal error is an error that causes a breakdown or a locked situation in the SYS600 program execution.

4.2.1 Handling

In case of a fatal error:

1. Write down the possible SYS600 error messages.
2. Shut down the SYS600 main program. If this cannot be done in the SYS600 Control Panel, try to end the task in Windows Task Manager.

Files may be damaged if the base system computers are shut down by switching the power off.

3. The data kept in the main memory at the moment of a fatal error is placed in the drwtsn32.log file with Windows 2003 Server, Windows XP and earlier. By default it is placed under %SYSTEMDRIVE%\Documents And Settings\All Users\Application Data \Microsoft\Dr Watson. Log and dump file paths can be checked with the drwtsn32 application. (Start -> run -> drwtsn32.exe). Analyze and copy the data in these files. Starting with Windows Server 2008 and Windows 7 the crash handling has changed. The location of the dump files can be read from the registry under the key HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting \LocalDumps. The DumpFolder value tells the location of the dump files. Collect the data from this location.
4. Restart the system.

Report the program break-down together with the possible SYS600 error messages and the information from the drwtsn32.log file to the SYS600 supplier.

4.2.2 Status codes

Error messages in SCIL are called status codes. A list of status codes and short explanations for them can be found in SYS600 Status Codes.

Section 5 Instructions

This section gives an overview of COM500*i*. It also gives instructions for installation, configuration, engineering, and upgrading the software.

5.1 Overview of COM500*i*

COM500*i* is a communication server which provides communication gateway functions for mapping signals between process devices and up to eight upper level systems, depending on the license. The upper level system is called the Network Control Center (NCC). COM500*i* sends information to the system for supervising and controlling the customer's processes and receives process control commands. COM500*i* also handles system co-ordination tasks, such as dynamic assignments of control command authorities and communication supervision.

COM500*i* supports a variety of protocols for connecting the process devices to upper level systems. It is typically connected to the network by some telecontrol protocols. For example, an NCC using the IEC 60870-5-104 protocol can be connected to IEC 61850, LON, SPA or IEC 60870-5-103 devices via the COM500*i* gateway.

5.1.1 Function

The main task of COM500*i* is to handle data transfer between process devices and up to eight network control centers. Data transfer usually involves protocol conversion. Other tasks such as communication supervision and command authority checking are also involved.

Both the process devices and the network control system may be products of Hitachi Power Grids or a third party. Certain documents, such as interoperability lists and device profiles, can be used for verifying the compatibility between COM500*i* and other systems.

5.1.2 Example system

COM500*i* can be used as a stand-alone gateway as well as combined with any other SYS600 product option, in order to, for example, add Single Line Diagrams, Event Lists, IED Tools or an archiving functionality; see Figure.

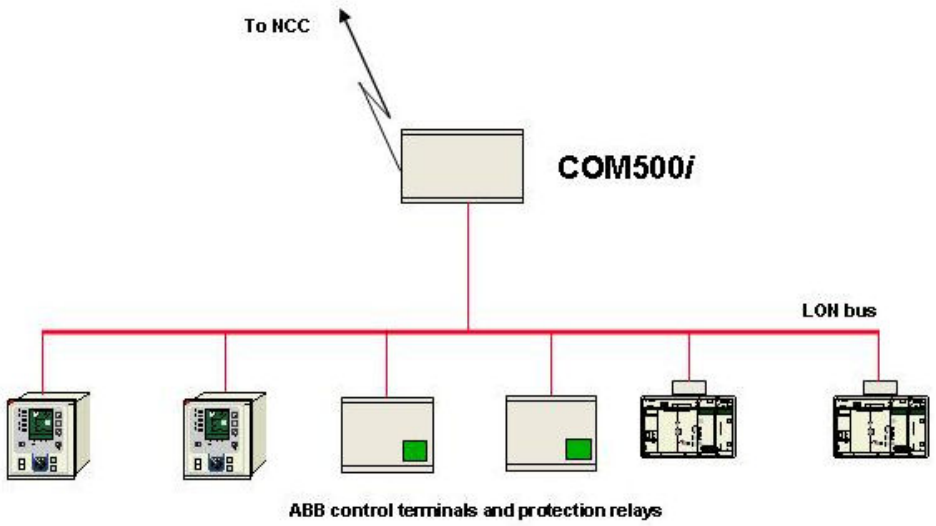


Figure 2: Typical stand-alone COM500i system

5.1.3 COM500*i* application

The use of a COM500*i* application depends on the type of the system. In the case of a stand-alone gateway, the application is used merely for signal rerouting. In the case of a combined SYS-COM, the application is also used for process control and management, and it acts both as a communication gateway and as a Substation Controlling System (SCS).

The above is based on the layered structure of the SYS600 software, presented in Figure 3. The application layer can have different functions, which are independent of the lower layers.

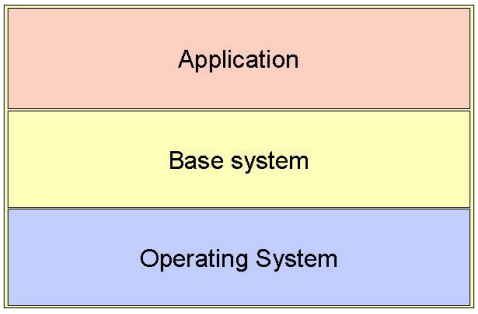


Figure 3: Software layers in COM500i

From the communication point of view, the COM500*i* application sees each process unit and NCC as a system object. Setting the attributes of the system objects can change the properties of the communication channels.

Communication between COM500*i* and an NCC is based on the command procedures implemented in the SCIL programming language. These procedures send information from the COM500*i* application to the communication module for protocol conversion and transmission. Usually, one command procedure is needed for each type of data. Data can also be transferred based on the application and system commands, such as general interrogation commands, and in special situations, such as at the application start-up and after a communication disturbance. When the system is running, process events generally activate the command procedures via event channels.

Commands and setpoints from an NCC to a substation are brought into the COM500*i* application via process objects. They activate the command procedures via event channels.

Command procedures send the actual control commands to the process units. System and application commands are also received using the process objects. The content of the COM500*i* command procedures is described in Section 6.

5.1.4 Functional environment

The functional environment of a COM500*i* application can be described by using the environment model shown in Figure 4. The COM500*i* application communicates through the base system and communication module. In addition to NCCs and the COM500*i* application, the base system can also communicate with an SCS application at the same time.

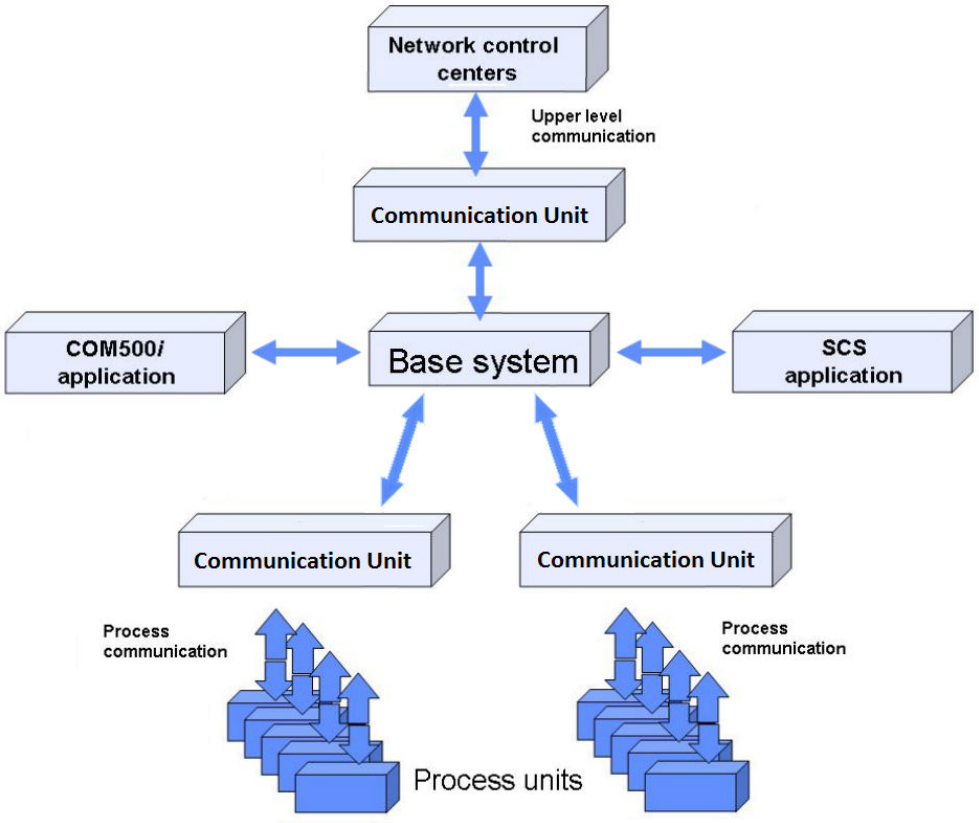


Figure 4: Environment model of COM500i

5.1.5 Communication Programming Interface

COM500*i* provides support for the Communication Programming Interface (CPI), which is an environment for protocol development that can be used for implementing new protocols in SYS600. CPI is a collection of functions programmed in the C language for making communication software that converts between the SYS600 internal protocol and other protocols. CPI is available on request.

CPI-based communication software can be used for process communication and for upper level communication. The CPI library contains functions for sending and receiving messages to/from COM500*i*. It also contains functions to pack and unpack data. The CPI-based communication software and COM500*i* communicate through the TCP/IP network. The program that uses the CPI interface in COM500*i* must emulate an RTU profile, which has process objects of the RTU-200 process database interface type. CPI is described in more detail in Section 6.

5.2 Installation

5.2.1 Upgrading from earlier revisions

The following notes should be observed when upgrading an existing COM500 application to COM500*i* revision 10:

COM500*i* revision 10 has a mechanism that makes all the required modifications to the application to update an older revision to revision 10. This mechanism is started when a monitor is opened to a COM500*i* application for the first time after the installation of COM500*i*. After the mechanism has been run, SYS600 must be restarted in order to take all the modifications in use. The modifications are described in detail in Section 5.2.3.

If any project specific modifications have been made to the command procedures of the previous COM500 revision, the modifications must be copied to the matching new command procedures. The signal configuration, that is the contents of the cross-reference tables, does not need to be changed. For more information about the content of the command procedures, see Section 6.

For more information about the changes required in the configuration files, see Section 5.3.

5.2.2 Software installation

5.2.2.1 Installation procedure

COM500 is included in SYS600 9.2 or newer as a license-dependent functionality, referred to as COM500*i*. COM500*i* is a communication gateway between process devices and network control centers (NCCs). The main tasks of COM500*i* are signal rerouting and protocol conversions.

To use the COM500*i* functions, select the **COM500i** tab from Tool Manager (see Figure 5) or select **COM500i** tab from Tool Launcher (see Figure 6).

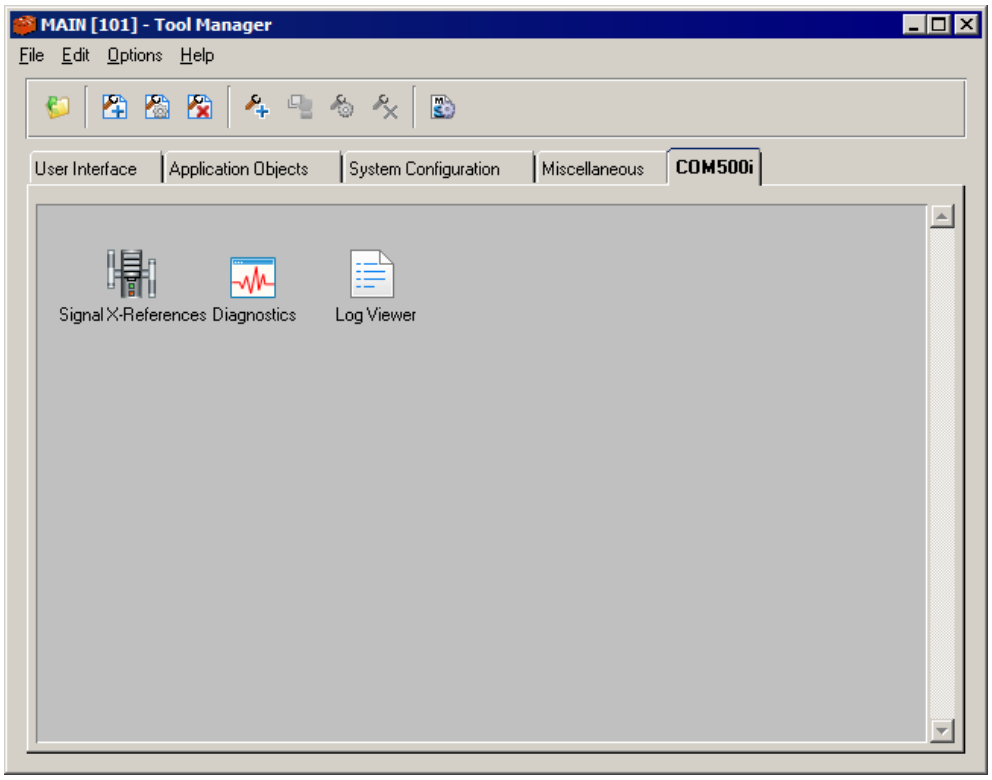


Figure 5: COM500i tab in Tool Manager

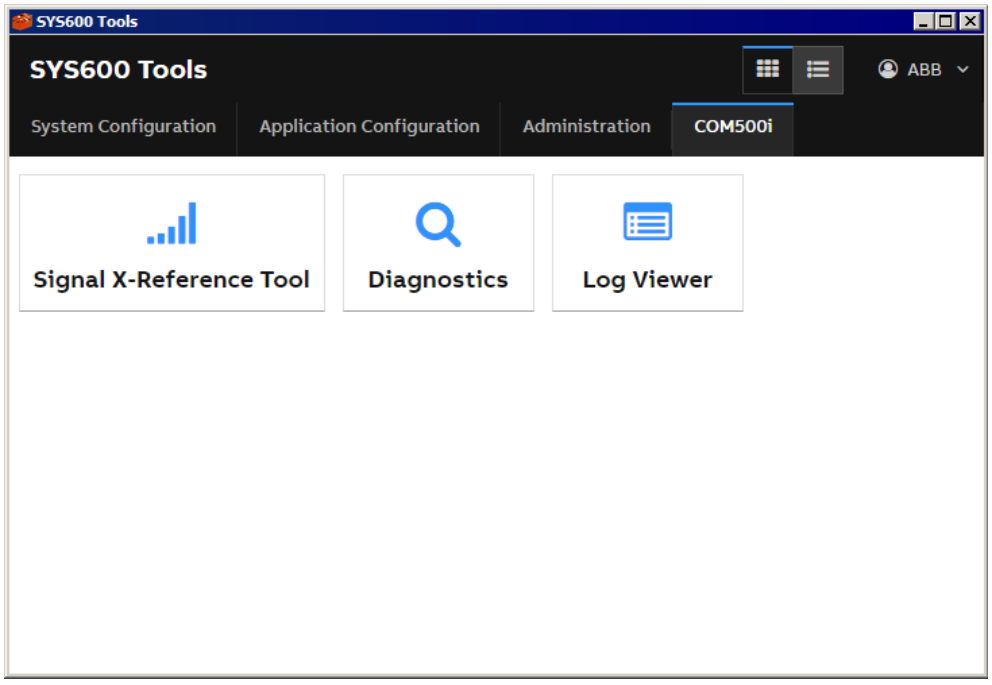


Figure 6: COM500i tab in Tool Launcher

5.2.3 COM500*i* start-up

5.2.3.1 Actions at start-up

When the COM500*i* license has been installed and SYS600 restarts, an application can be prepared for COM500*i* in Signal X-References tool. Signal X-References tool can be opened from Tool Manager or Tool Launcher (see Figure 7).

Figure 7: Preparing a COM500i application

If a new application is created with the same name than the old application. The following two lines from sc\com\active\com_\com_updt.txt file must be removed before preparation of the application. The lines do not need to be removed, when COM500i is updated from old version.

```
;## Executed by application 'apl_name'
;## at 20-11-30 13:00:00, This is not executed again while these lines
are here
```

5.2.3.2 Application preparation from Tool Manager

Tool Manager must be restarted after preparation (see Figure 8). After Tool Manager restart, COM500*i* automatically creates all the necessary application objects, such as event and time channels, and command procedures. COM500*i* also creates the directory \sc\apl\<name> \com500, which is used for storing cross-reference and parameter files automatically.

Figure 8: Requesting Tool Manager restart

5.2.3.3 Application preparation from Tool Launcher

Signal X-References must be restarted after preparation (see Figure 9). After Signal X-References tool restart, COM500*i* automatically creates all the necessary application objects, such as event and time channels, and command procedures. COM500*i* also creates the directory \sc\apl\<name>\com500, which is used for storing cross-reference and parameter files automatically.

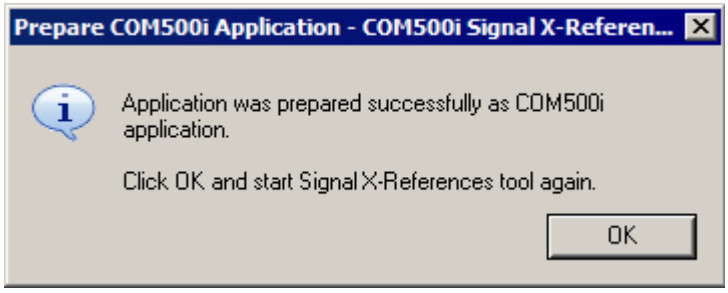


Figure 9: Requesting Signal X-References tool restart

5.3 System configuration

5.3.1 Configuration files

In COM500*i*, the base system is configured in the SYS_BASCON.COM file.

5.3.2 Base system objects

Each base system has a set of objects that specify the base system and its environment, hardware and software and the physical and logical connections of the base system and its applications.

Base system objects are defined with SCIL commands in the SYS_BASCON.COM file, which is executed each time the base system is started. With a few limitations, the base system objects can also be defined and modified at any time when COM500*i* is running. During the operation, the base system objects are in the primary memory of the base system computer.

5.3.3 Communication system objects

Each communication module contains a set of system objects, which specify communication line properties, connected devices and so on. These objects can be created, modified and deleted by SCIL, and setting the attributes of the objects can change the properties. Each communication line and each station created on a line is represented by one object.

In the case of PC-NET, the process communication system objects can be defined by using the System Configuration Tool or by using SCIL statements.

5.3.4 Base system configuration

5.3.4.1 COM500*i* base system

Basic configuration of the base system, as in the base system itself, nodes, links, and SYS600 monitors, is defined in the SYS_BASCON.COM file. For more information about the base system configuration, see SYS600 System Configuration.

5.3.4.2 SYS_BASCON.com file

COM500*i* application must be introduced into the SYS_BASCON.COM file:

```
#local COM500 = vector(TRUE) ;TRUE = COM500i application
```

5.3.5 Communication system configuration

5.3.5.1 Protocols

COM500*i* supports several protocols. IEC 61850, SPA, LON and IEC 60870-5-103 masters can be used for process communication, and IEC 60870-5-104 slave, IEC 60870-5-101 slave, DNP 3.0 slave, RP-570 slave, Modbus RTU slave, IEC 61850 Server and ICCP Server for upper level communication with Network Control Centers (NCC). For more information on the protocols, see Section 6.

The following sections provide information on the configuration of the NCC Communication using the above mentioned slave protocols.

When the upper level communication is configured, it is very important to match the parameters (attributes) of the communication line and station to the parameters of the upper level system. The attributes of the communication line and station system objects are described in detail in the configuration manual of the protocol or in the case of the RP-570 slave protocol, in the SYS600 System Objects documentation. For more information about communication system configuration required for a CPI application, see SYS600 Communication Programming Interface (CPI).

The communication system configuration required for the process communication (master protocols) is described in SYS600 System Configuration, SYS600 System Objects and in the protocol specific documentation.

5.3.5.2 PC-NET

PC-NET can be configured either by using System Configuration Tool or SCIL. Using System Configuration Tool is preferred when configuring COM500*i*. Detailed information about using System Configuration Tool is provided in SYS600 System Configuration.

If SCIL statements need to be used for configuration, COM500*i* provides standard command procedures for creating communication lines and stations for the RP 570 slave, IEC 60870-5-101 slave, and DNP 3.0 protocols. The command procedure COM_RPSCR contains communication line and station definitions for the RP 570 slave protocol. It must be executed every time PC-NET is restarted, for example, from the predefined command procedure APL_INIT_1:C.

As input the command procedure COM_RPSCR needs the NET number, line number, vector of stations, and message application number.

The following command defines the RP 570 slave protocol to line 1 of NET 1 and adds a master station with number 8 into it:

```
#EXEC COM_RPSCR:C (@NET=1, @LINE=1, @STATIONS=VECTOR(8), @APPLIC=1)
```

Command procedure COM_101SCR defines the lines and stations for the IEC 60870-5-101 slave protocol. It must be executed every time PC-NET is started. As input the command procedure needs the NET number, line number, communication mode (0 = balanced, 1 = unbalanced), vector of stations, and message application number.

The following command defines a balanced IEC 60870-5-101 slave protocol to line 2 of NET 1 and adds a master station with number 9 into it:

```
#EXEC COM_101SCR:C (@NET=1, @LINE=2, @MODE=0,
@STATIONS=VECTOR(9),@APPLIC=1)
```

The following command defines an IEC 60870-5-104 slave protocol to line 2 of NET 1 and adds a master station with number 9 into it:

```
#EXEC COM_104SCR:C (@NET=1, @LINE=2, @STATIONS=VECTOR(9),@APPLIC=1)
```

For a DNP 3.0 slave protocol the lines and the station can be created by using the command procedure COM_DNPSCR, which must be executed every time PC-NET is started. As input this command procedure needs the NET number, line number, vector of stations, and message application number.

The following command defines a DNP 3.0 slave protocol to line 3 of NET 1 and adds a master station with number 10 into it:

```
#EXEC COM_DNPSCR:C (@NET=1, @LINE=2, @MODE=0,-
@STATIONS=VECTOR(10),@APPLIC=1)
```

5.3.5.3 Modbus RTU Slave

Modbus RTU Slave Emulator is a gateway program which provides a connection between SYS600 and an NCC using Modbus RTU Master protocol or Modbus over TCP/IP protocol. From SYS600's point of view, Modbus Slave is seen as a NET node object and the STA objects connected to the Modbus slave node operates as a data storage for the data values which are requested by the Modbus master. This STA objects also receive control commands from an NCC and activate the handling of the commands in COM500*i* correspondingly.

In the COM500*i* start-up, the cross-referenced data points to the mentioned data storage are initialized and after this, communication is enabled using the DI attribute of the station objects. In Modbus RTU, only the static values are transferred, i.e. the transfer of time-stamped data is not supported.

When the Modbus Slave application is started, it reads the configuration file (config.ini) and establishes a connection to the base system. One SYS600 system can have multiple Modbus Slave instances running at the same time. For more information, see SYS600 Modbus Slave Protocol and the SYS600 System Configuration documentation.

5.3.5.4 IEC 61850 Server

IET600 is used to create the configuration for COM500*i* NCC type IEC 61850. This configuration is imported using IET Data Loader, which creates the configuration files needed by IEC 61850 server instances and also the process objects and configuration for the process devices if they are using IEC61850 protocol. IEC 61850 Server is a separate executable which connects itself to the LAN Link of base system using CPI Interface diagnostics and security extensions. One instance of IEC 61850 Server is started for each NCC definition in COM500i. STA object type and data base type is IEC (STY 29). For more information, see SYS600 IEC 61850 Server documentation.

5.3.5.5 ICCP Server

ICCP Configurator is used to create configuration for COM500*i* NCC type ICCP. ICCP Configurator creates cross-references data and the configuration files needed by ICCP server instances. ICCP NCC is created using Signal X-References tool. ICCP Server is a separate executable which connects itself to the LAN Link of base system using CPI Interface. Currently the instance count in one system is limited to 1. This means that only one ICCP NCC can be created in Signal X-References tool. STA object type and data base type is IEC (STY 29). For more information, see IEC 60870-6 (ICCP) Protocol documentation.

5.3.6 Configuration tips

5.3.6.1 IEC 60870-5-101/104 protocols

When configuring the IEC 60870-5-101 slave lines and stations it is very important to match the message field length attributes IL, CL, PL and SL (see SYS600 System Configuration for details) to the corresponding parameters of the master system. A mismatch in these attributes can lead to a situation where communication appears to be running correctly, but the messages are incorrectly interpreted or not set to the process database at all. In IEC 60870-5-104 protocol, the message field lengths are fixed by the standard and non-standard value should not be used.

Application and system commands, as well as unrecognized messages sent from the NCCs, are received in process objects and interpreted by the COM500*i* command procedures. To ensure that these commands are received and executed correctly, the following things should be checked:

- The MI attribute of each slave station should have its default value (29000 + station number).
- The CA attribute of each IEC 60870-5-101/104 slave station should have its default value (32000).
- The value of the PC (Process Data Confirmation) attribute of each IEC slave station should be set to 0 in order to ensure proper confirmation and termination of the IEC 60870-5-101/104 commands. System and application commands are not confirmed automatically, when the attribute is set to the automatic test mode.
- If COM500*i* is not synchronized from the NCC, the value of the RM (Running Mode) attribute of the IEC slave station should be so that bit 1 of the value is set (RM = 2, if no other bits are set). Otherwise data is sent with an invalid time tag status to the master.
- By default, COM500*i* waits for a secured command from the master. The select command must come before the execute command. In order to use direct commands, the station attribute RM bit 4 must be set. Otherwise a negative acknowledgement is sent to the master.
- In IEC60870-5-104, time-stamped commands are used by default. In order to accepts commands of this type, the COM500*i* must be synchronized to use the same time as the master. See station attributes CD and CC for more information.

5.3.6.2 DNP 3.0 protocol

Some requests and unknown messages sent from the NCC are received in process objects and interpreted by the COM500*i* command procedures. To ensure that these messages are received and interpreted correctly, the following things should be checked:

- The MI attribute of each slave station should have its default value (30 000 + station number).
- The CA attribute of each DNP 3.0 slave station should have its default value (32 000).
- By default, COM500*i* sends data with dynamic variations. If the value changes in indication but not in status, a variation without a status is sent. If the status changes, the status is also sent. If the station attribute RM bit 1 is set, the variations are fixed to same as in the master's request.

When data is sent as a double binary signal, two binary data objects with consecutive addresses are used for the two bits of the value. An alternative way to send double indications is to use a double bit input object. This object type has been added to the standard later and is not necessarily supported by all masters. Comparing device profile documents may be necessary.

5.4 Signal generation

The signals, i.e. process objects, for COM500*i*, have to be generated.

This can be done by:

- Building a COM500*i* application
- Importing signals
- Using IET Data Loader

5.4.1 Building a COM500*i* application

Building a COM500*i* application (creating the process database) is similar to building a SYS600 application: the actual gateway functionality is provided by COM500*i* and the definitions are made with Signal X-References in the signal engineering phase.

A COM500*i* application can be built in the following ways:

- Manually by creating process objects one by one.
- Using the **Install Standard Functions** Dialog of Object Navigator.
- Using the SCL Importer.

Building a COM500*i* application manually is rather complicated and comes into question only if the number of signals is very small. Process objects can be created by the Object Navigator in the base tools, using Signal X-References tool, or by SCIL.

If process objects already exist in the database, cross-reference information can be imported. For more information, see Section 5.5.17.

For more information on Standard Function Dialog and SCL Importer, see SYS600 Application Design.

5.4.1.1 Application check list

The following issues must be noted when building and maintaining the COM500*i* application.

- The Table Index (TI) attribute of the process objects has been reserved for
- The Free Integer (FI) attribute of those output process objects included in a command in Signal X-References is used for database queries. This attribute should not be changed.
- COM500*i* uses event channels named COM_* to activate command procedures. These must not be removed from the process objects. If any other event channel is to be attached to a process object, the corresponding command procedure (command procedure name = event channel name) must be attached as a secondary object of this event channel.
- Application objects with the logical name COM_* or BNCC* are internal COM500*i* objects and must not be modified.
- A free type object with the logical name COM_GENVAR is an internal COM500*i* object and must not be modified.
- APL:BSV elements 20... 29 are reserved for COM500*i*.
- Station local/remote switch must be in remote position to enable commands from any NCC.

5.4.2 Importing signals

Importing signals means that process objects are imported from another application using Application Object Export/Import tool.

For more information on Application Object Export/Import tool, see SYS600 Installation and Administrator manual.

5.4.3 IET Data Loader

IET Data Loader tool in SYS600 loads data exported from the IET600 engineering tool. Version 5.3 FP1 and later of IET600 supports NCCs of type IEC 101,104 and DNP 3.0. Version 5.3 FP6 and later of IET600 supports NCCs of type IEC 61850. For the configuration importing related to IEC61850 configuration in process direction, see SYS600 IEC61850 System Design manual for details.

5.5 Signal engineering

Signals are divided into indications and commands, i.e. input and output process objects. Indications are sent from process units to COM500*i*, where they are rerouted to one or several NCCs. Usually there are single indications, double indications, and measurements that need to be forwarded to the NCCs.

Commands are sent from the NCC to COM500*i*, where they are rerouted to process units. Secured commands, direct commands, and setpoints are typical commands that are sent. The rerouting of indications and commands is presented in Figure 10.

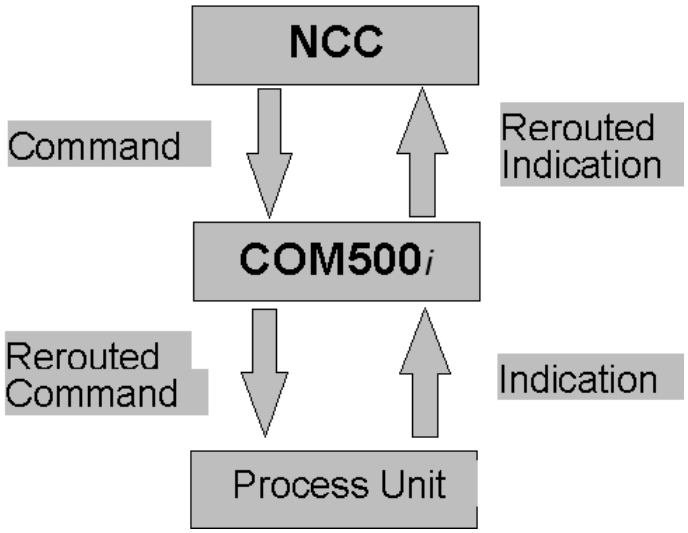


Figure 10: Rerouting of indications and commands

5.5.1 Signal Engineering process

Signal Engineering means that COM500*i* is told how to reroute the signals. This is done by using Signal X-References tool. Signal Engineering is done in the following steps:

1. Add the NCCs to the tool and define the information related to them, such as the protocol that should be used, and define the signals to use. This step is described in Section 5.5.4 and Section 5.5.7.
2. Check that all necessary indications and commands are shown in the tool. If they are not, add them. This step is described in Section 5.5.12.
3. Check that all the necessary attributes for indications and commands are show up correctly in the tool. If they do not, add the missing attributes or change their definitions. This step is described in Section 5.5.9.
4. Define the NCC to which COM500*i* should send the indications. Give the address and additional definitions for the signals. This step is described in Section 5.5.13.
5. Define the address to which COM500*i* should send the command received from the NCC. If the command is needed in a specific form, or if a reply for a command needs to be sent to the NCC, define the Response Indication on the **General** column of the **Commands** tab. This step is described in Section 5.5.14.

Signal Engineering is done in Signal X-References tool, which is described in the following sections.

5.5.2 Using Signal X-References tool

Signal X-References is a tool that is used for mapping signals from the process devices to the NCCs (monitoring direction) and from the NCCs to the process devices (controlling direction). Signal X-References can also be used for making NCC and alarm group definitions, as well as setting the system and application parameters.

If Signal X-References is already open in another monitor, Figure 11 is shown:

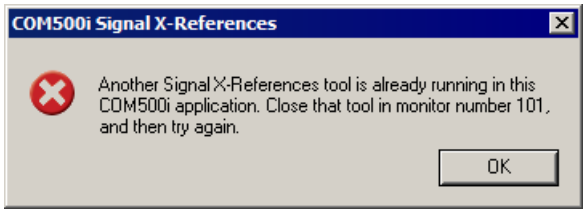


Figure 11: Caution dialog of Signal X-References

Signal X-References is shown in Figure 12. It contains a menubar at the top with six menus, which are the **File** menu, **Edit** menu, **Signal** menu, **View** menu, **Options** menu, and **Help** menu.

Below the menubar there is a toolbar with twelve shortcut buttons, a drop-down menu for the views and a button named **Define** to access the **View Definitions** dialog. Signal X-References contains four notebook tabs, which are the **Indications** tab, **Commands** tab, **NCCs** tab, and **Parameters** tab.

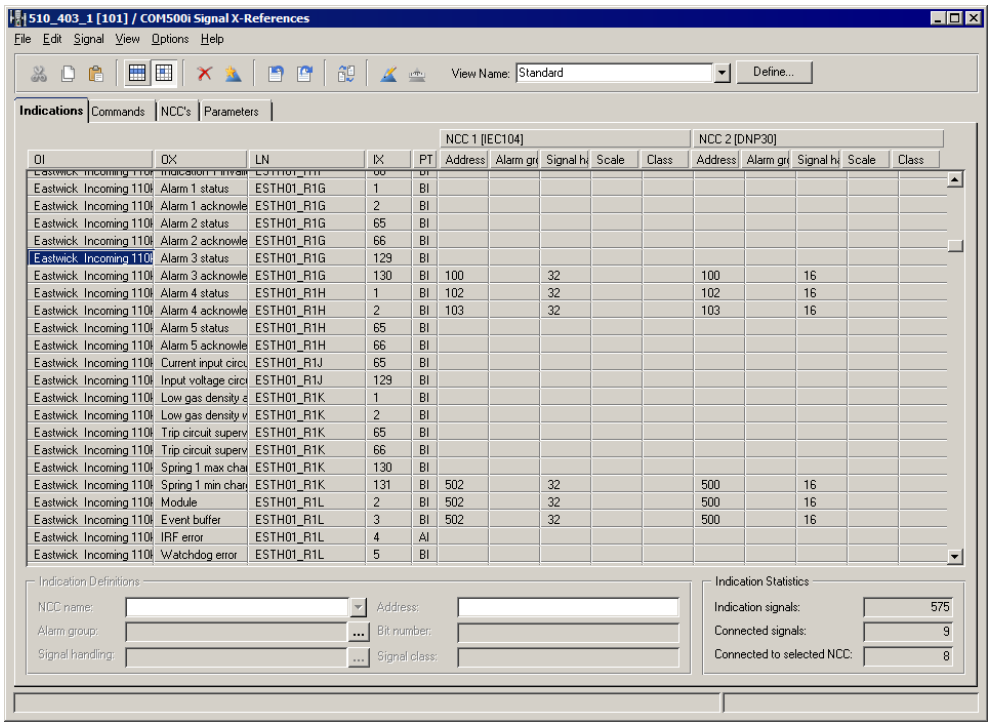


Figure 12: Signal X-References

5.5.2.1 Menus

The **File** menu is used for opening the Signal X-References tools Import, Export and Convert, the Print and Page Setup, and Print dialogs. Selecting **Exit** closes Signal X-References.

The **Edit** menu contains functions for cutting, copying, deleting and pasting text between the text boxes located in tabs and dialogs of Signal X-References. It is also used for opening the Find dialog that can be used for finding text strings from the signals and cross-reference data.

By using the **Signal** menu, signals (process objects) can be added and edited. The scale of analog input process objects can be edited and Signal Diagnostics dialog can be opened. The **Signal** menu is also used for opening the **Column Attributes** dialog, which is used for defining and ordering attributes to be shown in the **Indications** and **Commands** tabs. The last functions in the **Signal** menu are for Trip Signals functionality. For more information about Trip Signals, see Section 5.5.14.

The **View** menu can be used to change the view in Signal X-References. When a view name is selected from the **View** menu, the signals are read from the process database to the **Indications** and **Commands** tabs. The **View** menu can also be used for opening the **View Definitions** dialog. The **View Definitions** dialog is used to define search conditions for signals displayed in the **Indications** and **Commands** tabs. The signals are searched from the process database.

The **Options** menu is used for setting the toolbar visible and invisible and for choosing the select method to be used when editing signals or cross-reference information. The **Options** menu contains an item for enabling and disabling the address overlap check of input signals. The **Options** menu can also be used for opening the **Auto Addressing** dialog. By using the **Auto Addressing** dialog the address offsets for Indications and Commands can be defined. For more information, see Section 5.5.10. It is also possible to enable or disable the displaying of internal process objects in Signal X-References tool. When the menu item Internal Process Objects as Indications is unselected (default), the internal process objects with the logical

name prefix BNCC* are hidden in the **Indications** tab and in the list of Response Indications. When this menu item is selected, the internal process objects for COM500*i* can be seen.

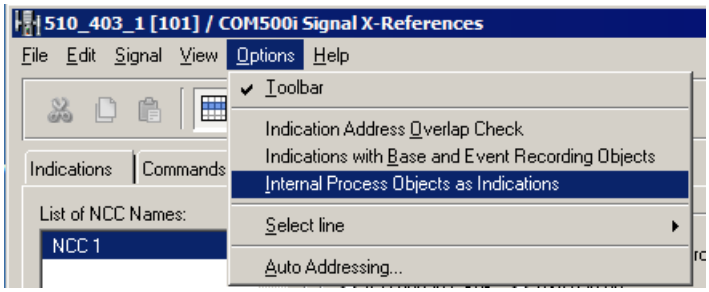


Figure 13: Options menu

The **Help** menu displays information concerning Signal X-References, such as identification, version number, revision, and license information.

5.5.2.2 Toolbar

The toolbar of Signal X-References tool shown in Figure 14 contains twelve shortcut buttons for quick access to the different functions. From left to right the functions of the buttons are: Cut, Copy, Paste, selection method Line, selection method Free, Delete, Add signal, Import, Export, Convert, Edit signal, and Scale. The toolbar also provides a drop-down menu for selecting the view and a **Define** button to access the **View Definitions** dialog.

Figure 14: Toolbar of Signal X-References

5.5.2.3 Tabs

Indications tab
The **Indications** tab is used to define signal cross-references for indications (input process objects). Defining the address specifies where the signal should be send to. The behavior of the signals can also be defined (see the NCC specific columns below).

By default, indication signals include columns for five attributes. The included attributes are:

- Object Identifier
- Object Text
- Logical Name
- Index
- Process Object Type

The tab also has NCC specific columns:

- Address/Identifier
- Alarm group
- Signal handling
- Scale
- Signal class

Signal related definitions are displayed inside the indication signal definition area at the bottom of the **Indications** tab. The indication signal statistics area includes numeric information concerning the signals of the application. The number of indication signals, connected (cross-referenced) signals and the number of signals that are connected to a selected NCC are shown.

Identifier of IEC 61850- and ICCP protocol must match with the value in CCF-file (IEC 61850) and LCC_S-file (ICCP). For more information about CCF-file from SYS600 IEC 61850 Server documentation and about LCC_S-file from SYS600 IEC 60870-6 (ICCP) Protocol documentation.

Commands tab
The **Commands** tab is used for defining signal cross-references for commands (output process objects). This tab includes columns for the same attributes as the **Indications** tab. The following information is entered for each command signal per NCC:

- Command type
- Purpose
- Command group
- Response indication
- Address/Identifier
- Signal handling attributes

Like the **Indication** tab, the **Commands** tab also includes statistics.

Identifier of IEC 61850- and ICCP protocol must match with the value in CCF-file (IEC 61850) and LCC_S-file (ICCP). For more information about CCF-file from SYS600 IEC 61850 Server documentation and about LCC_S-file from SYS600 IEC 60870-6 (ICCP) Protocol documentation.

NCCs tab
The **NCCs** tab is used for adding or deleting NCCs, or for defining the properties of the NCCs. This tab contains a list of the NCC names. At the bottom of the tab there are buttons for adding and deleting NCCs. NCC specific information is displayed on the right side of the tab. Protocol, station number, name, comment text operation mode and group alarm information are displayed for the selected NCC.

Alarm groups are listed inside the alarm information area. The alarm groups can be modified by clicking the corresponding **Add**, **Edit** or **Delete** buttons on the tab, when an NCC is selected.

Parameters tab
A separate tab for the common parameters is included in Signal X-References to enable defining the COM500*i* system and application parameters. These common parameters include time-out parameters and authorization check parameters.

These attributes and their values are saved into a parameter file. The parameter file is taken as input for both the command procedures and Signal X-References, when they are started. Changing the attributes affects the functionality of the active system.

5.5.3 Opening and closing Signal X-References tool

To open Signal X-References, double-click the **Signal X-References** tool icon on the **COM500i** tab of Tool Manager, or click the icon on the **COM500i** tab of Tool Launcher.

To close Signal X-References, select **File/Exit** or click the closing box.

5.5.3.1 Properties

When opening or closing Signal X-References, a Progress Indicator appears on the screen to display the progress of reading indication and command signals from the process database; see Figure 15. If the number of indication or command signals exceeds 10 000 according to the current view definition, a notification dialog box is shown on the screen; see Figure 16. If this happens, reduce the number of signals included in the current view definition.

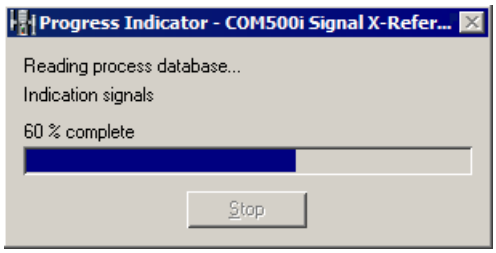


Figure 15: Progress Indicator

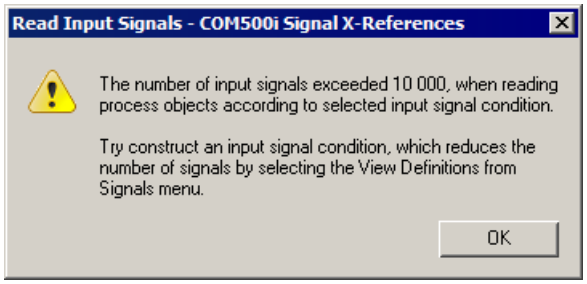


Figure 16: Notification dialog box

5.5.4 Defining NCC properties

NCC properties are defined on the **NCCs** tab; see Figure 17. If these fields are edited, the alarm group names and the drop-down menus of the NCC names shown on the **Indications** and **Commands** tabs are also changed.

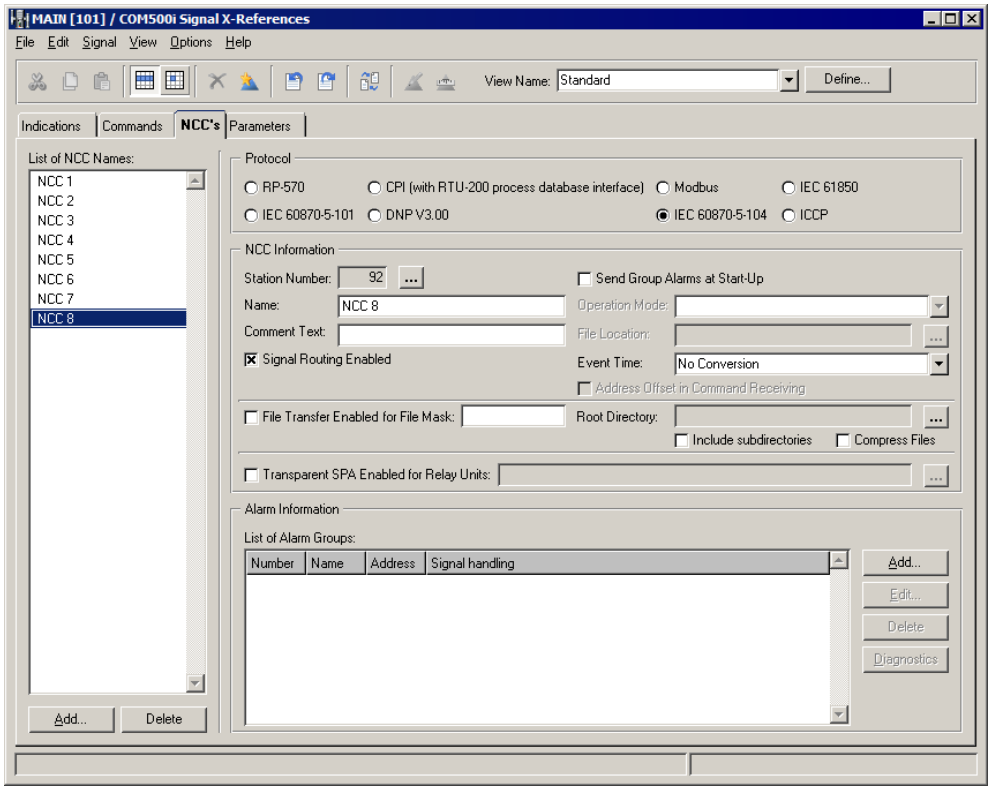


Figure 17: NCC tab of Signal X-References

NCC definitions concern the upper level systems that COM500*i* is supposed to communicate with. COM500*i* can be connected to up to eight NCCs. Before any alarm group or signal definitions can be made, NCC definitions should already exist.

5.5.4.1 Adding an NCC

To add an NCC, click the **Add** button at the bottom of the **NCCs** tab. A new NCC is added to the NCC name list. The NCC name is generated according to the following convention:

name [number], where the name is NCC and the number is 1 to 8.

The name can be edited and its maximum length is 10 characters.

When a new NCC is added, the following dialog is shown in Signal X-References; see Figure 18. In this dialog the user is able to select the NCC type. The possible selections can be chosen from the list of protocols.

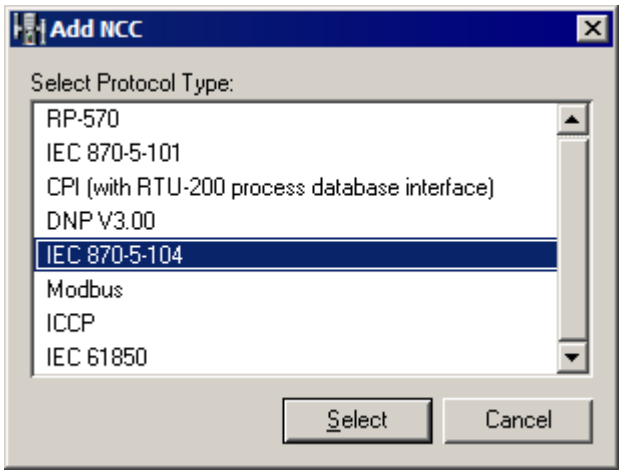


Figure 18: Add NCC dialog for selecting protocol type

When the NCC type has been selected, Signal X-References locates the configured station numbers from the base system that match the appropriate station type. When a correct station number is selected in this dialog, the new NCC type are added into Signal X-References (see Figure 19).

When the Modbus NCC type is selected, Signal X-References locates the station types of RTU from the base system. Figure 19 shows the found station numbers of the RTU station type.

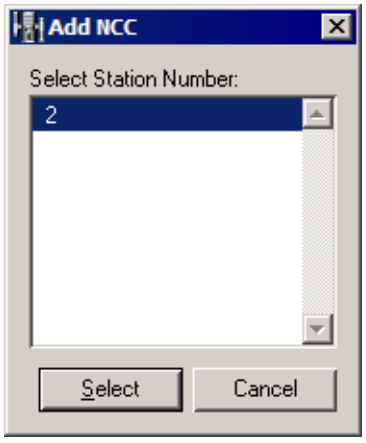


Figure 19: Add NCC2 dialog for selecting station number

After the NCC has been added to the list of protocols, the **NCC** tab shows the defined protocol of the NCC. The appropriate Protocol option is set (see Figure 17).

5.5.4.2 Deleting an NCC

To delete an NCC:

1. Click the NCC name on the list.
2. Click **Delete**. A notification dialog appears on the screen.
3. Click **Yes** and the NCC disappears.

When an NCC is deleted, all the alarm group and signal definitions related to the NCC in question are also deleted.

Only the last NCC can be deleted from the list.

5.5.4.3 Defining NCC properties

Each NCC should have the following properties defined:

- **Protocol**. Either RP 570, IEC 60870-5-101/104, Modbus, DNP 3.0, CPI, IEC 61850 or ICCP. This is the communication protocol used for communication with the upper level system.
- **Station number**. This is the number of the STA:S object number representing the upper level system.
- **Name**. A default name NCC 'n' (n=number) is given when a new NCC is added; see above. This name is also used as the command source name in the COM500*i* command authority check mechanism.
- **Comment Text**. This is free text with the maximum length of 30 characters.
- **Send Group Alarms at Start-up**. If this parameter is set, the value of the group alarm signal is send to the upper level system as a non-time-tagged binary message. This is applicable for IEC 60870-5-101/104. This parameter has no effect on spontaneous alarm signals which are generated in process devices after communication is established.
- **Function Table definitions**. See Section 5.5.5.
- **Address Offset in Command Receiving**. When this parameter is enabled, address offset is used in command receiving. Before the address offset can be used, bit 2 of the RM attribute of the DNP slave station must be 1.
- **Signal Routing Enabled**. As a default, the NCC signal routing is enabled. If explicitly required, the NCC signal routing can be disabled by unsetting the check box. Then data forwarding command procedures do not send data to the selected slave station object. This feature can be accessed by project specific engineering through free type object COM_GENVAR:IZ(5..12) where indexing is related to the NCC number used in configuration. For example, the third NCC in configuration uses COM_GENVAR:PIZ7.
- **Operation Mode**. This parameter is valid only for the DNP 3.0 protocol and it describes how messages are sent between the slave (COM500*i*) and the master (NCC). The descriptions of the operation modes are as follows (for more information, see DNP 3.0 documentation):
  Quiescent Operation. In this mode the master does not poll the slave and all the communication is based on unsolicited report-by-exception messages. The master can send application layer confirmations to the slave.
  Unsolicited Report-by Exception Operation. The communication is basically unsolicited, but the master occasionally sends integrity polls for Class 0 data to verify that its database is up-to-date.
  Polled Report-by Exception Operation. The master frequently polls for event data and occasionally for Class 0 data.
  Polled Static Operation. The master polls only for Class 0 data or the specific data it requires.
- **Event Time**.

Local Time to UTC Conversion. Timestamp of the event is transmitted to NCC in UTC time, when the timestamp is in local time. This feature is useful, when the COM500i is running in local time and NCC expects the events in UTC time. This parameter is valid for IEC 60870-5-104 and DNP 3.0 protocols.
UTC to Local Time Conversion. Timestamp of the event is transmitted to NCC in Local time, when the timestamp is in UTC time. This parameter is valid for IEC 60870-5-104 protocol.
No Conversion. Timestamp of the event is transmitted to NCC without any changes.

The following parameters are valid only for IEC 60870-5-101/104 and IEC 61850 protocols:

- **File Transfer Enabled**. The user can choose whether the file transfer is in use or not. When the file transfer is enabled, COM500*i* sends files from the root directory to the NCC. By default, this option is FALSE.
- **File Mask**. The user can use the file mask to identify the files to be transferred by COM500*i*. The following wildcard characters can be used:

  * Matches with any character string including the null string.

  ? Matches with any single character, at the end of name or extension. It also matches the null character.

  Default File Mask is *.*
- **Root Directory**. The root directory for the transferred files. The default folder is [Appl path]\RECORDER\ROOT.
- **Include subdirectories**. When Include subdirectories is selected, COM500*i* also transfers files from the subdirectories of the root directory.
- **Compress Files**. The sending files can be compressed using the 7Zip program. When COM500*i* compresses the sending files, it is possible to keep file type and directory structure unchanged.
- **Transparent SPA Enabled for Relay Units**. To enable the transparent SPA function, select the **Transparent SPA Enabled for Relay Units** checkbox.

Defining the properties of an NCC:

1. Check either the RP 570, IEC 60870-5-101/104, Modbus, CPI, DNP 3.0, IEC 61850 or ICCP Protocol checkbox to select the protocol the NCC uses.
2. Enter the Station number, Name, and Description for the NCC.
3. Check the **Send Group Alarms at Start-up** checkbox, if this feature needs to be in use.
4. In the case of the DNP 3.0 protocol, select the Operation Mode.

5.5.5 Definition of Function Table download

The Function Table download can be defined in the NCC information panel. Select the Function Table (FTAB) source, if the RP 570 or CPI protocol for the NCC is being used (see Figure 20).

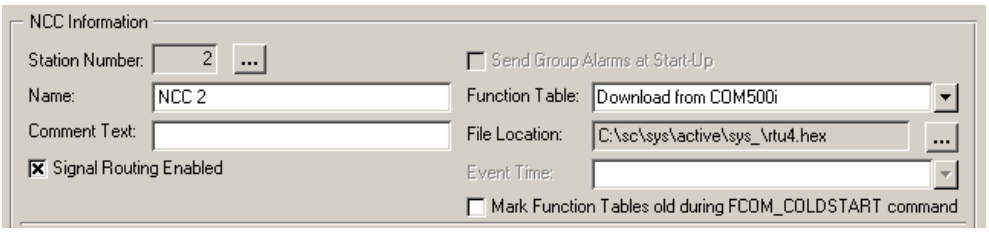


Figure 20: NCC Information panel

The Function Table file that has been chosen and the path to it can be seen in the **File Location** field. In order to change the Function Table source, click the **Browse** button (...)

After clicking the **Browse** button, a standard File Chooser opens. The default file extension is set to .hex (Function Table files).

If Function Table is downloaded from the NCC, the user can define it to be old when the NCC sends the FCOM_COLDSTART command. The definition can be done by checking the box next to Mark Function Tables old during FCOM_COLDSTART command below the **File Location** field. After this procedure, the NCC needs to download the Function Table to COM500*i* again.

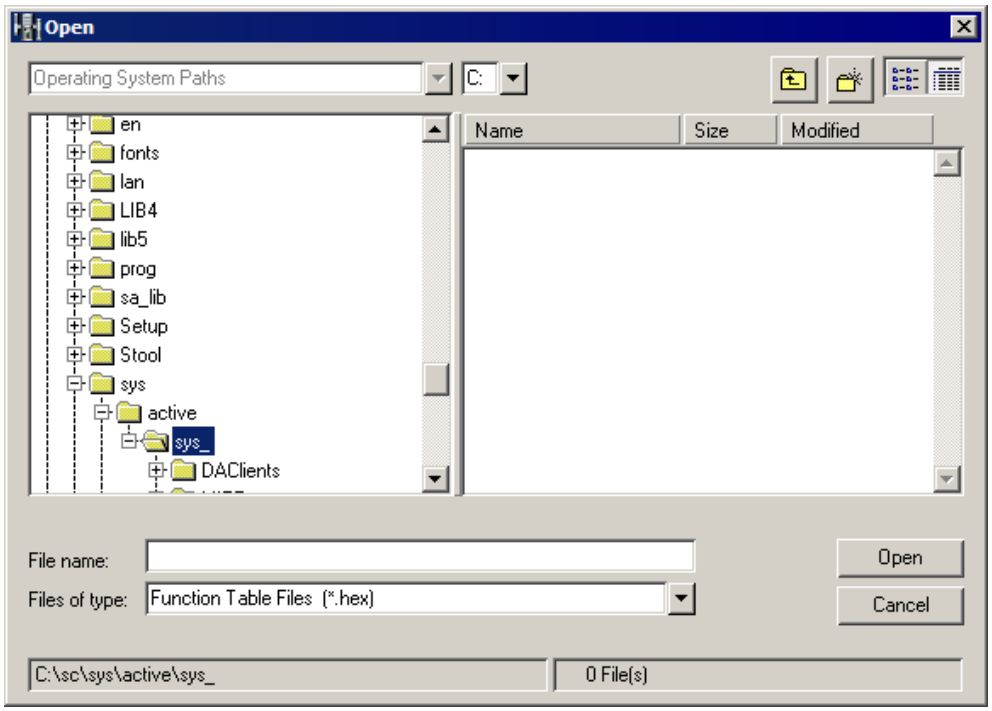


Figure 21: File Chooser

If a file, which is not a Function Table configuration file, is chosen in the File Chooser, the following dialog is displayed.

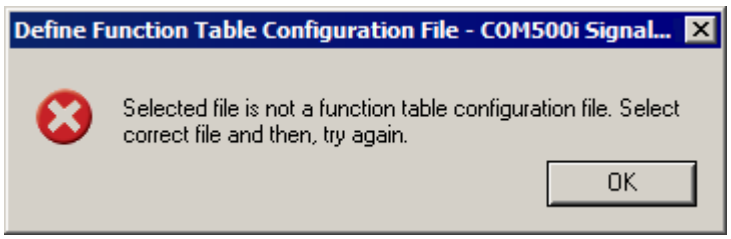


Figure 22: Error dialog when non-FTAB file has been chosen

5.5.6 Configuring transparent SPA

Signal X-References tool is used to configure the signal routing for transparent SPA through COM500*i*. It is possible to enable the transparent SPA functionality for each NCC of IEC 60870-5-101/104-type using the **NCC** tab; see Figure 23.

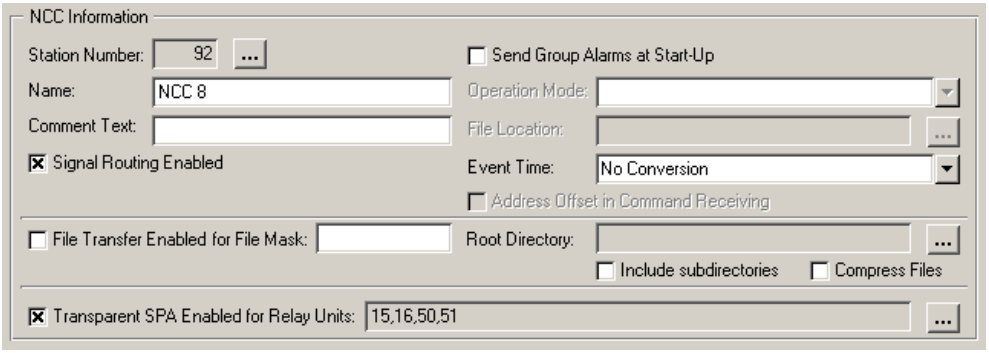


Figure 23: Transparent SPA in use

When the transparent SPA function is enabled, the related relay units and the IEC 60870-5-101/104 object addresses used for SPA responses are defined in a separate dialog; see Figure 24.

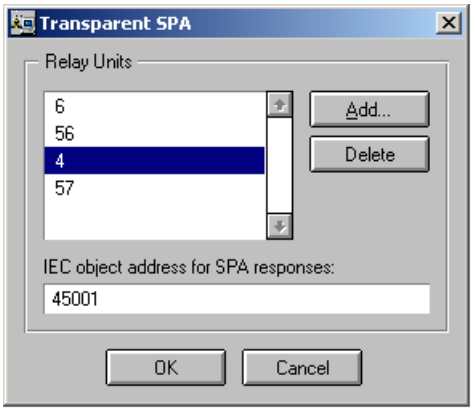


Figure 24: Transparent SPA relay units

An IEC 60870-5-101/104 object address related to the selected relay unit is added, changed, and deleted by using the corresponding dialog items. Adding a relay unit provides a list of the relay units available. Before deleting a relay unit, a confirmation dialog is displayed.

Whenever changes are made either to the relay unit list or the IEC 60870-5-101/104 object address with Signal X-References, the changes are reflected in the related application objects used by signal routing. If there is address overlap between application objects or the user enters invalid information, the corresponding messages are displayed and no change is reflected in the application objects; see Figure 25 and Figure 26.

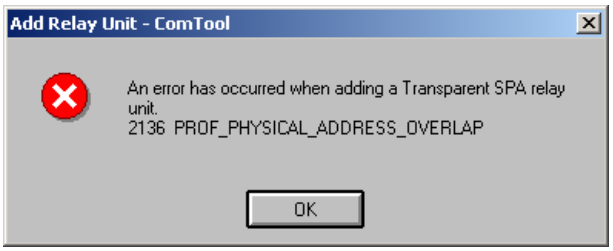


Figure 25: Address overlap

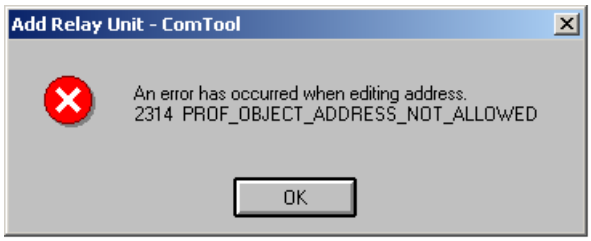


Figure 26: Invalid entry

5.5.7 Alarm groups

A group alarm collects several inputs into a single binary alarm signal, which is forwarded to the NCCs. All the alarms within one bay or all the trip signals can be grouped to reduce the amount of signals sent to the NCCs. The group alarm can also be used to differentiate the alarms. For example, the signals can be divided into different groups according to the priority of the alarm.

The COM500*i* group alarm works as follows:

- When the first of the connected signals gets into the alarming state, the group alarm is set, which means that the binary value 1 is sent to the alarm group address.
- As long as at least one of the connected signals is alarming, the group alarm remains set.
- When the last of the connected signals gets into the non-alarming state, the group alarm is reset, which means that the binary value 0 is sent to the alarm group address.

Each NCC can have up to 2000 alarm groups and multible signals can be connected to one alarm group. It is also possible to connect a signal to an alarm group without mapping the signal itself to an NCC, meaning that the value of the signal is not sent. A signal is counted to be cross-referenced also if only an alarm group has been given to the signal. Maximum amount of cross-referenced signals is 50000 in COM500*i*.

For more information on alarm groups and their diagnostics, see Section 5.6.1.2.

Alarm groups have not been supported for IEC 61850 and ICCP NCCs.

5.5.7.1 Adding alarm groups

To add an alarm group:

1. Click **Add**. The **Alarm Group Definitions** dialog appears on the screen; see Figure 27.

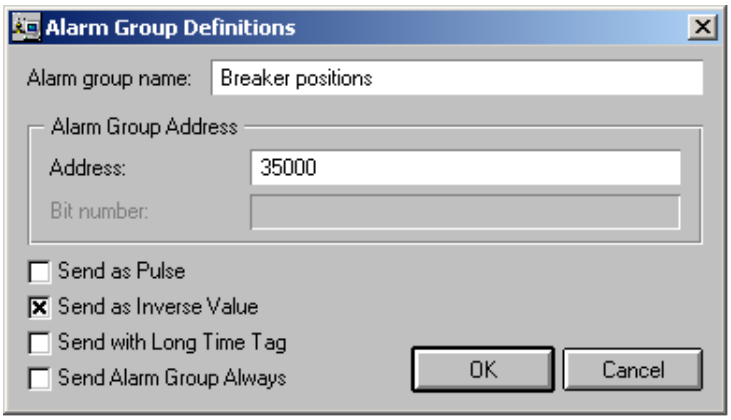


Figure 27: Alarm Group Definitions dialog

2. Type the Alarm Group name.
3. Enter the Alarm Group Address depending on the NCC protocol.
4. Check the appropriate boxes from the following:

   - **Send as Pulse**
     If the signals in an alarm group never get a non-alarm value from a device, the alarm group may always remain in the alarming state. If the alarm group value is defined to be sent as a pulse, every time an alarm occurs in a signal, it is immediately set with a non-alarm value in the alarm group. Thus, if a disturbance recorder starts, the NCC gets both the alarming and non-alarming events within a short time.
   - **Send as Inverse Value**
     By default, COM500*i* uses the value 0 as normal value, and 1 as alarm value. When the **Send as Inverse Value** box is checked, value 1 is the normal value and value 0 is the alarm value, when the alarm group state is sent to the NCC.
   - **Send with Long Time Tag**
     There is an option to send the alarm group information using Long Time Tag. Long Time Tag is mandatory for the IEC 104 Alarm Groups. By standard, it is not possible to use the IEC 101 style short time stamps in IEC 104.
   - **Send Alarm Group Always**
     If this option is selected, the alarm group's state is sent to the NCC every time the information related to the alarm group is updated. Therefore, the alarm group's state is resent each time an alarm signal is updated. Send Alarm Group Always is not selected as a default.

5. Click **OK**.
   A new alarm group is added to the alarm group list.

If an alarm group with the same address already exists, a notification dialog box is displayed on the screen. If this happens, change the address of the new or the existing alarm group.

5.5.7.2 Editing alarm groups

To edit an alarm group:

1. Select an alarm group from the list.
2. Click **Edit** to open the **Alarm Group Definitions** dialog.
3. Change the definitions in the corresponding text fields.
4. Click **OK**.

5.5.7.3 Deleting alarm groups

To delete an alarm group:

1. Click an alarm group on the list to select it.
2. Click the **Delete** button.
   A notification dialog appears on the screen.
3. Click **Yes**.

The alarm group disappears.

5.5.8 Defining views

The query conditions of the signals included on the **Indications** and **Commands** tabs are modified in the **View Definitions** dialog.

5.5.8.1 Opening the View Definitions dialog

To open the **View Definitions** dialog, select **View/Define**; see Figure 28. The **View Definitions** dialog can also be accessed by clicking the **Define** button on the toolbar.

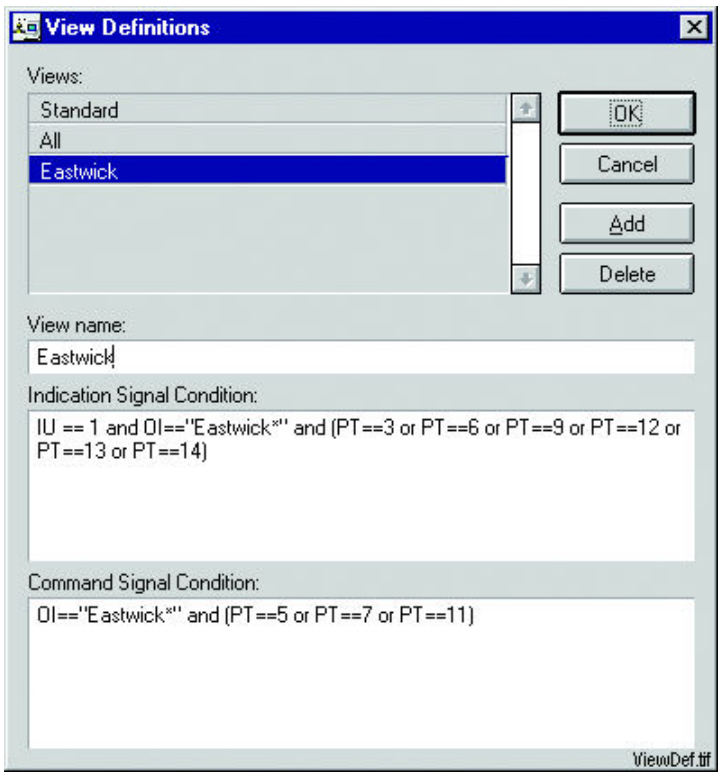


Figure 28: View Definitions dialog

5.5.8.2 View definitions

By default, a non-editable view called Standard is assigned with Signal X-References. To view the query conditions of the indication and command signals defined for a view, click the view name on the list. Figure 28 is an example of a view called Eastwick and its conditions.

5.5.8.3 Adding view definitions

To add a new view definition, click **Add**.

A new view name is added to the list based on the convention:

name [view number]

where the name is the name of the view and the view number is the number of the view in View Definitions.

The query conditions of the Standard view are copied to the new view by default. The conditions can be modified to suit the purposes of the new view.

5.5.8.4 Closing the View Definitions dialog

Click **OK** to close the **View Definitions** dialog. The validity of the new view condition is checked, and if it is invalid, a dialog is shown. The names of the valid view definitions are added as separate menu items to the **View** menu.

5.5.9 Defining attributes for columns

The column attributes attached to the **Indications** and **Commands** tabs can be modified by using the **Column Attributes** dialog.

5.5.9.1 Opening the Column Attributes dialog

To open the **Column Attributes** dialog, select **Signal/Add Column Attributes**(see Figure 29).

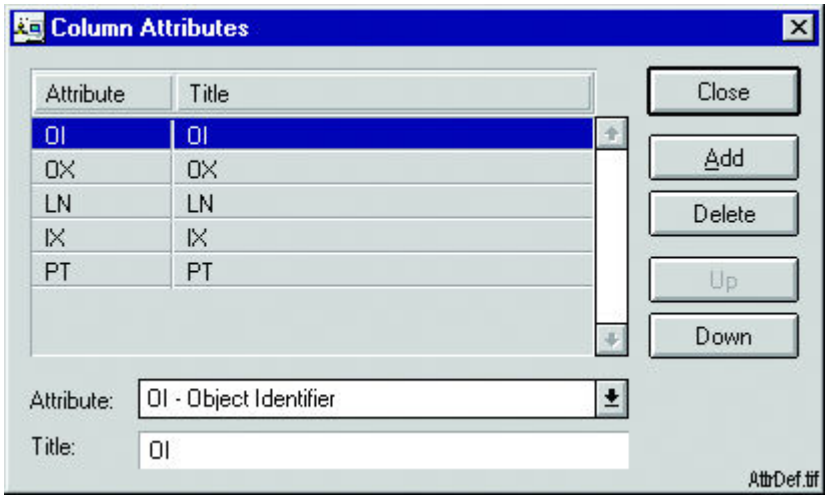


Figure 29: Column Attributes dialog

By default, five columns are attached to the **Indications** and **Commands** tabs. These are:

- Object Identifier (OI)
- Object Text (OX)
- Logical Name (LN)
- Index (IX)
- Process Object Type (PT)

5.5.9.2 Order of attributes

To change the order of the attributes, select an attribute to relocate and click the **Up** or **Down** button.

5.5.9.3 Closing the Column Attributes dialog

Click **OK** to close the dialog. If new attributes were added to the list, new columns are added to the **Indications** and **Commands** tabs. These are located on the right side of the tab.

5.5.9.4 Defining attribute column widths

To define the attribute column widths on the **Indications** and **Commands** tabs, move the rulers. The width of the column is set, when the ruler is dropped.

If the width of a column is changed on the **Indications** tab, the corresponding column is also changed on the **Commands** tab and vice versa.

5.5.9.5 Adding new attributes

To add a new attribute:

1. Click **Add**.
2. Select an attribute from the **Attribute** drop-down menu.
3. Type a Title for the new attribute. If no title is specified for the new attribute, a two-char attribute name is used as the default title.

The new attribute is added to the list.

The maximum number of columns in Signal X-references tool is 15.

5.5.9.6 Editing attributes

To edit an attribute in the **Column Attributes** dialog:

1. Select the attribute on the list.
2. Modify the attribute information either in the **Attribute** drop-down menu or in the title text field.

5.5.9.7 Deleting attributes

To delete an attribute from the **Column Attributes** dialog, select the attribute on the list and click **Delete**.

It is not possible to delete all the column attributes. At least one attribute must be defined.

5.5.10 Defining auto-address parameters

Auto-addressing parameters are used when cross-reference signals are copied and pasted into Signal X-References tool. The purpose of the auto-addressing mechanism is to define address offsets to avoid address overlaps when cross-reference data is copied and pasted. For example, if the cross-references of an indication connected to an RP 570 NCC are copied and the address of the signal is 002^001 while the RP 570 address offsets are 1 (block number) and 2 (bit number), the NCC address of the signal where the cross-reference is pasted is 003^003.

For the different NCC protocols the auto-addressing parameters are as follows:

- For RP 570 and CPI block address 1…255 and bit address 0…15 for binary objects and block 1...2000 for analog objects
- For IEC 60870-5-101/104 IEC address 1…65535
- For DNP 3.0 index 1…65535
- For Modbus block address 1...4096 and bit address 0...15 for binary objects and block 0...65535 for analog objects
- For IEC 61850 or ICCP cannot use auto-addressing. IEC 61850 and ICCP use identifier which must match with the value in CCF-file (IEC 61850) and LCC_S-file (ICCP).

IEC 60870-5-101/104 and DNP 3.0 addresses depend on the used IL (Information object address Length) station attribute value.

Auto-addressing parameters are defined using the **Auto Addressing** dialog shown in Figure 30.

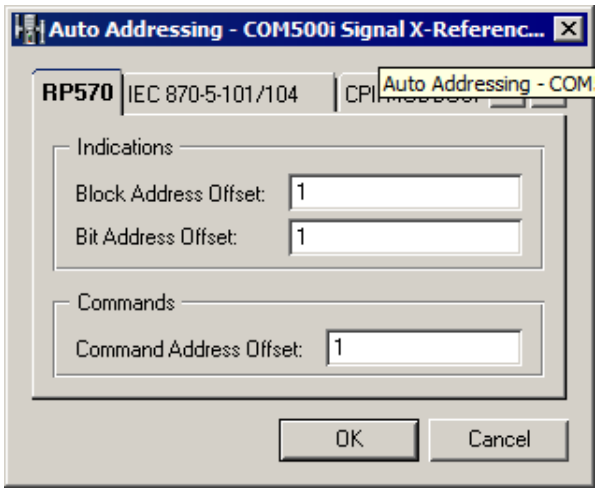


Figure 30: Auto Addressing dialog

5.5.10.1 Opening the Auto Addressing dialog

To open the **Auto Addressing** dialog, select **Settings/Auto Addressing**.

5.5.10.2 Defining auto-addressing parameters

To define the auto-addressing parameters, select the used NCC protocols and write the address offsets in the corresponding text fields.

5.5.10.3 Closing the Auto Addressing dialog

Click **OK** to close the dialog. The defined auto-addressing parameters are used when cross-reference information is pasted in Signal X-References tool.

5.5.11 Indication address overlap check

The Indication Address Overlap Check can be taken into use by selecting the corresponding item in the **Options** menu. When it is in use, this function checks for address overlaps every time a new address is given. If an overlap is detected, the user is notified with a dialog (see Figure 31). Unselecting the corresponding menu item disables the address overlap check.

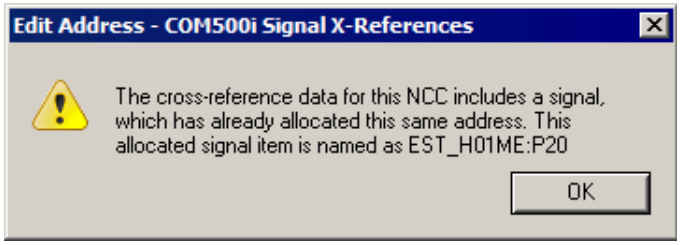


Figure 31: Address overlap notification dialog box

The notification dialog also appears in some other cases, such as, if consecutive bit addresses for DB (Double binary indication) type objects are given, or if the same bit address is given for DB and BI (Binary input) type objects. This is because the DB and BI type objects are found in the same memory space in the RP 570, Modbus and CPI protocols.

When the address overlap check is in use it may decrease the performance of Signal X-References, especially if the number of indication signals is significant.

5.5.12 Signal handling

It is possible to add, delete and edit signals (process objects) by using Signal X-References tool. The attributes of existing process objects can also be modified.

5.5.12.1 Adding signals

To add a signal:

1. Select **New** from the **Signal** menu. This opens the **New Signal** dialog shown in Figure 32.
2. Enter the logical name and index of the new signal.
3. Click **OK**
   The signal appears in Signal X-References.

For more information about creating new process objects, see SYS600 Application Objects.

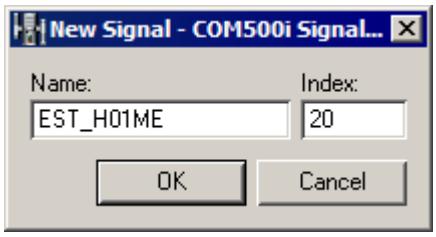


Figure 32: New Signal dialog

5.5.12.2 Editing signals

To edit the attributes of an existing signal:

1. Double-click the row of the signal to be edited.
   This opens Process Object Tool.
2. Edit the attributes.
3. Click **OK**.

5.5.12.3 Deleting signals

To delete a signal:

1. Change the Selection method to Line from the **Options** menu.
2. Select the line of the signal to be deleted.
3. Click right mouse button and select **Delete Signal** or select **Delete** from the **Edit** menu.
   This opens a confirmation dialog box.
4. Click **OK**
   The signal is deleted.

5.5.13 Defining indication cross-references

Cross-references for indication signals are defined on the **Indications** tab (see Figure 33). Cross-referencing indication signal means defining either the address or additional definitions. It is also possible to define the signal handling of a cross-referenced signal. When an address has been defined for the signal, the indication is sent to the NCCs. When an alarm group has been defined for the signal, the alarm group is activated.

COM500i supports maximum 50000 cross-referenced indications. An indication is counted to be cross-referenced, if an address or an alarm group is given at least to one NCC.

The following properties should be defined for each indication (input process object) :

- NCC address
  The address in the upper level system where the signal is sent to. In RP 570, Modbus and CPI the address consists of a block number (1... 255 for RP 570 and CPI, 1... 4095 for Modbus) and possibly a bit number (0... 15). In IEC 60870-5-101/104 and DNP 3.0 protocols the address is an integer (IEC address) the range of which is determined by the Information Address Length (IL) attribute of the corresponding station. The address can be omitted, if the signal is only connected to a group alarm and the value itself is not sent.
- Alarm group information
  The alarm group to which the signal is connected. This information can be omitted if the signal is not sent to any alarm group. Note that the alarm group is presented as a number in Signal X-References. Alarm groups have not been supported for IEC 61850 and ICCP NCCs.
- Signal handling attributes
  These attributes define how the signal is handled before it is sent to the NCCs. For example, a double binary signal can be sent as a single indication. Note that the alarm group is presented as a number (a bitmask of the numbers of the selected signal handling attributes) in Signal X-References.
- Signal class
  The signal class that it used when the signal is sent to the NCC can be selected for signals connected to IEC 60870-5-101/104 or DNP.
- Scale algorithm
  The scale algorithm that is used when the signal is sent to the NCC can be selected for each analog signal. If no signal is selected, the signal is scaled with the algorithm 1:1.
- Group interrogation and counter group interrogation for IEC 60870-5-101/104
  Interrogation group number (integer 1...16) or counter interrogation group number (integer 1...4) is added to the Free Integer (FI) attribute of the cross-reference process object. By default, the value of the FI attribute is 0. A signal specific group number is defined to Signal X-References as follows (See Figure 34):

  1. Add a FI attribute into Signal X-References from Column Attribute dialog.
  2. Rename the column title from FI - Free Integer to Group Nr.

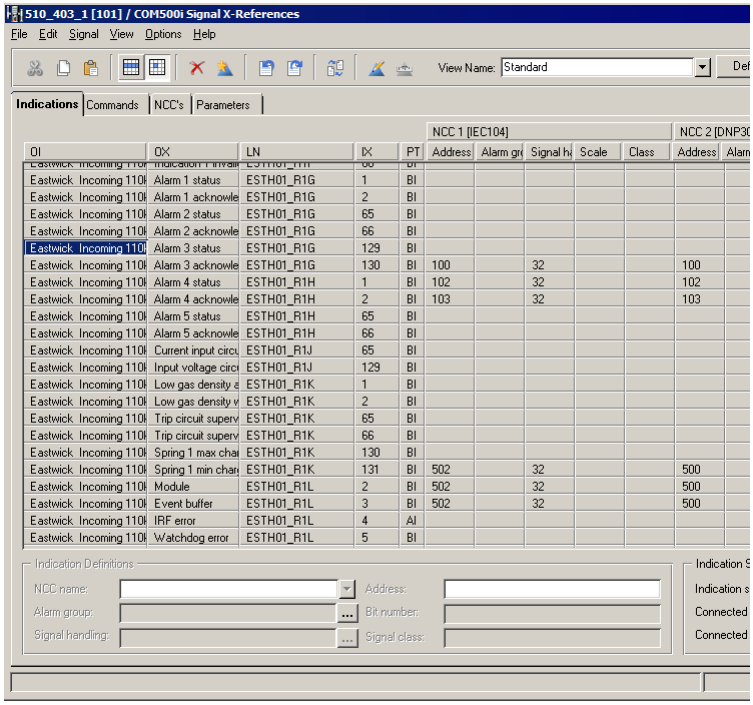


Figure 33: Indications tab of Signal X-References tool

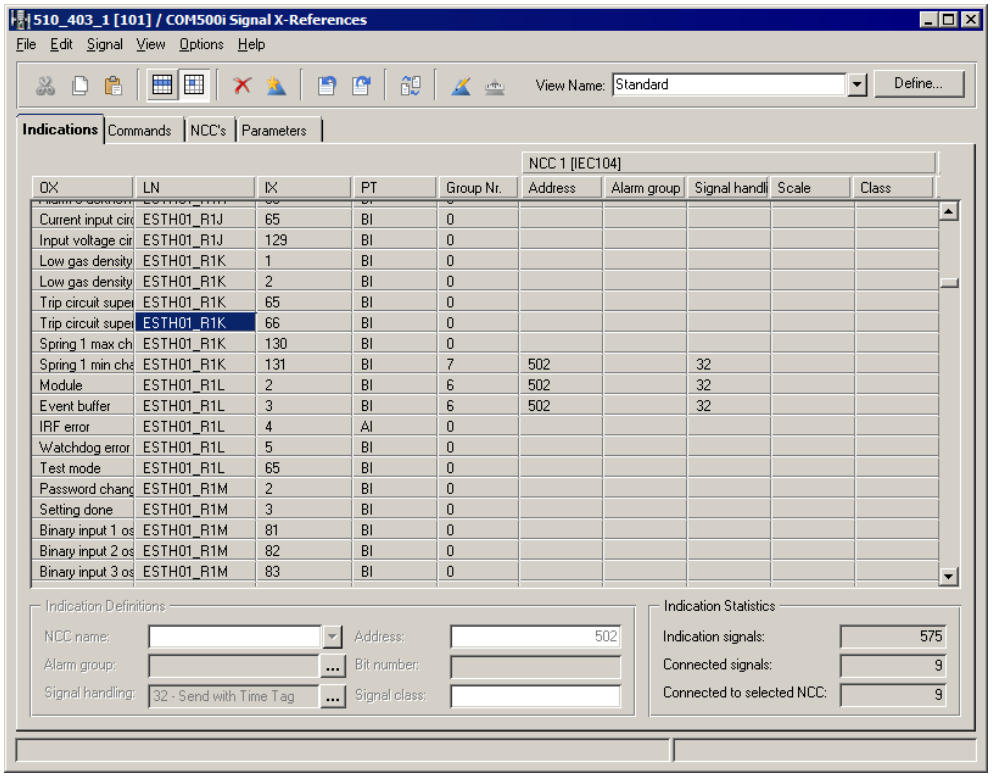


Figure 34: Group interrogation number for IEC 60870-5-101/104

5.5.13.1 Adding cross-references

To add a cross-reference for the indication signal:

1. Click the correct signal row in the indication signals list.
2. Select an NCC by clicking a field below the NCC column.

   If the data type of the selected signal is not supported in the protocol of the NCC, a message is shown on the status bar and the selections described below are disabled.
3. To add an alarm group, click the button on the right side of the alarm group field.
   The **Alarm Groups** dialog appears on the screen. The alternatives include the alarm groups that have been added to the NCC on the **NCCs** tab.
4. Select one alarm group from the list.
5. Define the address based on the block and bit number for RP 570 protocol, Modbus or CPI, IEC address for IEC 60870-5-101/104 protocol and index for DNP 3.0 protocol.
6. In the case of an analog input signal, define a scale. It is possible to select a different scale for each signal and for each NCC.
7. Click the button on the right side of the signal-handling field and the **Signal Handling Attributes** dialog appears on the screen (see Figure 35). Select one or several signal handling attributes from the list of attributes. To select several attributes, hold the CTRL key down while clicking the attributes.
8. Define the signal specific class.
   For the IEC 60870-5-101 protocol the signal classes are 1 and 2. If the class has not been defined, it is 1.
   For the DNP protocol the signal classes are 0, 1, 2 and 3. The default class is 1 for binary inputs and double binary indications. For other signal types the default class is 2.
9. Click **OK**.
   The cross-reference of the indication signal is ready. It is displayed under the cross-referenced NCC name.

Indication signal types and the corresponding signal handling attribute values are listed in Table 1. When the signal handling attribute Project Specific is selected, a specific block in the corresponding command procedure is executed to enable the project specific modifications. For more information, see Section 6.

Table 1: Signal handling attributes related to different data types

Data	RP 570 / CPI / Modbus	IEC 60870-5-101 / 104	DNP 3.0
Binary input	- Project Specific - Send with Time Tag - Send as Inverse Value - Send as Double Binary - None	- Project Specific - Send with Time Tag - Send as Inverse Value - Send as Double Binary - Send with and without Time Tag - Send with Long Time Tag - None	- Project Specific - Send as Inverse Value - Send as Double Binary - Send Change - Send Change with Time - Send Change with Relative Time - Send Always as Event - Send as Double Bit Input Object - None
Double binary	- Project Specific - Send with Time Tag - Send as Inverse Value - Send as Single Indication - None	- Project Specific - Send with Time Tag - Send as Inverse Value - Send as Single Indication - Send with and without Time Tag - Send with Long Time Tag - None	- Project Specific - Send as Inverse Value - Send as Single Indication - Send Change - Send Change with Time - Send Change with Relative Time - Send Always as Event - Send as Double Bit Input Object - None
Digital input	- Project Specific - Send with Time Tag - Send as Analog Value - None	- Project Specific - Send with Time Tag - Send as Analog Value - Send with and without Time Tag - Send with Long Time Tag - None	
Analog input	- Project Specific - Send with Time Tag - None	- Project Specific - Send with Time Tag - Send as Floating Point Value - Send with and without Time Tag - Send as Scaled Value - Send with Long Time Tag - Send as Binary Input - None	- Project Specific - Send as 16-bit Value - Send without Flag - Send Change Event without Time - Send Always as Event - Send Change Event with Time - Send as Binary Input - None
Pulse counter	- Project Specific - None	- Project Specific - Send with Time Tag - Send with and without Time Tag - Send with Long Time Tag - None	- Project Specific - Send as 32-bit value - Send as Delta Counter - Send without Flag - Send Change Event without Time - Send Always as Event - Send Change Event with Time - None
OPC Event	- Project Specific - Send with Time Tag - Send as Inverse Value - Send as Double Binary - Send as Analog Value - None	- Project Specific - Send with Time Tag - Send as Inverse Value - Send as Double Binary - Send with and without Time Tag - Send with Long Time Tag - Send as Analog Value - None	- Project Specific - Send as Inverse Value - Send as Double Binary - Send Change - Send Change with Time - Send Change with Relative Time - Send Always as Event - Send as Analog Value - None

Table 2: Signal handling attributes related to different data types (IEC 61850 and ICCP)

Data	IEC 61850	ICCP
Binary input	- Project Specific - Send as Inverse Value - Send as Double Indication - None	- Project Specific - Send as Inverse Value - Send as Protection Event - None
Double binary	- Project Specific - Send as Inverse Value - Send as Single Indication - None	- Project Specific - Send as Inverse Value - Send as Protection Event - None
Digital input	- Project Specific - None	- Project Specific - None
Analog input	- Project Specific - None	- Project Specific - Send as Integer Values - None
Pulse counter	- Project Specific - None	- Project Specific - None
OPC Event		

IEC 60870-5-104 signal handling attributes differ from IEC 60870-5-101 only by the usage of time stamp types. By standard, it is not possible to use the IEC 101 style short time stamps in IEC 104. These are however still possible to use in COM500*i*. By default, IEC 104 uses long time stamps with date.

The Modbus protocol does not have time tagged events. Also, it does not separate analog and digital inputs, which are both set to 16 or 32 bits registers.

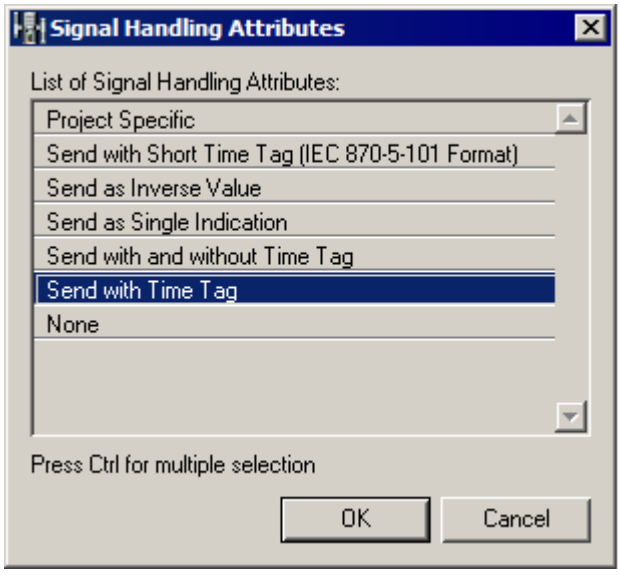


Figure 35: Signal Handling Attribute dialog

To select a scale for an analog input signal:

1. Click the Scale column of the selected NCC to change the **Signal Handling** field into the **Scale** field.
2. Click the button on the right side of this field to open the **Scale** dialog shown in Figure 36.
3. Select a scale from the list and click **OK**.
   Note that the name of the scale cannot be longer than ten characters.

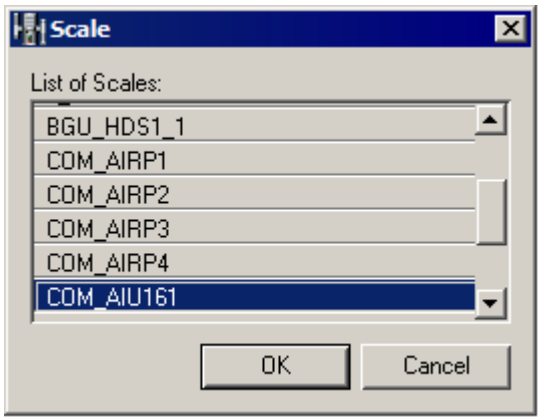


Figure 36: Scale dialog

To edit the attributes of a selected scale object:

1. Select a cell containing a scale name.
2. Select **Scale** from the **Signal** menu.
   This opens Scale Object Tool dialog shown in Figure 37.

The scaling of analog signals in COM500*i* is described in more detail in Section 6.

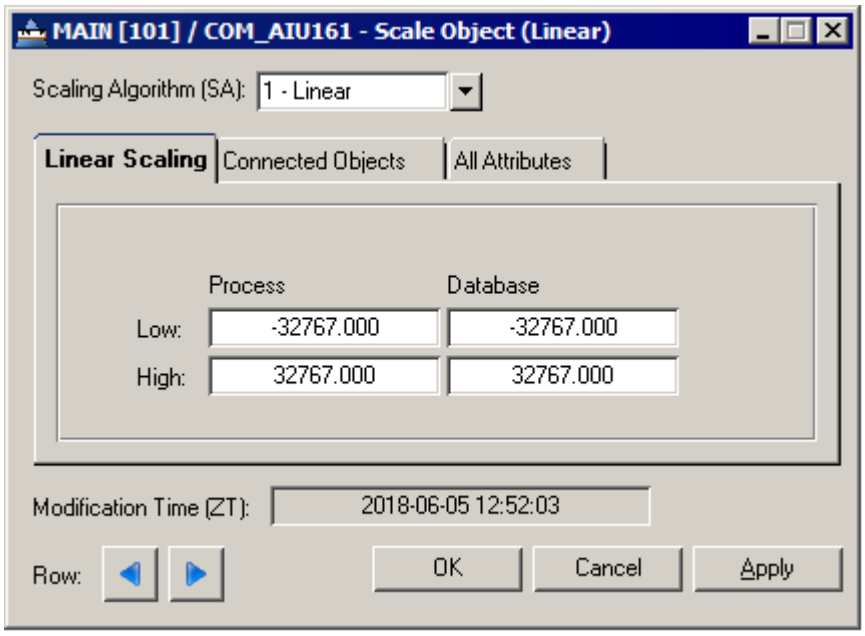


Figure 37: Scale Object Tool dialog

RTU specific configuration
When adding cross-references for signals of RTU type, the following convention should be checked:

- With binary data, low index is used for storing data which is received from block sending. High index is used for storing event data and its time tag. In COM500*i*, high index (low index + 100) is used for cross-referencing process data to the NCC.
- At start-up, the event recording object is in Not Sampled Status and low index is used for data source. The event recording object is always updated at start-up and can be used by COM500*i* if high index is not updated.
- Other types of data (analog input, pulse counter) are cross-referenced from their low index. Analog Inputs: high index is used for alarm/warning data.
- If data is to be attached to an alarm group, it is taken from the low index

Cut, copy and pasting cross-references
Cross-reference data can be cut, copied and pasted from one signal to another, or from a group of signals to another by taking the following steps:

1. Set the **Selecting method** as **Free** from the **Options** menu or the toolbar.
2. Highlight the cross-reference data to be cut/copied and pasted with mouse on the **Indication** tab.
3. Cut or copy the cross-reference data by selecting **Cut** or **Copy** from the **Edit** menu or by using the corresponding shortcut button of the toolbar.
4. Click the field in the upper left corner of the area where the cross-reference data is to be pasted.
5. Paste the cross-reference data by selecting **Edit/Paste** or by using the corresponding shortcut button.

5.5.13.2 Deleting cross-references

To delete a cross-reference from the signal:

1. Click an indication signal row.
2. Select an NCC by clicking the table below the appropriate NCC.
3. Clear the **Address** and the Signal class fields. If a signal handling attribute or scale is used, set them to None.
4. If an alarm group has been defined for the indication signal, select **None** from the **Alarm group** drop-down menu.
   The cross-reference is deleted between the indication signal and the corresponding NCC.

To delete the cross-reference using the delete function:

1. Set the **Selecting method** as **Free** from the **Options** menu or the toolbar.
2. Highlight the cross-reference data to be deleted with mouse on the **Indication** tab.
3. Delete the cross-reference data by selecting **Edit/Delete** or by using the corresponding button.

To clear the cross-reference using the clear function:

1. Set the **Selecting method** as **Line** from the **Options** menu or the toolbar.
2. Select a signal or multiple signals.
3. Righ click and select **Clear Cross-References**.

5.5.14 Initialization of event state signals

The initial state from the event state signals, such as trip signals, is not received during the communication start-up. The state is received only when an event occurs in the system. This can happen 1 minute or even 5 years after the communication start-up. These process objects are seen as erroneous until the first event occurs. Additional configuration is required to assign a default value to the process objects which have been cross-referenced in COM500*i* to be sent to the NCC. With this method the indication signals that have been collected into the Trip Signals list in Signal X-References get a default object value and a valid object status (OK_STATUS). All possible data types are supported. With measurements the value used is 0,

and with indications the default value is a non-alarm value (the setting of non-alarm values is based on information in Alarm Generation [AG] and Alarm Activation [LA] process object attributes).

5.5.14.1 Add indication signals to the Trip Signals list

To add indication signals to the Trip Signals list:

1. Select a signal in the **Indications** tab.
2. Click the right mouse button
   A shortcut menu appears on the screen.
3. Select **Add to Trip Signals**, see Figure 38

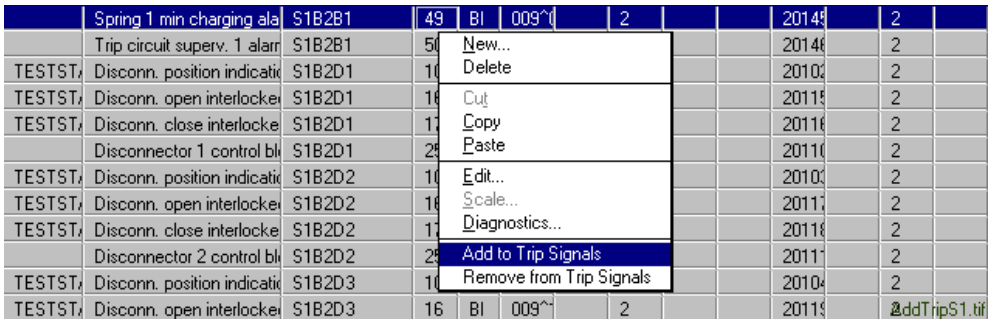


Figure 38: Adding indication signals to Trip Signals list

Indication signals can also be added to the Trip Signals list as follows:

1. Select a signal in the **Indications** tab.
2. Select Signal from the menu bar.
3. Select **Add to Trip Signals**, see Figure 39.

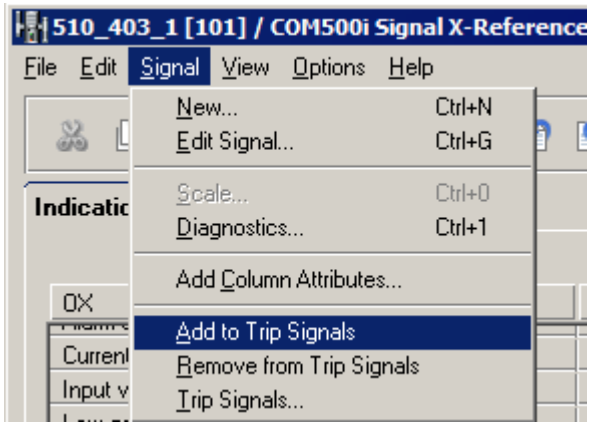


Figure 39: Add new indication signals to Trip Signals list.

5.5.14.2 Remove indication signals from the Trip Signals list

To remove indication signals from the Trip Signals list:

1. Select a signal in the **Indications** tab.
2. Click the right mouse button.
   A shortcut menu appears on the screen.
3. Select **Remove from Trip Signals**, see Figure 38.

This can also be done in another way:

1. Select a signal in the **Indications** tab.
2. Select **Signal** from the menu bar.
3. Select **Remove from Trip Signals**; see Figure 39.

5.5.14.3 View indication signals included in the Trip Signals list

To view the indication signals included in the Trip Signals list:

- Select **Signal/Trip Signals**.

A new dialog appears (see Figure 40). This dialog shows all the existing signals.

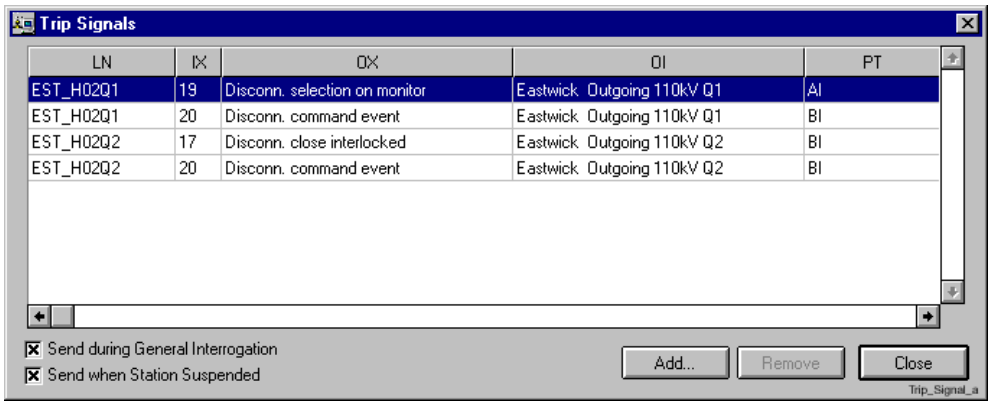


Figure 40: Trip Signals list

In the **Trip Signals** dialog, there are selections for the behavior of trip signals in the following situations:

- **Send during General Interrogation**
  It is possible to define whether COM500*i* sends trip signals during the general interrogation command from the NCC or not. By default, the trip signals are sent.
- **Send when Station Suspended**
  It is possible to define that COM500*i* sends trip signals when the station enters the suspended state. By default, the trip signals are sent.

5.5.14.4 Add new trip signals to the Trip Signals list

To add new trip signals to the list:

1. Click **Add**.
   A dialog with the List of Indications appears on the screen, see Figure 41.

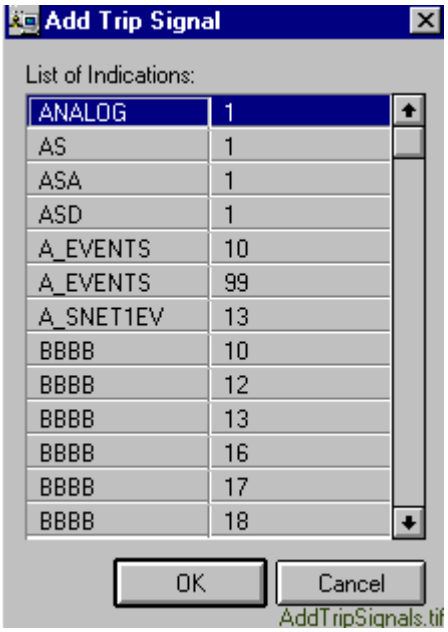


Figure 41: Add Trip Signals dialog

2. Select the indication signal
3. Click **OK**.

5.5.14.5 Clear a trip signal from the Trip Signals list

To clear an existing signal from the Trip Signals list:

1. Select a signal from the list.
2. Click **Remove**.

5.5.15 Defining command cross-references

Cross-references for commands (output process objects) are defined on the **Commands** tab of Signal X-References (see Figure 42). COM500i supports maximum 14000 cross-referenced commands.

Devices connected to SYS600 which communicate using different protocols are controlled through different command philosophies on the application level. Consequently, in addition being a protocol converter, COM500*i* has been able to make conversions between different methods of command handling. For this purpose, the following parameters must be given for each command in Signal:

1. **Type** specifies the control philosophy of the application level.
   For binary output, analog output and digital output process objects:
   - Direct command. Note that the object commands of IEC 60870-5-103 devices must be made using this command type.

   For binary output process objects:
   - Direct command with two output objects (for example REF 542, tap changer raise/lower commands).
   - Secured command with four output objects (for example SPACOM devices, REF 543).
   - Secured command with one output object (for example IEC 60870-5-101 devices, RTU 200).
   - Secured command with five output objects.

- Secured command with two output objects and Select.
- Secured command with three output objects.
- Secured command with six output objects (Three State Switch).

For analog output process objects:

- Secured command with one output object (for example REC 561 devices). This can be received in binary, double binary, or analog format depending on the used signal handling attribute and NCC protocol.
- Secured command with five output objects (IEC 61850 devices).
- Secured command with one output object and Select (IEC 60870-5-101/104 and DNP 3.0 devices)

2. **Purpose** specifies the function of an individual signal (output process object). The values displayed in the HSI depend on the Type and are as follows:

   - Open command (direct command with 2 output objects, secured command with 2 output objects and Secured command with 3 output objects)
   - Close command (direct command with 2 output objects, secured command with 2 output objects and Secured command with 3 output objects)
   - Select command (secured command with 3 output objects)
   - Open select (secured command with 4 output objects, 5 output objects and 6 output objects)
   - Close select (secured command with 4 output objects, 5 output objects and 6 output objects)
   - Free select (secured command with 6 output objects)
   - Earth select (secured command with 6 output objects)
   - Execute (secured command with 4 output objects and 6 output objects)
   - Open Execute (secured command with 5 output objects)
   - Close Execute (secured command with 5 output objects)
   - Cancel (secured command with 4 output objects, secured command with 5 output objects)

   For direct commands the purpose has no meaning and selecting it is disabled.
   Command Group specifies the group-controlled objects (for example breaker and truck) with output process objects sharing the same logical name. Output objects of the different object should be given different command group numbers. Value: 1 to 5.
   For each command signal (output process object), the following properties should be defined.
   General, signal-related information (common for all NCCs):

   - Command type, i.e. how the command is presented in the COM500*i* process database
   - Purpose of the signal
   - Number of the command group
   - Response indication. This is the input process object that is updated as a result of the command. For example, the position indication object of a breaker is the response indication of the breaker (open/close) command. This information ensures that the IEC 60870-5-101/104 or IEC 61850 messages are sent in the correct order to the NCC, when commands are received from the NCC. When a response indication is received from the process device, COM500*i* sends a command termination to an IEC 60870-5-101/104 or an IEC 61850 NCC. If a command is timed out or COM500*i* cannot operate the device, a negative command termination is sent to the NCC. If the IEC 60870-5-101/104 or IEC 61850 protocol is not used to connect any of the NCCs, this information can be omitted.

   Response indication cannot be defined for all direct type of commands. It is allowed only for the secured type commands and for 1 BO/1 AO direct command types.

   NCC related information:

- Signal handling attributes define how the signal is handled before it is sent to the process devices. For example, an object command can be received as an inverse value. Note that the alarm group is presented as a number (a bit mask of the numbers of the selected signal handling attributes) in Signal X-References.

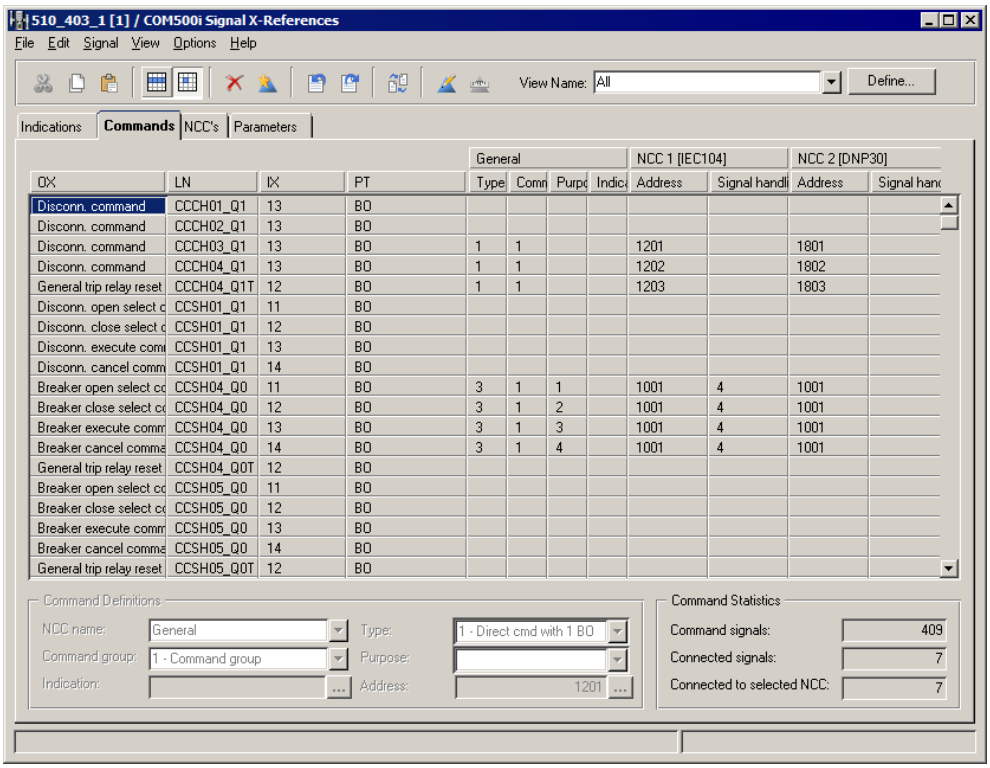


Figure 42: Commands tab

5.5.15.1 Adding cross-references

To add a cross-reference for a command signal:

1. Click a row in the command signals list.
2. Select the **General** column by clicking the same command signal row under **General** column.
3. Select a command type from the **Type** drop-down menu.
4. Select a signal purpose from the **Purpose** drop-down menu.
5. If an alarm group has been defined for the Indication signal, select **None** in the **Alarm Groups** dialog.
6. Select the Command Group number from the drop-down menu.
   When the command type is set to direct or Secured command with 1 output object, the command group of the cross-referenced signal is automatically set to 1.

   To change the command group number, the command type must first be set to None and then back.
7. Click the button next to the **Indication** field.
   The **Indication** dialog is opened (see Figure 43).

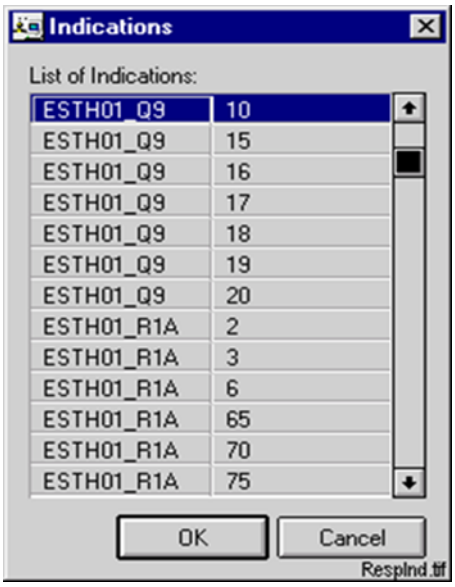


Figure 43: Indications dialog

8. Select the correct indication on the list and click **OK**.
   The selected return indication appears in the **Indication** field.
9. Enter the Address to the **command** field.
   Note that if there are several objects constituting one command (for example, a type secured command with 4 output objects), the same address is copied to all the signals of the command.
   Note also that if the data type of the selected signal is not supported in the protocol of the NCC, a message is shown on the status bar and the address cannot be entered.
10. Select the Signal Handling Attributes using the **Signal Handling Attributes** dialog.
    Note that if there are several objects constituting one command (for example, a type secured command with 4 output objects), the same address is copied to all the signals of the command.
    The available signal handling attributes are presented in Table 3

Table 3: Command signal handling attributes

Data type	RP 570/CPI/Modbus	IEC 60870-5-101/104	DNP 3.0
Binary output	- Project Specific - Inverse Value - Receive as Regulation Command - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Direct Command - None	- Project Specific - Inverse Value - Report Status to Master - None
Analog output Command Type 1	- Project Specific - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - None	- Project Specific - Inverse Value - Report Status to Master - Receive as Single Command - Report Status to Master as Single Precision Float - None
Analog output Command Type 4	- Project Specific - Inverse Value - Receive as Single Command - Controlled using Remote Command Values - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - Receive as Direct Command - Controlled using Remote Command Values - None	- Project Specific - Inverse Value - Report Status to Master - Receive as Single Command - Report Status to Master as Single Precision Float - Controlled using Remote Command Values - None
Analog output Command Type 5	- Project Specific - None	- Project Specific - None	- Project Specific - None
Analog output Command Type 6	- Project Specific - Inverse Value - Send without Synchrocheck - None	- Project Specific - Inverse Value - Receive as Double Command - Send without Synchrocheck - Receive as Direct Command - None	- Project Specific - Inverse Value - Report Status to Master - Send without Synchrocheck - None
Analog output Command Type 8	- Project Specific - None	- Project Specific - Receive as Direct Command - None	- Project Specific - Report Status to Master - Report Status to Master as Single Precision Float - None
Digital output	- Project Specific - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - None	

Table 4: Command signal handling attributes (IEC 61850 and ICCP)

Data type	IEC 61850	ICCP
Binary output	- Project Specific - Inverse Value - Receive as Double Command - Receive as Direct Command - None	- Project Specific - Inverse Value - Receive as Double Command - None
Analog output Command Type 1	- Project Specific - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - None
Analog output Command Type 4	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - Receive as Direct Command - Controlled using Remote Command Values - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - Controlled using Remote Command Values - None
Analog output Command Type 5	- Project Specific - None	- Project Specific - None
Analog output Command Type 6	- Project Specific - Inverse Value - Receive as Double Command - Send without Synchrocheck - Receive as Direct Command - None	- Project Specific - Inverse Value - Receive as Double Command - Send without Synchrocheck - None
Analog output Command Type 8	- Project Specific - Receive as Direct Command - None	- Project Specific - None
Digital output	- Project Specific - None	- Project Specific - Inverse Value - Receive as Double Command - Receive as Single Command - None

An example of a cross-referenced command consisting of four signals (secured command with 4 output objects) is presented in Figure 44. It is an example of a breaker open/close command of a REF 543 unit.

Breaker open select command	S1B1B1	11	BO		3	1	1	S1B	2047		10	1
Breaker close select command	S1B1B1	12	BO		3	1	2	S1B	2047		10	1
Breaker execute command	S1B1B1	13	BO		3	1	3	S1B	2047		10	1
Breaker cancel command	S1B1B1	14	BO		3	1	4	S1B	2047		10	1

Sec4Cmd.tif

Figure 44: Example of secured 4 output object command

An example in Figure 45 shows how a secured command with two output objects is defined for a raise/lower command of a SPACOM tap changer.

Disconn. open command	S1B2D1	13	BO		2	2	1	S1B	99		82	
Disconn. close command	S1B2D1	14	BO		2	2	2	S1B	99		82	Sec2Cmd.tif

Sec2Cmd.png

Figure 45: An example of a direct 2 output object command

An example of a cross-referenced command of five signals (secured command with 5 output objects) is presented in Figure 46.

Breaker open select command	S1B3B1	11	AO	6	1	1	S1B3	2048		204	
Breaker close select command	S1B3B1	12	AO	6	1	2	S1B3	2048		204	
Breaker open execute command	S1B3B1	13	AO	6	1	3	S1B3	2048		204	
Breaker close execute command	S1B3B1	14	AO	6	1	4	S1B3	2048		204	
Breaker cancel command	S1B3B1	25	AO	6	1	5	S1B3	2048		204	

Figure 46: An example of a secured 5 output object command

An example of a cross-referenced command of five signals (Synchrocheck and interlock override) is presented in Figure 47 and in Figure 48.

Breaker open select command	S1B4B1	11	AO	6	1	1		201,202,203,204
Breaker close select command	S1B4B1	12	AO	6	1	2		201,202,203,204
Breaker open execute command	S1B4B1	13	AO	6	1	3		201,202,203,204
Breaker close execute command	S1B4B1	14	AO	6	1	4		201,202,203,204
Breaker cancel command	S1B4B1	25	AO	6	1	5		201,202,203,204

Figure 47: An example of secured 5 output object command with synchrocheck and interlock override

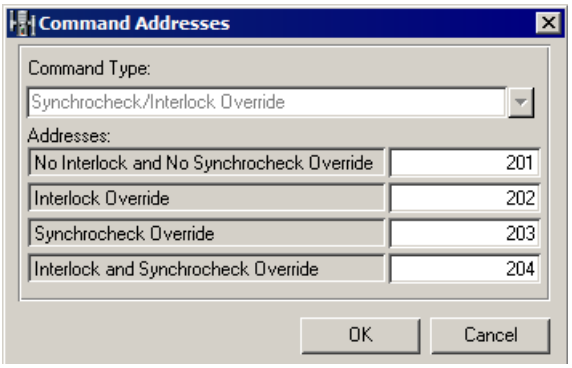


Figure 48: Addresses for synchrocheck and interlock override commands

IEC 60870-5-101/104 regulating step commands are handled as double commands in COM500*i*.

Cut, copy, and pasting cross-references
The delete, cut, copy, and paste functions for commands are the same as for indications, except for the fact that only one command can be cut or deleted at a time.

5.5.15.2 Deleting cross-references

To delete all the cross-references from a command signal:

- If the **None** is selected from the **Type** drop-down menu, all the related command cross-references are deleted. The **General** column is also cleared.
- If a command signal cross-reference contains multiple output objects, the Address, Signal handling, and Type definitions are removed from the cross-referenced indices that are in the same command group.

To delete a cross-reference from a command signal:

1. Click a command signal in the command signals list.
2. Select an NCC column which includes a cross-reference to this signal. This is done by clicking the row under the NCC column in question.
3. Clear the **Address** field.
4. If a signal handling attribute has been defined, select **None** from the **Signal Handling Attributes** dialog.

If there are multiple command signals in the same command group, their addresses and signal handling attributes are deleted automatically.

5. Repeat the address and possible signal handling attribute removal for all the NCC columns that are connected for the selected signal.

To clear the cross-reference using the clear function:

1. Set the **Selecting method** as **Line** from the **Options** menu or the toolbar.
2. Select a signal or multiple signals.
3. Click right mouse button and select **Clear Cross-References**.

5.5.16 Defining parameters

Parameters are defined and viewed on the **Parameters** tab shown in Figure 49. The **Parameters** tab enables the definition of the COM500*i* configuration. These common parameters include the following information:

- Time-out parameters
- Authorization checking parameters
- Miscellaneous parameters

These attributes and their values are saved into a parameter file or a free type object. The parameter file is taken as input for both the command procedures and Signal X-References, when they are started.

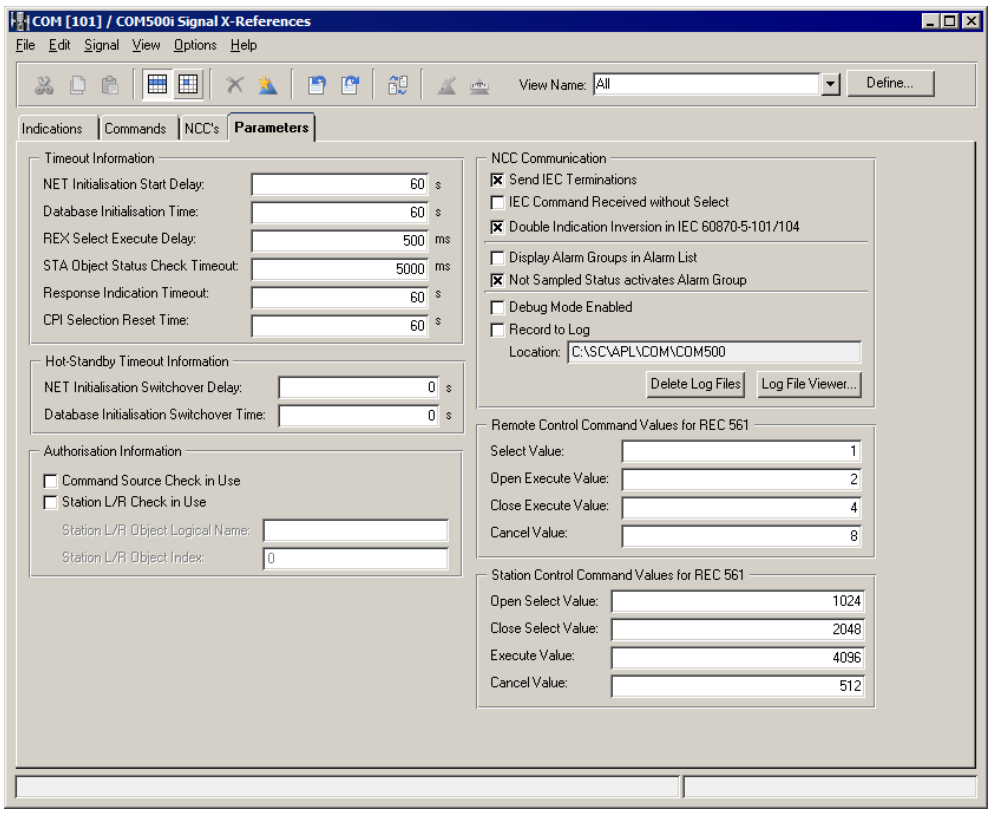


Figure 49: Parameters tab

The parameters listed below can be edited in Signal X-References tool.

Timeout Information

- **NET Initialization Start Delay**

Time (in seconds) after which the initialization of the protocol converters in NET is started. This parameter should be set to be the time from SYS600 start-up to the moment when all the NET lines and stations have been created. The default value is 60 s.

- **Database Initialization Time**
  Time (in seconds) in which the COM500*i* database is considered as initialized, meaning that all the input process objects connected to the process devices have been updated. After this time, COM500*i* sends a Database Initialized message to the NCCs and accepts NCC-specific commands from the NCCs. This parameter should be measured using the actual system with all the devices connected. The default value is 120 s.
- **REx Select Execute Delay**
  The delay (in milliseconds) between the consecutive select and the execute commands sent to a REx device. The default value is 500 ms.
- **STA Object Status Check Timeout**
  Timeout (in seconds) used when checking the state of a station reported as SUSPENDED. The default value is 5000 ms.
- **Response Indication Timeout**
  Time (in seconds) after which the command connected to a response indication is terminated. This is the time waited for an indication connected to a command to be updated, before a negative command termination is sent to the IEC 60870-5-101/104 NCC that sent the command. The default value is 60 s.
- **CPI Selection Reset Time**
  Time (in seconds) after which the internal selection of a CPI command is reset. The default value is 60 s.

Hot stand-by Timeout Information

- **NET Initialization Switchover Delay**
  Time (in seconds) after which the initialization of the protocol converters in NET is started. This parameter should be set to be the time from switchover to the moment when all the NET lines and stations have been set to in use. The default value is 0 s.
- **Database Initialization Time**
  Time (in seconds) after which NET database initialization is started (DNP 3.0 and RP 570) and the Database Initialized message in sent to the NCCs IEC 60870-5-101/104. The default value is 0 s.

Authorization Information

- **Command Source Check In Use**
  This parameter states whether the LIB 500 command source check is in use. When this parameter is set on, the name of the NCC (for example NCC 1) must be on the list of authorized command centers of the application. This list is edited using the **Command Authority** dialog that can be opened from the **LIB 500 Stations** menu. If the COM500*i* application is not built using LIB 5xx or no HSI (station pictures) is used, this parameter must be set. The default value is TRUE. When a stand-alone version of COM500*i* is being used, this selection is dimmed and it is not in use.
- **Station L/R Check In Use**
  This parameter states whether the Station Local/Remote switch check is in use or not. When this parameter is in use, the COM500*i* command procedures check that the value of the Station Local/Remote switch process object corresponds to that of the remote position of the switch. The default value is TRUE.
- **Station L/R Object Logical Name**
  Logical name of the Station Local/Remote switch process object. This parameter must be given if the station local/remote switch process object is not created by LIB 5xx or if a stand-alone COM500*i* is used. Otherwise it should be set to "", an empty text string. The default value is an empty text string.
- **Station L/R Object Index**
  Index of the Station Local/Remote switch process object. This parameter must be given if the Station Local/Remote switch process object is not created by LIB 5xx or if a stand-alone COM500*i* is used. Otherwise it should be set to zero. The default value is 0.

NCC Communication information

- **Send IEC Terminations**
  The user has the option to select whether the command terminations are sent to the master or not. By default, this option is enabled.
- **IEC Command Received without Select**
  It is possible to select whether the commands are accepted without the SELECT command or not. When this option is enabled, all commands must be sent from the NCC without the SELECT command. However, the commands are sent to the relay with the SELECT command, if their command type is Secured in the commands page. By default, this option is not enabled.
- **Double Indication Inversion in IEC 60870-5-101/104**
  The Double Indication Inversion functionality is recommended to be disabled, when IEC 61850 protocol is used in process communication. For example, if IEC 61850 and SPA protocols are used in process communication, the Double Indication Inversion functionality can be disabled, but the Inversed Value Signal Handling must be enabled for SPA double indication. By default, this option is enabled.
- **Display Alarm Groups in Alarm List**
  This option displays state of Alarm Group in Alarm List. By default, this option is disabled.
- **Not sampled status activates alarm group**
  If this option has been selected and a process object of the alarm group is in the not sampled state, the alarm group is sent with an alarming state at start-up.
- **Record to Log**
  The internal message forwarding of COM500*i* can be saved to a log and the contents can be read with a separate tool, which is opened by clicking the **Log File Viewer** button. It is possible to reset the log by clicking the **Delete Log Files** button. This mechanism is meant to be used only in the engineering phase or in cases where traceability is needed.
- **REC 561 Command Values**
  The binary commands are sent as analog values to REC 561. The user can define the values of the commands.

5.5.17 Importing and exporting cross-references

5.5.17.1 Exporting cross-references

Cross-reference signal information can be exported from Signal X-References into delimited text files. This functionality can be used for producing documentation of the cross-references or to modify the existing cross-references by editing the exported files with a spreadsheet or an ASCII editor. It is also possible to create new cross-references. The exported files have the following names and purposes:

COM_XRNCC.xrf;	NCC information
COM_XRGRP.xrf;	Alarm group information
COM_XRIND*.xrf;	Indication information
COM_XRCMD*.xrf;	Command information

To start the Export operation:

1. From the **File** menu in the menubar, select **Export**.
   This opens the **Export** dialog shown in Figure 50.
2. Select one of the alternatives listed below:

- **Indications**. Exported information includes all the attributes included in the column attributes, and for each the NCC address, alarm group number, signal handling attribute and scale.
- **Commands**. Exported information includes all the attributes included in the column attributes, command type, purpose, logical name and index of the response indication, and for each the NCC address and signal handling attribute.
- **NCCs** Exported information includes the station number, protocol, NCC name, comment text, operation mode, the parameter Send Group Alarms at Start-Up and Function Table Definition (FTAB). When NCCs are exported, alarm groups are exported to a separate file including the following information: NCC number, alarm group name, alarm group number, and address.
- **All** This includes all the alternatives listed above.

3. If necessary, select whether the first row in the export file should include column names (by default this is set) or not.
   The names of the export files are fixed and the directory is the PICT directory of the current application.
4. If necessary, change the following parameters:
   - **Field Separator**. This is the character that separates consecutive fields in the export file. The possible choices are comma, semicolon and tabulator.
   - **Text Delimiter** The possible choices are ""(double quote) and none (no delimiter).

   The Text Delimiter none is used, when the cross-reference data is to be exported to external application programs.
   The import functionality of Signal X-Reference does not support the none text delimiter.
5. Click **OK** to start the export.
   During the export of signals a Progress Indicator is shown to display the percentage of exported signals.

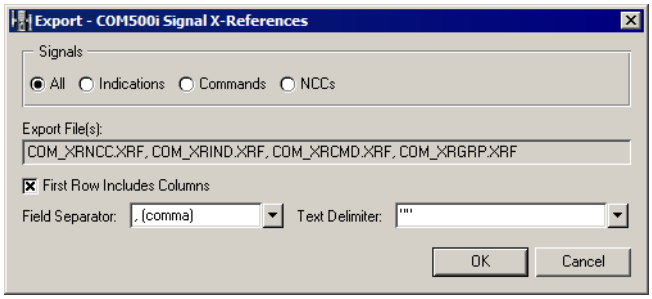


Figure 50: Export dialog

For more information on exporting cross-references with Microsoft Excel, see Section 6.

5.5.17.2 Importing cross-references

Importing cross-references from files modifies the cross-information of the current application. Figure 51 shows the **Import** dialog, which has two modes:

- **Modify existing cross-references**. In this mode, only the NCCs and signals found both in the imported files and in the current application can be modified. No new NCCs, alarm groups or signals are created. The row number in the imported file has to match the value of the Table Index (TI) attribute of the process object in the report database. (If a row is imported, but no matching row number is found in the report database, no cross-reference information is imported from the file.)

This function modifies the existing cross-reference data according to the related information in the rows of the imported file. The value of the Table Index attribute is not modified at all during the import function.

- **Construct new cross-references**. When cross-reference data is imported using this mode, all the existing cross-references are removed from the report database. After that, the cross-references are recreated according to the information in the rows of the imported file. If new NCCs, alarm groups or signals are detected, they are created. The Table Index attribute values are assigned into the found process objects in the report database as well as into the corresponding Action Name (AN) attribute values.

All the cross-reference definitions used by COM500*i* must appear in the imported file. Otherwise, some of the previously defined cross-references are lost.

To start the Import operation:

1. From the menubar, select **File/Import**.
   This opens the **Import** dialog shown in Figure 51.
2. Select one of the following alternatives:
   - **All**
   - **Indications**
   - **Commands**
   - **NCCs**

   The information included in the alternatives is the same as when exporting. The names of the imported files are also the same as when exporting.
3. Select NCCs to be imported:
   - All NCCs
   - NCC 1 - NCC 8

   Import Selected NCC feature has been supported for Indications and Commands.
4. Select the following parameters according the file to be imported:
   - **First Row Includes Columns**
   - **Field Separator**
   - **Text Delimiter**
5. Click **OK** to start the import.
   During the import operation a Progress Indicator is shown to display the amount of imported signals.

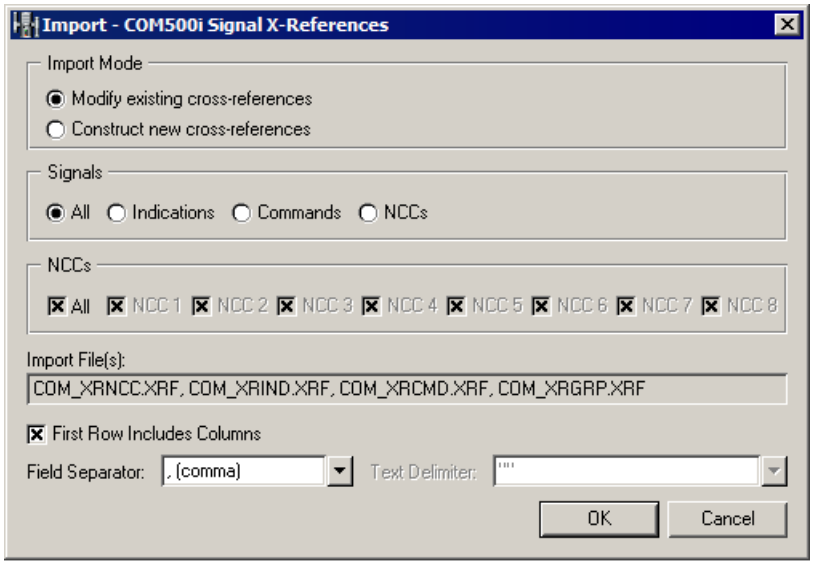


Figure 51: Import dialog

5.5.17.3 Exporting and importing cross-references with Microsoft Excel

If templates need to be constructed in Microsoft Excel, follow these three-stage instructions:

First:

1. Define cross-references for some signals in the Signal X-References tool and select **File/Export**.
   Figure 52 shows the **Export** dialog.

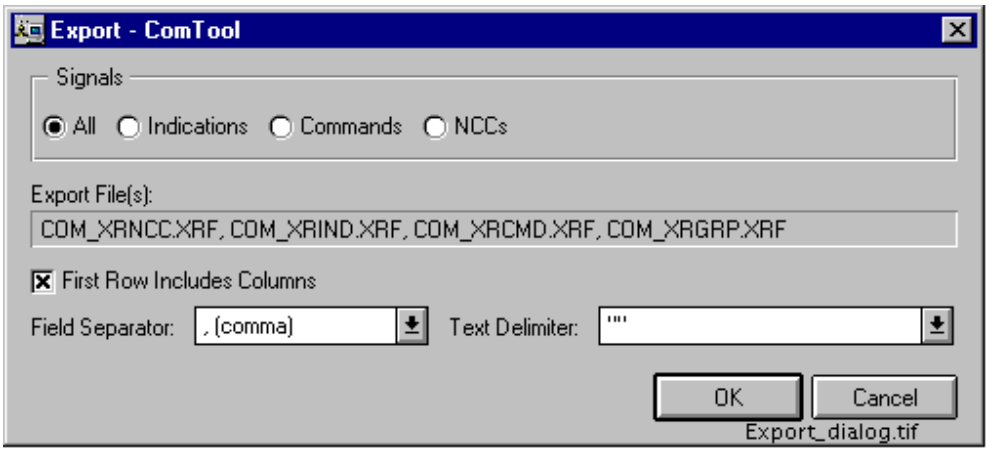


Figure 52: Export dialog of Signal X-References

2. From the dialog, define the Field Separator as TAB (tabulator).

This function generates the COM_XRIND.xrf file into the application's PICT folder and it can be used for importing the template into Microsoft Excel.

In Microsoft Excel:

1. Select **File/Open**.
2. Select files of type All Files (*.*) from the drop-down menu.
3. Navigate to the application's PICT folder and select the COM_XRIND.xrd file.
4. Click **Open**.
   When Microsoft Excel recognizes the file format, it launches the Text Import Wizard - Step 1 of 3 (see Figure 53).

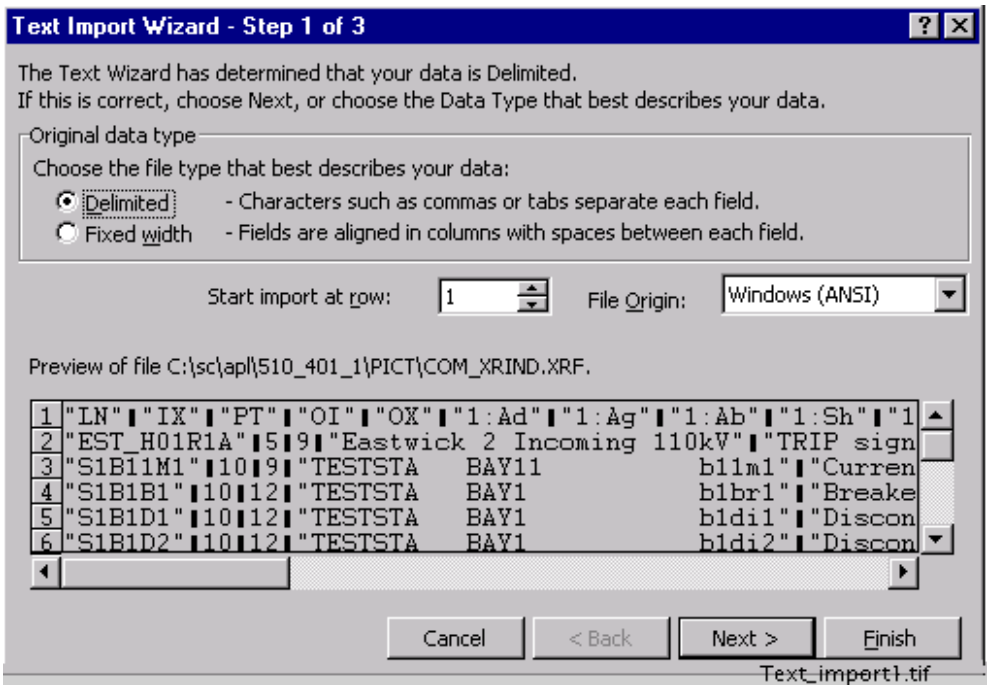


Figure 53: Text Import Wizard - Step 1 of 3 dialog of Microsoft Excel

5. Select the Delimited file type and Start import at row 1.
6. Click **Next**.
   This opens the Text Import Wizard - Step 2 of 3 (see Figure 54).

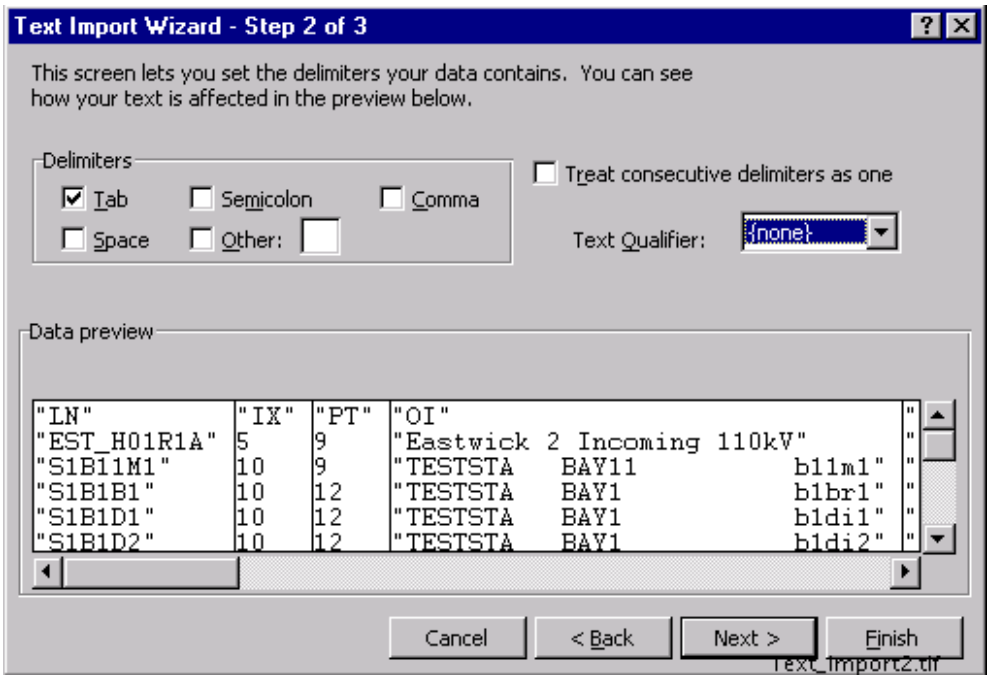


Figure 54: Text Import Wizard - Step 2 of 3 dialog of Microsoft Excel

7. Set the Delimiter to **Tab** and the **Text Qualifier** to **(none)**.
8. Click **Finish**.

The information included in the import file is displayed in Microsoft Excel, see Figure 55. The width of the columns can be adjusted according to the widest column cell by double-clicking at the top of intermediate column headers.

The information in the following columns is surrounded by double quotation marks:

- Logical name (LN)
- Object Identifier (OI)
- Object Text (OX)
- Address (Ad)
- Scale (Sc)

If empty contents are included, it is represented as "".

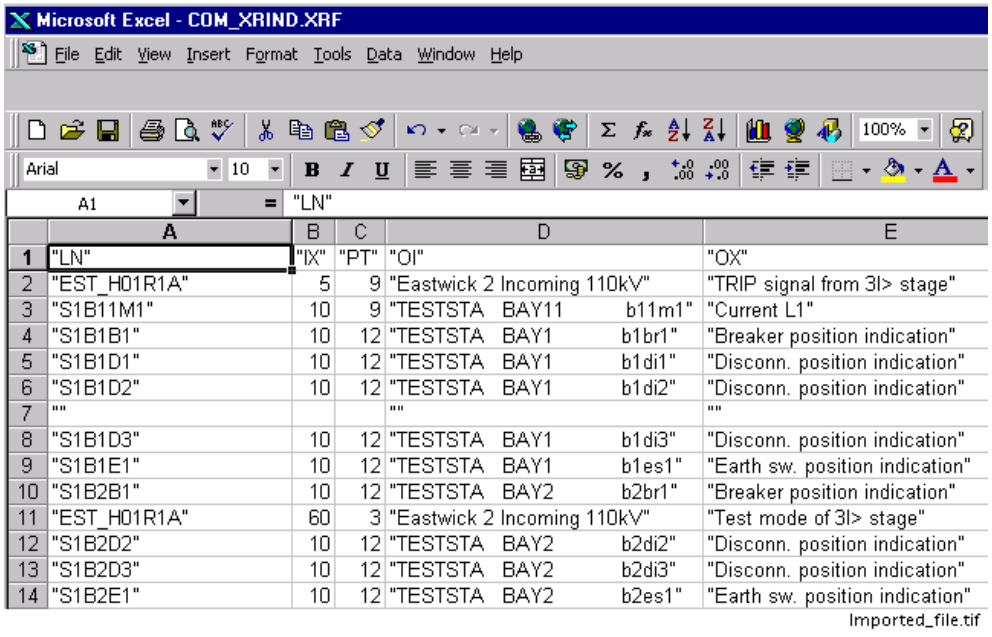
Other columns include integer values. These columns are, for example:

- Index (IX)
- Process Object Type (PT)
- Alarm Group (AG)
- Alarm Bit (AB)
- Signal Handling (SH)

If empty contents are included, the cell is empty.

The Address, Alarm Group, Alarm Bit, Signal Handling and Scale columns appear for NCCs in which the NCC number is used as a prefix together with a colon.

The contents of Alarm Bit (AB) column depends on the Alarm Group (AG) definition in the following way:
If the Alarm Group value has been defined to be for example 1, the first signal of this alarm group is identified with Alarm Bit value 1, the second signal of this alarm group is identified with Alarm Bit value 2 and so on. For the next Alarm Group, the alarm bit values start again from 1. If no alarm group definition exists, there is no need to fill the Alarm Bit column.



	A	B	C	D	E
1	"LN"	"IX"	"PT"	"OI"	"OX"
2	"EST_H01R1A"	5	9	"Eastwick 2 Incoming 110kV"	"TRIP signal from 3I> stage"
3	"S1B11M1"	10	9	"TESTSTA BAY11 b11m1"	"Current L1"
4	"S1B1B1"	10	12	"TESTSTA BAY1 b1br1"	"Breaker position indication"
5	"S1B1D1"	10	12	"TESTSTA BAY1 b1di1"	"Disconn. position indication"
6	"S1B1D2"	10	12	"TESTSTA BAY1 b1di2"	"Disconn. position indication"
7	""			""	""
8	"S1B1D3"	10	12	"TESTSTA BAY1 b1di3"	"Disconn. position indication"
9	"S1B1E1"	10	12	"TESTSTA BAY1 b1es1"	"Earth sw. position indication"
10	"S1B2B1"	10	12	"TESTSTA BAY2 b2br1"	"Breaker position indication"
11	"EST_H01R1A"	60	3	"Eastwick 2 Incoming 110kV"	"Test mode of 3I> stage"
12	"S1B2D2"	10	12	"TESTSTA BAY2 b2di2"	"Disconn. position indication"
13	"S1B2D3"	10	12	"TESTSTA BAY2 b2di3"	"Disconn. position indication"
14	"S1B2E1"	10	12	"TESTSTA BAY2 b2es1"	"Earth sw. position indication"

Figure 55: Imported file in Microsoft Excel

To import the signals into Signal X-References tool after the definitions have been typed into the Microsoft Excel sheet:

1. Determine the signals to be imported by selecting the rectangular area of the signals and cross-reference definitions.

This means that each column and row in the sheet is selected.

2. Copy the information to the clipboard by selecting **Edit/Copy**.
3. Paste the copied information to the Notepad application by selecting **Edit/Paste**in Notepad.
   Notepad is used to store the information to be imported into Signal X-References.

   When information is pasted from the clipboard, the tabulator is always used as a field separator.

4. Save the file to the application's PICT folder and name it COM_XRIND.xrf.

   Remember to verify that all the information is included. The indication cross-references are constructed into COM500*i* according to the information on that file. Previous cross-references are removed from COM500*i* when Import Mode Construct new cross-references is used and new cross-references are constructed according to the import file.

5. Select **File/Import**in Signal X-References to open the **Import ComTool** dialog (see Figure 56).

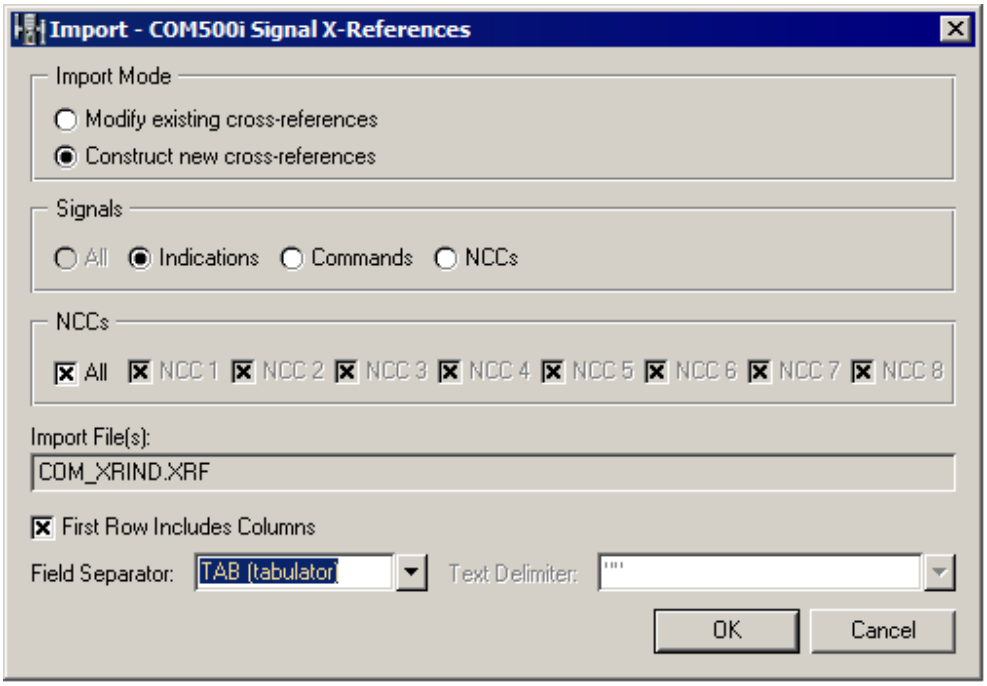


Figure 56: Import dialog of Signal X-References

6. Define the Import Mode as Construct new cross-references, Signals as indications, and Field Separator as TAB (tabulator).
7. Click **OK**.
8. When the importing into Signal X-References has finished, refresh the view.
   This is done in order to update the displayed information in Indications according to the imported data (see Figure 57).

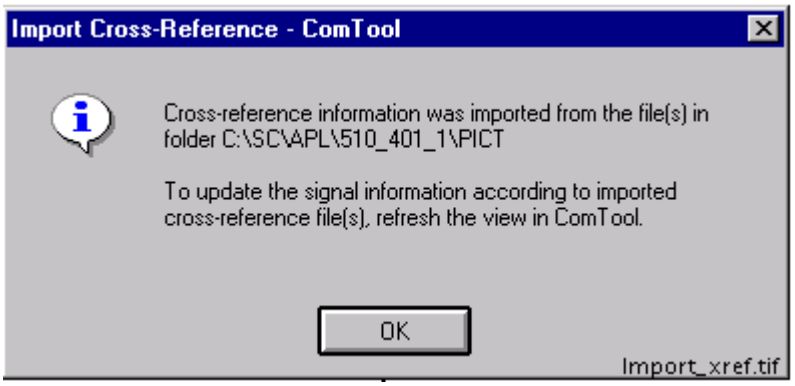


Figure 57: Dialog box indicating that importing of cross-references is ready

5.5.18 Converting cross-references from Microsoft Excel to COM500*i* cross-references

Typically, the signal lists contain all the signals related to the information between COM500*i* and the NCC. These lists contain the cross-references used for indication and command addresses between the substation (COM500*i*) and the upper level system (Network Control Center). These lists are mainly maintained outside COM500*i* and their contents may already be specified by the upper level system. Due to the great number of different signal lists (with different amount of sheets used, different column names, and so on), the generic functionality to convert the signal lists' cross-reference files (.XRF) used by COM500*i* has been introduced in Signal X-References.

This functionality can be used to convert the indication signals: binary inputs, double binary indications, analog inputs and digital inputs. Regarding the command addresses, the signal engineering needs to be done in Signal X-References only. At the moment, there is no similar functionality in the product for handling the command cross-references outside COM500*i*. Typically, the requirement to support signal lists is more important for indications. This is due to the huge amount of signal related information concerning the indications compared to commands.

5.5.18.1 Cross-Reference File Converter

To start the Converter select **File/Convert** from the Signal X-Reference menubar, or click the appropriate toolbar button  .

Several settings have been introduced to identify the cross-reference information during the conversion.

The Converter consists of five different tabs. The first four tabs are meant for each different indication type: Binary Inputs, Double Binary Indications, Analog Inputs and Digital Inputs. The fifth tab contains the Destination Parameters.

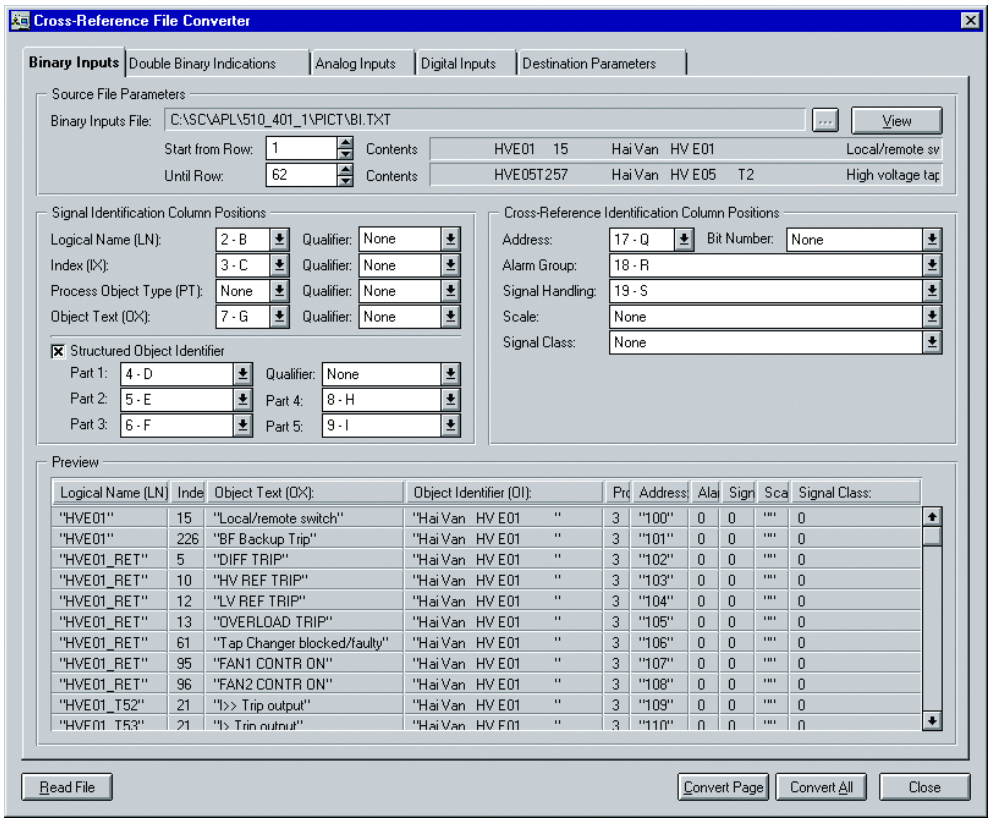


Figure 58: Main view of Converter

The following sections are common to the first 4 tabs: Source File Parameters, Signal Identification Column Positions and Cross-Reference Identification Column Positions. The functionality of **Destination Parameters** tab is described in its own Destination Parameters section later on.

5.5.18.2 Source File Parameters

Source File Parameter specifies the file name and its location used as a source signal list file for each different indication type.

To change the file name and location:

1. Clicking the ... button to open File Chooser.
2. Click **View**.
   The Converter opens the source file in SCIL Viewer. See Figure 59.

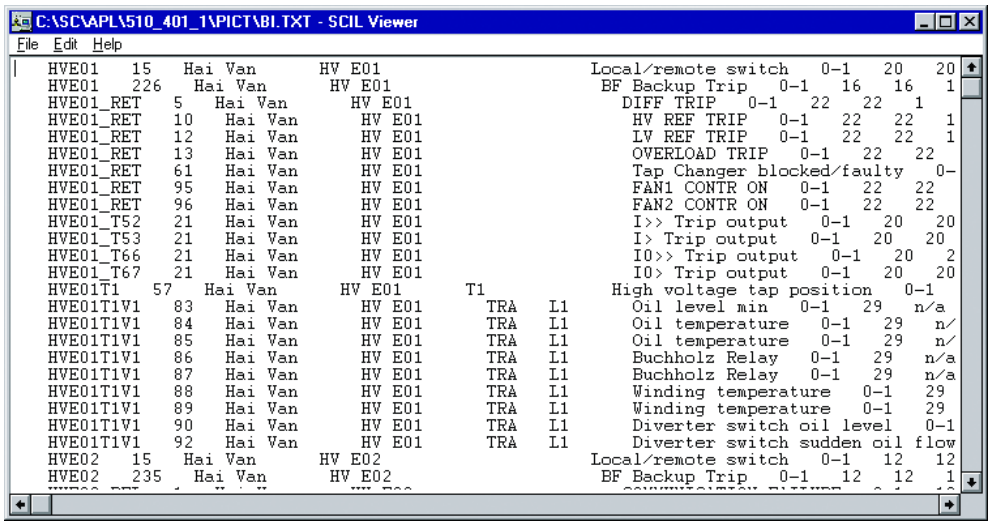


Figure 59: Binary inputs source file in SCIL Viewer

In SCIL Viewer, the contents of the source file is presented as usual in text editor applications. This quality is needed in the next step, in which the row numbers used as a source for the information during the conversion need to be specified.

Start from Row specifies the first row number to be handled. The default value is 1. If the source file contains a heading, the **Start from Row** value should be set to 2.

Until Row specifies the last row that is handled during the conversion. The default value is the last row of the source file.

Both **Start from Row** and **Until Row** values are read during the start-up of the Converter and when the source file name is changed via File Chooser.

5.5.18.3 Signal Identification Column Positions

This information is needed, when the signals are identified by their name and type during the conversion. Typically, the signal is identified based on the combination of Logical Name and Index attributes. The signal type is based on the value of Process Object Type. The column positions for LN and IX are mandatory. Column position PT is not needed, if the indication type specific tabs are used to convert signals of the same type, that is Binary Inputs tabs for binary input signals and Double Binary Indications tabs for double binary indication signals, and so on. If the column position has not been specified and the user reads the source file into some of the indication tabs, the Converter asks the user to verify that all the signals in the source file are the same type as selected tabs.

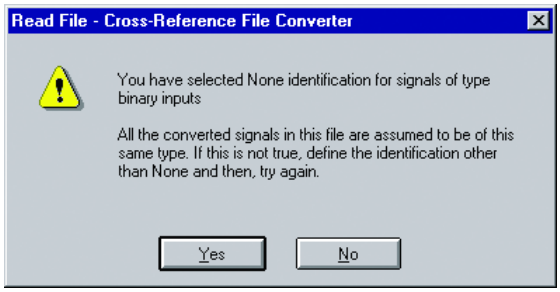


Figure 60: Reading source file without specifying PT column

If only one source file contains all the different indication types, then the conversion can be handled in one tab only. In this case, the column position for the PT attribute must be specified.

Other signal identification columns and Object Text and Object Identifier are meant for recognizing the signals in the Preview and the functional purpose of each row in the resulted file, if they are later transferred to another system. Regarding the Object Identifier, the Preview can also combine the contents of Object Identifier, if it has been structured into several columns (Substation, Bay, Device, and so on). During the conversion, these columns are merged into one column in the result file. The values of these columns are not applied to the process database when importing the converted file in Signal X-References later. This is because the import functionality is only meant for importing the cross-reference information.

The column position of each attribute is recognized through introducing its column position in the original file. Column positions are identified as numerical and alphanumerical information, for example 2 - B, where the number 2 refers to the second column from the left side of the file, and the letter B refers to the B column in the source application.

If the data in the source columns has been qualified with "" characters, the qualifier " should be selected for the appropriate attribute. As a default, None is applied as a qualifier.

5.5.18.4 Cross-Reference Identification Column Positions

These columns specify the actual cross-reference information which is used in COM500*i* signal routing. Typically, each cross-referenced signal contains at least the Address. In the case of NCC connection, the address information may consist of the Address only, or also include the Bit Number. The values for Address and possible Bit Number should be integer values that represent the address used between COM500*i* and the NCC. As a default, the Bit Number is selected as type None, that is not defined in the converted file. It should be noted that Bit Number should be located in its own column position in the file to be converted.

In addition to the address column positions, the Alarm Group and Signal Handling can be included in the converted file. The value for Alarm Group should be an integer value which represents the number of groups related to the NCC in question, whereas the value for Signal Handling should be an integer value which represents a different number related to the signal process object type and which is supported in COM500*i* signal routing.

The Scale column position should be defined in case the analog input signals are included in the converted file. The value for Scale should be text representing the Scale object in the COM500*i* process database. Note that the Cross-reference Converter does not check the existence of the Scale object in the COM500*i* process database. If the Scale object is not found during the signal routing, the message "COM500*i* Scale object is not found" is displayed in the Notify Window. The message also contains information about the Scale object used and the signal identification.

For the DNP 3.0 and IEC 60870-5-101/104 protocol it is also important to include the Signal Class column position. The value for Signal Class should be an integer value representing the number of signal classes applicable in communication.

As a default, both the Scale and Signal Class column positions are set as None, i.e. not defined.

5.5.18.5 Preview

The result of reading the file to be converted is produced in the Preview area. The Preview displays the information in the same order as it appears in the converted file.

The following table describes the functions of the buttons in this tool:

Table 5: Functions of the tool

Button	Function
View	Reads the input file in SCIL Viewer. It is possible to Print and Copy to clipboard the contents of opened file.
Read File	Reads the input file in selected tab and displays the result in Preview area
Convert tab	Converts the input file in selected tab and produces the file in output folder with specified name and format
Convert All	Converts all the input files in tabs and produces the file in output folder with specified name and format
Close	Closes the Cross-Reference File Converter tool

5.5.18.6 Destination Parameters

The conversion Output File Name and location can be defined on this tab. As a default, the conversion result is written in the PICT folder of running application.

If there is a need to specify another file, click the ... button in the Destination Parameters. It opens the File Chooser dialog. Use the dialog to define another output file name and location.

When the Append to File option is set (default), the conversion result is appended to the end of the file, if it exists. If Append to File is not set, the conversion result overwrites the file, if it exists.

On this tab, it is also possible to select the output file format. There are two possible formats: Version 3.0 and Version 4.0. Version 3.0 is used in COM500 product version 3.0 and related Service Packs running in the MicroSCADA 8.4.3 environment. Version 4.0 is used in COM500 product version 4.0 and related Service Packs running in the MicroSCADA 8.4.4 or later environments.

5.5.18.7 Conversion Example

The following is an example of how to convert the indication cross-references from the tabular sheet to the file, and from there to convert it using Cross-Reference File Converter to be later imported into Signal X-References.

1. Open the tabular sheet in the application including the indication signals.
2. Select **File/Save As** from the application.
3. Save the file in the PICT folder or the running application and use the Text type (Tab delimited) with .txt extension.
4. If the different sheets are included in each indication signal type, each sheet must be saved separately. Use file names such as: AI.txt, BI.txt, DB.txt and DI.txt, because these are the default names used by the Cross-Reference File Converter when converting input files.
5. Start Signal X-References tool.
6. Open the Cross-Reference File Converter tool by selecting **File/Convert**.
7. Specify the Source File Parameters, Signal Identification Column Positions and Cross-Reference Identification Column Positions according to each file to be converted. Click **Read File** to verify the result of column positions.
8. Define the definitions in the **Destination Parameters** tab according to the environment.
9. For each signal type, click the **Convert** tab or **Convert All** to convert the cross-references from .txt files to .xrf files.
10. Close the Cross-Reference File Converter tool by clicking **Close**.
11. Select **File/Import** from Signal X-References.

12. Set the option Construct new cross-references, First Row Includes Columns and Field Separator to comma (,).
13. Click **OK**.
14. When Importing is finished, refresh the view in Signal X-References by re-selecting the active view name from the **View** menu.

5.5.19 Printing cross-references

It is possible to print cross-references from Signal X-References tool. This function can be used to produce documentation of the application to be sent to the customer or the supplier of the Network Control Center system.

5.5.19.1 Configuring a printer

A printer can be connected in the following ways:

- A printer connected directly to computer's USB, serial or parallel port.
- To a LAN using a network printer or a printer server.

For more information on printer configuration, see SYS600 Installation and Administration Manual.

5.5.19.2 Printing

When printing from Signal X-References, the user can select one of the several options. These options are listed below with information on their content.

NCC printout

- Name of the NCC
- Description of the NCC
- Protocol of the NCC
- Station number of the NCC
- Alarm group names
- Alarm group addresses

Indications printout

- Logical name and index
- Unit number and address
- Object identification and object text
- NCC address
- Signal handling attributes
- Alarm Group number
- Scale
- Signal class

Commands printout

- Logical name and index
- Unit number and address
- Object Identification and Object Text
- Command type, purpose and command group
- NCC address
- Signal handling attributes

Parameters printout

- Name of the parameter
- Value of the parameter
- Unit of the parameter

Cross-reference information can be printed by selecting **File** and **Print** from the menubar. This opens the **Print** dialog shown in Figure 61. The print selection can be:

- **Current page**. When this option is selected, the printout contains the active notebook page, which can be indications, commands, NCCs, or parameters.
- **All pages**. This option includes all the cross-reference information in the printout.
- **Selected pages**. The printout is made according to further selections.
- **Include Empty Signals** When this option is selected, the printout includes cross-referenced and non-cross-referenced signals.

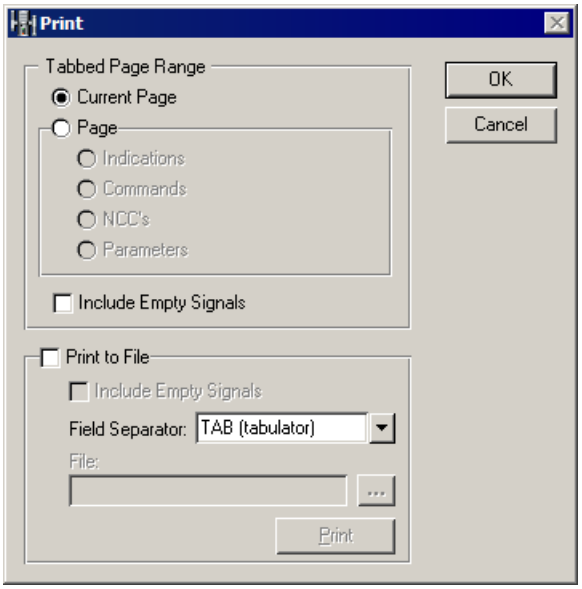


Figure 61: Print dialog

All the printed pages contain a header, a number of columns and column titles, and a footer. The title is centered in the header of each paper. Column titles are the names of the column attributes. Under each column, there is cross-reference information printed for each signal. Certain fixed signals are allocated to each printed page. The footer contains the name and number of the application, the page number, and the total number of pages (see Figure 62).

OX	LN	IX	PT	1:Ad	1:Ag	1:Sh	1:Sc	1:Cl	2:Ad	2:Ag	2:Sh	2:Sc	2:Cl
Bay selection on monitor	ESTH03_BAY	21	AI	1									
Topological busbar coloring	ESTH03_L1	254	AI	2					17				
Topological busbar coloring	ESTH03_L2	254	AI	3					18				
Topological busbar coloring	ESTH03_L3	254	AI	4					10				
Current L1	ESTH03_MEC	10	AI	5					11				
Active power P	ESTH03_MEC	20	AI	17					6				
Reactive power Q	ESTH03_MEC	21	AI	18					19				
Voltage U12	ESTH03_MEV	16	AI	10					20				
Residual voltage U0	ESTH03_MEV	19	AI	11					21				
Frequency f	ESTH03_MEV	24	AI	6					22				
Breaker position indication	ESTH03_Q0	10	DB	19		32			7		80		
Breaker device control block	ESTH03_Q0	15	BI	20		32			23		80		
Breaker open interlocked	ESTH03_Q0	16	BI	21		32			24		80		
Breaker close interlocked	ESTH03_Q0	17	BI	22		32			8		80		
Cause of interlocking	ESTH03_Q0	18	AI	7					9				
Breaker selection on monitor	ESTH03_Q0	19	AI	23					12				
Breaker command event	ESTH03_Q0	20	BI	24					13		80		
Disconn. position indication	ESTH03_Q1	10	DB	8		32			14		80		
Disconn. device control block	ESTH03_Q1	15	BI	9		32			25		80		
Disconn. open interlocked	ESTH03_Q1	16	BI	12		32			26		80		
Disconn. close interlocked	ESTH03_Q1	17	BI	13		32			15				
Cause of interlocking	ESTH03_Q1	18	AI	14					27				
Disconn. selection on monitor	ESTH03_Q1	19	AI	25					28				
Disconn. command event	ESTH03_Q1	20	BI	26					16				
Disconn. position indication	ESTH03_Q2	10	DB	15		32			29		80		
Disconn. device control block	ESTH03_Q2	15	BI	16		32			30		80		
Disconn. open interlocked	ESTH03_Q2	16	BI	27		32			31		80		
Disconn. close interlocked	ESTH03_Q2	17	BI	28		32			32		80		
Cause of interlocking	ESTH03_Q2	18	AI	29					33				
Disconn. selection on monitor	ESTH03_Q2	19	AI	30					34				
Disconn. command event	ESTH03_Q2	20	BI	31		32			35				
Disconn. position indication	ESTH03_Q3	10	DB	32		32			36				
Disconn. device control block	ESTH03_Q3	15	BI	33		32			37				
Disconn. open interlocked	ESTH03_Q3	16	BI	34		32			38				
Disconn. close interlocked	ESTH03_Q3	17	BI	35		32			39				
Cause of interlocking	ESTH03_Q3	18	AI	36					40				
Disconn. selection on monitor	ESTH03_Q3	19	AI	37									
Disconn. command event	ESTH03_Q3	20	BI	38									
Earth sw. position indication	ESTH03_Q9	10	DB	39									
Earth sw. selection on monitor	ESTH03_Q9	19	AI	40									
Tap position	ESTH03_T1	10	AI	41									

Figure 62: Example of printed List of Indication Signals

5.5.19.3 Page setup

It is possible to change the settings of the printed page by selecting **Cross-Reference/Page Setup**. This opens the **Page Setup** dialog shown in Figure 63. The page settings include the width of the top, bottom, left, and right margins. By default these settings are: 20, 20, 20 and 20.

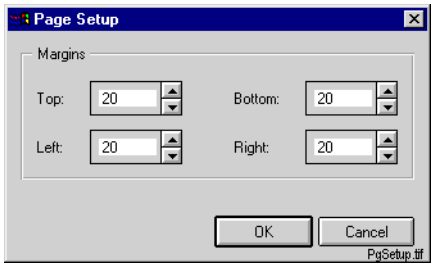


Figure 63: Page Setup dialog

5.5.19.4 Print setup

Printer settings can be changed by selecting **Cross-Reference/Print Setup**. This opens the **Print Setup** dialog shown in Figure 64. This dialog contains options for printer name, properties, paper size, source, and orientation.

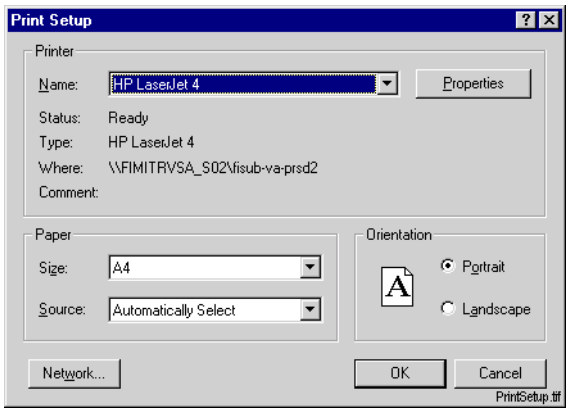


Figure 64: Print Setup dialog

5.6 Using COM500*i*

5.6.1 Diagnostics dialog

The **Diagnostics** dialog is a tool for displaying communication diagnostics for each NCC and system information in COM500*i*. Selected signals can be displayed here for diagnostic purposes. It is possible to open from the **COM500i** tab of Tool Manager or Tool Launcher. It is also possible to change the font used under the **Options** menu.

A separate tab is included in the **Diagnostics** dialog for each NCC of COM500*i*. The **Diagnostics** dialog is shared by two different notebook tabs. The first tab shows the Link Layer diagnostics and the other one shows the Alarm Groups diagnostics.

5.6.1.1 Link Layer diagnostics

Each NCC tab includes general information found from the cross-reference tables and diagnostics information.

General information displays the NCC's name and description, the station number of NCC and the protocol of NCC. Diagnostics information displays diagnostic counter indices, names and their values in this NCC. Depending on the protocol used for NCC, the number of diagnostic counters may be different according to the following tables below.

Table 6: NCC diagnostics counters IEC 60870-5-104, RP 570 and DNP 3.0

NCC (IEC 60870-5-104)	NCC (RP 570 Slave)	NCC (DNP 3.0)
Transmitted Telegrams	Transmitted Telegrams	Transmitted Telegrams
Failed Transmissions	Failed Transmissions	Failed Transmissions
Transmitted Timeouts	Timeout Errors	Transmitted Commands
Transmitted I (Information) Format Messages	Received Telegrams	Transmitted Replies
Transmitted S (Supervisory) Format Messages	Parity Errors	Received Telegrams
Transmitted U (Unnumbered control function) Format Messages	Overrun Errors	Parity Errors
Received I Format Messages	Redundancy Errors	Overrun Errors
Table continues on next page		

NCC (IEC 60870-5-104)	NCC (RP 570 Slave)	NCC (DNP 3.0)
Received S Format Messages	Framing Errors	Check Sum Errors
Received U Format Messages	Buffer Overflow Errors	Framing Errors
Received Messages / Telegrams		Buffer Overflow Errors
TCP Connect Count		TCP Connect Count [1)1)]
TCP Accept Count		TCP Accept Count [1)]
TCP Close Count		TCP Close Count [1)]
Duplicates and Losses		
Buffer Overflow Errors		

1) These are valid only for DNP 3.0 with TCP/UDP.

Table 7: NCC diagnostics counters IEC 60870-5-101, IEC 61850 and ICCP

NCC (IEC 60870-5-101)	NCC (IEC 61850)	NCC (ICCP)
Transmitted Telegrams	Number of Suspended Associations	Number of Suspended Associations
Failed Transmissions	Number of Suspension	Number of Suspension
Transmitted Commands	Transmitted Commands	Transmitted Device Commands
Transmitted Replies	Transmitted Command Replies	Transmitted Device Replies
Received Telegrams	Received Commands	Received Device Commands
Parity Errors	Received Command Replies	Received Device Replies
Overrun Errors	Received Data Messages	Received Data Messages
Check Sum Errors	Received System Items	Received Data Items
Framing Errors	Unknown Objects	Received System Data Items
Buffer Overflow Errors	Command Timeouts	Received Data not Defined
	Line in Commandmd Buffer Overflows	Device Command Timeouts
	Line in Data Buffer Overflows	Line in Command Buffer Overflows
	Line in Reply Buffer Overflows	Line in Data Buffer Overflows
	Device in Command Buffer Overflows	Line in Reply Buffer Overflows
	Device in Data buffer Overflows	Device in Command Buffer Overflows
	Command out Buffer Overflows	Device in Data Buffer Overflows
	Reply out Buffer Overflows	Device in Reply Buffer Overflows
	Lost Messages	Command out Reply Buffer Overflows
	Transmission Fails	Reply out Buffer Overflows
	Transmitted IAL Messages	Received Unaddressed Data Items

For the NCC of protocol type CPI or Modbus, only the **Alarm Groups** tab is displayed, not the **Link Layer** tab.

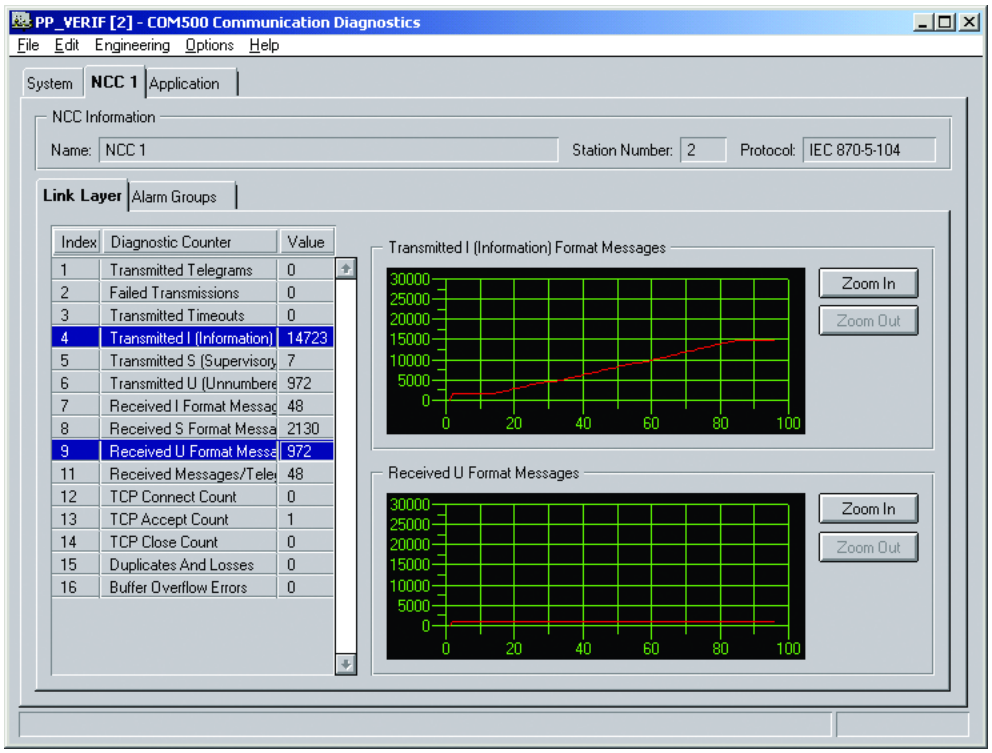


Figure 65: NCC Diagnostics dialog

On the right side of each NCC tab, there are two graphs that display the counter values in a plot diagram (see Figure 65). The diagnostic counters, which are displayed in these graphs, can be changed by selecting two appropriate counter names on the list. The first selected counter is displayed in the upper graph and the second in the lower graph.

Click on the first counter and then hold down the CTRL key while clicking on the second counter. By default, the first two counters are selected, when the **Diagnostics** dialog is opened. The counter values are refreshed every five seconds.

If there is no need to update the graphs, unselect the last active counter name on the list by pressing the space key. If the counter value exceeds its maximum value 30 000, the value restarts from 0.

For each plot graph there is a zoom function that allows zooming a selected plot area in each graph.

To zoom in a graph, click the **Zoom In** inside the appropriate graph, and select the zoom area by pressing and releasing the mouse button for a rectangular area in this graph. When the mouse button is released, the view is zoomed to the selected area.

To zoom out the selected graph click **Zoom Out**(see Figure 66).

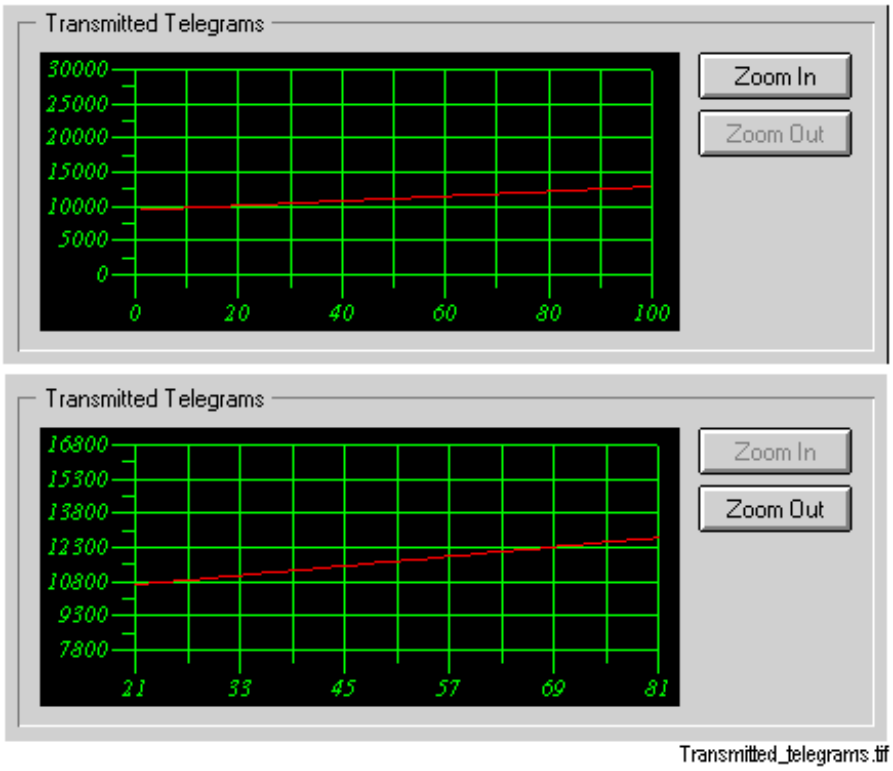


Figure 66: Zoom Functionality of plot graph

5.6.1.2 Alarm Group diagnostics

The Alarm Groups functionality means that a set of signals can be grouped to certain alarm groups. When one or several signals in an alarm group enter the alarm state, the alarm group itself also enters that state. When this occurs, COM500*i* sends the alarm value to the NCC. When all the signals in an alarm group return to the normal state, the alarm group itself also returns to that state. In this case, COM500*i* sends the normal value to the NCC. Note that if there are alarm state changes for some signals which do not change the alarm group state, COM500*i* does not send the same alarm group state to the NCC. However, all the alarm group changes are sent when communication is established between COM500*i* and an IEC 60870-5-101/104 NCC. This is one configurable feature on the NCC tab. Group alarms are also sent when a general interrogation occurs.

The **Alarm Groups** dialog (see Figure 67) shows the following information:

- **Group** specifies the alarm group number. The group number is displayed, when signals are connected to alarm groups in Signal X-References.
- **Name** displays the alarm group name.
- **Address** displays the alarm group address.
- **On** displays the alarm group state. When the alarm group is in the alarming state, the checkbox is set under the **On** column. When the alarm group is in the normal state, the checkbox is not set under the **On** column.
- **Time Stamp of Last Signal Update** displays the time stamp for when a signal in this alarm group in COM500*i* has been updated the last time without changing the alarm group state.
- **Time Stamp Sent to NCC** displays the time stamp for when COM500*i* has sent the alarm group state change to the NCC.

The **Alarm Groups** dialog can also be directly accessed from the Signal X-References, when an NCC is selected on the **NCCs** tab and the **Diagnostics** button is clicked in the Alarm Information panel.

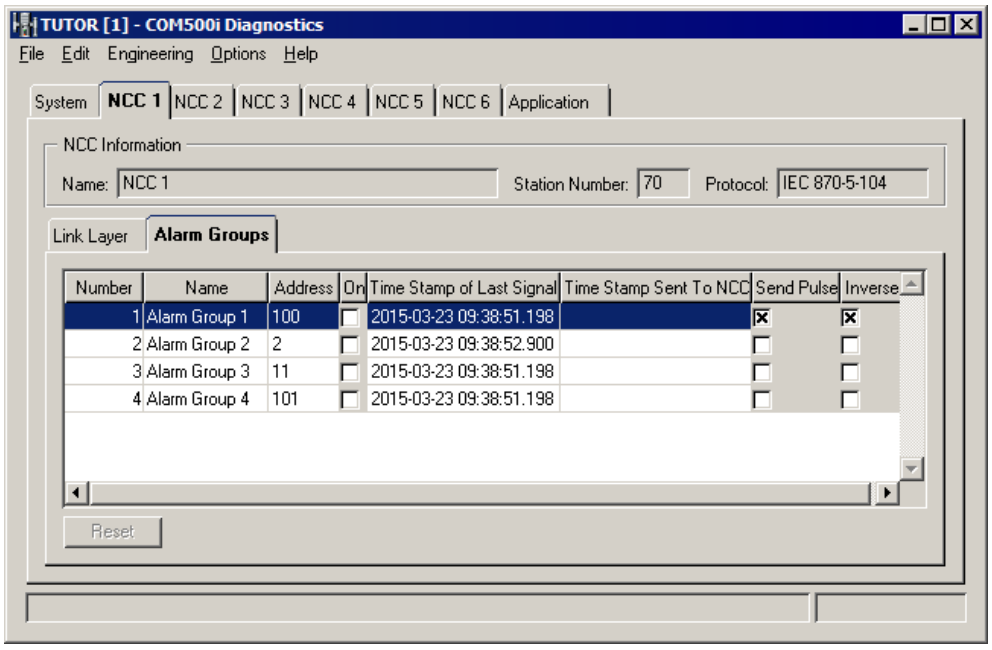


Figure 67: Alarm Groups dialog

By double-clicking an alarm group shows all the alarm signals in a specific group. The name of the alarm group whose alarm signals are displayed is identified in the title of the dialog (see Figure 68).

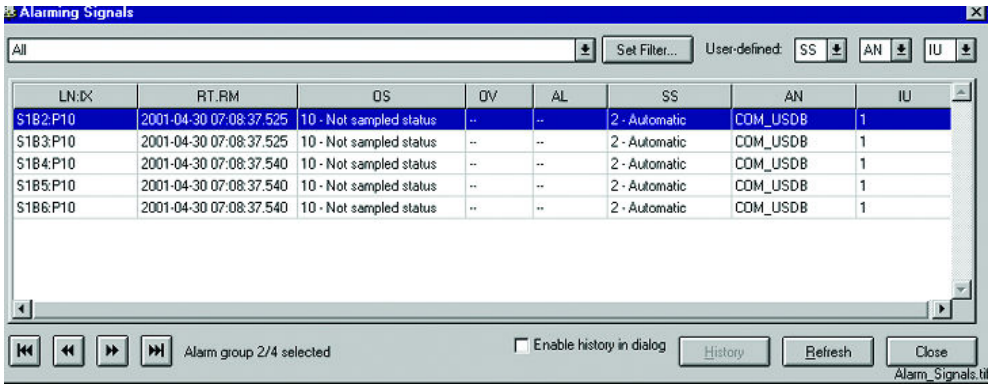


Figure 68: Alarm signals dialog

The different fields and buttons in the **Alarm signals** dialog are described in Table 8.

Table 8: Fields and buttons in Alarm signal dialog

Filter	The signals which are displayed can be defined using filters. The default filter name is "All". When it is selected after filtering, all the signals are shown. Predefined user filters can be chosen from a drop-down menu. A filter which is stored in a user profile is set by clicking the **Set Filter** button. By default, the number of filters is 10.
User-defined attributes	The User-defined attributes defines the attributes to be displayed on the three rightmost columns of the table (for example SS, AN and IU). User-defined attributes can be chosen from a drop-down menu. Selected attribute values are stored and restored. If the History function is enabled, changing the user-defined attributes clears old values from that column.
Navigation	The dialog is navigated by using the buttons in the lower left corner. These arrow buttons can be used to select one of the alarm groups (see below) whose alarm signals are displayed. Information on what group is active at that time is shown in the dialog.
Table continues on next page	

	Select the first alarm group.
	Select the previous alarm group.
	Select the following alarm group.
	Select the last alarm group.
History function	By setting the "**Enable history in dialog**" checkbox, history data of shown signals is collected. The 10 last changes of every signal are stored as long as the dialog stays open. Double-click a signal or select it and press the **History** button, to see the stored history. By default, the first rows in the signal history are shown. When the dialog is closed, history data is cleared.
Refresh	Click the **Refresh** button to check if new signals have gone to the alarming state in a group. Signals that are no longer alarming disappear from the list. History data is cleared for all signals.
Close	Click the **Close** button to close the signal dialog.
Alarm Signals Table	All the signal information is shown in a table. When there is a change in the signal state, the view is updated. There are 8 columns in the alarm signal view: 5 fixed and 3 user-defined columns. In the first fixed column, the logical signal names and their indexes are shown. The second column shows the time stamp of the last update. The third column shows the object status. The fourth column shows the object value, if it is readable (for example the value has been received from the process). The fifth column shows the alarm state for signals.

The condition for alarm signals inside an alarm group can be defined in the **Filter** dialog (see Figure 69). The condition can be constructed by using certain dialog items (for example Attribute and Value) and text fields, as illustrated in the figure below. It can also be directly entered as SCIL condition into the **Filter** text field.

In the example below, all the alarm signals which have a logical name with prefix S1B and index 10 are included in the condition.

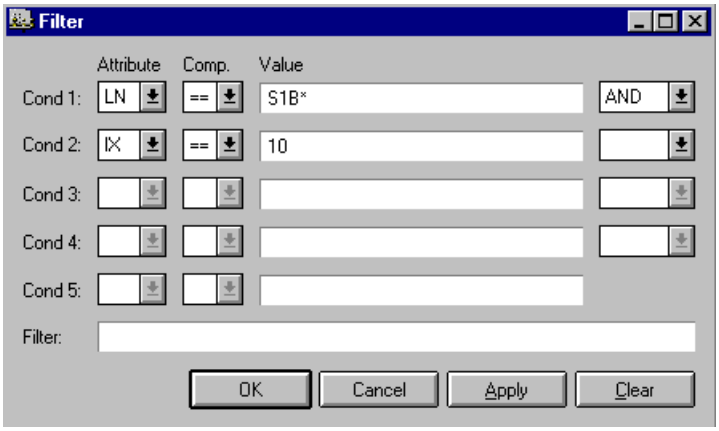


Figure 69: Filter dialog

5.6.1.3 Application tab

The diagnostics of the parallel queues, process event queues, and running objects are shown on the **Application** tab (Figure 70). These diagnostics display how the COM500*i* functionality increases the system load in the application.

Parallel Queues display the maximum length of parallel queues together with the length of the waiting executions in the parallel queues. The percentages of these values are shown as well.

The maximum length shown in the application is not an absolute limit, because the queue can continue growing.

Process Event Queues display the maximum number of process events that can be in the queue for an event channel activation. The number of event channel start up commands from the process in the event channel queue are shown together with the percentage of these values.

Running Objects displays the contents of common time and event channel queues and the contents of parallel queues. For COM500*i*-specific parallel queues, the contents are described more precisely in the descriptive text column. For example, Parallel queue 2, Command Terminations. The object column contains the names of the command procedures or data objects currently under execution in REPR queues. For example C COM_GENINT. The character C identifies the command procedure and D identifies the data object.

The objects in each of these queues are updated at an interval of 1 second. If an object is found in a queue, the type and name of the object are shown. For example, C COM_GENINT.

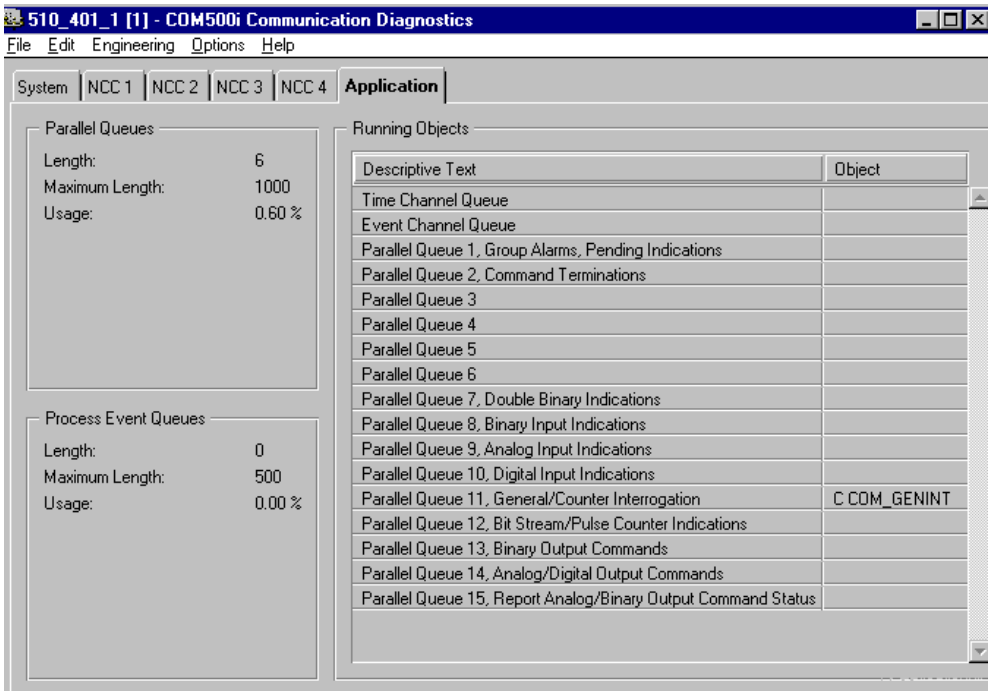


Figure 70: Application tab

5.6.2 Traceability

Usually there is a need to trace signals during the factory and site acceptance tests in the COM500*i* project. Especially if a stand-alone COM500*i* is used, this is the only easy way to test signals. During these tests, it is verified that each signal included in the system is transmitted and received in the correct way. Traceability in COM500*i* means that signal values with related attributes are displayed and updated as event-based in the **Signal Diagnostics** dialog. The set of signals to be traced is selected in Signal X-References. The selection can consist of signal rows inside a certain rectangular area or individual signal rows. Individual signal rows can be selected by pressing the CTRL key and clicking the rows that should be included in the selection. Both the indication and command signals can be selected. Each row included in the

Signal Diagnostics is updated, when a value update related to that signal occurs there. The related attributes to be displayed can be selected from the relevant drop-down menus. The history related to the signal changes is stored, when Enable history in the dialog is set. History is collected as long as the **Signal Diagnostics** dialog is displayed.

When using a signal dialog from a tool other than COM diagnostics, the following functions work in a different way:

- Navigation
  Shows information on a selected signal. The navigation buttons which can be used to scroll back and forth are active.
- Table
  There are 11 columns and the first column is a combination of the logical signal name and index.

Figure 71 shows the **Signal Diagnostics** dialog:

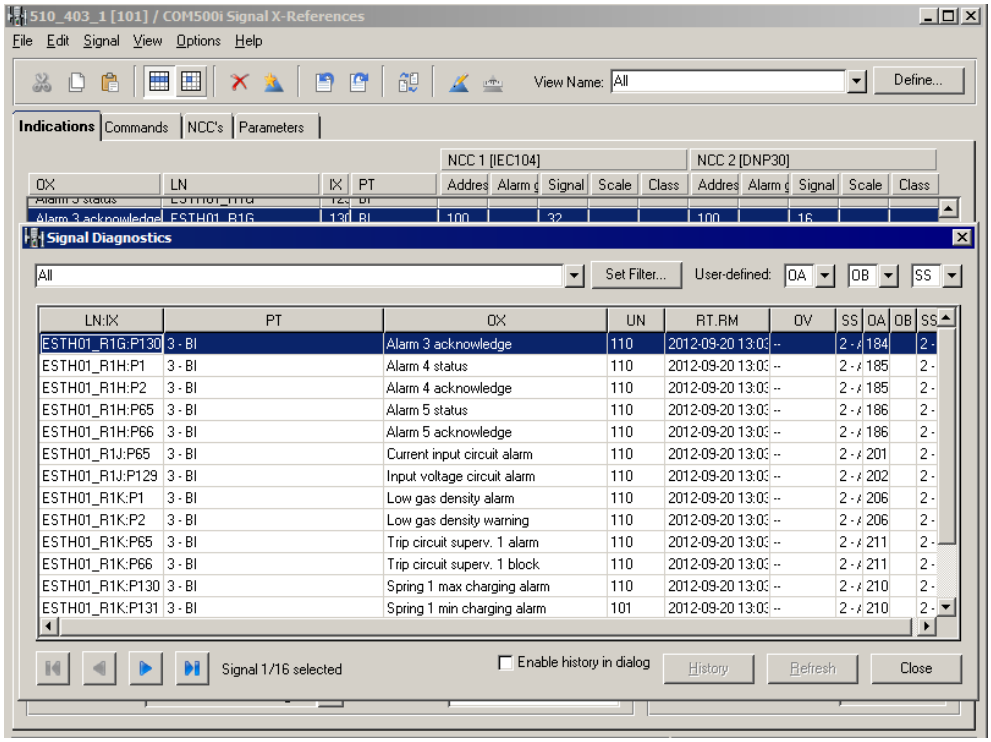


Figure 71: Signal Diagnostics dialog

The History viewer (Figure 72) shows the last 10 states of the selected signal. The name of the signal is shown in the dialog title. The attribute values and the row number are shown in the columns. If there is no value in a cell, the column attribute has been changed during the history gathering.

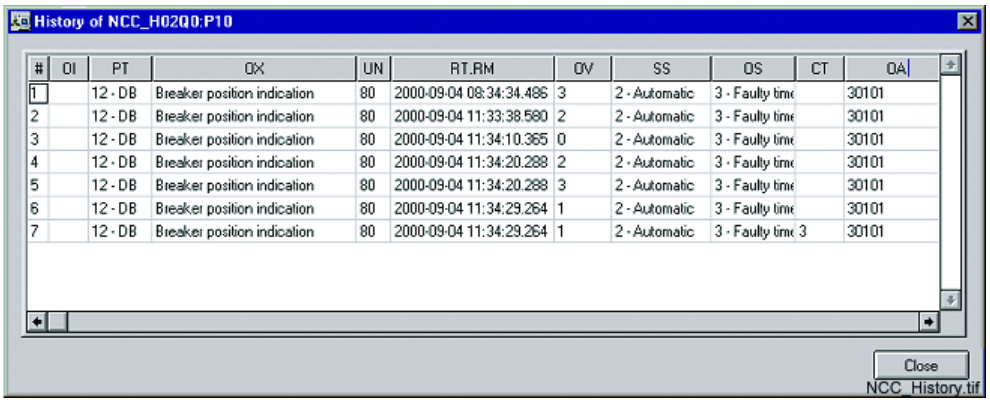


Figure 72: History viewer dialog

The **Refresh** button is in the disabled state. The button is enabled only when Signal diagnostic is opened from COM500*i* Diagnostics.

5.6.3 Command authorization

When a command is received, COM500*i* checks the command authority of the NCC. The command authority is determined by the following factors:

- Station Local/Remote switch must be in the remote or out of use position, if the parameter **Station L/R Check In Use** is set.
- NCC must be given the authority to make commands.

These factors can either be used simultaneously or individually.

In case a command is not authorized, a negative command confirmation is sent to the NCC, if the IEC 60870-5-101/104 protocol is used (see Section 6). If DNP 3.0 is used, the command is confirmed with status information that the outputs are in the local state.

When a stand-alone COM500*i* is used, the following user interface is displayed on the **Parameters** tab of Signal X-References (see Figure 73). In this case, the selection **Command Source Check In Use** is not applicable.

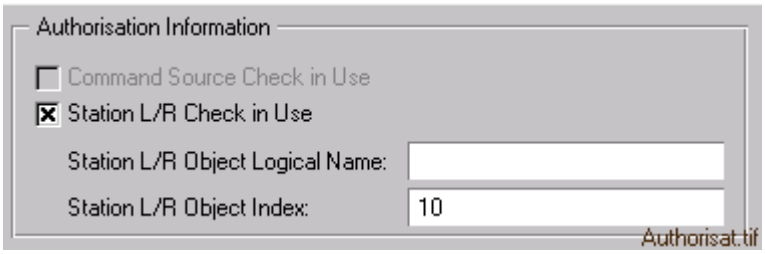


Figure 73: Authorization Information dialog

5.6.4 Recording signal routing to log files

If the **Record to Log** check box is set on the **Parameters** tab of Signal X-References, it is possible to log data forwarding between the COM500*i* database and the NCC system. The generated log files can be cleared from the Signal X-References using the **Delete Log Files** button. Logging is stopped when **Record to Log** check box is not set.

The log mechanism saves data from every transaction between the COM500*i* command procedures and the PC-NET. Some internal handling is also written to the log. Runtime log is written to free type objects. The number of these free type process objects is 10, each of

which consist of 1000 lines. When these objects are written, the log mechanism writes text files to a COM500*i*-specific path in the application directory. This path is shown on the **Parameters** tab of Signal X-References. Up to 10 Files can be generated, com_log1..10.log.

This log can be read with a specific tool, which is started from the **COM500i** tab of Tool Manager (see Figure 5), the **COM500i** tab of Tool Launcher (see Figure 6) or the **Parameters** tab of Signal X-References by using the **Log File Viewer** button (see Figure 74).

Figure 74: Access to Log File Viewer

When the tool is opened, it creates a new log file from the current runtime logs, com_log.log. The tool shows on the notebook tabs of all the defined NCCs, data sent to them and data received from them with some default data:

The main dialog of the COM500*i* Log Viewer tool is a logged events navigator with several notebook tabs, containing a table, toolbar and menu (see Figure 75).

The base functionality of the tool is displaying information from the log files with or without filtering. All the information is separated into notebook tabs. Each tab, except the last one, contains data related to a specific NCC. The last tab contains data which is not included in any of the NCC specific notebook tabs. Logged events are listed by tabs instead of showing all the information at the same time. Navigation buttons allow moving to the next and previous tabs, as well as to the beginning and to the last tab. The number of rows on a tab is configurable. The main table on each notebook tab contains only the most important columns, and detailed information about a record can be seen by double-clicking it. The table rows are highlighted with different colors depending on the signal type. The tool includes the option to customize filters for the following fields:

- Destination addresses range
- Process Object name (text filter)
- Date and time (for example all the signals from a certain period of time)
- Data source station number
- Command procedure

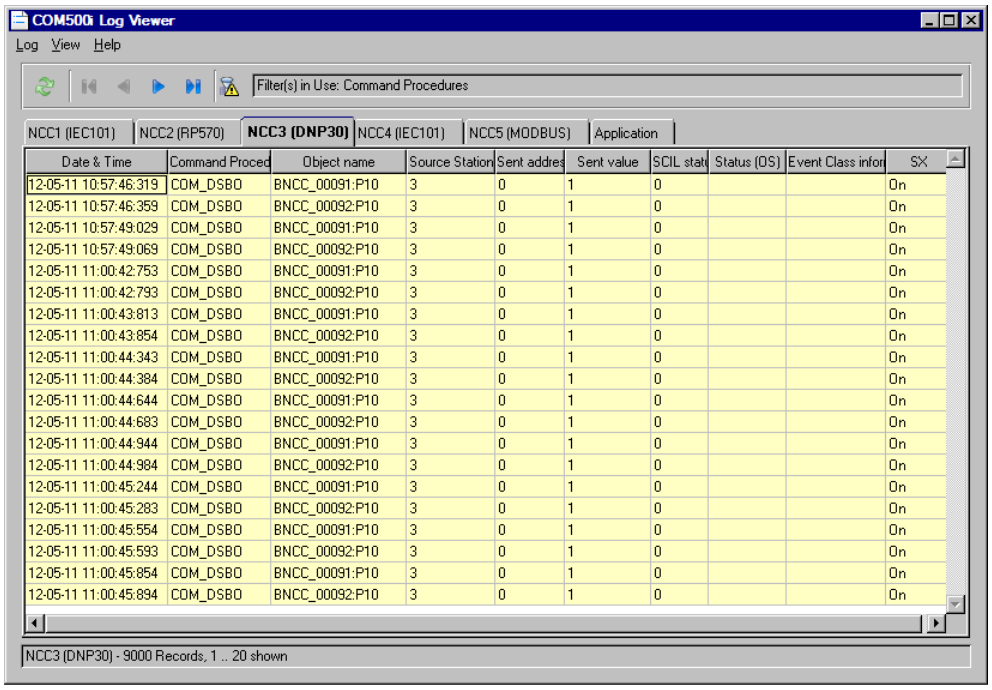


Figure 75: Main dialog of COM500i Log Viewer

There are four standard columns on each notebook tab:

- Date & Time from the triggered process object
- The COM500*i* command procedure used
- Triggered process object name and index
- Possible SCIL status (or text description)

Six columns are visible only on NCC specific notebook tabs:

- Source STA object number
- Sent NCC address
- Sent value
- Status of process object during signal routing
- Event Class
- State text (SX)

The rows in the tables are highlighted with different colors depending on the signal type: successfully sent upstream data is light green, downstream data is light yellow and other data is white in color. The lines with SCIL error are highlighted with red.

The user can switch between different NCC by selecting the appropriate notebook tab. The main table on each tab has a limited row count. If the number of possible log records is greater than this value, then full log data list is shown in tabs. The user can navigate the full list by using the toolbar buttons – First, Previous, Next, Last. The maximum row count is specified by the Table_PageSize parameter of the [LViewer] section in the COMLViewer.INI file in the user parameters directory. The default page size is 20. The status bar at the bottom of the table shows information on the total number of log records found for the currently selected NCC and currently displayed line numbers.

In case the log files are changed while the tool is running, it is possible to refresh log data by pressing the toolbar button **Refresh** or by selecting **Log/Refresh**.

When the user tries to set the Record to Log option in Com Tool, the following caution dialog appears on the screen (see Figure 76). If **Yes** is clicked, the Record to Log function is set, otherwise it is not.

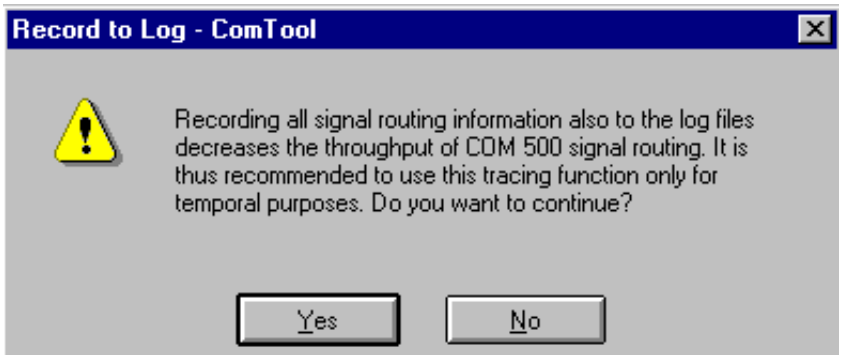


Figure 76: Caution dialog of Record to Log function

Using the logging mechanism generates more load to the system. In large systems logging cannot be used over a long period of time. The user must check from the diagnostic tool that the usage of the event channel does not increase when logging is set to use. It must also be remembered that analog indication points can generate a lot of updates and some delta values should be used in that case (for example, analog inputs in IEC 103 devices).

Detailed event information can be viewed in a separate dialog (see Figure 77), which is accessed by double-clicking the desired row in the main table. Another way to do it is to select the desired row in the main table, click the right mouse button and select **Detail** in the shortcut menu which appears on the screen.

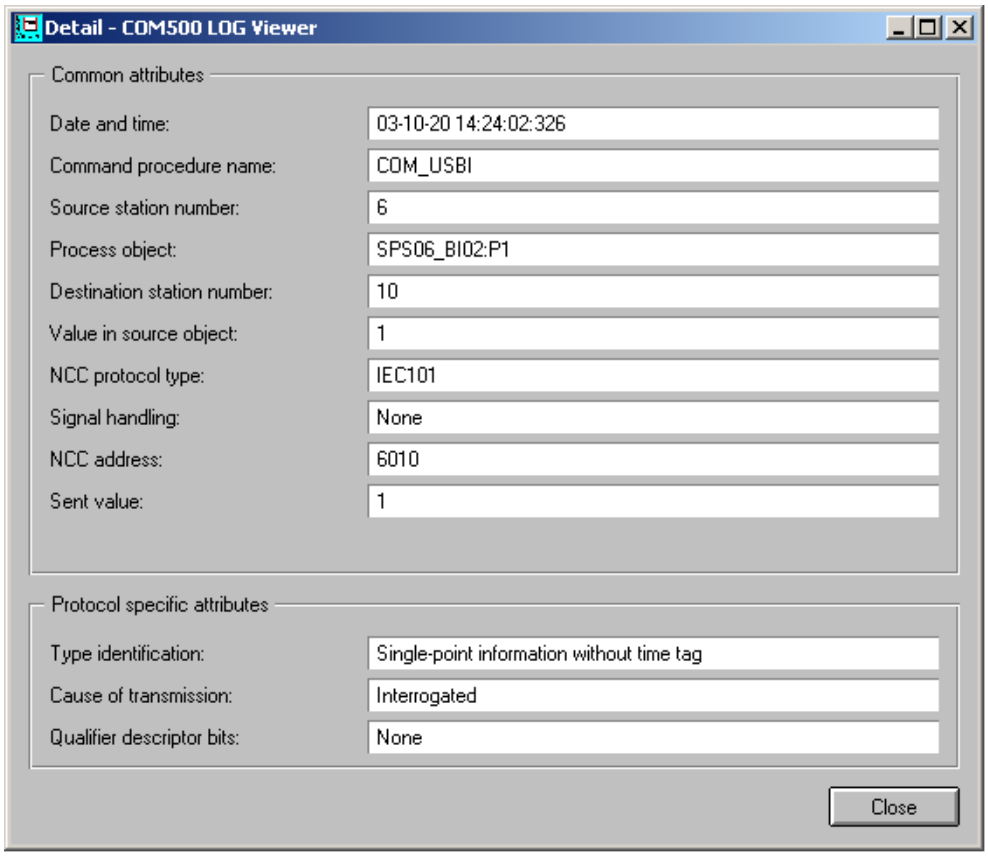


Figure 77: Details dialog

This dialog contains two groups of attributes: Common attributes, which are common for all NCC protocols, and Protocol specific attributes. When an alarm group is displayed in the **Detail** dialog, the alarm group name is included in the Common attributes group. When an analog input process object is displayed in the **Detail** dialog, the attached Scale object is included in the Common attributes group.

Only successfully sent upstream or downstream data can be viewed in this dialog.

The user can also define different filters to search specific data, up to 11000 lines logging data.

Logged events listed in the tables can be filtered by different attributes. The **Filter** dialog is accessible from the toolbar button or by selecting **View/Filter Events**. Figure 78 appears on the screen after the option is selected:

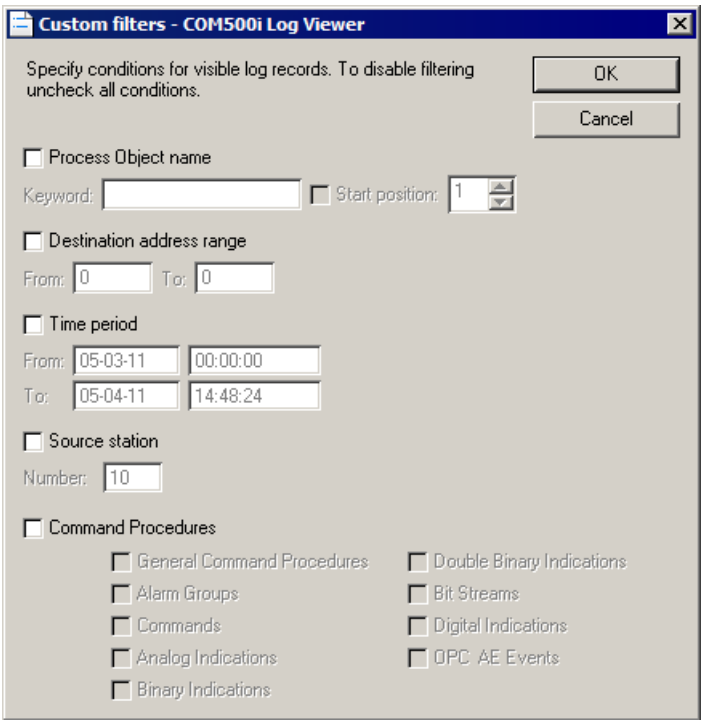


Figure 78: Custom Filters dialog

The user can specify any combination of the following filters:

- Process Object name, with a keyword which should be included in the process object name at a specified position
- Destination address range
- Time period
- Source station number
- Command procedure.

All the filters are saved in COMLViewer.INI file in the user parameters directory.

In the tool, there is also the possibility to fill the filter dialog automatically with attributes of a specific event. For that option the user should select the row which contains the desired event and select **Use as Filter...** from the shortcut menu (which is displayed by clicking the right mouse button).

The currently selected filters are displayed in the status bar, which is placed on the right side of the toolbar in the main dialog.

Section 6 Technical description

This section describes the functionality, design, and configuration of COM500*i*.

6.1 Hardware requirements

See SYS600 Installation and Administration Manual.

6.2 Available protocols

The protocols supported in COM500*i* are shown in Table 9. The master protocols are used in process communication and the slave protocols in upper level communication.

In addition to the supported protocols, the Communication Programming Interface (CPI) is also listed. CPI is in fact not a protocol, but rather an interface that can be used for implementing new protocols in the SYS600 environment. If protocols that are not listed below need to be used, there is a possibility to program the protocol conversion using CPI. The programming is done using the C language.

CDC-II slave is deprecated from version 10.0 since it used special hardware not supported in latest Windows versions and modern computers.

Table 9: Supported protocols in COM500i

Protocol	Type	Support
SPA	Master	PC-NET
LAG 1.4 (LON)	Master	PC-NET
IEC 60870-5-101	Master and slave	PC-NET
IEC 60870-5-103	Master	PC-NET
IEC 60870-5-104	Master and slave	PC-NET
IEC 61850	Master	IEC 61850 OPC Server
IEC 61850	Server	IEC 61850 Server
ANSI X3.28 - AB (ABB)	Master	PC-NET
RP 570	Master and slave	PC-NET
DNP 3.0 (Subset 3)	Master and slave	PC-NET
CPI	(supports both)	PC-NET
Modbus	Master and slave	PC-NET and external software
ICCP	Client and Server	ICCP Link
OPC DA	Client	External OPC Data Access Client / Base system
OPC DA	Server	Base system
OPC A&E	Client	Base system
OPC A&E	Server	Base system

6.2.1 Information about protocols

Some of the protocols listed above are master protocols, which means that the protocol is used for communicating with process devices, whereas slave protocols are used for communicating with upper level systems. The following sections provide reference documentation on the protocols supported in COM500*i*. In addition to the documents mentioned below, the SYS600 documentation can be used as reference.

6.2.1.1 SPA

SPA-Bus Communication protocol V2.4.

6.2.1.2 LAG 1.4 (LON)

LON Application Guidelines for Substation Automation. Version 1.4. This is the ABB standard for using LON as a substation automation protocol.

6.2.1.3 RP 570 master and slave

This protocol is used for communication between the RTU 200 family process devices and the SYS600 systems. Protocol Specification: RTU PROTOCOL 570 and 571 (RP 570 and RP 571). Doc. id: 1KSE 300000-VW. M. Vänskä 95-10-18.

6.2.1.4 ANSI X3.28 master

This protocol is used for communication with Allen-Bradley PLC devices. It is also used for communication with SRIO 1000M and SRIO 500M devices. Protocol Specifications:

- PLC-2-Family RS-232-C Interface Module. Doc. Id: 1771-6.5.8
- SRIO 1000M and 500MDoc. Id: 34 SRIO 100M 2 EN1 D

6.2.1.5 IEC 60870-5-103 master

This protocol is used for connecting the SYS600 system to protection and control devices. Protocol Specification: International Standard IEC 60870-5-103.

6.2.1.6 IEC 60870-5-101 master and slave

Protocol specification: INTERNATIONAL STANDARD IEC 60870-5-101.

6.2.1.7 IEC 60870-5-104 master and slave

Protocol specification: INTERNATIONAL STANDARD IEC 60870-5-104.

6.2.1.8 IEC 61850 Master

This protocol is used for connecting the SYS600 system to protection and control devices. For more information about using this protocol, see SYS600 IEC 61850 Master Protocol (OPC) and OPC Data Access Client user’s guides.

6.2.1.9 IEC 61850 Server

For more information about using this protocol, see SYS600 IEC 61850 Server.

6.2.1.10 DNP 3.0 master and slave

The following protocols describe the DNP 3.0 protocol:

- DNP 3.0 DATA LINK LAYER version 0.02 (P009-0PD.DL)
- DNP 3.0 APPLICATION LAYER version 0.03 (P009-0PD.APP)
- DNP 3.0 DATA OBJECT LIBRARY version 0.02 (P009-OBL)
- DNP 3.0 TRANSPORT FUNCTIONS version 0.01 (P009-0PD.TF)
- DNP 3.0 SUBSET DEFINITIONS version 2.00 (P009-01G.SUB)

6.2.1.11 Modbus master and slave

Protocol Specification: Modicon Modbus Protocol Reference Guide.

6.2.1.12 ICCP Client and Server

For more information about using this protocol, see SYS600 IEC 60870-6 (ICCP) Protocol documentation.

6.2.1.13 OPC DA Server

OPC DA Server can be used for connecting the SYS600 system to an upper level system, which contains the OPC DA Client. For more information on using this protocol, see SYS600 OPC Server.

6.2.1.14 OPC DA Client

OPC DA Client can be used for connecting the SYS600 system to protection and control devices which contain the OPC DA Server. For more information on using this protocol, see SYS600 External OPC Data Access Client, SYS600 Application Objects, and SYS600 System Objects.

6.2.1.15 OPC A&E Server

OPC A&E Server can be used for connecting the SYS600 system to an upper level system, which contains the OPC A&E Client. For more information, see SYS600 Application Objects and SYS600 System Objects.

6.2.1.16 OPC A&E Client

OPC A&E Client can be used for connecting the SYS600 system to protection and control devices which contain the OPC A&E Server. For more information, see SYS600 Application Objects and SYS600 System Objects.

6.2.2 CPI

CPI software can be used to implement both the master and slave protocols. The CPI library contains functions to send and receive messages. It also contains functions to pack and unpack data. The CPI-based communication software and COM500*i* communicate through the TCP/IP network.

When using a CPI program with COM500*i* for upper level communication (slave protocol), the following rules apply:

- The communication system object attribute interface of the device is as in RP 570 slave
- The process database interface is of RTU 200 type
- System messages are as in RP 570 slave

The CPI interface is designed to support connections to several applications in a base system or in several base systems. All applications in one base system can be reached using the same TCP/IP socket.

When using CPI, SYS600 Communication Programming Interface documentation should be used as a guideline. In the CPI projects it is recommended to contact Hitachi Power Grids.

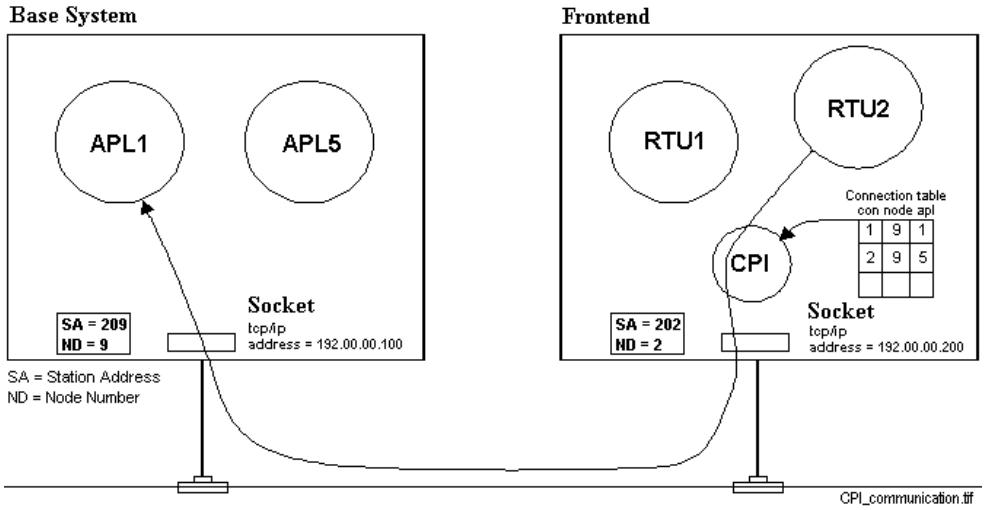


Figure 79: Process unit can communicate with base system through CPI

6.3 Data flow

Data flow through COM500*i* is handled by the signal routing mechanism which consists of the following parts:

- Cross-reference mechanism, which consists of cross-reference tables created and maintained by Signal X-References.
- Parameter files
- Command procedures

The other parts of the signal routing are:

- System message handling, application, and system command handling
- Command authority check
- Group alarm handling

6.3.1 Indications

6.3.1.1 Mechanism and data types

Indications (input process objects) receive data from the process devices. This data is then sent to the NCCs by COM500*i*. Data flow from a process device to one or several NCCs is shown in Figure 80.

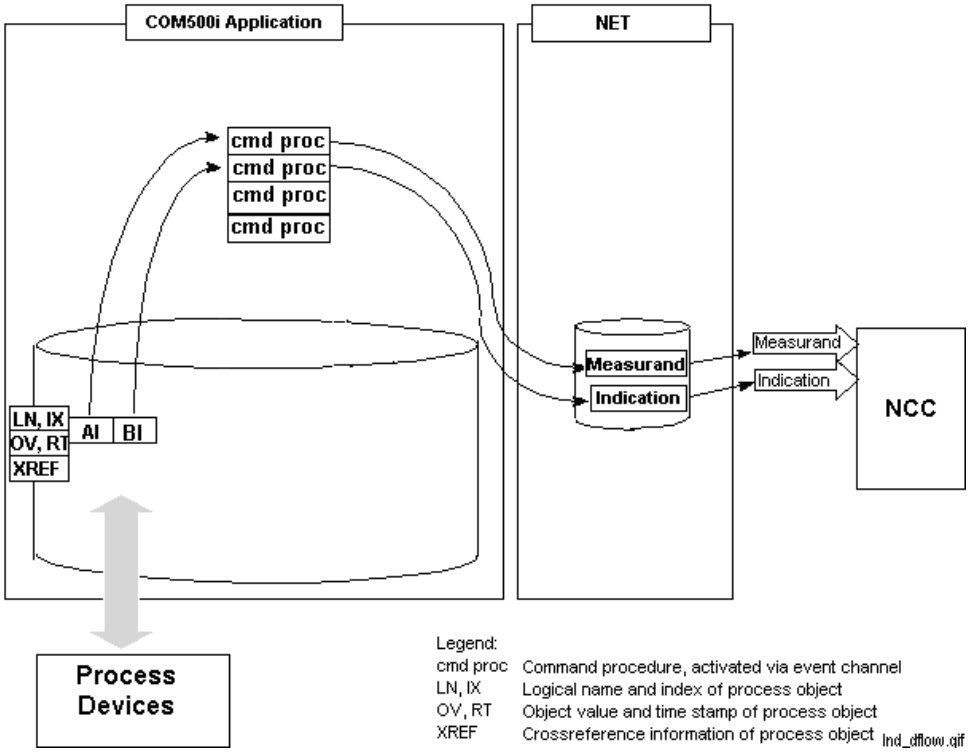


Figure 80: Indications from process devices to NCC

The following input data types are supported in COM500*i*:

- Binary input
- Double binary input
- Digital input
- Analog input
- Pulse counter
- Bit stream
- OPC event

Sending input signals to an NCC does not require any additional process objects, except for group alarm. The data is sent directly to the NET unit using a set of command procedures based on the data stored in the cross-reference tables. This data includes:

- Address to which the data is sent (NCC address)
- Alarm group information
- Information about the handling of the signal (signal handling attributes)
- The scale object used in the scaling of analog values

OPC event data type
When the signal handling Send as Analog is selected, the OE value of the OPC event process object is sent to an NCC using the analog data type. Otherwise, the OE value is sent to an NCC using the binary data type.

When the OE value is sent using the binary data type and the value is 0, the value 0 is sent. When the OE value is not equal to 0, the value 1 is sent.

6.3.1.2 Telegrams and parameters

The following table presents the different IEC 60870-5-101/104 Application Service Data Unit (ASDU) types that are used in COM500*i* when sending data to an NCC. The ASDU used depends

on the signal handling attributes selected for the signal in Signal X-References. Only those signal handling attributes that affect the ASDU used are mentioned.

For more information about defining the signal handling attributes, see Section 5.5.13.

Table 10: IEC 60870-5-101 ASDU types in COM500i

Data type	Signal Handling Attributes	ASDU Type
Binary input	None Send with Time Tag Send with Long Time Tag Send as Double Binary Send with Time Tag + Send as Double Binary Send with Long Time Tag + Send as Double Binary Send with and without Time Tag - Send with and without Time Tag + Send with Long Time Tag Send with and without Time Tag + Send as Double Binary Send with and without Time Tag + Send with Long Time Tag + Send as Double Binary	M_SP_NA_1(1) M_SP_TA_1(2) M_SP_TB_1 (30) M_DP_NA_1(3) M_DP_TA_1(4) - M_DP_TB_1 (31) - M_SP_NA_1(1) + M_SP_TA_1(2) M_SP_NA_1(1) + M_SP_TB_1(30) M_DP_NA_1(3) + M_DP_TA_1(4) M_DP_NA_1(3) + M_DP_TB_1(31)
Double binary	None Send with Time Tag Send with Long Time Tag Send as Single Indication Send with Time Tag + Send as Single Indication Send with Long Time Tag + Send as Single Indication - Send with and without Time Tag + Send with Long Time Tag Send with and without Time Tag + Send as Single Indication Send with and without Time Tag + Send with Long Time Tag + Send as Single Indication	M_DP_NA_1(3) M_DP_TA_1(4) M_DP_TB_1(31) M_SP_NA_1(1) M_SP_TA_1(2) - M_SP_TB_1 (30) - M_DP_NA_1(3) + M_DP_TA_1(4) - M_DP_NA_1(3) + M_DP_TB_1(31) - M_SP_NA_1(1) + M_SP_TA_1(2) M_SP_NA_1(1) + M_SP_TB_1(30)
Digital input	None Send with Long Time Tag Send with Time Tag Send as Analog Value Send with Time Tag + Send as Analog Value Send with Long Time Tag +Send as Analog Value Send with and without Time Tag Send with and without Time Tag + Send with Long Time Tag Send with and without Time Tag + Send as Analog Value Send with and without Time Tag + Send with Long Time Tag+ Send as Analog Value	M_ST_NA_1(5) M_ST_TA_1(6) M_ST_TB_1 (32) M_ME_NA_1(9) M_ME_TA_1(10) - M_ME_TD_1(34) - M_ST_NA_1(5) + M_ST_TA_1(6) M_ST_NA_1(5) + M_ST_TB_1(32) - M_ME_NA_1(9) + M_ME_TA_1(10) - M_ME_NA_1(9) + M_ME_TD_1(34)

Table continues on next page

Data type	Signal Handling Attributes	ASDU Type
Analog input	None Send with Time Tag Send with Long Time Tag Send as Floating Point Value Send as Scaled Value Send with Time Tag + Send as Floating Point Value Send with Time Tag + Send as Scaled Value Send with Long Time Tag + Send as Scaled Value Send with Long Time Tag + Send as Floating Point Value Send with and without Time Tag - Send with and without Time Tag + Send with Long Time Tag Send with and without Time Tag + Send as Floating Point Value Send with and without Time Tag + Send as Scaled Value Send with and without Time Tag + Send with Long Time Tag + Send as Floating Point Value Send as Binary Input Send as Binary Input + Send with Time Tag Send as Binary Input + Send with Long Time Tag Send as Binary Input + Send with and Without Time Tag Send as Binary Input + Send with and Without Time Tag + Send with Long Time Tag	M_ME_NA_1(9) M_ME_TA_1(10) M_ME_TD_1(34) M_ME_NC_1(13) M_ME_NB_1(11) M_ME_TC_1(14) - M_ME_TB_1(12) - M_ME_TE_1(35) - M_ME_TF_1(36) - M_ME_NA_1(9) + M_ME_TA_1(10) M_ME_NA_1(9) + M_ME_TD_1(34) M_ME_NC_1(13) + M_ME_TC_1(14) M_ME_NB_1(11) + M_ME_TB_1(12) M_ME_NC_1(11) + M_ME_TF_1(36) - M_SP_NA_1(1) M_SP_TA_1 (2) - M_SP_TB_1 (30) - M_SP_NA_1(1) + M_SP_TA_1(2) - M_SP_NA_1(1) + M_SP_TB_1(30)
Table continues on next page		

Data type	Signal Handling Attributes	ASDU Type
Pulse counter	None Send with Time Tag Send with Long Time Tag Send with and without Time Tag - Send with and without Time Tag + Send with Long Time Tag	M_IT_NA_1(15) M_IT_TA_1(16) M_IT_TB_1(37) M_IT_NA_1(15) + M_IT_TA_1(16) M_IT_NA_1(15) + M_IT_TB_1(37)
Bit Stream	None Send with Time Tag Send with Long Time Tag Send with and without Time Tag - Send with and without Time Tag + Send with Long Time Tag	M_BO_NA_1(7) M_BO_TA_1(8) M_BO_TB_1(33) M_BO_NA_1(7) + M_BO_TA_1(8) M_BO_NA_1(7) + M_BO_TB_1(33)
OPC event	None Send with Time Tag Send with Long Time Tag Send as Double Binary Send with Time Tag + Send as Double Binary Send with Long Time Tag + Send as Double Binary Send with and without Time Tag - Send with and without Time Tag + Send with Long Time Tag Send with and without Time Tag + Send as Double Binary Send with and without Time Tag + Send with Long Time Tag + Send as Double Binary Send as Analog Value Send with Time Tag + Send as Analog Value Send with Long Time Tag + Send as Analog Value	M_SP_NA_1(1) M_SP_TA_1(2) M_SP_TB_1 (30) M_DP_NA_1(3) M_DP_TA_1(4) - M_DP_TB_1 (31) - M_SP_NA_1(1) + M_SP_TB_1(30) M_DP_NA_1(3) + M_DP_TA_1(4) M_DP_NA_1(3) + M_DP_TB_1(31) M_ME_NA_1(9) M_ME_TA_1(10) - M_ME_TD_1(34)

The IEC 60870-5-104 protocol uses the same ASDUs as IEC 60870-5-101, but the default signal handling attribute for IEC 60870-5-104 is Send with Long Time Tag.

The IEC 60870-5-104 protocol standard does not define ASDUs with Short Time Tag. However, it is possible to use Send with Time Tag in the IEC 60870-5-104 protocol.

When Send with and without Time Tag has been selected, two consecutive messages are sent, one with and one without a time tag. This feature is suitable for masters that handle time-tagged and non-time-tagged data separately.

The table below presents the IEC 60870-5-101/104 cause of transmission values possible in COM500*i* for different types of data.

Table 11: IEC 60870-5-101/104 Cause Of Transmission (COT) values

Data type	COT	Explanation
Binary input	3 5 11 20	Spontaneous
Double binary	3 5 11 20	Spontaneous Return information caused by remote command Requested Interrogated by general interrogation
Digital input	3 5 11 20	Spontaneous Return information caused by remote command Requested Interrogated by general interrogation
Analog input	3 5 11 20	Spontaneous Return information caused by remote command Requested Interrogated by general interrogation
Pulse counter	3 5 11 20	Spontaneous Return information caused by remote command Requested Interrogated by general interrogation
OPC event	3 5 11 20	Spontaneous Return information caused by remote command Requested Interrogated by general interrogation

The following table describes the input data objects and variations used with the DNP 3.0 slave protocol. The data object and variation used depends on the signal handling attributes. The variations described in the table below are default variations, meaning variations that are used if no variation is specified by the master in the data request.

Table 12: DNP 3.0 input data objects and variations in COM500i

Data type	Signal Handling Attributes	Object	Variation
Binary input*	None Send Change Send Change with Time Send Change with Relative Time Send as Double Bit Input Object Send as Double Bit Input Object + Send Change Send as Double Bit Input Object + Send Change with time Send as Double Bit Input Object + Send Change with Relative time	1 2 2 2 3 4 - 4 - 4	1 (2) [1)1)] 1 2 3 2 1 - 2 - 3
Double binary*	None Send Change Send Change with Time Send Change with Relative Time Send as Double Bit Input Object Send as Double Bit Input Object + Send Change Send as Double Bit Input Object + Send Change with time Send as Double Bit Input Object + Send Change with Relative time	1 2 2 2 3 4 - 4 - 4	1 (2) [1)] 1 2 3 2 1 - 2 - 3
Binary output	Report Status to Master	10	2

Table continues on next page

Data type	Signal Handling Attributes	Object	Variation
Analog input	None Send as 16-bit Value Send without Flag Send as 16-bit Value +Send without Flag Send Change Event without Time Send as 16-bit Value + Send Change Event without Time Send Change Event with Time Send as 16-bit Value + Send Change Event with Time Send as Binary Input Send as Binary Input + Send Change Event without Time Send as Binary Input + Send Change Event with Time Send as Single Precision Float Send as Single Precision Float + Send Change Event without Time Send as Single Precision Float + Send Change Event with Time	30 30 30 30 - 32 32 - 32 32 - 1 2 - 2 - 30 32 - 32	1 2 3 4 - 1 2 - 3 4 - 1(2) [1] 1 - 2 - 5 5 - 7
Pulse counter	None Send as 32-bit Value Send as Delta Counter Send as 32-bit Value +Send as Delta Counter Send without Flag Send as 32-bit Value +Send without Flag Send as Delta Counter + Send without Flag Send as 32-bit Value +Send as Delta Counter +Send without Flag Send Change Event without Time Send Change Event without Time +Send as 32-bit Value Send Change Event with Time Send Change Event with Time + Send as 32-bit Value	20(21) [2][2] 20(21) [2] 20 20 - 20(21) [2] 20(21) [2] - 20 - 20 - - 22 22 - 22 22	2 1 4 3 - 6(10) [2] 5(9) [2] - 8 - 7 - - 2 1 - 6 5
Analog output	Report Status to Master Report Status to Master as Single - Precision Float	40 40	2 3
OPC event	None Send Change Send Change with Time Send Change with Relative Time Send As Analog Value Send As Analog Value + Send Change Send As Analog Value + Send Change with Time	1 2 2 2 30 32 - 32	1 (2) [1] 1 2 3 1 1 - 3

1) Variation 2 is used when the status of the process object changes.
2) Used if counters have been frozen by the master.

Signal specific classes

It is possible to define signal specific classes for the IEC 60870-5-101 and DNP 3.0 protocols in the COM500*i* functionality. With this feature, signals can have different priorities in events sent towards the NCC. To use a specific class in COM500*i*, write the class number in the column for the signal class, which is located on the **Indications** tab in Signal X-References (see Figure 33). If the cell is left empty, the default class is used.

The default class for the IEC 60870-5-101 protocol is 1. It is possible to select between classes 1 and 2; class 1 has a higher priority. The class definition for the IEC 60870-5-101 protocol is usable only when the unbalanced communication mode is used.

In DNP 3.0, there are two kinds of data: static data and event data. Static data in DNP 3.0 is called class 0 data. Event data can have three different classes or priorities: class 1 (high priority), class 2 (medium priority) and class 3 (low priority).

6.3.1.3 Handling of analog and digital values

Since some data types have a different value range in SYS600 and in the slave protocols supported in COM500*i*, some kind of scaling is needed. Analog input values are handled as follows:

In COM500*i*, the scaling of an analog signal can be defined separately for each NCC. This is done by selecting an existing scale object to the signal in question in Signal X-References.

The scaling algorithm is as follows:

- The Process value range is scaled to the SYS600 database value range of the scale object.
- The value sent to the NCC is limited to the value range defined by the message type of the NCC protocol.
- If the value is over this value range, the overflow bit of the analog telegram is set in the IEC 60780-5-101/104 and DNP 3.0 protocols, and in the RP 570 protocol the status of the signal is marked as invalid.

For example, if the value of an analog signal sent to an IEC 60870-5-101/104 master as a scaled value is to be divided by ten, the parameters of Scale Object Tool should be as shown in Figure 81. For more information on Scale Object Tool, see SYS600 Application Objects.

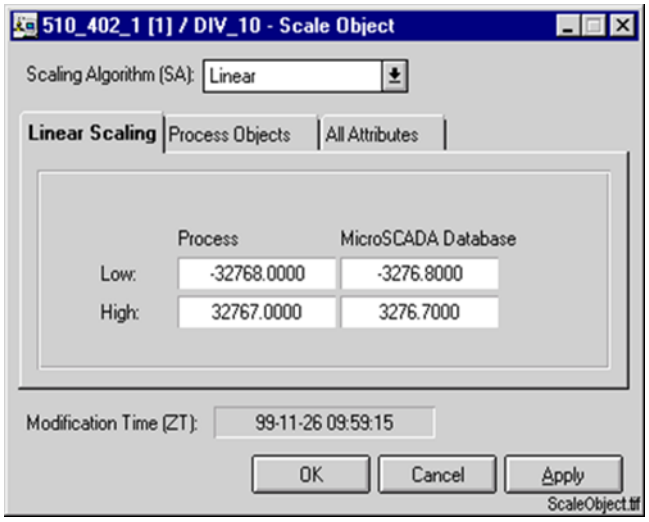


Figure 81: Scale Object Tool dialog

Digital input values are handled as follows:

- In SYS600, the value is 0…65535.
- In RP 570 and CPI, the value is 0…65535 or -2000…2000 (sent as analog value).
- In IEC 60870-5-101/104, the value (as seen in SCIL) is 0..127 (step point information) or -32767…32767 (sent as analog value).
- If the value range of the slave protocol is different than in SYS600, a digital input value is limited as follows: the value sent to the NCC is limited between 0 and the maximum of the value range of the slave protocol.
- In Modbus, the value is 0... 65535 (sent as an analog value).

6.3.1.4 Deadband

The transmission of analog values as events in DNP3.0 can be limited by using the deadband possibility. Currently, all changes to analog values are transmitted by the COM_USAI procedure to the NCC. The modification of the COM_USAI procedure is needed to define the deadband. Note that the modifications have an effect on all analog points transmitted to the NCC via DNP3.0.

The delta value can be hardcoded if the same delta value is acceptable throughout the whole system.

```
#WHEN 4 #BLOCK ; ********************** DNP 3.0 slave*******..
;protocol parameters
    @SD_CLASS = 2 ; middle weight (priority)
    #IF %SIGNAL_CLASSES(%NCC_COUNTER) == 0 #THEN #BLOCK
    @EV_CLASS = 0 ; use default event class
    #BLOCK_END
#ELSE @EV_CLASS = %SIGNAL_CLASSES(%NCC_COUNTER) ; use signal spes. class
    @DELTA = 25 ; original = 0
    @EV_ALL = 0
```

The event is only transmitted if the difference between the new value and the previously transmitted value is larger than 25.

The project specific definition can also be used in the following way:

```
#WHEN 4 #BLOCK ; ********************** DNP 3.0 slave*******
.
.
.
;***************** project-specific modifications*******************
    #IF BIT(%SIGNAL_HANDLING_ATTRS(%NCC_COUNTER),0) == 1 #THEN #BLOCK
    ;write the project-specific code in this block
    @DELTA = 30
    #BLOCK_END
```

The event is only transmitted if the difference between the new value and the previously transmitted value is larger than 30. This applies only to the points, which have the project-specific bit set in the Signal X-References tool. This modification is useful if only a small group of points generate a useless load to the NCC line.

If each point or each station must have a limiting value of its own, the usage of a threshold value in the process object may be considered, see SYS600 Application Objects. The TH attribute can be used to limit the amount of activation of COM_USAI procedure. If a nonzero value is used in TH, it is usually better not to modify COM_USAI as described above.

6.3.2 Commands

6.3.2.1 Mechanism and data types

COM500*i* receives a command from an NCC as an input, which activates a set of command procedures. It sends the command to the process objects that are connected to the process units based on the information stored in:

- Logical names and indexes of the output process objects
- Logical name and index of the response indication, if any are connected
- Information about the handling of the signal (signal handling attributes)

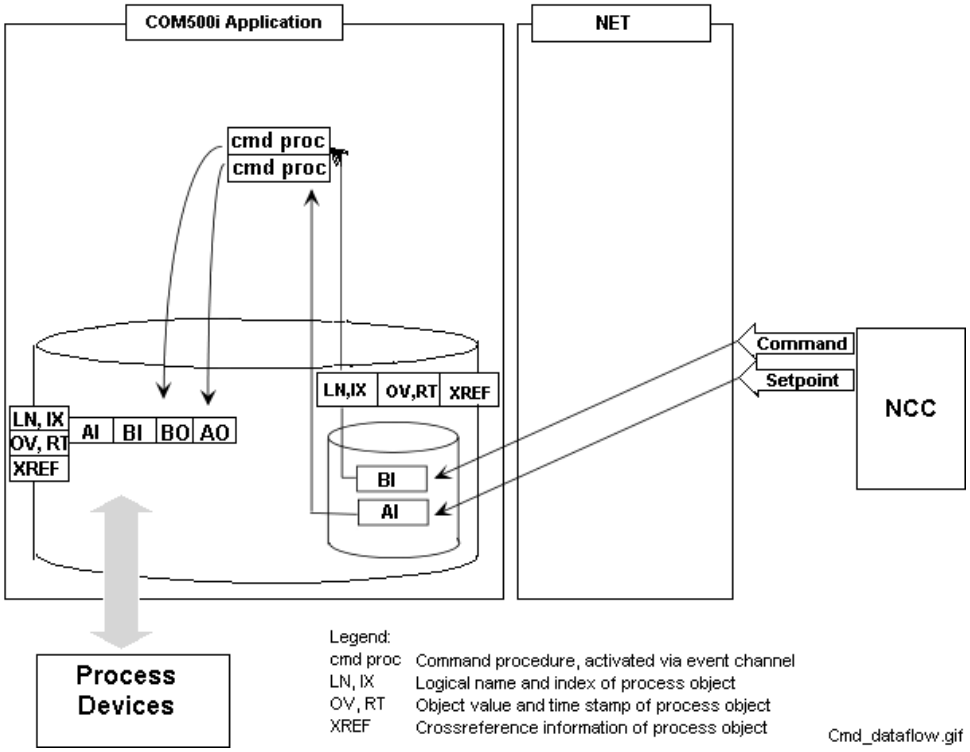


Figure 82: Command from the NCC into the COM500i application

The input process objects that receive commands from the NCC are created automatically by Signal X-References. The following output data types are supported in COM500*i*:

- Binary output
- Digital output
- Analog output

6.3.2.2 Telegrams and parameters

The following table presents the different IEC 60870-5-101/104 ASDU types that are expected in COM500*i* when receiving commands from an NCC. The ASDU used depends on the signal handling attributes selected for the signal in Signal X-References. Only the signal handling attributes that affect the ASDU used are mentioned.

If a digital command is made to a non-IEC protocol, the command type from an NCC must be analog. The Regulating Step Command is a binary command. Other protocols than IEC 60870-5-101/104 use the digital value itself in the command, for example a 16-bit value. Time-stamped commands (ASDUs 58..64) are not usually used with IEC 60870-5-101 since those are not defined in the IEC60870-5-101 standard.

Table 13: IEC 60870-5-101/104 ASDU types in COM500i

Data type	Signal Handling Attributes	ASDU Type
Binary output	Project Specific Inversed Value Receive as Double Command Receive as Direct Command None	- - C_DC_NA_1(46), C_DC_TA_1(59) - C_SC_NA_1(45), C_SC_TA_1(58)
Analog output	Project Specific Inversed Value Receive as Double Command Receive as Single Command Send without Synchrocheck - Receive as Direct Command None - -	- - C_DC_NA_1(46), C_DC_TA_1(59) C_SC_NA_1(45), C_SC_TA_1(58) C_DC_NA_1(46), C_DC_TA_1(59) C_SC_NA_1(45), C_SC_TA_1(58) - C_SE_NA_1(48), C_SE_TA_1(61) C_SE_NB_1(49), C_SE_TB_1(62) C_SE_NC_1(50), C_SE_TC_1(63)
Digital output	Project Specific Inversed Value Receive as Double Command Receive as Single Command None	- - C_DC_NA_1(46), C_DC_TA_1(59) C_SC_NA_1(45), C_SC_TA_1(58) C_SE_NA_1(48), C_SE_TA_1(61) C_SE_NB_1(49), C_SE_TB_1(62) C_SE_NC_1(50), C_SE_TC_1(63)

The following table describes the output data objects and variations used with the DNP 3.0 slave protocol.

Table 14: DNP 3.0 output data objects and variations in COM500i

Data type	Signal Handling Attributes	Object	Variation
Binary output	Project Specific Inversed Value Report Status to Master None	 10 12 12	 2 1 1
Analog output	Project Specific Inversed Value Report Status to Master Report Status to Master as Single - Precision Float Receive as Single Command Send without Synchrocheck None	 40 41 40 41 12 12 41	 2 2 3 3 1 1 2

Analog output values are scaled and limited using a reverse algorithm that is used when analog input values are sent to the NCC.

It is possible to define binary input commands received from the NCC for the analog output command. These are used when commanding a REC 561 type of device. It uses analog values for binary commands. These analog values can be defined by the user in the COMTool parameters, otherwise default values are used.

The RP 570 and CPI command analog output values are scaled and limited by using the scale COM_AIRP'ncc_number'. They are generated when the COM500*i* application is started for the first time.

6.3.2.3 Command confirmations in IEC 60870-5-101/104

The IEC 60870-5-101/104 protocol includes the concept of command confirmation and termination. In COM500*i*, commands sent from an NCC using the IEC 60870-5-101/104 protocol are confirmed and terminated as follows:

- System commands are always confirmed by COM500*i*.
- All application commands, except the reading of user data (ASDU 102), are confirmed and terminated by COM500*i*.
- Reset process commands (ASDU 105) are only confirmed, not terminated.
- If a time-stamped command (ASDU 58..64) comes too late, it is not confirmed. This functionality is based on IEC 60870-5-104 standard.

Direct data commands are confirmed and terminated as follows:

- A command is confirmed when the handling of the command in the corresponding command procedure begins.
- If the command is not authorized, a negative command confirmation is sent.
- If the response indication related to the command has not been connected to the command in Signal X-References, the command is terminated when the handling of the command in the corresponding command procedure is finished.

Secured data commands are confirmed and terminated as follows:

- A command is confirmed when the process device replies to the selected command. If the response is negative, a negative command confirmation is sent to the NCC.
- If the command is not authorized or it fails, a negative command confirmation is sent.
- A command is terminated when the process device replies to the executed command, and the command is not attached to a specific indication. If the reply is negative, a negative activation termination is sent to the NCC.
- If the response indication related to the command has not been connected to the command in Signal X-References, the command is terminated when the handling of the command in the corresponding command procedure is finished.

If an indication is connected to a data command, the following rules apply:

- If the output objects are of IEC type, the command is terminated when the termination is received from the device. The IEC/Analog Input termination process object must be in index (IX) with the indication process object index added with 100. The Object Address (OA) must be 1 000 000Hex added with a command object address to receive confirmation and termination from the device to the database. To use this function an indication must be connected to the command in the ComTool.
- In case of another output object type, the command is terminated when the connected response indication is updated.
- If the termination (IEC) or response indication (other types) is not received within the time-out parameter Response Indication Timeout defined in Signal X-References, a negative termination is sent.

The value of the PC attribute of each IEC slave station should be set to 0 when using COM500*i*. Otherwise, commands may not be properly confirmed or terminated. See the description of the station attribute PC from the protocol manuals for more information.

6.3.3 System messages, system and application commands

The NET unit generates protocol-specific system messages as status codes to inform the user about some special conditions, such as loss of communication.

NCC protocols provide specific application and system commands that are used in such tasks as time synchronization and interrogation of data. These commands are also handled by a COM500*i* application.

The system and application commands supported in COM500*i* can be found in the interoperability lists and device profiles of the NCC protocols and in the SYS600 manuals describing the implementation of these protocols.

The NET unit handles application and system commands. These commands are received by process objects that are created by Signal X-References. After this, the COM500*i* command procedures are activated. To ensure that these commands are received and executed properly, the values of the MI (Message Identification) and CA (Command Address) attributes of IEC and DNP slave stations should be checked as stated in Section 4 of this document.

6.3.4 Time synchronization

COM500*i* should be synchronized from max. one NCC by default. Until a synchronization message is received, the time stamps of the messages sent to the NCC are marked as invalid. After a synchronization message has been received, time stamps are marked as valid until the next system restart.

If the system is synchronized by other means, for example, by using a local GPS receiver or a SNTP client, there is no need to mark the time stamps as invalid. This can be done in different NCC protocols as follows:

- In IEC 60870-5-101/104 by setting the RM attribute of the IEC slave station so that bit 1 of the value is set (RM = 2 if no other bits are set)
- In RP570 slave by setting the TI attribute of the SPI station to 1
- In DNP 3.0 by setting the TC attribute of the DNP slave station to 1

6.3.5 File transfer

In general, the IEC 60870-5-101/104 file transfer feature can be used to transmit files of all types from relays to the SYS600 computer. The transmitted files can be selected freely from the gateway computer and transmitted to the NCC.

In the file transfer process, files from the process devices are stored to a directory in the COM500*i* file system. The time channel activates a command procedure and reads the files from the directory. The files are then transferred to the NCC. The file transfer from a process device to one or several NCCs is shown in Figure 83.

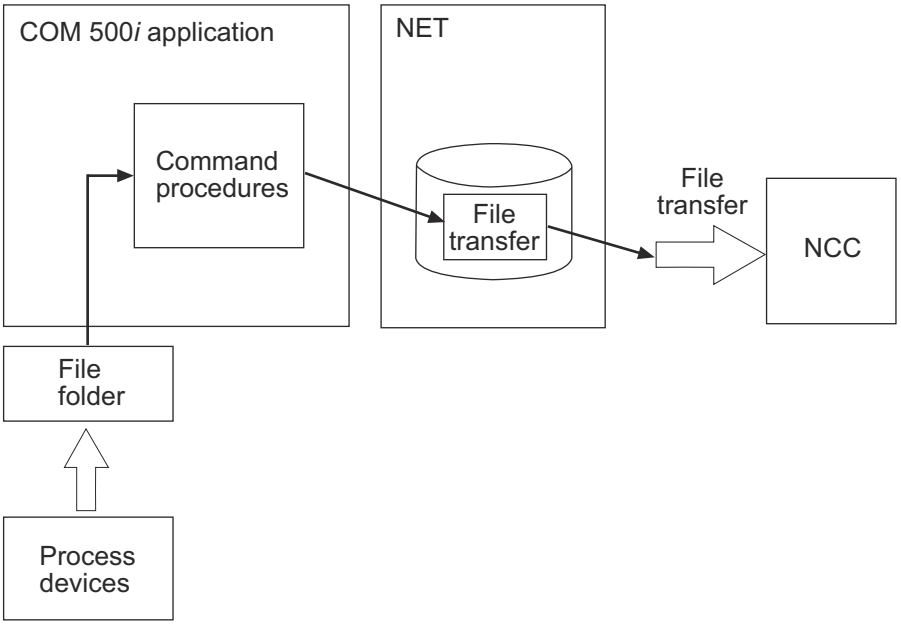


Figure 83: File transfer process

6.3.6 Transparent SPA messages

In general, the Transparent SPA functionality is used to upload or download relay parameters via the gateway using Relay Setting Tool. It is also possible to transfer disturbance collector files from the process units to the upper level system (NCC).

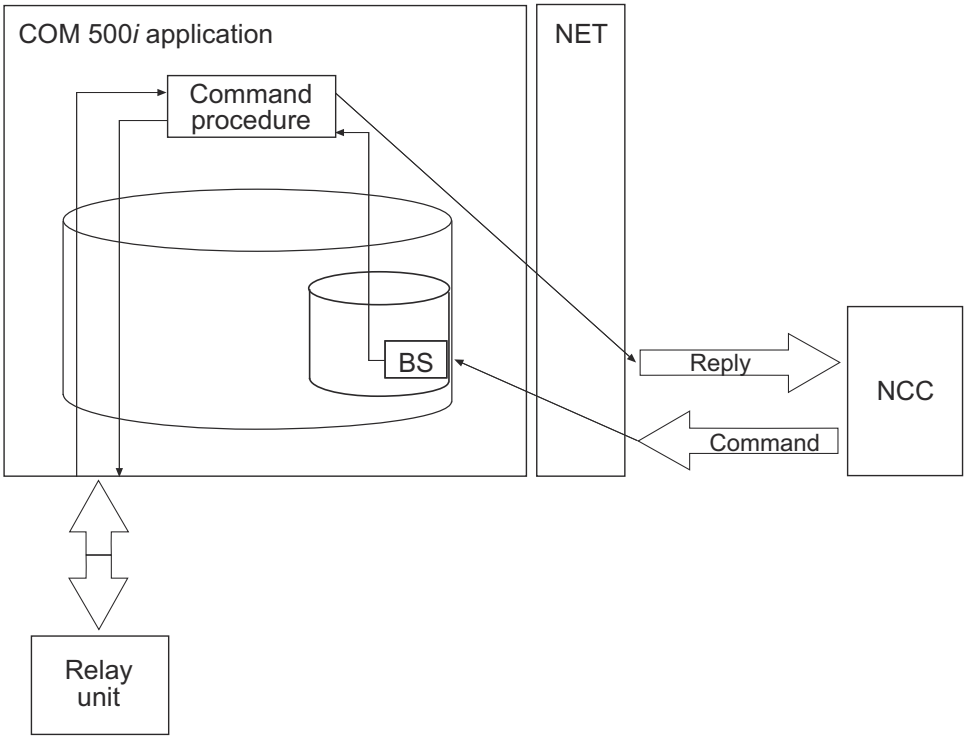


Figure 84: Transparent SPA process

In Relay Setting Tool, a separate IEC address must be set for each relay unit using the transparent SPA functionality of COM500*i*. Consequently, when using Transparent SPA, the IEC

60870-5-101/104 Gateway (COM 6xx/500) protocol is selected in CConfig COM500*i*. If the LIB 500/510 version is older than 4.2 Add-On 1, the IEC 60870-5-101/104 Gateway (COM 6xx) protocol should be selected.

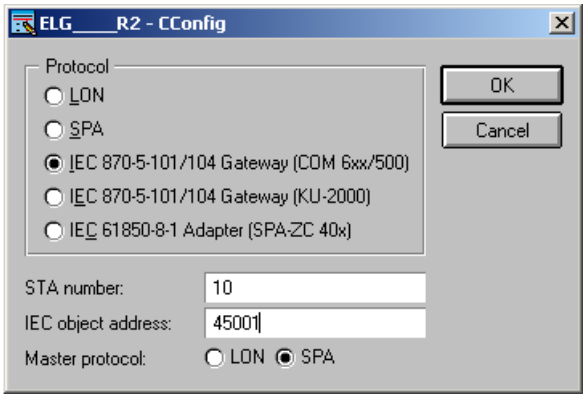


Figure 85: Protocol selection in CConfig (LIB 500/510 4.2 Add-On 1)

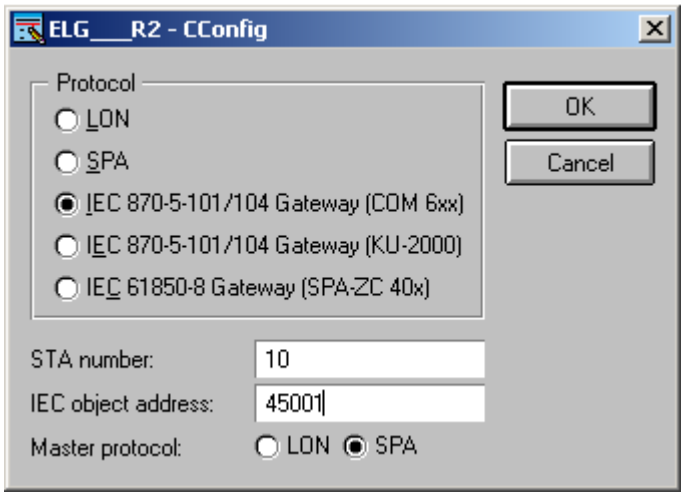


Figure 86: Protocol selection in CConfig (LIB 500/510 4.2)

6.4 Cross-reference and parameter files

Cross-reference information and various parameters are stored in free type process objects and ASCII text files.

The files described in this section should never be edited manually because it may lead to a severe application malfunction.

6.4.1 Cross-reference files

Signal X-References writes the cross-reference information to ASCII text files from where it is automatically loaded at start-up. The following text files are found in the directory /APL/<name>/com500:

- COM_XRNCC1.txt contains the NCC definitions
- COM_XRCMD*.txt contains cross-reference information of the commands (output process objects)

If the previous version of COM500*i* is used, the directory may also contain COM_XRIND*.txt files. These files are no longer used to store the indication cross-reference information.

The cross-reference information of indications is stored in free-type process objects.

6.4.2 System and application parameter file

COM500*i* uses a set of system and application parameters to control its operation. These parameters are used by Signal X-References and the COM500*i* command procedures. Some of the parameters can be edited on the **Parameters** tab of Signal X-References, and some are internal parameters of COM500*i*. The parameters are saved in the file Com500.ini.

6.4.3 Parameter files of Signal X-References

These parameter files are used for defining the user-interface, attributes and view definitions in Signal X-References. There are three parameter files, which have the following names, locations, and purposes:

- Attr_com.txt contains a list of valid attributes concerning process objects that can be attached as columns to the **Indications** and **Commands** tabs of Signal X-References. Location: /sc/Stool/SysConf.
- ComTool.ini is located in the directory: /sc/apl/<name>/par/<user> and it contains:
  - The definitions for Signal X-References co-ordinates on screen during the last session, number of columns attached to the **Indications** and **Commands** tabs and the number of selected view definitions.
  - The definitions for ruler positions located between adjacent columns to define the width of the column.
  - Definitions for attribute names and their titles in columns.
- ComView.ini contains the definitions for the number of views assigned to Signal X-References and the names of views defined using View Definitions. The assigned indication and command signal conditions for every view defined using View Definitions. Location: /sc/apl/<name>/par/<user>.

6.5 Application objects

6.5.1 Introduction

Several application objects are created by COM500*i*. Some of these objects are created automatically at the first start-up and others are created when definitions are made in Signal X-References. Also existing application objects are modified.

In COM500*i*, the following naming conventions are used:

- Event channels, time channels, free-type objects and command procedures are named COM_*:*
- Process objects and free-type process objects are named BNCC*:P

COM500*i* application objects should not be removed or modified.

6.5.2 Application objects created by COM500*i*

COM500*i* creates new application objects for the following purposes:

- Event channels and command procedures for sending data from the input process objects to the NCCs.
- Process objects for receiving commands from the NCCs.
- Event channels and command procedures for interpreting commands coming from the NCCs and sending them to the correct output process objects.
- Event channels, process objects, and command procedures for group alarm functionality.
- Event channels and command procedures for COM application start-up and initialization.
- Event channels and command procedures application and system commands.
- Time channels, event channels, and command procedures for command termination.
- Free-type objects and process objects for storing cross-reference data.
- Free-type object for storing internal parameters

In addition to the application objects described above, some other objects may also be needed for the COM functionality. Signal X-References creates these objects.

If an address overlap occurs when creating a process object with a predefined address, the execution of the COM500*i* command procedure is attached to the secondary objects of the event channel connected to the existing process object.

6.5.3 Application objects modified by COM500*i*

When a cross-reference is attached to a process object, the following modifications are made:

- The Table Index (TI) attribute of the process object is set to point to the due position of the cross-reference table.
- The event channel, for example, COM_USAI for analog input process objects, is attached to the process object and the event channel activation is set. If the event channel (AN attribute) of the process object is already reserved, the execution of the COM500*i* command procedure is attached to the secondary objects list of the event channel.

Other modifications made by COM500*i* include:

- The execution of the command procedure COM_COMINI is attached to the predefined command procedure APL_INIT_1.
- The execution of the command procedure COM_COMINI_H is attached to the predefined command procedure APL_INIT_H.
- The execution of the command procedure COM_SUSSTA is attached to the predefined command procedure APL_EVENT.
- The execution of the command procedure COM_SUSNOD is attached to the predefined command procedure SYS_EVENT.
- The execution sequence for the NET restart uses the system event process object for the NET node number in use.

6.6 COM500*i* command procedures

There are separate command procedures for delivering the different types of SYS600 data supported in COM500*i*. Each of these command procedures is executed in a parallel queue of its own. The data type and function of these command procedures can be defined to be based on the following naming conventions:

- The command procedures that send data to the upper level system are named COM_US + data type, for example COM_USDB for double binary indication data.
- The command procedures that forward commands from the upper level system to the process devices are named COM_DS + data type, for example, COM_DSAO for analog output data.

In addition to the command procedures designed for actual data delivery, there are command procedures for other purposes, such as parameter reading, interpreting cross-reference information, and executing the system and application commands. All the command procedures included in COM500*i* are described briefly in this section.

In some cases the command procedures are executed spontaneously, i.e. driven by event channel execution caused by the update of a process object, and in some cases the procedures are executed by other command procedures.

The command procedures are automatically created at start-up, when the COM500*i* license has been installed and a monitor is opened. The source code of the procedures is read from the text files located in the directory /com/active/com_. For performance reasons most of the command procedures are compiled when they are created, which means that the compiled code is executed during the signal routing.

To prevent the handling of different data types from affecting each other, COM500*i* uses different parallel queues for command procedures.

The following list shows the parallel queues used with the dedicated COM500*i* command procedures.

Table 15: Used Parallel Queues with Dedicated Command Procedures

Queue	Command Procedures
1	COM_COMINI, COM_COMINI_H, COM_RDGEN, COM_RDXREF, COM_NETINI, COM_NETINI_H, COM_GRPAL, COM_GRPSND, COM_RESPRC, COM_RDDATA, COM_SUSHOST, COM_SUSNOD, COM_SUSSTA, COM_PNDIND, COM_RESSEL, COM_DCNET, COM_REVDTA, COM_INIT, COM_PNDIND
2	COM_RPSCR, COM_RPSDI, COM_RPSFT, COM_RPSS, COM_101SCR, COM_104SCR, COM_IESS, COM_IESA, COM_IESEI, COM_IECTRM, COM_CPISS, COM_DNPSCR, COM_DNPSS, COM_DSUN, COM_CPIDI, COM_RPSY
7	COM_USDB
8	COM_USBI
9	COM_USAI, COM_USXREF
10	COM_USDI
11	COM_GENINT, COM_CNTINT, COM_IEGICL, COM_DBSTA, COM_IEGICL
12	COM_USBS, COM_USPC
13	COM_DSBO
14	COM_DSAO, COM_DSDO
15	COM_USAO, COM_USBO, COM_IEFT, COM_USFT, COM_TRPSPA
16	COM_USOE

The parallel queues from number 3 to 6 can be freely used in COM500*i* applications (revision 3.0 or later). It must be noted that LIB 500 uses those queues when it is installed. See the LIB 500 documentation for more details.

In situations where there is a need for free parallel queues, the following can be done to 5 free queues:

- Set all the COM500*i* command procedures in queue 1 to queue 0. Now these procedures use the first random queue.
- Move the command procedures from queue 12 and 15 to queue 10.
- Move the command procedures from queue 13 and 14 to queue 11.

For more information on how to change a queue, see SYS600 Application Objects.

6.6.1 Description of the command procedures

The different command procedures are described in Table 16:

Table 16: Command procedures

Command Procedure	Description
COM_101SCR:C	Creates an IEC 60870-5-101 slave line and station(s). This command procedure can be used for communication system configuration and executed for example from the predefined command procedure APL_INIT_1:C.
COM_104SCR:C	Creates an IEC 60870-5-104 slave line and station(s). This command procedure can be used for communication system configuration and executed for example from the predefined command procedure APL_INIT_1:C.
COM_AUTHCH:C	Executes the COM500*i* authorization check mechanism. For more information, see Section 5.5.16.This command procedure is executed by each of the command procedures COM_DS**:C.
COM_BOCMD:C	Executes a direct or secured object command depending on the command type, the protocol of the upper level system and the protocol of the process device. This command procedure is executed by the command procedure COM_DSBO:C.
COM_CNTINT:C	Executes a counter interrogation command sent from an NCC of protocol type IEC 60870-5-101. This command procedure is executed by the command procedure COM_IESA:C.
COM_COMINI:C	Executes the start-up mechanism of a COM500*i* application by executing a number of sub-procedures. The execution of this command procedure is automatically attached to the predefined command procedure APL_INIT_1:C.
COM_COMINI_H:C	Executes the start-up mechanism after switchover. The execution of this command procedure is automatically attached to the predefined command procedure APL_INIT_H:C.
COM_CPIDI:C	Sets the DI (Database Initialized) attribute of CPI when the COM500*i* process database has been updated after start-up (when the time set in the parameter Database Initialization Time has expired). For more information, see Section 5.5.16. This command is executed by the command procedure COM_NETINI:C.
COM_CPISS:C	Interprets the system messages coming from a CPI-connected NCC and executes a number of sub-procedures according to the received system messages.
COM_DBSTA:C	Marks the database initialized for the COM500*i* application. This command procedure is executed by COM_NETINI:C.
COM_DCNET:C	Creates event channels and command procedures for restarting a communication unit. This command procedure is executed once by COM_COMINI:C.
COM_DNPSCR:C	Creates a DNP 3.0 slave line and station. This command procedure can be used for communication system configuration and executed for example from the predefined command procedure APL_INIT_1:C.
COM_DNPSS:C	Interprets and confirms the application commands sent from an NCC of protocol type DNP 3.0. Application commands (for example cold start) are executed by a number of sub-procedures.
COM_DSAO:C	Forwards an analog output-type command coming from an NCC to the process device. The value of the command is handled according to the signal handling attributes selected in the Signal X-References.
COM_DSBO:C	Forwards a binary output-type command coming from an NCC to the process device. The value of the command is handled according to the signal handling attributes selected in Signal X-References. Executes the sub-procedure COM_BOCMD:C.
Table continues on next page	

COM_DSCMDBLK:C	Checks a command blocking. Tagout functionality uses this command procedure and Tagout installation adds the blocking code to the command procedure.
COM_DSDO:C	Forwards a digital output-type command coming from an NCC to the process device. The value of the command is handled according to the signal handling attributes selected in Signal X-References.
COM_DSUN:C	Sends a negative confirm command to IEC 60870-101/104 NCC and a confirm command with status 4 (Control operation is not supported for this point) to DNP 3.0 NCC, when the command address is not defined in the MicroSCADA database.
COM_DSXREF:C	Interprets and handles the data stored in the command cross-reference tables. This command procedure is executed by each of the command procedures COM_DS**:C.
COM_GENINT:C	Executes a general interrogation command sent from an NCC depending on the protocol of the NCC. This command procedure is executed by the command procedures COM_IESA:C COM_CPISS:C and COM_RPSS:C.
COM_GRPAL:C	This command procedure takes care of the group alarm handling.
COM_GRPSND:C	Performs a forced sending of group alarms, for example at start-up or when re-initializing a NET database.
COM_IECTRM:C	Forwards a command termination sent from an IEC 60870-5-101/103 process device to an NCC of protocol type IEC 60870-5-101/104.
COM_IEFT:C	Sends the file to the NCC of protocol type IEC 60870-5-101/104. This command procedure is executed by the command procedure COM_USFT:C.
COM_IEGICL:C	Resets GI data in IEC 60870-5-101/104 queues of pc_net when a GI Cancel command is received from IEC 60870-5-101/104 NCC.
COM_IESA:C	Interprets and confirms the application commands sent from an NCC of protocol type IEC 60870-5-101/104. Application commands (for example, general interrogation) are executed and terminated by a number of sub-procedures.
COM_IESEI:C	Sends an end-of-initialization message (ASDU 70) to an NCC of protocol type IEC 60870-5-101/104 when the COM500*i* process database has been updated after start-up (after the time set in the parameter Database Initialization Time has expired). For more information, see Section 5.5.16. This command procedure is executed by the command procedure COM_NETINI:C.
COM_IESS:C	Interprets and confirms the system commands sent from an NCC of protocol type IEC 60870-5-101/104.
COM_INIT:C	Initializes the selected Trip Signals in COM500i start-up or when the process device gets the running status.
COM_LOG:C	Writes signal routing data to log files.
COM_NETINI:C	Initializes the internal databases of the RP 570 and DNP 3.0 slave devices in NET. The CPI slave is also initialized. Executes the command procedures COM_GENITNT:C, COM_IESEI:C and COM_RPSDI:C (after the time set in the parameter Database Initialization Time has expired).
COM_NETINI_H:C	Initializes the internal databases of the RP 570 and DNP 3.0 slave devices in NET after switchover. The CPI slave is also initialized. Executes the command procedures COM_GENITNT:C, COM_IESEI:C and COM_RPSDI:C (after the time set in the parameter Database Initialization Switchover Time has expired).
COM_PNDIND:C	Checks whether there are pending response indications (indications connected to a command that have not yet been updated) older than the parameter Response Indication Time-out and if such indications are found, terminates the corresponding commands (negative termination). This command procedure is called by time channel COM_PNDIND. For more information, see Section 5.5.16.

Table continues on next page

COM_RDDATA:C	Executes a read of user data command sent from an NCC of protocol type IEC 60870-5-101/104. This command procedure is executed by the command procedure COM_IESA:C.
COM_RDGEN:C	Initializes and reads the system and application parameters either from the application itself or from the parameter file Com500.ini. This command procedure is executed either by the command procedure COM_COMINI:C at start-up or by the Signal X-References Tool when the parameters have been edited.
COM_RDXREF:C	Loads the cross-reference information from the downstream cross-reference files to RAM at start-up. This command procedure is executed by the command procedure COM_COMINI:C.
COM_RESPRC:C	Executes a reset process command sent from an NCC of protocol type IEC 60870-5-101/104. This command procedure is executed by the command procedure COM_IESA:C.
COM_RESSEL:C	Resets the internal selection flag of a command coming from a CPI-connected NCC (after the time set by the parameter CPI Command Reset Time has expired), if no execute command is received to the same address.
COM_REVDTA:C	Contains the revision information of the current COM500*i* version.
COM_RPSCR:C	Creates an RP 570 slave line and station. This command procedure can be used for communication system configuration. It can be executed from the predefined command procedure APL_INIT_1:C
COM_RPSDI:C	Sets the DI (database initialized) attribute of a RP 570 slave station after the COM500*i* process database has been updated after start-up (after the time set in the parameter Database Initialization Time has expired). This command procedure is executed by the command procedure COM_NETINI:C For more information, see Section 5.5.16.
COM_RPSFT:C	Loads FTABs from a text file to an RP 570 slave station. The FTAB file must be created using other software.
COM_RPSS:C	Interprets the system messages coming from an RP 570 slave NCC and executes a number of sub-procedures according to the system message.
COM_RPSY:C	Marks an RP 570 slave NCC as synchronized when a clock synchronization message has been received from the NCC.
COM_SUSHOST:C	Handles LOST, DISCONNECTED, DOWN, DISABLED, HOST_DISABLED, CONNECTED and RECONNECTED events from the Host application.
COM_SUSNOD:C	Executes command procedure COM_SUSSTA, when connection to a LAN node is lost. This command procedure has been connected to event channel SYS_EVENT.
COM_SUSSTA:C	Sends the indications connected to process devices to the NCC after the process device is suspended. This command procedure is connected to the predefined event channel APL_EVENT.
COM_TRPSPA:C	Receives and sends transparent SPA messages. The transparent SPA attributes are selected on the NCC tab of Signal X-References for the NCCs using the IEC 60870-5-101/104 protocol.
COM_USAI:C	Sends analog input values to the NCCs. The value of the signal is handled according to the signal handling attributes selected in the Signal X-References.
COM_USAO:C	Sends the status of analog output values to NCCs of protocol type DNP 3.0 if the corresponding signal handling attribute has been selected in Signal Cross-Reference.
COM_USBI:C	Sends binary input values to the NCCs. The value of the signal is handled according to the signal handling attributes selected in Signal Cross-Reference.
COM_USBO:C	Sends the status on binary output values to NCCs of protocol type DNP 3.0 if the corresponding signal handling attribute has been selected in Signal Cross-Reference.
COM_USBS:C	Sends the bit stream values to the NCC. The value of the signal is handled by the signal handling attributes selected in Signal Cross-Reference.

Table continues on next page

COM_USDB:C	Sends double binary input values to the NCCs. The value of the signal is handled according to the signal handling attributes selected in Signal Cross-Reference.
COM_USDI:C	Sends digital input values to the NCCs. The value of the signal is handled according to the signal handling attributes selected in Signal Cross-Reference.
COM_USFT:C	Sends the files to the NCCs. The file transfer attributes are selected for the NCCs, whose protocol type is IEC 60870-5-101/104 or IEC 61850, on the NCC page of Signal X-References. This command procedure is called by the time channel COM_USFT.
COM_USOE:C	Sends OPC event values to the NCCs. The value of the signal is handled according to the signal handling attributes selected in Signal X-References.
COM_USOXR:C	Interprets and handles the data stored in the command cross-reference tables when the status of output objects is sent to NCCs of protocol type DNP 3.0.
COM_USPC:C	Sends pulse values to the NCCs. The value of the signal is handled according to the signal handling attributes selected in Signal X-References.
COM_USXREF:C	Interprets and handles the data stored in the indication cross-reference tables. This command procedure is executed by each of the command procedures COM_US**:C

6.6.2 Modifications to the command procedures

Besides the exceptions mentioned in this section, the COM500*i* command procedures should not be modified.

Each of the command procedures COM_US**:C and COM_DS**:C are divided into four sections based on the NCC protocol. The first section is for RP 570 slave, the second for IEC 60870-5-101/104 slave, the third for CPI and the fourth is for DNP 3.0. In each of these sections there is the following empty SCIL block:

```
;********** project-specific modifications ***************
#IF LENGTH(SELECT(%SIGNAL_HANDLING_ATTRS(%NCC_NR),"==1")) > 0 #THEN #BLOCK
;write the project-specific code in this block
#BLOCK_END
```

This project specific block is for the project specific modifications that are protocol and data type specific and can be activated by selecting the Project Specific signal handling attribute in Signal X-References for an individual signal. For example, if an analog signal is connected to three NCCs and the Project Specific signal handling attribute is selected for the RP 570 NCC, the SCIL code in the project specific block in the RP 570 section of the command procedure COM_USAI:C is executed, when this individual signal is sent to the NCCs. The project specific block can be used for modifying the value, time stamp, status information and other parameters that are sent to the NCCs or the data of the command received from an NCC.

When editing the COM500*i* command procedures, make sure that the command procedures that are compiled are also compiled after editing. Otherwise, the performance of COM500*i* may decrease.

6.7 File summary

The following files are copied to the system when COM500*i* is installed. All the files are located in the directory /com/active/com_.

6.7.1 VSO files

File	Description
ComTool.vso	COM500*i* Signal X-References
Com_Stand.vso	COM500*i* Diagnostics Tool
COMLViewer.vso	COM500*i* Log Viewer

6.7.2 INI files

File	Description
Comstand.ini	Initialization file for Diagnostics Tool
Comtool.ini	Initialization file for Signal X-References
Tools.ini	Initialization file for Tool Manager
Toolsupd.ini	File used for updating Tool Manager
Toolview.ini	Initialization file for Tool Manager

6.7.3 Text files for command procedure source code

File	Description of the Procedure
COM_101SCR.txt	Creates a line and station for IEC 60870-5-101 slave.
COM_104SCR.txt	Creates a line and station for IEC 60870-5-104 slave.
COM_AUTHCH.txt	Checks the authority of a command from an NCC.
COM_BOCMD.txt	Performs a direct or secured command.
COM_CNTINT.txt	Performs a counter interrogation.
COM_COMINI.txt	Initializes a COM500*i* application.
COM_COMINI_H.txt	Initializes a COM500*i* application after switchover.
COM_CPIDI.txt	Sets the DI attribute of the CPI software.
COM_CPISS.txt	Receives and interprets the CPI system messages.
COM_DCNET.txt	Restarts COM500*i* if NET is restarted.
COM_DBSTA.txt	Marks the database initialized.
COM_DNPSCR.txt	Creates a line and a station for DNP 3.0 slave.
COM_DNPSS.txt	Interprets DNP 3.0 application commands.
COM_DSAO.txt	Performs a direct command using an AO object.
COM_DSBO.txt	Performs a command using BO objects.
COM_DSCMDBLK.txt	Checks a command blocking.
COM_DSDO.txt	Performs a direct command using a DO object.
COM_DSUN.txt	Sends a negative confirm when the command address is unknown.
COM_DSXREF.txt	Decodes command direction cross-reference data.
COM_GENINT.txt	Performs a general interrogation.
COM_GRPAL.txt	Sends a group alarm to an NCC.
COM_GRPSND.txt	Sends the value of all the group alarms to the NCC.
COM_IECTERM.txt	Forwards command termination to NCCs.
COM_IEFT.txt	Sends a file to the IEC 101/104 NCC.
COM_IEGICL.txt	Resets GI data in IEC 60870-5-101/104 queues of pc-net.

Table continues on next page

File	Description of the Procedure
COM_IESA.txt	Interprets the IEC 101/104 application commands.
COM_IESEI.txt	Sends an end-of-initialization message.
COM_IESS.txt	Interprets the IEC 101/104 system commands.
COM_INIT.txt	Initializes the selected Trip Signals.
COM_LOG.txt	Writes data to log files.
COM_NETINI.txt	Initializes the protocol converters in NET.
COM_NETINI_H.txt	Initializes the protocol converters in NET after switchover.
COM_PNDIND.txt	Terminates response indications after a timeout.
COM_RDDATA.txt	Performs a read-of-user-data command.
COM_RDGEN.txt	Reads application and system parameters.
COM_RDXREF.txt	Reads cross-reference information.
COM_RESPRC.txt	Performs a reset process command.
COM_RESSEL.txt	Resets selection flags of CPI commands.
COM_REVDTA.txt	COM500*i* revision information.
COM_RPSCR.txt	Creates a line and station for an RP 570 slave.
COM_RPSDI.txt	Sets the DI attribute of an RP 570 slave station.
COM_RPSFT.txt	Loads FTABs to an RP 570 slave.
COM_RPSS.txt	Interprets RP 570 system messages.
COM_RPSY.txt	Marks the RP 570 slave as synchronized.
COM_SUSHOST.txt	Handles connection events from Host application.
COM_SUSNOD.txt	Executes command procedure COM_SUSSTA, when connection to a LAN node is lost.
COM_SUSSTA.txt	Sends the signals connected to a suspended station.
COM_TRPSPA.txt	Receives and sends transparent SPA messages.
COM_USAI.txt	Sends analog data to the NCC.
COM_USAO.txt	Sends analog output status to the NCC.
COM_USBI.txt	Sends binary input data to the NCC.
COM_USBO.txt	Sends binary output status to the NCC.
COM_USBS.txt	Sends bit stream data to the NCC.
COM_USDB.txt	Sends double binary data to the NCC.
COM_USDI.txt	Sends digital input data to the NCC.
COM_USFT.txt	Sends a file to the NCC.
COM_USOE.txt	Sends OPC event data to the NCC.
COM_USOXR.txt	Decodes upstream direction cross-reference data.
COM_USPC.txt	Sends pulse counter data to the NCC.
COM_USXREF.txt	Decodes monitoring direction cross-reference data.

6.7.4 Text files for object creation and other purposes

File	Description
COM_COM.txt	Source file for the APL_COM.txt definition text
COM_START.txt	Makes definitions at application start-up.
COM_UPDT.txt	Updates the application from an older COM500*i*

Table continues on next page

File	Description
COM_APLOBJ.txt	Creates COM500*i* application objects.
COM_OBJTXT.txt	Contains the texts for the COM500*i* command procedures and event channels.
APL_COM.txt	Stores application start definitions for COM500*i*.
ATTR_COM.txt	Stores process object attribute definitions.

Index

Hitachi ABB Power Grids
Grid Automation Products
PL 688
65101 Vaasa, Finland

https://hitachiabb-powergrids.com/microscadax

Scan this QR code to visit our website

1MRK 511 468-UEN