Manual

CP143 H1 / TCP/IP

for

H1 and TCP/IP

Order-No.: VIPA SSN-HB89E
Rev. 00/07

Subject to change to cater for technical progress.

The information contained in this manual is subject to change without notice. The contents may be changed without VIPA giving prior notice at any time. The software described in this manual is supplied on the basis of a general license. This license entitles you to make copies for use within your organization only.

We will claim compensation for damages.

Phone.: +49 (9132) 744-0
Fax.: +49 (9132) 744-144
EMail: info@vipa.de
http://www.vipa.de

Hotline: +49 (9132) 744-114

About this manual

This manual describes applications of the CP143plus and CP143 TCP/IP modules. These modules are plug-compatible with any PLC (AG-115U trough AG-188U) and do not require forced air ventilation. The standard handling modules (SEND, RECEIVE) that are supplied by Siemens may be used for TCP/IP task processing by the PLC. The description of the handler modules is contained in the respective manuals available from your CPU.

The application of the two modules is described separately in this manual. The different chapters0 describe the H1-elements first, followed by the TCP/IP-specific elements.

Due to technical progress the illustrations of modules contained in this manual may differ from the original.

Overview

Chapter 1: Introduction

This chapter contains general safety and handling instructions for electrostatically sensitive modules. It also contains an overview of the CP143-packages available from VIPA and specifications on the construction and the operation of the modules.

Chapter 2: Communication protocols

This chapter contains information on the H1 and the TCP/IP communication protocols. Here you may find details on the structure and the operation of these protocols. The chapter also refers to the ISO-OSI level model and depicts the relationship to TCP/IP.

Chapter 3: Planning a network

This chapter contains an overview of the different ethernet network cabling types, the respective hardware and guidelines, as well as an explanation of the various combinations. The chapter also contains an outline of the planning required for a network and how to prepare and install the networking hardware.

Chapter 4: Network module CP143 H1 / TCP/IP

This chapter describes the physical structure of the module. It shows the position and the purpose of the jumpers and the configuration of the different sockets and plugs. The chapter also describes the control elements and the displays.

Chapter 5: Software

This chapter comprises two parts. The first part contains a general description of the configuration software. The second part details the use of the configuration software for the different protocols.

Here you can find information on how to use the software for H1 or TCP/IP.

Chapter 6: Commissioning

This chapter contains information on the plug-in locations and the operating characteristics of a properly configured module. An example of a small communication system illustrates the principles.

Contents

1 Introduction

1 Introduction

1.1 Safty and handling precautions for the user

1.1.1 Handling electrostatically sensitive modules

The modules produced by VIPA carry MOS-VLSI components. These components are highly susceptible to high voltages that may be caused by electrostatic discharges.

These modules bear the following symbol to highlight the sensitive nature of the components:

This symbol is attached to the module, module carriers or on the packing material and indicates that the module is sensitive to electrostatic discharge.

Modules that are sensitive to electrostatic discharges may be destroyed by energies and voltages that lie far below the threshold of human perception. It is possible that damaging voltages may occur if a person who has not been discharged handles a module that is electrostatically sensitive. The resulting damages to the components will seriously impair the operation of the module or render it unusable. Modules that have been damaged in this way are rarely recognized immediately. In many cases the error becomes apparent only after the module has been operating for a while.

Components that have been damaged by an electrostatic discharge may exhibit temporary malfunctions when they are subjected to temperature variations, shock or changes in electrical load.

The only effective method for the prevention of damages to modules that are sensitive to electrostatic discharge is the use of protective gear and the strict adherence to the respective rules.

1.1.2 Shipping electrostatically sensitive modules

Modules to be shipped must be packed in their original container. You may also wrap additional conductive material around modules that are about to be shipped. Conductive material comes in the form of antistatic foils or metallized enclosures.

> ⚠ *Some modules may contain a battery. Please ensure that the conductive packaging does not come into contact with or short out the battery.*

1.1.3 Tests and modifications to electrostatic sensitive modules

Please not the following when testing modules that are sensitive to electrostatic discharge:

- Any floating instrumentation must be discharged for a short period.
- Any test instruments must be grounded.

Please use a grounded soldering iron when you implement modifications to modules that are sensitive to electrostatic discharge.

> ⚠ *It is important that the person as well as the test gear and tools are grounded properly when modules that are sensitive to electrostatic discharge must be manipulated*

.

1.2 General

You may use the VIPA CP143plus or the CP143 TCP/IP to implement Industrial Ethernet H1 or TCP/IP networks in your offices or factories.

The software used to configure the modules is compatible with MS-DOS®, WINDOWS® 3.11, WINDOWS® 95, WINDOWS® NT and OS/2®.

VIPA offers the following packages:

CP143plus
CP143-module
with H1-protocol
(Order-No.:VIPA SSN-BG81A)

CP143 TCP/IP
CP143-module
with TCP/IP-protocol
(Order-No.:VIPA SSN-BG89A)

Both are delivered with the same configuration software and the same manual.

The specific requirements are covered by a single hardware - platform which differs only in the contents of the firmware.

The following passages refer to the hardware as "CP143 H1 / TCP/IP"

Fig. 1-1: Construction of the CP143 H1 or TCP/IP

1.3 Operation

The CP143 H1 / TCP/IP communication module is installed directly into the central processor or into the respective expansion unit. The module contains four interfaces:

- Interface for the LAN cable via a transceiver (AUI, DIX)
- Interface for the LAN cable via thin ethernet cable
- Interface for the LAN cable via RJ45-socket (twisted pair)
- Interface for compatibility with the programmer or a DIN interface for configuration purposes

The CP143 H1 / TCP/IP communication processor conducts the transfer of data via the cellular network independently and relieves the CPU of this task. This operation covers the entire ISO-OSI level model.

The VIPA CP143 H1 / TCP/IP module is controlled by the standard handler modules of the installed PLC.

The following modules are required for interfacing with the VIPA CP143 H1 / TCP/IP:

SEND ALL, RECEIVE ALL, SYNCHRON, SEND DIREKT, RECEIVE DIREKT, FETCH.

These function modules can be used to transfer DB/DX-modules with a maximum length of 4096 DW. These function modules can be configured directly or indirectly.

Special features

- The CP143 H1 / TCP/IP communication processor is supplied complete with a 12V power supply for the transceiver.
- The application of these modules does not require any hardware or software manipulation.
- Comes complete with thin ethernet interface.
- Contains a LED status indicator to display the operating status of the module and the network.

1.4 Construction

- 80486 SLC 33 MHz CPU
- 1 MB main memory
- Siemens CP1430 compatible programmer interface
- EPROM socket for EPROMs
- 4 KB Dual-Port-RAM as CP interface
- DIN interface for the PLC
- AUI interface (IEEE 802.3) for the LAN cable via a transceiver, RJ45 signal may be connected to the AUI socket (Siemens compatible)
- RJ45 connector for ethernet
- Thin ethernet (BNC)

2 Communication protocols

2 Communication protocols

2.1 H1 protocol

The Industrial Ethernet H1 LAN provides the basis for the implementation of large, high-tech, decentralized automation systems. Industrial Ethernet H1 is the complete solution for the implementation of large, open networks between the different departments of a company.

Compatible devices include:

- PLCs 115, 135, 150, 155
- programmers PG 685, PG 730, PG 750, PG 770
- computers
- other PLCs

The Industrial Ethernet H1 LAN may be used in the following applications:

- central control and supervision of production lines
- monitoring and communicating the status of machines
- management information
- production statistics
- the transfer of large amounts of data
- remote programming of PLCs

2.1.1 Structure of Industrial Ethernet H1

The Industrial Ethernet H1 LAN consists of separate segments. A single segment can support up to 100 terminal devices (PLCs, computers) and its length may be up to 500 m. Individual segments may be linked by means of repeaters. A maximum of 2 repeaters may be placed between any two terminal devices.

Repeaters may also be used to construct so-called remote-repeater-configurations. This combination requires two repeaters and two additional Industrial Ethernet H1 transceivers.

The LAN cable may either be the triaxial cable (yellow cable) or the thin ethernet-cable (cheapernet).

Triaxial cable:
All terminal devices or repeaters must be connected to the transceivers by means of a drop-cable (plug-in cable 727-1) (max. length 50 m). The connection between the bus cable and the transceiver is provided by coaxial plugs. A terminator is required at both ends of the bus cable.

A transceiver on the triaxial cable may be connected to a multiport repeater that can support up to 8 terminal devices.

Thin-Ethernet:
Terminal devices are connected directly to the cable, that is transceivers and drop-cables are not required. The bus cable is connected to the module by means of BNC plugs. A terminator is required at both ends of the bus cable.

2.1.2 Operation

The Industrial Ethernet H1 LAN (in conformance with the Ethernet system standard) employs a random access technique: every terminal device accesses the bus when required. Access to the bus is co-ordinated by a CSMA/CD (Carrier Sense Multiple Access/Collision Detection) mechanism: every terminal device “listens” on the LAN and accepts transmissions that match its own address. Terminal devices can start a transmission only if the bus line is not engaged. Should two terminal devices initiate a transmission simultaneously, they detect this event and stop transmitting. After a random delay time they will restart the aborted transmission.

The programmers PG 685, PG 730, PG 750 and PG 770 may be connected directly to the Industrial Ethernet H1 by means of a transceiver. You may then remotely modify the program of any PLC that is connected to the LAN. The 923C co-ordinator or the programming multiplexer can also be used to program a number of modules via a single connection.

The user services of LEVEL 4/4a (handler modules or libraries for PC-modules) are also available.

2.1.3 Structure and operation of the PLC interface

A dual-port RAM connects the CP143 H1 / TCP/IP directly to the back plane bus-LAN. At the PLC the dual-port RAM is available as a standard-CP-interface with 4 tiles. Any H1 data traffic is conducted via standard handler modules (SEND and RECEIVE).

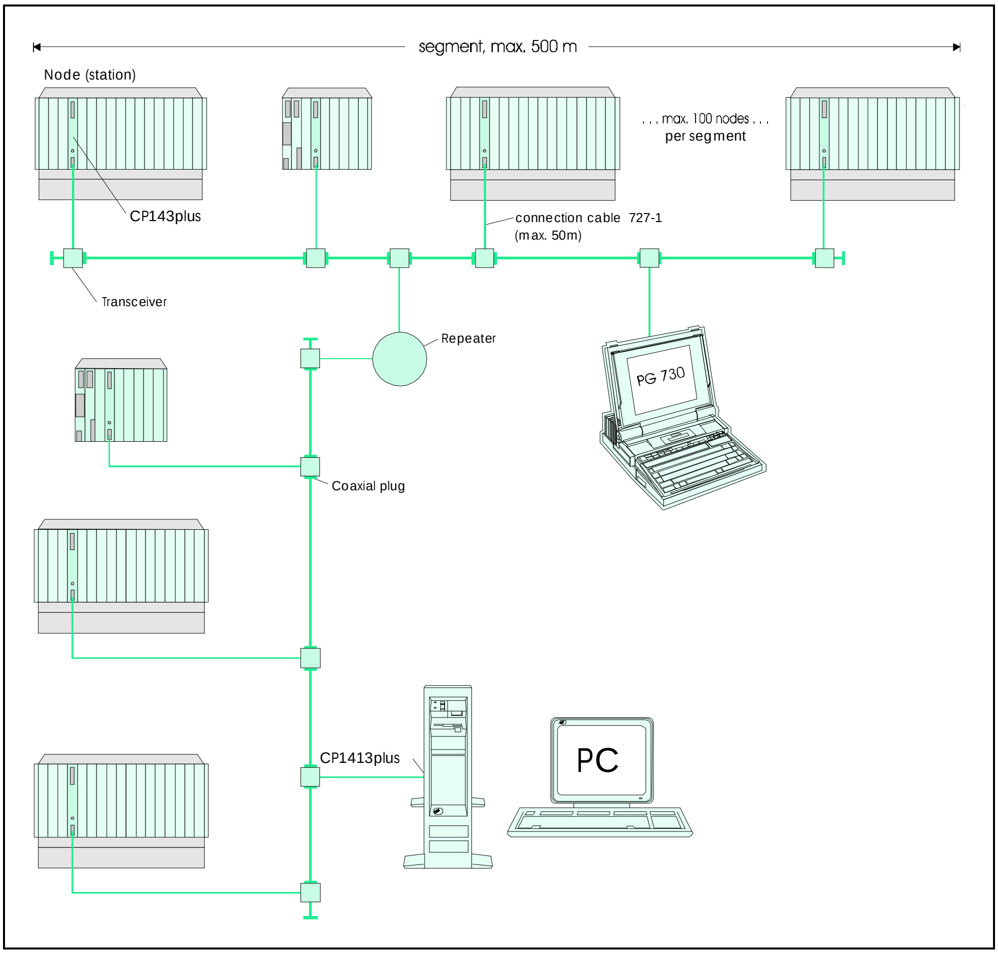


Fig. 2-1: Example for the cabling of the Industrial Ethernet H1 LAN

2.2 TCP/IP protocol

The TCP/IP protocol was an integral component of the Berkeley UNIX version 4.2 and as such it became a de-facto standard in local-area networks. Many manufacturers of workstations have since opted for the ethernet - TCP/IP -UNIX combination. All hardware systems that are currently of importance support versions of the TCP/IP protocol. This applies equally to the low-end PC, the typical mini-computer, right through to mainframe computers (the TCP/IP protocol has been implemented on IBM systems as well) and also to special purpose computers like vector processors and parallel processors. It is for this reason that TCP/IP is often used when heterogeneous systems must be linked.

Using TCP/IP it has become possible to implement comprehensive open networks linking different departments within an organization.

The following tasks may be solved by means of a TCP/IP bus system:

- central control and monitoring of manufacturing facilities
- transfer of the status of machines
- management information services
- production statistics
- the transfer of large amounts of data

2.2.1 ISO-OSI level model

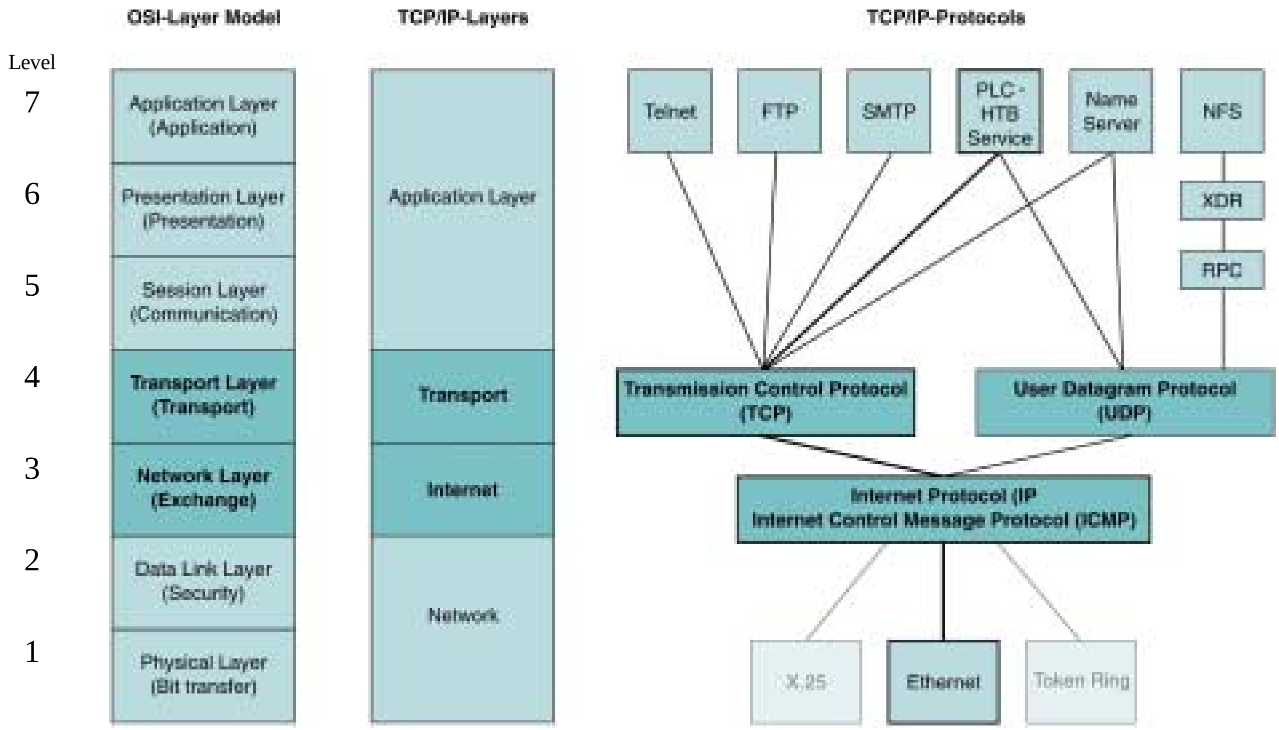

OSI-Layer Model
TCP/IP-Layers
TCP/IP-Protocols
Level
7
6
5
4
3
2
1
Application Layer (Application)
Presentation Layer (Presentation)
Session Layer (Communication)
Transport Layer (Transport)
Network Layer (Exchange)
Data Link Layer (Security)
Physical Layer (Bit transfer)
Application Layer
Transport
Internet
Network
Telnet
FTP
SMTP
PLC - HTB Service
Name Server
NFS
XDR
RPC
Transmission Control Protocol (TCP)
User Datagram Protocol (UDP)
Internet Protocol (IP
Internet Control Message Protocol (ICMP)
X.25
Ethernet
Token Ring

Fig. 2-1: ISO-OSI level model

2.2.1.1 General

The ISO-OSI level model is based upon a suggestion that was developed by the International Standards Organization (ISO). It represents the first step in the direction of an international standard for the different protocols currently in use. The model is referred to as ISO-OSI level model. OSI is the abbreviation for **O**pen **S**ystems **I**nterconnection, that is communication between open systems. The ISO-OSI level model is not a network architecture as the services and the protocols used on the different levels have not been defined. The model simply provides information about the task that each level must perform.

2.2.1.2 Description of the different levels

Level 1 Physical level (bit-transfer level)

The physical level transfers data bits via a communication channel. This level is generally responsible for the mechanical, electrical and procedural interfaces and for the physical communication medium that is used by the physical level:

- the voltages representing a logical 0 and a logical 1.
- the duration for which the voltage must be present to represent a valid bit.
- establishes a connection and terminates it when necessary.
- pin-allocation for the respective interface.

Level 2 Data link layer

The data link layer converts raw data into a data series ready for transfer. This is where the frame delimiters are inserted during transmission and where they are detected during reception. This is achieved by inserting a special bit-pattern at the start and at the end of a frame. In many instances a flow-control and error detection mechanism is integrated into the data link layer.

Level 3 Network layer

The network layer is concerned with the selection of the packet path between the source and the destination (other networks as well). Paths may be hard-wired in the network. They may also be dynamic, that is they may be determined on the fly for every packet in order that the load on the network optimized. The network layer must also ensure that different packets do not interfere with each other. This layer is responsible for administrative functions for the transferred data, that is counting of packets, characters or bits, etc.

The network layer must have the capacity to perform these tasks so that users of heterogeneous networks may communicate with each other.

Level 4 Transport layer

The purpose of the transport layer is to accept data from the session layer, then to divide it up into smaller portions, transfer this data to network layer and to ensure that every item arrives properly at the destination. The transport layer generates a unique network connection for every transport link required by the session layer. The transport layer determines the type of service that is provided to the session layer. It can also increase the throughput, establish multiple network connections and assign the data to these connections.

In addition to managing the use of multiple access channels by many message streams the transport layer must also administer connections that the network establishes and relinquishes.

Level 5 Session layer

The session layer provides the links that enable users of different systems to communicate. In addition to the standard transport services the session layer also provides high-level services that may be useful for certain applications. One of these services the control of interactive functions. The session layer administers the flow of data where data can only be transferred in one direction (similar to the token-management). A session can provide user-access to other systems or it can transfer a file from one system to another.

Another service of the session layer is that of synchronization. Certain reference points are superimposed on the data stream. If the network should crash it is only necessary to re-send the data from the most recent reference point.

Level 6 Presentation layer

The presentation layer executes those functions that are used very often and therefore justify a general procedure. This level is also responsible for the syntax and semantics of the transferred data.

One example of the services provided by the presentation layer is the co-ordination of data in a standard and pre-arranged manner. The presentation layer converts the different data structures and codes supplied by the various computers into a standard network code so that the users of these computers may communicate.

Level 7 Application layer

The application layer supports a large number of protocols.

This layer provides various file-transfer and data processing functions to the user. The application layer also adapts editor programs to the requirements of the respective terminal equipment.

Different file processing systems employ different conventions for file names and different methods to display lines of text, etc. You must remember these properties when you transfer files between different systems. This function as well as electronic mail, remote task entry and directory inquiries are also part of the services provided by the application layer.

2.2.1.3 TCP/IP in relation to the ISO-OSI level model

TCP/IP is older than the ISO-OSI level model. For this reason the different layers can not be applied directly to TCP/IP. The TCP/IP network layers correspond to level 1 and 2 of the ISO-OSI model. Levels 5 and 6 of the ISO-OSI model are already integrated into the protocols of the TCP/IP application layer.

Internet Protocol (IP)

The purpose of the internet protocol IP is to transport packets of data from a sender to a receiver via a number of interlinked networks. It may thus be compared with level 3 of the ISO-OSI level model. IP data transfer services are not protected and do not require a direct connection. The main IP services are: the transfer of datagrams via a multitude of interconnected networks, to address the respective partner correctly, to divide data packets according to the respective network conventions and to reassemble them, and to transfer information about the sequence of packets and their security characteristics.

Transmission Control Protocol (TCP)

The transmission control protocol TCP is intended to provide a reliable transport-end-to-end-connection between two different systems. For this reason TCP may be compared to level 4 of the ISO-OSI level model and it interfaces directly with the internet protocol IP.

TCP provides the following essential services:

- Orientation towards a connection
- Guaranteed sequences
- Loss detection
- Connection time-out detection
- Multiplexing
- Flow control
- Transparent data transfers
- The secure establishment and termination of connections

User Datagram Protocol (UDP)

This protocol can also be classed with level 4 of the ISO-OSI level model and interfaces directly with the internet protocol IP. This protocol does not make use of connections or error correction for network transfers. It also does not expect acknowledgements from the receiver of the data.

2.2.2 TCP/IP principle of operation

TCP and IP are only two of the many protocols required for the establishment of a complete network architecture. The application layer of the computer supplies programs like FTP and Telnet. The standard handler modules of the program define the application layer of the CP143 TCP/IP.

These application programs use the transport layer and the TCP or UDP protocols to transfer data, and these protocols communicate with the IP protocol of the internet layer.

The main task of the IP is to address data packets properly. This means that the IP functions like an envelope. The network determines the destination of the data packet by means of the address and uses this to route the packet accordingly. The protocol divides data into smaller units as different networks may require different length data frames. A number is assigned to every packet so that received packets may be acknowledged and to allow for reassembly of the original data. TCP as well as IP have their own envelope where they store the required numbers so that these sequence numbers can be transferred via the network. A packet of data is inserted into a TCP-envelope. This is then inserted into an IP envelope and transmitted via the network. On the network, TCP is responsible for the safe transport of data. TCP can recognize and correct transmission errors. These error detection and correction systems ensure that TCP connections are relatively safe. However, in this case UDP is much faster. It neither detects missing data packets nor does it check the sequence of the packets. UDP is an unprotected protocol.

2.2.3 TCP/IP services

OPEN / CONNECT

In an active situation this opens a virtual connection with a partner. In a passive situation this waits for a connection from a partner.

SEND

Transfer of a data buffer to TCP for transmission to a partner.

RECEIVE

Reception of data from a partner.

CLOSE

Termination of a connection.

3 Network planning

3 Network planning

3.1 Ethernet network-terminology

An ethernet network provides a communication link between different stations on the network. The stations on the network may consist of personal computers, industrial computers, automation equipment etc. Stations are connected to the network and located a certain minimum distance from each other. The stations on the network and the LAN cable form a network segment. (See Fig. 3-1).

The cable of the network segment usually consists of a number of network cables of varying length that are linked by means of connectors.

The length of the network segment and the number of stations that may be connected to it is limited. It is possible to extend a network segment by linking multiple segments via repeaters.

A repeater increases the physical topology of a network. The repeater requires two transceivers to link two different segments to each other. In this case the repeater is transparent to the network and combines the two segments into a single logical channel.

In this situation the repeaters task is to regenerate and to amplify the signals in both directions. It must also be able to recognize, process and to communicate inter-segment collisions. Repeaters do not possess a unique address and are therefore not accessible as they are invisible to the stations connected to the network.

The overall network cable consists of the sum of all segment cables.

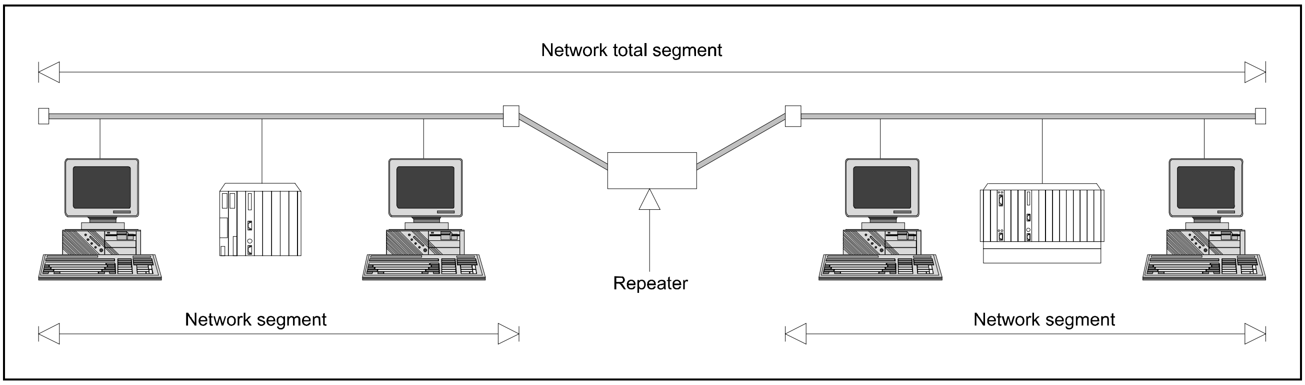


Fig. 3-1: Parts of an ethernet network

Three types of ethernet cable exist:

- Thin-ethernet-cable (also called thin ethernet cable or Cheapernet-cable).
- Thick-ethernet-cable (also called thick ethernet cable or standard ethernet-cable or yellow-cable).
- Twisted pair-cable (telephone cable)

Thin ethernet-cable is far more economical than thick ethernet-cable. This applies to the cost of the cable, the cost of installation and any additional hardware. It does, however, suffer from a higher level of susceptibility for EMC interference. Connections to the thin ethernet-cable are provided by BNC T-pieces.

Thick ethernet-cable consists of a thick yellow cable which is much like a pipe. The cable has markings at a distance of 2,5 m which indicate the positions where a connection to the cable may be attached.

These two types of network cable can be employed to produce in three different versions of ethernet network:

1. a network using only thin ethernet cable
2. a network consisting only of thick ethernet cable and
3. a combination of these two

Under certain conditions you may also use twisted telephone twin-cable instead of the coaxial cable (twisted pair).

Twisted pair cable consists of four insulated copper wires of 1 mm diameter that have been twisted together in pairs.

In contrast to the coaxial ethernet networks which provide a type of bus-topology, the twisted pair network is a point-to-point network. The resulting network has a star-topology. Every station is coupled individually with a central hub to form the ethernet network.

3.2 Thin-ethernet-cable networks

Thin-ethernet-cable network hardware

This chapter contains a description and an illustration (Fig. 3-2) of the hardware required for a thin-ethernet-network.

Network adapter

Network adapters are installed in every station on the network and linked by means of a network cable to provide the communication medium between these stations.

BNC-connectors

BNC-plugs and -sockets provide the electrical link between the hardware on the network. The network cable is connected to the BNC-socket located on the network adapter. BNC-plugs connected to both ends of the network cable link the different segments of the cable via a T-piece or a barrel connector. The T-piece is connected to the BNC-socket on the network adapter.

Thin-ethernet-cable

Thin-ethernet-cable consists of RG-58A/U 50 Ohm coaxial cable with a diameter of 0,2 inch (5,08 mm).

BNC-connectors

BNC-connectors are used to interconnect two thin-ethernet network cables. Where two portions of a thin-ethernet network must be linked these connectors have a distinct advantage over T-pieces. You may obtain these BNC-connectors from VIPA GmbH.

BNC-T-pieces

The two sockets located at opposite ends of the T-piece provide the connection to the BNC-connectors on the thin-ethernet cable. The third connector on the T-piece is connected to the BNC-connector on the network adapter and provides the physical link to the network. You may obtain BNC-T-pieces from VIPA.

BNC-terminator

Every network must end in a 50 Ω BNC terminator which acts as a matched load and which reduces electrical interference on the network. The terminator is connected to one of the two sockets of the T-piece when no other cable is connected. Some BNC terminators have a ground connection which may be used to ground the screen of the network cable. You may obtain BNC terminators with and without ground connection from VIPA GmbH.

⚠ *One side of a network segment (it is immaterial which side) must be fitted with a BNC-terminator which has a ground connection!*

The other side of the segment may not have a terminator!

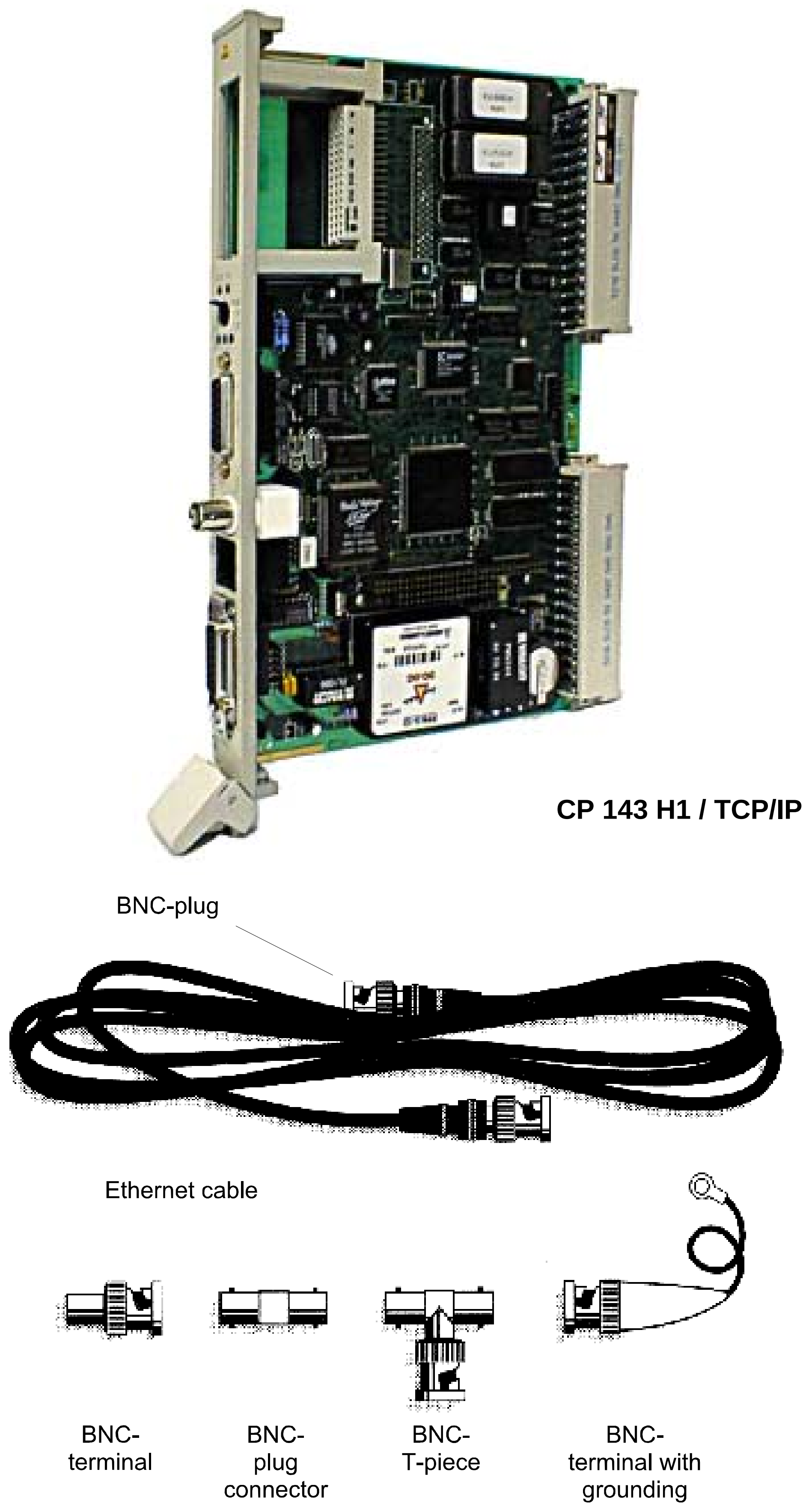


Fig. 3-2: Thin ethernet network hardware

3.2.1 Thin-ethernet-cable network-layout

Here follows a summary of the restrictions and the rules that apply to a thin-ethernet-cable network. Fig. 3-3 is an illustration of this summary.

Restrictions

- Maximum number of network segments: 5 (3 coaxial segments with network stations and 2 interconnecting segments without network stations)
- Maximum length of a network segment: 185 m
- Maximum total length of the network cable: 925 m
- Maximum number of network stations on a segment: 30 (every repeater is counted as a station)
- Minimum distance between two BNC-T-pieces: 0,5 m

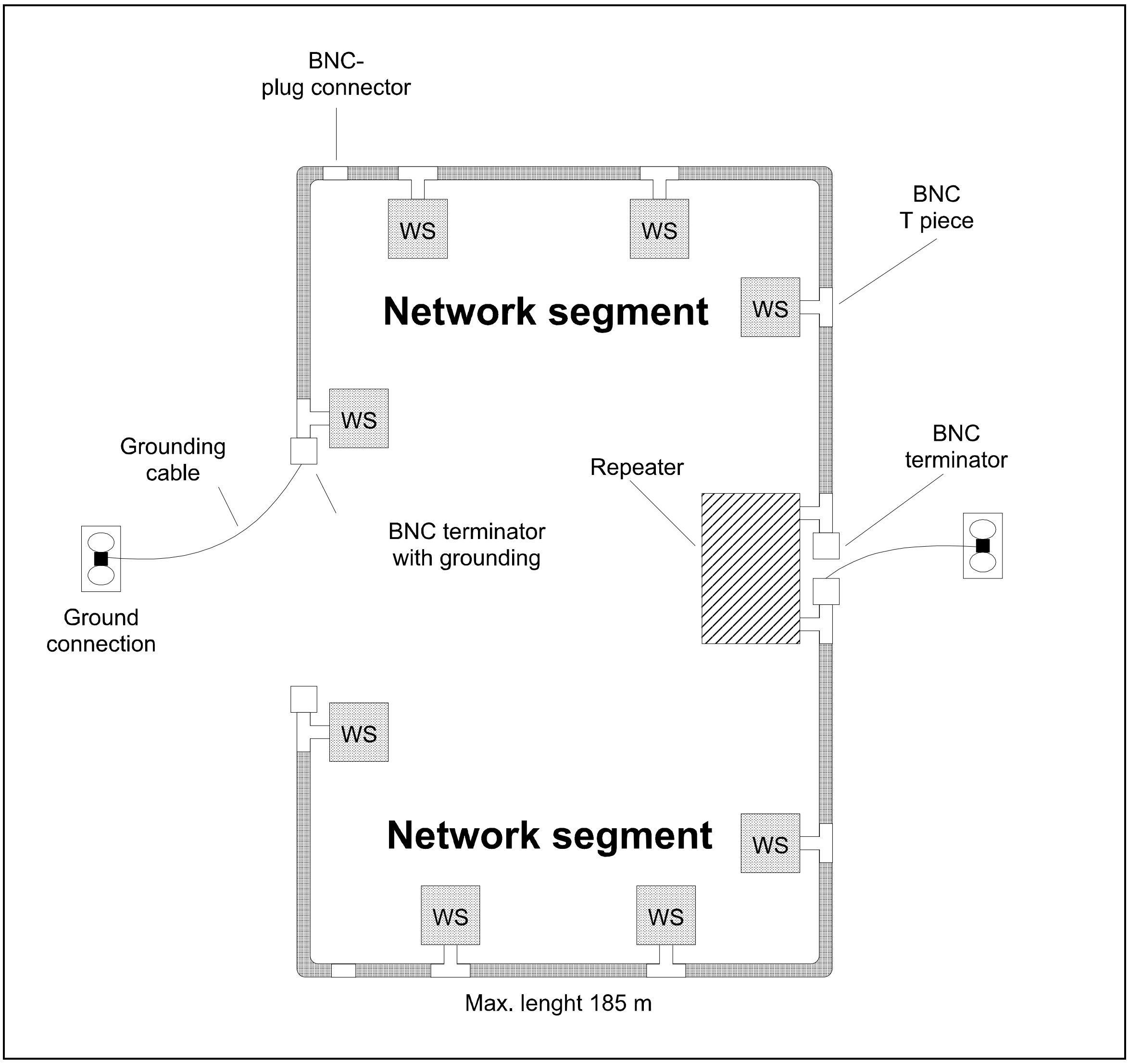


Fig. 3-3: Example for a thin-ethernet cable network

3.2.2 Regulations and specifications

Certain basic regulations must be observed when planning or installing a network:

- The cabling of a network consists of a number of segments. Each segment must be terminated with its characteristic impedance (terminator) of 50 Ω.

- According to the ISO 8802.3 specification the maximum length of an individual segment is 185 m.

- Wherever possible, the network should consist a homogeneous segments of coaxial cable as every connector introduces losses. When it becomes necessary to extend existing segments of coaxial cable you should always use preconfigured cables. Please note that the maximum length of a segment may not exceed 185 m.

- The screen of the coaxial cable may only be grounded at a single point on the entire cable. This point must posses a defined low impedance. All other connectors and terminators must be insulated from ground.

- Any two stations on the network may only have a single connection between them. Where multiple physical paths exist the resulting interference will be interpreted as a signal and cause collisions which may abort the data transfer.

- Single drop-cables are not permitted.

Guidelines for the routing and installation of network components

- As a rule, a minimum distance of 1m is prescribed with respect to electrical equipment, cabling and other components that could produce electromagnetic and electrostatic fields. This includes parallel runs of power or high-tension cables, circuit breakers etc.

- The ISO-specifications allow for a maximum field strength of 2 V/m at frequencies between 10 kHz and 30 MHz and 5 V/m at frequencies between 30 MHz and 1 GHz.

- Once a coaxial cable segment and all its components have been installed it is essential that the worst-case reflections on the cable are determined. These should never exceed 7% of the injected amplitude of the original signal. The reflection coefficient can be checked by means of a reflectometer.

- The overall loop resistance of a coaxial cable segment together with all its plugs and connectors may not exceed a maximum of 4Ω.

- The minimum radius of any bend in the cable may not be less than 5 cm. When multiple bending is anticipated the radius must not be less than 8 cm.

3.2.3 Technical data thin-ethernet

Coaxial cable

Impedance	50 Ω ± 2 Ω
Attenuation	≤ 18 dB/km for a 10 MHz sine wave
	≤ 12 dB/km for a 5 MHz sine wave
Propagation	≥ 0,77 c (velocity of light)
Minimum radius for bends, single bend	5 cm
Minimum radius for bends, multiple bends	8 cm
Diameter of center conductorsolid copper	0,89 mm ± 0,05 mm
Outside diameter for polyvinylchloride (PVC)	4,9 mm
Outside diameter for fluorpolymer	4,8 mm

3.3 Thick-ethernet-cable networks

Thick-ethernet-cable network hardware

This chapter contains a description and an illustration (Fig. 3-4) of the hardware required for a thick-ethernet-network.

Network adapter

Network adapters are installed in every station on the network and linked by means of a network cable to provide the communication medium between these stations.

Transceiver

The stations on a thick-ethernet cable communicate via an external transceiver which is attached to the network cable. External transceivers are required when a thick-ethernet cable is used as communication medium.

Transceiver-cable

The transceiver-cable (drop-cable) connects the station (network adapter) to the external transceiver.

AUI-connector

The end of a transceiver cable is terminated in an AUI (DIX) plug or an AUI-socket. The AUI-plug is connected to the network adapter located in a station. The AUI-socket is connected to the external transceiver.

Thick-ethernet-cable

The thick-ethernet-cable consists of a 50 Ohm coaxial cable with a diameter of 0,4 inch (10,16 mm). Different lengths of this cable are available complete with standard 0,4 inch N-type plugs at each end. The cable may be obtained from any reputable cabling supplier.

N-type plug

An N-type plug is attached to each end of the thick-ethernet cable.

N-type connector

N-type connectors are used to connect two thick-ethernet cables together.

N-type-terminator

An N-type 50 Ω termination resistance “terminates” the network and prevents electrical interference from the network. This is attached to the N-type socket of the transceiver that is not connected to another cable. Certain terminators are provided with a grounding strap to connect the screen of the network cable to ground.

⚠ *One side of a network segment (it is immaterial which side) must be fitted with a terminator which has a ground connection!*

The other side of the segment may not have a terminator!

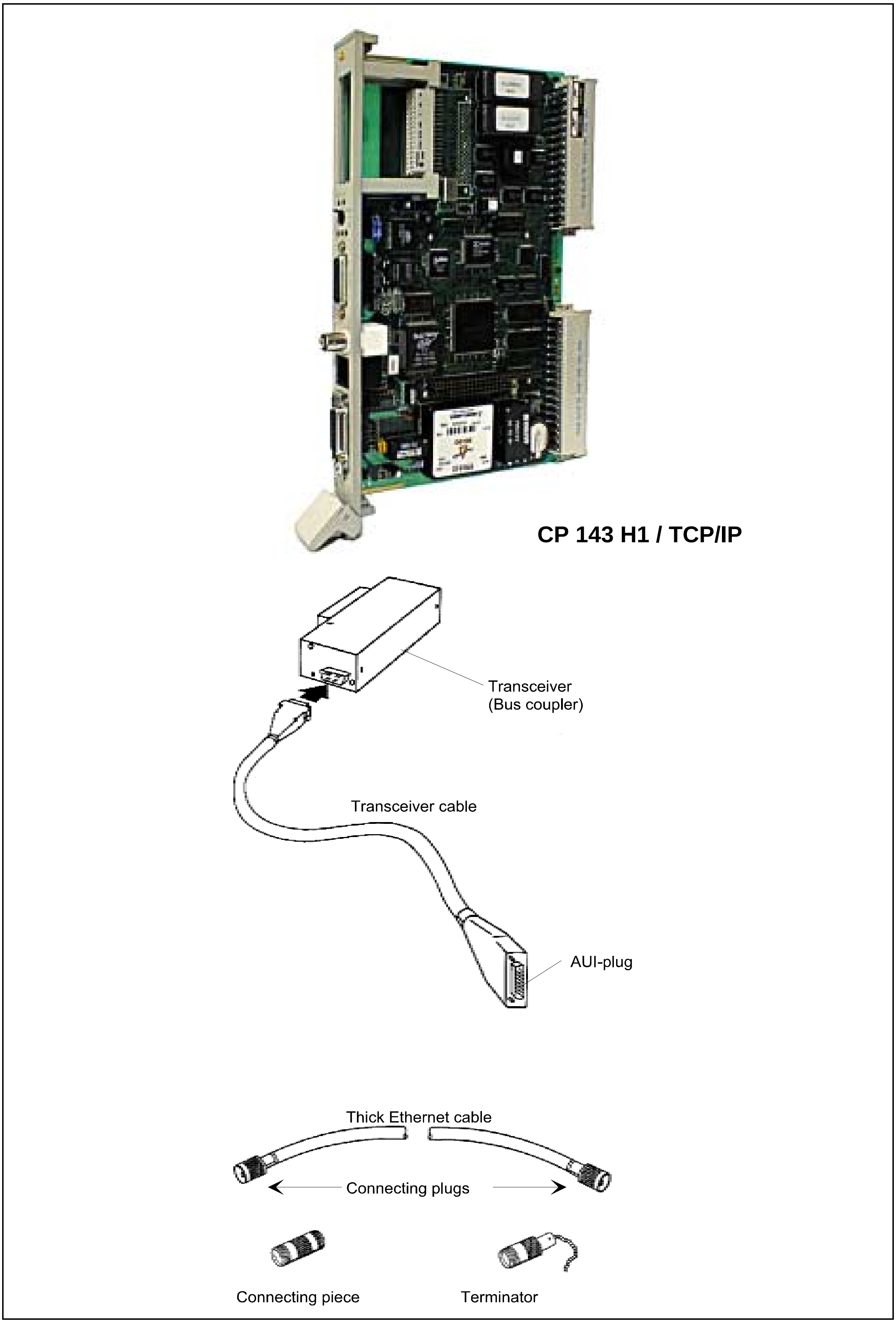


Fig. 3-4: thick-ethernet network hardware

3.3.1 Thick-ethernet-cable network-layout

Here follows a summary of the restrictions and the rules that apply to a thick-ethernet-cable network. Fig. 3-5 is an illustration of this summary.

Restrictions

• Maximum number of network segments:	5	(3 coaxial segments with network stations and 2 interconnecting segments without network stations)
• Maximum length of a network segment:	500 m	
• Maximum total length of the network cable:	2.500 m	
• Maximum number of network stations on a segment:	100	(every repeater is counted as a station)
• Minimum distance between two transceivers:	2,5 m	
• Maximum length for a transceiver cable:	50 m	

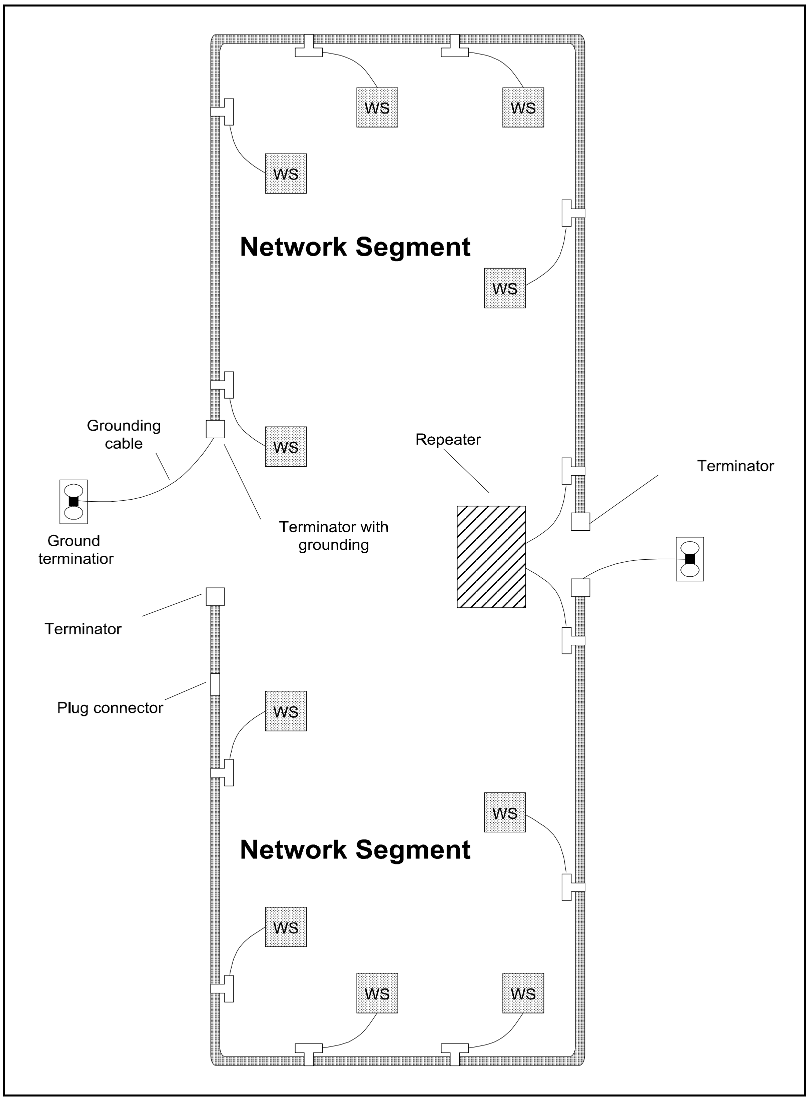


Fig. 3-5: Example for a thick-ethernet network

3.3.2 Regulations and specifications

Certain basic regulations must be observed when planning or installing a network:

- The cabling of a network consists of a number of segments. Both ends of each segment must be terminated with its characteristic impedance (terminator) of 50 Ω.

- According to the ISO 8802.3 specification the maximum length of an individual segment is 500 m.

- Any segment may consist of a number of different parts. In this case you must ensure that the length of each part is an odd multiple of 23,4 m to eliminate reflections at the joints and the ends of any portion of the cable.

- The length of any part may thus be calculated as follows:
  TSL = (2n + 1) * 23,4 m;   N = 0, 1, 2, 3...

- Wherever possible, the network should consist of a homogeneous segments of coaxial cable as every connector introduces losses. When it becomes necessary to extend existing segments of coaxial cable you should always use standard portions of the cable. Please note that the maximum length of a segment may not exceed 500 m.

- The screen of the coaxial cable may only be grounded at a single point on the entire cable. This point must posses a defined low impedance. All other connectors and terminators must be insulated from ground.

- Any two stations on the network may only have a single connection between them. Where multiple physical paths exist the resulting interference will be interpreted as a signal and cause collisions which may abort the data transfer.

- Single drop-cables are not permitted.

Guidelines for the routing and installation of network components:

- As a rule, a minimum distance of 1m is prescribed with respect to electrical equipment, cabling and other components that could produce electromagnetic and electrostatic fields. This includes parallel runs of power or high-tension cables, circuit breakers etc.

- The ISO-specifications allow for a maximum field strength of 2 V/m at frequencies between 10 kHz and 30 MHz and 5 V/m at frequencies between 30 MHz and 1 GHz.

- Once a coaxial cable segment and all its components have been installed it is essential that the worst-case reflections on the cable are determined. These should never exceed 7% of the injected amplitude of the original signal. The reflection coefficient can be checked by means of a reflectometer.

- The overall loop resistance of a coaxial cable segment together with all its plugs and connectors may not exceed a maximum of 5Ω.

- The minimum radius of any bend in the cable may not be less than 21 cm. Where multiple bending is anticipated the radius must not be less than 40 cm.

3.3.3 Technical data thick-ethernet

Coaxial cable

Impedance	50 Ω ± 2 Ω
Impedance linearity	± 3 Ω with a period > 2 m
Inductivity	0,21 µH/km
Capacity	85 pF ± 5 pF
Attenuation	≤ 18 dB/km for a 10 MHz sine wave
	≤ 12 dB/km for a 5 MHz sine wave
Propagation	≥ 0,77 c (velocity of light)
Loop resistance	≤ 10 m Ω/m at 20° C
Minimum radius for bends, single bend	21 cm
Minimum radius for bends, multiple bend	40 cm
Diameter of center conductor	2,17 mm ± 0,013 mm
solid copper	
Thickness of the dielectric	1,99 mm
Thickness of the screen	1,06 mm ± 0,089 mm
Screening effectiveness	≥ 92 %
Thickness of the outer insulator	1,25 mm ± 0,089 mm
Outside diameter for polyvinylchloride (PVC)	10,287 mm ± 0,178 mm
Outside diameter for fluorpolymer	9,525 mm ± 0,178 mm

Transceiver cable

Maximum operating voltage	30 V
Maximum current capacity	2,8 A
Capacity of a single pair of lines	55 pF/m
Impedance	78 Ω ± 5 Ω
Screening effectiveness	78 %
Maximum attenuation	3 dB / 50 m

The yellow outer cover of the cable bears imprinted marks at a distance of 2,5 m ± 5 cm. These marks indicate the positions where a transceiver may be connected. The coaxial connectors at the end of every cable are N-type plugs.

3.4 Combining thin/thick ethernet cable in networks

It is possible to combine thin-ethernet and thick-ethernet cabling in a single network.. The lower price of the thin-ethernet cable may provide a more cost-effective solution than a system that is purely based on thick-ethernet cable.

When thick-ethernet cable is used to link repeaters the distance may be substantially larger than if the link was provided by a thin-ethernet cable.

You may use one of the two methods below to establish a network that combines thick and thin-ethernet cable:

- Connecting the thin-ethernet cable segment to a network based on thick-ethernet cable via a repeater. It is possible to combine a maximum of five unlike network segments (using four repeaters) in this manner.

- It is also possible to use thin and thick-ethernet cables within the same network segment. The structure of such a network is described below under the heading “Combination thin/thick ethernet cable hardware".

3.4.1 Combination of thin/thick ethernet cable hardware

Networks that use a combination of thin and thick ethernet cable employ the network hardware as the individual thin-ethernet and the thick-ethernet networks would if they are not combined. It is however necessary to include adapters that provide the interconnection between the different types of cable. Fig. 3-6 shows two such adapters.

- N-type jack to BNC-jack
- N-type plug to BNC-plug

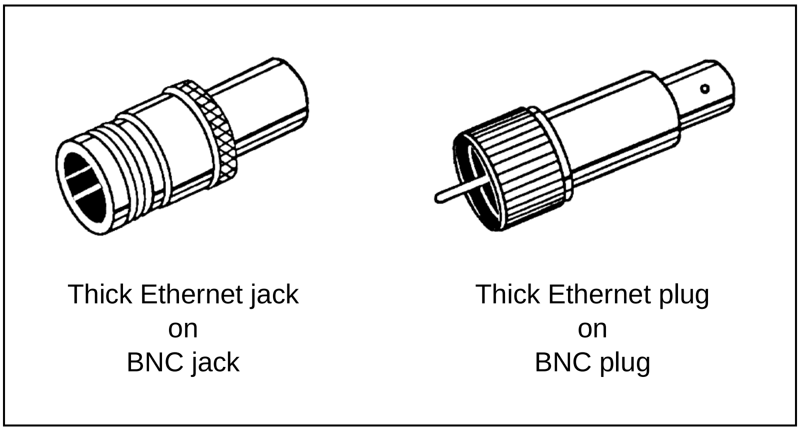


Fig. 3-6: Adapters for linking unlike segments

3.4.2 Thin/thick ethernet cable combination network-layout

This chapter describes a method of combining thin ethernet cable and thick ethernet cable in a single network segment. This method employs as much thin ethernet cable as possible. Fig 2-7 illustrates this technique of combining thin and thick ethernet cable.

The length of a network employing a combination of thin and thick ethernet cable may be between 185 m and 500 m. In this case the minimum length has been specified as 185 m as shorter segments can consist entirely of thin ethernet cable. The maximum length of 500m is determined by the physical limitations of the thick ethernet cable.

You may use the following equation to calculate the maximum length of the thin ethernet cable in a combination network:

$$\frac{500\mathrm{m} - L}{3.28} = t$$

(L = length of network segment that must be implemented

t = maximum length of the thin ethernet cable)

e.g. you must implement a network segment of 457 m and reduce the cost of the network hardware to a minimum. You would use the equation above to determine the maximum length of the thin ethernet cable as follows.

$$\frac{500\mathrm{m} - 457\mathrm{m}}{3.28} = 13{,}1\mathrm{m}$$

You may thus use up to 13,1 m of the cost-effective thin ethernet cable. For the remainder of 443,9 m of the network segment you must use thick ethernet cable.

Please note: you can interconnect up to 5 different network segments!

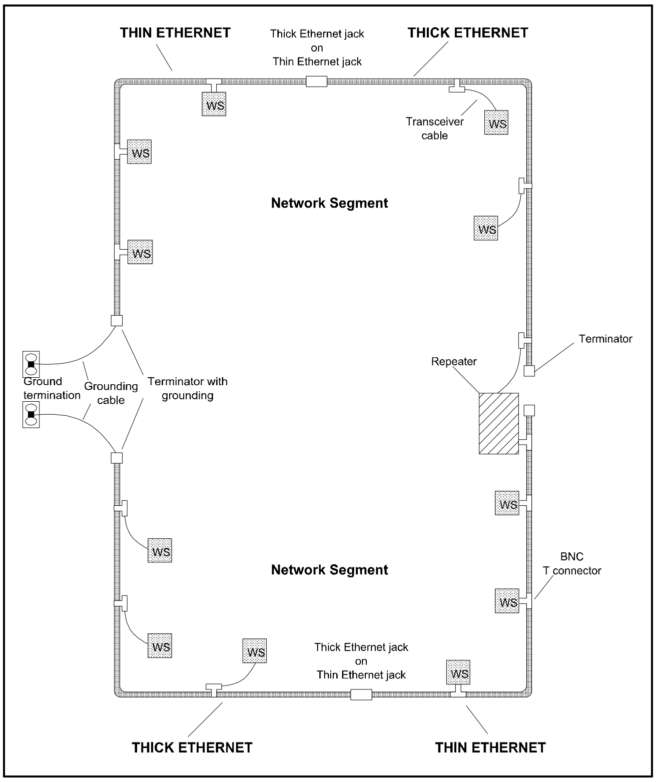


Fig. 3-7: Example of a network combining thin and thick ethernet cable

3.5 Twisted Pair

Twisted Pair network hardware

A twisted pair-network can only have a star-type configuration. This topology requires a hub that supports the star-type configuration.

The following paragraphs contain a description of the hardware required for implementing a twisted pair network.

Network adapter

The network adapter which is installed in every station on the network provides the interface that allows every station to communicate with every other station..

Hub (star-type configuration)

The Hub is the central element that provides the facilities required to implement an ethernet network using twisted pair cable. Every station is connected directly to the hub to form the ethernet network. A hub operates like a repeater. It may be interfaced with thin or thick ethernet cable.

Twisted pair-cable

The twisted pair cable consists of a four-core cable where the cores are twisted together in pairs. The conductors have a diameter between 0,4 and 0,6 mm.

3.5.1 Twisted pair-cable network-layout

Here follows a summary of the restrictions and the rules that apply to twisted pair networks

Restrictions

- Maximum number of hubs per segment: 2
- Maximum segment length: 100 m

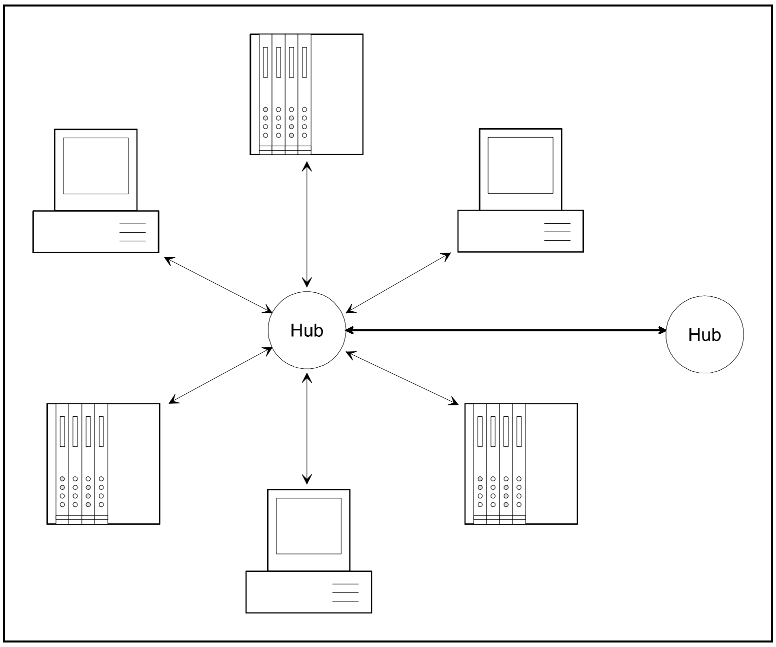


Fig. 3-8: Star topology of a twisted pair network

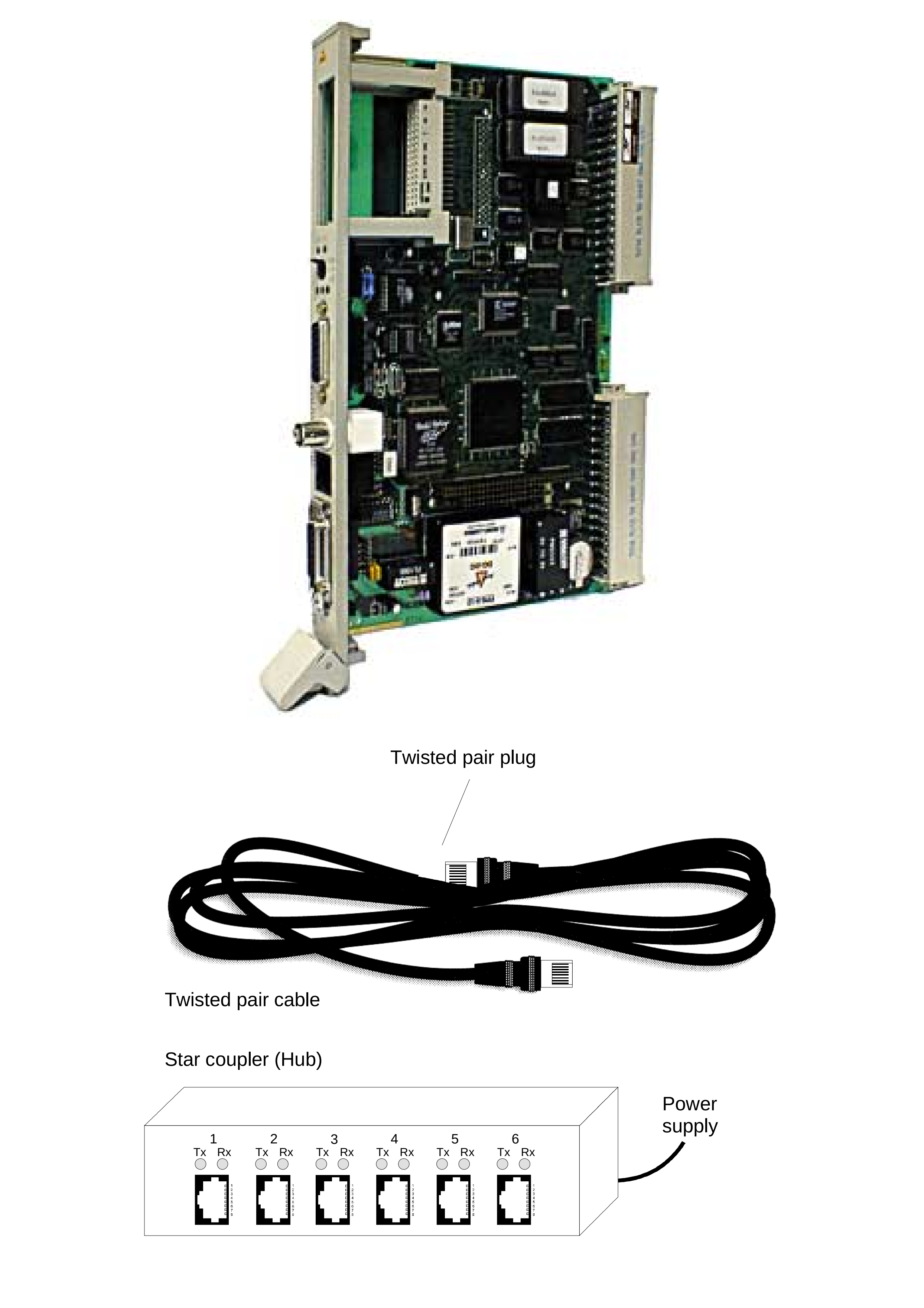


Fig. 3-9: Twisted pair network hardware

3.6 Planning a network-layout

At this point you are aware of the limitations of the different cabling systems, and you can provide answers to the following questions with respect to the network that you have to implement:

Determination of network requirements

- What is the extent of the coverage of the proposed network?
- What is the optimum number of network segments required to satisfy the physical (space, interference) conditions applicable to the proposed system?
- How many network stations (H1 TCP/IP, IPC, PC, transceiver, possible bridges) should be connected to the network?
- What is the distance between the different stations on the network?
- What is the expected increase in coverage and number of connections that the system must be able to handle?

Drawing a plan of the network

Prepare a drawing of the plan of the network! Identify every item of hardware (like cables, transceivers, amplifiers, termination resistors). Observe the rules and limitations that were outlined in the previous chapters.

Measure the distance between all components to ensure that they are within the prescribed parameters (min. distance, max. length...).

3.7 Standards and specifications

The main characteristic of a LAN is that it provides a single physical communication path. The physical communication medium may consist of:

- one or more electrical connections (twisted cables)
- coaxial cable (triaxial cable)
- fiber-optic cable

The communications between individual stations is subject to certain standards and rules that must be met by every station on the network. These determine the form of the communication protocol, the access method to the LAN and other basic principles that are important for communicating on the network.

The VIPA H1-network was developed to satisfy the ISO standards and specifications.

The standards and specifications for networking systems were determined by international and national committees.

ANSI **A**merican **N**ational **S**tandards **I**nstitute

This body is currently busy formulating agreements for high-speed LANs (100 MB/s) based on fiber-optic technology in ANSI X3T9.5. (FDDI) Fiber Distributed Data Interface.

CCITT **C**ommittee **C**onsultative **I**nternationale de **T**elephone et **T**elegraph.

Amongst others, this advisory committee produces the specifications governing the interfacing of industrial communication networks (MAP) and of office networks (TOP) to Wide Area Networks (WAN).

ECMA **E**uropean **C**omputer **M**anufacturers **A**ssociation.

generates the different standards for MAP and TOP.

EIA **E**lectrical **I**ndustries **A**ssociation (USA)

this committee determines standards like RS-232 (V.24) and RS-511.

IEC **I**nternational **E**lectrotechnical **C**ommission.

This committee sets specific standards, e.g. those for the Field Bus.

ISO **I**nternational **O**rganisation for **S**tandardization.

This association of national standards offices has determined the OSI-model (ISO/TC97/SC16). It outlines the framework for the standards that are applicable to data communications. The ISO standards are converted to the different national standards like UL and DIN.

IEEE **I**nstitute of **E**lectrical and **E**lectronic **E**ngineers (USA).

The project group 802 determines LAN-standards for communication speeds from 1 to 20 MB/s. IEEE standards are often the basis ISO-standards, e.g. IEEE 802.3 = ISO 8802.3.

The serial Industrial Ethernet H1 LAN is an open communication system that can be used for the exchange of data between all devices that comply with a set of common rules. These devices may also include units manufactured by other companies.

Data is communicated on the basis of the OSI ISO-reference model. This model provides the specifications for all the processes, rules and the strategy for common data communications.

4 The CP143 H1 / TCP/IP network board

4 The CP143 H1 / TCP/IP network board

4.1 The construction of the module

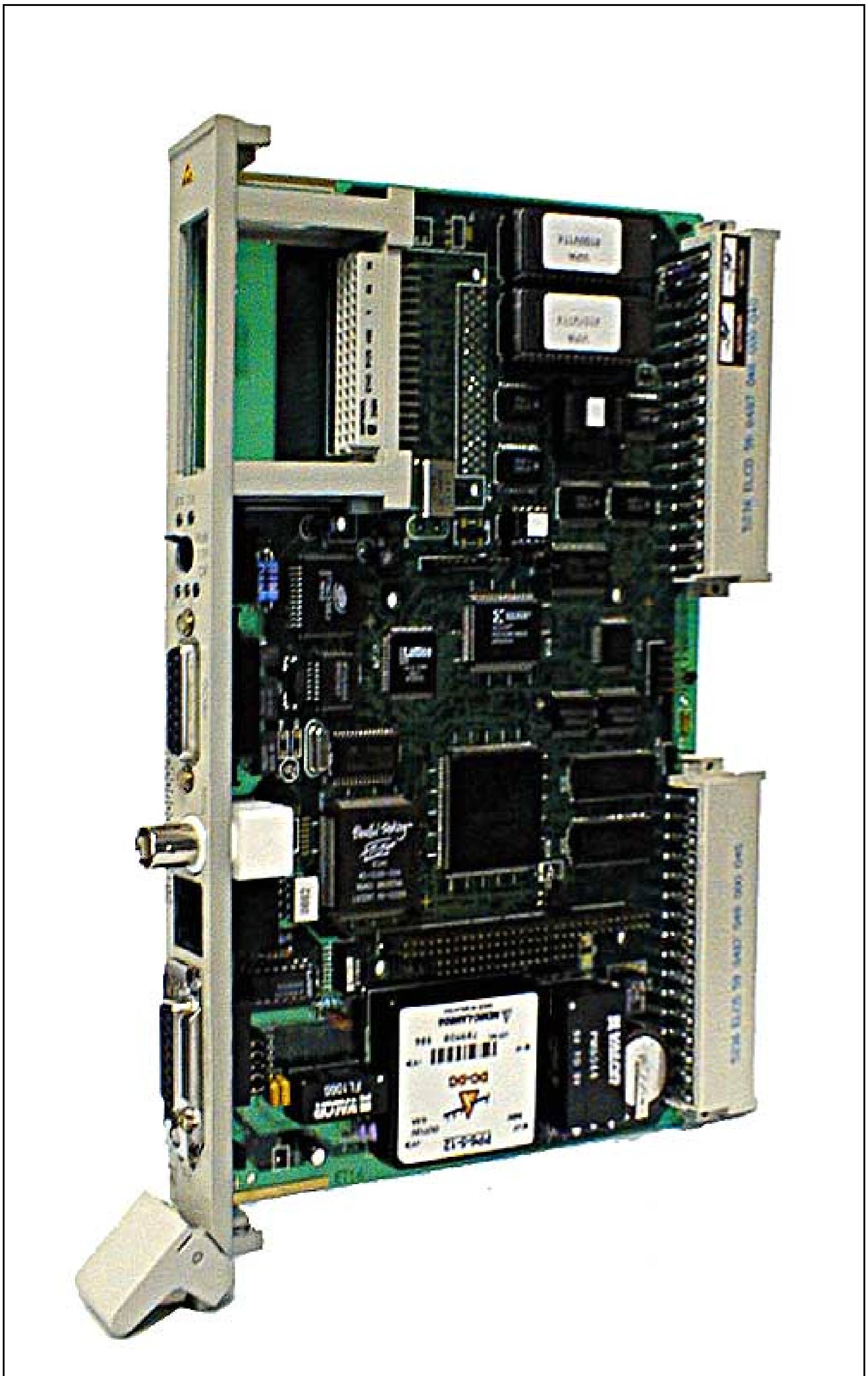


Fig. 4-1: Construction of the CP143 H1 / TCP/IP

4.1.1 Components of the module

- 80486 SLC 33 MHz CPU
- 1MB main memory
- PG interface which is compatible with the Siemens CP1430
- EPROM socket
- 4KB dual-port RAM as CP interface
- DIN interface to the PLC
- An AUI connection (IEEE 802.3) for interfacing to a bus cable via a transceiver, RJ45 signal may be connected to the AUI socket (Siemens compatible)
- RJ45 interface connection for ethernet
- Thin ethernet connector (BNC)
- power supply for transceiver

Controls and indicators located on the front panel

- LEDs for

TX	(yellow)	indicates that data is being transmitted.
RX	(yellow)	indicates any data on the network.
RUN	(green)	The module has been synchronized, i.e. the PLC has completed the SYNC-procedure.
COMM	(yellow)	error occurred during the boot phase / FLASH-ROM is burned.
STOP	(red)	The module is busy initializing.
15 V	(green)	15 V supply available
	(red)	under voltage
	Both LEDs off indicates that no power is available.	

- Slot for EPROMs
- 15-pole D-type plug for the PG interface
- AUI interface for a transceiver
- BNC socket for thin ethernet
- RJ45 socket for ethernet
- RUN-/STOP-/RESET switch

  RESET sets the module to defauft-setting. To reset the module you have to keep pressed the RUN-/STOP-/RESET-switch in position OG about 5 seconds.

4.1.2 Jumper settings

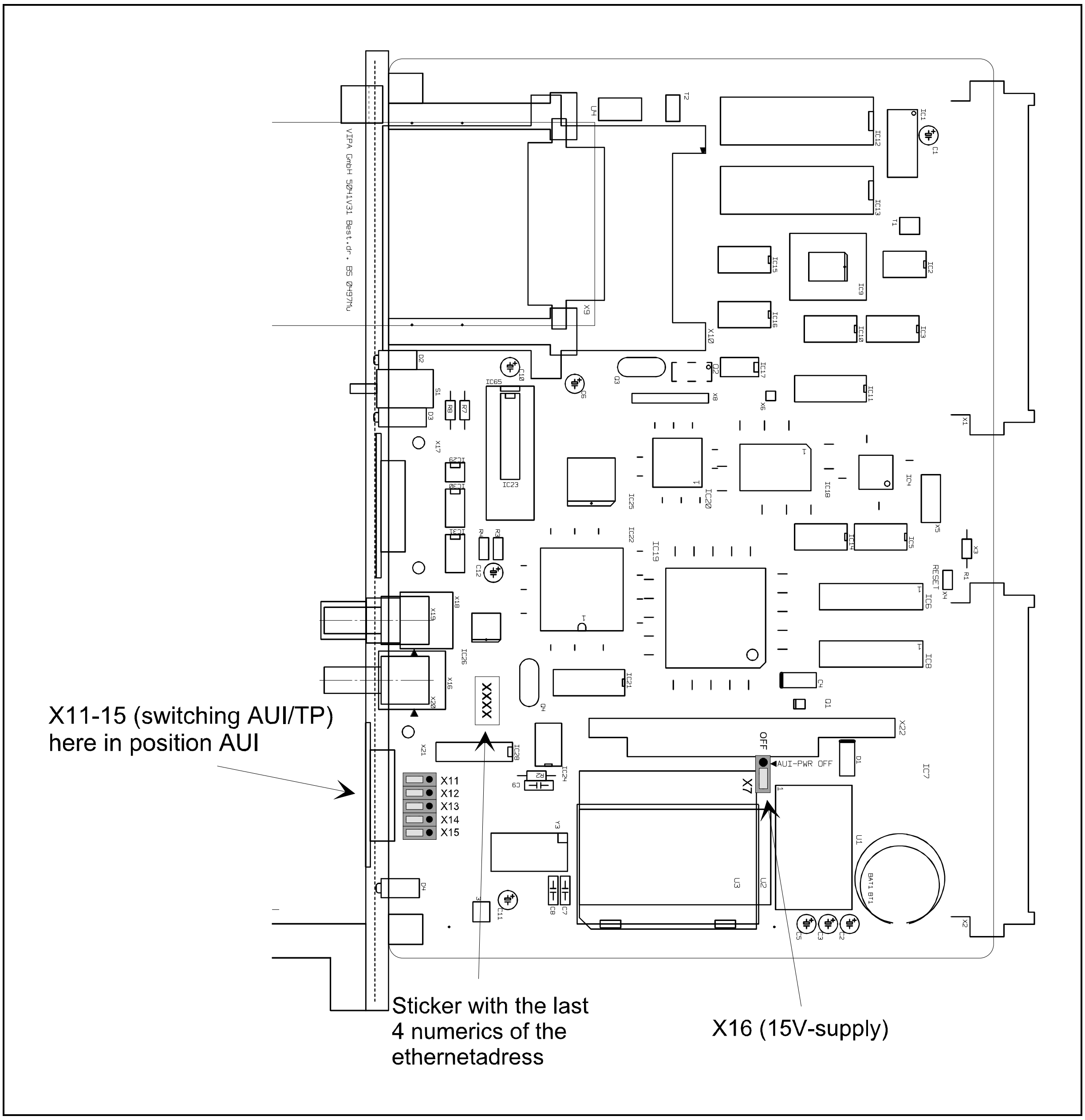


Fig. 4-2: Location of jumpers

! *Jumpers X11 to X15 must either all face left or face right! Any other combination could cause permanent damages to the module!*

The configuration of this module is determined by means of jumpers. You must install 5 jumpers to change from AUI to TP mode. X7 is used to enable the 15V supply voltage for the transceiver. The remaining jumpers on the module do not affect the configuration. However, they must not be altered to ensure the module operates properly.

4.1.2.1 AUI / TP (X11-X15) changeover

This jumper determines the pin-allocation of the 15-pin socket. You may choose between twisted pair (TP) and AUI.

Jumpers X11 to X15 must always be in the same position.

1-2	linked	The 15-pin socket is set for AUI.	(default)
2-3	linked	The 15-pin socket is set for twisted pair operation.	(Siemens compatible)

! *Jumpers X11 to X15 must either all face left or face right! Any other combination could cause permanent damages to the module!*

4.1.2.2 Selection of +15V supply voltage (X7)

This jumper enables the +15V supply for the transceiver:

1-2	linked	transceiver supply voltage off.	
2-3	linked	transceiver supply voltage on.	(default)

! *The jumpers that are not described in this section must be installed for normal operation but they may not be changed!*

4.1.3 Sockets and plugs

4.1.3.1 The position of sockets and plugs

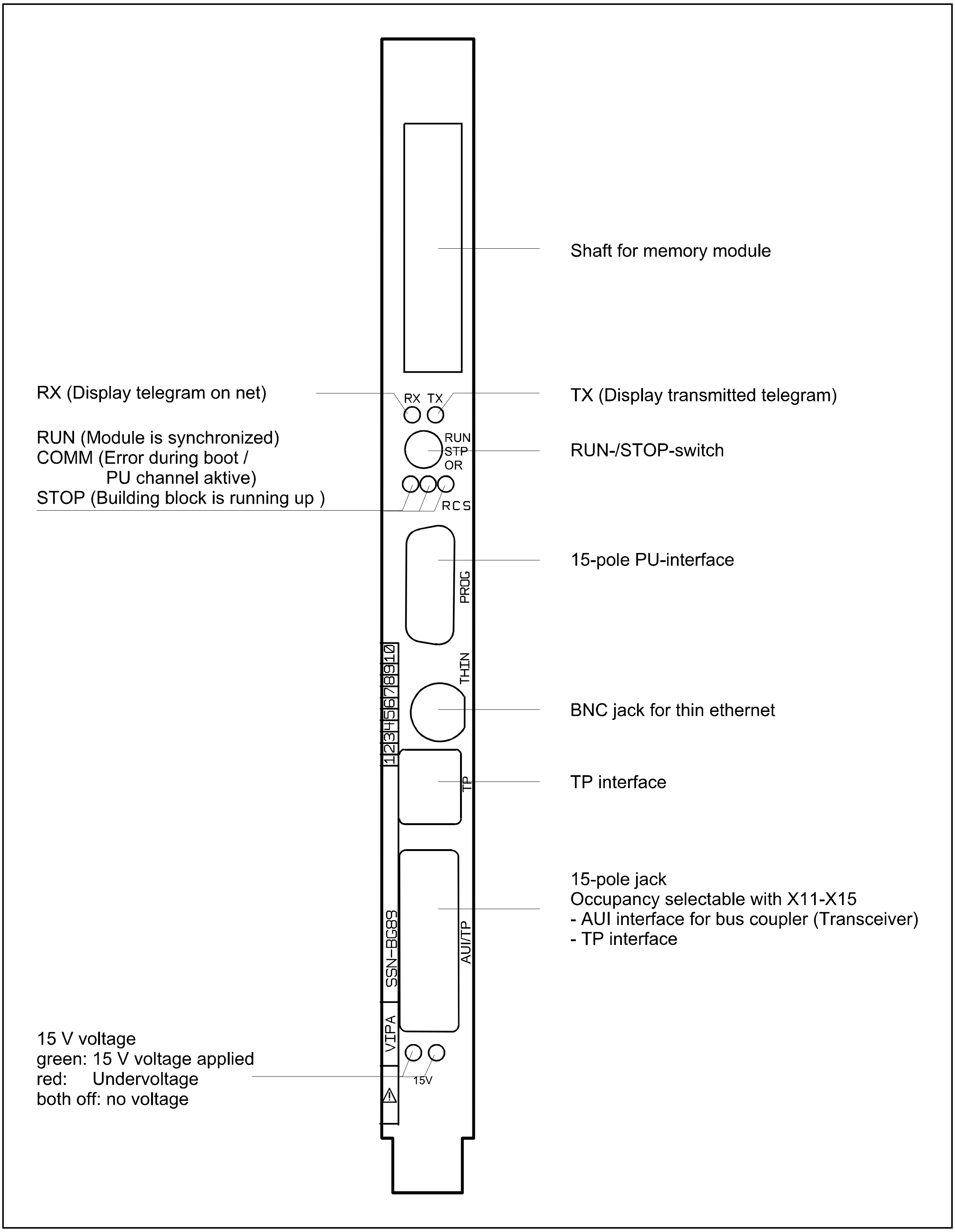

Shaft for memory module
RX (Display telegram on net)
RX
TX
TX (Display transmitted telegram)
RUN (Module is synchronized)
COMM (Error during boot / PU channel aktive)
STOP (Building block is running up)
RUN
STP
OR
RUN-/STOP-switch
R C S
15-pole PU-interface
PROG
1 2 3 4 5 6 7 8 9 10
THIN
BNC jack for thin ethernet
TP
TP interface
15-pole jack
Occupancy selectable with X11-X15
- AUI interface for bus coupler (Transceiver)
- TP interface
SSN-BG89
AUI/TP
VIPA
15 V voltage
green: 15 V voltage applied
red: Undervoltage
both off: no voltage
15V

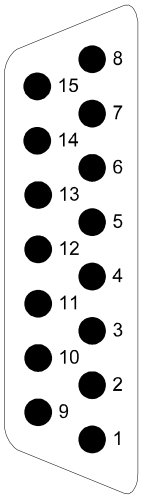
Fig. 4-3: Front view CP143 H1 / TCP/IP

4.1.3.2 Pin-configuration for sockets and plugs

The figures show the respective sockets when the module is viewed from the frame. The top of the figure is equivalent to the top of the frame.

4.1.3.2.1 Pin-assignment for the AUI-/TP socket (15-pole)

The 15-pole D-type socket connects to a transceiver via a drop-cable. It has a sliding interlock. The pin-assignment for this connector for both, the AUI interface and the TP interface (twisted pair) is depicted below.



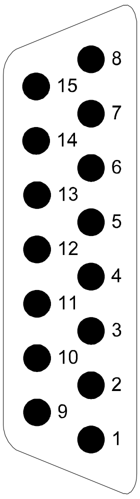
Pin	AUI	TP
1	Signal ground	-
2	CLSN (Collision+)	-
3	TRMT (Transmit+)	TRMT (Transmit+)
4	Signal ground	-
5	RCV (Receive+)	RCV (Receive+)
6	12V return	Link to 7
7	-	Link to 6
8	Signal ground	-
9	CLSN (Collision-)	-
10	TRMT (Transmit-)	TRMT (Transmit-)
11	Signal ground	-
12	RCV (Receive-)	RCV (Receive-)
13	+12V	-
14	Signal ground	-
15	-	-

Tab. 4-1: Pin configuration of the 15-pole AUI-/TP socket

Pins that are not connected may not be used as this could damage the module or the connected peripheral!

4.1.3.2.2 Pin configuration for the D-type socket with PG interface (15-pole)

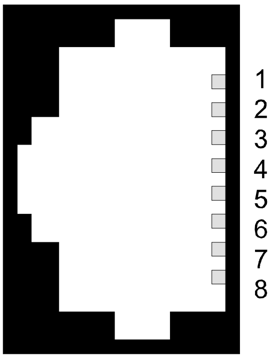
The 15-pole D-type socket with the screw-type locks is intended for the connection of a programmer. The following table shows the pin-configuration for the socket:



Pin	Signal
1	MEXT (external ground, screen)
2	TTY IN - (current output)
3	-
4	+24V
5	Ground (internal ground)
6	TTY OUT + (current input)
7	TTY OUT - (current output)
8	MEXT (external ground)
9	TTY IN + (current input)
10	24V return
11	20mA current source of the transmitter
12	Ground (internal ground)
13	20mA current source of the receiver
14	Master signal
15	Ground (internal ground)

Tab. 4-2: Pin configuration of the PG interface

4.1.3.2.3 Pin configuration of the TP socket (RJ45)



Pin	Signal
1	Transmit +
2	Transmit -
3	Receive +
4	-
5	-
6	Receive -
7	-
8	-

Tab. 4-3: Configuration of the TP socket (RJ45)

4.1.3.2.4 Base connector (48-pole knife-edge connector)

Base connector X1

d	b	z	
nc	M	+5V	2
UBAT	PESP	nc	4
ADB12	ADB00	/CPKL	6
ADB13	ADB01	/MEMR	8
ADB14	ADB02	/MEMW	10
ADB15	ADB03	/RDY	12
nc	ADB04	DBO	14
nc	ADB05	DB1	16
nc	ADB06	DB2	18
nc	ADB07	DB3	20
nc	ADB08	DB4	22
nc	ADB09	DB5	24
nc	ADB10	DB6	26
DSI	ADB11	DB7	28
+24V	BASP	M24V	30
nc	M	nc	32

Base connector X2

d	b	z	
nc	M	+5V	2
nc	nc	nc	4
nc	nc	nc	6
nc	nc	nc	8
nc	nc	nc	10
nc	nc	nc	12
nc	nc	nc	14
nc	nc	nc	16
nc	nc	nc	18
nc	nc	nc	20
/TxDSN	nc	nc	22
nc	nc	nc	24
nc	/RxDSN	nc	26
nc	nc	nc	28
nc	nc	M24V	30
nc	M	+24V	32

4.1.3.2.5 EPROM socket

	a	b	c
1	VCC	GND	A12
2	A2	A1	A0
3	A5	A4	A3
4	A8	A7	A6
5	A11	A10	A9
6	-RD	A14	A13
7	D9	D8	-WR
8	D12	D11	D0
9	D15	D14	D13
10	D2	D1	D0
11	D5	D4	D3
12	K1	D7	D6
13	K2	-CS3	-CS1
14	K3	-CS4	-CS2
15	K4	PSW	UBAT
16	K5	GND	VPP

4.2 The CP143plus firmware

Fig. 4-4 shows the structure of the CP143plus firmware. This figure contains those components that interact to determine the characteristics of the CP.

4.2.1 Components

The individual components communicate via the operating system functions of a real-time kernel (RTK). In addition to this, class libraries have been defined which provide a defined CALL interface.

4.2.1.1 Interface

The interface between the PLC and the CP is provided by a dual-port-RAM that has been divided into four equal segments, so-called tiles. In a multi-processor PLC every CPU may communicate independently with the CP143plus by means of one of these four tiles.

The handler modules send-request and receive-request have been implemented by the use of interrupts (interrupt handler) and by exchanging data via the dual-port-Ram.

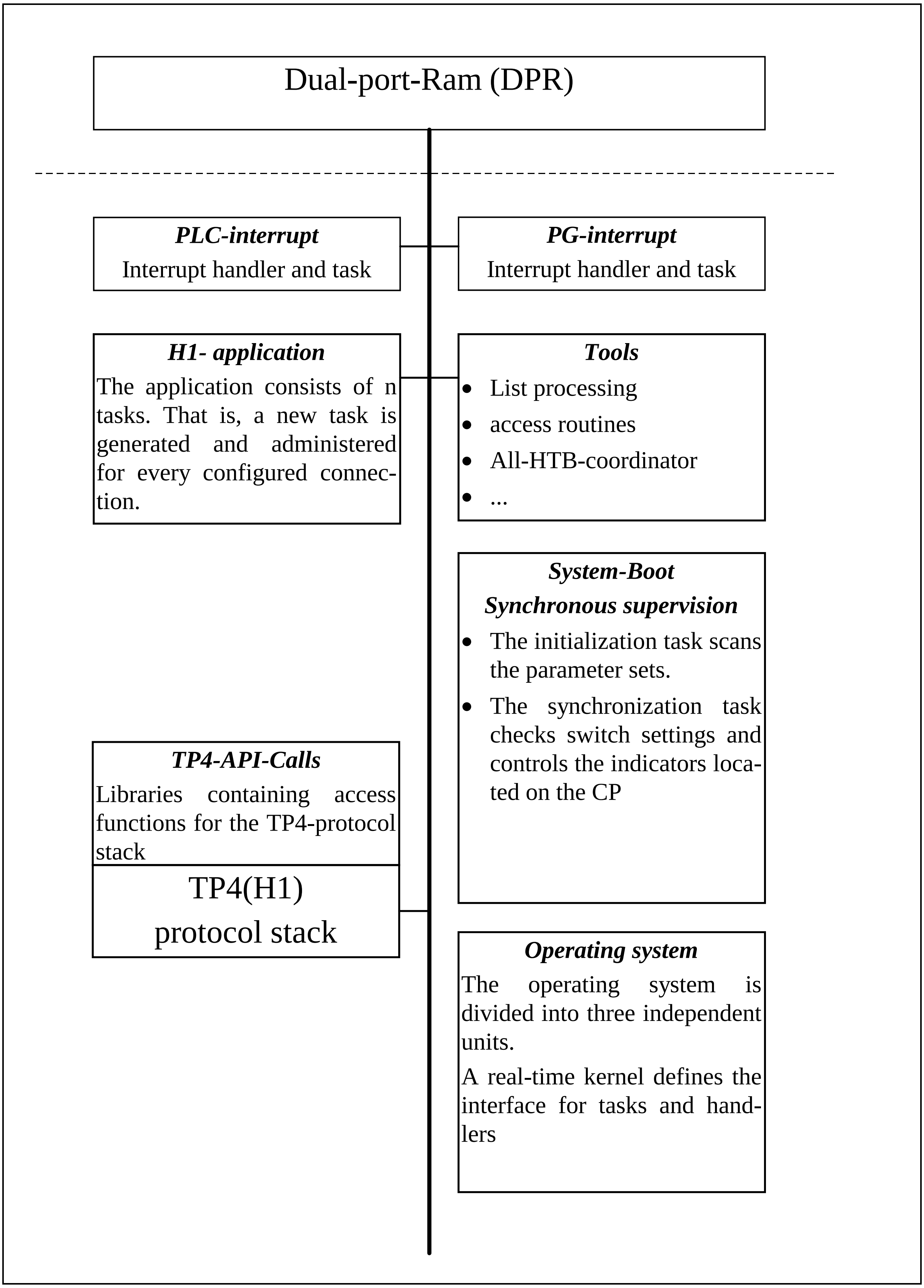
4.2.1.2 H1-application

The H1-application is the protocol controller for the functions defined for the CP.

4.2.1.3 H1-interface

The H1-API-calls are available as a class library and they define the interface for the protocol stack.

4.2.2 The structure of the CP143plus



The application consists of n tasks. That is, a new task is generated and administered for every configured connection.

- List processing
- access routines
- All-HTB-coordinator
- ...

- The initialization task scans the parameter sets.
- The synchronization task checks switch settings and controls the indicators located on the CP

Libraries containing access functions for the TP4-protocol stack

The operating system is divided into three independent units.

A real-time kernel defines the interface for tasks and handlers

Fig. 4-4: The structure of the CP143plus

4.2.3 Further information on the CP143plus

4.2.3.1 System properties

The system properties of a CP must not be regarded as limitations or as design faults. Instead, certain functions may not be accessible or they have been removed for the security of the overall system.

- The run-up time of the CP143plus is app. 10 seconds. For this reason the requirements of balanced synchronization characteristics (an HTB synchronization module waits between 8 and 10 seconds, depending on the type of CPU) can not be met. Thus the start-up OBs 20-22 must perform the following actions:
    - poll the PAFE of the HTB in synchronous mode until this becomes 0. After 10 seconds the tile will be set to active by the CP.
    - If the synchronous HTB still detects a PAFE after 15 seconds then the PLC, the configuration of the PLC or the CP is incorrect. In this case the respective configuration must be checked.
- Depending on the sequence in which the synchronous HTBs were initiated, the CP may detect a synchronous request and re-boot when more than one tile of a CP must be synchronized. Under these conditions the CP would never be synchronized. Cure: switch the CP to STOP by means of the respective switch, then synchronize all the required tiles. Change the switch back to RUN. At this point the CP has been synchronized. This reaction of the system can only occur during commissioning, and then only, if the number of tiles of the CP is increased and the new synchronous HTB is not accessed as the last one in the sequence.
- Tasks having a priority of 0/1 can only send and/or receive the amount of data as defined by the synchronous HTB. Tasks of this type are not blocked. This results in a maximum data transfer rate of 512 bytes per task and a block size of 255 (see also block size).
- RECEIVE tasks, that are mirrored as broadcast communication, can not receive all data messages from a fast cyclic transmitter. Messages that have not been received are discarded to protect the system.

4.2.3.2 Generation of a multicast address from the ethernet address

The TP4 protocol stack generates a MC address from the ethernet address. This function is required as the ethernet controller (hardware) uses the MC address for message filters.

4.2.3.3 Broadcast

The broadcast address is always FFFFFFFFFFFFh.

4.2.3.4 block size

Every configured interface of the CP must be synchronized by the handler module SYNCHRON when a PLC is started. This applies to every type of start-up of the respective PLC:

- OB20 for a new start
- OB21 for a manual re-start
- OB22 for a re-start after a mains power failure

block size	CP block size in bytes [1]
0	Default block size for a CPU. This may differ, depending on the type of CPU.
1	16
2	32
3	64
4	128
5	256
6	512 (for the READ/WRITE function)
255	512

Tab. 4-4: block size

[1)] The first transfered block for the READ/WRITE function is 16 Bytes (length of READ/WRITE Header) shorter as the set block length.

4.3 CP143 TCP/IP firmware

Fig. 4-5 shows the structure of the CP143 TCP/IP firmware. This figure contains those components that interact to provide the functionality of the CP.

4.3.1 Components

The individual components communicate via the operating system functions of a real-time kernel (RTK). In addition to this, class libraries have been defined which provide a defined CALL interface.

4.3.1.1 Interface

The interface between the PLC and the CP is provided by a dual-port-RAM that has been divided into four equal segments, so-called tiles. In a multi-processor PLC every CPU may communicate independently with the CP143 TCP/IP by means of one of these four tiles.

The handler modules send-request and receive-request have been implemented by the use of interrupts (interrupt handler) and by exchanging data via the dual-port-Ram.

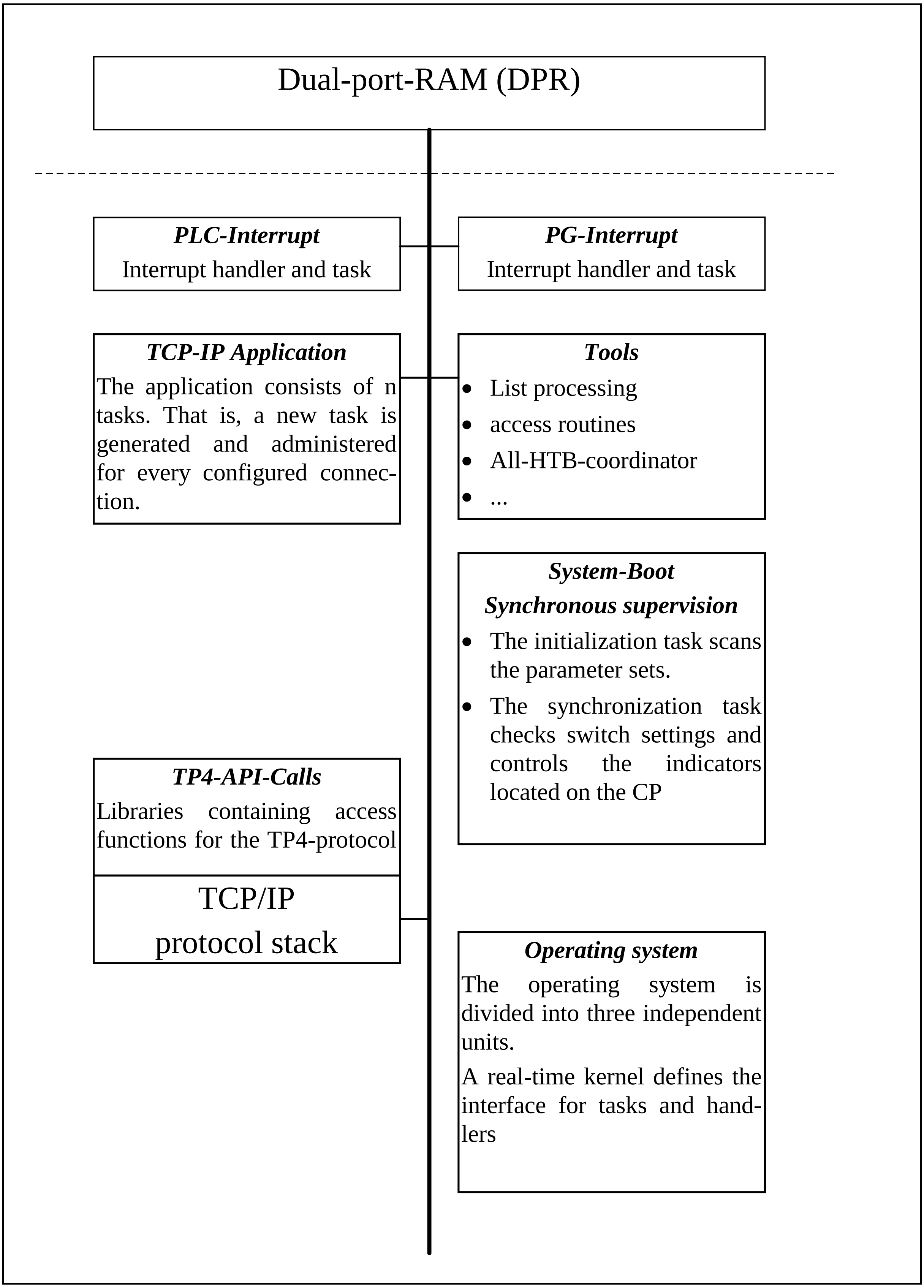
4.3.1.2 TCP/IP application

The TCP/IP-application is the protocol controller for the functions defined for the CP.

4.3.1.3 TCP/IP-Interface

The TCP/IP-API-calls are available as a class library and they define the interface for the protocol stack.

4.3.2 The structure of the CP143 TCP/IP



Dual-port-RAM (DPR)

PLC-Interrupt

Interrupt handler and task

PG-Interrupt

Interrupt handler and task

TCP-IP Application

The application consists of n tasks. That is, a new task is generated and administered for every configured connection.

Tools

- List processing
- access routines
- All-HTB-coordinator
- ...

System-Boot

Synchronous supervision

- The initialization task scans the parameter sets.
- The synchronization task checks switch settings and controls the indicators located on the CP

TP4-API-Calls

Libraries containing access functions for the TP4-protocol

TCP/IP

protocol stack

Operating system

The operating system is divided into three independent units.

A real-time kernel defines the interface for tasks and handlers

Fig. 4-5: The structure of the CP143 TCP/IP

4.3.3 Further information on the CP143 TCP/IP

4.3.3.1 System properties

The system properties of a CP must not be regarded as limitations or as design faults. Instead, certain functions may not be accessible or they are purposely removed for the security of the overall system.

- The run-up time of the CP143 TCP/IP is app. 10 seconds. For this reason the requirements of balanced syncronization characteristics (an HTB synchronization module waits between 8 and 10 seconds, depending on the type of CPU) can not be met. Thus the start-up OBs 20-22 must perform the following actions:
  - poll the PAFE of the HTB in synchronous mode until this becomes 0. After 10 seconds the tile will be set to active by the CP.
  - If the synchronous HTB still detects a PAFE after 15 seconds then the PLC, the configuration of the PLC or the CP is incorrect. In this case the respective configuration must be checked.
- Depending on the sequence in which the synchronous HTBs were initiated, the CP may detect a synchronous request and re-boot when more than one tile of a CP must be synchronized. Under these conditions the CP would never be synchronized. Cure: use the respective switch to STOP the CP, then synchronize all the required tiles. Change the switch back to RUN. At this point the CP has been synchronized. This is reaction of the system can only occur during commissioning, and then only, if the number of tiles of the CP is increased and the new synchronous HTB is not accessed as the last one in the sequence.
- The joker size (-1, 0xFFFF) is not permitted for the length of the ORG-format, the application (in this case the PLC user) must specify the length of the receive data exactly.
- Tasks having a priority of 1 can only send and/or receive the amount of data as defined by the synchronous HTB. Tasks of this type are not blocked. This results in a maximum data transfer rate of 512 bytes per task and a block size of 255 (see also block size).
- RECEIVE tasks, that are mirrored as broadcast communication, can not receive all data messages from a fast cyclic transmitter. Messages that have not been received are discarded to protect the system.
- The TCP/IP protocol stack has a global pool of buffers that contain the receive and transmit buffers. It is possible that system collisions occur in this area if:
  - the data for a receive task was not retrieved. After a certain period of time a lack of resources occurs, and other connections will issue disconnects once the respective predefined time-out has occurred. Proper communications will only be possible once the receive buffers of the original connection have been released (termination) or when the available data was retrieved by means of the RECEIVE HTBs.
  - one or more cyclic transmitters place a heavy load on the CP. The CP will also terminate connections when a lack of resources occurs.
  - a transmitter has sent two or more messages and the receiver did not have the opportunity to retrieve them. In this case the unknown type of data would cause a collision in the receiver. The CP does not permit that this situation occurs. The PLC application demands a definite receive size which excludes the joker-size. The receive size of prio 0/1-RECEIVE tasks is set implicitly by the defined block size (16, 32, 64, 128, 256, 512 bytes).

4.3.3.2 The relationship between the ethernet ADR for H1 and the IP-ADR for TCP/IP

The CP 143 TCP/IP has a default ethernet address after it was turned on for the first time or after it has been reset. This address is supplied as “Node Address” on the “test report” that accompanies the module.

You can determine the IP address from the ethernet address by means of the following calculation:

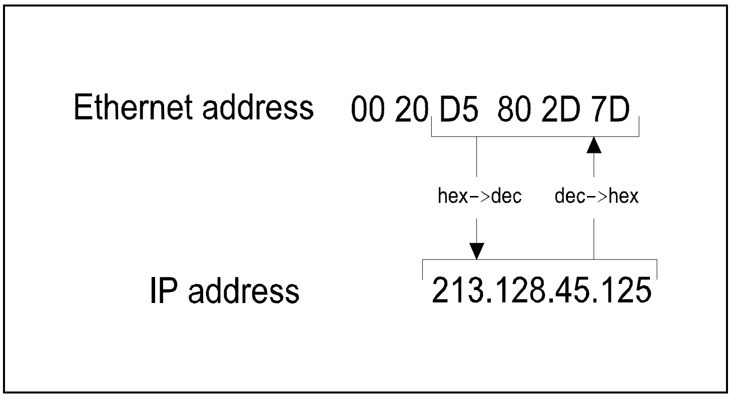


Fig. 4-6: Conversion of ethernet address into an IP address

The remote IP address is the address of the destination module. This is a 32-bit address that must be unique within the network. The address consists of 4 numbers that are separated from each other by a full stop.

4.3.3.3 Broadcast

The broadcast address is always 255.255.255.255.

4.3.3.4 block size

Every configured interface of the CP must be synchronized by the handler module SYNCHRON when a PLC is started. This applies to every type of start-up of the respective PLC:

- OB20 for a new start
- OB21 for a manual re-start
- OB22 for a re-start after a mains power failure

block size	CP block size in bytes [1]
0	Default block size for a CPU. This may differ, depending on the type of CPU.
1	16
2	32
3	64
4	128
5	256
6	512
255	512

Tab. 4-5: block size

[1)] The first transfered block for the READ/WRITE function is 16 Bytes (length of READ/WRITE Header) shorter as the set block length.

4.3.4 Organization format (ORG format)

The organization format is another name for an addressing structure that describes the data areas of the PLC. For READ and WRITE the ORG block is optional.

The ORG identifier defines the memory area that must be addressed.

The ERW identifier is important when data modules must be addressed. In this case the respective data module number must be entered into this location. The start address and the size address the memory area and they are entered in HIGH-/LOW format (Motorola formatted addressing)

Description	Type	Area
ORG identifier	BYTE	1..x
ERW identifier	BYTE	1..255
Start address	HILOWORD	0..y
Size	HILOWORD	1..z

Tab. 4-6: Organization format

PLC area	DB	MB	EB	AB
ORG identifier	01h	02h	03h	04h
Description	source/destination data from/into data module in main memory	source/destination data from/into clipboard	source/destination data from/into process image of the inputs (PAE)	source/destination data from/into process image of the outputs (PAA)
DBNR valid range:	DB, from where source data is read or to which destination data is written. 1...255	irrelevant	irrelevant	irrelevant
Start address Significance valid range:	DW-No., from where data is read or where data will be written. 1...2047	MB-No., from where data is read or where data will be written. 0...255*)	EB-No., from where data is read or where data will be written. 0...127	AB-No., from where data is read or where data will be written. 0...127
Quantity Significance valid range:	Length of the source/ destination data block in words 1...2048	Length of the source/ destination data block in bytes 1...256	Length of the source/ destination data block in bytes 1...128	Length of the source/ destination data block in bytes 1...128

Tab. 4-7: ORG-formats for PLCs (ORG identifier 01h-04h)

*) only up to MB199 for the AG150

PLC area	PB	ZB	TB	BS
ORG-identifier	05h	06h	07h	08h
Description	Source/destination data from/into peripheral module. Input modules for source data, output modules for destination data	Source/destination data from/into counter cells	Source/destination data from/into timer cells	Source/destination data from/into system data area (BS and BS')
DBNR	irrelevant	irrelevant	irrelevant	irrelevant
Start address Significance valid range:	PB-No., from where the data is read or where data is written. 0...127 digital periph. 128...255 anal. periph.	ZB-No., from where the data is read or where data is written. 0...255	TB-No., from where the data is read or where data is written. 0...255	Number of the BS-word from where the data is read or to where the data is written. 0...511
Quantity Significance valid range:	Length of source/ destination data block in bytes 1...256	Length of source/ destination data block in words (counter cell = 1 word) 1...256	Length of source/ destination data block in words (counter cell = 1 word) 1...256	Length of source/ destination data block in words 1...128

Tab. 4-8: ORG-formats for PLCs (ORG-identifier 05h-08h)

PLC area	AS	DX	QB	SM
ORG-identifier	09h	0Ah	11h	10h
Description	Source/destination data from/into absolute addressed memory cells	Source/destination data from/into extended data module (for AG135U)	Source/destination from/in periph.-module in the ext. peripheral area. Input module for source data, output module for destination data (AG150U only)	Source/destination data from/in special flag area
DBNR valid range:	irrelevant	DX, from where source data is read or where destination data is written. 1...255	irrelevant	irrelevant
Start address Significance valid range:	absolute start address, from where the data is read or where data is written. 0h...FFFFh	DW-No. from where the data is read or where data is written. 0...2047	PB-No. from where the data is read or where data is written. 0...511	Special flag-Nr. from where the data is read or where data is written
Quantity Significance valid range:	Length of source/ destination data block in words 1...32767	Length of source/ destination data block in words 1...2048	Length of source/ destination data block in Bytes 1...256	Lenght of source/ destination data block in Bytes 1...1024

Tab. 4-9: ORG-formats for PLCs (ORG-identifier 09h-11h)

4.3.5 Structure of the PLC-header

The following messages lists the valid ORG identifiers. The entered "length" must never be -1 (FFFFh).

4.3.5.1 FETCH/WRITE

for WRITE

Request message

System identifier	="S"
	="5"
Header length	=16d
Ident. OP-code	=01
OP-code length	=03
OP-code	**=03**
ORG-block	=03
ORG-block length	=08
ORG-identifier	
DBNR	
Start-	H
address	L
Length	H
	L
Unused block	=FFh
Length unused bl.	=02
Up to 64K data but only for error No.=0	

Acknowledgement message

System identifier	="S"
	="5"
Header length	=16d
Ident. OP-code	=01
OP-code length	=03
OP-code	**=04**
Acknowl. block	=0Fh
ACK-block length	=03
Error No.	=No.
Unused block	=FFh
Length unused block	=07
not used	

for READ

Request message

System identifier	="S"
	="5"
Header length	=16d
Ident. OP-code	=01
OP-code length	=03
OP-code	**=05**
ORG-block	=03
ORG-block length	=08
ORG-identifier	
DBNR	
Start-	H
address	L
Length	H
	L
Unused block	=FFh
Length unused bl.	=02

Acknowledgement message

System identifier	="S"
	="5"
Header length	=16d
Ident. OP-code	=01
OP-code length	=03
OP-code	**=06**
Acknowl. block	=0Fh
ACK-block length	=03
Error No.	=No.
Unused block	=FFh
Length unused block	=07
not used	
Up to 64K data but only for error No.=0	

4.3.5.2 SEND / RECEIVE with TRADA (Transparent data exchange)

TRADA stands for transparent data exchange. Transparent data exchange allows to transmit data of varied length. A 16-byte header which indicates the length is preceding the user data which have to be transmitted.

For the use of TRADA it is permitted to give in the PLC user program a joker length within the length instruction.

Through the input of –1 as a lengths instruction into the RECEIVE-FB (parameter: ZLAE) a variable user data length (joker length) can be adjusted. For the use of a joker length the respective data length will be taken on.

The as following described header precedes with TRADA functionality a SEND command and is evaluated from the RECEIVE command.

SEND with the type TRADA
OP-code = **07**

System identifier	="S"
	="5"
Header length	=16d
Ident.OP-code	=01
OP-code lenght	=03
OP-code	=**07**
ORG-block	=03
ORG-block length	=08
ORG-identifier	
DBNR (**nicht relevant**)	
Start- H	
adress	L
Length	H
	L
Unused block	=FFh
Length unused block	=02
Up to 64K data but only for error No.=0	

→ length instruction of used data

length instruction

With the length instruction the number of the bytes of a data block is named.

If you synchronise with a block size of 6 (512 Byte), the length instruction results in words

4.3.6 Status- and error indicators

Status and error indicators are provided by the handler modules

- via the ANZW indicator word (information for task processing)
- via the configuration error byte PAFE (indicator for an incorrect configuration for a task)

4.3.6.1 Contents and structure of the indicator word ANZW

The indicator word has the following structure:

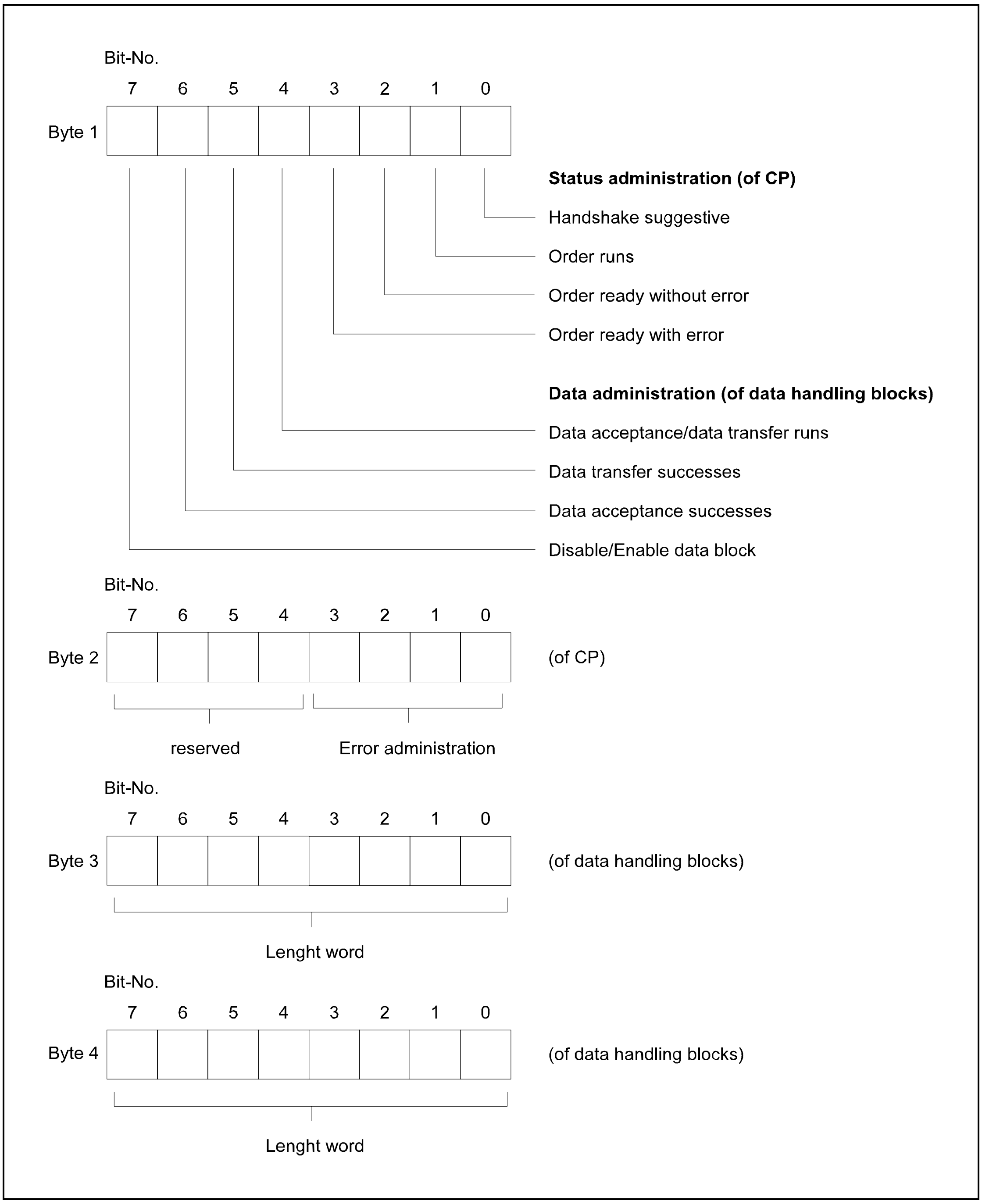


Fig. 4-7: The structure of the indicator word ANZW

- **Status management byte 1, bit 0 to bit 3**

  These indicate whether a task has already been started, whether an error has occurred or whether the task has been inhibited, e.g. if a virtual connection does not exist.

Bit 0: Handshake possible

Set: by the interface according to the "Reset"-indicator in the task status bit. Handshake possible (=1) is used by the RECEIVE-module. (Message available for PRIO 1 - or - RECEIVE-initiation possible for PRIO 2/3)

Check: by means of the RECEIVE-module: RECEIVE will only initiate a handshake with the CP when this bit is set. By the application: for RECEIVE-request (check, whether a message is available for PRIO 1).

Bit 1: Task active

Set: by the interface, when the task was sent to the CP.

Reset: by the interface once a task has been completed (e.g. acknowledgement received).

Check: by means of handler modules: a new task will only be issued when the “old” task has been completed.
by the user: to check whether it is sensible to trigger a new task.

Bit 2: Task completed without errors

Set: by the interface, once the respective task has been completed without errors.

Reset: by the interface when the task is started again.

Check: by the user, to check whether the task was completed without errors.

Bit 3: Task completed with errors

Set: by the interface, if the respective task was completed with errors. The reason for the error is available from the high part of the indicator word.

Reset: by the interface, when the task is started again.

Check: by the user: to check whether the task was completed with errors. If the bit for “task completed with error” is set, then the high byte of the indicator word contains the reason for the error.

- **Data management byte 1, bit 4 to bit 7**

  This area indicates whether a data transfer is still active or whether data transmission or reception has been completed. The "Enable / Disable" bit can be used to inhibit the transfer of data for the respective task. (Disable = 1; Enable = 0).

Bit 4: Data transmission / reception is active

Set: by handler modules SEND, RECEIVE, once the transmission/reception of data for a task has been started. The transfer may be executed by means of the ALL function (instead of a DMA) that was initiated by SEND-DIREKT.

Reset: by handler modules SEND, RECEIVE, after the data transfer has been completed (last partial block has been transferred).

Check: by the user: the user must not change the data of a task while the CP<< >>PLC transfer is active. This is not critical for PRIO 0/1 tasks as the transfer can be completed by a single run of the respective module. Larger quantities of data, however, can only be transferred in multiple blocks. This blocking requires a number of PLC-cycles. To ensure the consistency of the data the user must check whether the respective block is currently being transferred before proceeding with changes.

BIT 5: Data transmission completed

Set: by handler module SEND, after the transmission of data for the respective task has completed.

Reset: by handler module SEND, once the data transmission for a new task (new trigger) has started.
by the user: when the status was checked (edge triggered).

Check: by the user: this bit may be used to determine whether a data set for a task has been transferred to the CP, or whether a new data set may be supplied to the active task (e.g. cyclic data transfer)

Bit 6: Data reception completed

Set: by handler module RECEIVE, after the reception of data for the respective task has completed.

Reset: by handler module RECEIVE, once the data transfer from a new task (new trigger) to the PLC has started. By the user, when the status was checked (edge triggered).

Check: by the user: this bit indicates whether the transfer of the current data set for the respective task has been completed or whether the transfer of a new data set to the PLC has been completed.

Bit 7: Disable / enable blocking of data

Set: by the user: to prevent overwriting of a block of data by the RECEIVE-module or to stop transmission of a data block by the SEND-module (only for the first data block).

Reset: by the user: to release the respective data block.

Check: by means of handler modules SEND and RECEIVE. When bit 7 is set these modules will not transfer any data and indicate to the CP that an error has occurred.

- **Error management byte 2, bit 0 to bit 3**

This location contains the error flags returned by a task. The error flags are only valid when the bit “Task completed with errors” in the status bit is set as well.

The following error messages are possible:

0 no error
If the “Task completed with errors” bit is set, then the CP143 H1 / TCP/IP has re-established the connection, e.g. after a restart or a RESET.

1 incorrect Q/ZTYP at the HTB
The TYP-identifier for the task was incorrect.

2 The specified area does not exist in the PLC
The configuration of the DB (DBNR) for a task was incorrect.

3 The area in the PLC is too small
The sum of Q/ZANF and Q/ZLAE exceeds the limits of the area. The area-limits of data modules is determined by the size of the module. The limits for flags, timers, counters, etc. depend on the PLC.

4 QVZ-error in the PLC
The source or destination parameters refer to an area in the PLC that contains defective memory or no memory at all. The error QVZ can only occur for Q/ZTYP AS, PB, QB or when memory has been damaged.

5 Error in the indicator word.
The configured indicator word can not be processed. This error may occur when ANZW refers to a data word or double word that never did or is no longer located in the specified data module. That is, the DB is too small or does not exist.

6 no valid ORG format
The destination or the source for the data was not specified, neither in the handler module (Q/TYP="NN") nor in the connection module.

7 Reserved

8 no unused transport connections
The capacity for transport connections has been exceeded. Please delete unnecessary connections.

9 Remote error
An error occurred at the remote station during a READ/WRITE task.

A Connection error
The connection for a task has not been established. This error will disappear as soon as a connection can be established. When all connections of a CP are in the interrupted state the module or the LAN cable may have been damaged. This error may also occur due to a configuration error, e.g. incorrect addressing.

B Handshake error
This error may be a system error or the size of the data block is too large.

C Initiation error
A task was initiated by means of the wrong handler module or the transferred data block was too large.

D Termination after RESET
This is an operational indicator. The connection is terminated for priority 1 and 2 and it will only be re-established once the remote station is ready for a new connection. Priority 3 connections are terminated. These may be initiated again.

E Task with boot-function
This is an operational indicator. This is a READ/WRITE-PASSIV task and can therefore not be started by the PLC.

F Task does not exist
The task that was addressed has not been configured on the CP143 H1 / TCP/IP. This error may occur if the SSNR/A-NR combination has been entered incorrectly into the handler module or when a connection module has not been entered.

Bits 4 to 7 of byte 2 are reserved and intended for future expansion.

- **Length-word byte 3 and byte 4**

  The length word is used by the handler modules (SEND, RECEIVE) to deposit the amount of data for the respective task that has already been transferred. This means the amount of data already received for receive tasks and the amount of data transmitted for send tasks.

Write: this word is changed by SEND and RECEIVE while the data transfer is active. The "length-word" is calculated from:

current amount transferred + quantity of data already transferred

Reset: overwrite or by each new SEND, RECEIVE, FETCH task.

The “length-word” contains the current source or destination quantity when the bit "Task completed without error" or "Data transmission/reception completed" is set.

When the bit “Task completed with errors” is set, then the length word contains the amount of data that was transferred before an error occurred.

4.3.6.2 Important status and error indicators of the CP143 H1 / TCP/IP

Here follow important status and error indicators that may be signalled via the “indicator word”. The data consists of "HEX"-patterns that are similar to those of the status / control-var-test function of the PG on the PLC. The X indicates “arbitrary” or “irrelevant"; No. is the error number.

Indicator words that may occur are

Indicator word: X F X A

Error indicator "F" signifies that the respective task has not been defined on the CP 143. The task is blocked by the status indicator A (for SEND / FETCH and RECEIVE).

Indicator word: X A X A

Error indicator "A" signifies that the connection for the communication task has not been established. Status indicator "A" inhibits SEND as well as RECEIVE and FETCH.

Indicator word: X 0 X 8

The connection has been established again (e.g. when the CP has restarted) and SEND is enabled (SEND-communication task).

Indicator word: X 0 X 9

The connection has been established again, RECEIVE is enabled. (RECEIVE-communication task).

Indicator word: X 0 2 4

SEND was processed without errors, the data has been transferred.

Indicator word: X 0 4 5

RECEIVE was processed without errors, the data was received by the PLC.

Indicator word: X 0 X 2

The SEND, RECEIVE, READ or the WRITE task is active. For a SEND task the indicator signifies that the remote has not yet changed to RECEIVE. For a RECEIVE task the indicator signifies that the remote has not issued a SEND.

The following tables show the most important indicator word combinations:

Indicators for SEND

H1 condition	Prio 0/1	Prio 2	Prio 3/4
TCP/IP condition	Prio 1	Prio 2	Prio 3
after a restart	0 A 0 A	0 A 0 A	0 0 0 8
after connection has been established	X 0 X 8	X 0 X 8	.....
after a trigger	X 0 X 2	X 0 X 2	X 0 X 2
completed without errors	X 0 2 4	X 0 2 4	X 0 2 4
completed with errors	X No X 8	X No X 8	X No X 8
after RESET	X D X A	X D X A	X D X 8

Tab. 4-10: Indicator word conditions for SEND

Indicators for RECEIVE

H1 condition	Prio 0/1	Prio 2	Prio 3/4
TCP/IP condition	Prio 1	Prio 2	Prio 3
after a restart	0 A 0 A	0 A 0 A	0 0 0 1
after connection has been established	X 0 X 4	X 0 0 9	.....
after a trigger	X 0 X 2	X 0 X 2	X 0 X 2
Message available	X 0 X 1	.....	.....
completed without errors	X 0 4 1	X 0 4 5	X 0 4 5
completed with errors	X No X8	X No X 9	X No X 9
after RESET	X D X A	X D X A	X D X 9

Tab. 4-11: Indicator word conditions for RECEIVE

Indicators for READ/WRITE-AKTIV

H1 condition	Prio 0/1	Prio 2	Prio 3/4
TCP/IP condition	Prio 1	Prio 2	Prio 3
after a restart		0 A 0 A	
after connection has been established		X 0 0 8	
after a trigger		X 0 X 2	
READ completed		X 0 4 4	
WRITE completed		X 0 2 4	
completed with errors		X No X 8	
after RESET		X D X A	

Tab. 4-12: Indicator word conditions for READ/WRITE-AKTIV

Here follow the indicators for SEND or RECEIVE when the handler module identifier is "NN" (no source / destination parameters)

H1 condition	Prio 0/1	Prio 2	Prio 3/4
TCP/IP condition	Prio 1	Prio 2	Prio 3
completed without errors	X 0 0 4	X 0 0 4	X 0 0 4

Tab. 4-13: Indicator word for SEND with an HTB-identifier "NN"

H1 condition	Prio 0/1	Prio 2	Prio 3/4
TCP/IP condition	Prio 1	Prio 2	Prio 3
completed without errors	X 0 0 4	X 0 0 5	X 0 0 5

Tab. 4-14: Indicator word for RECEIVE with an HTB-identifier "NN"

4.3.6.3 Parameter configuration error byte PAFE

The PAFE is set when the handler module detects a parameter error.

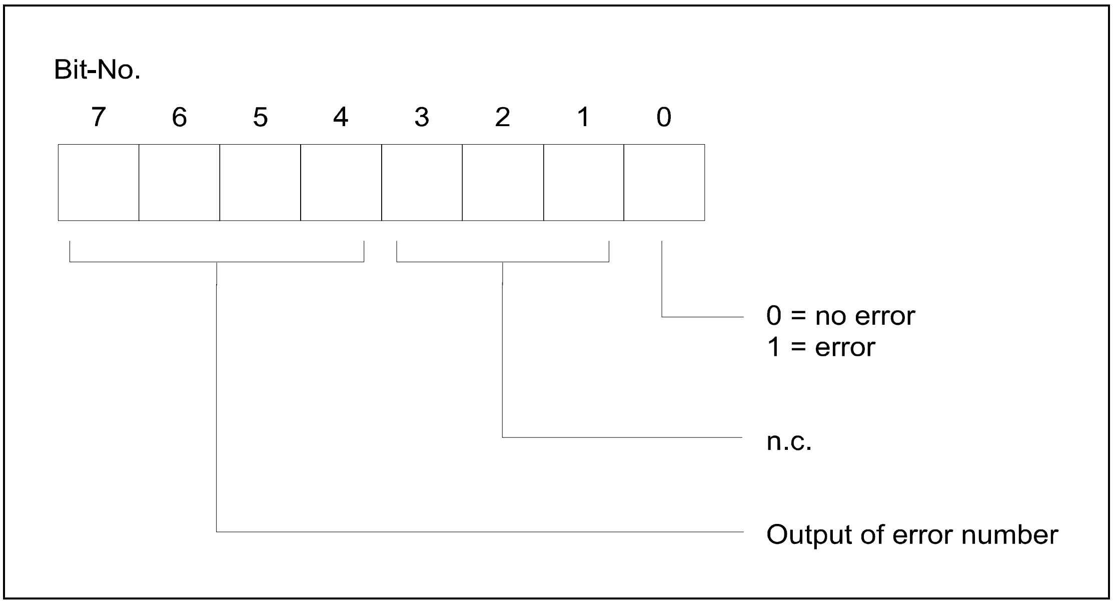


Fig. 4-8: Parameter error byte PAFE

Error number:

0	no error
1	incorrect ORG-format
2	area does not exist
3	area too small
4	QVZ-error
5	incorrect indicator word
6	no source / destination parameter for SEND / RECEIVE ALL
7	interface does not exist
8	interface error
9	interface overload
A	unused
B	illegal task number ANR
C	interface has not received an acknowledgement or it has not been released.
D	not allocated
E	not allocated
F	not allocated

5 Software

5 Software

5.1 General

This chapter consists of two parts. The first part contains a general description of the configuration software. The second part describes the application of the configuration software to the different protocols.

Different styles are used for the description to clarify certain aspects of the software elements. Here follows a listing of these elements:

Style	Example	Description
< >	**<ESC>**	A key on your keyboard that you should press. A + indicates that more than one key must be pressed simultaneously, e.g. **<Alt>**+**<F1>**.
[ ]	**[Help]**	Button in an interactive window.
bold	**File**	An item on the main menu
italics	*Save*	Items on sub-menus

This software package contains the NCS configuration program NCS. The letters NCS are used for **N**etwork-**C**ommunication-**S**ystem. The configuration program is an MS-DOS program. Its user interface is mouse driven and uses drop-down and pop-up windows. The program can also run in the DOS-shell of the respective operating system, i.e. Windows 3.1x and Windows 95 as well as the DOS-emulation of Windows NT.

This program can be used to configure H1 or TCP/IP system data, that is transport connections, datagrams, TCP-, UDP-connections and interfaces to the PLC. The configuration applies to SEND, RECEIVE, FETCH and WRITE jobs in the active and in the passive state.

You may use the following configuration options:

- Off-line

  You configure the connection and transfer the base file into an EPROM. You may use the VIPA EPROM programmer (Order-No.: VIPA MC5-KS85) or a PG7xx to program the EPROM.

- On-line

  - Serial
  - Serial H1
  - TCP/IP direct

The connection is identified by the job number. The job number is the link between the PLC and the H1 or the TCP/IP side. The CP module can handle a maximum of 24 connections (applications).

You determine the type of connection by means of the job.

5.2 Installing and starting the program

The following files are located on the disk supplied with the package:

`NCS.EXE` This is the executable configuration program.

`NCS.TXT` Except for the help text, this file contains all the documentation of the configuration program.

`NCS.HLP` This is the help system.

`NCS.BIB` A library containing programming algorithms for EPROMs.

`RUNNCS.BAT` Sample Siemens EPROM driver installation.

`TCPEVAL.EXE` Test program for TCP/IP connections.
(only for Windows 95 and Windows NT)

You may install this software under MS-DOS or under one of the graphic user interfaces.

The following sequence is required for an MS-DOS installation:

- Change to your hard disk
  ```
  C:\
  ```
- Create a directory for the programs on your hard disk.
  ```
  md NCS
  ```
- Copy all files from the diskette into this directory:
  ```
  COPY A:\*.*  c:\NCS
  ```
- Add the following statement to your path if you require access to the NCS-software from different subdirectories.
  ```
  PATH=...C:\NCS
  ```
- Install the TCP/IP drivers if you are using a CP1413 TCP/IP adapter.

To execute the software on a PG7xx you must also install the EPROM driver and the EPROM programmer. Chapter 5.5.3.2 describes the installation of the driver.

You can now run the program under MS-DOS.

At this stage it is possible to install the program for another user interface or another operating system. Please refer to the respective user manual for instructions on how to install an executable program.

To start the program under MS-DOS you can simply type NCS. The English version is started by entering NCS -e.

Other user interfaces or operating systems usually require a double-click on the icon to start the program. The parameters for starting the program (-e) should be entered into the field "Command line" which is located under "Properties". Once the program has started it displays a start window containing the version number of the software and other data.

Press a key on the keyboard and you will be presented with the main menu.

5.3 Overview of available functions

Overview of available functionsThe overview contains a summary of the available menu functions. This chapter also contains references to those chapters containing a description of the respective functions.

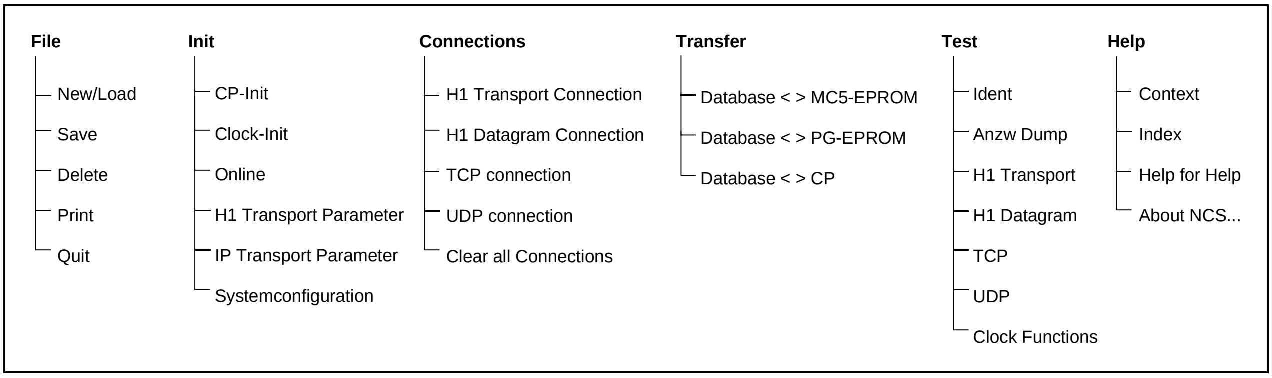


Fig. 5-1: The menu structure of the parameter configuration program

File

The menu item **File** provides access to all the file handling functions required for the manipulation of the base file. You may open, create, save and delete a base file. *Print* sends the base file to a printer file.

See chapter 5.5

Init

New CP143 H1 / TCP/IP modules must be initialized before they can be configured. The module must have a unique network identifier. This is assigned by means of *CP-Init*. The *clock init* command initializes the master clock.

The *Online* function is provided for the configuration of the parameters for online operations. These parameters are required for *Test* functions and for direct transfers between the database and the CP. Parameters applicable to H1 or TCP/IP are entered under *H1 Transport Parameter* or *IP Transport Parameter*.

You can optimize the performance of your system by means of the *Systemconfiguration* function.

See chapter 5.6

Connections

This menu item provides access to the form that you may use to configure the connection. Here you may configure H1 transport- and datagram-connections as well as TCP/IP TCP- and UDP-connections.

You may close all connections by means of a single menu command.

See chapter 5.8

Transfer

You may use the VIPA-EPROM programmer or the PG7xx-EPROM programmer to transfer your data into an EPROM.

It is also possible to configure the CP in on-line mode directly via the network or via the serial interface (remote configuration).

See chapter 5.5.3

Test

This menu item contains functions for displaying data from a specific target module. You must specify the target module in *Init*/*Online*. *Ident* returns the address and the serial number of the target module. *Anzw Dump* displays all indicator words from 0 through 223 in hexadecimal form.

See chapter 5.9

Help

Help text contains an abridged version of the explanations of the manual.

Context contains an overview of the menu items. Under *Index* you will find a table of contents of all the expressions and functions for which help entries are available. The sub-menu *Help on Help* explains the operation of the window and *Info about NCS...* shows the start window that contains the version number.

See chapter 5.4

5.4 Using on-line help

Whenever you require help you may press the **<F1>** key . You will also find help buttons at many points throughout the program. You may use the left mouse button to click on these buttons or you may access them by means of the **<Tab>**-key (tabulator) and the Enter-key. This will display the help text associated with the respective function.

You can press the **<F1>**-key while the help window is being displayed should you wish to learn how to use the help function.

The help text contains many cross-references to related subjects. These are displayed as highlighted text and may be accessed by means of the **<Tab>**-key. Press the Enter key (< ↵ >-key) to display the respective item.

If you are using a mouse you may also access the cross-reference by clicking on the text with the left mouse key.

The following buttons are available in the Help-window.

< ESC >-key

Quit from the help window and continue with the program.

< ↵ >-key

Displays the currently selected cross-reference. If you have not selected a cross-reference, the help window is closed and the program will continue.

< F1 >-key

Help on how to use the help window. This item displays a summary of the keys that may be used within the help window.

If the size of the help text exceeds the capacity of the help window, the visible portion may be moved by means of the cursor keys:

< ↑ >-key or < ↓ >-key

Line by line scrolling through the help text.

< PgUp >-key or < PgDn >key

Paging through the help text.

<Home>-key

Jump to the beginning of the help text.

<End>-key

Jump to the end of the help text.

<Tab>-key or <Shift>+<Tab>

Jump to the next or the previous reference. If you press the Enter key the help text that is associated with the selected reference is displayed.

<Alt>+<F1>-key

Displays the previous help text.

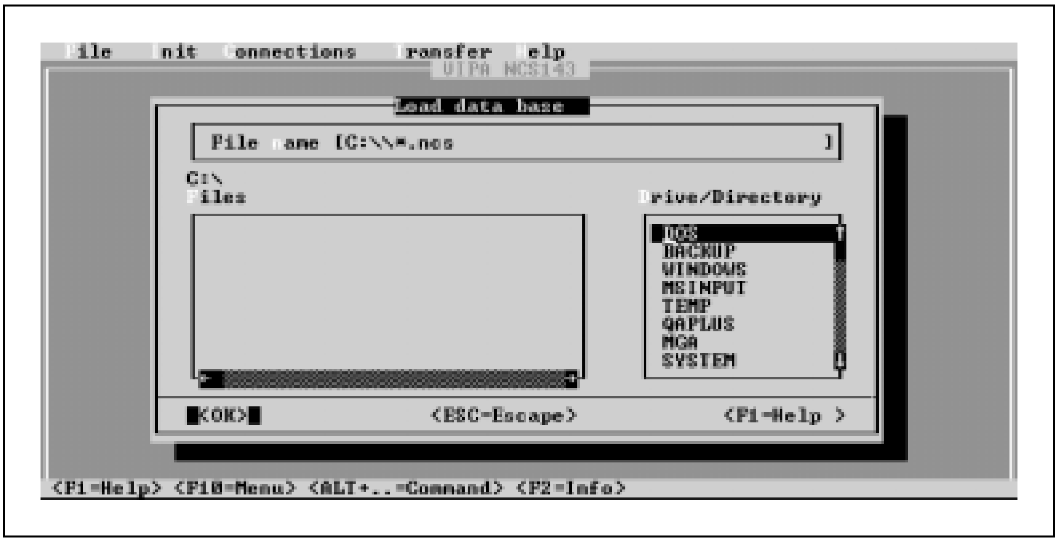
5.5 The base file

The base file, which is also called the data base file or connection parameter file, contains all the parameters that may be configured by means of the NCS configuration software. A saved base file may be transferred into an EPROM or directly to the CP. It is also possible to read an EPROM or a CP and to save the respective parameters into base file. The default extension of the base file is ".ncs" but you may change this as desired.

5.5.1 Loading, saving and deleting the base file

Base files may be created/loaded, saved and deleted by means of an interactive window. This interactive window may be accessed from the menu item **File**.

Interactive windows show subdirectory entries. You may also select another drive or select files from this window. In the following paragraph we will explain the operation of interactive windows by means of the load function. The interactive window may be different, depending on the selected function, but this does not affect the basic operation.



File ame [C:*.ncs]

C:\

Fig. 5-2: File selection window by means of the example for *Load/Create*

This interactive window contains all the elements that are also required for saving and deleting files:

- An input field for direct entry of a file name and path
The field contains the default path + *.ncs. This means that any file with the extension ".ncs" in the specified directory will be displayed. If you replace the asterisk (*) in the input field with a complete file name the respective file is loaded. If this file does not exist it will be created. Your entry is completed when you press the Enter key or when you click on **[OK]**.
- Display field containing the path selected in drive and directories.
- A field to select a drive and a directory.
Select the desired drive and directory by means of a mouse click and press the Enter key. The path in the display field is updated and all file names with an extension of ".ncs" are displayed. Click on “..” and press the Enter key to access the next higher level of the directory tree.
- A field to enter a file name for the selected directory.
Select the required file name and press the Enter key. The selected file will be loaded.

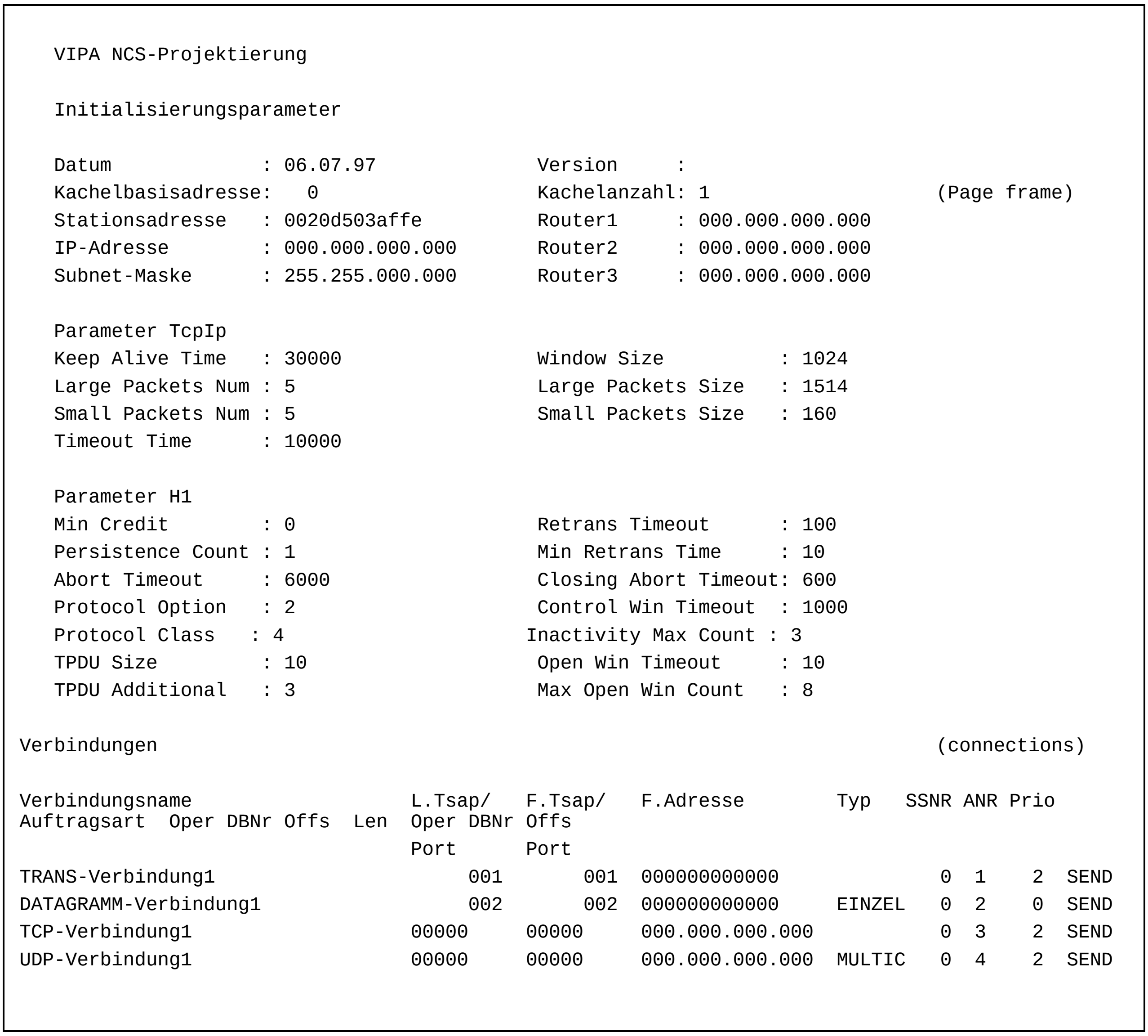
5.5.2 Printing the base file

You can send the contents of the base file to a text file by means of file/print. This file contains all initalization data and the configured connections.

This text file has the name of the base file with an extension prn. It is created in the current directory.

You can use a text processor to open this file and to edit its contents.

Here follows an example of the base file in text form:



```
   VIPA NCS-Projektierung

   Initialisierungsparameter

   Datum             : 06.07.97             Version     :
   Kachelbasisadresse:   0                  Kachelanzahl: 1                    (Page frame)
   Stationsadresse   : 0020d503affe         Router1     : 000.000.000.000
   IP-Adresse        : 000.000.000.000      Router2     : 000.000.000.000
   Subnet-Maske      : 255.255.000.000      Router3     : 000.000.000.000

   Parameter TcpIp
   Keep Alive Time   : 30000                Window Size          : 1024
   Large Packets Num : 5                    Large Packets Size   : 1514
   Small Packets Num : 5                    Small Packets Size   : 160
   Timeout Time      : 10000

   Parameter H1
   Min Credit        : 0                    Retrans Timeout      : 100
   Persistence Count : 1                    Min Retrans Time     : 10
   Abort Timeout     : 6000                 Closing Abort Timeout: 600
   Protocol Option   : 2                    Control Win Timeout  : 1000
   Protocol Class   : 4                    Inactivity Max Count : 3
   TPDU Size         : 10                   Open Win Timeout     : 10
   TPDU Additional   : 3                    Max Open Win Count   : 8

Verbindungen                                                                    (connections)

Verbindungsname                   L.Tsap/   F.Tsap/   F.Adresse        Typ   SSNR ANR Prio
Auftragsart  Oper DBNr Offs  Len  Oper DBNr Offs
                                  Port      Port
TRANS-Verbindung1                      001       001  000000000000               0  1    2  SEND
DATAGRAMM-Verbindung1                  002       002  000000000000     EINZEL    0  2    0  SEND
TCP-Verbindung1                   00000     00000     000.000.000.000            0  3    2  SEND
UDP-Verbindung1                   00000     00000     000.000.000.000  MULTIC    0  4    2  SEND
```

Fig. 5-3: Sample base file in text form

5.5.3 Transferring the base file

The NCS provides two options to transfer the base file into the CP:

- by means of an EPROM that you have created using the VIPA-EPROM programmer or the PG.
- Via the serial interface or the network in remote mode.

The menu item **Transfer** contains all the functions required for a data transfer between the PC, the EPROM programmer and the CP.

5.5.3.1 Data base ↔ MC5-EPROM

You can only use the EPROM functions if you have installed a VIPA EPROM programming adapter into your PC (Order No.: VIPA MC5-KS85). If not, you will receive an error message!

You may also use the PG to program EPROMs. In this case you can not use the following functions.

The adapter is compatible with the following EPROMs:

• 6ES5	376-0AA11	Memory size	16 KByte	NMOS
• 6ES5	376-0AA21	Memory size	32 KByte	NMOS
• 6ES5	376-0AA31	Memory size	64 KByte	NMOS
• 6ES5	376-1AA11	Memory size	16 KByte	CMOS
• 6ES5	376-1AA21	Memory size	32 KByte	CMOS
• 6ES5	376-1AA31	Memory size	64 KByte	CMOS

For the transfer you may select a base file from one of the following 4 sources:

[Dbase->EPROM]

You will be presented with a file selection window. Select the file for the transfer. Now you must select the type of EPROM from the module selection window. When you depress the Enter key the data is transferred to the EPROM.

[EPROM->Dbase]

You will be presented with a file selection window. Enter the file name to which the data read from the EPROM should be saved.

[NCS->EP]

You will be presented with an EPROM selection window. Select the type of EPROM you wish to program. When you depress the Enter key the data located in the NCS is programmed into the EPROM.

[EP->NCS]

You will be presented with an EPROM selection window. Select the type of EPROM you wish to program. When you depress the Enter key the data is read from the EPROM and transferred to the NCS configuration program.

5.5.3.2 Data base ↔ PG-EPROM

You can only use the EPROM functions in conjunction with a PG7xx and the respective EPROM drivers.

Otherwise you will receive an error message!

The diskette included in the package contains a program RUNNCS.BAT. This program loads the EPROM driver as a memory resident TSR and starts the configuration program NCS.EXE. When the configuration program is terminated the driver is removed from memory.

Please start RUNNCS.BAT. If you are working with a PG710 through to a PG740 the driver should load without errors and NCS should start. In this case you may omit the chapter "Modification of RUNNCS.BAT".

If you encounter errors the RUNNCS.BAT file must be modified.

Modification of RUNNCS.BAT for a PG7xx

Data base ↔ EPROM can only be executed on a PG7xx if the EPROM driver for the EPROM programmer was loaded into the PG7xx.

The EPROM driver is delivered with the PG7xx.

The driver consists of a memory resident TSR that must be loaded by means of a LoadHigh command.

The driver is specific to the PG and it is already located in the PG. The driver is called:

PG710, PG720, PG730 and PG740:	`SIM_730.EXE`
PG750:	`TRNOSIMA.EXE`

You can find the driver by locating the file S5.BAT. This file contains the LoadHigh command for the driver, e.g. for the PG720:

```
LH   C:\STEP5\S5_ST\SIM_730.EXE    IV
```

Make a note of the respective path and replace the path in RUNNCS.BAT by this new path.

Please be careful!

The file RUNNCS.BAT contains two entries for this path (install and uninstall), i.e. it must specified twice.

Execute RUNNCS. The EPROM driver will be installed and the NCS program loaded. Now you may use EPROMs on your PG7xx for configuration purposes.

Your PG is compatible with the following EPROMs:

• 6ES5	376-0AA11	Memory size	16 KByte	NMOS
• 6ES5	376-0AA21	Memory size	32 KByte	NMOS
• 6ES5	376-0AA31	Memory size	64 KByte	NMOS
• 6ES5	376-1AA11	Memory size	16 KByte	CMOS
• 6ES5	376-1AA21	Memory size	32 KByte	CMOS
• 6ES5	376-1AA31	Memory size	64 KByte	CMOS

Data base ↔ EPROM

When you select *Data base ↔ EPROM* from **Transfer** the interactive window "Data base<>EPROM" will be displayed.

If the program responds with an error message "Eprom-driver not installed!" the driver could not be installed.

Check the file RUNNCS.BAT for the correct path statement.

You may select a base file from one of the following 4 source / destination combinations for the transfer:

[Dbase->EP]

You are presented with a file selection window. Select the file for the transfer. Now you must select the type of EPROM from the module selection window. When you depress the Enter key the data is programmed into the EPROM.

[EP->Dbase]

You will be presented with a file selection window. Enter the file name to which the data read from the EPROM should be saved.

[NCS->EP]

You will be presented with an EPROM selection window. Select the type of EPROM you wish to program. When you depress the Enter key the data located in the NCS is programmed into the EPROM.

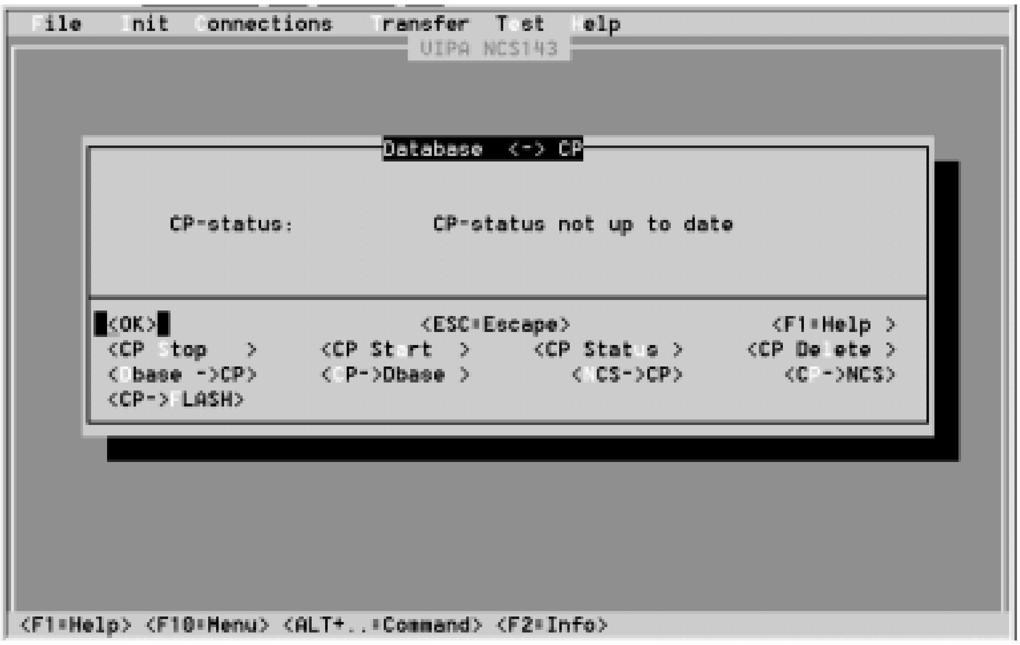
[EP->NCS]

You will be presented with an EPROM selection window. Select the type of EPROM you wish to program. When you depress the Enter key the data is read from the EPROM and transferred to the NCS configuration program on your PC.

5.5.3.3 Data base ↔ CP

The remote functions provide the different functions required for remotely controlling the CP module (start, stop and delete), to request the status from the CP and to transfer data base files into the CP.

Please make sure that you have specified the online transfer parameters in the section *Online* (see chapter 5.6.2).



CP-status: CP-status not up to date

Fig. 5-4: Window for remote configuration

Remote control of the CP

Switch the RUN/STOP switch on your CP module to RUN. The status display is does not change automatically when a change of status occurs.

[Stop the CP]

The status of the CP module is changed from RUN to STOP.

[Start the CP]

The status of the CP module is changed from STOP to RUN.

[Delete the CP]

The base file in the CP is deleted. You must first execute a STOP, otherwise the delete command will be ignored.

CP-status

This output field contains the CP-status. This defaults to the "CP-status not up to date".

Click on **[Status]** if you wish to display the current status. The displayed status is brought up to date.

Transferring the base file

The following points must be noted when base data is transferred between a CP and a PC:

- Communication parameters must have been set in *Online*.
- A transfer is only possible when the PLC is stopped (STOP status). Please ensure that your PLC has been STOPped.
- You can only overwrite the data on the CP if the CP does not contain an EPROM.

For the transfer you may select a base file from one of the following 4 source / destination combinations:

[Dbase→CP]

You are presented with a file selection window. Select the base file for the transfer. This will be transferred to the CP.

[CP→Dbase]

The data base is located in the CP from where it is transferred to a file. Enter a file name into the file selection window.

[NCS→CP]

All connections that have been entered into the connection masks of NCS are transferred directly into the CP.

[CP→NCS]

The data base is located on the CP from where it is transferred directly into the connection masks of the NCS.

[CP→FLASH]

The data base is located on the CP from where it is transferred directly into a flash-EPROM.

This function is only available from revision level 20 and firmware level V116.

5.6 Initialization

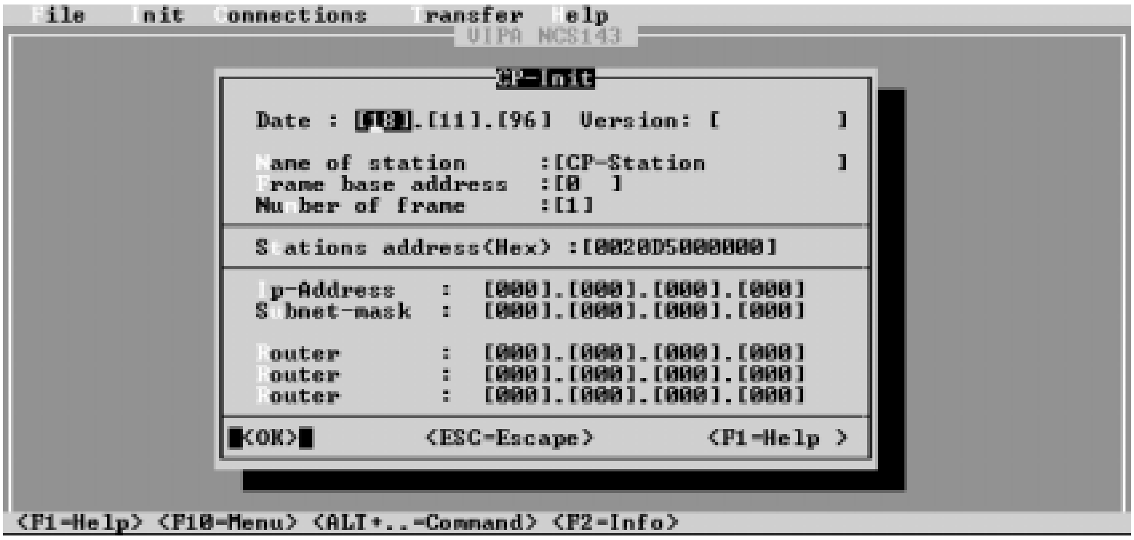
Every now configuration requires initialization. It is necessary to initialize the identification parameters as well as H1 and TCP/IP-parameters of the CP143 H1 / TCP/IP-module.

The initialization is not required if you have already loaded a base file. You can find the initialization functions under menu item **Init**.

You enter online transfer parameters in the section *Online*.

5.6.1 CP init

The entry *CP-Init* under **Init** places you in an interactive window that provides access to the CP-base initialization:


File Init Connections Transfer Help
VIPA NCS143
CP-Init
<OK>
<ESC-Escape>
<F1-Help >
<F1-Help> <F10-Menu> <ALT+..=Command> <F2-Info>

```
Date : [18].[11].[96]  Version: [          ]

Name of station      :[CP-Station          ]
Frame base address   :[0   ]
Number of frame      :[1]

Stations address(Hex) :[0020D5000000]

Ip-Address    :  [000].[000].[000].[000]
Subnet-mask   :  [000].[000].[000].[000]

Router        :  [000].[000].[000].[000]
Router        :  [000].[000].[000].[000]
Router        :  [000].[000].[000].[000]
```

Fig. 5-5: CP-base initialization (initial initialization)

Here you must configure the addresses and other identification parameters of the CP.

Before entering a connection, you must also enter a frame base address into this interactive mask. The entered data is transferred when you depress the Enter key.

Parameters of the CP-initialization

Date, Version

The entries for date and version are for documentation purposes only. You can use these entries to check at what time a configuration was performed. The version number will always contain the number associated with the file that is currently loaded.

Name of station

You may specify a name for your base station. The name may consist of a maximum of 19 characters. Station names are used to simplify the configuration of connections. The station name is important where on-line configurations are concerned.

Frame base address

The frame base address Frame base address determines the origin of the frames that are inserted into the memory area of the controller. The PLC program normally operates on the base frame. Any other frames may be used by other controller or systems.

The frame base address must be divisible by 4 and it is located between 0 and 255. This means that you may enter the values 0, 4, 8, 12, ... 252.

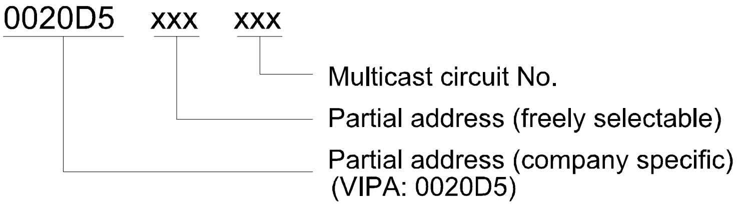
The frame base addresses on a single controller may not overlap.

Number of frames

Specifies the number of frames that the CP may use to communicate with the PLC. The CP always occupies 4 frames (e.g. 4, 5, 6, 7).

Station address

Here you must specify the ethernet address (station address). The structure of the ethernet address is as follows:


0020D5 xxx xxx
Multicast circuit No.
Partial address (freely selectable)
Partial address (company specific)
(VIPA: 0020D5)

IP-address

The IP-address is a 32-bit-address which must be unique within the network. The address consists of 4 numbers that are separated by a full stop.

Range: 000.000.000.000 to 255.255.255.255

You can obtain the IP-address from your system administrator.

Subnet-mask

The IP-subnet-mask is a 32-bit filter mask for IP-addresses. This address is structured in the same way as the IP-address. It also has the same range.

Subnets allow better control of the addressing area. Subnet-masks should be used where it is necessary to address a network on both sides of a router although only one network number is available.

Router

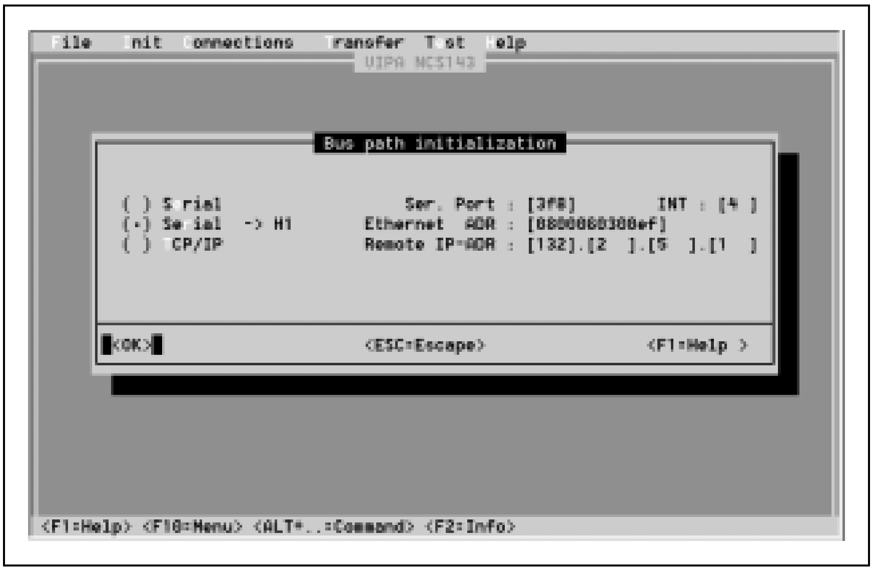
A Router (formerly gateway) can provide a link between different network topologies, e.g. ethernet to token ring. It may also be used to connect IP-networks with different network numbers. A router has an IP-address for every network. This IP-address provides the logical connection between the networks. 2 addresses exist where 2 networks are connected via a router. If you wish to avoid the use of router-IP-addresses when a connection is established you may enter 0.0.0.0 for the router.

5.6.2 Online

Online parameters must be specifies when functions are to be used that access the module directly via the network or via the serial interface. Online functions are only available from firmware level V110 and an NCS V2.1.

The function *database <–> CP* under **Transfer** and die functions under **Test** make use of online parameters.

The online dialog window provides access to the following settings:



```
( ) Serial              Ser. Port : [3F8]         INT : [4 ]
(•) Serial   -> H1      Ethernet  ADR : [0800060300ef]
( ) TCP/IP              Remote IP-ADR : [132].[2  ].[5  ].[1  ]
```

Fig. 5-6: Parameters for online connections

serial

Data is transferred via the serial COM port of your PC to the PG interface of the CP143 H1 / TCP/IP. You must specify the port no. and the interrupt.

The following are standard settings:

COM1	=	Port	3F8	with INT 4	
COM2	=	Port	2F8	with INT 3	(default)
COM1	=	Port	3E8	with INT 4	

serial -> H1

Data is transferred via the serial COM port to the PG interface of an H1 module and from there via the H1 network to a target module identified by means of the respective ethernet address.

This mode requires specification of the ethernet address of the target module and the serial parameters (port no. and INT).

TCP/IP

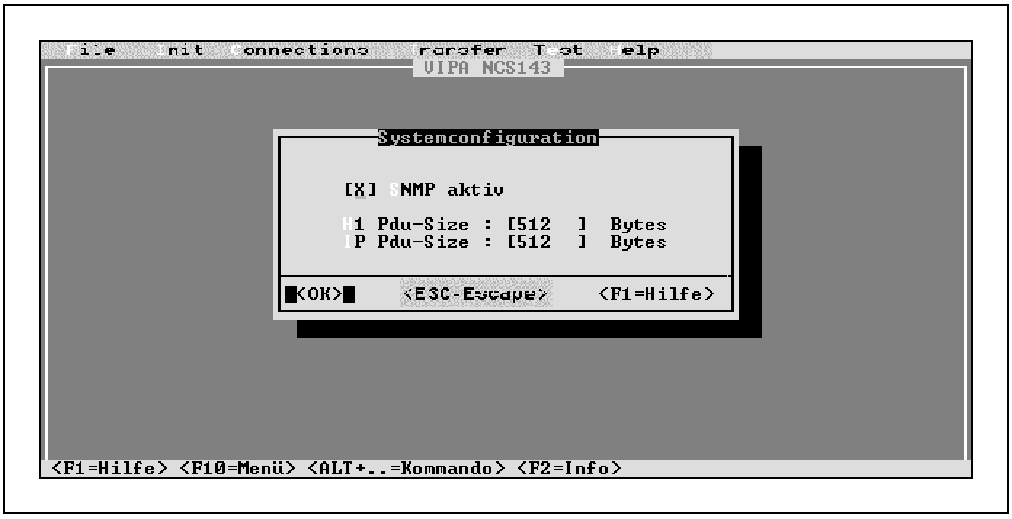
Data is transferred directly via the network to a TCP/IP module. The target module is specified by means of an IP address.

Please note!

When using TCP/IP you must enter an ethernet ADR 000000000000h.

5.6.3 System configuration

When you select system configuration a dialog window containing the general settings for optimizing the system performance is displayed.


File Init Connections Transfer Test Help
VIPA NCS143
Systemconfiguration
<OK>
<ESC-Escape>
<F1=Hilfe>
<F1=Hilfe> <F10=Menü> <ALT+..=Kommando> <F2=Info>

```
[X] SNMP aktiv

H1 Pdu-Size : [512   ]  Bytes
IP Pdu-Size : [512   ]  Bytes
```

Fig. 5-7: System configuration parameters

SNMP active

When this item is selected the **S**imple **N**etwork **M**anagement **P**rotocol will be used. SNMP is normally used for UDP connections.

Default setting: ticked

H1 PDU-Size

Maximum size for H1 data blocks (in byte) that may be transferred via the network level. PDU is the abbreviation for **P**rotocol **D**ata **U**nit and refers to a data block.

Range of values: 0...65535, default: 512

IP PDU-Size

Maximum size for IP data blocks (in byte) that may be transferred via the network level. PDU is the abbreviation for **P**rotocol **D**ata **U**nit and refers to a data block.

Range of values: 0...65535, default: 512

5.7 Real-time clock functions

5.7.1 General

Many practical situations require that the operations are synchronized and referenced to real time. The real-time clock functions of the CP143 module may be used to synchronize up to 256 modules with an accuracy of 10ms. The synchronization is maintained even if the time standard transmitter should fail.

The real-time clock functions provide the following options

- The time of day is maintained by the system and a battery sustains it should the power supply fall away.
- The time of the various CPs can be synchronized by means of synchronization messages. In this way all CPs on the same network use the same time.
- Any CP can be designated as master clock to issue time synchronization messages.

5.7.2 Graphic representation of the real-time clock functions

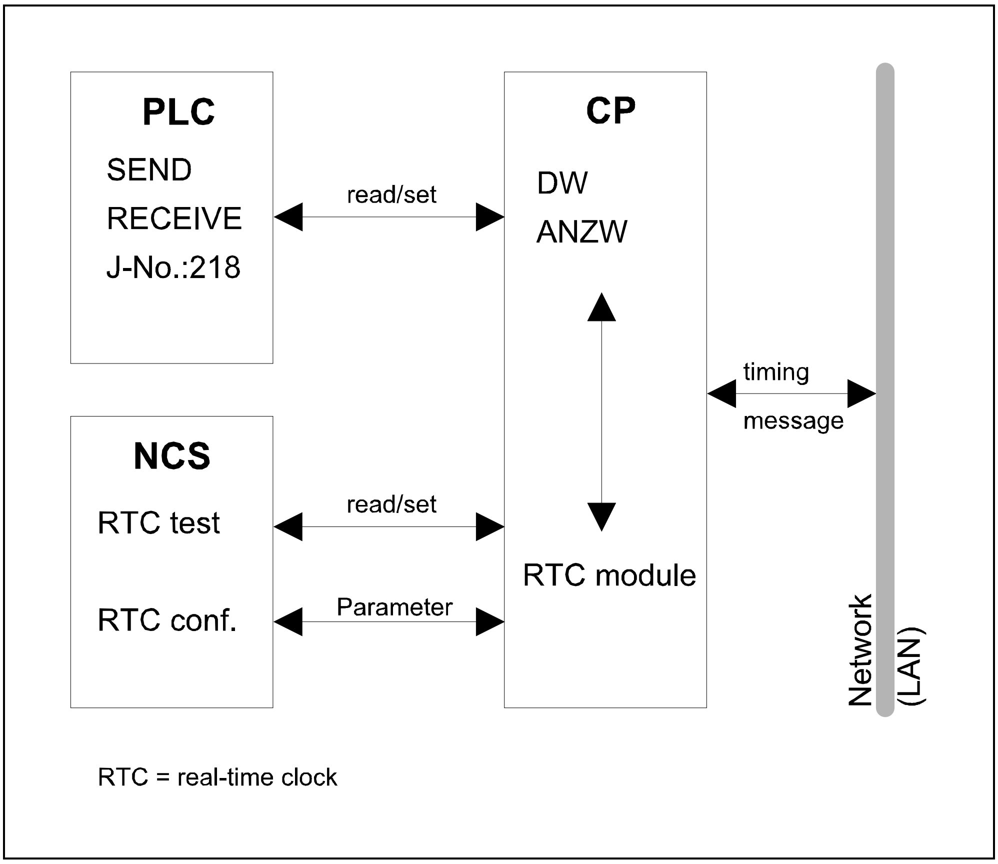


Fig. 5-8: Graphic representation of the real-time clock functions

5.7.3 Construction of a synchronized system:

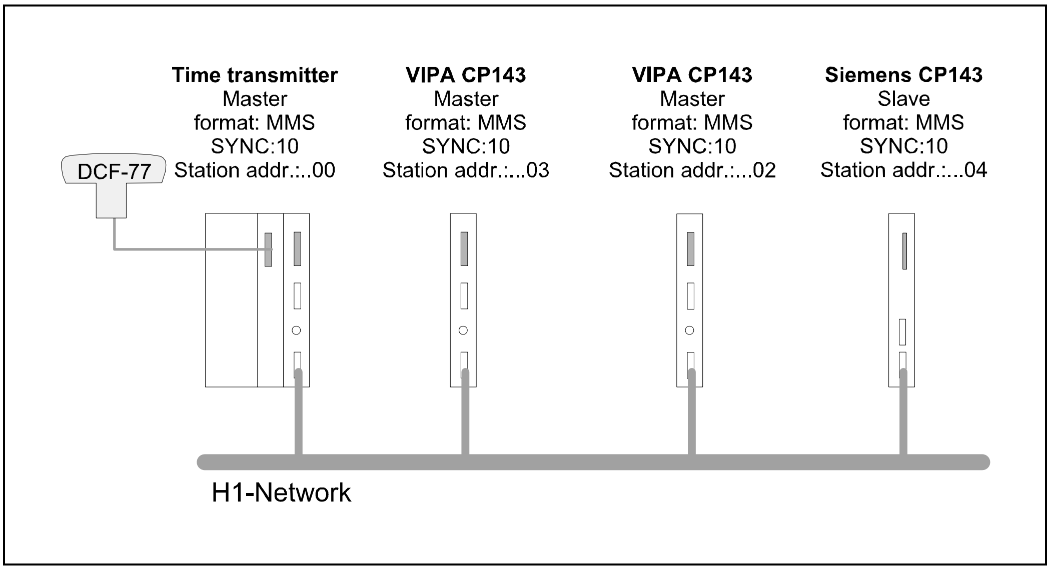


Fig. 5-9: Construction of a synchronized system

The DCF-77 module receives highly accurate time information and transfers this to the PLC. The PLC generates a timing message that is transferred via the LAN. This message synchronizes all CP143 modules connected to the H1-network. The real-time ICs of the different CPs maintain the time until the next timing message is received.

5.7.4 Mechanism for assigning the master clock

No other CP can become master clock while the time transmission supplies the timing message in accordance with the SYNC clock.

When timing messages are not received for more than the SYNC cycle time every CP will attempt to become the master clock in accordance with the priority of its delay time. The priority of the individual CPs is determined by means of the last byte of the ethernet address. This byte determines the delay time before the station will attempt to become the master clock. Higher settings for this value are equivalent to higher delay times before the station will make an attempt to acquire the master function.

The dynamic master-CP will return to receive mode (slave) if timing messages with a higher priority are queued or if the DCF-77 becomes active again.

Only modules where the *clock init* setting was master clock can become a master clock.

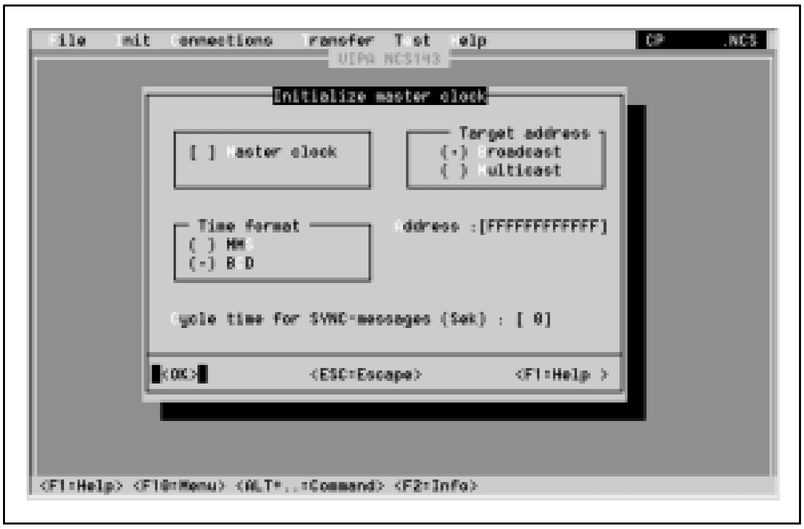
! *The DCF-77 (time transmitter) must always be assigned to the highest priority, i.e. the last byte of the ethernet address (byte 6) should be 00h.*

5.7.5 Master clock initialization

Every CP may assume the role of master provided that it has been configured as a master.

You may mix and match VIPA CPs and Siemens CPs. Please ensure that you use the MMS-timing format as Siemens supports only this format.

From **Init** you can use *Uhr-Init* to access the following initialization window:



```
[ ] Master clock

Target address
(•) Broadcast
( ) Multicast

Time format
( ) MMS
(•) BCD

Address :[FFFFFFFFFFFF]

Cycle time for SYNC-messages (Sek) : [ 0]
```

Fig. 5-10: Master clock initialization

Parameters for master clock initialization

Master clock

When a high level master clock fails the module configured by means of this parameter may assume the role of dynamic master. However, the CP can only become the dynamic master if it has the highest priority and if you have ticked master clock.

master clock:	you must supply the target address (broadcast or multicast) and, in the case of multicast, the target ethernet address and the mandatory time-format and cycle time.
Slave:	enter the mandatory time-format and the cycle time

Target address

Broadcast
Timing messages are sent to every station on the LAN. The address is set automatically to FFFFFFFFFFFFh. This can not be changed.
Multicast
Timing messages are only sent to stations on the LAN that are a member of the multicast circuit specified by means of the address.

Address

This address is FFFFFFFFFFFFh when broadcast has been selected. For multicast you must specify a single address or a multicast circuit by means of the address.

Cycle time for SYNC messages

Here you may determine the cycle time for the transmission of synchronization messages in the range of 1...60 seconds.

The synchronization message is transferred in MMS- or in BCD format.

Please note: the format for the time and the cycle time must be configured in the same way throughout your system!

The last byte of the ethernet address of each CP must be different!

Time format

MMS format

This is the only format allowed for synchronizing foreign modules. The MMS formatted timing message is formatted as follows:

----	tttt	Byte 0	Time in ms, starting from 0:00 hours
tttt	tttt	Byte 1	
tttt	tttt	Byte 2	
tttt	tttt	Byte 3	
dddd	dddd	Byte 4	Date in days, starting from 1.1.1984
dddd	dddd	Byte 5	
vkkk	kk--	Byte 6	Sign v: 0=+, 1=-; time difference steps of in 0,5 hrs.
----	--e-	Byte 7	Flag e: 0=time from DCF; 1=from dynamic master

BCD-Format

The BCD format is a plain and readable alternative. If you are only employing VIPA CP143 modules the CPs can be configured to accept BCD formatted messages.

1/10s	1/100s	Byte 0	Fractional seconds
10s	1s	Byte 1	Seconds
10m	1m	Byte 2	Minutes
ab10h	1h	Byte 3	Mode a: 0=12h, 1=24h; b: 0=AM, 1=PM; hours
wwww	----	Byte 4	0=Monday ... 6=Sunday
10T	1T	Byte 5	Day of Month
10M	1M	Byte 6	Month
10J	1J	Byte 7	Year
v--k	kkkk	Byte 8	Sign v: 0=+, 1=-; time difference in steps of 0,5 hours.
----	1/1000s	Byte 9	1/1000s
ssss	--em	Byte 10	Clock status s: (=clock status in ANZW); Flag e: 0=time from DCF; 1=time from dynamic master Module validity m: 0=valid, 1=not valid

The module validity indicates that the current master CP was configured successfully and that it does not use default parameters.

5.7.6 Important considerations when using real-time clock functions

Here follow the most important considerations that you should take into account when using the real-time clock functions:

- use consistent cycle times for SYNC messages
- use a consistent format for time
- the format for time must be MMS when you are employing CPs from VIPA as well as Siemens.
- byte 6 of the ethernet address of every CP must be different, otherwise it could happen that two CPs would attempt to become master simultaneously and therefore block the System.
- byte 6 of the ethernet address of the time transmitter must be 00h to guarantee that the time transmitter is always the master.

5.7.7 Setting and interrogating the clock from the PLC

CP communications with the PLC is interrupt controlled and occurs via the Dual Port RAM. Job number 218 is set aside for time-related processing in the PLC.

A SEND using this job number initiates a write operation and a RECEIVE initiates a read operation for the time of the CP.

The respective time value is transferred by means of data words.

The status message is returned in an indicator word.

Configuration of data words for transferring data from the clock (6 DW):

10s	1s	1/10s	1/100s	DW0
xx 10h	1h	10m	1m	DW1
10D	1D	day of week	----	DW2
10Y	1Y	10M	1M	DW3
----	1/1000s	v—k	kkkk	DW4
10Jh	1Jh	ssss	--em	DW5
15...12	11...8	7...4	3...0	Bit

Indicator word (ANZW) configuration for the transfer of the status:

Byte 1		Byte 0	
0 0 0 0	clock-status	0 0 0 0	x x B1 B0

B1	B0	Clock may be set	Clock can be read
0	0	yes	no
0	1	yes	yes
1	0	no	no
1	1	no	yes

Clock status, status message in the indicator word

Setting the clock

Hex	Description
0h	completed without errors
1h	protocol error, bad time

No errors when reading clock

Hex	Description	Explanation
6h	clock master	CP is master clock and is currently active
7h	clock slave synchronous	CP is slave clock (the time transmitter is active)
8h	clock slave invalid	CP has an invalid clock module
9h	clock slave asynchronous	CP not receiving timing messages
Ah	clock slave/master	CP is slave clock; prepare for master function or vice versa
Ch	clock slave dynamic synchronous	CP is slave clock and is receiving timing message from the higher level master clock CP

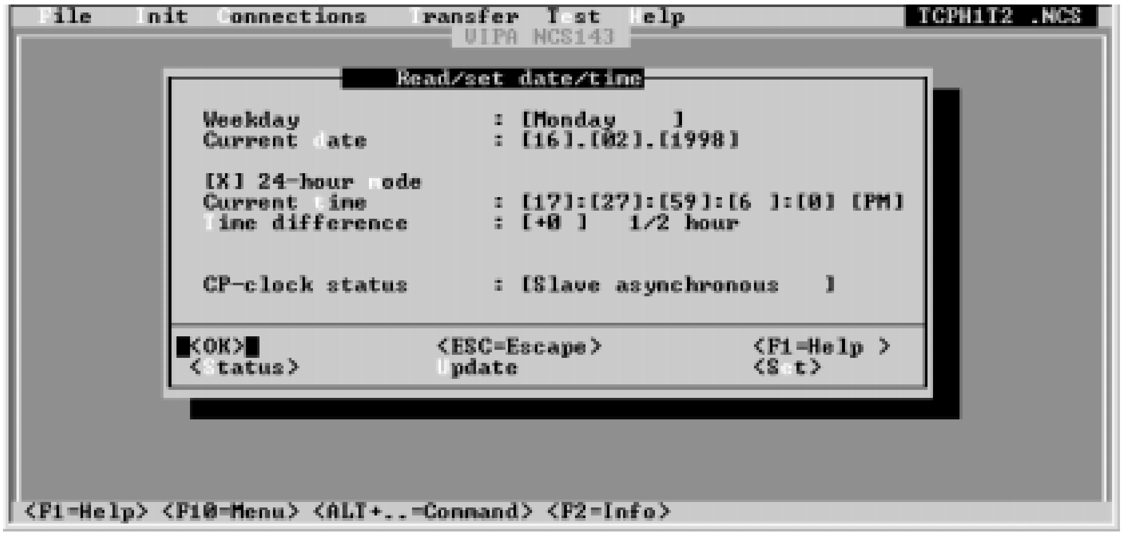
5.7.8 Test functions for the clock

The test functions for the clock are for use in online mode. You must specify parameters for online transfers using the *Online* function located under **Init** (see chapter 5.6.2).

The test functions for the clock are available via the menu item **Test.**

The mask for clock test functions of the respective CP143 module provides options for setting or reading the clock.

When you select this function a read message for is issued to the clock of the currently selected CP. This CP returns a timing message containing the current time. The returned values are displayed by means of the following window:



```
 File   Init  Connections   Transfer  Test  Help                TCPH1T2 .NCS
                                  VIPA NCS143

                 Read/set date/time

      Weekday                : [Monday     ]
      Current date           : [16].[02].[1998]

      [X] 24-hour mode
      Current time           : [17]:[27]:[59]:[6  ]:[0] [PM]
      Time difference        : [+0 ]   1/2 hour

      CP-clock status        : [Slave asynchronous    ]

    <OK>                 <ESC=Escape>            <F1=Help >
     <Status>             Update                 <Set>

 <F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 5-11: Dialog window for the test functions for the clock

Parameters

day of week

The program calculates the day of the week from the date. This value can not be modified.

current date

Current date

You can modify this value and it is transferred to the CP when you select "set".

24 hour mode

You can select between 24- or 12 hour mode.

You can modify this value and it is transferred to the CP when you select "set".

current time

The current time. You can modify this value and it is transferred to the CP when you select "set".

time difference

You may enter the time difference in steps of 0,5 hours. This option is useful for daylight saving applications (summer and winter time).

CP clock status

Displays the status of the clock in readable form. The following messages are possible:

Setting the clock

Message	Explanation
no error occurred	The clock on the CP was set as required
protocol error, bad time	a transfer error occurred

Reading the clock

Message	Explanation
clock master	The CP is the active master clock
clock slave synchronous	The CP is the slave clock (the time transmitter is active)
clock slave invalid	The CP has an invalid clock module
clock slave asyncronous	The CP is not receiving timing messages
clock slave/master	The CP is the slave clock; prepare for master or slave function
clock slave dynamic synchronous	The CP is the slave clock and receives timing messages from a higher level master clock CP

Buttons

status

Displays the timing data of the CP in online mode and updates the dialog window continuously until you terminate the operation by means of the <ESC> key.

update

Displays the timing data of the CP in online mode and updates the dialog window.

set

The data entered into the dialog window is transferred to the CP when [set] is pressed. When the transfer is completed without errors you receive the message " completed without error" in the CP clock status field.

> ⚠ *You can only set a module that is a "master clock".*
>
> *The master clock is configured under* **Init** *by means of the function* clock init *(see chapter 5.7.5).*

5.8 Configuration of connections

New configurations require that an initialization is performed. (see chapter 5.6). The initialization determines the station that must be configured.

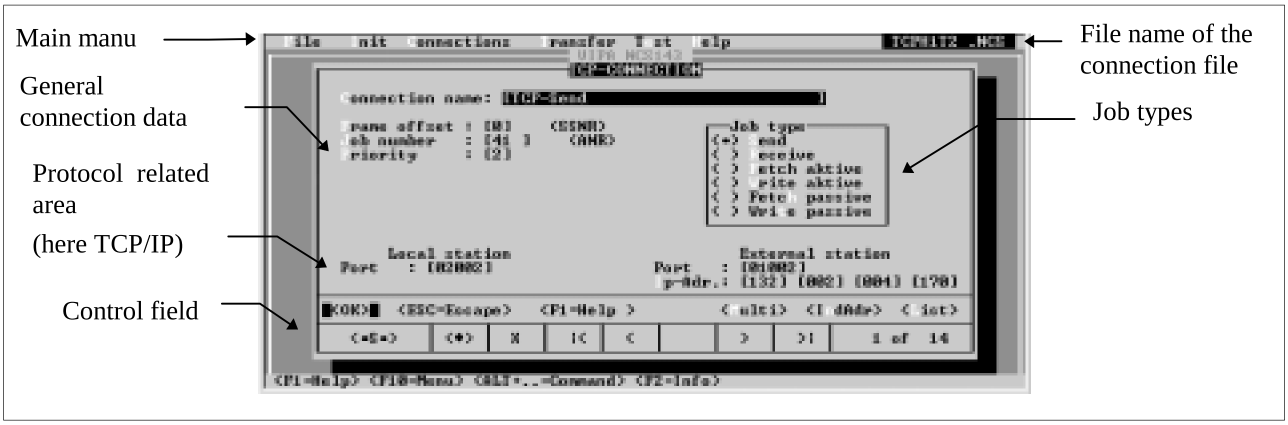
The CP143 H1 / TCP/IP module is compatible with four types of communication connection:

- H1-transport connections
- H1-datagram connections
- TCP-connections
- UDP-connections

The configuration of H1-connections and TCP/IP-connections differs in the specification of the ethernet address and TSAPs (for H1) or the IP-address and ports (for TCP/IP).

5.8.1 Structure of the configuration windows

The structure of the configuration windows that you can access under **Connections** is nearly identical:



```
Connection name: [TCP-Send                    ]

Frame offset : [0]    <SSNR>          Job type
Job number   : [41 ]   <ANR>          (•) Send
Priority     : [2]                    ( ) Receive
                                      ( ) Fetch aktive
                                      ( ) Write aktive
                                      ( ) Fetch passive
                                      ( ) Write passive

        Local station                        External station
Port    : [02002]                     Port    : [01002]
                                      Ip-Adr.: [132] [002] [004] [170]
```

Fig. 5-12: Structure of configuration windows

- General connection data

  Here you may enter general data for connections that are used with both, H1 and TCP/IP. General connection data consists of:
  Connection name, frame offset, job number and if required, type of job

- Type of jobs

  this is where you configure the job types like SEND, RECEIVE, FETCH and WRITE

- Protocol related area

  Here you must enter the protocol related data, i.e. the ethernet address and the TSAP for H1 and the IP-address and the port for TCP/IP.

- Control field

  The control field is used to navigate within the interactive window. The program provides the control field shown in Fig. 5-13 to manipulate the configuration window. You may use the mouse or the cursor keys to page through your connections.

5.8.2 The control field of the configuration window

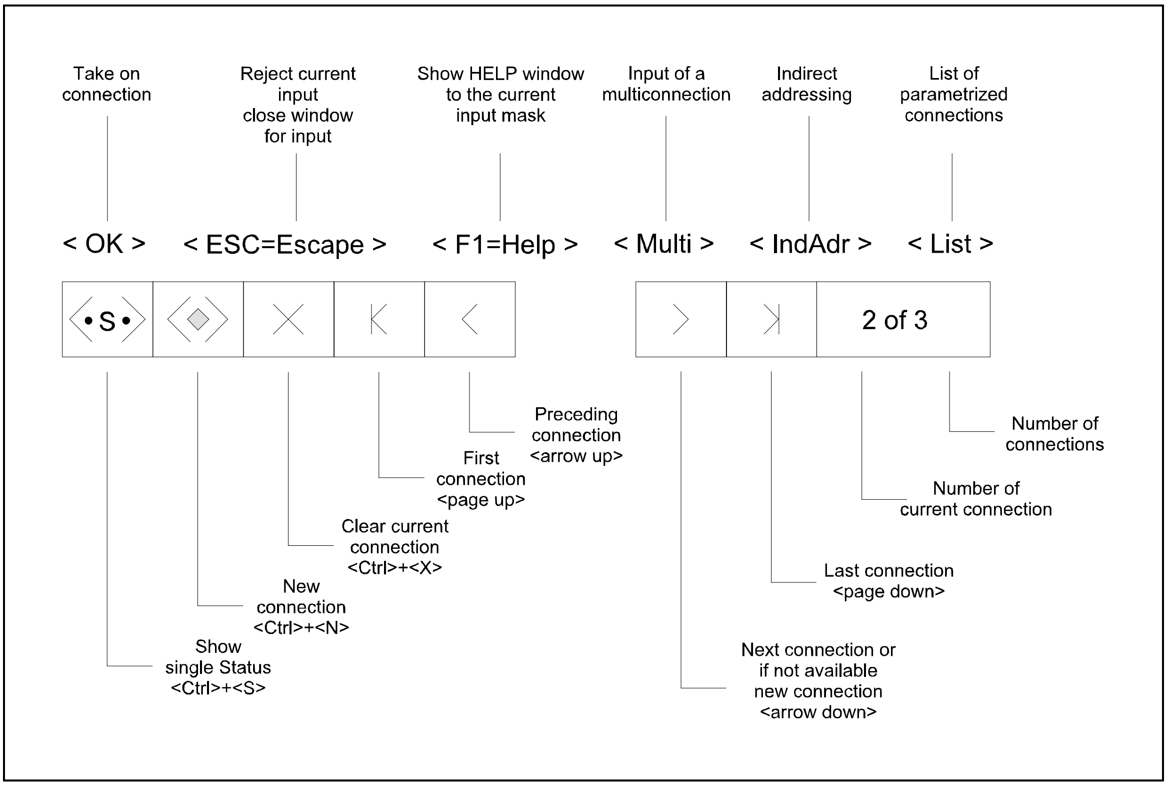

Take on connection
Reject current input close window for input
Show HELP window to the current input mask
Input of a multiconnection
Indirect addressing
List of parametrized connections
< OK >
< ESC=Escape >
< F1=Help >
< Multi >
< IndAdr >
< List >
• S •
2 of 3
Preceding connection <arrow up>
First connection <page up>
Clear current connection <Ctrl>+<X>
New connection <Ctrl>+<N>
Show single Status <Ctrl>+<S>
Number of connections
Number of current connection
Last connection <page down>
Next connection or if not available new connection <arrow down>

Fig. 5-13: Control field of a parameter configuration window

[OK]

After confirmation the new connection is accepted. You may also press the Enter key instead of [OK].

[ESC=Escape]

The interactive window is closed and the entries are discarded. Connections that have previously been configured are not discarded however.

[F1=Help]

Displays the help window for the current interactive window.

[Multi]

You may use this button to configure multi-connections. Multi-connections are dual connections that share the same TSAP or the same port with another job number and a selectable frame. The opposite job types are configured automatically.

[IndAdr]

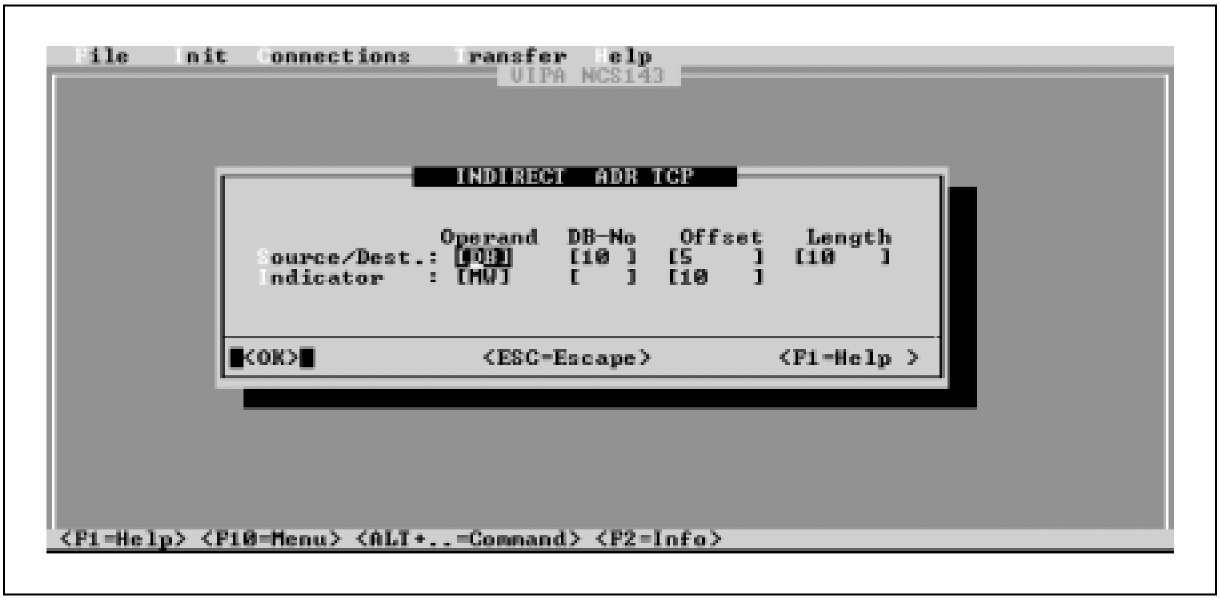
This button provides access to the input window for indirect addressing. Here you can enter a static area containing specifications with respect to the source and the destination for the data as well as the indicator word.

[List]

Lists any previously configured connections for the respective protocol. This provides easy access to any of your connections for checking and revising. You may also edit configured multi-connections.

5.8.3 Indirect addressing

The configuration windows for H1-transport or TCP-connections allow access to the input window for indirect addressing via the **< IndAdr >** button. Indirect addressing can only be used with the job types FETCH and WRITE.



File Init Connections Transfer Help
VIPA NCS143
INDIRECT ADR TCP
<OK>
<ESC=Escape>
<F1=Help >
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

```
               Operand  DB-No   Offset    Length
Source/Dest.:  [DB]     [10 ]   [5     ]  [10    ]
Indicator   :  [MW]     [   ]   [10    ]
```

Fig. 5-14: Indirect addressing for TCP

Here you may specify a static area where the specifications for source and destination data as well as the indicator word may be entered.

Source/destination

Contains the source or destination for FETCH/WRITE in the local PLC. You may supply the following entries:

Operand	DB-No.	Offset	Length	Type
DB	1..255	0..2047	1..2048	Data module
MB		0..255	1..256	Flag byte
EB		0..127	1..128	Input byte
AB		0..127	1..128	Output byte
PB		0..255	1..256	Peripheral byte
ZB		0..255	1..256	Counter cells
TB		0..255	1..256	Timer cells
BS		0..511	1..512	System data
AS		0..32767	1..32768	absolute addresses
DX	1..255	0..2047	1..2048	extended data module
DE	1..255	0..2047	1..2048	external data module
QB		0..255	1..256	extended peripheral byte

Tab. 5-1: Possible entries for source/destination with indirect addressing

Indicator word

Specifies a data word on the PLC that contains the specifications for FETCH and WRITE

Operand	DB-No.	Address	Type
DB	1..255	0..2040	Data module
DX	1..255	0..2040	extended data module
MW		0..252	Flag word

Tab. 5-2: Data word for the indicator word for indirect addressing

Notes on the structure of the data modules for indirect addressing are contained in the description of the standard handler modules provided by the Siemens manuals.

Example

Indirect addressing using the SEND module

Call to the SEND-FBs

```
A      DB X          ; Open the data module x
SPA    FB244         ; Call the SEND module
NAME : SEND
SSNR : KY 255,1      ; 255 is the key for indirect addressing
                     ; 1 is interpreted as a pointer to DW 1
    A-NR :  KY 0,0   ; irrelevant
    ANZW :  MW 0     ; irrelevant
       :
       :
```

Data module

```
DB X

DW 1   KY 0,1
DW 2   KY 0,31
DW 3   KC MW
DW 4   KY 0,200
```

The left byte of parameter SSNR is used as to decide between direct and indirect addressing. If the left byte > 0 the addressing is of the indirect type.

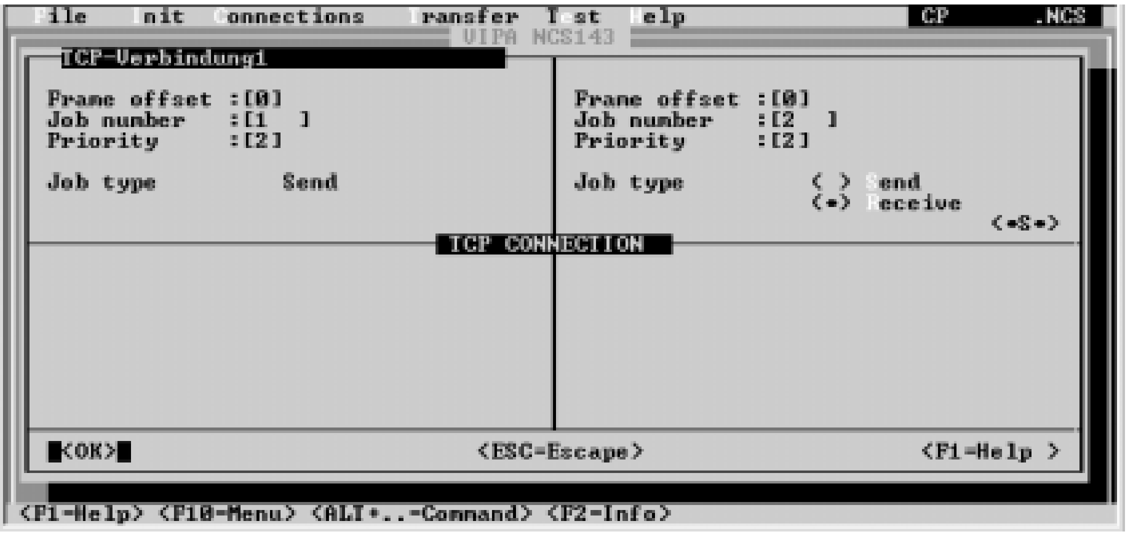
In this case the right byte represents the pointer to the parameter field.

5.8.4 Multi-connections

Except for the configuration window for H1-datagram connections, all configuration windows provide the facility to configure multi-connections. Multi-connections refer to multi-drop connections. Here you may configure a second SSNR/ANR for an existing SSNR/ANR and its TSAP or port. The respective and opposite job types are configured automatically. You may also choose to let the multi-connection operate via a different channel.

You access the input window for multi-connections by means of the < **Multi** > button. Only the descriptions of the input windows for multi-connections are different.

For example, the input window for a multi-TCP-connection has the following contents:


File Init Connections Transfer Test Help
CP .NCS
VIPA NCS143
TCP-Verbindung1
TCP CONNECTION
<OK>
<ESC=Escape>
<F1=Help >
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

```
Frame offset  :[0]                    Frame offset  :[0]
Job number    :[1   ]                 Job number    :[2   ]
Priority      :[2]                    Priority      :[2]

Job type           Send               Job type           ( ) Send
                                                         (•) Receive
                                                                  <•S•>
```

Fig. 5-15: Interactive window for multi-connections

Parameters of a multi-connection

Frame offset (SSNR-offset)

You may specify a different frame offset for a multi-connection. The frame offset (SSNR-offset) determines the frame from the start address that is used as communication job by the current job. Frame offsets are normally only used in multi-processor systems with values larger than zero.

Please use the following values when assigning frame offsets:

- Frame offset 0 for CPU 1
- Frame offset 1 for CPU 2
- Frame offset 2 for CPU 3
- Frame offset 3 for CPU 4

Range: 0 to 3

Job number (ANR)

A job number**{** XE "Job number" **}** must be allocated to each connection that should be used by Siemens handler modules. This number is used to access the connection from the PLC program. The job number together with the frame offset number determines the connection. This precludes multiple assignments. The job number and the respective frame offset number must be supplied in the control program.

Range: 1 to 199

Priority

The respective value is retrieved from the default connection. You can not alter this value where multi-connections are concerned.

The parameter specifies the priority of the message for the CP143 H1 / TCP/IP. The priority may range from 1 (highest priority) to 3 (lowest priority).

1 is also called the express priority, **2 and 3** are normal priorities. Priorities 0 and 4 are not supported. Please remember that the data transfer for the express priority 1 is no faster than that of the normal priorities.

Range: 1 to 3

Job type

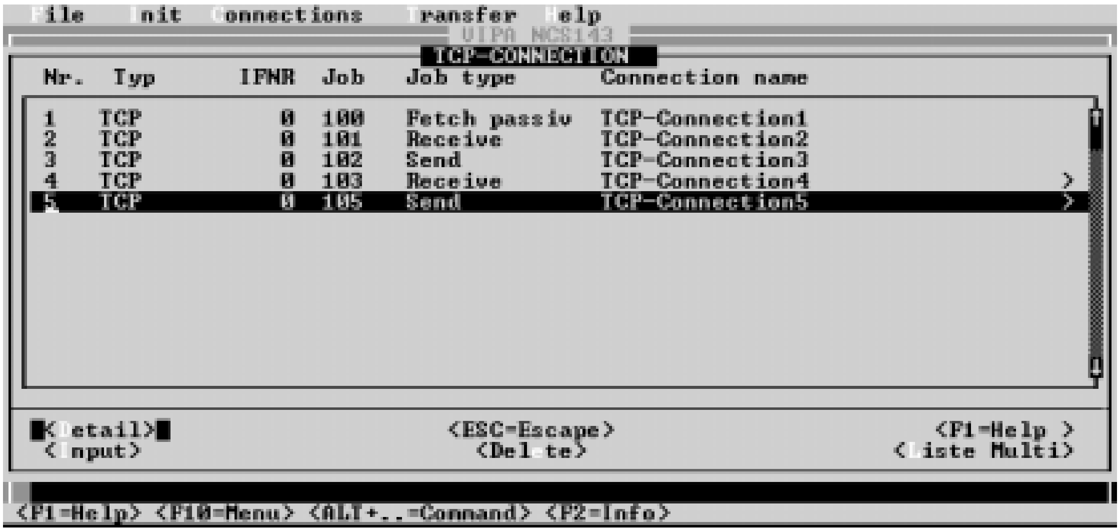
The opposite job type is selected automatically.

The following types of job are available:

• Send	transmission of unstructured data.
• Receive	reception of unstructured data.

5.8.5 List of connections

You may obtain a list of the configured connections for the respective protocol from the configuration window. This means that the content of the connection list is dependent on the interactive window from where the list was requested. The display will only show the connections for a single protocol. Depress the **< List >** button in the interactive window to request a list of connections. This will display a list of the connections that you have created for the respective protocol.



```
Nr.  Typ        IFNR  Job   Job type       Connection name

1    TCP           0  100   Fetch passiv   TCP-Connection1
2    TCP           0  101   Receive        TCP-Connection2
3    TCP           0  102   Send           TCP-Connection3
4    TCP           0  103   Receive        TCP-Connection4                >
5    TCP           0  105   Send           TCP-Connection5                >
```

Fig. 5-16: List of connections

The parameters of a connection list

No.

The number of the respective connection. Multi-connections are displayed under a single number.

Type

This indicates the type of protocol for the connection.

SSNR

This is the frame offset. The frame offset (SSNR-offset) determines the frame from the start address that is used as communication job by the current job. Frame offsets are normally only used in multi-processor systems with values larger than zero. Please use the following values when assigning frame offsets:

- Frame offset 0 for CPU 1
- Frame offset 1 for CPU 2
- Frame offset 2 for CPU 3
- Frame offset 3 for CPU 4

Range: 0 to 3

ANR

ANR represents the job number. A job number must be allocated to each connection that should be used by Siemens handler modules. This number is used to access the connection from the PLC program. The job number together with the frame offset number determines the connection. This precludes multiple assignments. The job number and the respective frame offset number must be supplied in the control program.

Range: 1 to 199

Job type

This indicates the type of job. The following job types may be displayed:

- Send Direkt transmission of unstructured data.
- Receive Direkt reception of unstructured data.
- Fetch Aktiv read requested data from another controller.
- Fetch Passiv used for the passive side of the fetch.
- Write Aktiv write the requested data into another controller.
- Write Passiv used on the passive side for a write.

Connection name

This is the name of the connection.

">" following the connection names

Represents a multi-connection. You receive a listing of multi-connections by depressing the **[List Multi]** button. In addition to < **ESC=Escape** > and < **F1=Help** > the interactive window also contains the following buttons:

< Detail >

This button provides access to the input screen for detailed connection parameters. There you may alter the values and accept them by depressing **< OK >**.

< Input >

The < **Input** > button displays an input screen for detailed connection parameters for a new connection..

< Delete >

The selected connection is deleted after you have confirmed the deletion.

< Liste Multi >

You may display any multi-connections that exist for the selected connection by depressing the **< Liste Multi >** button.

5.8.6 Clearing all connections

The NCS provides an option to clear all connections with a single command. The initialization parameters are saved. This function is located under the menu item **connections**.

You are asked to verify the clearing of all connections to avoid accidental deletions.

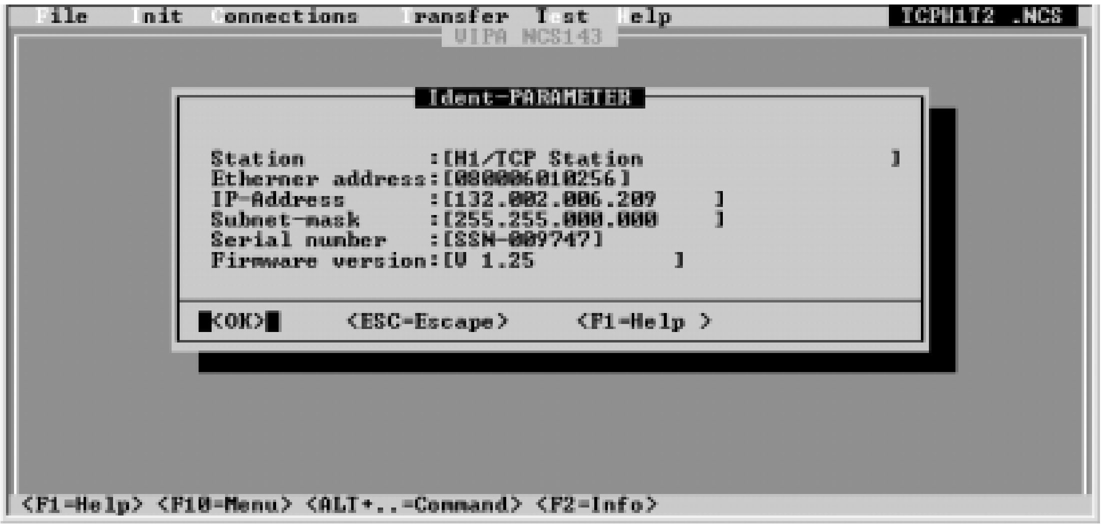
5.9 General test functions

The menu item **Test** provides access to an extensive set of functions that return information on a module that is connected in online mode.

You must supply the online parameters before you use this function (**Init** sub-menu *Online*). See also 5.6.2.

5.9.1 Ident

You can use this function to identify the module in online mode. The identification function displays all important addresses, the serial number and the name of the module in the following window:



```
Station          :[H1/TCP Station                 ]
Etherner address:[080006010256]
IP-Address       :[132.002.006.209     ]
Subnet-mask      :[255.255.000.000     ]
Serial number    :[SSN-009747]
Firmware version:[V 1.25            ]
```

Fig. 5-17: Display window for Ident parameters

The data displayed in this output window can not be modified.

Ident parameters

Station

The name of the respective station. This may be entered and/or modified under **Init** *CP Init*.

The name is not displayed if the module has not been loaded with a database. Station names increase readability and assist with online configurations.

Ethernet address

The ethernet address of the respective station. This may be entered and/or modified under **Init** *CP Init*.

The original ethernet address is displayed if the module has not been loaded with a database.

IP-Adresse

The IP address of the module. This may be entered and/or modified under **Init** *CP Init*.

The IP address is calculated from the ethernet address if the module has not been loaded with a database.

Subnet-Mask

The Subnet-Mask for IP addresses. This may be entered and/or modified under **Init** *CP Init*.

Serial number

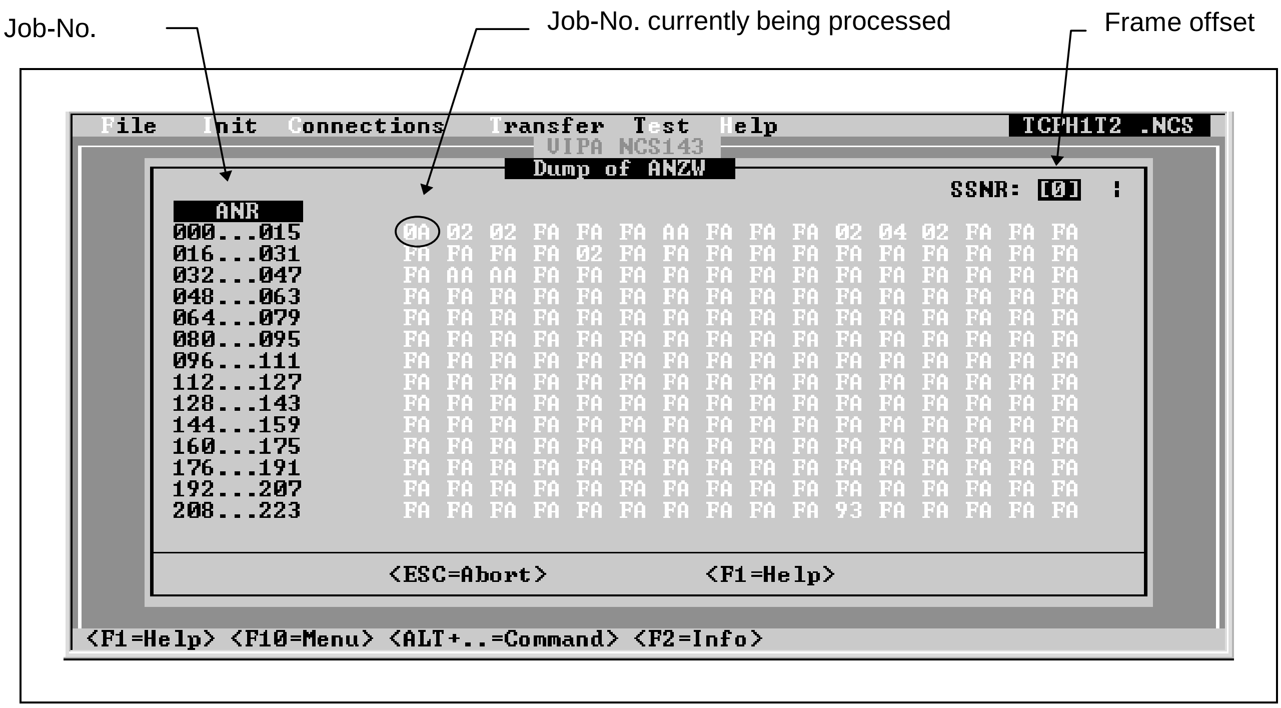
The serial number of the module. This value can not be modified.

Firmware-Version

The version number of the firmware installed on the module. This value can not be modified.

5.9.2 Anzw Dump

The function *Anzw Dump* opens a window containing the indicator words between 0 and 223. The indicator words are displayed as hexadecimal bytes.


Job-No.
Job-No. currently being processed
Frame offset
File Init Connections Transfer Test Help
TCPH1T2 .NCS
VIPA NCS143
Dump of ANZW
<ESC=Abort>
<F1=Help>
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

```
                                                          SSNR: [0]   :
   ANR
000...015     0A 02 02 FA FA FA AA FA FA FA 02 04 02 FA FA FA
016...031     FA FA FA FA 02 FA FA FA FA FA FA FA FA FA FA FA
032...047     FA AA AA FA FA FA FA FA FA FA FA FA FA FA FA FA
048...063     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
064...079     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
080...095     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
096...111     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
112...127     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
128...143     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
144...159     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
160...175     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
176...191     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
192...207     FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA FA
208...223     FA FA FA FA FA FA FA FA FA FA 93 FA FA FA FA FA
```

Fig. 5-18: Output window of Anzw Dump

Please note!

The indicator word with the job no. 0 has special significance. Job no. 0 contains the current job number that is being processed by SEND ALL or RECEIVE ALL tasks on the module.

The display is constantly updated. You can terminate the process by means of the **<ESC> key**. You may then modify the page frame offset under SSNR.

A single update of the displayed values can be initiated by clicking on **[**update**]**.

To restart the continuous update of the display you must click **[OK]**.

If you press **<ESC>** again the window is closed.

Indicator words returned by Anzw Dump

Indicator word: FA

Error indicator "F" means that the respective job has not been defined on the CP 143. Status indicator A means that the job has been inhibited (for SEND / FETCH and RECEIVE).

Indicator word: AA

Error indicator "A" means that the connection of the communication task has not or not yet been established. Status indicator "A" inhibits SEND as well as the RECEIVE and FETCH.

Indicator word: 08

The connection has been established again (e.g., when the CP has re-started), SEND is released (SEND communication task).

Indicator word: 09

The connection has been established again, RECEIVE is released. (RECEIVE communication task).

Indicator word: 04

SEND was completed without errors, the respective data was transferred.

Indicator word: 05

RECEIVE was completed without errors, the respective data was transferred arrived on the PLC.

Indicator word: 02

The SEND-, RECEIVE-, READ- or WRITE-job is active. The partner of the SEND task has not yet changed to RECEIVE. The partner of the RECEIVE task has not yet issued a SEND.

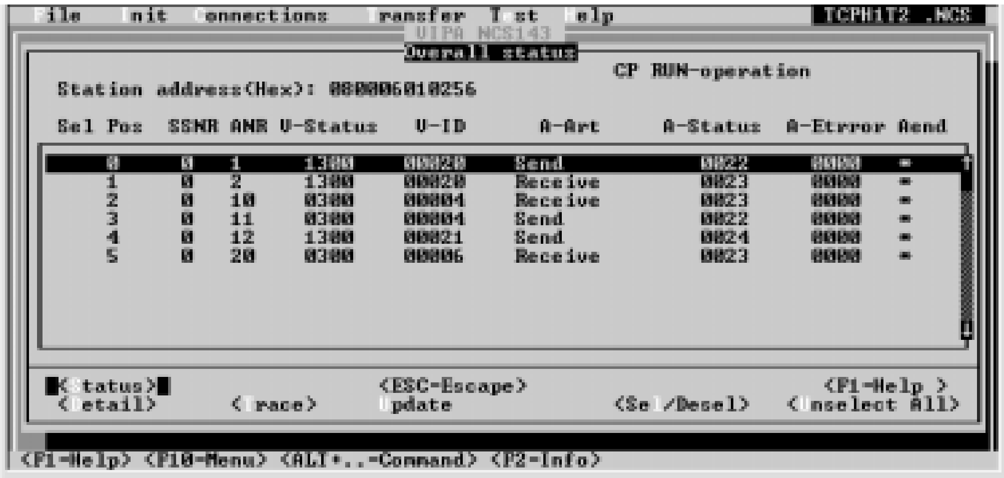
5.9.3 Overall status functions

Menu item **Test** provides access to four sub-menus for displaying the overall status. The overall status function returns information on all the configured connections. Up to a maximum of 10 connections can be displayed simultaneously.

Depending on the selected menu item the display will show all transport, datagram, TCP/IP or UDP connections. The overall status display provides an option to select a connection, to monitor the status of this connection in online mode by means of [Detail], or to display the current status of the connection by means of [Trace].

You may also mark individual connections by means of [Sel/Desel] and display the respective status by means of [Status].

The basic structure of the overall status display window is as follows:



```
 File    Init   Connections    Transfer  Test  Help                 TCPH1T2 .NCS
                                    VIPA NCS143
                                  Overall status
                                                   CP RUN-operation
  Station address(Hex): 080006010256

  Sel Pos   SSNR ANR V-Status    V-ID      A-Art       A-Status  A-Etrror Aend

        0      0    1      1300    00020    Send            0022     0000    =
        1      0    2      1300    00020    Receive         0023     0000    =
        2      0    10     0300    00004    Receive         0023     0000    =
        3      0    11     0300    00004    Send            0022     0000    =
        4      0    12     1300    00021    Send            0024     0000    =
        5      0    20     0300    00006    Receive         0023     0000    =

  <Status>                      <ESC-Escape>                          <F1-Help >
   <Detail>        <Trace>      Update                <Sel/Desel>  <Unselect All>

<F1-Help> <F10-Menu> <ALT+..-Command> <F2-Info>
```

Fig. 5-19: Structure overall status window

Column headings

Sel
column indicating the connection(s) that have been marked by means of [Sel/Desel].

Pos
the position of the connection in the list

SSNR
the number of the page frame that is used by the connection.

ANR
job number of the respective connection

V-Status
connection status (see table in chapter 5.9.3.1.1)

V-ID
internal connection number used by the system.

A-Art

displays the job type: SEND, RECEIVE, READ AKTIV (PASSIV), WRITE AKTIV (PASSIV).

A-Status

returns the processing status of the job (see table in chapter 5.9.3.1.2).

A-Fehler

errors that occurred when processing jobs (see chapter 5.9.3.1.3)

Aend

status changes are depicted by a "*".

Buttons

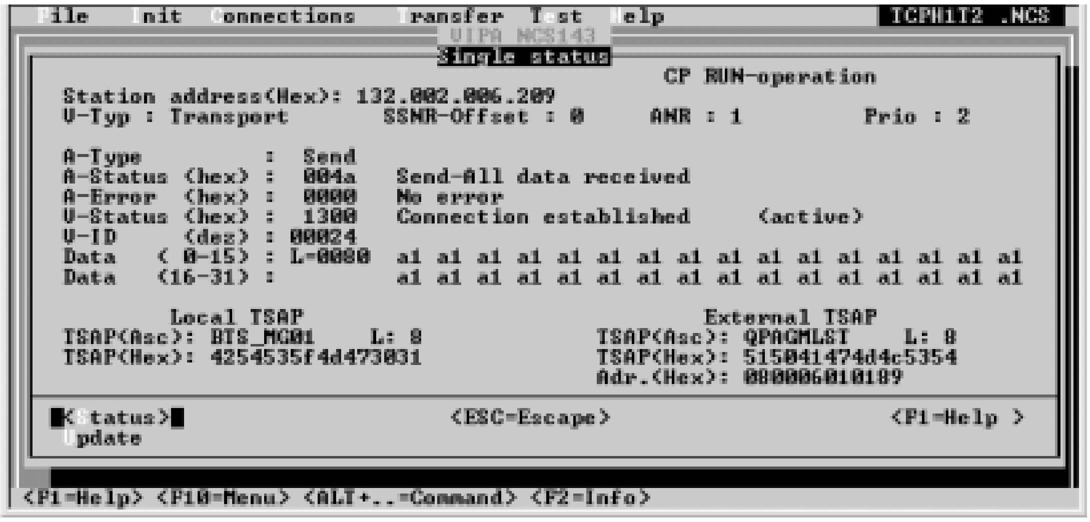
Status

returns the status in online mode for all the connections that are visible in the mask. The data in the respective tables is updated on a cyclic basis.

When you have marked certain connections by means of [Sel/Desel] only the status of the respective connections is displayed.

Detail

When you select [Detail] the status for the selected connection (selection bar) is displayed:



```
 ile    nit   onnections     ransfer  T st   elp                    TCPH1T2 .NCS
                                  VIPA NCS143
                                 Single status
                                                CP RUN-operation
  Station address(Hex): 132.002.006.209
  V-Typ : Transport        SSNR-Offset : 0     ANR : 1          Prio : 2

  A-Type           :  Send
  A-Status  (hex)  :  004a   Send-All data received
  A-Error   (hex)  :  0000   No error
  V-Status  (hex)  :  1300   Connection established     (active)
  V-ID      (dez)  : 00024
  Data    ( 0-15)  : L=0080  a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1
  Data    (16-31)  :         a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1 a1

          Local TSAP                               External TSAP
  TSAP(Asc): BTS_MG01     L: 8              TSAP(Asc): QPAGMLST     L: 8
  TSAP(Hex): 4254535f4d473031               TSAP(Hex): 515041474d4c5354
                                            Adr.(Hex): 080006010189

 <Status>                       <ESC=Escape>                       <F1=Help >
  pdate

<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

This mask contains all the information for a connection message. When you terminate the update cycle by means of <ESC> key you are placed in a mode where you can step through the connection and display the status by means of [update].

Trace

All messages for the selected connection (selection bar) are listed in online mode. This list can accommodate a maximum of 200 of the most recent messages. When this limit is exceeded the older messages are overwritten.

You can page through the list as soon as you depress the <ESC> key.

To view the data that is transferred with the individual messages you can click on [display data]. If you then return to online trace mode by means of [Status] the most up to date messages are displayed along with the respective data.

⚠ The *Detail* and *Trace* functions influence the run time behavior of your connection!

update

The display of connection data in the overall status list is brought up to date for this one time.

Sel/Desel

You may mark or remove the marking of selected connections by means of [Sel/Desel].

Unselect All

All markings are removed.

5.9.3.1 Status and error indication for the overall status functions

5.9.3.1.1 V-Status

V-status displays the status of a connection.

Code	Description
sF00	Connection error
s000	Initialization phase active
s100	Connection is being established
s101	Connection is being re-established
s300	Connection established
s500	Local termination of connection
s501	Connection terminated by partner

Tab. 5-3: Status of a connection

s: 1=aktive; 0=passive; 2=dynamic

5.9.3.1.2 A-Status

A-status displays the status of the current job

Code	Description
0000	Currently no job processing
0020	Data to remote station
0021	Data from remote station
0022	Wait for SEND acknowledgement
0023	Wait for data from remote station
0024	Local SEND acknowledgement
0040	Received SEND-DIREKT from the PLC
0048	Wait for SEND-ALL initiation
004A	SEND-ALL data received
0050	Wait for RECEIVE-DIREKT initiation
0052	Ack. received from RECEIVE-DIREKT
0058	Wait for RECEIVE-ALL initiation
005A	RECEIVE-ALL acknowledgement received
0060	FETCH-DIRECT received from PLC

Tab. 5-4: Status of job processing

5.9.3.1.3 A-error

Any error that has occurred during error processing is displayed here.

The following error messages are possible:

0 No error
if the bit "job completed with error" is set the CP143 H1 / TCP/IP had to re-establish the connection, i.e. after a restart or a RESET.

1 Bad Q/ZTYP on the HTB
the configuration for the job had a bad TYP parameter.

2 Area does not exist in the PLC
when the job was initiated the configuration contained a bad DB (DBNR).

3 Area in the PLC is too small
the sum of Q/ZANF and Q/ZLAE exceeds the area limits. The area limit of data modules is determined by the module's size. The area limits for flags, timers, counters etc. depends on the PLC.

4 QVZ error in the PLC
The source or target parameter refers to a PLC memory area that contains damaged memory modules or where memory modules have not been installed. The QVZ error can only occur for Q/ZTYP AS, PB, QB or when a memory error has occurred.

5 Error in the indicator word
The configured indicator word could not be processed. This error signifies that the data word or double word designated by ANZW is not or no longer located in the specified data module, i.e. the DB was too small or did not exist.

6 Not a valid ORG format
The target or the source for the data was not specified in the handle module (Q/TYP="NN") or in the connection module.

7 Reserved

8 No available transport connection
The capacity for establishing transport connections has been exceeded. Please clear unnecessary connections

9 Remote error
An error occurred in the communication partner for a READ/WRITE job.

A Connection error
The connection for the respective job has not or not yet been established. The error disappears when a connection can be established. When all connections of a CP are disconnected this error indicates that the module or the LAN cable has been damaged. This error can also be caused by an incorrect configuration, e.g. bad addressing.

B Handshake error
This may indicate a system error or that the selected data block size is too large.

C Initiation error
The incorrect handler module was used to initiate the job or the transferred data block was too large.

D Termination after RESET
This is a message from the operating system. For priority 1 and 2 the connection has been interrupted and will be re-established once the communication partner has reached the stage that it can accommodate a new connection. In the case of priority 3 connections the connection has been cleared but a new connection may be initiated at any time.

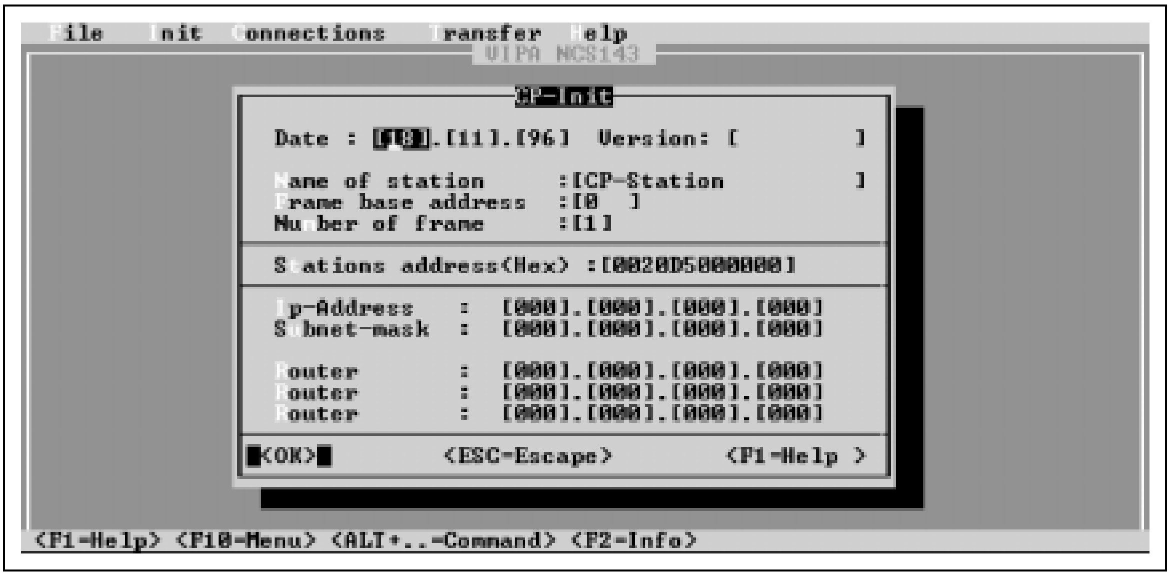
E Job with bootstrap function
This is a message from the operating system. The job is a READ/WRITE-PASSIV job and can not be started from the PLC.

F Job does not exist
The selected job does not exist on the CP143 H1 / TCP/IP. This error may occur if the SSNR/A-NR combination in the handler module is bad or when the connection module has been omitted.

5.10 Application in combination with H1

5.10.1 Initialization

The entry *CP-Init* under **Init** places you in an interactive window that provides access to the CP-base initialization:


File Init Connections Transfer Help
VIPA NCS143
CP-Init
<OK>
<ESC-Escape>
<F1-Help >
<F1-Help> <F10-Menu> <ALT+..-Command> <F2-Info>

```
Date : [18].[11].[96]  Version: [        ]

Name of station      :[CP-Station        ]
Frame base address   :[0  ]
Number of frame      :[1]

Stations address(Hex) :[0020D5000000]

Ip-Address   :  [000].[000].[000].[000]
Subnet-mask  :  [000].[000].[000].[000]

Router       :  [000].[000].[000].[000]
Router       :  [000].[000].[000].[000]
Router       :  [000].[000].[000].[000]
```

Fig. 5-20: CP-base initialization (initial initialization)

Here you must configure the addresses and other identification parameters of the CP.

Before entering a connection, you must also enter a frame base address into this interactive mask. The entered data is transferred when you depress the Enter key.

Parameters of the CP-initialization

Date, version

The entries for date and version are for documentation purposes only. You can use these entries to check at what time a configuration was performed. The version number will always contain the number associated with the file that is currently loaded.

Station name

You may specify a name for your base station. The name may consist of a maximum of 19 characters. Station names are used to simplify the configuration of connections. The station name is important where on-line configurations are concerned.

Frame base address

The frame base addressFrame base address determines the origin of the 4 frames that are inserted into the memory area of the controller. The PLC program normally operates on the base frame. Any other frames may be used by other controller or systems.

The frame base address must be divisible by 4 and it is located between 0 and 255. This means that you may enter the values 0, 4, 8, 12, ... 252.

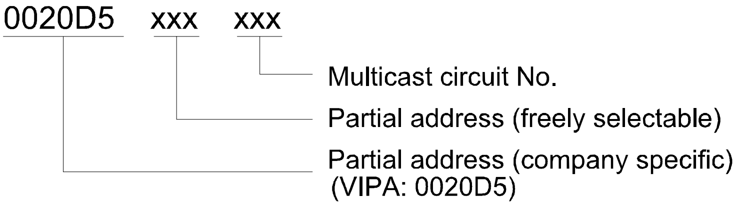
The frame base addresses on a single controller may not overlap.

Number of frames

Specifies the number of frames that the CP may use to communicate with the PLC. The CP always occupies 4 frames (e.g. 4, 5, 6, 7).

Station address

Here you must specify the ethernet address (station address). The structure of the ethernet address is as follows:


0020D5 xxx xxx
Multicast circuit No.
Partial address (freely selectable)
Partial address (company specific)
(VIPA: 0020D5)

When this entry is 000000000000 the hardware address of the module will be used.

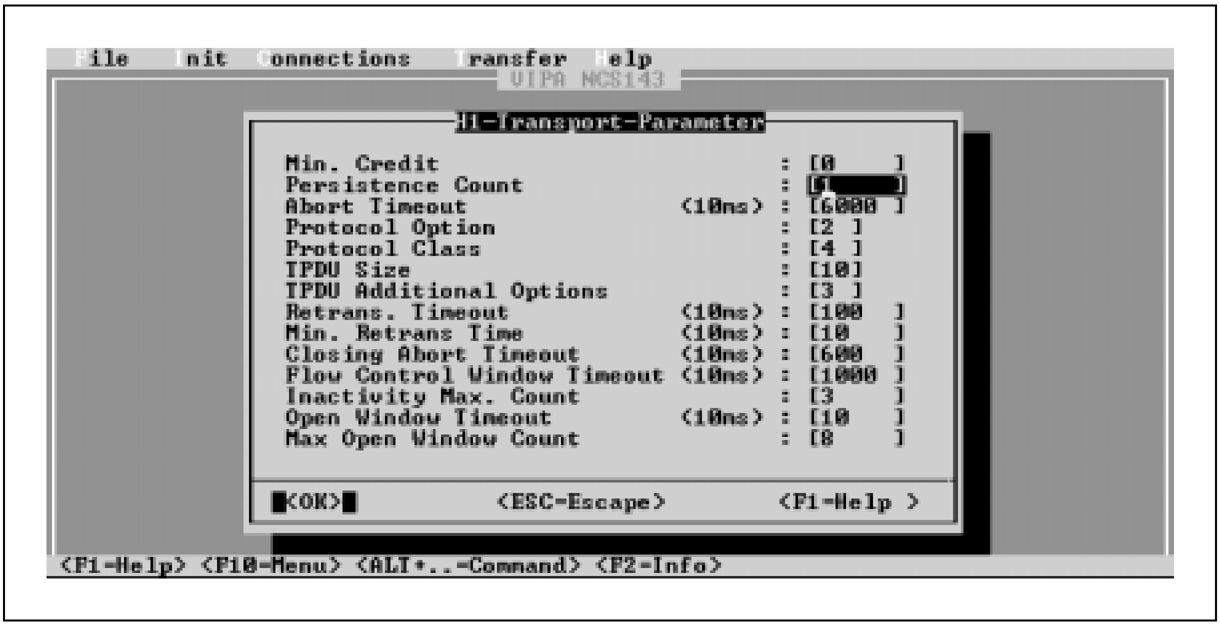
IP-Address, subnet-mask and router

No entries required when you are working in combination with H1.

5.10.1.1 Initialization of H1 transport parameters

The H1 system parameters represent the operational parameters of level 4 that are required for master-slave communications. Only under special circumstances should these values be modified.

You can access the following dialog window via *H1 Transport-Parameter* under **Init**:



```
Min. Credit                             : [0     ]
Persistence Count                       : [1     ]
Abort Timeout                   <10ms>  : [6000  ]
Protocol Option                         : [2 ]
Protocol Class                          : [4 ]
TPDU Size                               : [10]
TPDU Additional Options                 : [3 ]
Retrans. Timeout                <10ms>  : [100   ]
Min. Retrans Time               <10ms>  : [10    ]
Closing Abort Timeout           <10ms>  : [600   ]
Flow Control Window Timeout     <10ms>  : [1000  ]
Inactivity Max. Count                   : [3     ]
Open Window Timeout             <10ms>  : [10    ]
Max Open Window Count                   : [8     ]
```

Fig. 5-21: H1 Transport parameters

The parameters for H1 initialization

Min. Credit

The CP basically executes a 0 credit procedure and it therefore reacts like a Siemens CP143.

This value is fixed at 0. It can not be modified.

Persistence Count

Range: 0 to 65535 (Default: 1)

Abort Timeout

Persistence count x Abort Time = max. time during which the TP4 protocol stack establishes a connection on its own or in which it waits for a connection

Range: 0 to 65535, in units of 10 ms (Default: 6000).

Protocol Option

This value is fixed at 2. It can not be modified.

Protocol Class

This value is fixed at 4 It can not be modified.

TPDU Size

Range: 0 to 13 (Default: 10)

TPDU Additional Options

This value is fixed at 3 It can not be modified.

Retrans. Timeout

Interval for the connect retry in the abort-timeout time.

Range: 0 to 65535 in units of 10ms (Default: 100)

Min. Retrans Time

Retransmission of unacknowledged data frames.

Range: 0 to 65535 in units of 10ms (Default: 10)

Closing Abort Timeout

Maximum time during which the partner can acknowledge frames.

Range: 0 to 65535 in units of 10ms (Default: 600)

Flow Control Window Timeout

Idle supervision for a connection

Range: 0 to 65535 in units of 10ms (Default: 1000)

Inactivity Max. Count

Idle supervision for a connection.

Range: 1 to 65535 (Default: 3)

Open Window Timeout

Retransmission of unacknowledged acknowledges with a credit > 1.

Range: 0 to 65535 in units of 10ms (Default: 10)

Max Open Window Count

Retransmission of unacknowledged acknowledges with a credit > 1.

Range: 1 to 65535 (Default: 8)

5.10.2 Configuration of H1 connections

H1 connections are configured via the submenu items *H1-Transport-connection* and *H1 Datagram-connection* located under **Connections**.

H1-connections are supported by the TP4-protocolstack.

Connection-oriented communications (transport control) are controlled by *H1-Transport-connection*.

Multicast- and broadcast messages are sent via the *H1-Datagram-connection*. This type of connection is not connection oriented.

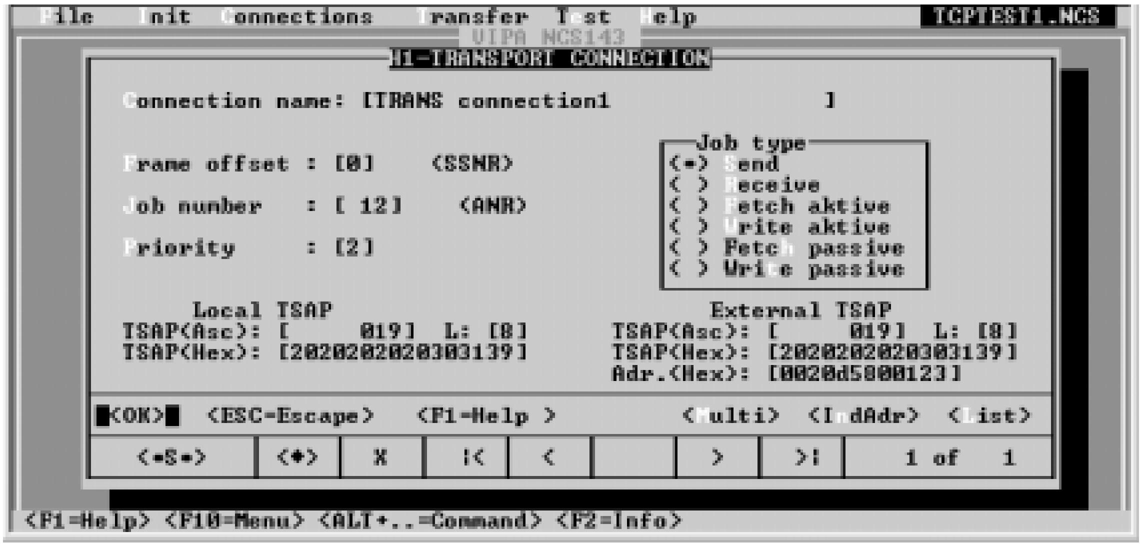
The following table contains all job types that are available for configuration under H1:

Job type	Job priority	Type of connection	Connection	On the CP1430 of Siemens this corresponds to	
				Job	READ/ WRITE
SEND	0/1	Active	Data	SEND	no
RECEIVE	0/1	Passive	Data	RECEIVE	no
SEND	2	Active	Trans	SEND	no
RECEIVE	2	Passive	Trans	RECEIVE	no
SEND RECEIVE	2	Active	Trans (multi-connection)	SEND RECEIVE	no
RECEIVE SEND	2	Passive	Trans (multi-connection)	RECEIVE SEND	no
SEND RECEIVE SEND RECEIVE	2 2 0/1 0/1	Active	Trans (multi-connection)	SEND RECEIVE SEND RECEIVE	no
SEND	3	Active	Trans	SEND	no
RECEIVE	3	Passive	Trans	RECEIVE	no
SEND	4	Active	Trans	SEND	no
RECEIVE	4	Passive	Trans	RECEIVE	no
FETCH aktiv	2 (implicit)	Active	Trans	FETCH	yes
FETCH passive	2 (implicit)	Passive	Trans	FETCH	yes
WRITE aktiv	2 (implicit)	Active	Trans	SEND	yes
WRITE passiv	2 (implicit)	Passive	Trans	RECEIVE	yes

Tab. 5-3: H1 job types

5.10.2.1 H1 Transport-connections

Select the sub-menu *H1Transport-connection* from the menu item **Connection** to establish connection oriented communications. This opens the following interactive window:



```
 File    Init   Connections    Transfer  Test  Help              TCPTEST1.NCS
                                  VIPA NCS143
                          H1-TRANSPORT CONNECTION

     Connection name: [TRANS connection1                ]

                                                   Job type
     Frame offset : [0]    (SSNR)                  (•) Send
                                                   ( ) Receive
     Job number   : [ 12]    (ANR)                 ( ) Fetch aktive
                                                   ( ) Write aktive
     Priority     : [2]                            ( ) Fetch passive
                                                   ( ) Write passive

          Local TSAP                                 External TSAP
     TSAP(Asc): [      019]  L: [8]          TSAP(Asc): [      019]  L: [8]
     TSAP(Hex): [2020202020303139]           TSAP(Hex): [2020202020303139]
                                             Adr.(Hex): [0020d5800123]

 <OK>  <ESC=Escape>  <F1=Help >            <Multi>  <IndAdr>  <List>
 <•S•> | <*> | X | |< | < |   | > | >| |   1 of   1
 <F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 5-22: Interactive window H1 connections

Parameters for an H1 Transport-connection

Connection name

You may specify a name for the connection. The name may consist of a maximum of 32 characters.

Frame offset

The frame offset (SSNR-offset) determines the frame from the start address that is used as communication job by the current job. Frame offsets are normally only used in multi-processor systems with values larger than zero.

Please use the following values when assigning frame offsets:

- Frame offset 0 for CPU 1
- Frame offset 1 for CPU 2
- Frame offset 2 for CPU 3
- Frame offset 3 for CPU 4

Range: 0 to 3

Job number

A job number must be allocated to each connection that should be used by handler modules. This number is used to access the connection from the PLC program. The job number together with the frame offset number determines the connection. This precludes multiple assignments. The job number and the respective frame offset number must be supplied in the control program.

Range: 1 to 199

Priority

This parameter specifies the priority of the message for the CP143plus. The priority may range from 0 (highest priority) to 4 (lowest priority).

When establishing a connection, priority 0,1 and 2 connections are so-called **static connections.** This means that the connection will be re-established, if possible, should it have been terminated (e.g. by the reset handler module).

When establishing a connection, priority 3 and 4 connections are so-called **dynamic connections**. In this case the connection is established by means of a job that initiates the application for the respective connection. Once a connection with priority 3 is established it exists until it is terminated. The CP will not attempt to re-establish the connection. Priority 4 connections are terminated automatically when the respective message has been transferred.

Range: 0 to 4

Job type

Here you specify the type of job. The following job types are available:

• Send	transmission of unstructured data.
• Receive	reception of unstructured data.
• Fetch Aktiv	read requested data from another controller.
• Fetch Passiv	used for the passive side of the fetch.
• Write Aktiv	write the requested data into another controller.
• Write Passiv	used on the passive side for a write.

Ethernet address (Adr)

The ethernet address identifies a station. The ethernet address of the station must be unique within the network. The ethernet address consists of 6 bytes. The first three bytes are specific to the manufacturer. These bytes are allocated by the IEEE-committee. The remaining three bytes are supplied by the end-user. The allocation of ethernet addresses are the responsibility of the systems network administrator.

Transport Service Access Point (TSAP)

Under H1 the TSAPs are used to establish communication channels for a connection. The length of a TSAP is 8 bytes. TSAP assignments may consist of hex or ASCII entries.

Local TSAP

This is the TSAP of your own (local) station.

External TSAP

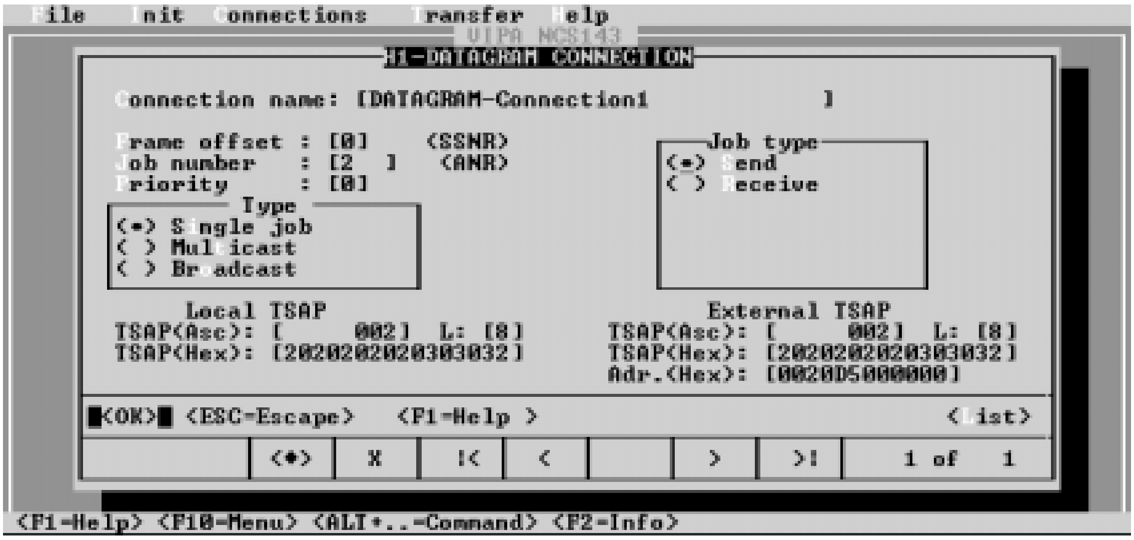
This is the TSAP of the destination station (external station).

5.10.2.2 H1 Datagram-connections

H1-Datagram connections are unsecured connections. The transfer is not subject to a negotiated type of connection. No acknowledgement is required when data is received. Lost datagrams are not detected, e.g. when the receive buffer is too small.

The datagram service is intended for multicast- and broadcast messages.

To initiate an unsecured connection select the sub-menu *H1 Datagram-connection* from the menu item **Connection**. The following interactive window will be displayed:



```
 File   Init  Connections   Transfer  Help
                     VIPA NCS143
                 H1-DATAGRAM CONNECTION

  Connection name: [DATAGRAM-Connection1          ]

  Frame offset : [0]    <SSNR>             Job type
  Job number   : [2  ]   <ANR>           (•) Send
  Priority     : [0]                     ( ) Receive
         Type
  (•) Single job
  ( ) Multicast
  ( ) Broadcast

        Local TSAP                           External TSAP
  TSAP(Asc): [      002]  L: [8]       TSAP(Asc): [      002]  L: [8]
  TSAP(Hex): [2020202020303032]        TSAP(Hex): [2020202020303032]
                                       Adr.(Hex): [0020D5000000]

 <OK> <ESC=Escape>   <F1=Help >                               <List>
            <•>   X   |<   <       >   >|        1 of    1
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 5-23: Interactive window H1 datagram-connections

Parameters for an H1 datagram-connection

Connection name

You may specify a name for the connection. The name may consist of a maximum of 32 characters.

Frame offset

The frame offset (SSNR-offset) determines the frame from the start address that is used as communication job by the current job. Frame offsets are normally only used in multi-processor systems with values larger than zero.

Please use the following values when assigning frame offsets:

- Frame offset 0 for CPU 1
- Frame offset 1 for CPU 2
- Frame offset 2 for CPU 3
- Frame offset 3 for CPU 4

Range: 0 to 3

Job number

A job number must be allocated to each connection that should be used by handler modules. This number is used to access the connection from the PLC program. The job number together with the frame offset number determines the connection. This precludes multiple assignments. The job number and the respective frame offset number must be supplied in the control program.

Range: 1 to 199

Priority

This parameter specifies the priority of the message for the CP143plus. The priority may range from 0 (highest priority) to 41 (lowest priority).

When establishing a connection, priority 0 and 1 connections are so-called **static connections.** This means that the connection will be re-established, if possible, should it have been terminated (e.g. by the reset handler module).

Range: 0 to 41

Job type

Here you specify the type of job. The following job types are available:

- Send transmission of unstructured data.
- Receive reception of unstructured data.

Type

Here you must specify the type of transfer. You may select to send unsecured messages to a specific receiver, to a selected group of receivers or to all receivers.

- Point-to-point job Point-to-point job an unsecured job for a single station.
- Multicast an unsecured job for selected multicast-circuits.

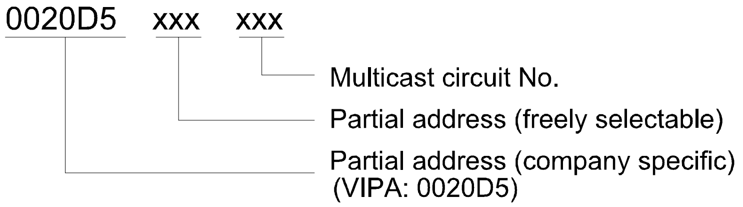
  When you select multicast, the multicast-identifier is transferred to the ethernet-address. You determine the multicast circuits by means of the last three nibbles of the ethernet address.
- Broadcast transmits a single message to all the participants that have been allocated the pre-set broadcast address or reception of single messages from participants that transmit via the pre-set broadcast address.

  The H1 broadcast address is FFFFFFFFFFFFh.

Ethernet address (Adr)

The ethernet address identifies a station. The ethernet address of the station must be unique within the network.

The structure of the ethernet address is as follows:


0020D5 xxx xxx
Multicast circuit No.
Partial address (freely selectable)
Partial address (company specific)
(VIPA: 0020D5)

The ethernet address consists of 6 bytes. The first three bytes are specific to the manufacturer. These bytes are allocated by the IEEE-committee. The remaining three bytes are supplied by the end-user. The allocation of ethernet addresses are the responsibility of the systems network administrator.

Broadcast-messages do not require the use of an ethernet address.

Transport Service Access Point (TSAP)

Under H1 the TSAPs are used to establish communication channels for a connection. The length of a TSAP is 8 bytes. TSAPs must always be specified (even for broadcast messages).

To simplify matters you may assign the same identifier to the external and the local TSAP.

Local TSAP

This is the TSAP of your own (local) station.

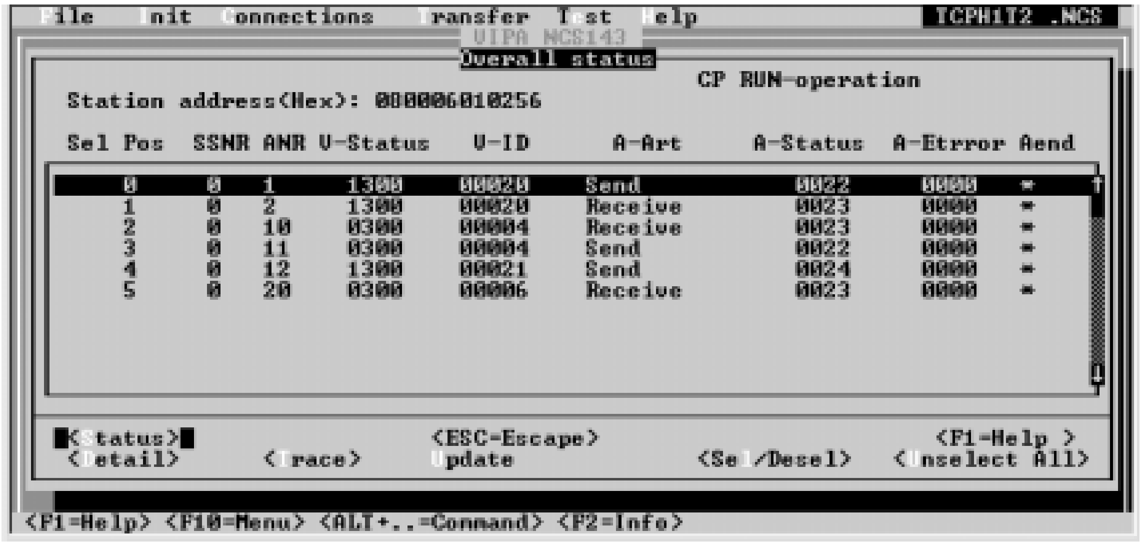
External TSAP

This is the TSAP of the destination station (external station).

5.10.3 Test functions

The menu item **Test** provides access to test and diagnostic functions for your H1 transport- or datagram jobs.

When you select *H1-Transport* or *H1-Datagramm* the following dialog window appears. This contains the overall status for all transport and datagram jobs of the respective module:



```
                                             CP RUN-operation
 Station address(Hex): 000006010256

 Sel Pos  SSNR ANR V-Status   V-ID      A-Art      A-Status  A-Etrror Aend

      0     0   1     1300    00020    Send           0022     0000    *
      1     0   2     1300    00020    Receive        0023     0000    *
      2     0   10    0300    00004    Receive        0023     0000    *
      3     0   11    0300    00004    Send           0022     0000    *
      4     0   12    1300    00021    Send           0024     0000    *
      5     0   20    0300    00006    Receive        0023     0000    *
```

Fig. 5-24: Test function under H1

Once you have selected the display of the overall status you can select a connection and monitor the status of this connection in online mode by means of [Detail] or to display the current status of this connection by means of [Trace].

You may also mark specific connections by means of [Sel/Desel] to display the only the selected status by means of [Status].

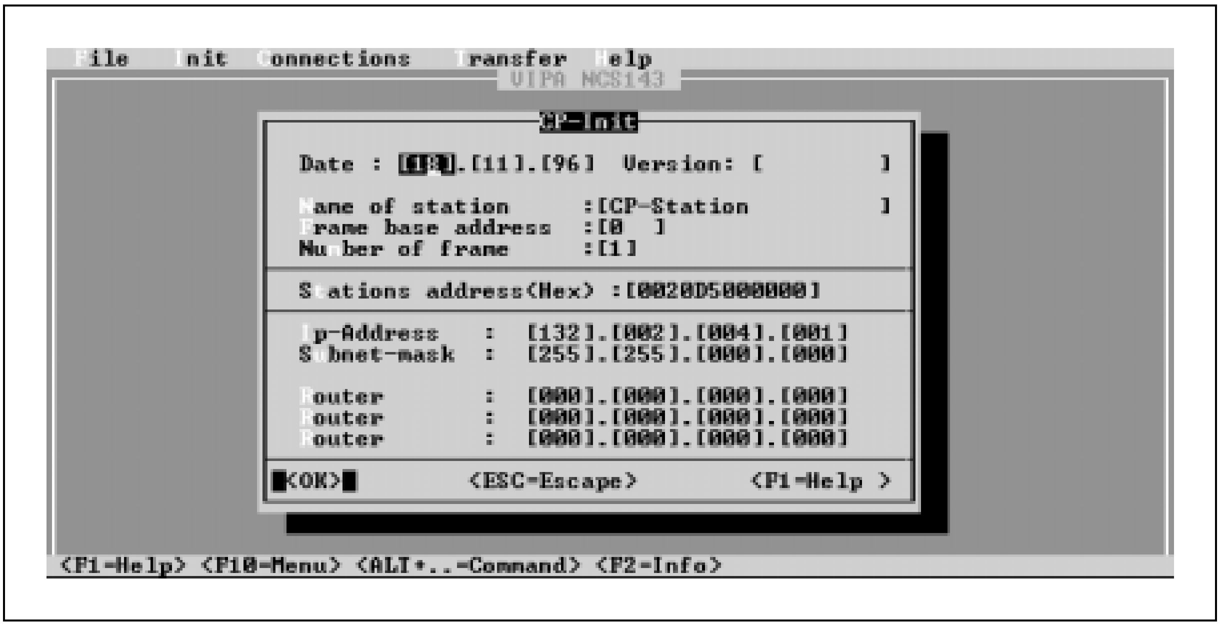
Chapter 5.9.3 contains a detailed description of this dialog window.

5.11 Application in combination with TCP/IP

5.11.1 Initialization

5.11.1.1 CP initialization

The entry *CP-Init* under **Init** places you in an interactive window that provides access to the CP-base initialization:


File Init Connections Transfer Help
VIPA NCS143
CP-Init

```
Date : [[illegible]].[11].[96]  Version: [         ]

Name of station      :[CP-Station        ]
Frame base address   :[0  ]
Number of frame      :[1]

Stations address(Hex) :[0020D5000000]

Ip-Address    :  [132].[002].[004].[001]
Subnet-mask   :  [255].[255].[000].[000]

Router        :  [000].[000].[000].[000]
Router        :  [000].[000].[000].[000]
Router        :  [000].[000].[000].[000]
```

Fig. 5-25: CP base initialization (initial initialization)

Here you must configure the addresses and other identification parameters of the CP.

Before entering a connection, you must also enter a frame base address into this interactive mask. The entered data is transferred when you depress the Enter key.

Parameter of the CP-initialization

Date, version

The entries for date and version are for documentation purposes only. You can use these entries to check at what time a configuration was performed. The version number will always contain the number associated with the file that is currently loaded.

Station name

You may specify a name for your base station. The name may consist of a maximum of 19 characters. Station names are used to simplify the configuration of connections. The station name is important where on-line configurations are concerned.

Frame base address

The frame base addressFrame base address determines the origin of the 4 frames that are inserted into the memory area of the controller. The PLC program normally operates on the base frame. Any other frames may be used by other controller or systems.

The frame base address must be divisible by 4 and it is located between 0 and 255. This means that you may enter the values 0, 4, 8, 12, ... 252.

The frame base addresses on a single controller may not overlap.

Number of frames

Specifies the number of frames that the CP may use to communicate with the PLC. The CP always occupies 4 frames (e.g. 4, 5, 6, 7).

Station address

For TCP/IP you must specify 000000000000h or you may omit the entry.

IP-address

The IP-address is a 32-bit-address that must be unique within the network. The address consists of 4 numbers that are separated by a full stop.

Range: 000.000.000.000 to 255.255.255.255

You can obtain the IP-address from your system administrator.

Subnet-mask

The IP-subnet-mask is a 32-bit filter mask for IP-addresses. This address is structured in the same way as the IP-address. It also has the same range.

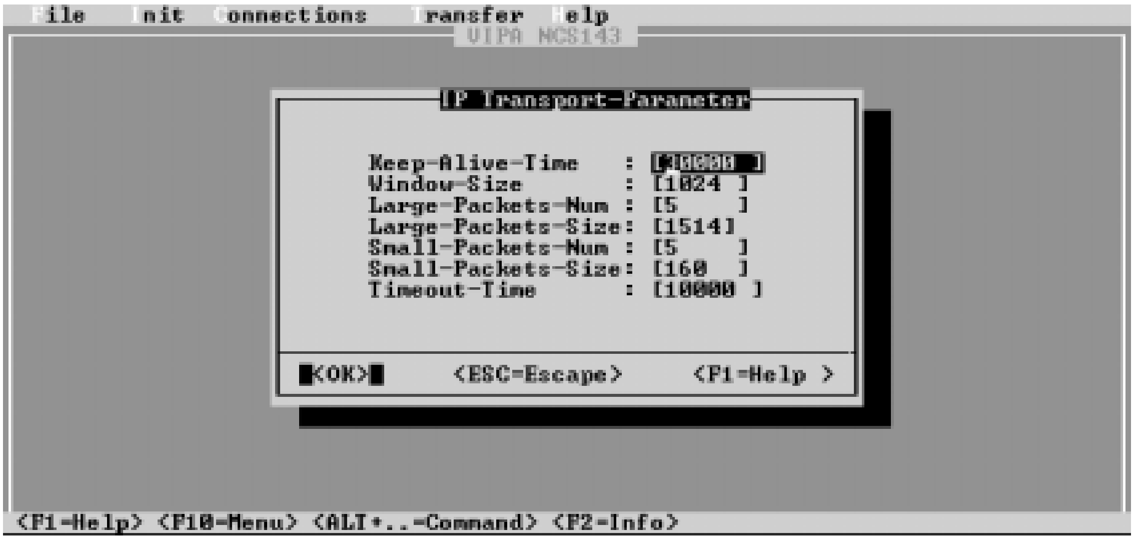
Subnets allow better control of the addressing area. Subnet-masks should be used where it is necessary to address a network on both sides of a router although only one network number is available.

Router

A Router (formerly gateway) can provide a link between different network topologies, e.g. ethernet to token ring. It may also be used to connect IP-networks with different network numbers. A router has an IP-address for every network. This IP-address provides the logical connection between the networks. 2 addresses exist where 2 networks are connected via a router. If you wish to avoid the use of router-IP-addresses when a connection is established you may enter 0.0.0.0 for the router.

5.11.1.2 TCP/IP-transport parameter initialization

The entry *IP Transport-parameter* under **Init** places you in an interactive window that allows initialization of the TCP/IP-Transport parameters for the local (own) CP (base-CP):


File Init Connections Transfer Help
VIPA NCS143
IP Transport-Parameter
<OK> <ESC=Escape> <F1=Help >
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

```
Keep-Alive-Time    : [30000 ]
Window-Size        : [1024  ]
Large-Packets-Num  : [5     ]
Large-Packets-Size: [1514]
Small-Packets-Num  : [5     ]
Small-Packets-Size: [160   ]
Timeout-Time       : [10000 ]
```

Fig. 5-26: IP Transport-parameters

Parameters for the TCP/IP-initialization

Keep-Alive-Time

This parameter specifies the idle-time limit for the TCP-connection in milliseconds. The idle-time limit is a timer that monitors inactive connections.

Range: 0 to 999999 ms (default: 30000 ms)

Window-Size

The size of the data window

Range: 1 to 65535 (default: 1024)

Large-Packets-Num

Number of transmit and receive buffers.

Range: 1 to 65535 (default: 5)

Large-Packets-Size

For ethernet this is fixed at 1514.

Small-Packets-Num

Number of acknowledgements

Range: 1 to 65535 (default: 5)

Small-Packets-Size

Specifies the size of small packets.

Range: 1 to 65535 (default: 160)

Time-out-Time

Internal CP-parameter; this should not be altered.

Range: 1 to 999999 (default: 10000)

5.11.2 Configuration of TCP/IP-connections

You can access the configuration options for TCP/IP-Connections via the sub-menu items *TCP-Connection* and *UDP-Connection* under **Connections**.

H1-connections are supported by the TP4-protocolstack.

TCP is used to provide connection-oriented communications (transport control)

Multicast- and broadcast messages are sent via the UDP-interface. UDP is not connection oriented.

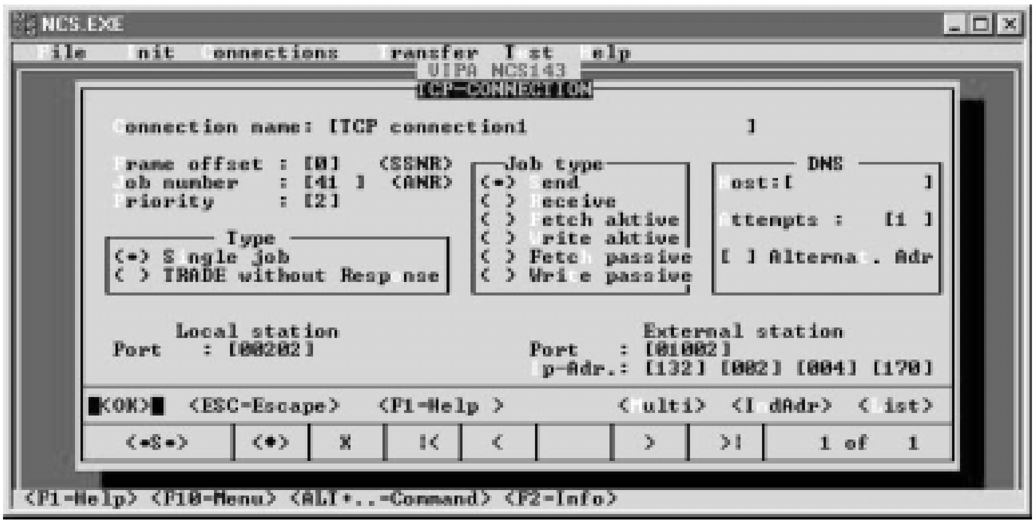
The following table contains a number of possible job types and connection types and the respective combinations:

Job type	Job priority	Type of connection establishment	Connection
SEND	1/2	Active	TCP / UDP / TRADA
RECEIVE	1/2	Passive	TCP / UDP
SEND RECEIVE	1/2	Active	TCP / UDP / TRADA (Multi-connection)
RECEIVE SEND	1/2	Passive	TCP / UDP / TRADA (Multi-connection)
SEND	3	Active	TCP / TRADA
RECEIVE	3	Passive	TCP / TRADA
FETCH aktiv	2 (implicit)	Active	TCP
FETCH passiv	2 (implicit)	Passive	TCP
WRITE aktiv	2 (implicit)	Active	TCP
WRITE passiv	2 (implicit)	Passive	TCP

Tab. 5-4: TCP/IP job types

5.11.2.1 TCP-Connection

You may enter the TCP configuration via the interactive mask below. TCP is a connection oriented protocol. TCP uses port numbers to address services on the application level. TCP provides reliable end-to-end services. It also provides configurable performance improvements for slower long-distance connections (PTT). The use of TCP is advisable where reliable data transfer services are required.


NCS.EXE
File Init Connections Transfer Test Help
VIPA NCS143
TCP-CONNECTION
<OK> <ESC=Escape> <F1=Help > <Multi> <IndAdr> <List>
<*S*> <*> X |< < > >| 1 of 1
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

```
Connection name: [TCP connection1                    ]

Frame offset : [0]    <SSNR>     Job type              DNS
Job number   : [41 ]  <ANR>      (*) Send              Host:[          ]
Priority     : [2]               ( ) Receive
                                 ( ) Fetch aktive      Attempts :   [1 ]
        Type                     ( ) Write aktive
(*) Single job                   ( ) Fetch passive     [ ] Alternat. Adr
( ) TRADE without Response       ( ) Write passive

      Local station                          External station
Port    : [00202]                    Port    : [01002]
                                     Ip-Adr.: [132] [002] [004] [170]
```

Fig. 5-27: Interactive window TCP-Connection

Parameters for a TCP/IP-Connection

Connection name

You may specify a name for the connection. The name may consist of a maximum of 32 characters.

Frame offset (SSNR-Offset)

The frame offset (SSNR-offset) determines the frame from the start address that is used as communication job by the current job. Frame offsets are normally only used in multi-processor systems with values larger than zero.

Please use the following values when assigning frame offsets:

- Frame offset 0 for CPU 1
- Frame offset 1 for CPU 2
- Frame offset 2 for CPU 3
- Frame offset 3 for CPU 4

Range: 0 to 3

Job number (ANR)

A job number must be allocated to each connection that should be used by handler modules. This number is used to access the connection from the PLC program. The job number together with the frame offset number determines the connection. This precludes multiple assignments. The job number and the respective frame offset number must be supplied in the control program.

Range: 1 to 199

Priority

This parameter specifies the priority of the message for the CP143plus. The priority may range from 1 (highest priority) to 3 (lowest priority).

When establishing a connection, priority 1 and 2 connections are so-called **static connections.** This means that the connection will be re-established, if possible, should it have been terminated (e.g. by the reset handler module).

When establishing a connection, priority 3 connections are so-called **dynamic connections**. In this case the connection is established by means of a job that initiates the application for the respective connection. Once a connection is established, it exists until it is terminated. The CP will not attempt to re-establish the connection.

Jobs with a priority of 1 differ from those with priority 2 and 3 in the manner that the job is processed internally. The size of the user data for priority 1 jobs is limited to the pre-set block size. Data can not be blocked by means of ALL handler modules.

These statements only apply to connection oriented jobs that make use of TCP.

Range: 1 to 3

Type

Single job

The order types SEND and RECEIVE, send and receive the messages in TCP protokolls as reference data without any additional Header.

With the type single order, the message data exchange should consist of defined, fixed message lengths!

This means that the Default Parameter "-1" cannot be used in the length data in the FB RECEIVE!

TRADA without Response

The reference data is preceded by a Header in the order type SEND (Header set-up see Chap. 4.3.5.2). A message sent from the remote station will be examined at the Header with the order type RECEIVE and the reference data will be transferred to the control according to the transferred length.

Different telegram lengths are possible with the type TRADE!

The Default Parameter "-1" is permissable in the length data in FB RECEIVE!

DNS

Domain Name System: makes a relation between the symbolic names and IP-Adress.

The communication is made operator friendly by this. A DNS-Server comes into use over which the communication abaut a domain is carried out.

The DNS-Server manages the symbollic names and with the help of the "Host-Name-Table" it makes a relation between the symbolic names an IP-Adress during the communication.

HOST

The IP-Adress data of the DNS-Server takes place here.

Attemps and alternative Adresses

The communication is controlled over the DNS-Server with the set-up of a connection. If the DNS-Server is not available then the altenative IP-Adress will be used after the attempt if it is marked. If the alternative adress is not marked then an error message will be given.

Job type

Here you specify the type of job. The following job types are available:

• Send	transmission of unstructured data.
• Receive	reception of unstructured data.
• Fetch Aktiv	read requested data from another controller.
• Fetch Passiv	used for the passive side of the fetch.
• Write Aktiv	write the requested data into another controller.
• Write Passiv	used on the passive side for a write.

Port address for local and for external stations

The port address determines the address of the connection that is used for the transfer of data. The port address for the local station may be compared to the local (own) TSAP on the H1-level. A port address consists of a decimal number (16-bit-address) of 5-digits. Both, TCP as well as UDP use port addresses for data transfers to the respective application level. Please note that port addresses 1024, 7777 and the range from 0 to 255 have been reserved for common services. The port address for the local station and the external station must be different.

Range: 0 to 65536

IP-address

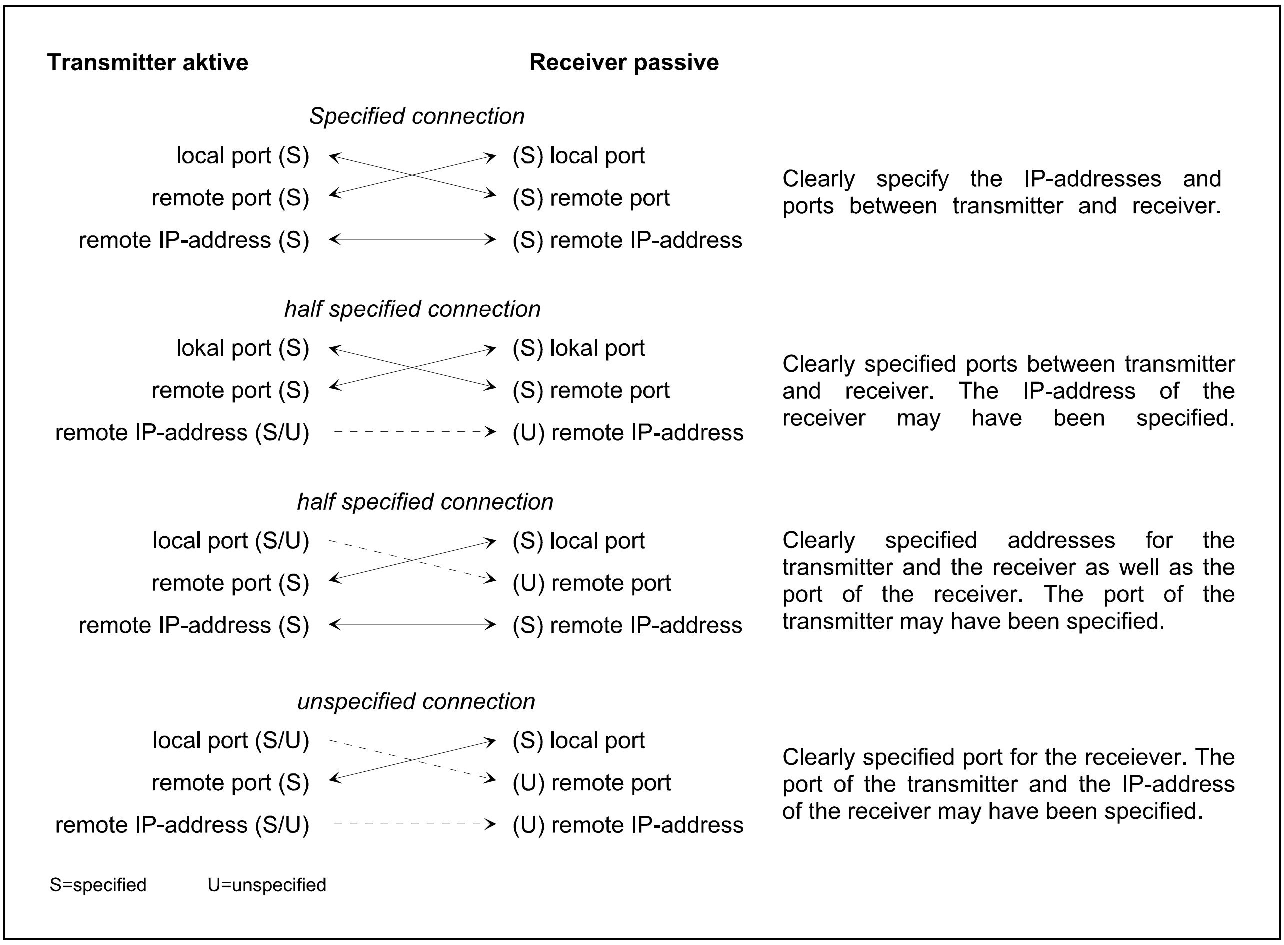
The IP-address is a 32-bit-address that must be unique within the network. The address consists of 4 numbers that are separated by a full stop.

Range: 000.000.000.000 to 255.255.255.255

You can obtain the IP-address from your system administrator.

You may enter an unspecified port address or IP address, i.e. the port or IP address are irrelevant when evaluating messages if the post address is o or the IP address is 0.0.0.0.

The following figure contains possible applications:



Clearly specify the IP-addresses and ports between transmitter and receiver.

Clearly specified ports between transmitter and receiver. The IP-address of the receiver may have been specified.

Clearly specified addresses for the transmitter and the receiver as well as the port of the receiver. The port of the transmitter may have been specified.

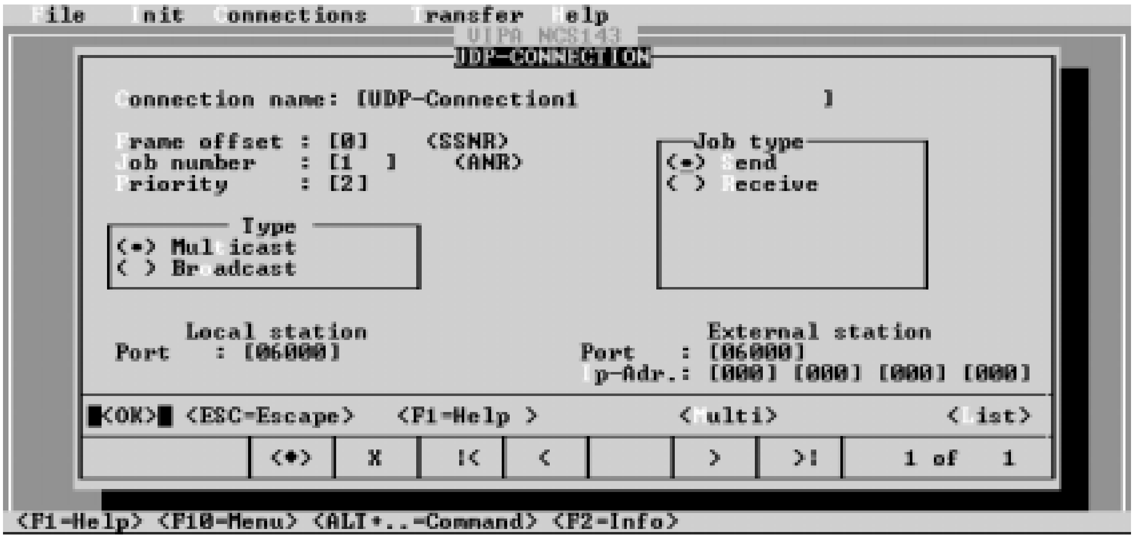
Clearly specified port for the receiever. The port of the transmitter and the IP-address of the receiver may have been specified.

Fig. 5-28: Unspecified configuration options for TCP/IP

5.11.2.2 UDP-Connection

The UDP protocol is a communication control service that does not require a connection and that is not always reliable. This protocol may be used to transfer data with one or more processors without first establishing a virtual connection. Individual datagrams are transferred to an external node but the reception of transmissions is not acknowledged.

(UDP may be used to transfer information that changes continuously. Should the transfer of one set of data not be successful, then the next set that is more up to date will be transferred after only a few seconds.



ile nit onnections ransfer elp
VIPA NCS143
UDP-CONNECTION

```
onnection name: [UDP-Connection1              ]

rame offset : [0]    <SSNR>
ob number   : [1  ]    <ANR>
riority     : [2]

        Type                              Job type
(•) Mul icast                             (•)  end
( ) Br adcast                             ( )  eceive

      Local station                                  External station
Port    : [06000]                         Port   : [06000]
                                           p-Adr.: [000] [000] [000] [000]
```

<OK> <ESC=Escape> <F1=Help > < ulti> < ist>
<•> X |< < > >| 1 of 1
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

Fig. 5-29: Interactive window UDP-Connection

Parameters for a UDP-Connection

Connection name

You may specify a name for the connection. The name may consist of a maximum of 32 characters.

Frame offset (SSNR-Offset)

The frame offset (SSNR-offset) determines the frame from the start address that is used as communication job by the current job. Frame offsets are normally only used in multi-processor systems with values larger than zero.

Please use the following values when assigning frame offsets:

- Frame offset 0 for CPU 1
- Frame offset 1 for CPU 2
- Frame offset 2 for CPU 3
- Frame offset 3 for CPU 4

Range: 0 to 3

Job number (ANR)

A job number must be allocated to each connection that should be used by Siemens handler modules. This number is used to access the connection from the PLC program. The job number together with the frame offset number determines the connection. This precludes multiple assignments. The job number and the respective frame offset number must be supplied in the control program.

Range: 1 to 199

Priority

The parameter specifies the priority of the message for the CP143 H1 / TCP/IP. UDP connections support priorities 1 and 2. Priority 1 is the highest priority.

When establishing a connection, priority 1 and 2 connections are so-called **static connections.** This means that the connection will be re-established, if possible, should it have been terminated (e.g. by the reset handler module).

Jobs with a priority of 1 differ from those with priority 2 in that the job is processed internally. The size of the user data for priority 1 jobs is limited to the pre-set block size. Data can not be blocked by means of ALL handler modules.

Range: 1 to 2

Job type

Here you specify the type of job. The following job types are available:

- Send transmission of unstructured data.
- Receive reception of unstructured data.

Type

Here you can specify who should receive the information

- Multicast are connections that are not directed at all participants but only those that have the same multicast circuit number. You specify the target group by means of the IP address.
  When you select multicast the multicast ID is transferred to the IP-address.
- Broadcast transmits a single message to all the participants that have been allocated the pre-set broadcast address or reception of single messages from participants that transmit via the preset broadcast address.

  The H1 broadcast address is 255.255.255.255.

Port address for local and for external stations

The port address determines the address of the connection that is used for the transfer of data. The port address for the local station may be compared to the local (own) TSAP on the H1-level. The port address of the local station consists of a decimal number (16-bit-address) of 5-digits. Both, TCP as well as UDP use port addresses for data transfers to the respective application level. Please note that certain ports in the range from 0 to 255 have been reserved for common services. The port address for the local station and the external station must be different.

Range: 0 to 65535

IP-address

The IP-address is a 32-bit-address that must be unique within the network. The address consists of 4 numbers that are separated by a full stop. This is also where multicast circuits are defined. Multicast circuit definitions may range from 224.0.0.0 to 239.255.255.255.

You can obtain the IP-address from your system administrator.

Range: 000.000.000.000 to 255.255.255.255

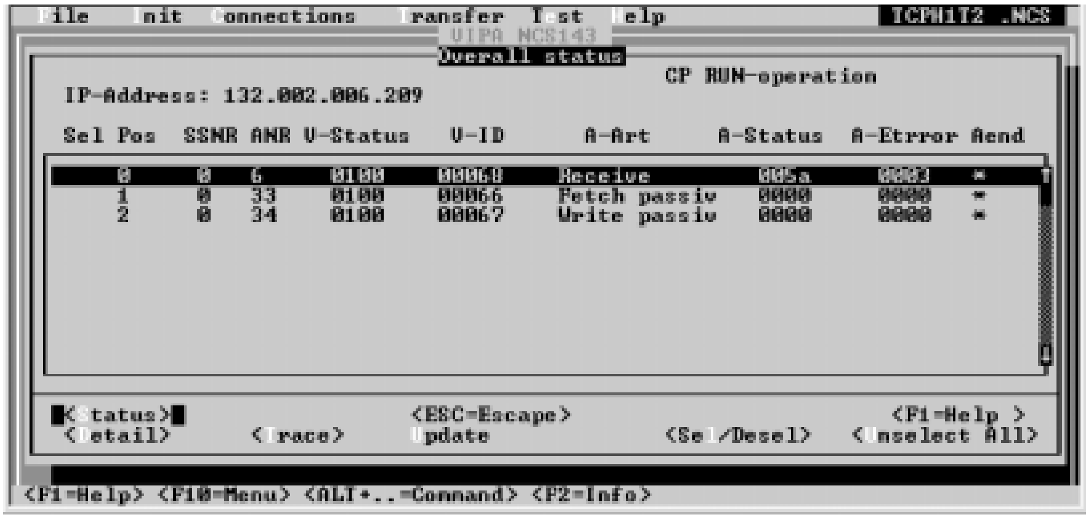
You may enter an unspecified port address or IP address, i.e. the port or IP address are irrelevant when evaluating messages if the post address is o or the IP address is 0.0.0.0.

Fig. 5-28 contains possible applications.

5.11.3 Test functions

The menu item **Test** provides access to test and diagnostic functions for your TCP or UDP connections.

When you select *TCP* or *UDP* the following dialog window appears. This contains the overall status for all TCP or UDP connections of the respective module:



```
  File    Init   Connections    Transfer  Test  Help                TCPH1T2 .NCS
                                         VIPA NCS143
                                        Overall status
                                                      CP RUN-operation
    IP-Address: 132.002.006.209

    Sel Pos  SSNR ANR V-Status    V-ID      A-Art     A-Status  A-Etrror Aend

            0     0   6    0100    00068    Receive          005a      0003    *
            1     0   33   0100    00066    Fetch passiv     0000      0000    *
            2     0   34   0100    00067    Write passiv     0000      0000    *

    <Status>                    <ESC=Escape>                          <F1=Help >
     <Detail>       <Trace>     Update              <Sel/Desel>   <Unselect All>

 <F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 5-30: Test function under TCP/IP

Once you have selected the display of the overall status you can select a connection and monitor the status of this connection in online mode by means of [Detail] or to display the current status of this connection by means of [Trace].

You may also mark specific connections by means of [Sel/Desel] to display the only the selected status by means of [Status].

Chapter 5.9.3 contains a detailed description of the dialog window.

5.11.4 OnNet-Kernel for TCP/IP of ftp

5.11.4.1 General

When NCS is to be used in conjunction with TCP/IP it is necessary to install a TCP/IP-Kernel. The kernel provides the basic functions of an operating system. The kernel has network functions as well as certain tools required for network operations.

A kernel is available from ftp. The OnNet-Kernel also provides the TCP/IP-Kernel that supports the TCP/IP-Stack.

The TCP-Kernel of the OnNet-Kernel has been implemented as a virtual device (VxD). This contains all the network functions required for DOS and for Windows.

The kernel uses the memory management functions of Windows.

A memory resident MS-DOS based version of the TCP/IP-stack is also available. This provides TCP/IP network functions for DOS as well as Windows. The installation of this driver will not be described here.

The following passages show the installation of the TCP/IP-Stacks supplied by ftp for Win 3.x and Win 95.

5.11.4.2 System requirements

- 386 Processor with 4MB
- At least 5MB free space on the hard disk
- Win 3.x or Win 95 operating system
- Network adapter with driver software

⚠

Some computer systems may not operate properly in conjunction with the ftp kernel under Windows 95 and return errors.

In this case you must please use the TCP/IP kernel provided by Microsoft (see chapter 5.11.5)

5.11.4.3 Checks before installation of the OnNet-Kernel

- The system requirements must be satisfied.
- Please check that the CP1413 TCP/IP-adapter is installed in your PC.
- Before the OnNet-Kernel is installed, the driver for the network adapter must be installed.
- Enter the data required for the installation into the table below.

5.11.4.4 Data required for the installation

The required data is usually available from your system administrator.

General Data

Serial Number

Authentication Key

Data for the network adapter

for Windows 3.x

Name of Network Card

Interrupt (IRQ)

Base I/O Port (hex.)

Base Memory Address (hex.)

for Windows 95

Network Type (SLIP or PPP, DIX-Ethernet,
IEEE Ethernet or IEEE Token Ring)

Data for the Network

Internet (IP) Address of your PC

Subnet Mask

IP Address of Default Router (maximum 3)

Hostname of your PC

Domain Name of your Host

Type of Name Resolution (DNS or NIS)

IP Address of Name Server(s) (maximum 3)

Your Username

Time Zone of your Geographic Area

5.11.4.5 OnNet kernel installation

The data you have noted down above will be required during the installation. You may obtain more information on the required items by clicking on <Help>.

The installation program will only request the data required by your system.

1. Insert installation diskette (Disk 1) into drive A:. Start the Setup-program

   Win 3.x: In Windows go to *Run* from the pull-down menu **File**. Enter the following: `A:\SETUP` and click on <OK>.

   Win 95: From the START-menu select *RUN*. Enter the following: `A:\SETUP` and click on <OK>.

   The installation program is started and a dialogue box will be opened. Here you can access the latest information on the software or you may continue with the installation by means of <CONTINUE>.

2. A window showing the licensing conditions will be displayed. Click on <ACCEPT> to continue with the installation. Click on <EXIT> if you wish to abort the installation.

3. The program will now suggest the "Express"-installation. Please click on <CONTINUE> to execute the "Express"-installation.

4. The next dialogue box requests the "Serial Number" and "Authentication Key" from you.

5. The next dialogue box shows a path for the installation of the software. You may change this path as required. Click on <CONTINUE> to accept the path and continue with the installation.

6. The contents of the next dialog window depends on your system.

   Win 3.x: The dialogue box lists all the drivers located on the diskette. Click on <CONTINUE> to confirm the drivers selection.

   Win 95: If the Microsoft TCP/IP-stack was installed on your system the window will indicate this. Select *FTP Software stack* from this window.

7. Various windows of the installation program will be displayed to request all the information you have noted down on the sheet above:

   Network Type, IP Configuration, Name Server Configuration, Username Configuration and Time Zone Configuration

8. Click on <CONTINUE> in the "Copy File" dialogue box. Now the program will request for the respective diskette.

9. When the program asks whether the AUTOEXEC.BAT and CONFIG.SYS files should be changed click on <OK> and then on <RESTART SYSTEM>.

This concludes the installation and your system will be restarted.

The module is delivered with a TCP/IP test program. This is described below.

5.11.5 Microsoft kernel for TCP/IP under WINDOWS 95/NT

5.11.5.1 General

When NCS is to be used in conjunction with TCP/IP under Windows 95 or Windows NT it is necessary to install the TCP/IP-Kernel supplied by Microsoft.

These paragraphs describe the installation of the Microsoft kernel for Windows 95 or Windows NT V4.0.

5.11.5.2 Installation

From the START menu click on *Settings* and *Control panel*. Double-click on "Network" in the control panel window.

Select the "Protocol" tab to start the installation of the TCP/IP protocol and click on [Add]. You will see a list containing all available protocols. Select the "TCP/IP protocol" from the list.

The TCP/IP kernel will be installed. Once the installation has been completed you may provide the various settings required. Please specify which PC-board should be used by the protocol.

For online configurations the subnet mask must be set to 0.0.0.0.

When connecting with WINDOWS NT the remote or the local port of the VIPA CP143 must be set to 0. If this is not done it could take up to 30s before a new connection is established after an existing connection has been terminated!

5.11.6 Test program for TCP/IP-Connections

The name of the test program is TCPEVAL.EXE. This program is only compatible with Windows 95.

You may use so-called PING-applications for tests under Windows 3.x. These applications can be used to send a PING to a host computer and this will answer if the installation is OK.

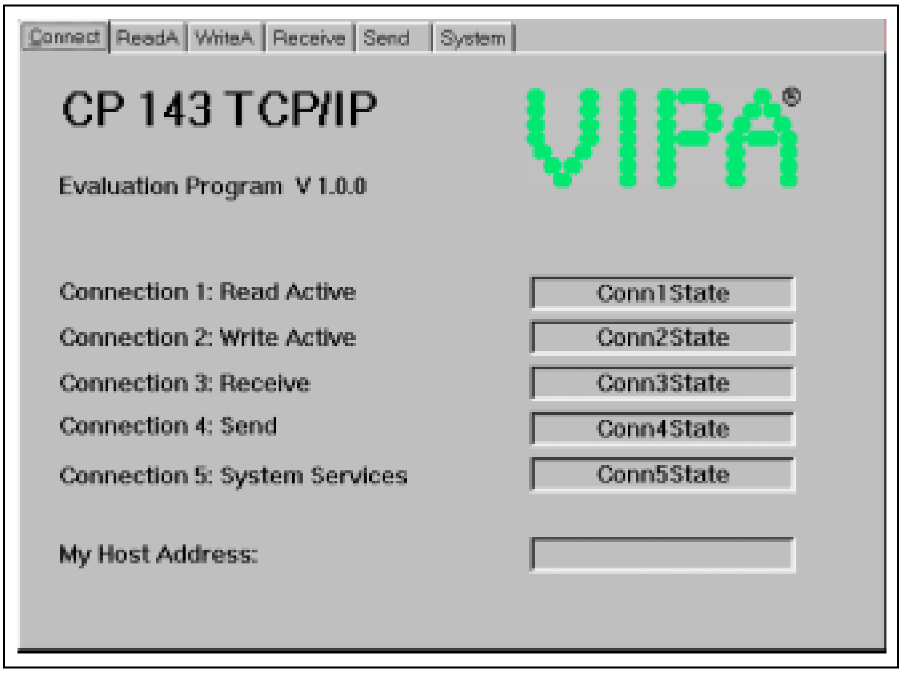
Start the test program for Win 95:

Insert the supplied diskette (SSN-SW893, Disk 1) into drive A:.

Go to the START-menu and select *RUN*.

Enter the following: `A:\TCPEVAL` and click on <OK>.

The test program starts and displays the following window:



CP 143 TCP/IP

Evaluation Program V 1.0.0

Connection 1: Read Active Conn1State

Connection 2: Write Active Conn2State

Connection 3: Receive Conn3State

Connection 4: Send Conn4State

Connection 5: System Services Conn5State

My Host Address:

Fig. 5-31: Test program "Connect"

The menu has the form of pages in a file. You can bring any page to the foreground by clicking on the tab with the left mouse key.

File tabs

Connect Contains the status indicators for the connection and your own IP-address.

ReadA Configuration of a READ AKTIV-connection (FETCH).

WriteA Configuration of a WRITE AKTIV-connection.

Receive Configuration of a RECEIVE-job.

Send Configuration of a SEND-job.

System Control window for requesting the status and to change RUN/STOP on the CP.

Every one of the above pages has an additional menu. This is accessible by means of the right mouse button.

The additional menu contains the following options:

Menu provided by the right mouse key

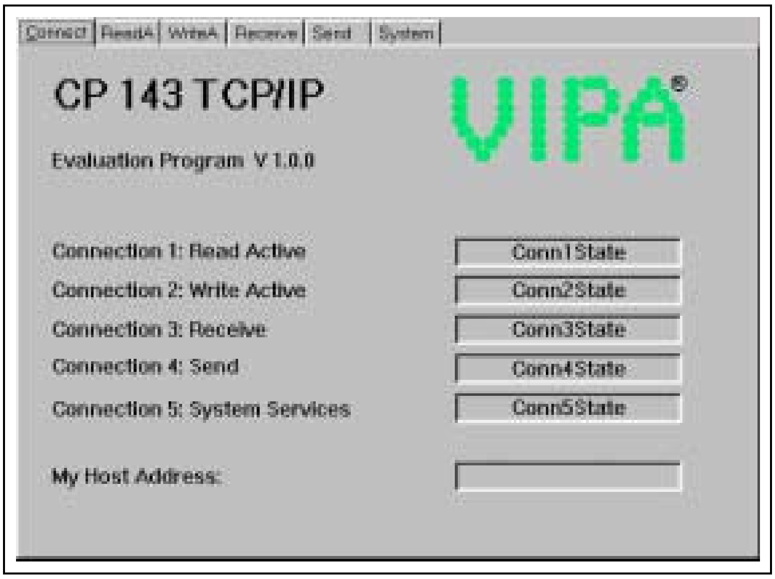
Save All save all parameters.

Save Conn 1
to save the respective connection.
Save Conn5

Save Win Pos save the current window position.

Show Hints displays a help summary for any input field or button.

5.11.6.1 Connect-tab



CP 143 TCP/IP

Evaluation Program V 1.0.0

Connection 1: Read Active Conn1State

Connection 2: Write Active Conn2State

Connection 3: Receive Conn3State

Connection 4: Send Conn4State

Connection 5: System Services Conn5State

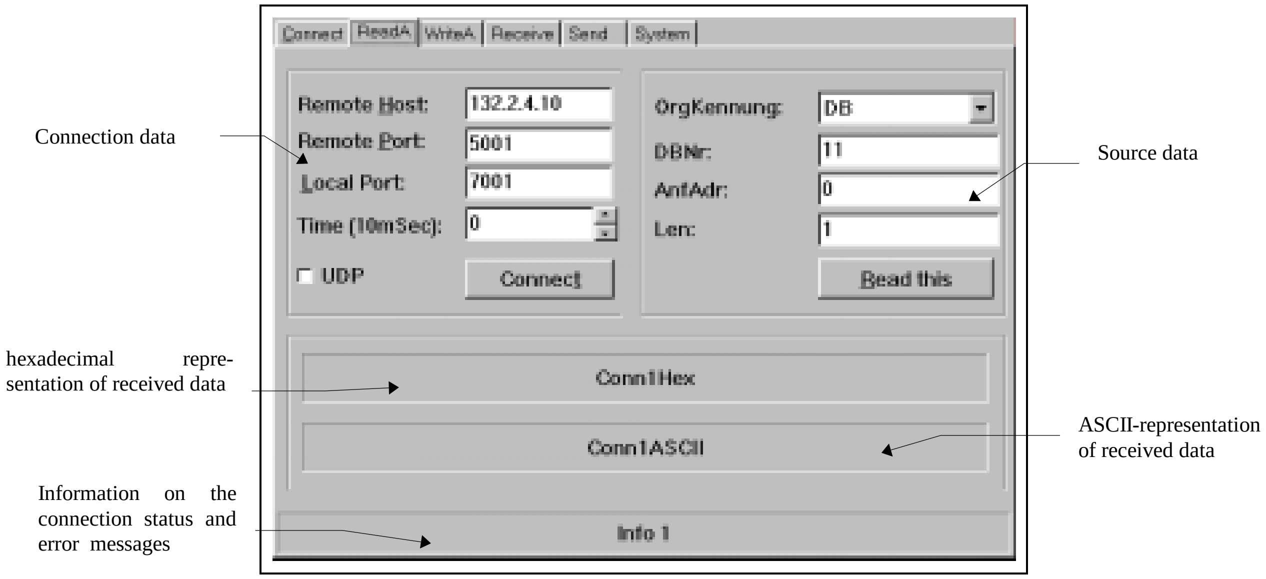
My Host Address:

Fig. 5-32: Test program "Connect"

This window displays the status of all the connections that may be configured by means of the program. You can see immediately which connections are OK and which connections are suspect. This window displays status changes of registers as they occur.

The window also shows your own IP address for reference purposes.

5.11.6.2 ReadA-tab



Remote Host: 132.2.4.10
Remote Port: 5001
Local Port: 7001
Time (10mSec): 0
□ UDP

OrgKennung: DB
DBNr: 11
AnfAdr: 0
Len: 1

Connection data

Source data

hexadecimal representation of received data

ASCII-representation of received data

Information on the connection status and error messages

Fig. 5-33: Test program "Read A"

This window may be used to configure an active read connection.

You must specify the data required for establishing the connection and the source from where the data should be read.

Input fields

Remote Host	IP-Address of the station from where the data must be read.
Remote Port	Connection address of the external station.
Local Port	Connection address of the local station. To simplify matters, you may specify the same address for remote and local.
Time (10mSec)	Timer interval for cyclic reads.
OrgKennung	Type of source module
DBNo	Number of the source module
AnfAdr	Start address of the source module
Len	Word length of the source module

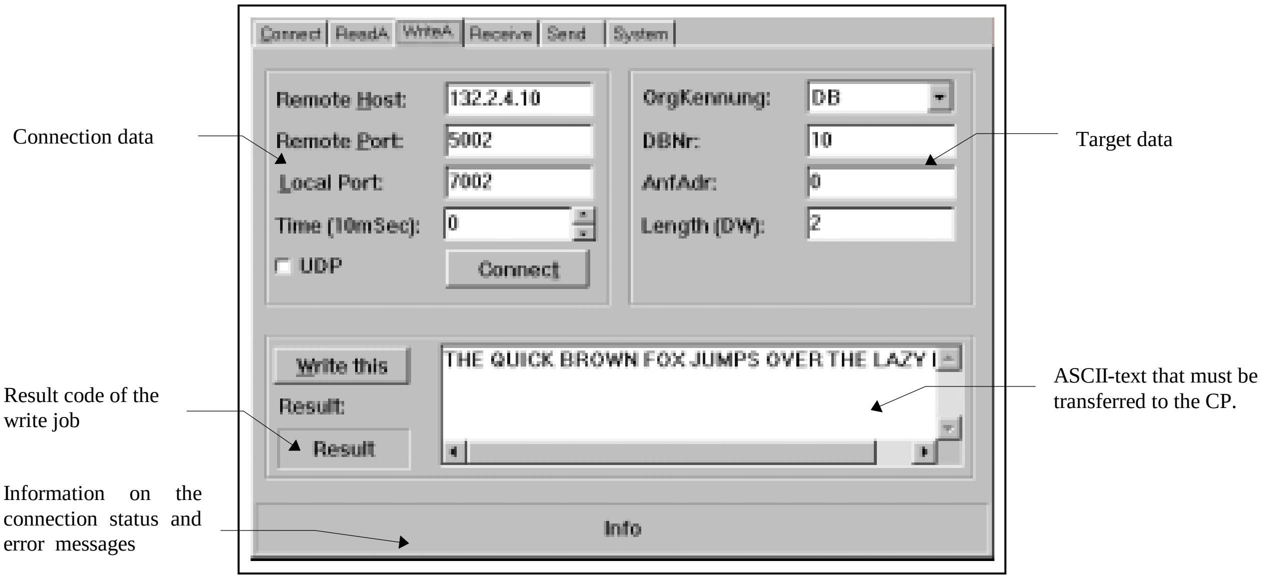
Control box

UDP Here you may select an unsecured protocol for communicating. Unsecured communications do not use virtual connections. In this manner you may select to display only UDP-messages.

Buttons

Connect	The connection is established and prepared for the read command.
Read this	The requested data is read via the specified connection.

5.11.6.3 WriteA-tab



Remote Host: 132.2.4.10
Remote Port: 5002
Local Port: 7002
Time (10mSec): 0
□ UDP
OrgKennung: DB
DBNr: 10
AnfAdr: 0
Length (DW): 2
Result:

THE QUICK BROWN FOX JUMPS OVER THE LAZY I

Connection data

Target data

Result code of the write job

ASCII-text that must be transferred to the CP.

Information on the connection status and error messages

Fig. 5-34: Test program "WriteA" tab

This is where you activate an active write connection

In the same manner as specified for the READ AKTIV-command you must enter the data required to establish the connection as well as the target module for the data.

Input fields

Remote Host	IP-Address of the station to which the data must be written.
Remote Port	Connection address of the external station.
Local Port	Connection address of the local station. To simplify matters, you may specify the same address for remote and local.
Time (10mSec)	Timer interval for cyclic writes. For cyclic writes the value of the timer must be at least 5.
OrgKennung	Type of target module
DBNo	Number of the target module
AnfAdr	Start address of the target module
Len	Word length of the target module

Control box

UDP	Here you may select an unsecured protocol for communicating. Unsecured communications do not use virtual connections. In this manner you may select to display only UDP-messages.

Buttons

Connect	The connection is established and prepared for the write command.
Write this	The data entered into the ASCII fields is written to the CP via the connection established by the *Connect*.

5.11.6.4 Receive-Register

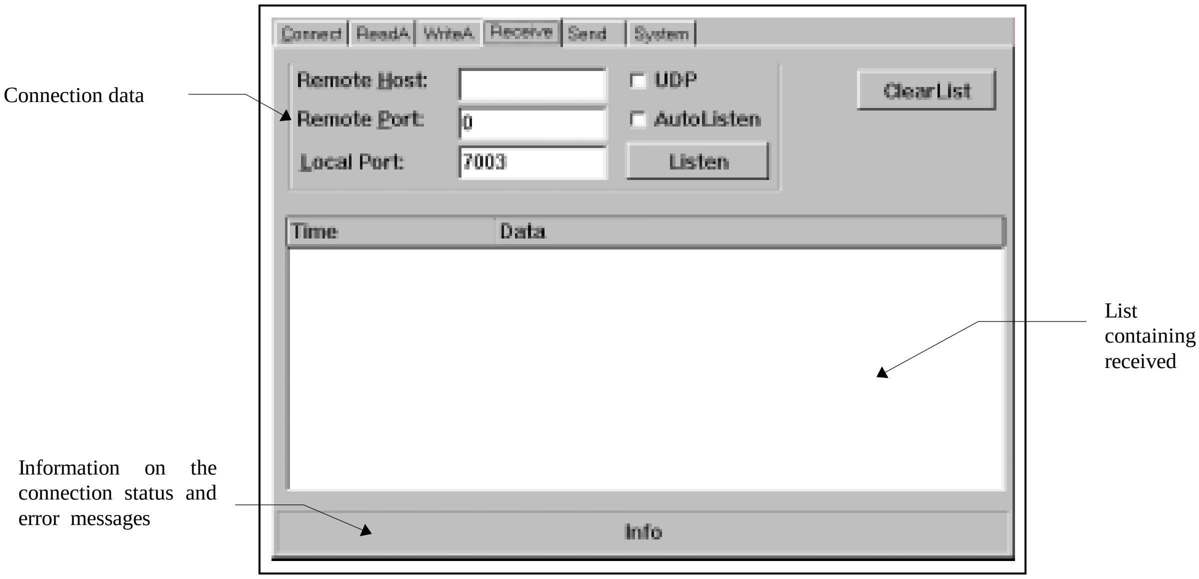


Fig. 5-35: Test program "Receive"

In this dialogue window you can configure the reception of messages from a specific host processor.

Input fields

Remote Host IP-Address of the station to which the data must be written.

Remote Port Connection address of the external station.

Local Port Connection address of the local station. To simplify matters, you may specify the same address for remote and local.

Control box

UDP Here you may select an unsecured protocol for communicating. Unsecured communications do not use virtual connections. In this manner you may select to display only UDP-messages.

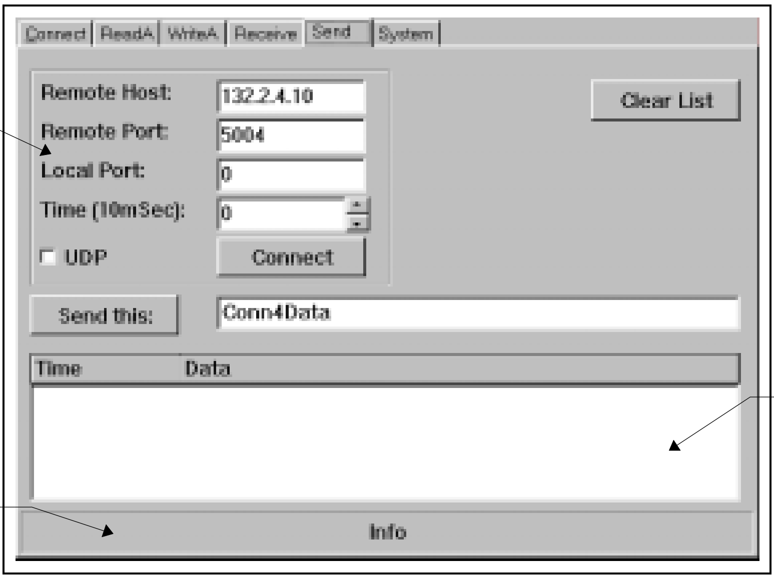
AutoListen When you select "AutoListen" the program goes to receive mode. In this mode every received message from the remote CP is displayed in the list. Disconnects will be detected and displayed but the program remains in receive mode. The program will continue listing messages when the connection is re-established.

Buttons

Listen Any received messages will be listed. The listing is stopped when you click on the "STOP" button or when the connection is terminated. You may also stop the listing by entering new connection parameters.

ClearList All entries are deleted from the receive list and the listing will continue from the first line of the list.

5.11.6.5 Send-tab



Remote Host: 132.2.4.10

Remote Port: 5004

Local Port: 0

Time (10mSec): 0

☐ UDP

Send this: Conn4Data

Connection data

List containing received

Information on the connection status and error messages

Fig. 5-36: Test program "Send"

You may use this window to send a message to a specific host processor.

Input fields

Remote Host IP-Address of the station to which the data must be written.

Remote Port Connection address of the external station.

Local Port Connection address of the local station. To simplify matters, you may specify the same address for remote and local.

Time (10mSec) Timer interval for cyclic writes. For cyclic writes the value of the timer must be at least 5.

Control box

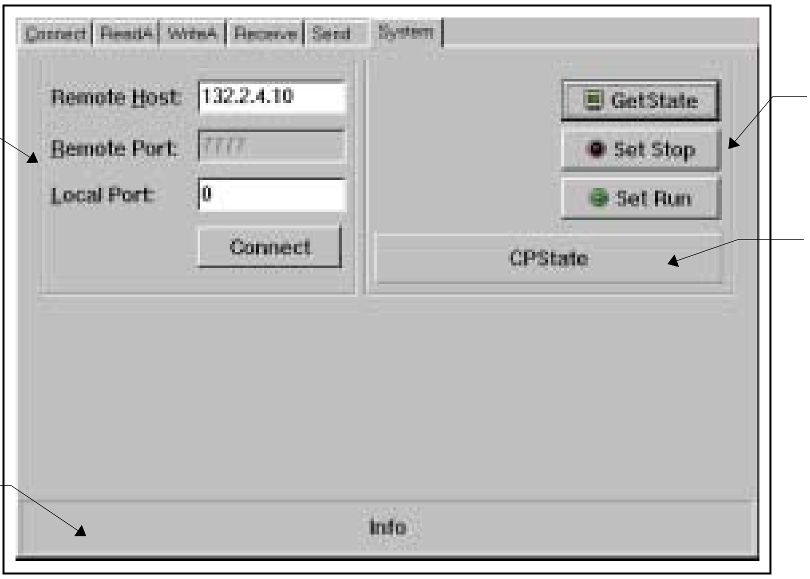
UDP Here you may select an unsecured protocol for communicating. Unsecured communications do not use virtual connections. In this manner you may select to display only UDP-messages.

Buttons

Connect The connection is established and prepared for the write command.

Send this The data entered into the ASCII fields is written to the CP via the connection established by the *Connect*.

5.11.6.6 System-tab



Remote Host: 132.2.4.10

Remote Port: ????

Local Port: 0

Connection data

Host-CP contol panel:
...Status check
...in Stop
...in Run

Status display requested with GetState

Information on the connection status and error messages

Fig. 5-37: Test program "System"

This window returns information about the selected host-CP.

Input fields

Remote Host IP-Address of the station to which the data must be written.

Remote Port Connection address of the external station.

Local Port Connection address of the local station. To simplify matters, you may specify the same address for remote and local.

Buttons

Connect The connection is established and prepared for the write command.

GetState The status of the CP is returned via the connection that was established with the *Connect*. The following may be displayed:

- Hardware-Stop (Run/Stop-switch on the CP is in Stop-position) The CP can not be controlled remotely by the test program.
- Hardware-Run (Run/Stop-switch on the CP is in Run-position) The test program is able to control the CP remotely.
- Software-Stop (Run/Stop-switch on the CP must be in Run-position) The CP was stopped by a *SetStop* command.
- Software-Run (Run/Stop-switch on the CP must be in Run-position) The CP was started by a *SetRun* commend.

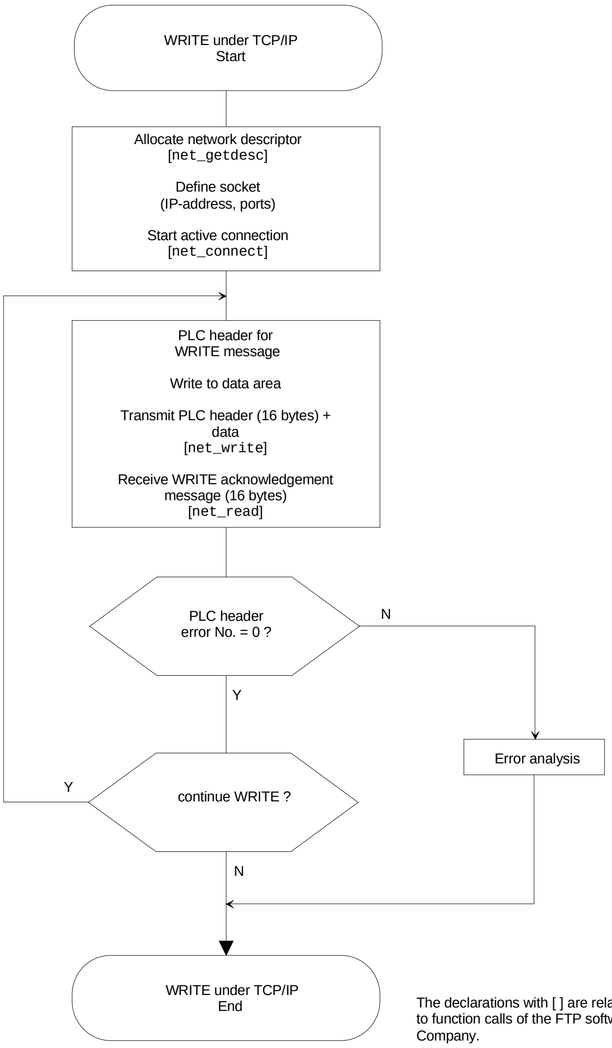
SetStop The CP is stopped. This function is only available when the Run/Stop-switch on the CP is in the Run-position.

SetRun The CP is started. This function is only available when the Run/Stop-switch on the CP is in the Run-position.

5.11.7 Accessing the PLC from the PC

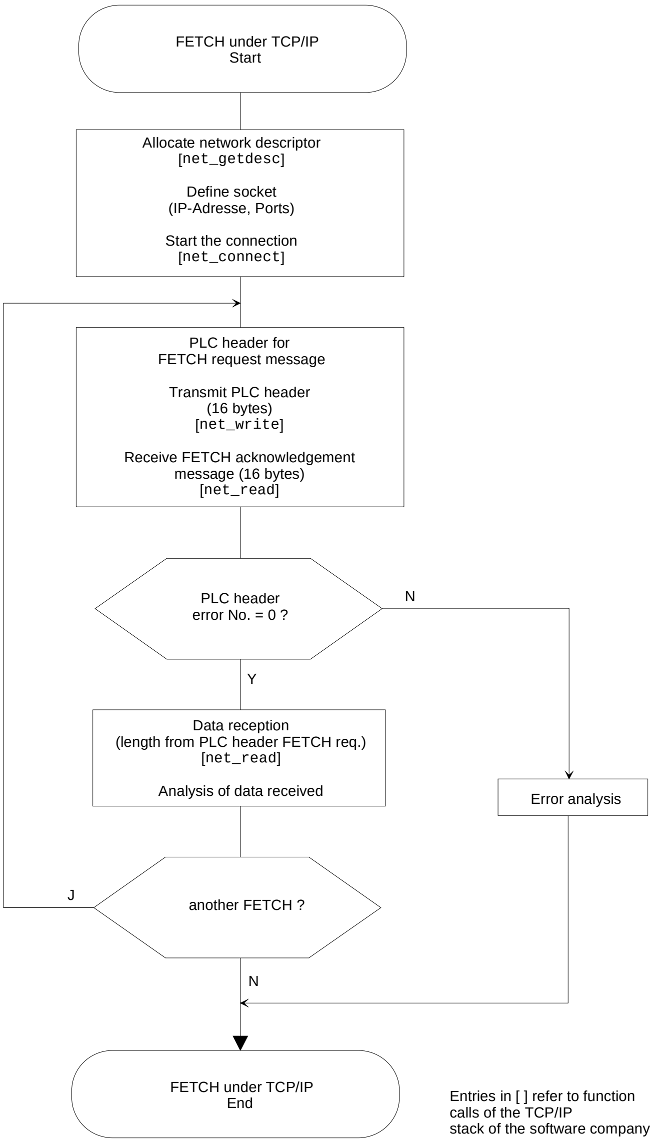
The following flowcharts are intended as a guideline for programming WRITE and FETCH jobs. Entries in [] refer to function calls to the TCP/IP stack provided by FTP.

WRITE job



The declarations with [ ] are related to function calls of the FTP software Company.

FETCH job



Entries in [ ] refer to function calls of the TCP/IP stack of the software company FTP

6 Commissioning

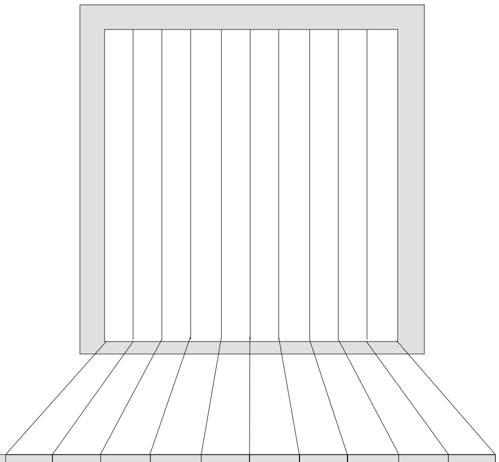
6 Commissioning

6.1 Plug-in slots in the PLC

No configuration settings are required before the module is installed.

The following diagrams indicate possible slots (indicated by means of the grey squares) where the CP143 H1 / TCP/IP may be installed into the different PLC frames.

6.1.1 Plug-in slots in the AG-115U



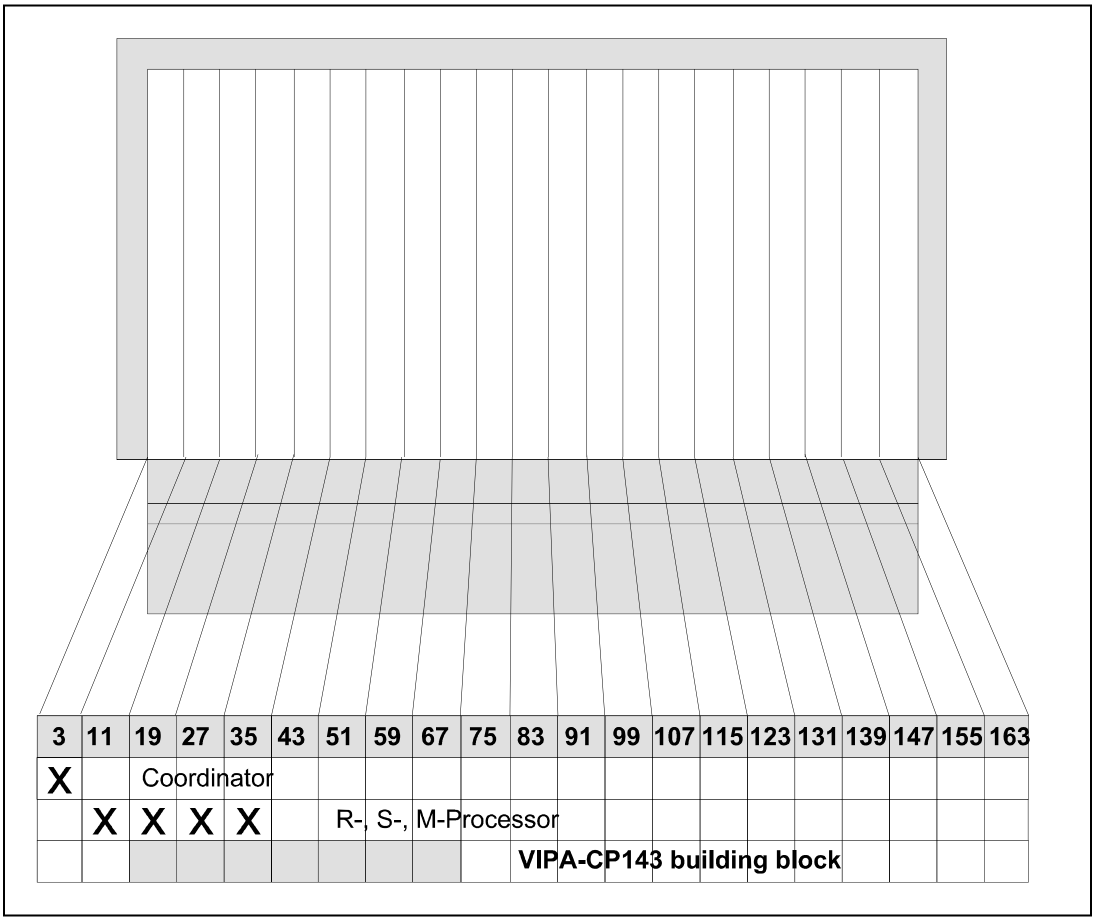
Mounting places	PS	CPU	0	1	2	3	4	5	6	IM
Power supply building block										
Central building block										
Building block frame CR700-1			▨							
Building block frame CR700-2			▨	▨	▨	▨	▨	▨		
Building block frame CR700-3			▨	▨	▨	▨	▨	▨		
Building block frame CR700-0LA			▨							
Building block frame CR700-0LB			▨	▨	▨	▨				
Building block frame ER701-0										
Building block frame ER701-1										
Building block frame ER701-2										
Building block frame ER701-3			▨ ①	▨ ①	▨ ①	▨ ①	▨ ①	▨ ①	▨ ①	

Fig. 6-1: Plug-in slots in the AG-115U

The grey squares indicate the locations where the CP143 H1 / TCP/IP adapter may be installed.

① with the IM 304 / 314 interface

6.1.2 Plug-in slots in the AG-135U

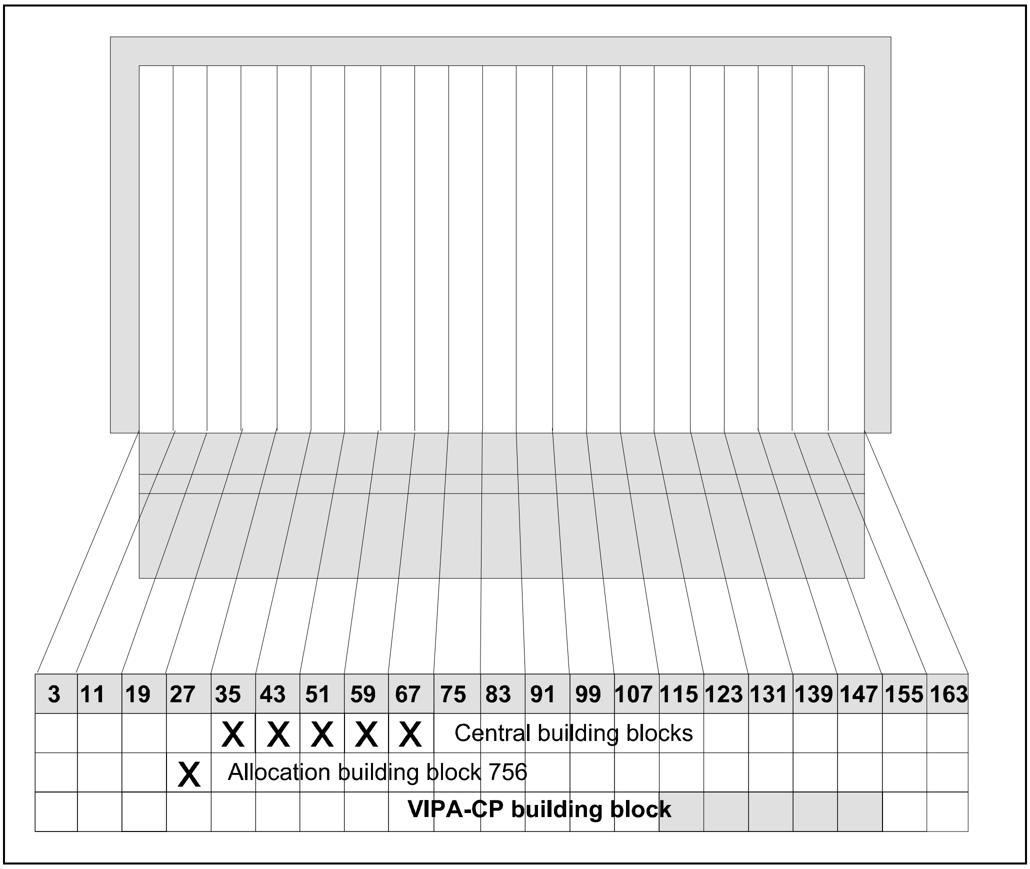


3	11	19	27	35	43	51	59	67	75	83	91	99	107	115	123	131	139	147	155	163
X		Coordinator																		
	X	X	X	X		R-, S-, M-Processor														
		▨	▨	▨	▨	▨	▨	▨		VIPA-CP143 building block										

Fig. 6-2: Plug-in slots in the AG-135U

The grey squares indicate the locations where the CP143 H1 / TCP/IP adapter may be installed.

6.1.3 Plug-in slots in the AG-150U

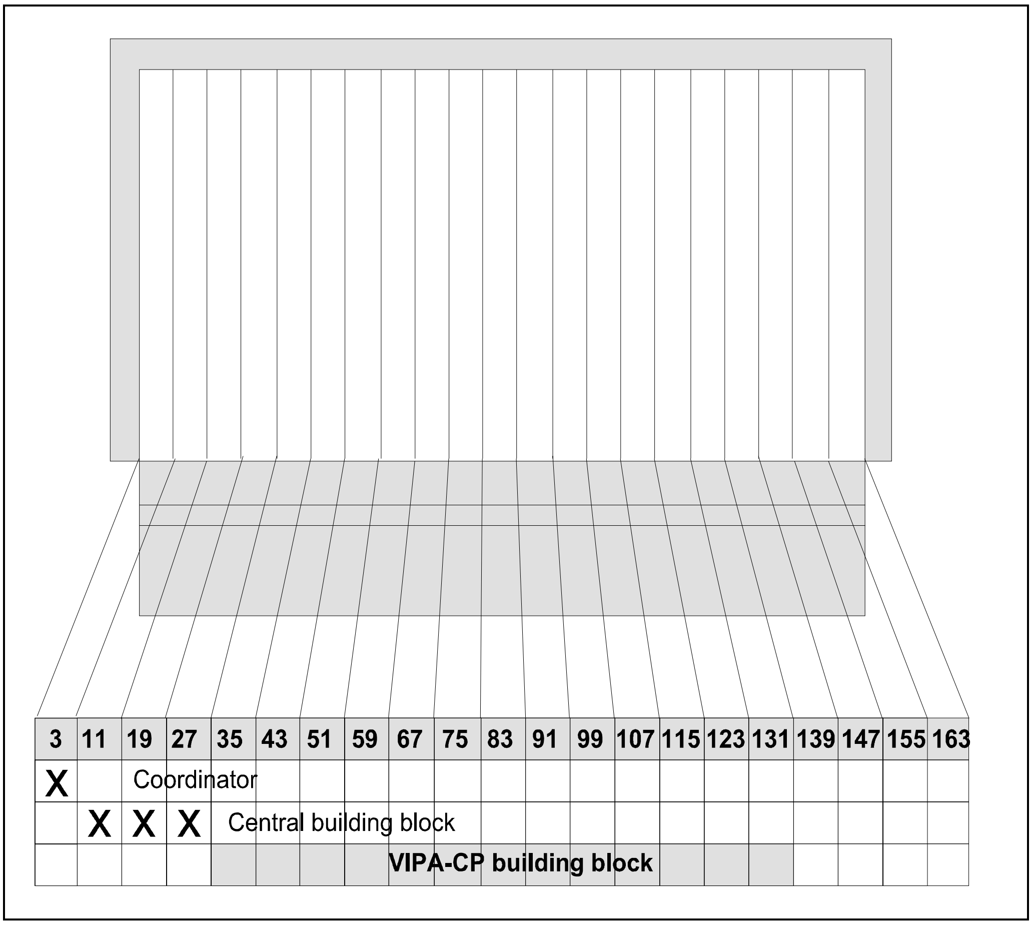


3	11	19	27	35	43	51	59	67	75	83	91	99	107	115	123	131	139	147	155	163
				X	X	X	X	X	Central building blocks											
			X	Allocation building block 756																
								VIPA-CP building block						▨	▨	▨	▨	▨		

Fig. 6-3: Plug-in slots in the AG-150U

The grey squares indicate the locations where the CP143 H1 / TCP/IP adapter may be installed.

6.1.4 Plug-in slots in the AG-155U



3	11	19	27	35	43	51	59	67	75	83	91	99	107	115	123	131	139	147	155	163
X		Coordinator																		
	X	X	X	Central building block																
				▨	▨	▨	▨	▨ VIPA-CP building block	▨	▨	▨	▨	▨	▨	▨	▨				

Fig. 6-4: Plug-in slots in the AG-155U

The grey squares indicate the locations where the CP143 H1 / TCP/IP adapter may be installed.

6.1.5 Plug-in slots in the EG-185U

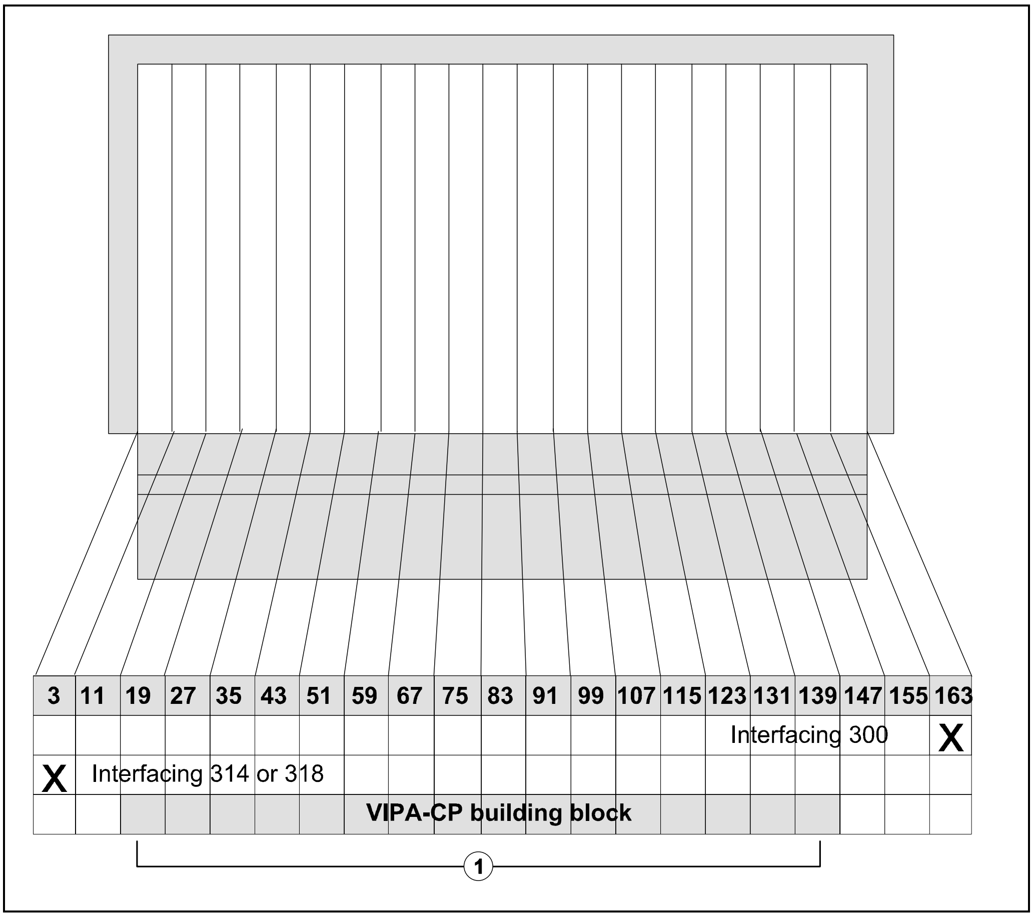
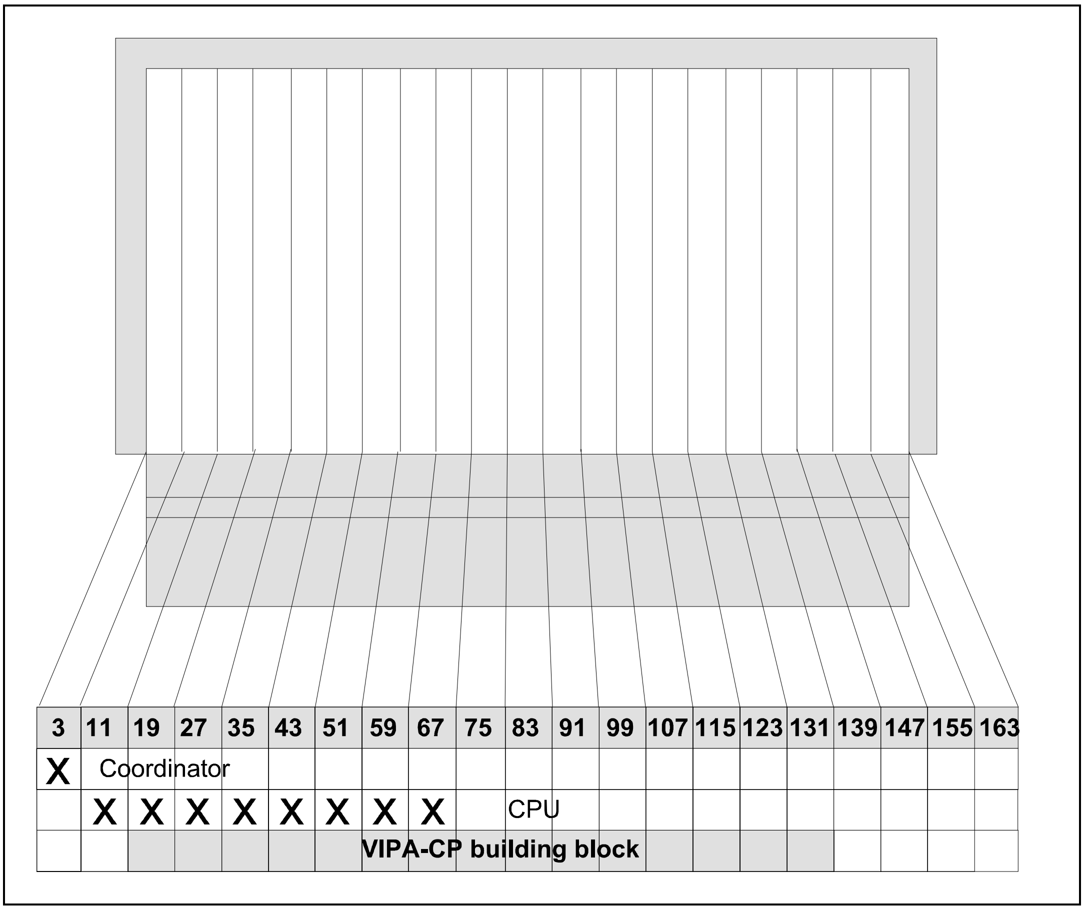


Fig. 6-5: Plug-in slots in the EG-185U

The grey squares indicate the locations where the CP143 H1 / TCP/IP adapter may be installed.

① 304/314 interface

6.1.6 Plug-in slots in the AG-188U



3	11	19	27	35	43	51	59	67	75	83	91	99	107	115	123	131	139	147	155	163
X	Coordinator																			
	X	X	X	X	X	X	X	X		CPU										
		▨	▨	▨	▨	▨	▨ VIPA-CP building block	▨	▨	▨	▨	▨	▨	▨	▨	▨				

Fig. 6-6: Plug-in slots in the AG-188U

The grey squares indicate the locations where the CP143 H1 / TCP/IP adapter may be installed.

6.2 Start-up properties

When power is turned on the CP143 H1 / TCP/IP executes the BIOS routines (hardware initialization, memory test, initialization of drivers).

Once the CP-job is started the CPs program control is initiated. The frames of the DPR are only controlled from this point onwards, i.e. after the CP-job has booted up (app. 10 seconds after power on). At this time the unit waits for the synchronization from the PLC. Data communications with the PLC is inhibited in this state (IDLE; RUN and STOP-LED LEDs are on) and it is only enabled again after synchronization has been completed.

- For this reason the requirements of balanced synchronization characteristics (a HTB synchronization module waits between 8 and 10 seconds, depending on the type of CPU) can not be met. The start-up OBs 20-22 must therefore perform the following actions:
    - poll the PAFE of the HTB in synchronous mode until this == 0. After 10 seconds the frame will be set to active by the CP.
    - If the synchronous HTB still detects a PAFE != 0 after 15 seconds then the PLC, the configuration of the PLC or the CP is incorrect.
    - In this case the respective configuration must be checked.

Depending on the sequence in which the synchronous HTBs were initiated, the CP may detect a synchronous request and re-boot when more than one frame of a CP must be synchronized. Under these conditions the CP would never be synchronized. Cure: switch the CP to STOP by means of the respective switch, then synchronize all the required frames. Change the switch back to RUN. At this point the CP has been synchronized. This reaction of the system that can only occur during commissioning, and then only, if the number of frames of the CP is increased and the new synchronous HTB is not accessed as the last one in the sequence.

6.2.1 Status after the CP has started

The CP143 H1 / TCP/IP executes a cold/warm start when the status changes from STOP to RUN and from RUN via STOP back to RUN. Any active connection is terminated and re-established after the CP-job has booted.

These status changes can have four possible reasons

- re-sychronization of a CP that is already in synchronous mode by means of the synchronous HTBs of the CPU (warm start),
- STOP/START switch (cold start),
- STOP/START function of the configuration tool NCS (warm start),
- reset-all (warm start)

Depending on the START-/STOP switch, the CP143 H1 / TCP/IP branches to one of the following:

Before synchronization

- to the STOP state, if the switch is in the STOP position, or
- to the IDLE state, if the switch is in the RUN position.

After synchronization

- to the STOP state, if the switch is in the STOP position, or
- to the RUN state, if the switch is in the RUN position.

6.2.2 Status indicators

CP status	Indication	Activity
Start-up phase	RUN LED off STOP LED on COM LED off	Hardware test
Lack of resources	COM LED blinking RUN/STOP LED indicates the CPs -status.	Check the parameter file, start the job, initialize the interrupt handler, etc.
IDLE (not synchronous)	RUN LED on STOP LED on COM LED off	Waiting for synchronization request from PLC. Productive communication via the frames is inhibited.
RUN	RUN LED on STOP LED off COM LED off	PLC synchronization request was executed for every configured frame. Productive communication via the frames is enabled.
STOP	RUN LED off STOP LED on COM LED off	Productive communication is inhibited although the CP may be synchronized with the PLC. At this time a new data base may be loaded.

Tab. 6-1: Status indicators

6.3 PLC programming

6.3.1 General

To execute H1 jobs or TCP/IP jobs you require the standard handler modules supplied by Siemens. The description and documentation of these standard handler modules is also available from Siemens.

The following table lists the handler modules required for the PLC.

Handler module	Description
SYNCHRON	The SYNCHRON module synchronizes the PLC and the CP during the start-up phase. All the frames are erased and the block-size for PLC to CP communications is determined. Active data communications can only occur via synchronized frames.
SEND	Send a job to the CP
SEND-ALL	Initiate the transfer of data between the PLC and the CP
RECEIVE	Receive a job from the CP
RECEIVE-ALL	Initiate the transfer of data between the PLC and the CP
FETCH	The FETCH module is used to obtain data. A fetch-HTB must be accompanied by the RW identifier and it initiates the READ job.
RESET	The RESET module is used to reset a job for the specified connection.
RESET-ALL	RESET-ALL (ANR=0) forces a system reset of the CP.
CONTROL	The CONTROL module is used to request the status of a job, i.e. the ANZW (indicator word) of the specified job is brought up to date.

Tab. 6-2: Handler modules required for the PLC

Handler modules are supplied by special function modules of the PLC. The following table contains the respective allocation:

HTB	CPU type	
	CPU 941(B) to 944(B) and CPU 945	CPU 922, 928, 928B and CPU 948
SEND	FB244	FB120
SEND-ALL	FB244 with ANR = 0	FB126
RECEIVE	FB245	FB121
RECEIVE-ALL	FB245 with ANR = 0	FB127
FETCH	FB246	FB122
CONTROL	FB247	FB123
RESET	FB248	FB124
SYNCHRON	FB249	FB125

Tab. 6-3: HTB numbers for the different types of CPU

6.3.2 Programming

With the exception of FB125, which must be replaced by FB249 for the CPUs of type 115, the FB22 frame applies to all types of PLC.

OB22

```
      BIB   #18075
;                     Restart after power on
00000      :
00002      :SPA FB 22                          Delay time and synchron.
      NAME #POWERON
00009      :BE
```

FB22

```
      BSTNAME #POWERON
      BIB     #2085
0000E      :
00010      :L   KT 20.2                        max. Delay time  20s
00014      :U   M 0.0
00016      :UN  M 0.0
00018      :SV  T 43                           VKE 0 for neg. edge
0001A      :O   M 0.0
0001C      :ON  M 0.0
0001E      :SV  T 43                           VKE 1 to start the timer
00020      :
00022 SYN  :SPA FB 125                         Synchron-module
      NAME #SYNCHRON
      SSNR =KY 0,0                             SSNR or frame base 0
      BLGR =KY 0,6                             Block size 6 (512 bytes)
      PAFE =MB 199                             Configuration error byte MB199
0002C      :
0002E      :UN  M 199.0                        end if no error occurred
00030      :BEB
00032      :
00034      :U   T 43                           as long as timer is active
00036      :SPB =SYN                           attempt synchronization
00038      :L   KH 2222                        load error ID into accumulator
0003C      :STP                                stop the PLC
0003E      :BE
```

The CP143 H1 / TCP/IP requires app. 10 s to start after power is turned on. You will receive an error message via the configuration error byte if the PLC attempts to synchronize during this period. This message is removed as soon as the CP has completed the start-up procedure.

Initially the timer of this module is set to 20s. If the units are unable to synchronize properly during this time all processing will be stopped.

6.4 Example

6.4.1 Overview and conditions

This chapter is intended to introduce the first-time user of the CP143 H1 / TCP/IP to the H1 LAN and TCP/IP for the PLC. The emphasis will be on the use of the configuration software for the CP143 H1 / TCP/IP.

The chapter describes simple communication system that allows control of the communication processes.

You need to have some knowledge of the CP handler modules. These CP handlers are standard function modules. Programs on the PLCs provide the required access to communication functions.

You should have the following equipment to execute the examples:

- 2 PLCs
- 2 communication processors CP143plus (BG81A) or CP143 TCP/IP (BG89A).
- Communication link consisting of
  - 1 LAN cable with 2 coaxial T-pieces.
  - 2 Terminators
- Software package
  - Configuration software SSN-SW893
  - MC5 Programming package for programming of PLCs
  - Standard handler modules for the respective PLCs.

The basis for this problem is described in chapter 6.4.2. To implement the example it is necessary to program the PLCs in the STEP®5 language from Siemens and to configure the communication processors CP143plus and CP143 TCP/IP. Chapter 6.4.3 contains the respective programs for the AG135U or AG155U PLCs.

The results obtained from the communication processes are described in chapter 6.4.6.

6.4.2 Problem

This introductory example for the use of the H1 and the TCP/IP protocols is based upon the communication problem described below.

Fig. 6-7 and Fig. 6-8 show the required system as well as the components mentioned in chapter 6.4.1.

The structure of the system depends on the respective protocol.

6.4.2.1 System structure required for the H1 protocol

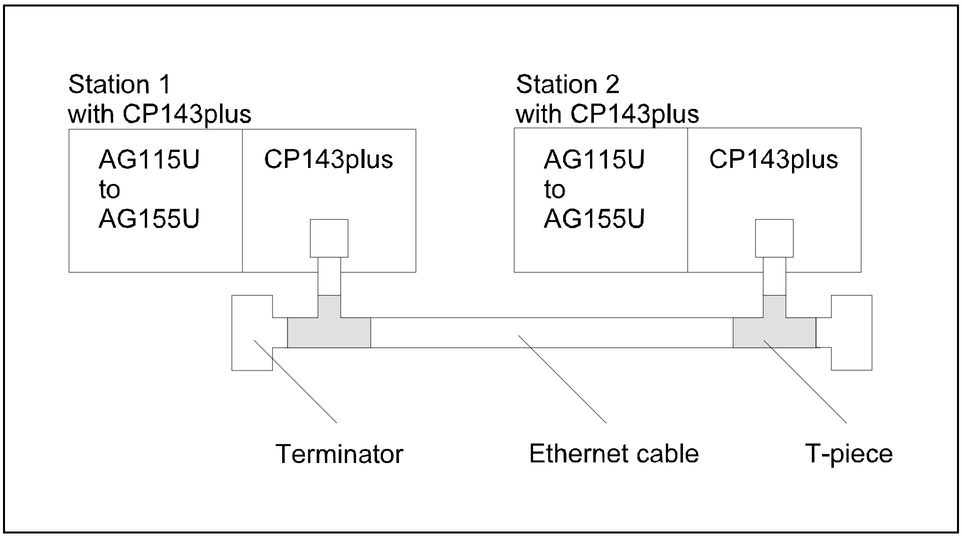


Fig. 6-7: System structure for the introductory example using H1

6.4.2.2 System structure required for the TCP/IP protocol

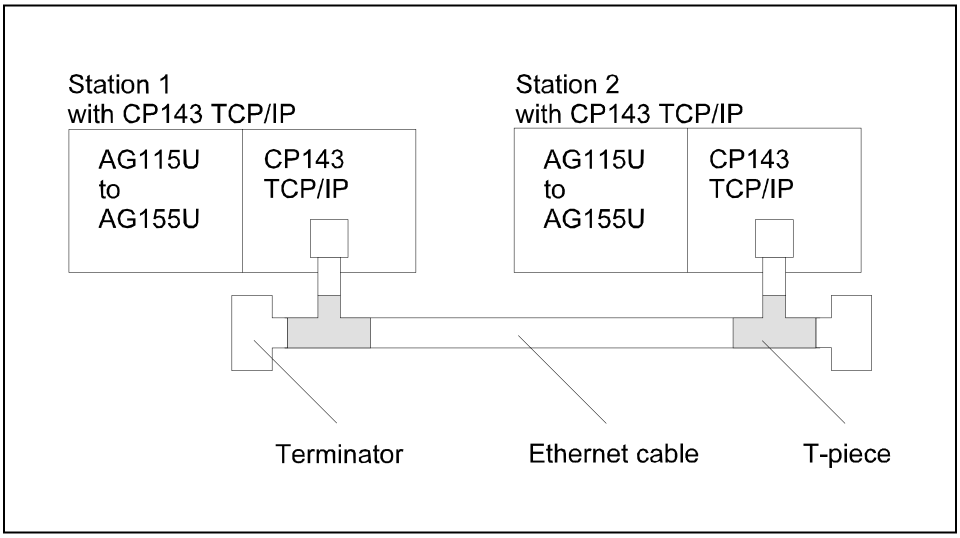


Fig. 6-8: System structure for the introductory example using TCP/IP

6.4.2.3 Jobs to be performed by the PLCs

Jobs for PLC 1

- Every 100ms the data words DW0 to DW99 are transferred via the data module DB11.
- The data word DW0 of DB11 is used as a message counter. It is incremented only when the previous send job has been completed correctly (complete without errors). The remaining data words (DW1 to DW99) may be used to transfer user data.
- SEND is configured by the job number A-NR = 11 and the base-frame number SSNR = 0 .
- Source parameters must be configured directly.

Jobs for PLC 2

- PLC2 must receive the data transmitted by PLC1 and deposit it in data module DB12. The RECEIVE handler module must be used for this purpose.
- Handler module RECEIVE is configured by means of the job number A-NR = 12 and the base-frame number SSNR = 0 .
- The destination parameters are saved in the data module DB12 from data word DW0.
- The entered base-frame number SSNR = 0 must correspond with the configuration of the CP143 H1 / TCP/IP module. For this example this configuration is identical to the one shown for the CP143 H1 / TCP/IP in PLC 1

This concludes the description of the problem and the required configuration settings. For further details on the configuration of the handler modules you may refer to the programs in chapter 6.4.3. Chapter 6.4.4 describes the respective configuration for the CP143plus under H1 and Chapter 6.4.5 CP143 TCP/IP under TCP/IP in detail.

6.4.3 Programs for the PLCs

The programming of the PLCs does not depend on the protocol and may be used for H1 and TCP/IP.

The following sample programs are provided for PLCs of the type AG135U and AG155U. These programs may also be used on PLCs of the type 115U and 150U after the respective handler modules have been changed.

6.4.3.1 Synchronization of the interfaces (program for PLC 1 and PLC 2)

When a PLC is started every interface of the CP must be synchronized by means of the handler module SYNCHRON. Due to the fact that this is applicable to all types of start-up of the PLC, the required number of SYNCHRON modules must be initiated in

- OB20 for a new start
- OB21 for manual re-start
- OB22 for a re-start after power failed

The function module NETZEIN (FB22) (POWERON) checks whether synchronization errors occurred or not. When an error is detected the program will be terminated by means of a stop command (STP).

The following modules must be transferred into PLC1 and PLC2.

Operational modules:

OB20, OB21, OB22

```
Network 1
;                     Restart after power on
00000      :
00002      :SPA FB 22                          Delay time and synchron.
      NAME #POWERON
00006      :
00008      :BE
```

Function module:

FB22

```
      BSTNAME #POWERON
      BIB     #2085
0000E      :
00010      :L   KT 20.2                       max. Delay time  20s
00014      :U   M 0.0
00016      :UN  M 0.0
00018      :SV  T 43                          VKE 0 for neg. edge
0001A      :O   M 0.0
0001C      :ON  M 0.0
0001E      :SV  T 43                          VKE 1 to start the timer
00020      :
00022 SYN  :SPA FB 125                        Synchronization-module
      NAME #SYNCHRON
      SSNR =KY 0,0                            SSNR or frame base 0
      BLGR =KY 0,6                            Block size 6 (512 bytes)
      PAFE =MB 199                            Configuration error byte MB199
0002C      :
0002E      :UN  M 199.0                       terminate if no error occurred
00030      :BEB
00032      :
00034      :U   T 43                          as long as timer is active
00036      :SPB =SYN                          attempt synchronization
00038      :L   KH 2222                       load error ID into accumulator
0003C      :STP                               stop the PLC
0003E      :BE
```

The CP143 H1 / TCP/IP requires app. 10 s to start after power is turned on. You will receive an error message via the configuration error byte if the PLC attempts to synchronize during this period. This message is removed as soon as the CP has completed the start-up procedure.

Initially the timer of this module is set to 20s. If the units are unable to synchronize properly during this time all processing will be stopped.

6.4.3.2 Cycle, function and data module for PLC1

The transmission from PLC1 is initiated by means of the handler module SEND. This call is issued in function module FB1. FB1 is accessed by the first command, a jump, in the cycle organization module OB1. The send-job is configured as follows.

Cycle operation module:

OB1

```
Network 1
;                     Cycle
00001      :SPA FB 1                           FB1 send job
      NAME #FB1
      TIME =KT 10.0                            Transmit timer  10 means 100ms
00002      :O  M 0.0
00004      :ON M 0.0                           Logical result 1
00006      :SPA FB 126                         Send All
      NAME #SEND-A
      SSNR =KY 0,0                             SSNR or frame base  0
      A-NR =KY 0,0                             A-Nr. 0 for  ALL-job
      ANZW =MW 190                             Indicator word MW190
      PAFE =MB 199                             Configuration error byte MB199
00012      :
00014      :SPA FB 127                         Receive All
      NAME #REC-A
      SSNR =KY 0,0                             SSNR or frame base 0
      A-NR =KY 0,0                             A/No. 0 for All/Job
      ANZW =MW 194                             Indicator word MW194
      PAFE =MB 198                             Configuration error byte MB198
00020      :
00030      :BE
```

Function module:

FB1

```
      BSTNAME #FB1
      BIB     #19075
      BEZ     #TIME  D:KT
00010      :
00014      :SPA FB 123                         Control module
      NAME #CONTROL
      SSNR =KY 0,0                             Frame base 0
      A-NR =KY 0,11                            Job No. 11
      ANZW =MW 0                               Indicator word into MW0
      PAFE =MB 189                             Configuration error byte MB189
00020      :
00022      :O   M 1.1                          as long as the job is active
00024      :O   T 11                           or the timer is active
00026      :BEB                                no further processing
00028      :
0002A      :LW  =TIME                          load the pre-configured time
0002C      :U   M 0.0                          and start the timer again
0002E      :UN  M 0.0
00030      :SV  T 11                           VKE 0 timer start
00032      :O   M 0.0
00034      :ON  M 0.0
00036      :SV  T 11                           VKE 1 timer start
00038      :
0003A      :U   M 1.3                          if completed with error(s),
0003C      :SPB =SEND                          do not increment counter
0003E      :
00040      :A   DB 11                          Send-DB
00042      :L   DW 0                           Message counter
00044      :L   KB 1                           in DW0, which is also transmitted
00046      :+F
00048      :T   DW 0
0004A      :
0004C SEND :
0004E      :SPA FB 120                         Send module
      NAME #SEND
      SSNR =KY 0,0                             Interface No. 0
      A-NR =KY 0,11                            Job No. 11
      ANZW =MW 0                               Indicator word in MW0
      QTYP =KC DB                              Send data from DB
      DBNR =KY 0,11                            DB number 11
      QANF =KF +0                              from data word 0
      QLAE =KF +100                            100 elements (DWs)
      PAFE =MB 99
00062      :
00064      :BE
```

FB1 is a simple send module that checks the indicator word flags:

- job active
- completed with error(s)

Data module:

DB11

```
MODULE#DB11
      BIB     #9095
00000:       KH    = 0000                      1st Data    message counter
00001:   100 (                                 100 other data words
00001:       KH    = 1111
00002:       )
```

When FB1 is called, the value for the timer determines how often the SEND job is triggered. In this example the timer has been set to 100ms. The sample program will thus trigger the SEND job every 100ms.

Data word DW0 of the data module DB11 is incremented before a SEND call transmitting a proper message is executed. This occurs in function module FB1.

It is possible to transmit 99 user data items along with DW0.

6.4.3.3 Cycle, function and data module for PLC2

The data that was transmitted by PLC1 is received in PLC2 by means of handler module RECEIVE. The respective job is initiated in organization module OB1.

Cycle operation module:

OB1

```
Network 1
;                     Cycle
00001      :SPA FB 2                           FB2 receive section
      NAME #FB2
00002      :O  M 0.0
00004      :ON M 0.0                           Logical result 1
00006      :SPA FB 126                         Send All
      NAME #SEND-A
      SSNR =KY 0,0                             SSNR or Frame base  0
      A-NR =KY 0,0                             A-Nr. 0 for  ALL-job
      ANZW =MW 190                             Indicator word MW190
      PAFE =MB 199                             Configuration error byte MB199
00012      :
00014      :SPA FB 127                         Receive All
      NAME #REC-A
      SSNR =KY 0,0                             SSNR or Frame base 0
      A-NR =KY 0,0                             A/Nr. 0 for ALL/job
      ANZW =MW 194                             Indicator word MW194
      PAFE =MB 198                             Configuration error byte MB198
00020      :
00030      :BE
```

Function module:

FB2

```
      BSTNAME #FB2
      BIB     #19075
0000C      :
0000E      :SPA FB 123                             Control module
      NAME #CONTROL
      SSNR =KY 0,0                                 Interface No. 0
      A-NR =KY 0,12                                Job No. 12
      ANZW =MW 4                                   Indicator word MW4
      PAFE =MB 199
0001A      :
0001C      :UN  M 5.0                              If no data is available,
0001E      :BEB                                    end
00020      :
00028      :L   MW 12                              Increment receive counter
0002A      :L   KB 1
0002C      :+F
0002E      :T   MW 12
00030      :
00032 REC  :
00034      :SPA FB 121                             Receive module
      NAME #RECEIVE
      SSNR =KY 0,0                                 Interface No. 0
      A-NR =KY 0,12                                Job No. 12
      ANZW =MW 4                                   Indicator word MW4
      ZTYP =KC DB                                  Save data in data module
      DBNR =KY 0,12                                DB number 12
      ZANF =KF +0                                  from DW0
      ZLAE =KF +100                                Length 100 elements
      PAFE =MB 199
00048      :
0004A      :BE
```

FB2 is a simple receive module which checks the indicator word flags:

- handshake required
- completed with error

Data module:

DB12

```
MODULE#DB12
      BIB     #9095
;                          Receive data module (A-No. 12)
00000:       KH    = 0000                      1st data   transmit message ctr.
00001:   100 (                                 100 further data words
00001:       KH    = 0000
00002:       )
```

6.4.4 Module configuration under H1

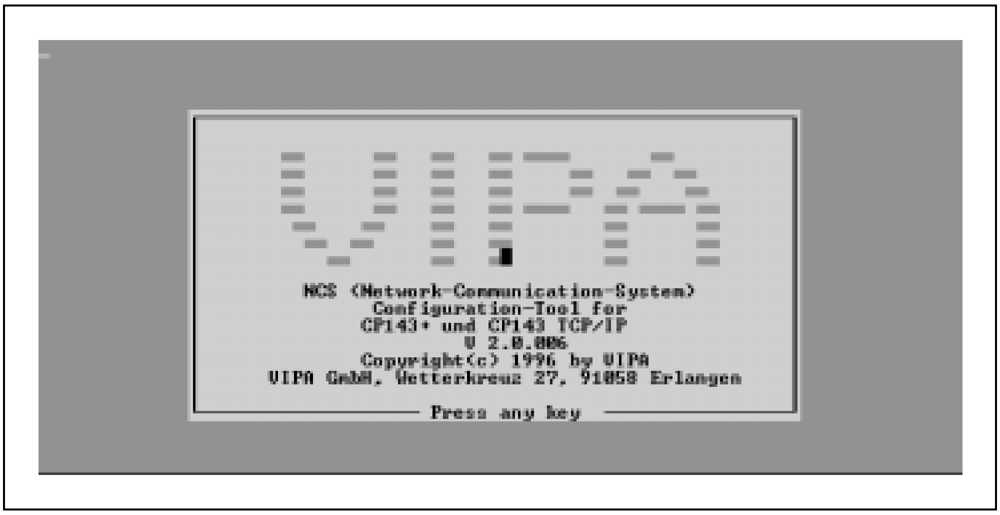
The configuration for station 1 and 2 differs only in the respective parameters. The method is the same. The following steps show how both stations must be configured. The parameters for each station must be set individually.

To configure the two stations you must please perform the following steps:

- Construct a network as per Fig. 6-7.
- If you have not yet installed the configuration software, do so now. The installation procedure and instructions on how the program should be started are available from chapter 5.
- Generate the base data and transfer these into the CP
    1. Start the program and initialize the CP
    2. Check transport parameters and change where necessary
    3. Configure the connection
    4. Save base data
    5. Transfer base data to the CP

Re: 1. Start the program and initialize the CP

A window with data is displayed after the program has started. This window contains information on the version no. This you will need for future enquiries from our Hotline.



```
NCS (Network-Communication-System)
Configuration-Tool for
CP143+ und CP143 TCP/IP
V 2.0.006
Copyright(c) 1996 by VIPA
VIPA GmbH, Wetterkreuz 27, 91058 Erlangen
```

Fig. 6-9: Initial display of the configuration program

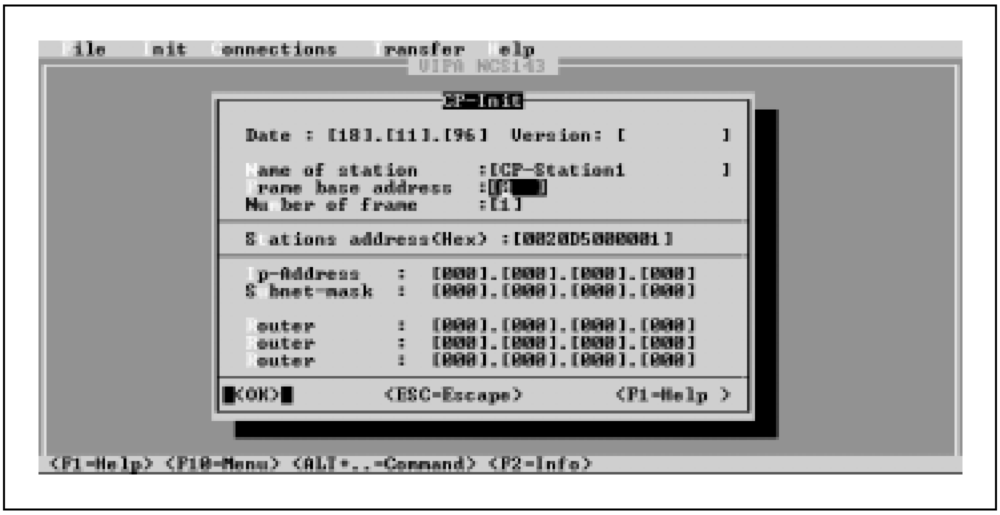
Press any key to continue to the main menu. Select the menu item **Init** and then *CP-Init* to open the CP initialization window.

The description of the configuration for both stations follows.

Every station must be configured individually. Please enter the data for station 1, save the data and transfer it to station 1.

Now you must perform the same steps for station 2.

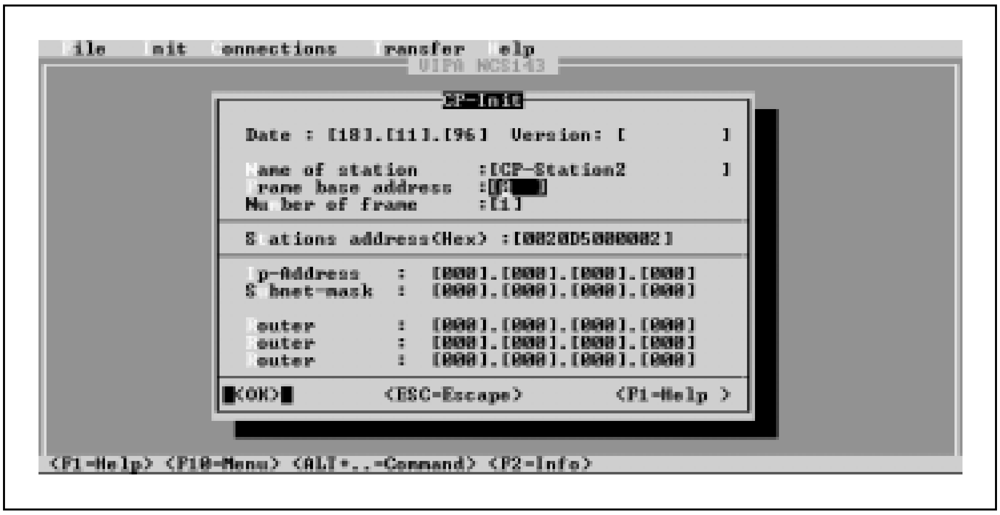
Station 1



```
 File    Init  Connections    Transfer  Help
                              VIPA NCS143
                    CP-Init
  Date : [18].[11].[96]  Version: [          ]

  Name of station       :[CP-Station1        ]
  Frame base address    :[     ]
  Number of frame       :[1]

  Stations address(Hex) :[0020D5000001]

  Ip-Address    :   [000].[000].[000].[000]
  Subnet-mask   :   [000].[000].[000].[000]

  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]

 <OK>          <ESC-Escape>         <P1-Help >
<F1-Help> <F10-Menu> <ALT+..-Command> <F2-Info>
```

Fig. 6-10: CP initialization station 1 for H1

Station 2

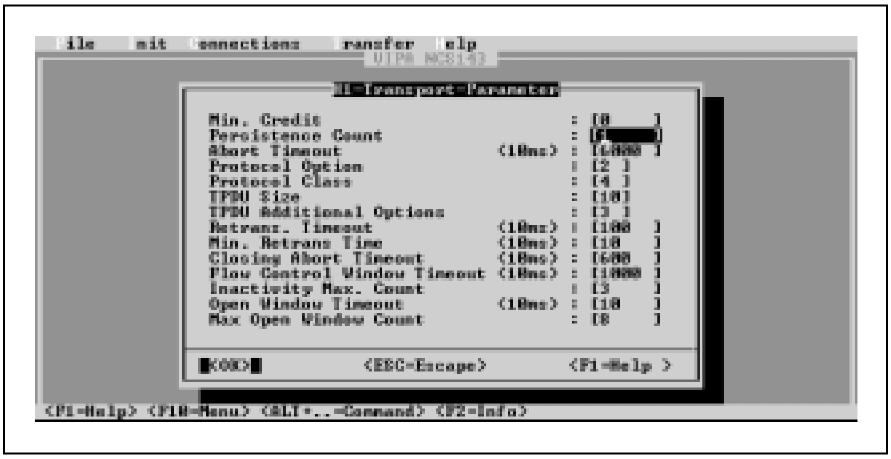


```
 File    Init  Connections    Transfer  Help
                              VIPA NCS143
                    CP-Init
  Date : [18].[11].[96]  Version: [          ]

  Name of station       :[CP-Station2        ]
  Frame base address    :[     ]
  Number of frame       :[1]

  Stations address(Hex) :[0020D5000002]

  Ip-Address    :   [000].[000].[000].[000]
  Subnet-mask   :   [000].[000].[000].[000]

  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]

 <OK>          <ESC-Escape>         <P1-Help >
<F1-Help> <F10-Menu> <ALT+..-Command> <F2-Info>
```

Fig. 6-11: CP initialization station 2 for H1

Re: 2. Check transport parameters

Normally, transport parameters do not require changing. The dialogue window is displayed so that the H1 transport parameters may be checked.. Please note, that the transport parameters for all stations are the same.


File Edit Connections Transfer Help
VIPA NCS143
H1-Transport-Parameter
<OK> <ESC=Escape> <F1=Help >
<F1=Help> <F10=Menu> <ALT+...=Command> <F2=Info>

```
Min. Credit                             : [0    ]
Persistence Count                       : [     ]
Abort Timeout                   (10ms)  : [6000 ]
Protocol Option                         : [2 ]
Protocol Class                          : [4 ]
TPDU Size                               : [10]
TPDU Additional Options                 : [3 ]
Retrans. Timeout                (10ms)  : [100  ]
Min. Retrans Time               (10ms)  : [10   ]
Closing Abort Timeout           (10ms)  : [600  ]
Flow Control Window Timeout     (10ms)  : [1000 ]
Inactivity Max. Count                   : [3    ]
Open Window Timeout             (10ms)  : [10   ]
Max Open Window Count                   : [8    ]
```

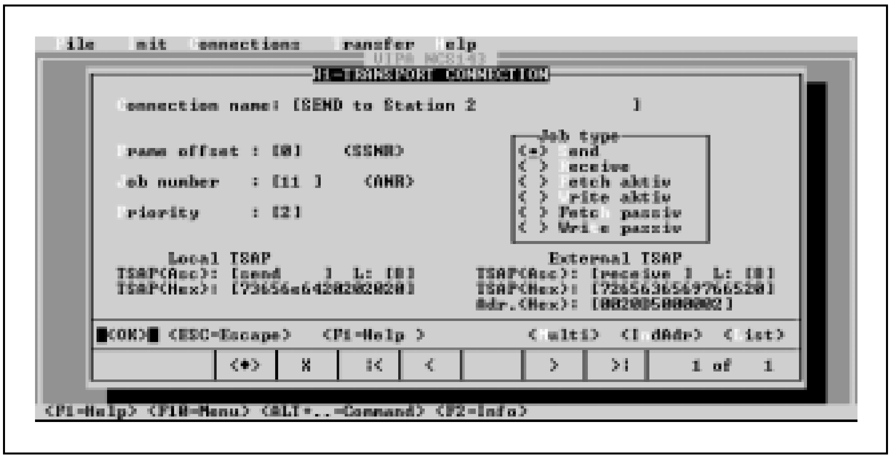
Fig. 6-12: Transport-parameter of station 1 and station 2

Re: 3. Configure the connection

Station 1

This example is concerned with H1 connections. From the menu item **Verbindungen** (Connections) you must select *H1-Transport-Verbindung* (H1-transport connection). This will display the window for TCP connections. Enter the following data for station 1:

- Connection name: SEND to station 2
- Job number: 11
- Priority: 2
- Type of job: SEND
- TSAP local station (ASCII): send
- TSAP remote station (ASCII): receive
- Adr. of remote st.: enter the ethernet-address of station 2.


File Edit Connections Transfer Help
VIPA NCS143
H1-TRANSPORT CONNECTION
<OK> <ESC=Escape> <F1=Help > <Multi> <IndAdr> <List>
<*> X |< < > >| 1 of 1
<F1=Help> <F10=Menu> <ALT+...=Command> <F2=Info>

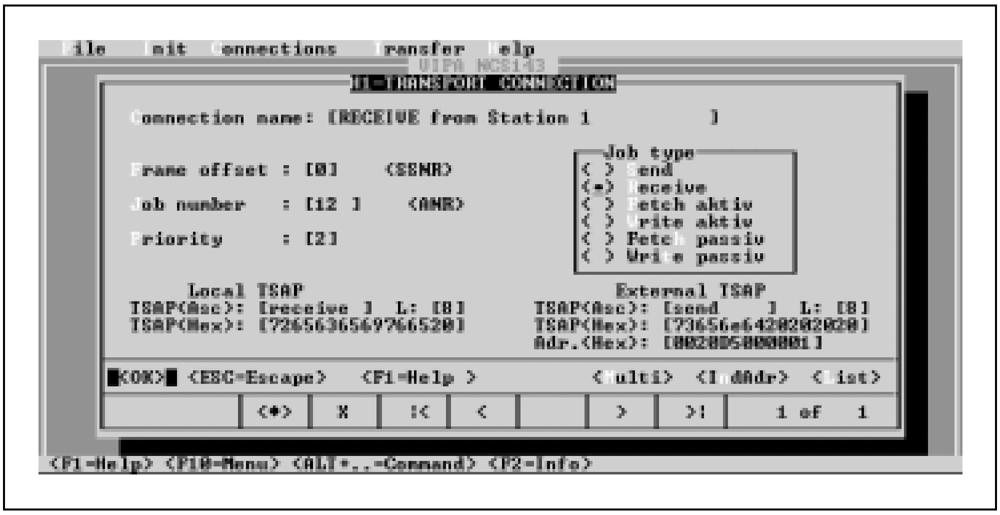
```
Connection name: [SEND to Station 2              ]

Frame offset : [0]    <SSNR>                 Job type
                                             (*) Send
Job number   : [11 ]    <ANR>                ( ) Receive
                                             ( ) Fetch aktiv
Priority     : [2]                           ( ) Write aktiv
                                             ( ) Fetch passiv
                                             ( ) Write passiv

      Local TSAP                               External TSAP
TSAP(Asc): [send    ]  L: [8]           TSAP(Asc): [receive ]  L: [8]
TSAP(Hex): [73656e6420202020]           TSAP(Hex): [7265636569766520]
                                        Adr.(Hex): [00200500002]
```

Fig. 6-13: H1 connection parameters station 1

Station 2

Enter the following values for station 2:

- Connection name: RECEIVE from Station 1
- Job number: 12
- Priority: 2
- Type of job: RECEIVE
- TSAP local station (ASCII): receive
- TSAP remote station (ASCII): send
- Adr. remote st.: enter the ethernet address of station 1.


File Init Connections Transfer Help
VIPA NCS143
H1-TRANSPORT CONNECTION
<OK> <ESC=Escape> <F1=Help > <Multi> <IndAdr> <List>
<*> X |< < > >| 1 of 1
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>

```
Connection name: [RECEIVE from Station 1          ]

Frame offset : [0]    <SSNR>
Job number   : [12 ]    <ANR>
Priority     : [2]

Job type
( ) Send
(•) Receive
( ) Fetch aktiv
( ) Write aktiv
( ) Fetch passiv
( ) Write passiv

     Local TSAP                            External TSAP
TSAP(Asc): [receive ]  L: [8]      TSAP(Asc): [send    ]  L: [8]
TSAP(Hex): [7265636569766520]      TSAP(Hex): [73656e6420202020]
                                   Adr.(Hex): [0020D5000001]
```

Fig. 6-14: H1 transport parameter station 2

Click on **[OK]** when you have completed the entries. You will be asked whether the connection should be accepted. Please confirm to enable the connection. This concludes the configuration of the different connections for the respective stations.

Re: 4. Save base data

Save the connection for the respective station. From menu item **File** (Datei) select the entry *Save* (Speichern).

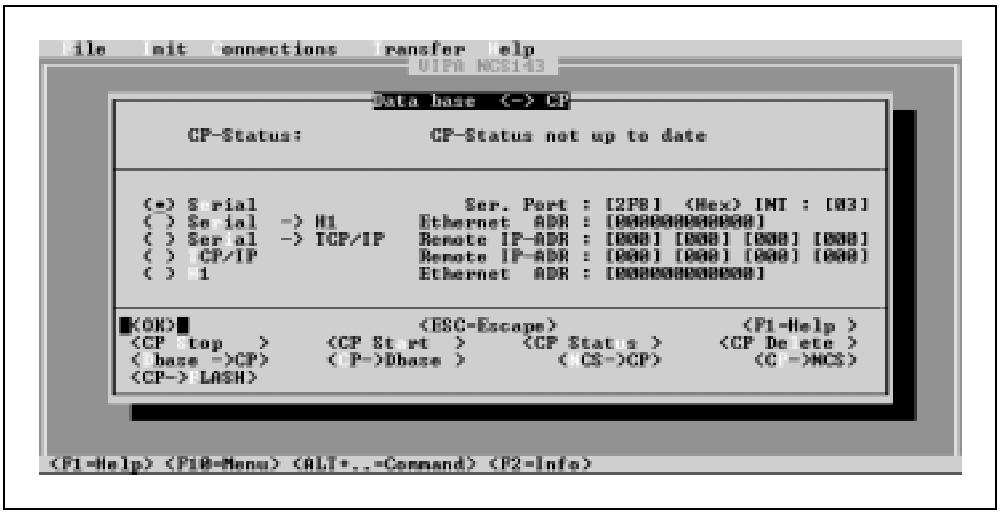
Enter "CP1.NCS" for the file name where the base data must be saved.

Repeat these activities for station 2 but use the file name "CP2.NCS".

Re: 5. Transfer base data to the CP

You can select one of two methods to transfer the base data into the CP: you may write the data to an EPROM or you may transfer the data via the serial interface to the respective CP. Here we will transfer the data directly via the serial interface. First you have to connect your PC with the CP 1.

Load the data base CP1.NCS for station 1. Check the data by selecting the respective window from **Init** and **connections**. If the data is correct you may transfer it to the CP. Select the function *Database* (Datenbasis) ↔ *CP* from the menu item **Transfer** and the dialogue window for remote configuration will be displayed:



```
 File    Init   Connections    Transfer   Help
                                VIPA NCS143
                  Data base  <-> CP

      CP-Status:            CP-Status not up to date

    (•) Serial                   Ser. Port  : [2F8]  <Hex> INT : [03]
    ( ) Serial   -> H1       Ethernet   ADR : [000000000000]
    ( ) Serial   -> TCP/IP   Remote IP-ADR  : [000] [000] [000] [000]
    ( ) TCP/IP               Remote IP-ADR  : [000] [000] [000] [000]
    ( ) H1                   Ethernet   ADR : [000000000000]

 <OK>                    <ESC=Escape>                       <F1=Help >
  <CP Stop    >      <CP Start   >      <CP Status >      <CP Delete >
  <Dbase ->CP>       <CP->Dbase >          <NCS->CP>          <CP->NCS>
  <CP->FLASH>

 <F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 6-15: Serial configuration

Select Serial and enter the the port address 2F8 and the interrupt 03. Press the “Enter” key or click on **[OK]**. You are now in control of the CP if the port address and the interrupt are valid.

STOP the CP to start the transfer of data. Request the status from the CP to check your actions.

In this example we will transfer the data directly from the NCS. Click on the button **[NCS→CP]**. Once you have verified your action the data will be transferred directly to the CP. The connections will be activated when you change the CPs status back to RUN.

Use the same method to transfer the database into the CP of station 2. The data communication link will be activated as soon as the data has been transferred and both CPs are RUNning.

You may now monitor the configured connections by means of the VIPA-MC5 package. The respective actions are described in chapter 6.4.6.

6.4.5 Module configuration under TCP/IP

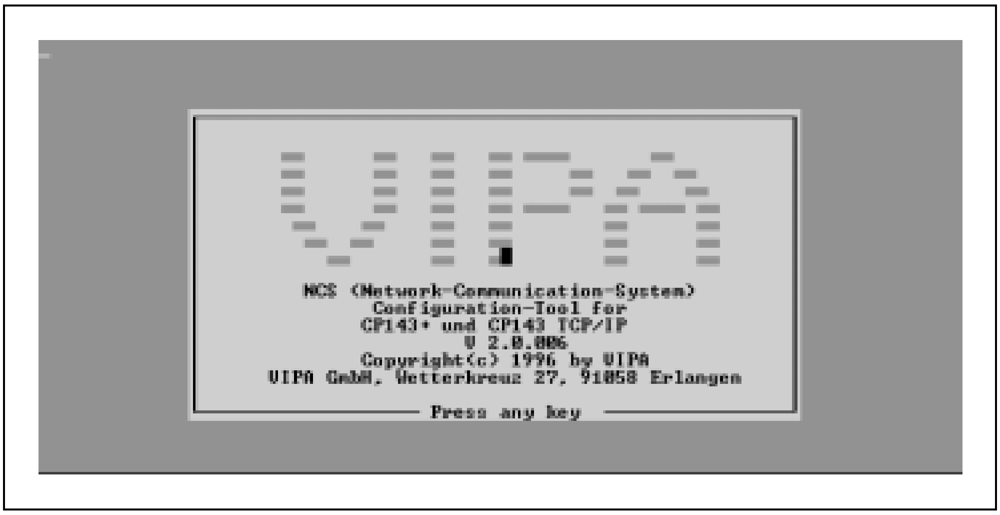
The configuration for station 1 and 2 differs only in the respective parameters. The method is the same. The following steps show how both stations must be configured. The parameters for each station must be set individually.

To configure the two stations you must please perform the following steps:

- Construct a network as per Fig. 6-8.
- If you have not yet installed the configuration software, do so now. The installation procedure and instructions on how the program should be started are available from chapter 5.
- Generate the base data and transfer these into the CP
  1. Start the program and initialize the CP
  2. Check transport parameters and change where necessary
  3. Configure the connection
  4. Save base data
  5. Transfer the base data to the CP

Re: 1. Start the program and initialize the CP

A window with data is displayed after the program has started. This window contains information on the version no. This you will need for future enquiries from our Hotline.



```
NCS (Network-Communication-System)
Configuration-Tool for
CP143+ und CP143 TCP/IP
V 2.0.006
Copyright(c) 1996 by VIPA
VIPA GmbH, Wetterkreuz 27, 91058 Erlangen
─────────── Press any key ───────────
```

Fig. 6-16: Start-up screen of the configuration program

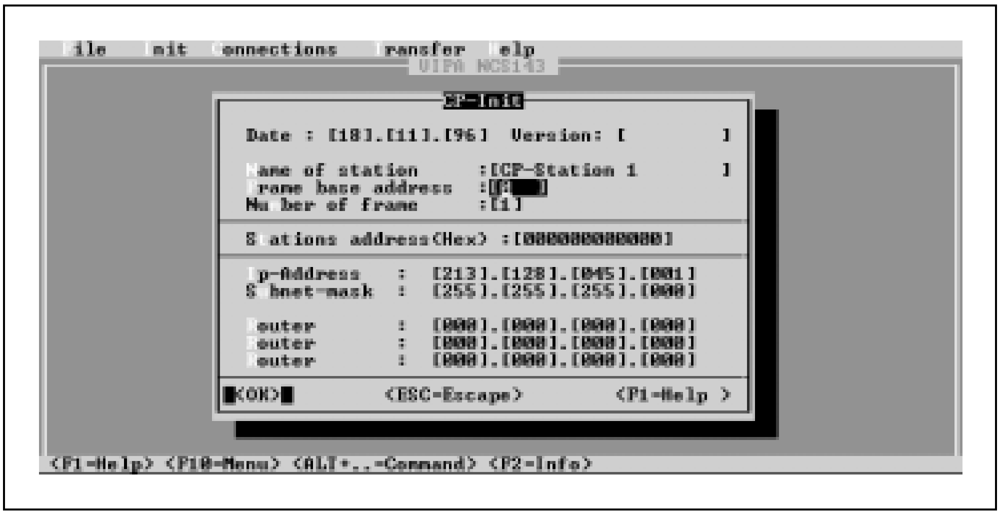
Press any key to access the main menu. From the menu item **Init** select *CP-Init* and the CP initialization window will be displayed.

The description of the configuration for both stations follows.

Every station must be configured individually. Please enter the data for station 1, save the data and transfer it to station 1.

Now you must perform the same steps for station 2.

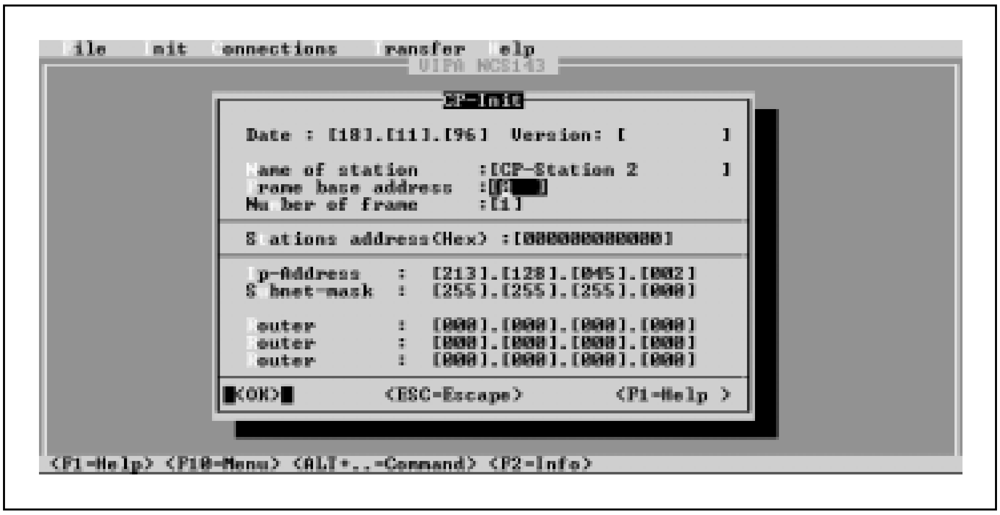
Station 1



```
 File    Init  Connections    Transfer  Help
                          VIPA NCS143
                       CP-Init
  Date : [18].[11].[96]  Version: [          ]

  Name of station      :[CP-Station 1        ]
  Frame base address   :[[illegible]]
  Number of frame      :[1]

  Stations address(Hex) :[000000000000]

  Ip-Address    :   [213].[128].[045].[001]
  Subnet-mask   :   [255].[255].[255].[000]

  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]

<OK>            <ESC=Escape>          <P1=Help >
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 6-17: CP initialization station 1 for TCP

Station 2

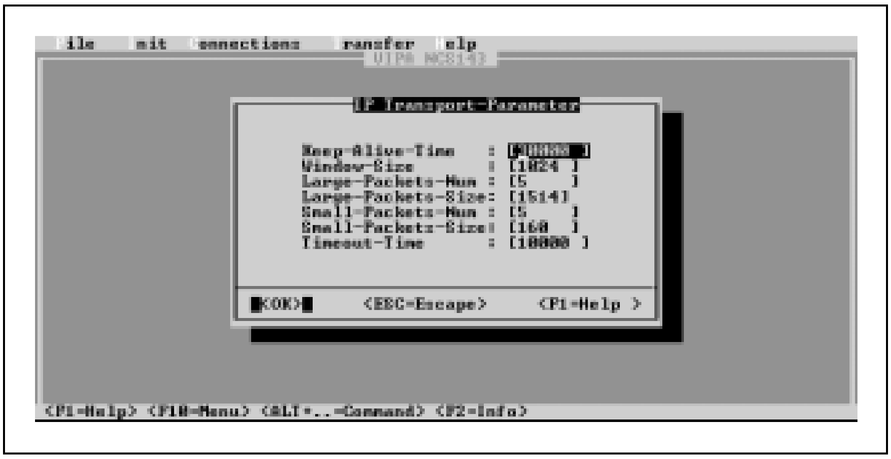


```
 File    Init  Connections    Transfer  Help
                          VIPA NCS143
                       CP-Init
  Date : [18].[11].[96]  Version: [          ]

  Name of station      :[CP-Station 2        ]
  Frame base address   :[[illegible]]
  Number of frame      :[1]

  Stations address(Hex) :[000000000000]

  Ip-Address    :   [213].[128].[045].[002]
  Subnet-mask   :   [255].[255].[255].[000]

  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]
  Router        :   [000].[000].[000].[000]

<OK>            <ESC=Escape>          <P1=Help >
<F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 6-18: CP initialization station 2 for TCP

Re: 2. Check transport parameters

Normally, transport parameters do not require changing. The dialogue window is displayed so that the IP transport parameters may be checked.. Please note, that the transport parameters for all stations are the same.


File Init Connections Transfer Help
VIPA NCS143
IP Transport-Parameter
<OK> <ESC=Escape> <F1=Help >
<F1=Help> <F10=Menu> <ALT+...=Command> <F2=Info>

```
Keep-Alive-Time     : [illegible]
Window-Size         : [1024 ]
Large-Packets-Num   : [5    ]
Large-Packets-Size: [1514]
Small-Packets-Num   : [5    ]
Small-Packets-Size: [160  ]
Timeout-Time        : [10000 ]
```

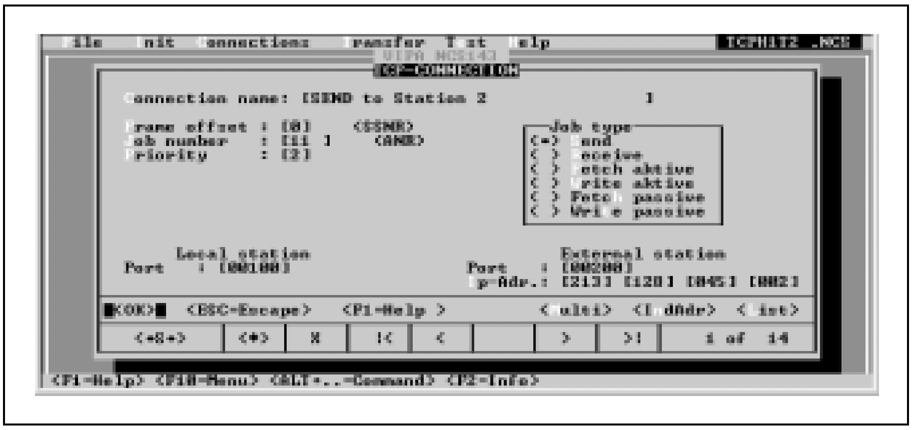
Fig. 6-19: Transport parameters of station 1 and station 2

Re: 3. Configure the connection

Station 1

This example is concerned with the respective TCP connections. Select *TCP-Verbindung* (H1-transport connection) from the menu item **Verbindungen** (Connections). This will display the window for TCP connections. Enter the following data for station 1:

Connection name:	SEND to station 2
Job number:	11
Priority:	2
Type of job:	SEND
Port local station:	00100
Port remote station:	00200
IP-Adr. remote St.:	Enter the IP address of station 2.


File Init Connections Transfer Test Help
TCPH1T2 .NCS
VIPA NCS143
TCP-CONNECTION
<SSNR>
<ANR>
<OK> <ESC=Escape> <F1=Help > <Multi> <IndAdr> <List>
<+S+> <+> X |< < > >| 1 of 14
<F1=Help> <F10=Menu> <ALT+...=Command> <F2=Info>

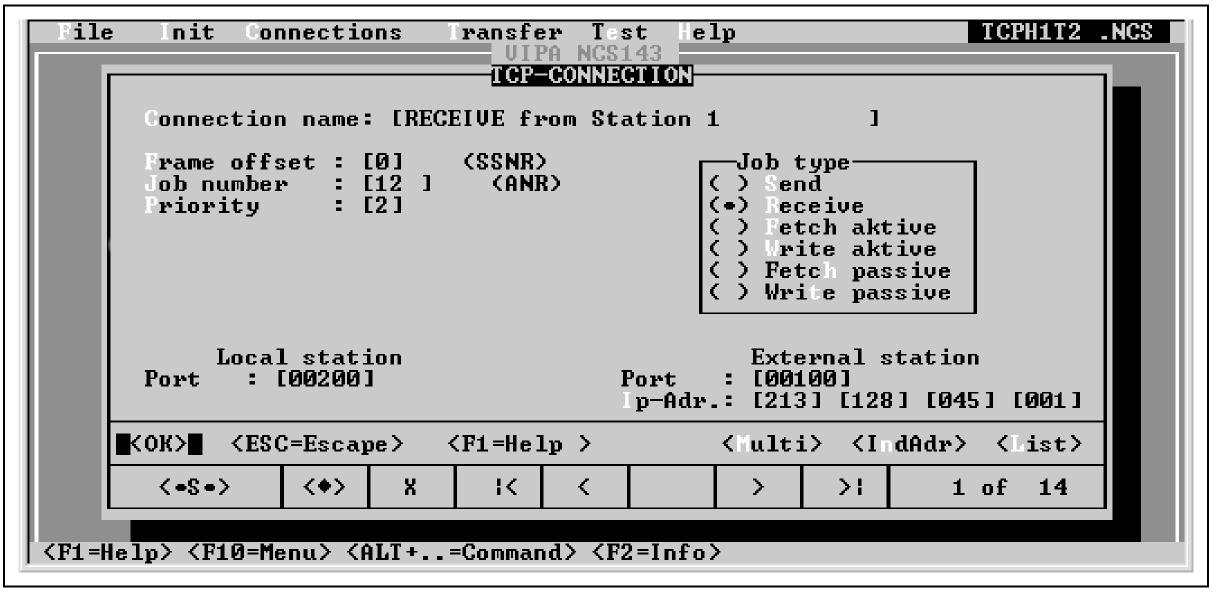
```
Connection name: [SEND to Station 2              ]

Frame offset  : [0]
Job number    : [11 ]
Priority      : [2]

Job type
(•) Send
( ) Receive
( ) Fetch aktive
( ) Write aktive
( ) Fetch passive
( ) Write passive

Local station                External station
Port    : [00100]            Port    : [00200]
                             Ip-Adr.: [213] [128] [045] [002]
```

Fig. 6-20: TCP connection parameters station 1

Station 2

Enter the following values for station 2:

- Connection name: RECEIVE from Station 1
- Job number: 12
- Priority: 2
- Type of job: RECEIVE
- Port local station: 00200
- Port remote station: 00100
- IP-Adr. remote St.: Enter the IP address of Station 1.



File Init Connections Transfer Test Help
TCPH1T2 .NCS
VIPA NCS143
TCP-CONNECTION

```
Connection name: [RECEIVE from Station 1          ]

Frame offset  : [0]     <SSNR>          ┌Job type──────────┐
Job number    : [12 ]     <ANR>         │( ) Send          │
Priority      : [2]                     │(•) Receive       │
                                        │( ) Fetch aktive  │
                                        │( ) Write aktive  │
                                        │( ) Fetch passive │
                                        │( ) Write passive │
                                        └──────────────────┘

       Local station                          External station
Port    : [00200]                    Port    : [00100]
                                     Ip-Adr.: [213] [128] [045] [001]
```

Fig. 6-21: TCP transport parameters station 2

Click on **[OK]** when you have completed the entries. You will be asked whether the connection should be accepted. Please confirm to enable the connection. This concludes the configuration of the different connections for the respective stations.

Re: 4. Save base data

Save the connection for the respective station. Select the entry *Save* (Speichern) from menu item **File** (Datei).

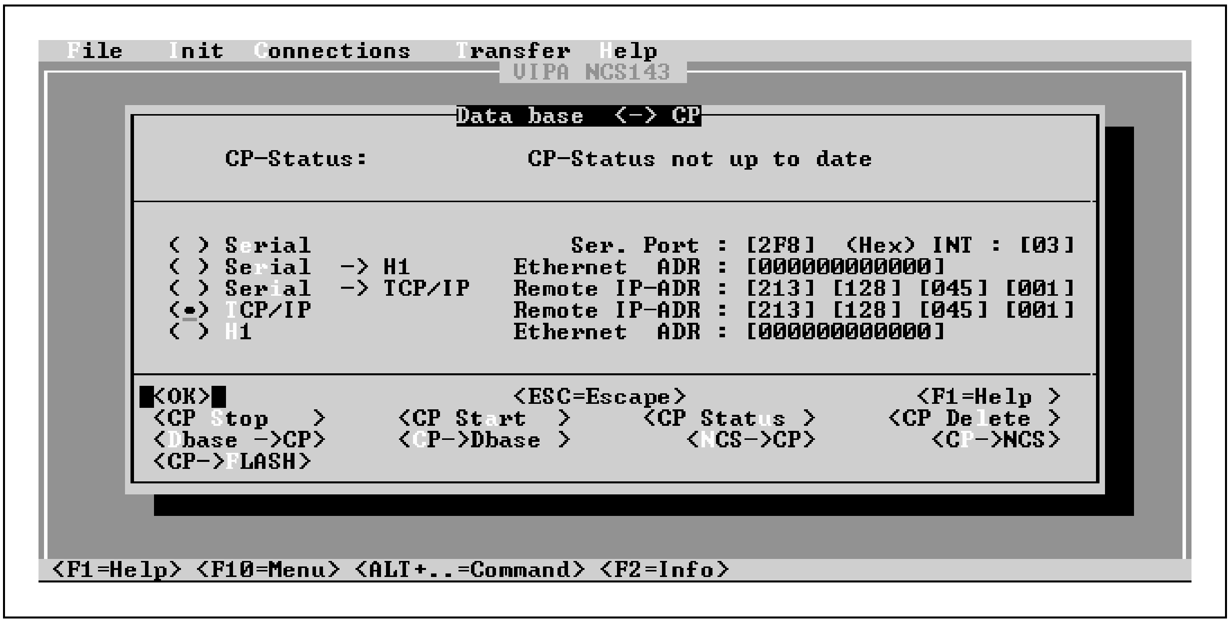
Enter "CP1.NCS" for the file name where the base data must be saved.

Repeat these activities for station 2 but use the file name "CP2.NCS".

Re: 5. Transfer base data to the CP

You can select one of two methods to transfer the base data into the CP: you may write the data to an EPROM or you may transfer the data via the LAN or the serial interface to the respective CP. Here we will transfer the data directly via the LAN. To do this, the TCP/IP-stack of ftp-Software must be up and running.

Load the data base CP1.NCS for station 1. Check the data by selecting the respective window from **Init** and **connection** (Verbindungen). If the data is correct you may transfer it to the CP. Select the function *Database* (Datenbasis) ↔ *CP* from the menu item **Transfer** and the dialogue window for remote configuration will be displayed:



```
 File     Init   Connections     Transfer   Help
                              VIPA NCS143

                      Data base  <-> CP

      CP-Status:           CP-Status not up to date

    ( ) Serial                     Ser. Port : [2F8]  (Hex) INT  : [03]
    ( ) Serial   -> H1         Ethernet   ADR : [000000000000]
    ( ) Serial   -> TCP/IP     Remote IP-ADR : [213] [128] [045] [001]
    (•) TCP/IP                 Remote IP-ADR : [213] [128] [045] [001]
    ( ) H1                     Ethernet   ADR : [000000000000]

 <OK>                          <ESC=Escape>                   <F1=Help >
  <CP Stop   >      <CP Start  >      <CP Status >      <CP Delete >
  <Dbase ->CP>      <CP->Dbase >           <NCS->CP>           <CP->NCS>
  <CP->FLASH>

 <F1=Help> <F10=Menu> <ALT+..=Command> <F2=Info>
```

Fig. 6-22: Remote configuration

Select TCP/IP and enter the IP address of station 1 under remote IP addr. Press the “Enter” key or click on **[OK]**. You are now in control of the CP if the address is valid and if the station exists on the LAN.

STOP the CP to start the transfer of data. Request the status from the CP to check your actions.

In this example we will transfer the data directly from the NCS. Click on the button **[NCS→CP]**. Once you have verified your action the data will be transferred directly to the CP. The connections will be activated when you change the CPs status back to RUN.

Use the same method to transfer the database into the CP of station 2. The data communication link will be activated as soon as the data has been transferred and both CPs are RUNning.

You may now monitor the configured connections by means of the VIPA-MC5 package. The respective actions are described in chapter 6.4.6.

6.4.6 Monitoring the transfer by means of the MC5 package

Both, modules and PLCs must be programmed to allow monitoring of connect requests.

It is assumed that all modules have been configured and that the PLCs have not been erased. The RUN/STOP switches of the PLCs must be in the STOP location. You must now load the PLC-programs described in chapter 6.4.3 into the two PLCs. Start the PLCs by switching the RUN/STOP switches to RUN. At this point only the green RUN LEDs may be on the PLCs and on the CPs. The two modules are communicating. This is evident from the KOMM-LED.

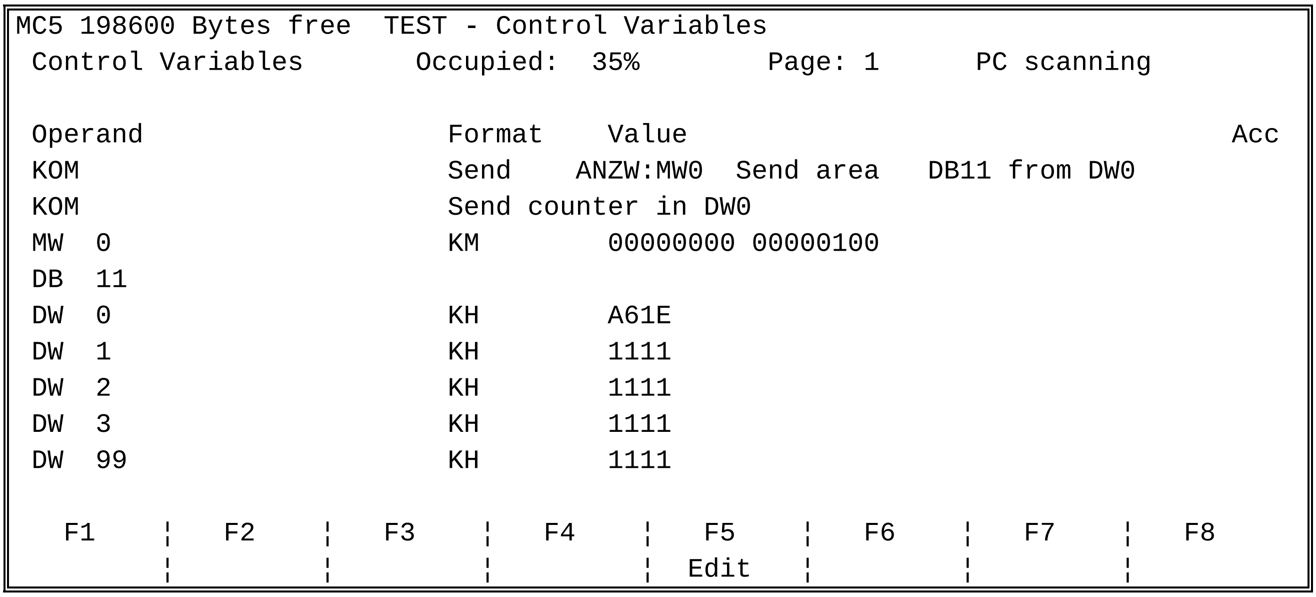
Start the MC5 package on on your PC and perform the following steps to monitor the transmit job:

- Call the test menu (F8) from the main menu.
- Change to the sub-menu “Control Variables” ("Steuern Variablen.") by means of "Steu.Var" (F5).

  The screen display indicates the respective function in the heading, the occupation of the current pages (in %) and the current number of pages (from 1 to 20). Below this the display is split into three columns. The "Operand" column is used to enter process variables by means of an editing function or by loading a list of variables. The "Format" column may be used for alterations, depending on the type of operand.

  Refer to the MC5 manual for more detailed information on the operation of the test menu of the MC5 package.
- Enter the respective data modules (here DB11) and data words (here DW0 to DW99) into the column with the heading "Operand".
- User data may be entered from DW1. Place the cursor on “Value” (Wert) and enter the value that you wish to transfer. In the example this “Value” contains "1111".

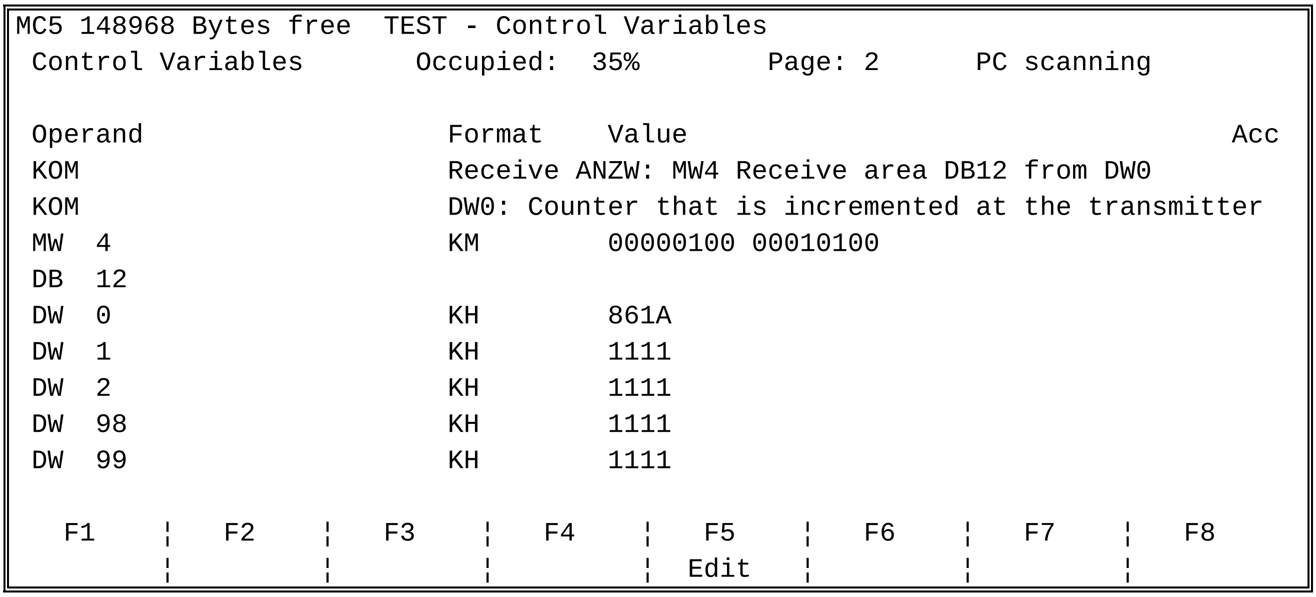
- Press F1 (Start). The entered values are transferred and the following display will appear:



```
MC5 198600 Bytes free  TEST - Control Variables
 Control Variables       Occupied:  35%        Page: 1      PC scanning

 Operand                    Format    Value                               Acc
 KOM                        Send    ANZW:MW0  Send area   DB11 from DW0
 KOM                        Send counter in DW0
 MW  0                      KM        00000000 00000100
 DB  11
 DW  0                      KH        A61E
 DW  1                      KH        1111
 DW  2                      KH        1111
 DW  3                      KH        1111
 DW  99                     KH        1111

   F1    ¦   F2    ¦   F3    ¦   F4    ¦   F5    ¦   F6    ¦   F7    ¦   F8
         ¦         ¦         ¦         ¦  Edit   ¦         ¦         ¦
```

Fig. 6-23: Display of the send job in the MC5 package

Connect the PC to PLC2 to display the receive job. Execute the same steps as shown for PLC1 and enter the following operands:



```
MC5 148968 Bytes free  TEST - Control Variables
 Control Variables       Occupied:  35%        Page: 2      PC scanning

 Operand                    Format    Value                               Acc
 KOM                        Receive ANZW: MW4 Receive area DB12 from DW0
 KOM                        DW0: Counter that is incremented at the transmitter
 MW  4                      KM        00000100 00010100
 DB  12
 DW  0                      KH        861A
 DW  1                      KH        1111
 DW  2                      KH        1111
 DW  98                     KH        1111
 DW  99                     KH        1111

   F1    ¦   F2    ¦   F3    ¦   F4    ¦   F5    ¦   F6    ¦   F7    ¦   F8
         ¦         ¦         ¦         ¦  Edit   ¦         ¦         ¦
```

Fig. 6-24: Display of the receive job in the MC5 package

The display is brought up to date when "Start" has been activated, i.e. transferred data is displayed.

In locations where two MC5 packages are available it is possible to monitor both PLCs in parallel. If the program is correct data words DW0 to DW99 in the data modules DB11 and DB12 are equal and will be updated simultaneously (the only delay is caused by the transfer time).

Appendix

Appendix

A Technical data

Operating voltage	5 V DC ± 5%
Current consumption	1,5 A max. without transceiver
Processor	CPU 80486 SLC
Clock speed	33 MHz
RAM	1 MB
Serial interface	TTY
Configuration memory	60 KB CMOS
EPROM socket	for EPROMs 376 (EPROM)
Ethernet interface	AUI (IEEE 802.3) BNC RJ-45
CP interface	4 tiles with 1 KB each
Watchdog	triggerable
System-Bios	Quadtel, according to VIPA specifications
Slot space	1 plug-in slot
Weight	0,6 kg
Internal battery backup buffer	6 months

B Error messages from the NCS configuration program

General errors:

Nullpointer detected

An internal error has occurred during the transfer of parameters. Please contact us with the respective details.

Insufficient memory

DOS main memory is full, remove all unneccessary programs from main memory.

Message administration

The message file could not be found. The file "ncs.txt" is not available.

No message available

This error occurs when a text message is not available. Please contact us with the respective details.

Message file read error

An error occurred when a text message was read. Please check for errors in the file "ncs.txt". If necessary you should use an analyser tool.

Disk drive errors:

Access denied

Access to the respective file was refused. Please check whether you have access rights for the file.

Too many files, use a file filter

A directory contains too many files.

Drive not available

The specified drive was not available.

Drive is write protected

The specified drive is write protected.

File write error

Use an analyser to check the condition of your drive.

Connection configuration error:

SSNR/ANR already assigned

The entered combination of SSNR/ANR exists.

Invalid task number (valid range 1..199)

The task number (ANR) must be a number between 1-199 (please note any reserved ANRs).

No task number was specified

Every connection must be associated with a task number.

The tile offset was omitted

Each connection must have a tile offset.

No priority specified

Each connection must have a priority.

Multiple connection n, SSNR/ANR already assigned

The specified combination of SSNR/ANR in a multiple connection has already been specified.

Multiple connecion n, invalid task number (1..199)

The task number (ANR) for the multiple connection exists.

Multiple connection n, the tile offset was omitted

Every multiple connection must be associated with a tile offset.

Multiple connection n, no priority specified

Every multiple connection must have a priority.

Invalid tile base (valid range 0...252)

The tile base for the basic initialization is not valid.

No tile base specified

You did not specify a tile base address for the basic initialization.

This task type does not allow multiple connections

Multiple connections are only possible for a SEND or RECEIVE type connection.

Multiple connections only possible with Prio 1,2

You may only enter a multiple connection for Prio 1 or 2.

The specified task type is not suitable

The type of the multiple task must differ from that of the main task.

EPROM errors:

EPROM function terminated

EPROM programming was terminated by the user.

EPROM programming error

An error occurred while an eprom was being programmed. Please erase the EPROM or exchange it for a clean one, then repeat the operation.

EPROM programmer faulty

The controller on the ISA interface board is faulty.

EPROM programmer does not exist

The interface board is faulty or does not exist.

EPROM segment error

An error occurred while a file was being loaded. Please check the file "ncs.bib".

EPROM comparison error

Differences were detected between the contents of the EPROM and the original. Please erase the EPROM or exchange it for a clean one, then repeat the programming operation.

EPROM access error

Please erase the EPROM or exchange it for a clean one, then repeat the programming operation.

A lack of memory occurred when the EPROM function was being executed

No more DOS memory available. Remove all unnecessary programms from main memory.

EPROM-module library not available or defective

The EPROM-module library "ncs.bib" is not accessible or has been damaged. At the DOS prompt check that this file exists and that it is in order.

EPROM defective

Erase the EPROM or exchange it for a clean one and repeat the programming operation.

EPROM read error

Erase the EPROM or exchange it for a clean one and repeat the programming operation.

EPROM func. internal error (nullptr. detected)

This error should not occur. Please inform us if you should encounter this type of error.

EPROM area overflow

This error should not occur. Please inform us if you should encounter this type of error.

EPROM error in language area

This error should not occur. Please inform us if you should encounter this type of error.

No EPROM or incorrect EPROM inserted

The identifier of the EPROM is invalid. Check the type of EPROM and exchange if necessary.

Database file functions:

Module handler syspar eror

An error in the syspar area occurred when reading from or writing to the database file. This error can occur only when you load a file that is not a database file. It is also not possible to load database files that were created with version 1.00 of NCS.

Module handler boot error

An error occurred in the boot module when reading from or writing to the database file. This error should not occur. Please inform us if you should encounter this type of error.

Module handler initalization error

An error occurred during the initialization of the module handler. This error should not occur. Please inform us if you should encounter this type of error.

Module handler erase error

An error occurred when when erasing by means of the module handler. This error should not occur. Please inform us if you should encounter this type of error.

Module handler file error

An error occurred when writing to or reading from the database by means of the module handler. Check your drive. If necessary you should use an analyzer tool.

Module handler undefined error

This error should not occur. Please inform us if you should encounter this type of error.

C List of abbreviations

ANR	Task number
ANZW	Display word
CP	Communication Processor
CSMA/CD	Carrier Sense Multiple Access / Collision Detection
FTP	File Transfer Protocol
HTB	Handler module
IP	Internet Protocol
NCS	Network-Communication-System
OSI	Open System Interconnection
PAFE	Parameter configuration error
QVZ	Overdue acknowledgement
RTK	Real-Time-Kernel
SSNR	Interface number (tile offset)
TCP	Transport Control Protocol
TP	Twisted Pair
UDP	User Datagram Protocol

D List of figures

E List of tables

F Index

M. Stich