**KAESER**
**KOMPRESSOREN**

# Technical Description

# SIGMA AIR MANAGER

*S5-Compatible Communication with
SIMATIC S5 and S7
Programmable Logic Controllers*

GL-No.: *7_7760_0-PLC_01E*

## Contents

# 1   General

The **SEND/RECEIVE option package** is conditional for the control of certain functions in SIGMA AIR MANAGER in networked peer to peer communications (remote control), for example by a PC, a workstation or a control centre. In addition, the compressed air system's process data are transferred to the networked communications peer via the SEND/RECEIVE interface and are available there for visualisation or monitoring.

The transfer of control and process data between SIGMA AIR MANAGER and the communications peer takes place via the SEND/RECEIVE communications protocol on an Ethernet and TCP/IP with RFC 1006 (ISO transport connection, ISO on TCP connection).

The SEND/RECEIVE interface was developed by Siemens to communicate with an S5 PLC via Ethernet and is therefore described as being **S5 compatible**.

To create the communications link the SEND/RECEIVE option must be enabled on SIGMA AIR MANAGER by entering the registration code provided by KAESER. When this has been done the SEND/RECEIVE protocol on SIGMA AIR MANAGER can be activated by selecting the *FETCH/WRITE* or *SEND/RECEIVE* specification.  The communications peer (PC) then has write access (control data) and read access (process data) to **defined, released** S5 data blocks in SIGMA AIR MANAGER.

It is especially the configuration of S5-compatible communication between SIGMA AIR MANAGER and the SIMATIC S5 and SIMATIC S7 programmable logic controllers that is described in detail in this document and explained with the aid of example projects.

This documentation is aimed exclusively at qualified project engineering and/or commissioning personnel with corresponding expertise in dealing with the programmable logic controllers and their programming described here.

# 2   System Requirements

## 2.1   SIGMA AIR MANAGER

The optional SEND/RECEIVE package can only be used with SIGMA AIR MANAGER's Ethernet interface and not with the RS 232 or modem link.

**SEND/RECEIVE can only run on SAM 4/4, 8/4 and 16/8 models and not on the older 4/4SAM model that is not fitted with an Ethernet interface.**

A further condition for SEND/RECEIVE is that SIGMA AIR MANAGER is provided with certain software versions. Old software must be updated.

**SEND/RECEIVE can only run if SIGMA AIR MANAGER's operating system is version 1.02 (120) or later and the KAESER software is version STEP5 v0.20 or later.**

## 2.2   Communications Peer

Communication with SIGMA AIR MANAGER requires the existence or implementation of the protocol for S5-compatible communication (RFC 1006 protocol) in the communications peer.

Also, from a communications control point of view, the communications peer has to represent an **active subscriber** that sends *FETCH/WRITE* and *SEND/RECEIVE* communications commands.

The communications peer has to be networked with SIGMA AIR MANAGER via Ethernet and operate in the same network. The standard system requirements for TCP/IP networks have to be conformed with.

# 3   Limitations

**Only one S5-combatible communication link can be made with a communications peer** from SIGMA AIR MANAGER at any time (**point-to-point connection**). Only when the existing link is disconnected can another communications peer create a new link. This means that redundant systems with several communications peers that can simultaneously communicate with SIGMA AIR MANAGER cannot be set up.

The link is **permanently implemented** in SIGMA AIR MANAGER and **cannot be changed**.

In communication terms, SIGMA AIR MANAGER only represents a **passive subscriber** that accepts and carries out communications commands (*FETCH/WRITE or SEND/RECEIVE*) from active peers.

SIGMA AIR MANAGER supports a **user data record of maximum 240 data bytes** (120 data words) per data message. Generally, this limitation has to be kept to by the communications peer, for example, during the execution of a FETCH command.

Commensurate with the general SEND/RECEIVE specification, only one predetermined data area from an S5 data block can be written and read per *SEND/RECEIVE* connection. With SIGMA AIR MANAGER only the S5 data block **DB34** may be **written** (see chapter 5.1 and 5.2) and only the S5 data block **DB31** (see chapter 5.1) as well as data blocks DB120-DB135 (SIGMA CONTROL Compr.1 – Compr.16, see chapter 5.3) may be **read**.

With *FETCH/WRITE*, it is possible to **read** all S5 data blocks containing **process data** (see chapter 5.3). However, only the S5 data block **DB34** may be **written** in this case (see chapter 5.1).

SIGMA AIR MANAGER **does not support the ISO transport protocol**.

The maximum transfer rate with SEND/RECEIVE communication is **10 Mbit/s**.

# 4 Initial Start-Up

## 4.1 SIGMA AIR MANAGER

First, the SEND/RECEIVE option package has to be registered at KAESER. To do this, the "Release Code Request" form supplied is to be filled out fully by the customer and faxed or e-mailed to KAESER. KAESER then registers SEND/RECEIVE and grants a single license for the customer's SIGMA AIR MANAGER according to the license conditions attached, in that KAESER provides the customer with the **registration code** (release code). The registration code is needed to unlock the SEND/RECEIVE software on SIGMA AIR MANAGER. System requirements must be borne in mind (see 2). The registration code is to be entered in SIGMA AIR MANAGER once only. The appropriate menu option is accessed from the main menu.

*[main menu] ➢ [settings] ➢ [system] ➢ [communication] ➢ [SEND/RECEIVE].*

If the registration code has been entered correctly, the *FETCH/WRITE* or *SEND/RECEIVE* specification can be activated in the same menu option. If needed, SEND/RECEIVE can be deactivated here at any time. The specification *SEND/RECEIVE* has the options „*SAM*" and „*SAM+SC*". By choosing the option „*SAM*" only the process image of the SIGMA AIR MANAGER will be transferred. By choosing the option „*SAM+SC*" additionally the process images of the SIGMA CONTROL compressors linked to SIGMA AIR MANAGER will be transferred.

As communication via SEND/RECEIVE is based on Ethernet and TCP/IP, SIGMA AIR MANAGER and the communications peer have to be networked via Ethernet and exist in the same TCP/IP network. In addition, SIGMA AIR MANAGER's IP address, subnet mask and, if required, the IP address of a router can be set up. The appropriate menu option is accessed from the main menu:

[*main menu] ➢ [settings] ➢ [system] ➢ [interfaces] ➢ [Ethernet].*

**SIGMA AIR MANAGER has to be restarted to activate the SEND/RECEIVE and the TCP/IP settings that were made.** Switch the power supply to SIGMA AIR MANAGER off and then on again (using the cutout in the control cabinet or the user's power switch).

**The remote control functions via SEND/RECEIVE are only effective if SIGMA AIR MANAGER is running in remote control mode. If remote control mode is ended, the current operating mode (OFF/ON, Manual/Automatic, Clock OFF/ON and compressor selection) is retained and the local settings are effective again.**

Remote control mode is switched on or off with the REMOTE key and indicated by the LED (see SIGMA AIR MANAGER service manual).

## 4.2 The SIMATIC S5 PLC as a Communication Peer

### 4.2.1 Preparatory work

Communication module CP1430TCP is purpose-built for the link-up between SIGMA AIR MANAGER and the SIMATIC S5 station. It makes possible the networking of the S5-115U/H, S5-135U or S5-155U/H systems with other communications peers (e.g. SIGMA AIR MANAGER) via Ethernet and, in addition, supports the RFC 1006 protocol (S5-compatible communication) with the *SEND/RECEIVE* and *FETCH/WRITE* specifications. This is why, depending on the application in question, the *SEND/RECEIVE* or *FETCH/WRITE* specification has to be activated.

The appropriate SIGMA AIR MANAGER menu option is accessed from the main menu:

*[main menu]* ➢ *[settings]* ➢ *[system]* ➢ *[communication]* ➢ *[SEND/RECEIVE].*

The **"STEP5"** software package from Siemens must be installed on a PC to allow setting up and/or programming of the SIMATIC S5 programmable logic controller. The **SINEC NCM COM 1430 TCP** set-up software is required to set up and program the CP1430TCP communications module. A TTY line link is needed to transfer the settings entered in the S5-CPU and CP1430 TCP.

**As this description is only meant to be an explanation of a simplified procedure for initial start-up of S5-compatible communication between a SIMATIC S5 station and a SIGMA AIR MANAGER, detailed areas of the software will only be described with regard to their necessity.**

Further information on the **"STEP5"** and **SINEC NCM COM 1430 TCP** programming software can be found in the appropriate user manuals.

Additional information on the S5 hardware and software and its accessories can be found in the **Siemens Catalogue ST50**.

### 4.2.2 Setting up the S5 station with the SEND/RECEIVE connection

In the main, this involves the following steps:

- Configuration of the CP1430 TCP communications module

- Setting up the **active** communications link

- Embedding the function module in the STEP5 program

The individual steps are explained below: The images shown refer to an S5 station fitted with an S5-115U with the CPU944B and the CP1430TCP communications module. Settings that lie within the judgment of the user are not explained further.

#### 4.2.2.1 Configuration of the CP1430 TCP communications module

The CP1430TCP communications module is the connecting link providing communication between the S5-CPU and the industrial Ethernet-linked SIGMA AIR MANAGER. It exchanges messages with the S5-CPU through a common interface (dual port RAM). Unambiguous access to/from the communication peers in the Ethernet network is dependent on the IP-address. The **SINEC NCM COM 1430 TCP** program assists in the stipulation of the area of the interface needed in the dual port RAM and in the IP address set-up. First of all, the **STEP5 program** has to be started. Next, load the STEP5 project software to allow the project directory to be determined. Afterwards, click on **Change▶ further… F9** and the user interface for **Other SIMATIC S5 programs** appears. The **SINEC NCM COM 1430 TCP** program is started by marking **SINEC NCM COMs** and then clicking **<Start>**.
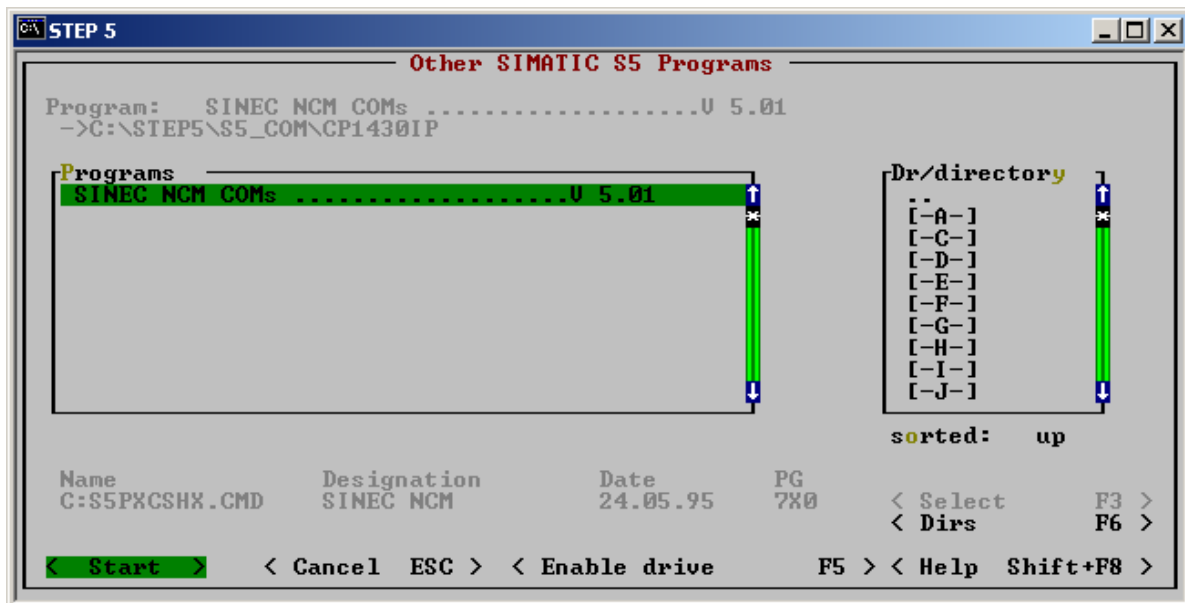


Image 1: "SINEC NCM COM1430" program start

The basic settings are accessed with **File ➢ Select**. Here, the input fields **CP1430IP** as **CP type** and **OFFLINE FD** as **Status** are marked for offline setup and then selected with **F8 SELECT**. In addition, the **filename** of the CP1430 setup file (must start with the character **T**) and the working directory, which corresponds with the project directory determined in **STEP5**, is defined. The remaining settings can be ignored here. The settings are entered with **F7 OK** and the main menu appears again.
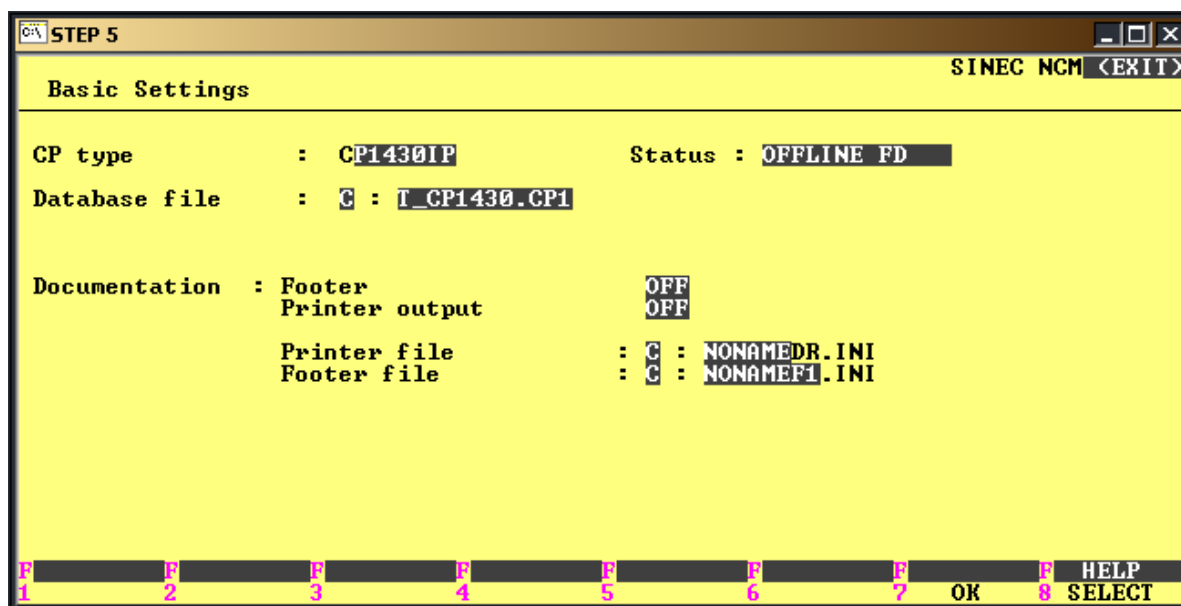
```
STEP 5                                                    _ □ ×
                                               SINEC NCM <EXIT>
   Basic Settings

CP type            : CP1430IP      Status : OFFLINE FD
Database file      : C : T_CP1430.CP1


Documentation  : Footer                OFF
                 Printer output        OFF

                 Printer file    : C : NONAMEDR.INI
                 Footer file     : C : NONAMEF1.INI



F    F     F     F     F     F     F        HELP
1    2     3     4     5     6     7  OK   8 SELECT
```

Image 2: Basic settings for CP1430TCP

The dialog for the identification setup of CP1430TCP in the Ethernet is accessed with menu options **Edit ➢ CP Init**. A net-wide unambiguous address has to be entered as a physical address (**MAC address**). If the CP1430TCP is operated in a local Ethernet then the default value can be retained. A number that can be divided by four is entered as the **Base interface number**. In principle, this represents the address with which an unambiguous memory area in the Dual Port Ram of the CP1430 TCP is determined. The remaining settings can be ignored here. The settings are entered with **F7 OK** and the main menu appears again.

```
STEP 5                                                          _ □ ✕
                                          COM1430TCP/IP V2.07 ⟨EXIT⟩
CP Basic Initialization        Source:  C:T_CP1430.CP1 ⟨OFFLINE⟩

     MAC address ⟨HEX⟩       :  080006010001
   SIMATIC details    :

     Base SSNR             :  8

     Interface
     communication ⟨P/ /B⟩ :  P        1        2        3
     SSNR offset           :  0        1        2        3

   Informative parameters:

     Card type            :            Database size   :  64  Kb
     Module ID            :  CP1430IP

     Firmware version     :            Date created    :  030120
     Plant designation    :  CP1430<->SAM

F          F         F         F        F        F        F    F  HELP
1          2         3         4        5        6        7    OK 8 SELECT
```
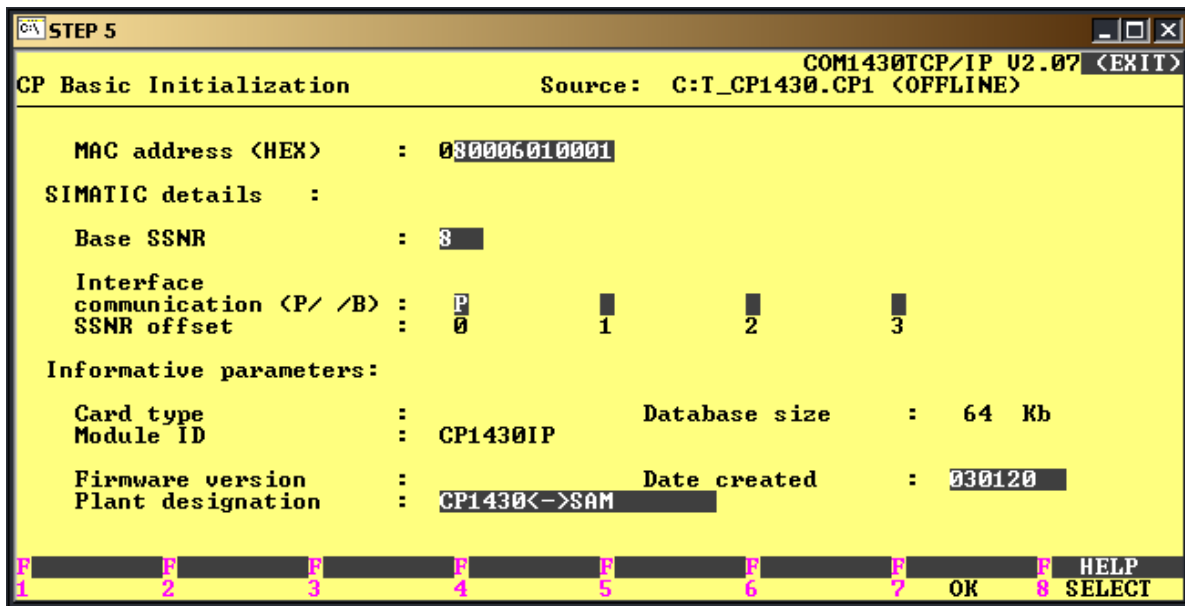
Image 3: Initialization of the CP1430TCP

The **TCP/IP initialization** dialog is then accessed with **Edit ➢ TCP/IP Init**. The settings for the local TCP/IP specifications of the CP1430TCP are entered here. Any character sequence (max. 25 characters) can be entered as a host name, although this is not absolutely necessary. The IP-address for the CP1430TCP must exist in the same subnet as the link partner (SIGMA AIR MANAGER). The subnet setting in SIGMA AIR MANAGER under

*[main menu] ➢ [settings] ➢ [system] ➢ [interfaces] ➢ [Ethernet]*;

must also be entered here. If the Ethernet link to SIGMA AIR MANAGER is via a gateway, then the address of the default gateway has to be entered as well. The settings are entered with **F7 OK** and the main menu appears again.
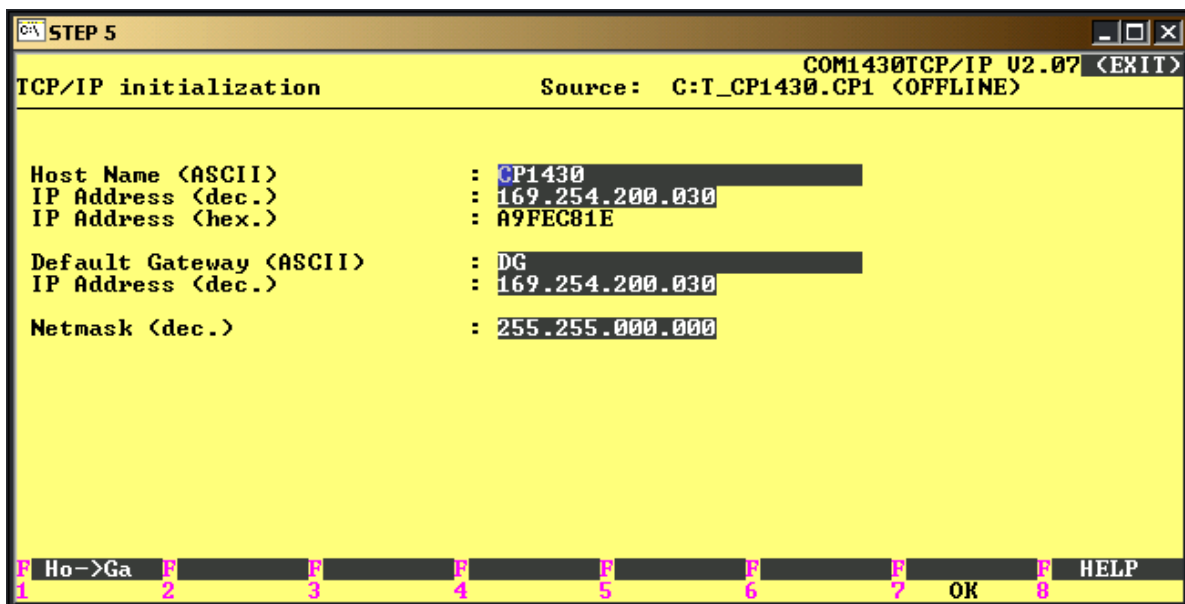
```
STEP 5                                                          _ □ ✕
                                          COM1430TCP/IP V2.07 ⟨EXIT⟩
TCP/IP initialization          Source:  C:T_CP1430.CP1 ⟨OFFLINE⟩


   Host Name ⟨ASCII⟩         :  CP1430
   IP Address ⟨dec.⟩         :  169.254.200.030
   IP Address ⟨hex.⟩         :  A9FEC81E

   Default Gateway ⟨ASCII⟩   :  DG
   IP Address ⟨dec.⟩         :  169.254.200.030

   Netmask ⟨dec.⟩            :  255.255.000.000

F Ho->Ga  F         F         F        F        F        F    F  HELP
1          2         3         4        5        6        7    OK 8
```

Image 4: TCP/IP initialization of the CP1430TCP

### 4.2.2.2  *Setting up the active communications connection*

As only **one** S5 compatible communications connection from SIGMA AIR MANAGER can exist at one time, only one bi-directional connection is required with the *SEND/RECEIVE* specification. This is accessed from the main menu with **Edit ➢ Connections ➢ Transport Conn. (RFC1006)** in the dialog **Transport Connection (TCP/RFC1006).** New transport connection jobs are entered with **F3 INPUT**. A selection box can be displayed with **F8 SELECT** on all input fields for which a selection list is available. The selection is entered with the ENTER key.

The **SSNR offset** parameter, in conjunction with the basic interface number (see chapter 5.2.2.1), determines the final interface number for the communications channel. The default value **0** should be kept.

The **ANR** parameter represents a job number. Together with the CP1430TCP interface number it identifies a communications job if an S5 standard function module (function block) – provided with the appropriate formal *SSNR* und *A-NR* parameters – is called up in the STEP5 user program.

The **Job type** determines whether it is a SEND, RECEIVE or FETCH job. In this case, select **SEND**.

The **Active/passive (A/P)** parameter indicates whether or not the transport connection has initiated the connection set-up. This is to be set to **A.**

With regard to the *SEND/RECEIVE* specification, the **Read/write (Y/N)** parameter determines whether or not the default parameter comes from the communications peer. This is to be set to **N.**

Enter **2** (1 of 2) for the **Number of jobs per TSAP** parameter. This means that a TSAP pair is assigned two jobs (TSAP = Transport Service Access Point), i.e., the TSAP of the local and the TSAP of the remote communications peer. The second job will be the RECEIVE job, which is the next to be setup.

The **S-KAESER** ID must be entered as the **TSAP (ASC)** of the local communications peer (CP1430 TCP) in the column **Loc. Parameters:**. The entries for **TSAP (hex)** and **TSAP length** are added automatically.

The settings for the remote communications peer (SIGMA AIR MANAGER) are entered in the column **Rem. Parameters:**. The IP address entered in the SIGMA AIR MANAGER under

*[main menu] ➢ [settings] ➢ [system] ➢ [interfaces] ➢ [Ethernet]*

is to be entered as the **IP address**. If an identifier with its IP address is assigned to the communications peer in the dialog **Hostnames** with **Edit ➢ Hostnames**, this can be entered under **Hostname.** The ID **S-KAESER** must be entered as the **TSAP (ASC)**. The entries for **TSAP (hex)** and **TSAP length** are added automatically.

```
STEP 5                                                          _ □ ✕
                                          COM1430TCP/IP U2.07 <EXIT>
Transport Connection <TCP/RFC1006>   Source:  C:T_CP1430.CP1 <OFFLINE>

 SSNR offset        : 0              ANR                    :  1

 Job type           : SEND           Active/passive <A/P> : A

 Read/write <Y/N>   : N              TCP Keep-Alive <Y/N> : Y

 Number of jobs per TSAP      :  1  of  2

 Transport addresses:
 Loc. parameters:                     Rem. parameters:

                                      Hostname    : SAM
                                      IP address : 169.254.200.014
 TSAP <ASC> : S-KAESER                TSAP <ASC> : S-KAESER
 TSAP <HEX> : 53 2D 4B 41 45 53 45 52 TSAP <HEX> : 53 2D 4B 41 45 53 45 52
 TSAP length: 8                       TSAP length: 8


 F      F      F         F          F      F        F          F  HELP
 1  + 1  2  - 1  3  INPUT  4 DELETE  5      6        7    OK   8  SELECT
```
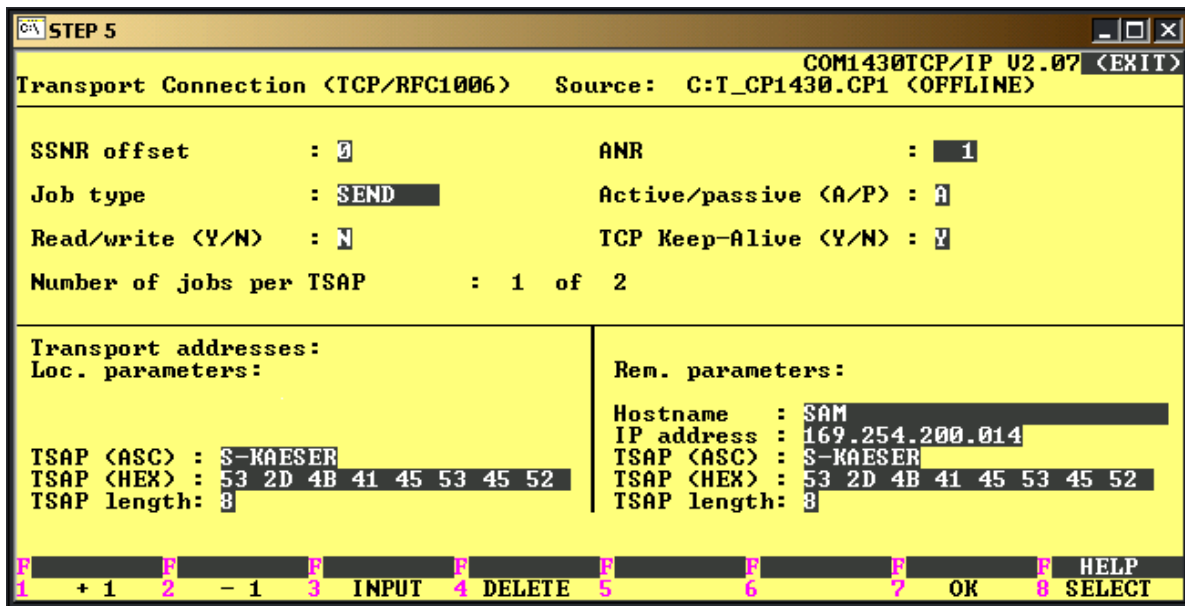
Image 5: Setting up a SEND job

Save the planned job with **F7 OK**. As the value **2** was selected as the setting for the **Number of jobs per TSAP**, setup of the second job (RECEIVE job) is expected straightaway. The job number **ANR** is automatically increased by one. Both this preset value and that of the **SSNR offset** parameter are adopted. **RECEIVE** is now selected for the **Job type** parameter. Save the planned job with **F7 OK**.
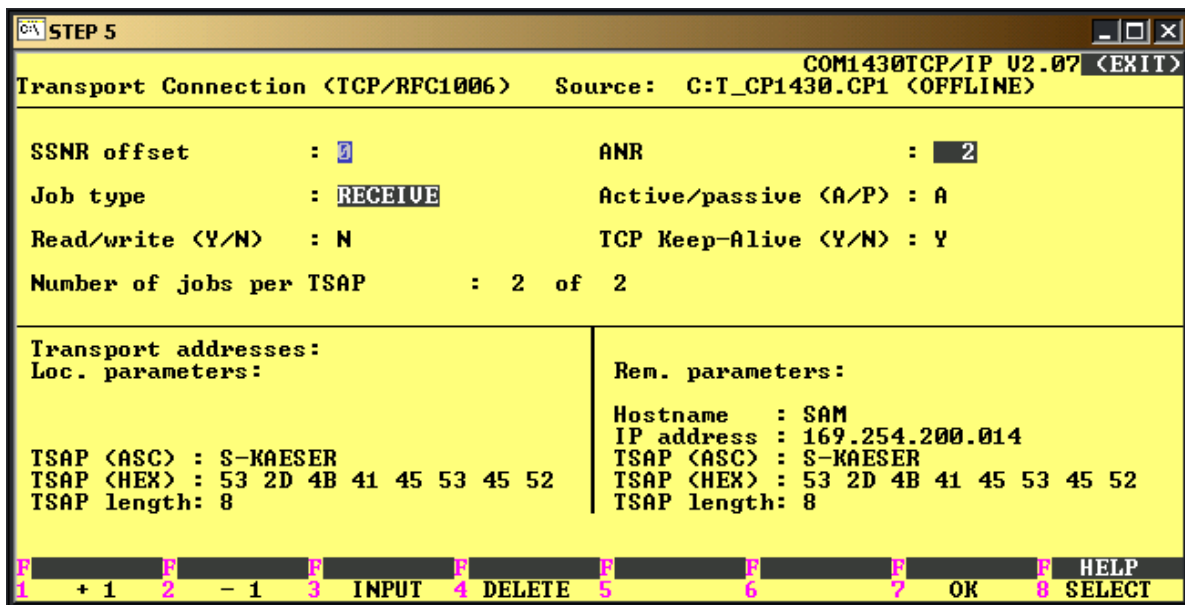Confirm the overwrite job query with **F1 YES**.

```
STEP 5                                                          _ □ ✕
                                          COM1430TCP/IP U2.07 <EXIT>
Transport Connection <TCP/RFC1006>   Source:  C:T_CP1430.CP1 <OFFLINE>

 SSNR offset        : 0              ANR                    :  2

 Job type           : RECEIVE        Active/passive <A/P> : A

 Read/write <Y/N>   : N              TCP Keep-Alive <Y/N> : Y

 Number of jobs per TSAP      :  2  of  2

 Transport addresses:
 Loc. parameters:                     Rem. parameters:

                                      Hostname    : SAM
                                      IP address : 169.254.200.014
 TSAP <ASC> : S-KAESER                TSAP <ASC> : S-KAESER
 TSAP <HEX> : 53 2D 4B 41 45 53 45 52 TSAP <HEX> : 53 2D 4B 41 45 53 45 52
 TSAP length: 8                       TSAP length: 8


 F      F      F         F          F      F        F          F  HELP
 1  + 1  2  - 1  3  INPUT  4 DELETE  5      6        7    OK   8  SELECT
```

Image 6: Setting up a RECEIVE job

Now, the CP1430TCP is set to the STOP status with **CP-Functions ➢ Stop** so that the file can be transferred to the CP1430TCP with **Transfer ➢ FD→CP**. All job data blocks are loaded into the CP1430TCP with **F2 TOTAL**. Afterwards, set the CP1430TCP to the RUN status again with **CP-Functions ➢ Start**.

Finally, exit the **SINEC NCM COM 1430 TCP** program with **File ➢ Exit** and return to the **STEP5** program with **<Cancel ESC>**.

### 4.2.2.3   Embedding the function modules in the STEP5 user program

After configuring the CP1430TCP, the S5 standard function modules (function blocks) have to be embedded in the STEP5 user program. The function blocks, integrated in the operating system of the S5 CPU and listed below, control, among other things, the exchange of data between the CP1430TCP and the S5 CPU.

> FB249 - SYNCHRONOUS
> FB244 - SEND
> FB245 - RECEIVE

The SYNCHRONOUS FB, called up from the start-up organization block (OB21 or OB22) of the STEP5 application program, initiates definition of the interface on the communication module for communication between the CP1430TCP and the S5-CPU when the complete S5 PLC system has run up. This synchronization is necessary to make sure that all other function blocks can function correctly.

The FB SEND is needed in the S5 application program primarily for transfer of data from the S5 CPU to the CP1430TCP (**SEND ALL with the job number 0**).
In order to be able to actually carry out an active **SEND** connection job planned in the CP1430TCP, the SEND FB has to be called up additionally once per job with the appropriate parameters (**SEND DIRECT with the job number n**).

As the communications peer initially sends the data to the CP1430TCP, the RECEIVE FB (**RECEIVE ALL with the job number 0**) has to be called up for forwarding to the S5 CPU. The actual initiation, however, first takes place through a further call-up of the RECEIVE FB (**RECEIVE DIRECT with the job number n**). This has to be provided with the appropriate parameters corresponding with the **RECEIVE** connection job that was set up in the CP1430 TCP.

A detailed description of the S5 standard function modules and their various call-up variants is to be found in the equipment manual for the automation equipment used, e.g. "SIMATIC S5 Programmable Logic Controller S5-115U". The STEP5 example program in **Annex A** can be used as a template for the compilation of the STEP5 user program.

### 4.2.3 Setting up the S5 station with the FETCH/WRITE connection

In the main, this involves the following steps:

- Configuration of the CP1430 TCP communications module

- Setting up the **active** communications link

- Embedding the function module in the STEP5 program

The individual setting-up steps are explained below: The images shown for illustration refer to an S5 station fitted with an S5-115U with the CPU944B and the CP1430TCP communications module. Settings that lie within the judgment of the user are not explained further.

#### 4.2.3.1  Configuration of the CP1430 TCP communications module

The procedure is as already described in chapter 4.2.2.1.

#### 4.2.3.2  Setting up the active communications link

As only **one** S5 compatible communications connection from SIGMA AIR MANAGER can exist at one time, only one bi-directional connection is required with the *SEND/RECEIVE* specification. This is accessed from the main menu with **Edit ≻ Connections ≻ Transport Conn. (RFC1006)** in the dialog **Transport Connection (TCP/RFC1006).** New transport connection jobs are entered with **F3 INPUT**. A selection box can be displayed with **F8 SELECT** on all input fields for which a selection list is available. The selection is entered with the ENTER key.

The **SSNR offset** parameter, in conjunction with the basic interface number (see chapter 5.2.2.1), determines the final interface number for the communications channel. The default value **0** should be kept.

The **ANR** parameter represents a job number. Together with the CP1430TCP interface number it identifies a communications job if an S5 standard function module (function block) – provided with the appropriate formal *SSNR* und *A-NR* parameters – is called up in the STEP5 user program.

The **Job type** determines whether it is a SEND, RECEIVE or FETCH job. In this case, select **SEND**.

The **Active/passive (A/P)** parameter indicates whether or not the transport connection has initiated the connection set-up. This is to be set to **A.**

With the *SEND/RECEIVE* specification the **Read/write (Y/N)** parameter determines whether or not the default parameter comes from the communications peer. This is to be set to **Y.**

Enter **1** (1 of 1) for the **Number of jobs per TSAP** parameter. This means that a TSAP pair (TSAP = Transport Service Access Point), i.e., the TSAP of the local and the TSAP of the remote communications peer, is assigned one job.

The ID **W-KAESER** must be entered as the **TSAP (ASC)** of the local communications peer (CP1430 TCP) in the **Loc. Parameters** column. The entries for **TSAP (hex)** and **TSAP length** are added automatically.

The settings for the remote communications peer (SIGMA AIR MANAGER) are entered in the column **Rem. Parameters:**. The IP address entered in the SIGMA AIR MANAGER under

*[main menu] ➤ [settings] ➤ [system] ➤ [interfaces] ➤ [Ethernet]*

is to be entered as the **IP address**. If an identifier with its IP address is assigned to the communications peer in the dialog **Hostnames** with **Edit ➤ Hostnames**, this can be entered under **Hostname.** The ID **W-KAESER** must be entered as the **TSAP (ASC)**. The entries for **TSAP (hex)** and **TSAP length** are added automatically.



Image 7: Setting up a WRITE job

Save the planned job with **F7 OK**. Confirm the overwrite job query with **F1 YES**.

Now create a new job with **F3 INPUT**.

Accept the **SSNR offset** parameter with its default value.

Set the **ANR** parameter with the higher job number proposed. As previously mentioned, this job number serves ultimately, together with the interface number of the CP1430 TCP, the unambiguous allocation of the communication job. In the STEP5 application program the *SSNR* and *A-NR* formal parameters have to be set with these values when the S5 standard function module (function block) is called up.

Enter the receive job type **FETCH** in the **Job type** field.

The **Active/passive (A/P)** parameter indicates whether or not the transport connection has initiated the connection set-up. This is to be set to **A.**

Set the **Read/write (Y/N)** parameter to **Y**.

Enter **1** (1 of 1) for the **Number of jobs per TSAP** parameter.

The ID **F-KAESER** must be entered as the **TSAP (ASC)** of the local communications peer (CP1430 TCP) in the **Loc. Parameters** column. The entries for **TSAP (hex)** and **TSAP length** are added automatically.

The settings for the remote communications peer (SIGMA AIR MANAGER) are entered in the column **Rem. Parameters:**. Under **IP address** the address entered in the SIGMA AIR MANAGER menu under

 *[main menu] ➤ [settings] ➤ [system] ➤ [interfaces] ➤ [Ethernet]*

is to be entered. If an identifier with its IP address is assigned to the communications peer in the dialog **Hostnames** with **Edit ➢ Hostnames**, this can be entered under **Hostname.** The ID **F-KAESER** must be entered as the **TSAP (ASC)**. The entries for **TSAP (hex)** and **TSAP length** are added automatically.
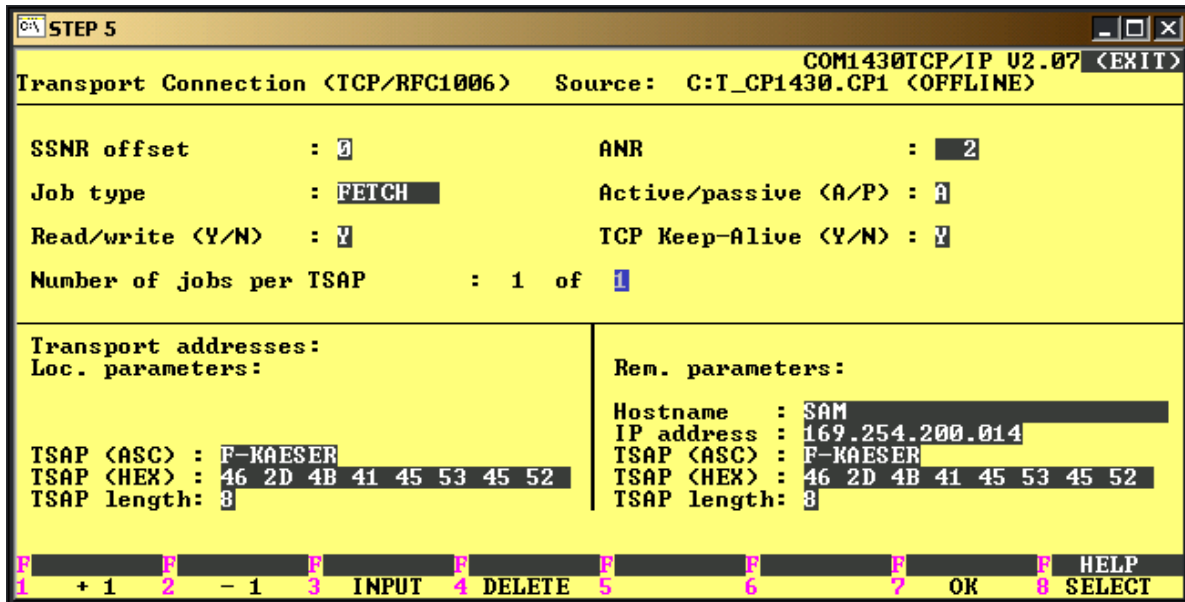
```
STEP 5                                                                  _ □ ✕
                                                    COM1430TCP/IP V2.07 <EXIT>
Transport Connection <TCP/RFC1006>    Source:  C:T_CP1430.CP1 <OFFLINE>

  SSNR offset        : 0           ANR                  :   2

  Job type           : FETCH       Active/passive <A/P> : A

  Read/write <Y/N>   : Y           TCP Keep-Alive <Y/N> : Y

  Number of jobs per TSAP      :  1  of  1

  ┌──────────────────────────────────────────────────────────────────────┐
  │ Transport addresses:                                                   │
  │ Loc. parameters:                    Rem. parameters:                   │
  │                                                                        │
  │                                     Hostname    : SAM                  │
  │                                     IP address : 169.254.200.014       │
  │ TSAP <ASC> : F-KAESER               TSAP <ASC> : F-KAESER              │
  │ TSAP <HEX> : 46 2D 4B 41 45 53 45 52  TSAP <HEX> : 46 2D 4B 41 45 53 45 52│
  │ TSAP length: 8                      TSAP length: 8                      │
  └──────────────────────────────────────────────────────────────────────┘

 F         F         F         F         F         F         F         F   HELP
 1   + 1   2   − 1   3   INPUT 4 DELETE 5         6         7     OK   8 SELECT
```

Image 8: Setting up a FETCH job


Now, the CP1430TCP is set to the STOP status with **CP-Functions ➢ Stop** so that the file can be transferred to the CP1430TCP with **Transfer ➢ FD→CP**. All job data blocks are loaded into the CP1430TCP with **F2 TOTAL**. Afterwards, set the CP1430TCP to the RUN status again with **CP-Functions ➢ Start**.
Finally, exit the **SINEC NCM COM 1430 TCP** program with **File ➢ Exit** and return to the **STEP5** program with **<Cancel ESC>**.

### 4.2.3.3  Embedding the function modules in the STEP5 user program

After configuring the CP1430TCP, the S5 standard function modules (function blocks) have to be embedded in the STEP5 user program. The function blocks, integrated in the operating system of the S5 CPU and listed below, control, among other things, the exchange of data between the CP1430TCP and the S5 CPU.

> FB249 - SYNCHRONOUS
> FB244 - SEND
> FB245 - RECEIVE
> FB246 - FETCH

The SYNCHRONOUS FB called up from the start-up organization block (OB21 or OB22) of the STEP5 application program initiates definition of the interface on the communication module for communication between the CP1430TCP and the S5-CPU when the complete S5 PLC system has run up. This synchronization is necessary to make sure that all other function blocks can function correctly.

The FB SEND is needed in the S5 application program primarily for transfer of data from the S5 CPU to the CP1430TCP (**SEND ALL with the job number 0**).
In order to able to actually carry out an active **WRITE** connection job planned in the CP1430TCP, the SEND FB has to be called up additionally once per job with the appropriate parameters (**SEND DIRECT with the job number n**).

As the communications peer initially sends the data to the CP1430TCP, the RECEIVE FB (**RECEIVE ALL with the job number 0**) has to be called up for forwarding to the S5 CPU. The actual initiation, however, first takes place through a further call-up of the FETCH FB (**FETCH DIRECT** with the job number **n**). This has to be provided with the appropriate parameters corresponding with the **FETCH** connection job that was set up in the CP1430 TCP.

A detailed description of the S5 standard function modules and their various call-up variants is to be found in the equipment manual for the automation equipment used, e.g. "SIMATIC S5 Programmable Logic Controller S5-115U". The STEP5 example program in **Annex B** can be used as a template for the compilation of the STEP5 application program.

## 4.3　The SIMATIC S7 PLC as a Communications Peer

### 4.3.1　Preparatory work

Communication modules CP343-1 and CP443-1 are purpose-built for the connection between SIGMA AIR MANAGER and the S7 300 and S7 400 SIMATIC S7 stations. Both modules support S5-compatible communication with *SEND/RECEIVE* specification.

The appropriate SIGMA AIR MANAGER menu option is accessed from the main menu:

*[main menu] ➤ [settings] ➤ [system] ➤ [communication] ➤ [SEND/RECEIVE].*

The software package **"STEP7"** from Siemens must be installed on a PC to allow setting up and/or programming of the SIMATIC S7 station. Furthermore, the project planning PC and the corresponding S7 CPU must be connected via an MPI (Multi Point Interface) cable.

Additional information on the S7 hardware and software needed and its accessories can be found in the **Siemens ST70 Catalogue**.

### 4.3.2　Setting up the S7 station with the SEND/RECEIVE connection

In the main, this involves the following steps:

- Entering the project components

- Arrangement and parameterization of the S7 hardware in the configuration table

- Configuration of the link peer

- Setting up the **active** communications link

- Embedding the function module in the STEP7 program

The individual steps are explained below: The images refer to an S7 300 station with a CPU315-2DP and a CP343-1 communications module. Settings that lie within the judgment of the user are not explained further.

#### 4.3.2.1　Entering the project components

In order to set up the S7 station, the STEP7 setup and programming user interface has to be started with **Start ➤ Simatic ➤ STEP7 ➤ SIMATIC Manager**. Before configuration can be carried out, an S7 project must be either started or a new one made up in **SIMATIC Manager**. A **SIMATIC 300 or 400 station** with an **S7 program**, **Other station** and finally **Industrial Ethernet** as the subnet are inserted into the opened project. Select the opened project by clicking on it before each insertion.

### SIMATIC 300 Station:

For the own S7 program (Rack, PS, CPU, CP343-1, ...): **Insert ⯈ Station ⯈ SIMATIC 400 station** and **Insert ⯈ Program ⯈ S7 program**

For the link peer (SIGMA AIR MANAGER): **Insert ⯈ Station ⯈ Other station**

For an industrial Ethernet network between the SIMATIC 300 station and the link peer: **Insert ⯈ Subnet ⯈ Industrial Ethernet.**
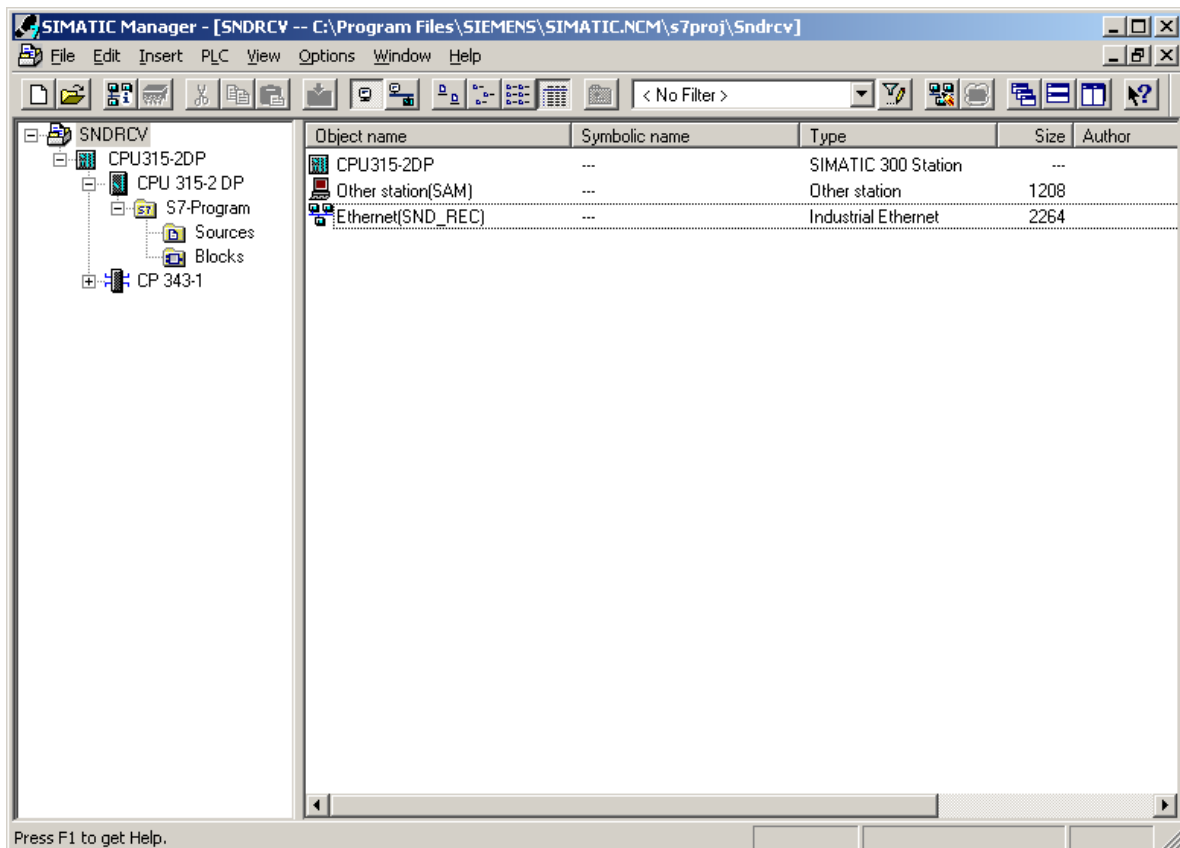
### SIMATIC 400 Station:

For own S7 program (Rack, PS, CPU, CP443-1, ...): **Insert ⯈ Station ⯈ SIMATIC 400 station** and **Insert ⯈ Program ⯈ S7 program**

For the link peer (SIGMA AIR MANAGER): **Insert ⯈ Station ⯈ Other station**

For an industrial Ethernet network between the SIMATIC 400 station and the link peer: **Insert ⯈ Subnet ⯈ Industrial Ethernet.**



Image 9: Overall view of the SIMATIC S7 station with the project components

#### 4.3.2.2 Arrangement and parameterization of the S7 hardware in the configuration table

The S7 hardware components used for the hardware configuration are now arranged on the rack according to their card slots and their properties are determined. To do this, the hardware configuration (HW config) is started by selecting the SIMATIC 300 station and double-clicking **Hardware** (or **Edit ⮞ Open object**).

<u>**SIMATIC 300 Station:**</u>

Insert a RACK 300, a PS-300, a CPU-300 from SIMATIC 300 and the CP Industrial Ethernet CP343-1 from CP-300 with the corresponding order numbers in each case with **Insert ⮞ Insert Object**.

<u>**SIMATIC 400 Station:**</u>

Insert a RACK 400, a PS-400, a CPU-400 from SIMATIC 400 and the CP Industrial Ethernet CP443-1 from CP-400 with the corresponding order numbers in each case with **Insert ⮞ Insert Object**.

The procedure for the configuration of S7 300- or S7 400 modules is described in detail in the "User's Manual for STEP7".
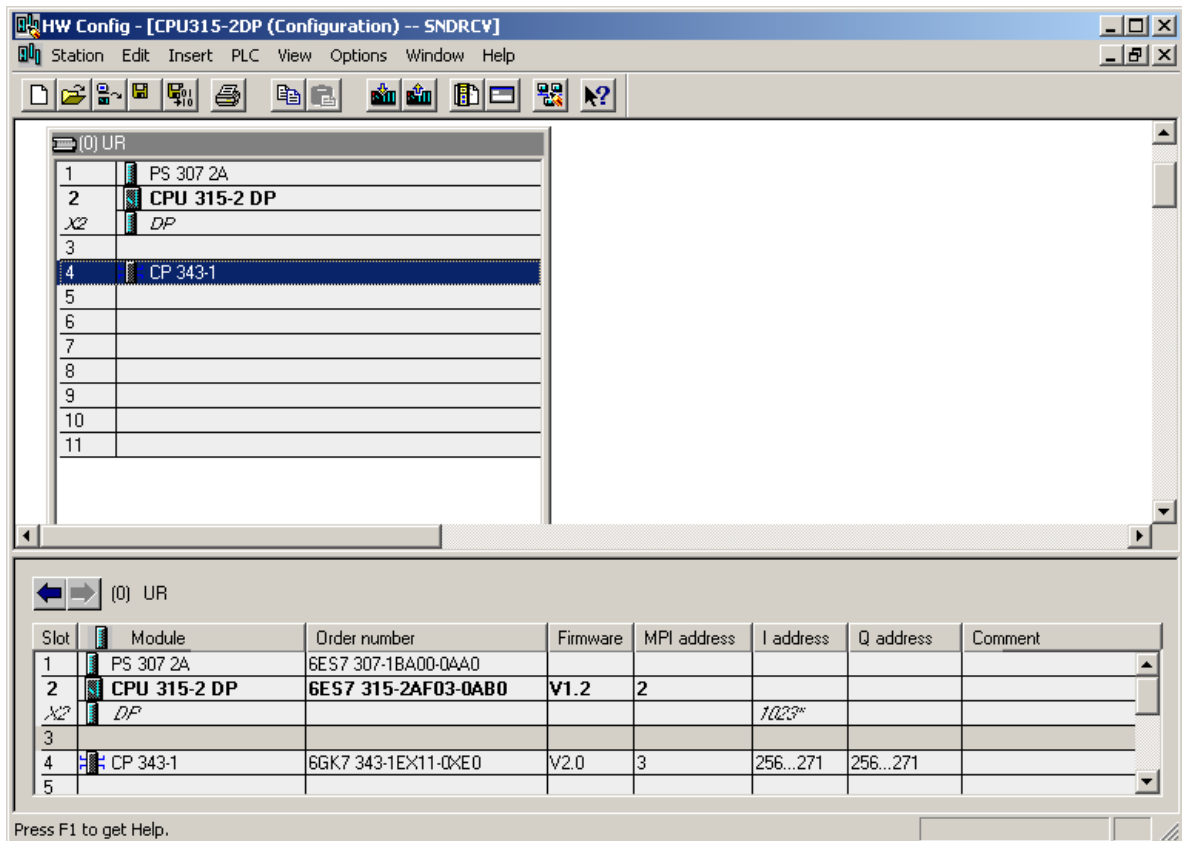


mage 10: Hardware configuration of the SIMATIC S7 station

Now, the CP343-1 or CP443-1 communications module has to be parametrized. To do this start the parameterizing user interface by double clicking on **CP x43-1** in the dialog window for the hardware configuration (HW Config) or by marking the CP and selecting the **Edit ➢ Object Properties** menu option. Select **Properties** in the **General** tab to access the Ethernet user interface with the **Parameters** tab in which the communications processor is given its IP address and the appropriate subnet mask. The IP-address for the CP must exist in the same subnet as the link peer (SIGMA AIR MANAGER). The subnet setting in SIGMA AIR MANAGER under

*[main menu] ➢ [settings] ➢ [system] ➢ [interfaces] ➢ [Ethernet]*;

must also be entered here. If the Ethernet link to SIGMA AIR MANAGER is via a router, then the address of the router has to be entered as well. Also, the CP has to be linked to the industrial Ethernet subnet in the **Subnet** field. This is accomplished by marking the appropriate entry. On newly created Ethernet subnets this is normally designated with *Ethernet(n)*.

The creation of a MAC address is not necessary for SEND/RECEIVE communication.
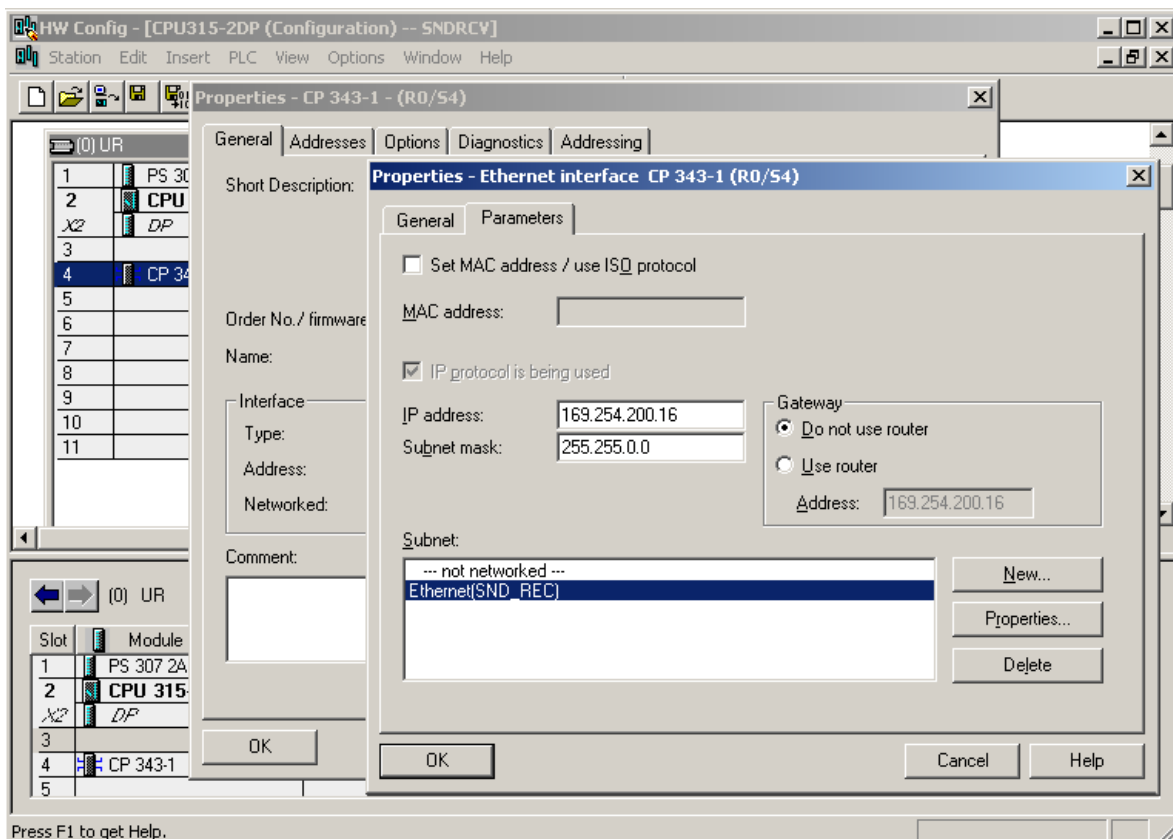


Image 11: Configuration of the S7 communications module

The parameterization of the Ethernet interface is entered with the **OK** button. The input and output start addresses can be noted in the **Address** tab. This is normally predefined by the system. This start address is needed later for provision of the formal parameters *LADDR* when calling up the S7 system function block in STEP7 for send/receive of data in the STEP7 source code.
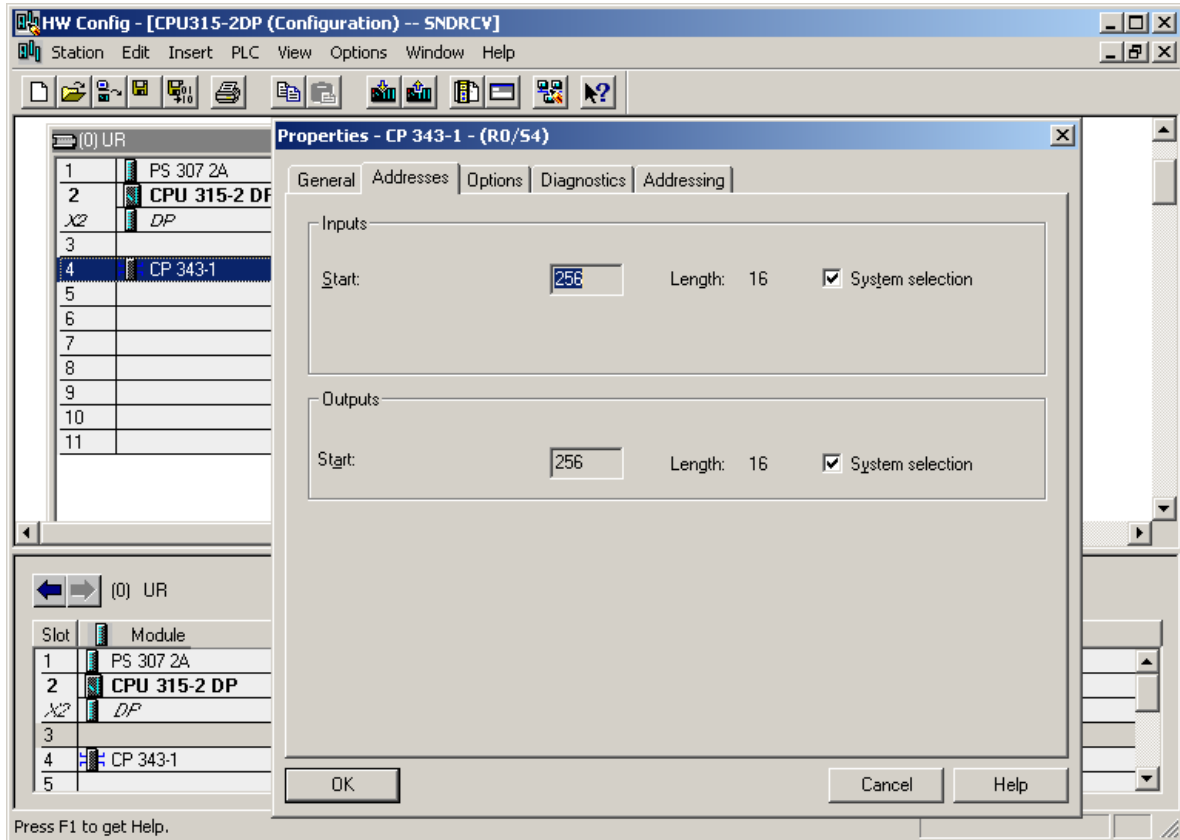


Image 12: Input and Start addresses of the S7 communications module

A data length of more than 240 bytes for processing by the communications module can be chosen in the **Options** tab. Select the setting **TP/ITP at 10Mbps full duplex** as the **Transmission medium/duplex**.

All the other default settings can be accepted. As SIGMA AIR MANAGER supports a **user data record of maximum 240 data bytes** (120 data words) per data message, this setting is of no consequence.
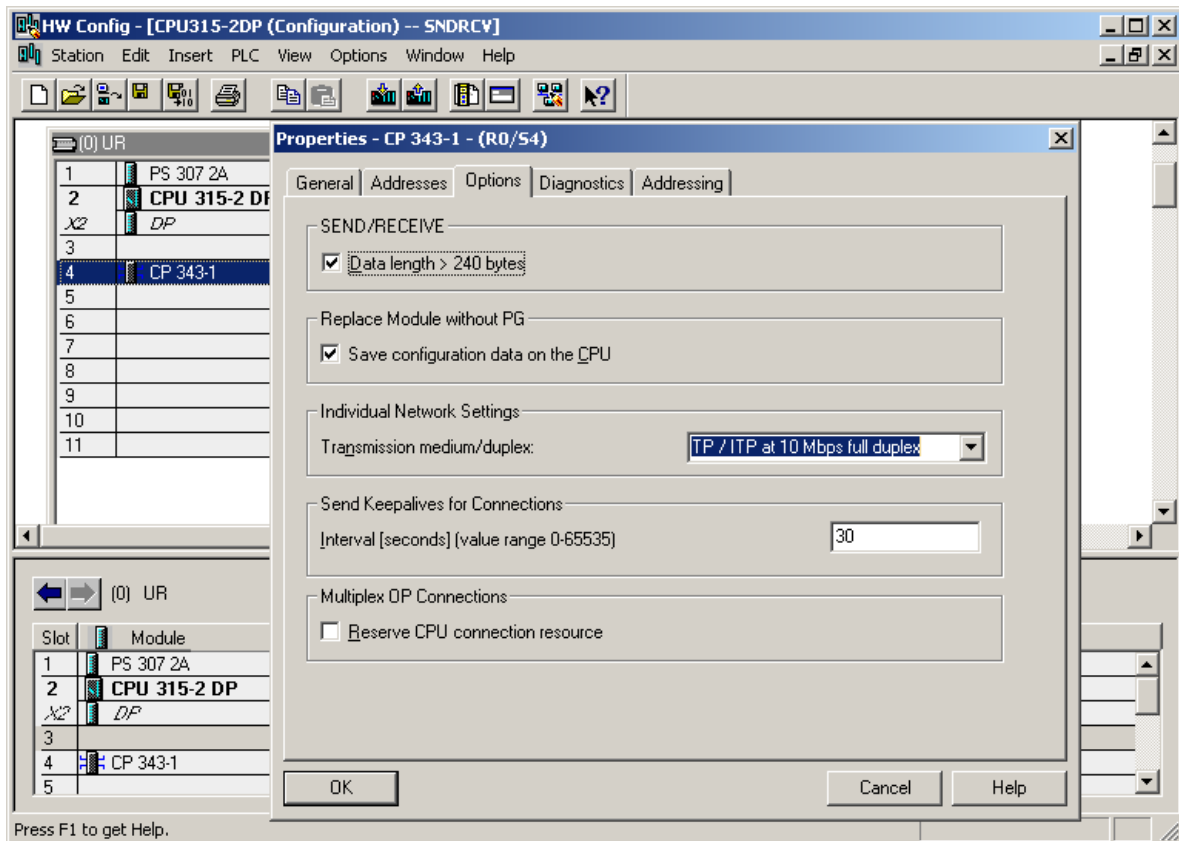
Image 13: Optional settings for the S7 communications module

To close the overall parameterization of the CP343-1 or CP434-1, close the **Properties - CP343-1** or **Properties - CP434-1** dialog by clicking **OK**. Having returned to the hardware configuration (HW Config) window, the hardware configuration for the complete S7 station has to be closed with **Station ➢ Save and Compile**. Finally, the complete hardware configuration has to be transferred into the target module CPU3xx or CPU 4xx with **PLC ➢ Download.**

### 4.3.2.3 *Configuring the communication peer's station*

Assuming that the STEP7 project in SIMATIC Manager is open, the properties of the communication peer's **Other station** now have to be configured by double clicking it (or marking it and selecting with the menu option **Edit ➢ Object Properties**). The interface type is selected in the **Interfaces** tab by clicking **New,** selecting **Industrial Ethernet** and accepting with **OK**. In the popup Ethernet interface properties dialog the IP address and the communication peer's (SIGMA AIR MANAGER) subnet mask are entered. Basically, these settings are the same as those in the SIGMA AIR MANAGER menu

 *[main menu] ➢ [settings] ➢ [system] ➢ [interfaces] ➢ [Ethernet]*

that are already entered. The communications peer has to be linked to the industrial Ethernet subnet in the **Subnet** field. This is accomplished by marking the appropriate entry. Click **OK** twice to return to the SIMATIC MANAGER's main menu.
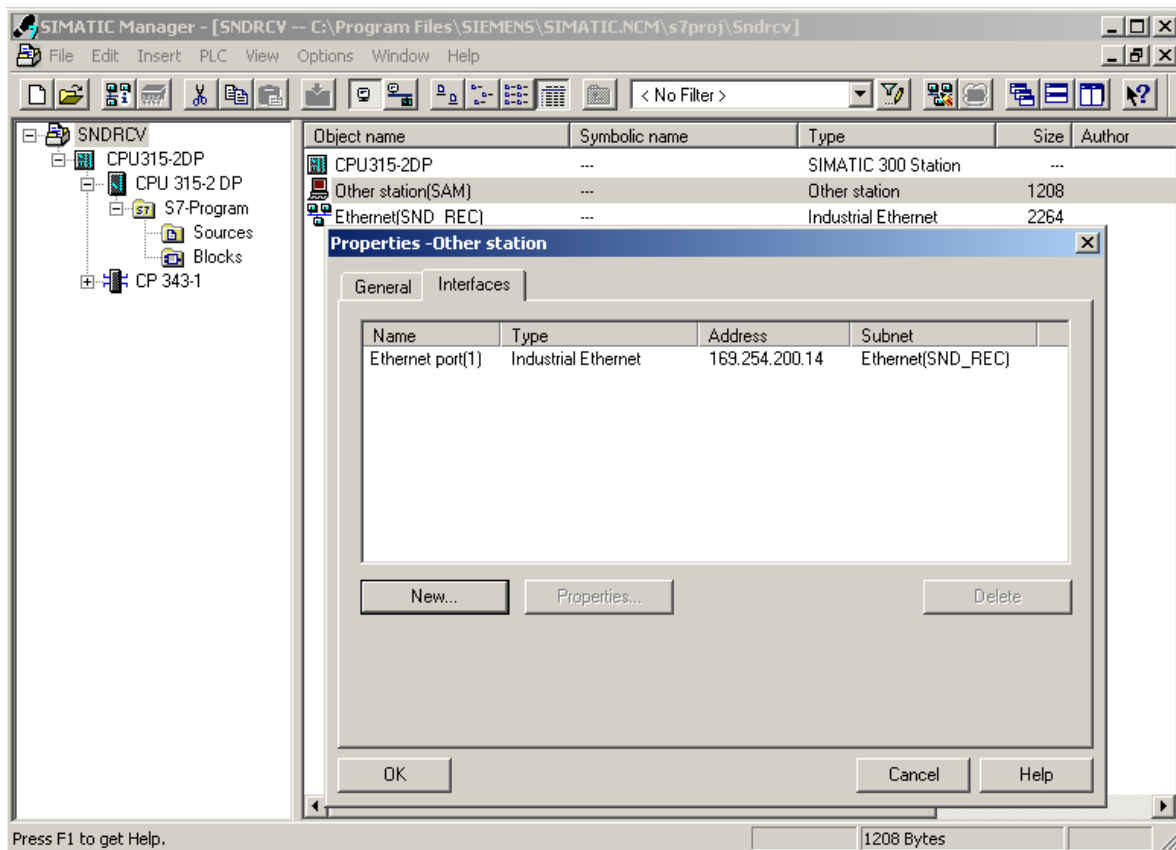


Image 14: Configuring the communication peer's station

#### 4.3.2.4  Setting up the active communications link

The CP343-1 or CP443-1 communications module is the connecting link providing communication between the S7-CPU and the industrial Ethernet-linked SIGMA AIR MANAGER. This means that for the connection of the relevant interface to the communications peer, suitable connection settings corresponding to the **unchangeable** connection settings in SIGMA AIR MANAGER have to be entered in the S7 station.

In the STEP7 project the CPU is selected in the opened S7 300- or S7 400 station for this purpose and the program **NetPro** is started by double clicking **Connections**. A block diagram of all stations and all planned sub-systems is displayed.
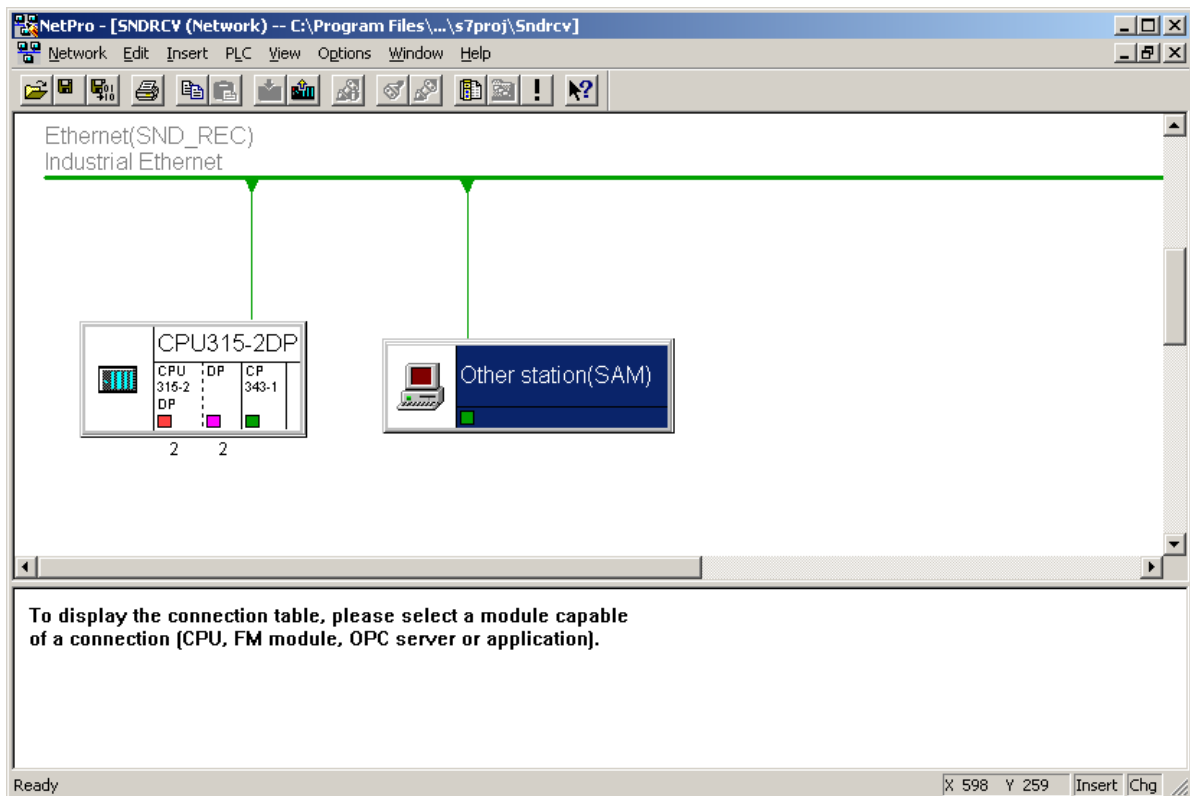


Image 15: Diagram of all networked communication peers

Mark the S7-CPU – a list of connections appears in the lower window – and by selecting **Edit**➢ **New connection,** the **Insert New Connection** dialog is accessed.

The communications peer to be selected is "**Other station** with connection type **ISO-on-TCP link**. Confirm these settings with **OK**.
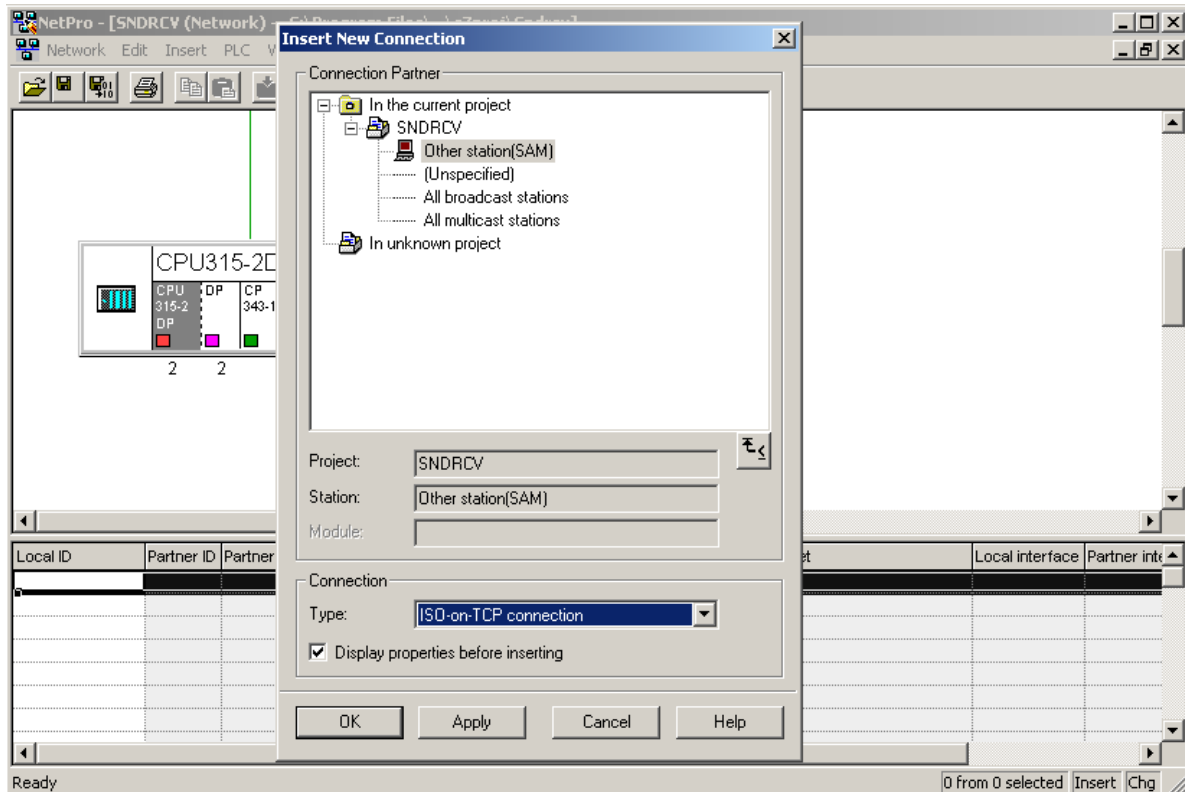


Image 16: Creating a new communication connection

The dialog for the ISO-on-TCP connection properties now opens. The new connection can be renamed, for example, **SNDRCV↔SAM**, in the **General** tab. The connection must be created **actively** via the S7 station.

The local ID, e.g. **0001_A050** can be selected in the drop-down menu. The left part **0001** of the ID number represents the connection job number that is needed later for the entering the *ID* formal parameter when the S7 system function module in STEP7 source code is called up.
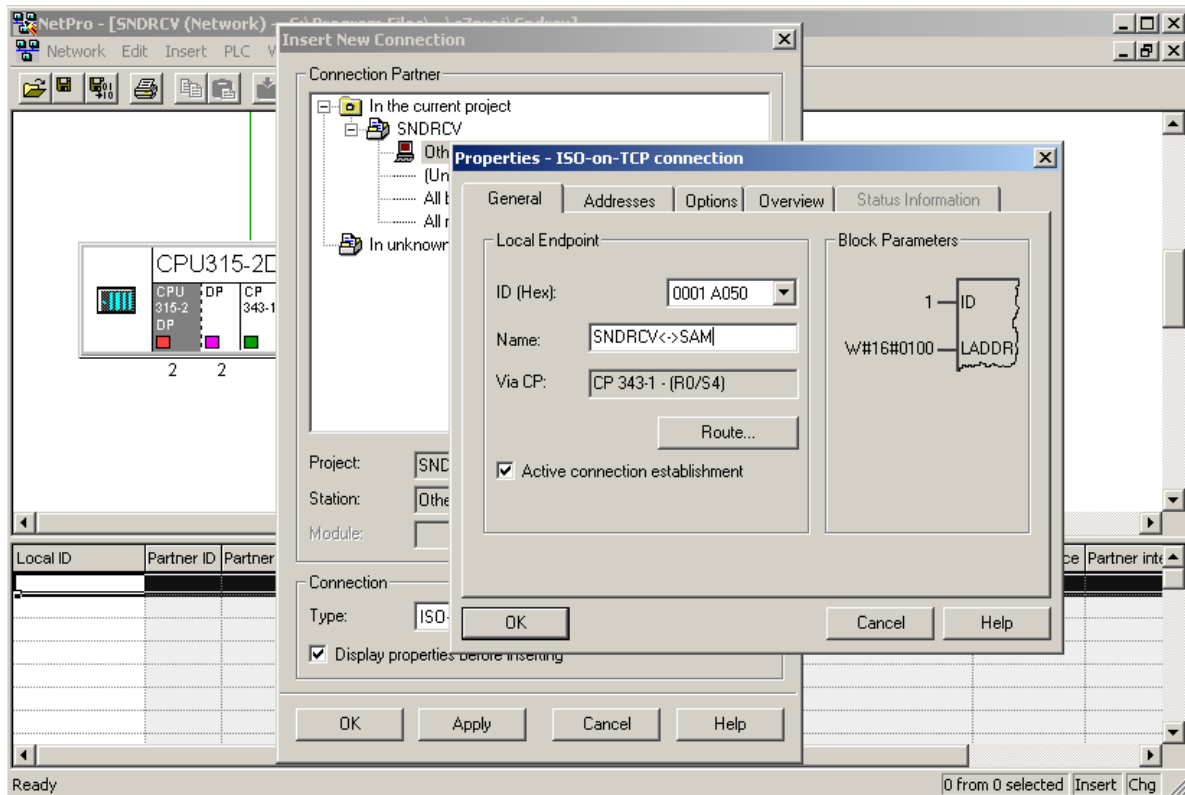


Image 17: General settings for the new communications connection

Afterwards, the details for setting up the connection are determined. The SIGMA AIR MANAGER IP address can be seen in the **Partner** column in the gray field in the **Addresses** tab. The ID **S-KAESER** is to be entered as the **TSAP (ASC)** (Transport Service Access Point) in both the local S7 station and in the partner (SIGMA AIR MANAGER) fields. The entries for **TSAP (hex)** and **TSAP length** are added automatically.
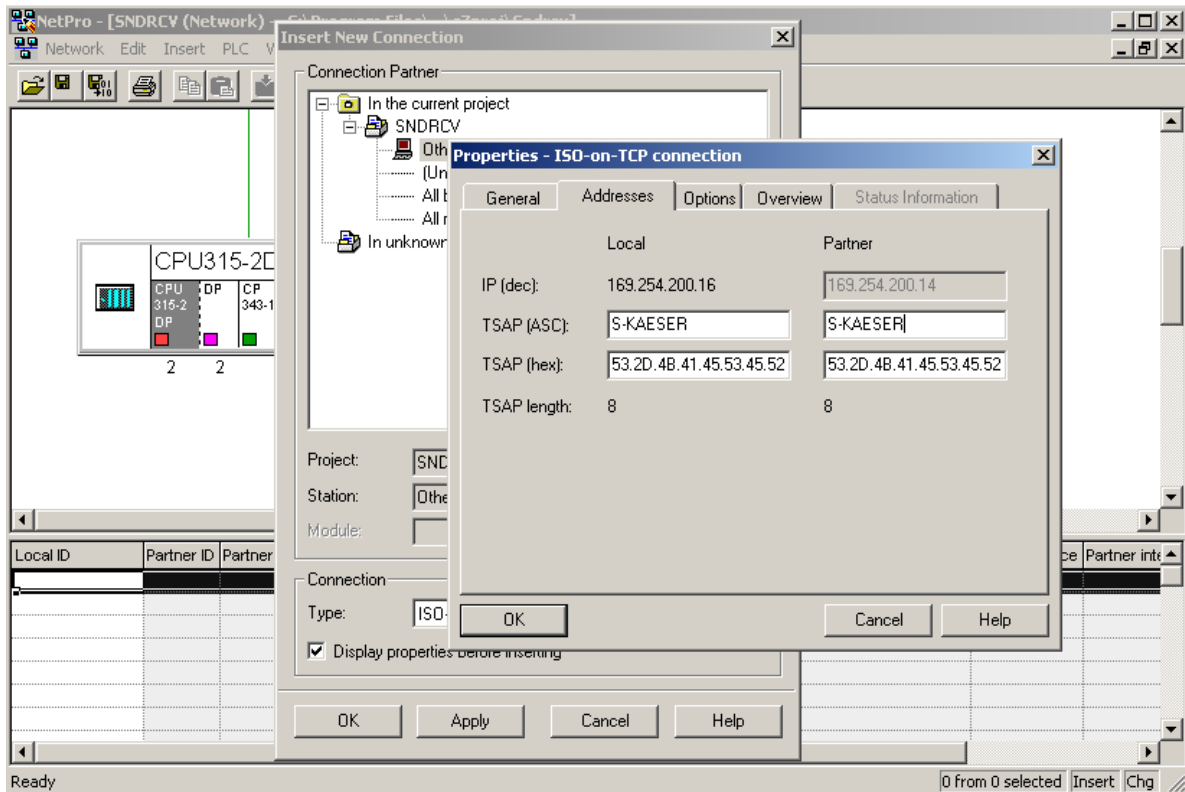


Image 18: TSAP configuration for the new communication connection

As only *SEND/RECEIVE* is possible as the operating mode of a TCP connection with read/write access on S7 stations, the mode **Send/Recv** is shown in the gray field in the **Options** tab.
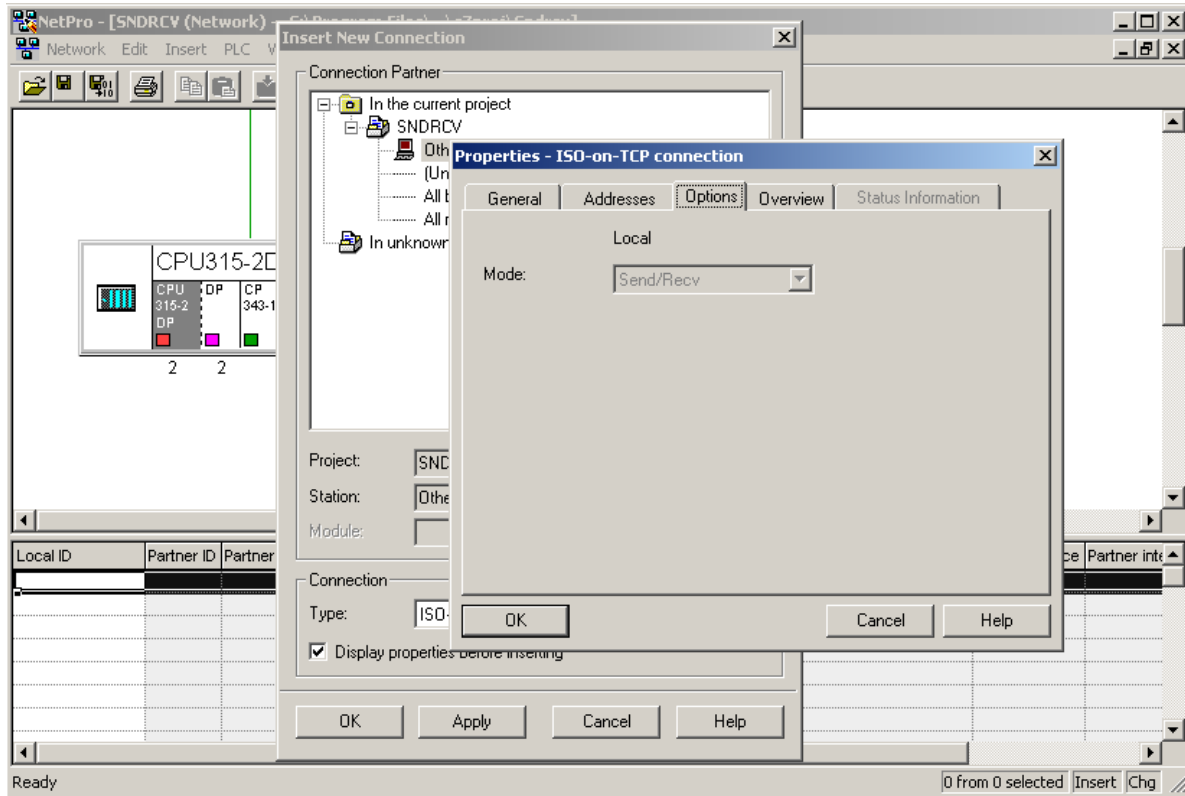


Image 19: SEND/RECEIVE specification for the new communication connection

The SEND/RECEIVE link is now parameterized sufficiently. All the settings are confirmed with OK. They appear in the lower window (list of connections).
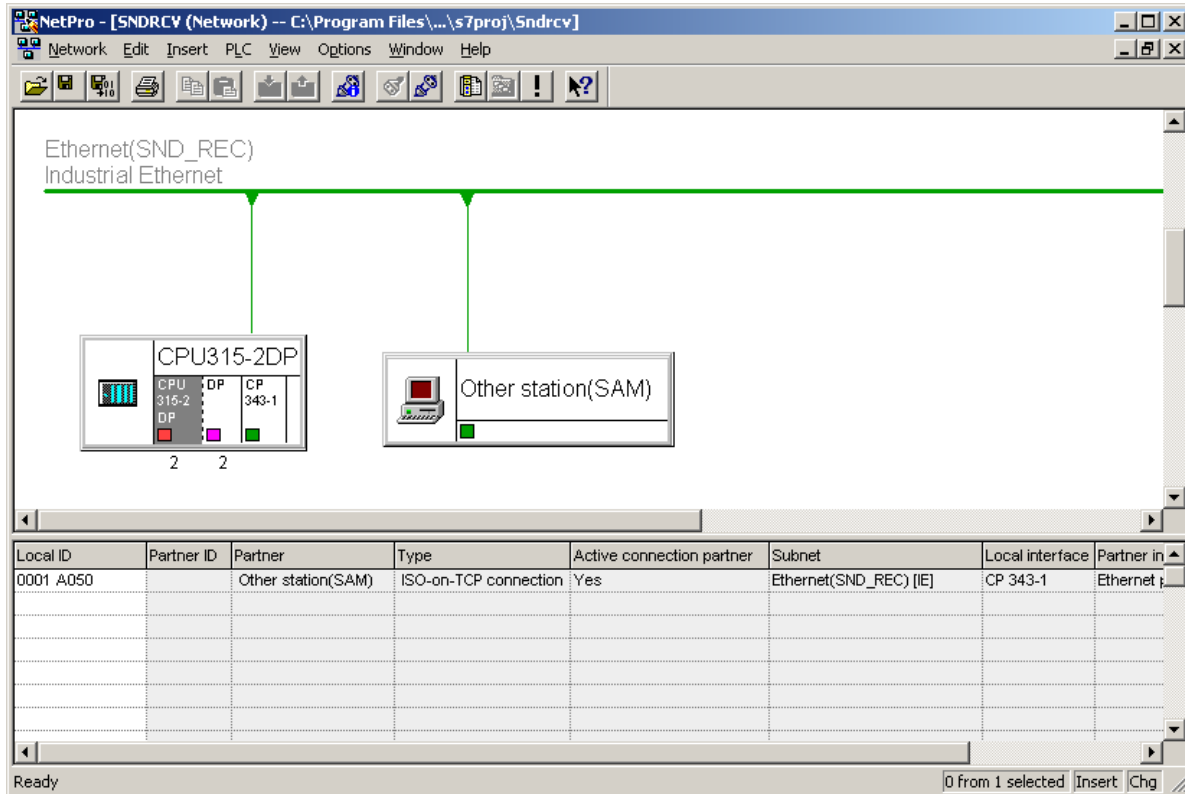


Image 20: Diagram of all communication peers and connections

The complete configuration now has to be saved with **Network ➢ Save and Compile**. In the next dialog select **Compile and check everything**. Now start the process by clicking on **OK**. Subsequently, mark the S7-CPU in the block diagram again and transfer the complete network configuration into the CPU3xx or CPU 4xx target module group via the MPI connection with **PLC ➢ Download to current project ➢ Selected Stations**.

When the translation process is successfully ended, close the **NetPro** program with **Network ➢ Exit**. The basic menu for SIMATIC Manager appears.

### 4.3.2.5 Embedding the function modules in the STEP7 user program

Next, the communication modules **FC5** (AG_SEND) and **FC6** (AG_RECV) have to be embedded in the STEP7 program. The FC5 and FC6 functions are found in the **SIMATIC_NET_CP** library that is delivered together with the **NCM S7 for Industrial Ethernet** software packet. Select **File ➢ Open** to open the **Open project** dialog. Here, the **SIMATIC_NET_CP** library is located in the **Libraries** tab. Mark this library and select with **OK**.
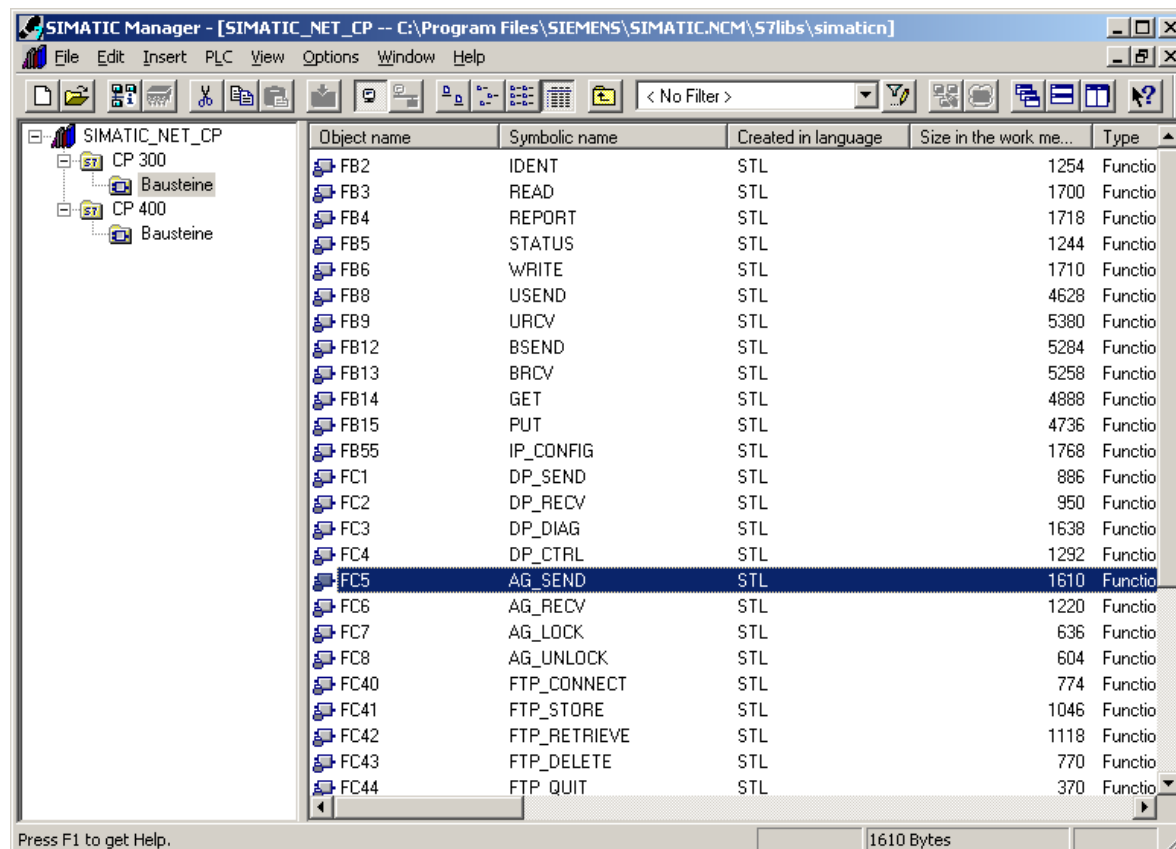


Image 21: List of S7 library modules

**SIMATIC 300-Station:** Double click on the **CP300** folder and then on **Blocks** to open the complete library with all the communications modules. Now select the **FC5** (AG_SEND) and **FC6** (AG_RECV) functions and copy them onto the notepad with **Edit ➢ Copy**. After changing to your own STEP7 project with **Window ➢ "Own project"** and double clicking on **S7-program** (below the open S7-CPU of own S7 station) and then on **Blocks**, insert the **FC5** and **FC6** functions into the project with **Edit ➢ Paste**.

**SIMATIC 400-Station:** Double click on the **CP400** folder and then on **Blocks** to open the complete library with all the communications modules. Now select the **FC5** (AG_SEND) and **FC6** (AG_RECV) functions and copy them onto the notepad with **Edit ➢ Copy**. After changing to your own STEP7 project with **Window ➢ "Own project"** and double clicking on **S7-program** (below the open S7-CPU of own S7 station) and then on **Blocks**, insert the **FC5** and **FC6** functions into the project with **Edit ➢ Paste**.
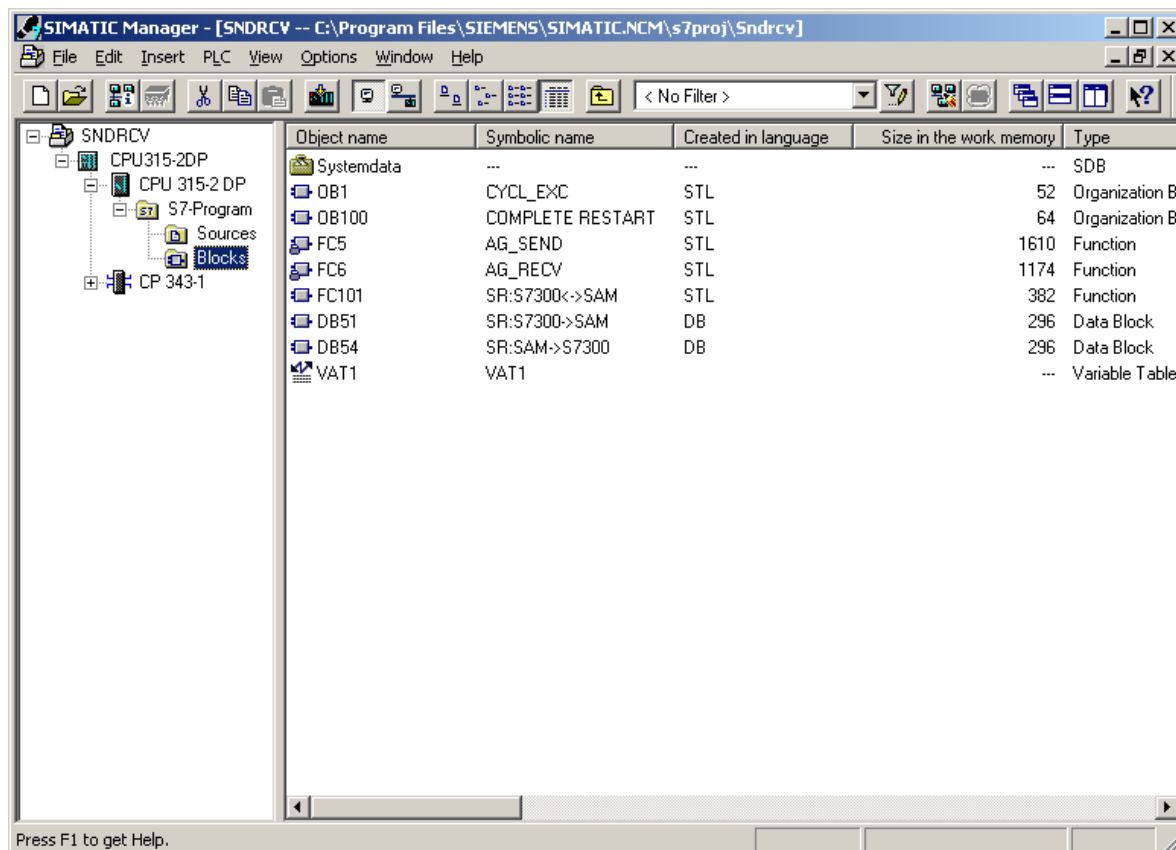
Image 22: List of S7 library modules

The purpose of the formal parameters that have to be provided with current parameters when the **FC5** and **FC6** communication modules are called up is to be found in the STEP7 reference manual. Alternatively, the online help supplied with STEP7 can be used. After marking the FC5 or FC 6 function call up line in source code, press **F1** to display the related topic in online help.
The example STEP7 program in **Annex C** can be used as a template in the same way.
All objects and system data in the target module CPU3xx or CPU4xx are to be transferred with selection of **PLC ➢ Download** and **"Blocks"**.

# 5 Data Interface

## 5.1 Monitoring the Connection

The SEND/RECEIVE connection between SIGMA AIR MANAGER and the communications peer can be monitored in remote control mode. This is made possible by a **handshake** program that is stored in SIGMA AIR MANAGER and activated and used by the communications peer. The parameters for the handshake program are the activating bit, the handshake bit and the handshake period that can all be preset via SEND/RECEIVE. The related S5 addresses (data block, data word) are quoted in the table of all available remote control functions (see 5.2). The handshake period is 10 to 9990 seconds.

The handshake bit has to be cyclically set by the communications peer when the handshake is activated (activating bit). SIGMA AIR MANAGER resets the handshake bit and starts the handshake period at the same time. The handshake bit has to be recognized as set again before the period expires. Otherwise SIGMA AIR MANAGER reports a SEND/RECEIVE fault, closes down all SEND/RECEIVE functions and reverts back to LOCAL settings (SAM-internal settings).

## 5.2 Remote Control Functions

The requirements for the remote control of SIGMA AIR MANAGER functions are that SEND/RECEIVE is activated and that remote control mode is activated.

**SEND/RECEIVE control functions have a higher priority than corresponding operating functions on SIGMA AIR MANAGER or input functions (control of digital inputs).**

If, for example, the command **System ON** is sent, then the air system cannot be switched off, neither by pressing the **OFF** key on SIGMA AIR MANAGER nor via the activated **Remote OFF/ON** input function. A corresponding instruction is displayed on SIGMA AIR MANAGER every time the operator presses the deactivated ON/OFF key. To switch off the air system on SIGMA AIR MANAGER, end remote control mode by pressing the **REMOTE** key and then the **OFF** key.

As well as the handshake function for link monitoring (see 5.1), the following remote control functions are available:

- **Compressed air system OFF/ON function**
- **Toggling between "Manual" and "Automatic" modes**
- **Clock mode ON/OFF**
- **Reset of current alarms, service and warning messages**
- **Selection and deselection of online compressors (each compressor)**
- **Selection and deselection of standby compressors (each group)**
- **Preset of pressure setpoint (in millibar)**
- **Preset of group sequence**

**SEND/RECEIVE control data that have been released are in S5 data block DB34 (see Technical Description *7_7725_PAGEN_*[*]). Other data may not be written. If other data are written, the internal data structure will be damaged and function errors or failure of SIGMA AIR MANAGER will occur. KAESER will not accept any liability or provide any warranty for such damage.**

There is an **activating bit** for every SEND/RECEIVE function and one or more **control bits** or **control values**. The activating bit activates the respective function and deactivates the respective local setting (SAM-internal setting). The control bits or control values control the function itself.

## 5.3   Process Data

It is a requirement for readout of SIGMA AIR MANAGER's process data that SEND/RECEIVE is activated by selecting the *FETCH/WRITE* or *SEND/RECEIVE* specification. Remote control mode does not have to be activated in order to read out data. activated

The process data are continually actualized (PLC–cyclic) and contain all the relevant operational states and process values of SIGMA AIR MANAGER and the online compressors. The following process data are available:

- **Process image of SIGMA AIR MANAGER in the S5 data block 31 (see Technical Description *7_7725_PAGEN_*)**

- **Process images of the online SIGMA CONTROL compressors in the S5 data blocks DB64 to DB79 (see Technical Description *7_7000_PASNDRCV_*)**

# Annex A: Example STEP5 SEND/RECEIVE Program

The example STEP5 program shown below is a non-binding example for an S5 station. It contains the most important modules needed to set up an S5-combatible communications connection with the *SEND/RECEIVE* specification between an S5 station and SIGMA AIR MANAGER.

In the example, the content of the S5 station's DB107 S5 data block (DW 0…15) is read and then written into SIGMA AIR MANAGER's DB34 data block. In return, the content of SIGMA AIR MANAGER's data block DB31 (DW 0… 119) is copied into the S5 station's S5 data block DB106.

```
OB21&OB22:
----------

    :SPA FB249//Call FB249 (S5-Standardfunction: SYNCHRON)
Name:SYNCHRON
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
BLGR:   KY 0,6//"0"=not used, "6"=Blocksize 512Byte
PAFE:   MB 40//
    :***
//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

OB1:
----

    :SPA FB100//Call FB100 (Communication SAM<->S5)
Name:SNDRCV
    :***


//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

FB100:
------

//SEND-ALL
    :SPA FB244//Call FB244 (S5-Standardfunction: SEND)
Name:SEND
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,0//"0"=not used, "0"=ALL-function
ANZW:   MW 100//Statusword (MW100)
QTYP:   KC NN//"NN"=Parameter set on CP1430TCP
DBNR:   KY 0,0//irrelevant
QANF:   KF +0//irrelevant
QLAE:   KF +0//irrelevant
PAFE:   MB 42//

//RECEIVE-ALL
    :SPA FB245//Call FB245 (S5-Standardfunction: RECEIVE)
Name:RECEIVE
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,0//"0"=not used, "0"=ALL-function
ANZW:   MW 110//Statusword (MW110)
QTYP:   KC NN//"NN"=Parameter set on CP1430TCP
DBNR:   KY 0,0//irrelevant
QANF:   KF +0//irrelevant
QLAE:   KF +0//irrelevant
PAFE:   MB 44//
```

```
//SEND-DIRECT (SEND)
    :UN  M141.1//SEND-Job active?  Y:Refresh "ANZW"   N:Toggle RECEIVE-Job
    :SPA FB244//Call FB244 (S5-Standardfunction: SEND)
Name:SEND
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,1//Job-Number (see CP1430TCP configuration)
ANZW:   MW 140//Statuswords (MW140, MW142)
QTYP:   KC DB//Source-DB for Data to send
DBNR:   KY 0,107//Send Data of DB107
QANF:   KF +0//Start sending Data at DW0 (DB107)
QLAE:   KF +16//16DW's to send (DB107 DW0 to DW15)
PAFE:   MB 50//

//RECEIVE-DIRECT (RECEIVE)
    :UN  M151.1//RECEIVE-Job active? Y:Refresh "ANZW"   N:Toggle RECEIVE-Job
    :SPA FB245//Call FB245 (S5-Standardfunction: RECEIVE)
Name:SEND
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,2//Job-Number (see CP1430TCP configuration)
ANZW:   MW 150//Statuswords (MW150, MW152)
QTYP:   KC DB//Target-DB for Data to receive
DBNR:   KY 0,106//Store received Data in DB106
QANF:   KF +0//Start to store Data at DW0 (DB106)
QLAE:   KF +120//120DW's to store (DB106 DW0 to DW119)
PAFE:   MB 52//
    :***

//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DB106 (Target-DB in S5 for received SAM-Data):
----------------------------------------------
  0:   KH = 0000;//SIGMA AIR MANAGER: Processdataword n
  1:   KH = 0000;//SIGMA AIR MANAGER: Processdataword n+1
  ..
118:   KH = 0000;//SIGMA AIR MANAGER: Processdataword n+118
119:   KH = 0000;//SIGMA AIR MANAGER: Processdataword n+119

//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DB107 (Source-DB in S5 for SAM RemoteControl-Data):
---------------------------------------------------
  0:   KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n
  1:   KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n+1
  ..
 14:   KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n+14
 15:   KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n+15

//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

# Annex B: Example STEP5 FETCH/WRITE Program

The example STEP5 program shown below is a non-binding example for an S5 station. It contains the most important modules needed to set up an S5-combatible communications connection with the *FETCH/WRITE* specification between an S5 station and SIGMA AIR MANAGER.

In the example, the content of the S5 station's DB105 S5 data block (DW 0…15) is written into SIGMA AIR MANAGER's DB34 data block. In return, the content of SIGMA AIR MANAGER's data block DB31 (DW 0… 119) is read and then written into the S5 station's S5 data block DB104. In addition, the process data of compressors 1-16 that is stored in SIGMA AIR MANAGER's data blocks DB64…79 is copied into the S5 station's S5 data blocks DB120…135.

```
//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
OB21&OB22:
----------

    :SPA FB249//Call FB249 (S5-Standardfunction: SYNCHRON)
Name:SYNCHRON
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
BLGR:   KY 0,6//"0"=not used, "6"=Blocksize 512Byte
PAFE:   MB 40//
    :***


//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

OB1:
----
    :SPA FB101//Call FB101 (Fetch SAM-Data&Data of COMP1..16)
Name:FETCHZ
    :
    :SPA FB100//Call FB100 (Communication SAM<->S5)
Name:SNDRCV
    :***


//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

FB101:
------

    :U   M  131.2//FETCH-Job done (Statusword FETCH)
    :UN  M  100.0
    :=   M  100.1//Impulse FETCH-Job done
    :U   M  131.2
    :=   M  100.0
    :
    :UN  M  100.1//Impulse FETCH-Job done? Y:Handle flags
    :SPB =LA1                         N:Jump over
    :L   MW 106
    :SLW     1//Shift Flag-Group for reading data of Compr.1 to 16 in each cycle
    :T   MW 106
    :L   KF +0
    :><F//Flag-Group zero? Y:Handle flag for reading SAM-Data
    :SPB =LA1          N:Read data of next Compressor
    :UN  M  102.0
    :=   M  102.0//Set flag for reading SAM-Data when it wasn't set before
    :SPB =LA1
    :L   KM 00000000 00000001
    :T   MW 106//Init word for start reading again at Compr.1
LA1 :NOP 0
    :***
```

```
    :A   DB 112//Open DB for indirect FETCH addressing
    :L   KF +31//SIGMA AIR MANAGER: Source-DB for SAM-Data to send is DB31
    :T   DW  11
    :L   KF +0//SIGMA AIR MANAGER: Start to send SAM-Data at DW0 (DB31)
    :T   DW  12
    :L   KF +120//SIGMA AIR MANAGER: 120DW's to send
    :T   DW  13
    :L   KF +104//S5: Target-DB for SAM-Data to receive is DB104
    :T   DW  15
    :L   KF +0//S5: Start to store SAM-Data at DW0 (DB104)
    :T   DW  16
    :L   KF +120;//S5: 120DW's to store (DB104 DW0 to DW119)
    :T   DW  17
    :L   MW 106//Flag-Group zero? Y:Jump over all
    :L   KH 0000                  N:Init Fetch addressing for reading Compr.-Data
    :!=F
    :SPB =END
    :L   KF +0
    :T   MB 240//Init Loop-Index
    :L   KF +64//Startaddress of Compressor-DB (SAM-DB64..79)
    :T   MW 236//Pointer for Compressor-DB
    :L   KF +120//Startaddress of Target-DB in S5 for Compr.-Data (S5-DB120..135)
    :T   MW 238//Pointer for Target-DB in S5
LOOP:L   MB 240//BEGIN:Handle Flag in Flag-Group for FETCH-Data of Compressor1..16
    :L   KF +8
    :<F
    :SPB =M107
    :-F
    :SLW     8
    :L   KY 0,106
    :SPA =M106
M107:L   KY 0,107
    :L   MB 240
    :SLW     8
M106:OW
    :T   MW 242//END:Handle Flag in Flag-Group for FETCH-Data of Compressor1..16
    :B   MW 242
    :UN  -M  0.0//Flag for FETCH-Data of Compressor n TRUE? Y:FETCH addressing
    :SPB =KOMP                                         N:Jump over addressing
    :L   MW 236//SIGMA AIR MANAGER: Source-DB for Compr.-Data to send is DB64..79
    :T   DW  11
    :L   KF +0//SIGMA AIR MANAGER: Start to send Data at DW0 (DB64..79)
    :T   DW  12
    :L   KF +64//SIGMA AIR MANAGER: 64DW's to send
    :T   DW  13
    :L   MW 238//S5: Target-DB for Compr.-Data to receive is DB120..135
    :T   DW  15
    :L   KF +0//S5: Start to store Data at DW0 (DB120..135)
    :T   DW  16
    :L   KF +64;//S5: 64DW's to store (DB120..135)
    :T   DW  17
KOMP:L   MW 236
    :I       1//Increment Pointer for Compressor-DB (SAM-DB64..79)
    :T   MW 236
    :L   MW 238
    :I       1//Increment Pointer for Target-DB in S5 (S5-DB120..135)
    :T   MW 238
    :L   MB 240
    :I       1//Increment Loop-Index
    :T   MB 240
    :L   KF +15//Loop-Index <=15? Y:Break
    :<=F                        N:Handle next Compressor
    :SPB =LOOP
END :
    :***
```

```
//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

FB100:
------

//SEND-ALL
    :SPA FB244//Call FB244 (S5-Standardfunction: SEND)
Name:SEND
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,0//"0"=not used, "0"=ALL-function
ANZW:   MW 100//Statusword (MW100)
QTYP:   KC NN//"NN"=Parameter set on CP1430TCP
DBNR:   KY 0,0//irrelevant
QANF:   KF +0//irrelevant
QLAE:   KF +0//irrelevant
PAFE:   MB 42//

//RECEIVE-ALL
    :SPA FB245//Call FB245 (S5-Standardfunction: RECEIVE)
Name:RECEIVE
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,0//"0"=not used, "0"=ALL-function
ANZW:   MW 110//Statusword (MW110)
QTYP:   KC NN//"NN"=Parameter set on CP1430TCP
DBNR:   KY 0,0//irrelevant
QANF:   KF +0//irrelevant
QLAE:   KF +0//irrelevant
PAFE:   MB 44//

//SEND-DIRECT (WRITE)
    :UN  M121.1//WRITE-Job active?  Y:Refresh "ANZW"   N:Toggle RECEIVE-Job
    :SPA FB244//Call FB244 (S5-Standardfunction: SEND)
Name:SEND
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,1//"0"=not used, "1"=Job-Number (see CP1430TCP configuration)
ANZW:   MW 120//Statuswords (MW120, MW122)
QTYP:   KC RW//"RW"=Indirect addressing for configuration ->DB112
DBNR:   KY 0,112////"0"=not used, Config-DB for this WRITE-Job is DB112
QANF:   KF +0//Configuration begins at DB112DW0
QLAE:   KF +0//irrelevant
PAFE:   MB 46//

//RECEIVE-DIRECT (FETCH)
    :UN  M131.1//FETCH-Job active?  Y:Refresh "ANZW"   N:Toggle RECEIVE-Job
    :SPA FB246//Call FB246 (S5-Standardfunction: FETCH)
Name:FETCH
SSNR:   KY 0,8//"0"=Direct parameterising, "8"=Base interface number of CP1430TCP
A-NR:   KY 0,2////"0"=not used, "2"=Job-Number (see CP1430TCP configuration)
ANZW:   MW 130//Statuswords (MW130, MW132)
QTYP:   KC RW//"RW"=Indirect addressing for configuration ->DB112
DBNR:   KY 0,112//"0"=not used, Config-DB for this FETCH-Job ist DB112
QANF:   KF +10//Configuration begins at DB112DW10
QLAE:   KF +0//irrelevant
PAFE:   MB 48//
    :***


//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DB112:
------
 0:    KC = 'DB';//SIGMA AIR MANAGER: Source of Datablock to receive is a DB
 1:    KY = 000,034;//SIGMA AIR MANAGER: Target-DB for Datablock to receive is DB34
 2:    KF = +00000;//SIGMA AIR MANAGER: Start to store Data at DW0 (DB34)
 3:    KF = +00016;//SIGMA AIR MANAGER: 16DW's to store (DB34 DW0 to DW15)
 4:    KC = 'DB';//S5: Source of Datablock to send is a DB
 5:    KC = 000,105//S5: Source-DB for Datablock to send is DB105
 6:    KF = +00000;//S5: Start to send Data at DW0 (DB105)
 7:    KF = +00016;//S5: 16DW's to send (DB105 DW0 to DW15)
 8:    KF = +00000;//not used
 9:    KF = +00000;//not used
10:    KC = 'DB';//S5: Source of Datablock to receive is a DB
11:    KY = 000,104;//S5: Target-DB for Datablock to receive is DB104
12:    KF = +00000;//S5: Start to store Data at DW0 (DB104)
13:    KF = +00120;//S5: 120DW's to store (DB104 DW0 to DW119)
14:    KC = 'DB';//SIGMA AIR MANAGER: Source of Datablock to send is a DB
15:    KC = 000,31;//SIGMA AIR MANAGER: Source-DB for Datablock to send is DB31
16:    KF = +00000;//SIGMA AIR MANAGER: Start to send Data at DW0 (DB31)
17:    KF = +00120;//SIGMA AIR MANAGER: 120DW's to send

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DB104 (Target-DB in S5 for received SAM-Data):
----------------------------------------------
  0:    KH = 0000;//SIGMA AIR MANAGER: Processdataword n
  1:    KH = 0000;//SIGMA AIR MANAGER: Processdataword n+1
   ..
118:    KH = 0000;//SIGMA AIR MANAGER: Processdataword n+118
119:    KH = 0000;//SIGMA AIR MANAGER: Processdataword n+119

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DB105 (Source-DB in S5 for SAM RemoteControl-Data):
---------------------------------------------------
  0:    KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n
  1:    KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n+1
   ..
 14:    KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n+14
 15:    KH = 0000;//SIGMA AIR MANAGER: RemoteControl-Dataword n+15

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DB120..DB135 (Target-DB in S5 for received Compressor-Data
                 ;Compressor1=DB120..Compressor16=DB135):
---------------------------------------------------------
  0:    KH = 0000;//SIGMA CONTROL: Processdataword n
  1:    KH = 0000;//SIGMA CONTROL: Processdataword n+1
   ..
 62:    KH = 0000;//SIGMA CONTROL: Processdataword n+62
 63:    KH = 0000;//SIGMA CONTROL: Processdataword n+63

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

# Annex C: Example STEP7 Program

The example STEP7 program shown below is a non-binding example for an S7 300 station. It contains the most important modules needed to set up a communications connection between an S7 station and SIGMA AIR MANAGER. The example is an S7 source file. The code can be inserted into a source file created with the help of the text editor. The corresponding STEP7 data blocks can then be generated by translation from this source file.

In the example, the content of the S7 station's DB51 S7 data block (DW 0…15) is copied into SIGMA AIR MANAGER's DB34 S5 data block. In return, the content of SIGMA AIR MANAGER's S5 data block DB31 (DW 0…119; SIGMA AIR MANAGER process image) is copied into the S7 station's S7 data block DB54 if in SIGMA AIR MANAGER's menu option under

*[main menu]* ➢ *[settings]* ➢ *[system]* ➢ *[communication]* ➢ *[SEND/RECEIVE]*

*SAM* is entered. If *SAM+SC* is entered in this menu option, the process images of the SIGMA CONTROL compressors linked to SIGMA AIR MANAGER are copied sequentially into the S7 station's S7 data blocks DB120-DB135 (compr.1 – compr.16).

```
//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DATA_BLOCK DB51 //"SR:S7300->SAM"
TITLE =
VERSION : 0.1


  STRUCT
   DB_VAR : INT ;  //DW0
   DB_VAR1 : INT ;  //DW1
    .
   //DB_VAR2 to DB_VAR13 have to be inserted before compiling!!!
    .
   DB_VAR14 : INT ; //DW14
   DB_VAR15 : INT ; //DW15
  END_STRUCT ;
BEGIN
   DB_VAR := 0;
   DB_VAR1 := 1;
    .
   //DB_VAR2 to DB_VAR13 have to be inserted before compiling!!!
    .
   DB_VAR14 := 14;
   DB_VAR15 := 15;
END_DATA_BLOCK

//+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DATA_BLOCK DB55 //"SR:SAM->S7300_Buffer"
TITLE =
VERSION : 0.1


  STRUCT
   DB_VAR : INT ;   //DW0
   DB_VAR1 : INT ;   //DW1
   .
   //DB_VAR2 to DB_VAR117 have to be inserted before compiling!!!
   .
   DB_VAR118 : INT ; //DW118
   DB_VAR119 : INT ; //DW119
  END_STRUCT ;
BEGIN
   DB_VAR := 0;
   DB_VAR1 := 1;
   .
   //DB_VAR2 to DB_VAR117 have to be inserted before compiling!!!
   .
   DB_VAR118 := 118;
   DB_VAR119 := 119;
END_DATA_BLOCK

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DATA_BLOCK DB54 //"SR:SAM->S7300_ProcessdataSigmaAirManager"
TITLE =
VERSION : 0.1


  STRUCT
   DB_VAR : INT ;   //DW0
   DB_VAR1 : INT ;   //DW1
   .
   //DB_VAR2 to DB_VAR117 have to be inserted before compiling!!!
   .
   DB_VAR118 : INT ; //DW118
   DB_VAR119 : INT ; //DW119
  END_STRUCT ;
BEGIN
   DB_VAR := 0;
   DB_VAR1 := 1;
   .
   //DB_VAR2 to DB_VAR117 have to be inserted before compiling!!!
   .
   DB_VAR118 := 118;
   DB_VAR119 := 119;
END_DATA_BLOCK

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

DATA_BLOCK DB120 //"SR:SAM->S7300_ProcessdataSigmaControlCompressor_1"
TITLE =
VERSION : 0.1


  STRUCT
   DB_VAR : INT ; //DW0
   DB_VAR1 : INT ; //DW1
   .
   //DB_VAR2 to DB_VAR117 have to be inserted before compiling!!!
   .
   DB_VAR118 : INT ; //DW118
   DB_VAR119 : INT ; //DW119
  END_STRUCT ;
BEGIN
   DB_VAR := 0;
   DB_VAR1 := 1;
   .
   //DB_VAR2 to DB_VAR117 have to be inserted before compiling!!!
   .
   DB_VAR118 := 118;
   DB_VAR119 := 119;
END_DATA_BLOCK

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


Analog for SigmaControlCompressor_2 to SigmaControlCompressor_16 !!!

DATA_BLOCK DB121 //"SR:SAM->S7300_ProcessdataSigmaControlCompressor_2"

to

DATA_BLOCK DB135 //"SR:SAM->S7300_ProcessdataSigmaControlCompressor_16"
```

```
//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

FUNCTION FC101 : VOID // "SR:S7300<->SAM"
TITLE =Communication S7300 <-> SAM
// --------------------------------------------------------------------------
AUTHOR : Tester
FAMILY : S7300
NAME : FCSNDRCV
VERSION : 1.0


VAR_TEMP
  Target : ANY ;
END_VAR
BEGIN
NETWORK
TITLE =Send Data of DB51, DW0 to DW15 into SAM-Receive-Buffer
//
      NOP   0; //CALL "AG_SEND"

      CALL FC     5 (
           ACT                      := M    100.0,
           ID                       := 1,//Local-ID (see NetPro!)
           LADDR                    := W#16#100,//Hex 100 <-> Dec 256
           SEND                     := P#DB51.DBX 0.0 WORD 16,//DB51, DW0 to DW15
           LEN                      := 32,//16DW's = 32Bytes
           DONE                     := M    100.1,
           ERROR                    := M    100.2,
           STATUS                   := MW   102);

// --------------------------------------------------------------------------

      R    M    100.0; //Reset ACTIV-Bit
      SET  ;
      A    M    100.1; //When DONE-Bit is set -> Jump to Label "done"
      JC   done;
      SET  ;
      A    M    100.2; //When ERROR-Bit is set -> Jump to Label "err"
      JC   err; //
      JU   recv; //When ERROR-Bit & DONE-Bit isn't set->continue receiving Data

done: S    M    100.0; //Set ACTIV-Bit for next Send-Command and
      JU   recv; // Jump to Label "recv"
err:  NOP  1;
      S    M    100.0; //Set ACTIV-Bit for next try


NETWORK
TITLE =Receive SAM-Data and store it in Buffer-DB55, DW0 to DW119

recv: NOP   0; //      CALL "AG_RECV"

      CALL FC     6 (
           ID                       := 1,//Local-ID (see NetPro!)
           LADDR                    := W#16#100,/Hex 100 <-> Dec 256
           RECV                     := P#DB55.DBX 0.0 WORD 120,//DB55, DW0 to DW119
           NDR                      := M    110.5,
           ERROR                    := M    110.1,
           STATUS                   := MW   104,
           LEN                      := MW   106);
```

```
NETWORK
TITLE = Route the received data from Buffer_DB

      L    DB55.DBB  183; //Corresponding Slavenumber in DR91 of Buffer-DB55
      JL   SPLE;
      JU   SPLE; //Slavenumber=0
      JU   SPLE; //Slavenumber=1
      JU   SAM; //Slavenumber=2 -> SigmaAirManager
      JU   CO1; //Slavenumber=3 -> SigmaControl Compressor 1
      JU   CO2; //Slavenumber=4 -> SigmaControl Compressor 2
      JU   CO3; //Slavenumber=5 -> SigmaControl Compressor 3
      JU   CO4; //Slavenumber=6 -> SigmaControl Compressor 4
      JU   CO5; //Slavenumber=7 -> SigmaControl Compressor 5
      JU   CO6; //Slavenumber=8 -> SigmaControl Compressor 6
      JU   CO7; //Slavenumber=9 -> SigmaControl Compressor 7
      JU   CO8; //Slavenumber=10-> SigmaControl Compressor 8
      JU   CO9; //Slavenumber=11-> SigmaControl Compressor 9
      JU   C10; //Slavenumber=12-> SigmaControl Compressor 10
      JU   C11; //Slavenumber=13-> SigmaControl Compressor 11
      JU   C12; //Slavenumber=14-> SigmaControl Compressor 12
      JU   C13; //Slavenumber=15-> SigmaControl Compressor 13
      JU   C14; //Slavenumber=16-> SigmaControl Compressor 14
      JU   C15; //Slavenumber=17-> SigmaControl Compressor 15
      JU   C16; //Slavenumber=18-> SigmaControl Compressor 16
SPLE: JU   COMM; // Jump to Label "COMM"
SAM:  NOP  0;
      L    54; //Set DataBlock DB53 for ProcessdataSigmaAirManager
      JU   COMM; // Jump to Label "COMM"
CO1:  NOP  0;
      L    120; //Set DataBlock DB120 for ProcessdataSigmaControlCompressor_1"
      JU   COMM; // Jump to Label "COMM"
CO2:  NOP  0;
      L    121; //Set DataBlock DB121 for ProcessdataSigmaControlCompressor_2"
      JU   COMM; // Jump to Label "COMM"
CO3:  NOP  0;
      L    122; //Set DataBlock DB122 for ProcessdataSigmaControlCompressor_3"
      JU   COMM; // Jump to Label "COMM"
CO4:  NOP  0;
      L    123; //Set DataBlock DB123 for ProcessdataSigmaControlCompressor_4"
      JU   COMM; // Jump to Label "COMM"
CO5:  NOP  0;
      L    124; //Set DataBlock DB124 for ProcessdataSigmaControlCompressor_5"
      JU   COMM; // Jump to Label "COMM"
CO6:  NOP  0;
      L    125; //Set DataBlock DB125 for ProcessdataSigmaControlCompressor_6"
      JU   COMM; // Jump to Label "COMM"
CO7:  NOP  0;
      L    126; //Set DataBlock DB126 for ProcessdataSigmaControlCompressor_7"
      JU   COMM; // Jump to Label "COMM"
CO8:  NOP  0;
      L    127; //Set DataBlock DB127 for ProcessdataSigmaControlCompressor_8"
      JU   COMM; // Jump to Label "COMM"
CO9:  NOP  0;
      L    128; //Set DataBlock DB128 for ProcessdataSigmaControlCompressor_9"
      JU   COMM; // Jump to Label "COMM"
C10:  NOP  0;
      L    129; //Set DataBlock DB129 for ProcessdataSigmaControlCompressor_10"
      JU   COMM; // Jump to Label "COMM"
C11:  NOP  0;
      L    130; //Set DataBlock DB130 for ProcessdataSigmaControlCompressor_11"
      JU   COMM; // Jump to Label "COMM"
C12:  NOP  0;
      L    131; //Set DataBlock DB131 for ProcessdataSigmaControlCompressor_12"
```

```
      JU    COMM; // Jump to Label "COMM"
C13:  NOP   0;
      L     132; //Set DataBlock DB132 for ProcessdataSigmaControlCompressor_13"
      JU    COMM; // Jump to Label "COMM"
C14:  NOP   0;
      L     133; //Set DataBlock DB133 for ProcessdataSigmaControlCompressor_14"
      JU    COMM; // Jump to Label "COMM"
C15:  NOP   0;
      L     134; //Set DataBlock DB134 for ProcessdataSigmaControlCompressor_15"
      JU    COMM; // Jump to Label "COMM"
C16:  NOP   0;
      L     135; //Set DataBlock DB135 for ProcessdataSigmaControlCompressor_16"
      JU    COMM; // Jump to Label "COMM"
COMM: NOP   0;


//Setting values of the local ANY-pointer "Target"


      T     LW    4; //DB-Number
      L     B#16#2; //Datatype Byte
      T     LB    1;
      L     B#16#10; //System-ID for S7, always 10Hex.
      T     LB    0;
      L     240; //repetition factor
      T     LW    2;
      L     B#16#84; //memory is a DataBlock
      T     LB    6;
      L     0;
      T     LB    7;
      T     LW    8; //ByteAdress=0, BitAdress=0


          //CALL "BLKMOV"
      CALL SFC   20 (
          SRCBLK                   := P#DB55.DBX0.0 BYTE 240, //Source=Buffer_DB55
          RET_VAL                  := MW   108,
          DSTBLK                   := #Target);//Destination=DB54 or DB120/121...

END_FUNCTION

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

ORGANIZATION_BLOCK OB 1 //"CYCL_EXC"
TITLE = "Main Program Sweep (Cycle)"
VERSION : 0.1


VAR_TEMP
  OB1_EV_CLASS : BYTE ; //Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
  OB1_SCAN_1 : BYTE ;    //1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
  OB1_PRIORITY : BYTE ; //Priority of OB Execution
  OB1_OB_NUMBR : BYTE ; //1 (Organization block 1, OB1)
  OB1_RESERVED_1 : BYTE ;   //Reserved for system
  OB1_RESERVED_2 : BYTE ;   //Reserved for system
  OB1_PREV_CYCLE : INT ;    //Cycle time of previous OB1 scan (milliseconds)
  OB1_MIN_CYCLE : INT ; //Minimum cycle time of OB1 (milliseconds)
  OB1_MAX_CYCLE : INT ; //Maximum cycle time of OB1 (milliseconds)
  OB1_DATE_TIME : DATE_AND_TIME ;   //Date and time OB1 started
END_VAR
BEGIN
NETWORK
TITLE =

      CALL FC   101 ; //CALL "SR:S7300<->SAM"

END_ORGANIZATION_BLOCK


//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

ORGANIZATION_BLOCK OB 100 //"COMPLETE RESTART"
TITLE = "Complete Restart"
VERSION : 0.1


VAR_TEMP
  OB100_EV_CLASS : BYTE ;   //16#13, Event class 1, Entering event state, Event
logged in diagnostic buffer
  OB100_STRTUP : BYTE ; //16#81/82/83/84 Method of startup
  OB100_PRIORITY : BYTE ;   //Priority of OB Execution
  OB100_OB_NUMBR : BYTE ;   //100 (Organization block 100, OB100)
  OB100_RESERVED_1 : BYTE ; //Reserved for system
  OB100_RESERVED_2 : BYTE ; //Reserved for system
  OB100_STOP : WORD ;   //Event that caused CPU to stop (16#4xxx)
  OB100_STRT_INFO : DWORD ; //Information on how system started
  OB100_DATE_TIME : DATE_AND_TIME ; //Date and time OB100 started
END_VAR
BEGIN
NETWORK
TITLE =

      SET   ;
      =     M   100.0; //Set ACTIV-Bit once for the first Send-Command

END_ORGANIZATION_BLOCK

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```