VIPA System 300V

CPU | Manual

HB130E_CPU | Rev. 11/50

December 2011

VIPA®
art of automation

Copyright © VIPA GmbH. All Rights Reserved.

This document contains proprietary information of VIPA and is not to be disclosed or used except in accordance with applicable agreements.

This material is protected by the copyright laws. It may not be reproduced, distributed, or altered in any fashion by any entity (either internal or external to VIPA), except in accordance with applicable agreements, contracts or licensing, without the express written consent of VIPA and the business management owner of the material.

For permission to reproduce or distribute, please contact:
VIPA, Gesellschaft für Visualisierung und Prozessautomatisierung mbH
Ohmstraße 4, D-91074 Herzogenaurach,Germany
Tel.: +49 (91 32) 744 -0
Fax.: +49 9132 744 1864
EMail: info@vipa.de
http://www.vipa.de

Note

Every effort has been made to ensure that the information contained in this document was complete and accurate at the time of publishing. Nevertheless, the authors retain the right to modify the information. This customer document describes all the hardware units and functions known at the present time. Descriptions may be included for units which are not present at the customer site. The exact scope of delivery is described in the respective purchase contract.

CE Conformity

Hereby, VIPA GmbH declares that the products and systems are in compliance with the essential requirements and other relevant provisions of the following directives:

- 2004/108/EC Electromagnetic Compatibility Directive
- 2006/95/EC Low Voltage Directive

Conformity is indicated by the CE marking affixed to the product.

Conformity Information

For more information regarding CE marking and Declaration of Conformity (DoC), please contact your local VIPA customer service organization.

Trademarks

VIPA, SLIO, System 100V, System 200V, System 300V, System 300S, System 400V, System 500S and Commander Compact are registered trademarks of VIPA Gesellschaft für Visualisierung und Prozessautomatisierung mbH.

SPEED7 is a registered trademark of profichip GmbH.

SIMATIC, STEP, SINEC, S7-300 and S7-400 are registered trademarks of Siemens AG.

Microsoft und Windows are registered trademarks of Microsoft Inc., USA.

Portable Document Format (PDF) and Postscript are registered trademarks of Adobe Systems, Inc.

All other trademarks, logos and service or product marks specified herein are owned by their respective companies.

Information product support

Contact your local VIPA Customer Service Organization representative if you wish to report errors or questions regarding the contents of this document. If you are unable to locate a customer service center, contact VIPA as follows:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telefax:+49 9132 744 1204
EMail: documentation@vipa.de

Technical support

Contact your local VIPA Customer Service Organization representative if you encounter problems with the product or have questions regarding the product. If you are unable to locate a customer service center, contact VIPA as follows:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telephone: +49 9132 744 1150 (Hotline)
EMail: support@vipa.de

Contents

About this Manual

This manual describes the operation of the CPU 31x in the System 300V from VIPA. The text provides details on the hardware, the programming and the functions integrated into the unit as well as Profibus and TCP/IP applications.

Overview

Chapter 1: Basics

This introduction includes recommendations on the handling of the modules of the VIPA System 300V and introduces you to central res. decentral automation systems.

Chapter 2: Assembly and installation guidelines

All information that you need for installation and cabling of a PLC with components of the System 300V.

Chapter 3: Hardware description

The CPUs 314 to 317 are available in different variants. This chapter describes the specifications of these variants.

Chapter 4: Deployment CPU 31x

General information for operating the CPU31x like address allocation, operating modes, usage of the MMC and the MP-Interface.

Chapter 5: Deployment CPU 31x with Profibus

This chapter contains an description of the installation and configuration of the System 300V CPUs via Profibus.

Chapter 6: Deployment CPU 31x with TCP/IP

This chapter describes the operating of the CPU31xNET and the communication via TCP/IP.

Objective and contents

This manual describes the CPUs 31x from VIPA which can be used at the System 300V. It contains a description of structure, project engineering and technical data.

Target audience

The manual is targeted at users who have a background in automation technology.

Structure of the manual

The manual consists of chapters. Every chapter provides a self-contained description of a specific topic.

Guide to the document

The following guides are available in the manual:

- an overall table of contents at the beginning of the manual
- an overview of the topics for every chapter
- an index at the end of the manual.

Availability

The manual is available in:

- printed form, on paper
- in electronic form as PDF-file (Adobe Acrobat Reader)

Icons Headings

Important passages in the text are highlighted by following icons and headings:

⚠

Danger!

Immediate or likely danger.

Personal injury is possible.

⚠

Attention!

Damages to property is likely if these warnings are not heeded.

ℹ

Note!

Supplementary information and useful tips.

Safety information

Applications conforming with specifications

The CPUs 31x are constructed and manufactured for:

- System 300 components from VIPA and Siemens
- communication and process control
- general control and automation applications
- industrial applications
- operation within the environmental conditions specified in the technical data
- installation into a cubicle

Danger!

This device is not certified for applications in

- in explosive environments (EX-zone)

Documentation

The manual must be available to all personnel in the

- project design department
- installation department
- commissioning
- operation

The following conditions must be met before using or commissioning the components described in this manual:

- Modification to the process control system should only be carried out when the system has been disconnected from power!
- Installation and modifications only by properly trained personnel
- The national rules and regulations of the respective country must be satisfied (installation, safety, EMC ...)

Disposal

National rules and regulations apply to the disposal of the unit!

Chapter 1 Basics

Outline

Main theme of this chapter is to give you an overview about the System 300V from VIPA. We will outline the possibilities of the installation of central res. decentral systems.

This chapter also contains general information about the System 300V like measurements, hints for installation and the environmental conditions.

Content

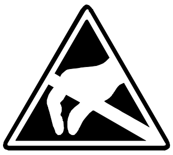
Topic **Page**

Safety Information for Users

Handling of electrostatically sensitive modules

VIPA modules make use of highly integrated components in MOS-Technology. These components are extremely sensitive to over-voltages that can occur during electrostatic discharges.

The following symbol is attached to modules that can be destroyed by electrostatic discharges.



The symbol is located on the module, the module rack or on packing material and it indicates the presence of electrostatically sensitive equipment.

It is possible that electrostatically sensitive equipment is destroyed by energies and voltages that are far less than the human threshold of perception. These voltages can occur where persons do not discharge themselves before handling electrostatically sensitive modules and they can damage components thereby, causing the module to become inoperable or unusable. Modules, damaged in this way, are normally not immediately recognized. The according error may occur only after a while of operation.

Modules that have been damaged by electrostatic discharges can fail after a temperature change, mechanical shock or changes in the electrical load.

Only the consequent implementation of protection devices and meticulous attention to the applicable rules and regulations for handling the respective equipment can prevent failures of electrostatically sensitive modules.

Shipping of modules

Modules must be shipped in the original packing material.

Measurements and alterations on electrostatically sensitive modules

When you are conducting measurements on electrostatically sensitive modules you should take the following precautions:

- Floating instruments must be discharged before use.
- Instruments must be grounded.

Modifying electrostatically sensitive modules you should only use soldering irons with grounded tips.

Attention!

Personnel and instruments should be grounded when working on electrostatically sensitive modules.

General description of the System 300V

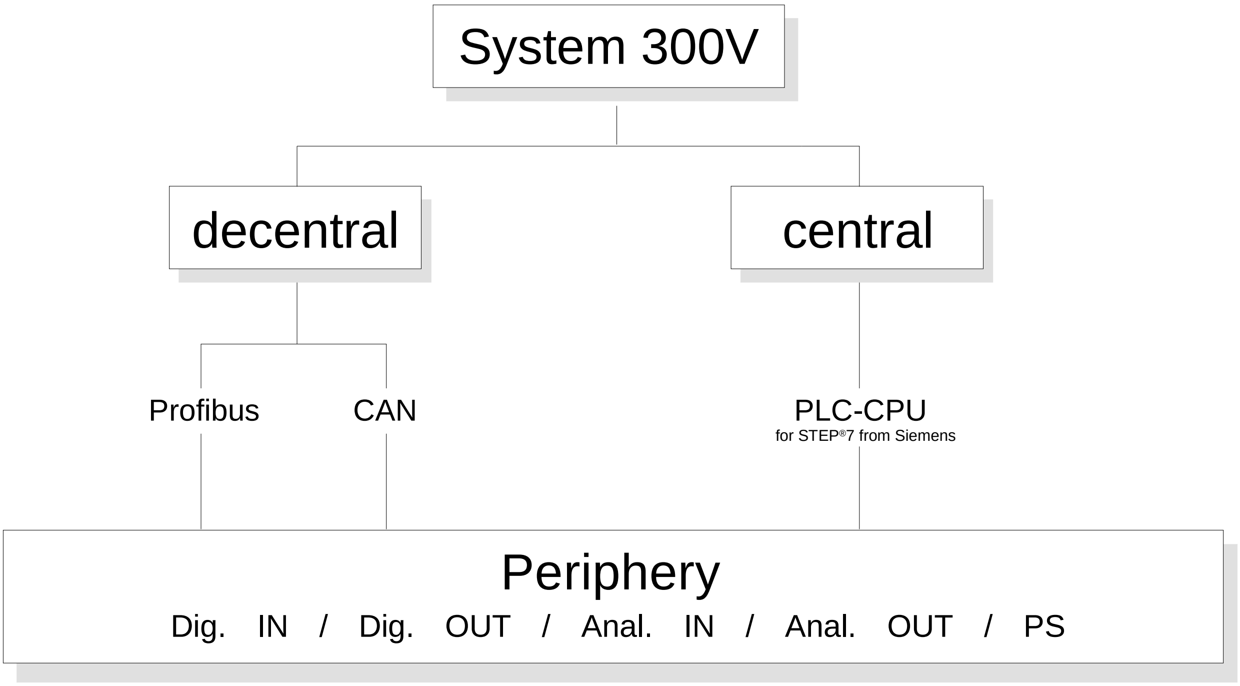
The System 300V

The System 300V is a modular automation system for middle and high performance needs, that you can use either distributed or non-distributed. The single modules are directly clipped to a 530 mm backplane and are connected together with the help of bus clips at the backside.

The single modules of the VIPA System 300V are design compatible to Siemens. Due to the compatible backplane bus it is no problem to mix the modules from VIPA and Siemens.

The CPUs of the System 300V are instruction set compatible to S7-300 from Siemens. The CPUs are programmed via the VIPA programming software WinPLC7 or the SIMATIC manager from Siemens or other available programming tools.

The following picture illustrates the performance range of the System 300V:



Components

Central system

The System 200V series consists of a number of PLC-CPUs. These are programmed in STEP®7 from Siemens. Herefore you may use WinPLC7 from VIPA or the SIMATIC manager from Siemens.

CPUs with integrated Ethernet interfaces or additional serial interfaces simplify the integration of the PLC into an existing network or the connection of additional peripheral equipment.

The application program is saved in Flash or an additional plug-in memory module.

Because of the automatic addressing, up to 32 peripheral modules can be called by the System 300V CPUs.

Decentral system

In combination with a Profibus DP master and slave the PLC-CPUs or the PC-CPU form the basis for a Profibus-DP network in accordance with DIN 19245-3.

The DP network can be configured with the hardware configurator from Siemens. Together with the hardware configuration you transfer your project into the CPU via MPI. Another component of the decentral system is the CAN-Slave. It allows the link-up to the fieldbus system CANopen.

Peripheral modules

A large number of peripheral modules are available from VIPA, for example digital as well as analog inputs/outputs.

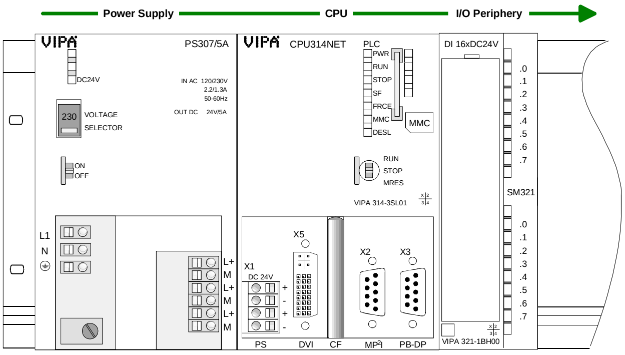
These peripheral modules can be deployed central as well as decentral.

Dimensions/ Weight

- Profile rail 530mm
- Peripheral modules with recessed labeling
- Dimensions of the basic enclosure:
  1tier width: (WxHxD) in mm: 40x125x120
  2tier width: (WxHxD) in mm: 80x125x120
  3tier width: (WxHxD) in mm: 120x125x120

Installation

Please regard that the power supply and header modules like CPUs and couplers may only plugged-in at the left side.



Reliability

- Wiring by means of spring pressure connections (CageClamps) at the front connector
- Core cross-section 0.08...2.5mm^2 or 1.5 mm^2
- Total isolation of the wiring at module change
- Potential separation of all modules to the backplane bus
- Burst/ESD acc. IEC 61000-4-2/IEC 61000-4-4 (up to level 3)
- Shock resistance acc. IEC 60068-2-6 / IEC 60068-2-27 (1G/12G)

Environmental conditions

- Operating temperature: 0 ... +60°C
- Storage temperature: -25 ... +70°C
- Relative humidity: 5...95% without condensation
- Ventilation by means of a fan is not required

Green Cable for project engineering

For project engineering of your DP slave you may transfer your projects from your PC to the CPU serial via MPI by using the "Green Cable". Please also regard the hints to the Green Cable in this chapter!

Integrated power supply

Every Profibus slave has an internal power supply. This power supply requires DC 24V. In addition to the electronics on the bus coupler, the supply voltage is also used to power any modules connected to the backplane bus. Please note that the maximum current that the integrated power supply can deliver to the backplane bus is 3.5A.

The power supply is protected against reverse polarity and overcurrent.

Compatibility

The digital in-/output modules of the System 300V from VIPA are pin and function compatible to Siemens.

The project engineering happens in the SIMATIC manager from Siemens.

Note!

For programming of a System 300V CPU from VIPA please use always the **CPU 315-2DP (6ES7 315-2AF03 V1.2)** from Siemens in the hardware catalog.

Please note the Profibus address 1 of the CPU 31x is system dependent reserved.

For the project engineering, a thorough knowledge of the Siemens SIMATIC manager and the hardware configurator is required!

Chapter 2 Assembly and installation guidelines

Outline

In this chapter you will find all information, required for the installation and the cabling of a process control with the components of the System 300V.

Content

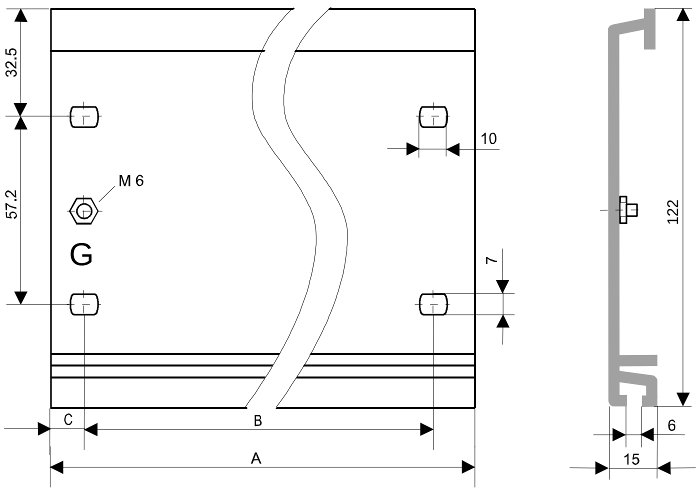
Overview

General

The single modules are directly installed on a profile rail and connected via the backplane bus coupler. Before installing the modules you have to clip the backplane bus coupler to the module from the backside.

The backplane bus coupler are included in the delivery of the peripheral modules.

Profile rail

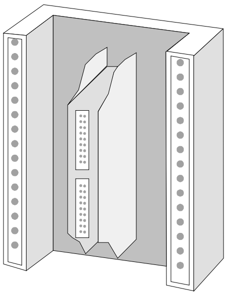


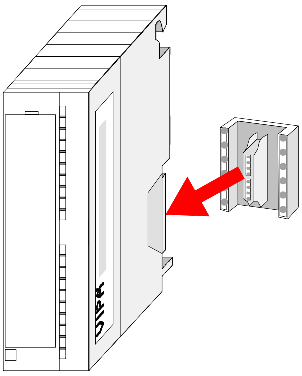
Order number	A	B	C
VIPA 390-1AB60	160mm	140mm	10mm
VIPA 390-1AE80	482mm	466mm	8.3mm
VIPA 390-1AF30	530mm	500mm	15mm
VIPA 390-1AJ30	830mm	800mm	15mm
VIPA 390-9BC00*	2000mm	no Drillings	15mm

* Unit pack: 10 pieces

Bus connector

For the communication between the modules the System 300V uses a backplane bus connector. The backplane bus connector are included in the delivering of the peripheral modules and are clipped at the module from behind before installing it to the profile rail.





Installation dimensions

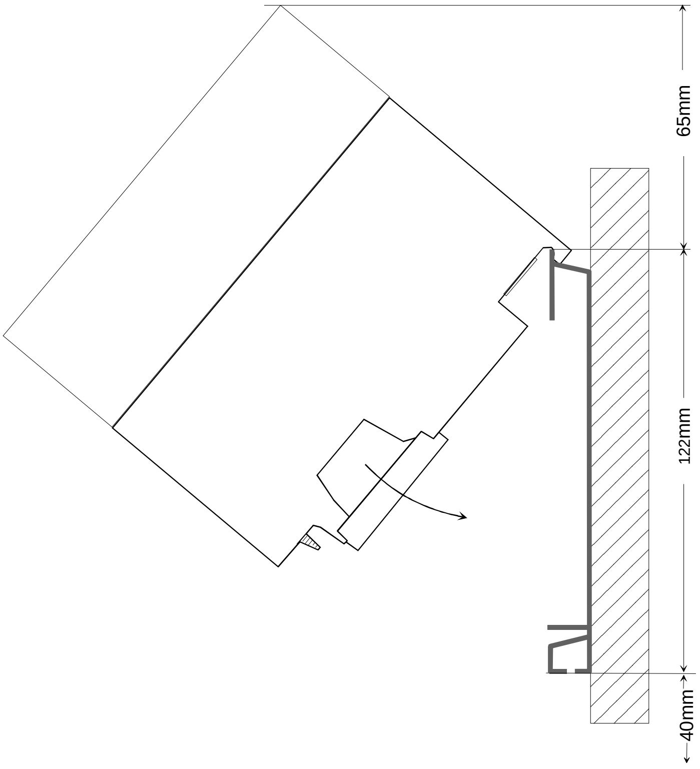
Overview

Here follows all the important dimensions of the System 300V.

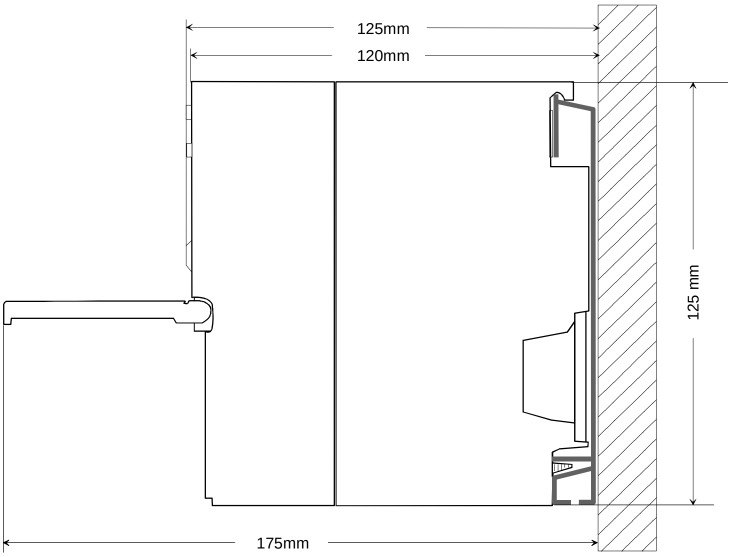
Dimensions Basic enclosure

1tier width (WxHxD) in mm: 40 x 125 x 120
2tier width (WxHxD) in mm: 80 x 125 x 120
3tier width (WxHxD) in mm: 120 x 125 x 120

Dimensions



Installation dimensions



Installation at the profile rail

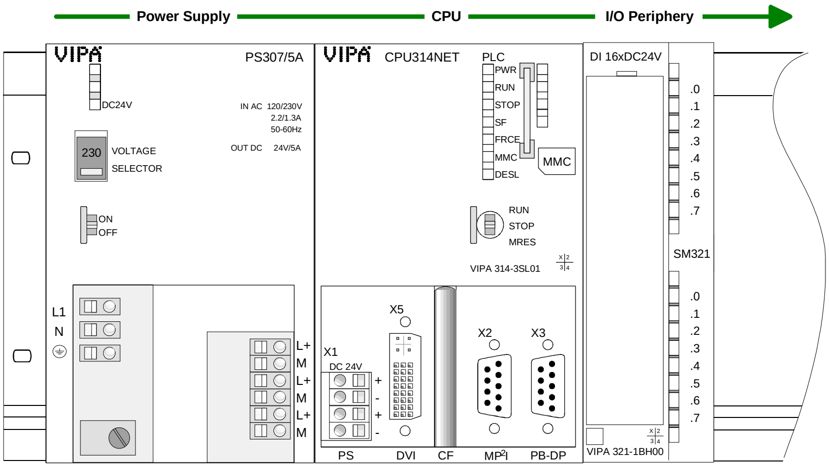
Structure:

You may install the System 300V as well horizontal as vertical. Please regard the allowed environment temperatures:

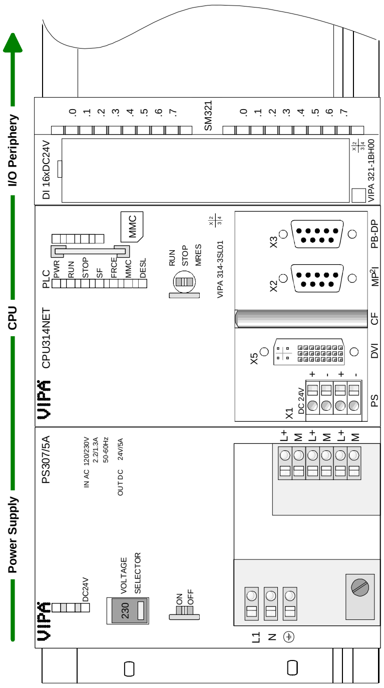
- horizontal structure: from 0 to 60°
- vertical structure: from 0 to 40°

The horizontal structure always starts at the left side with the power supply and the CPU, then you plug-in the peripheral modules beside to the right.

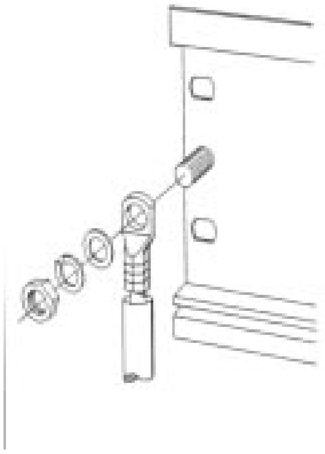
You may plug-in maximum 32 peripheral modules to the CPU.



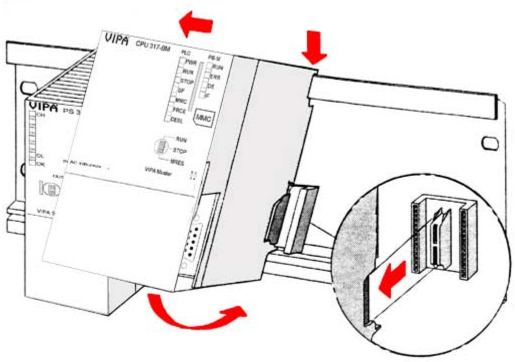
The vertical structure is turned for 90° against the clockwise direction.



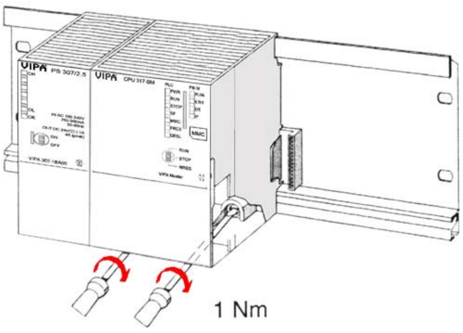
Approach



- Bolt the profile rail with the background (screw size: M6), so that you still have minimum 65mm space above and 40mm below the profile rail.
- If the background is a grounded metal or device plate, please look for a low-impedance connection between profile rail and background.
- Connect the profile rail with the protected earth conductor. For this purpose there is a bolt with M6-thread.
- The minimum cross-section of the cable to the protected earth conductor has to be 10mm^2.



- Stick the power supply to the profile rail and pull it to the left side to the grounding bolt of the profile rail.
- Fix the power supply by screwing.
- Take a bus coupler and click it at the CPU from behind like shown in the picture.
- Stick the CPU to the profile rail right from the power supply and pull it to the power supply.



- Click the CPU downwards and bolt it like shown.
- Repeat this procedure with the peripheral modules, by clicking a backplane bus coupler, stick the module right from the modules you've already fixed, click it downwards and connect it with the backplane bus coupler of the last module and bolt it.

Danger!

- Before installing or overhauling the System 300V, the power supplies must be disconnected from voltage (pull the plug or remove the fuse)!
- Installation and modifications only by properly trained personnel!

Cabling

Overview

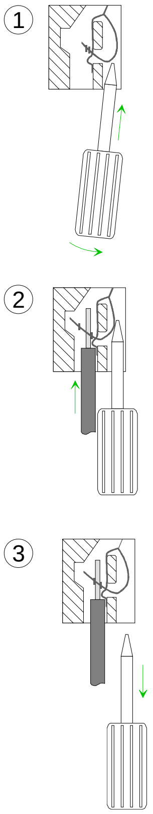
The power supplies and CPUs are exclusively delivered with CageClamp contacts. For the signal modules the front connectors are available from VIPA with screw contacts. In the following all connecting types of the power supplies, CPUs and input/output modules are described.

Danger!

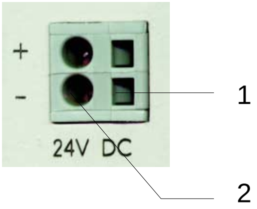
- Before installation or overhauling, the power supplies must be disconnected from voltage (pull the plug or remove the fuse)!
- Installation and modifications only by properly trained personnel!

CageClamp technology (gray)

For the cabling of power supplies, bus couplers and parts of the CPU, gray connectors with CageClamp technology are used.



You may connect wires with a cross-section of 0.08mm^2 to 2.5mm^2. You can use flexible wires without end case as well as stiff wires.



[1] Rectangular opening for screwdriver
[2] Round opening for wires

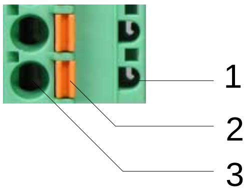
The picture on the left side shows the cabling step by step from top view.

- To conduct a wire you plug a fitting screwdriver obliquely into the rectangular opening like shown in the picture.
- To open the contact spring you have to push the screwdriver in the opposite direction and hold it.
- Insert the insulation striped wire into the round opening. You may use wires with a cross-section from 0.08mm^2 to 2.5mm^2.
- By removing the screwdriver the wire is connected safely with the plug connector via a spring.

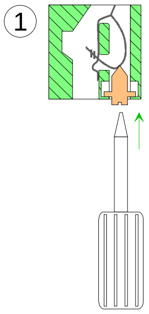
CageClamp technology (green)

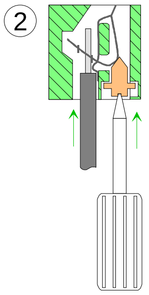
For the cabling of e.g. the power supply of a CPU, green plugs with CageClamp technology are deployed.

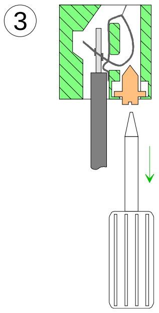
Here also you may connect wires with a cross-section of 0.08mm^2 to 2.5mm^2. You can use flexible wires without end case as well as stiff wires.



[1] Test point for 2mm test tip
[2] Locking (orange) for screwdriver
[3] Round opening for wires







The picture on the left side shows the cabling step by step from top view.

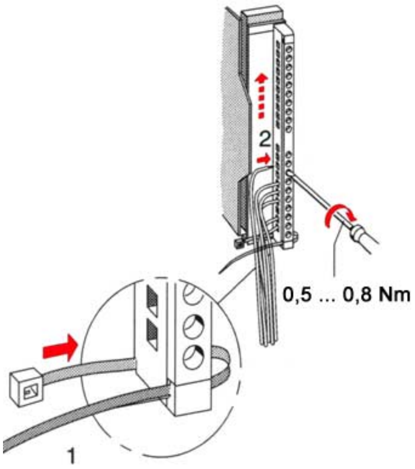
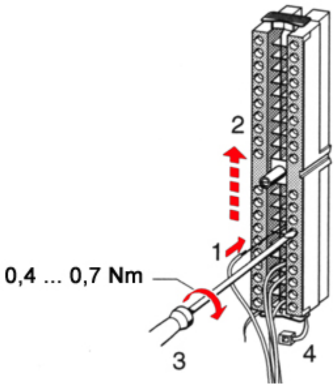
- For cabling you push the locking vertical to the inside with a suiting screwdriver and hold the screwdriver in this position.
- Insert the insulation striped wire into the round opening. You may use wires with a cross-section from 0.08mm^2 to 2.5mm^2.
- By removing the screwdriver the wire is connected safely with the plug connector via a spring.

Note!
In opposite to the gray connection clamp from above, the green connection clamp is realized as plug that can be clipped off carefully even if it is still cabled.

Front connectors of the in-/output modules

In the following the cabling of the three variants of the front-facing connector is shown:

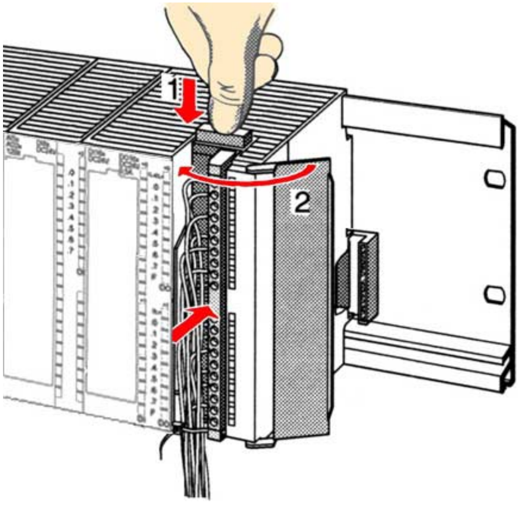
For the I/O modules the following plugs are available at VIPA:

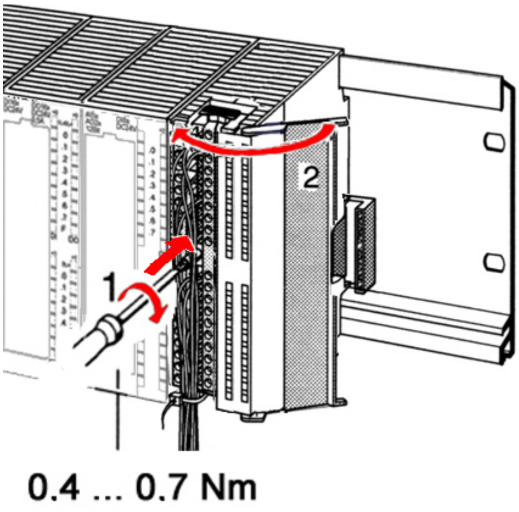
20pole screw connection VIPA 392-1AJ00	**40pole screw connection** VIPA 392-1AM00
Open the front flap of your I/O module.	
Bring the front connector in cabling position. Herefore you plug the front connector on the module until it locks. In this position the front connector juts out of the module and has no contact yet.	
Deisolate your wires. If needed, use core end cases.	
Thread the included cable binder into the front connector.	
If you want to lead out your cables from the bottom of the module, start with the cabling from bottom to top, res. from top to bottom, if the cables should be led out at the top.	
Bolt also the connection screws of not cabled screw clamps.	
	Put the included cable binder around the cable bundle and the front connector. 
Fix the cable binder for the cable bundle.	

continued ...

... continue

<table>
<tr><th>20pole screw connection</th><th>40pole screw connection</th></tr>
<tr><td>Push the release key at the front connector on the upper side of the module and at the same time push the front connector into the module until it locks.


</td><td>Bolt the fixing screw of the front connector.


</td></tr>
<tr><td colspan="2">Now the front connector is electrically connected with your module.</td></tr>
<tr><td colspan="2">Close the front flap.</td></tr>
<tr><td colspan="2">Fill out the labeling strip to mark the single channels and push the strip into the front flap.</td></tr>
</table>

Installation Guidelines

General

The installation guidelines contain information about the interference free deployment of System 300V systems. There is the description of the ways, interference may occur in your control, how you can make sure the electromagnetic digestibility (EMC), and how you manage the isolation.

What means EMC?

Electromagnetic digestibility (EMC) means the ability of an electrical device, to function error free in an electromagnetic environment without being interferenced res. without interferencing the environment.

All System 300V components are developed for the deployment in hard industrial environments and fulfill high demands on the EMC. Nevertheless you should project an EMC planning before installing the components and take conceivable interference causes into account.

Possible interference causes

Electromagnetic interferences may interfere your control via different ways:

- Fields
- I/O signal conductors
- Bus system
- Current supply
- Protected earth conductor

Depending on the spreading medium (lead bound or lead free) and the distance to the interference cause, interferences to your control occur by means of different coupling mechanisms.

One differs:

- galvanic coupling
- capacitve coupling
- inductive coupling
- radiant coupling

Basic rules for EMC

In the most times it is enough to take care of some elementary rules to guarantee the EMC. Please regard the following basic rules when installing your PLC.

- Take care of a correct area-wide grounding of the inactive metal parts when installing your components.
    - Install a central connection between the ground and the protected earth conductor system.
    - Connect all inactive metal extensive and impedance-low.
    - Please try not to use aluminum parts. Aluminum is easily oxidizing and is therefore less suitable for grounding.
- When cabling, take care of the correct line routing.
    - Organize your cabling in line groups (high voltage, current supply, signal and data lines).
    - Always lay your high voltage lines and signal res. data lines in separate channels or bundles.
    - Route the signal and data lines as near as possible beside ground areas (e.g. suspension bars, metal rails, tin cabinet).
- Proof the correct fixing of the lead isolation.
    - Data lines must be laid isolated.
    - Analog lines must be laid isolated. When transmitting signals with small amplitudes the one sided laying of the isolation may be favorable.
    - Lay the line isolation extensively on a isolation/protected earth conductor rail directly after the cabinet entry and fix the isolation with cable clamps.
    - Make sure that the isolation/protected earth conductor rail is connected impedance-low with the cabinet.
    - Use metallic or metallized plug cases for isolated data lines.
- In special use cases you should appoint special EMC actions.
    - Wire all inductivities with erase links, that are not addressed by the System 300V modules.
    - For lightening cabinets you should prefer incandescent lamps and avoid luminescent lamps.
- Create an homogeneous reference potential and ground all electrical operating supplies when possible.
    - Please take care for the targeted employment of the grounding actions. The grounding of the PLC is a protection and functionality activity.
    - Connect installation parts and cabinets with the System 300V in star topology with the isolation/protected earth conductor system. So you avoid ground loops.
    - If potential differences between installation parts and cabinets occur, lay sufficiently dimensioned potential compensation lines.

Isolation of conductors

Electrical, magnetical and electromagnetical interference fields are weakened by means of an isolation, one talks of absorption.

Via the isolation rail, that is connected conductive with the rack, interference currents are shunt via cable isolation to the ground. Hereby you have to make sure, that the connection to the protected earth conductor is impedance-low, because otherwise the interference currents may appear as interference cause.

When isolating cables you have to regard the following:

- If possible, use only cables with isolation tangle.
- The hiding power of the isolation should be higher than 80%.
- Normally you should always lay the isolation of cables on both sides. Only by means of the both-sided connection of the isolation you achieve a high quality interference suppression in the higher frequency area.

  Only as exception you may also lay the isolation one-sided. Then you only achieve the absorption of the lower frequencies. A one-sided isolation connection may be convenient, if:

  - the conduction of a potential compensating line is not possible
  - analog signals (some mV res. µA) are transferred
  - foil isolations (static isolations) are used.
- With data lines always use metallic or metallized plugs for serial couplings. Fix the isolation of the data line at the plug rack. Do not lay the isolation on the PIN 1 of the plug bar!
- At stationary operation it is convenient to deisolate the isolated cable interruption free and lay it on the isolation/protected earth conductor line.
- To fix the isolation tangles use cable clamps out of metal. The clamps must clasp the isolation extensively and have well contact.
- Lay the isolation on an isolation rail directly after the entry of the cable in the cabinet. Lead the isolation further on to the System 300V module and **don't** lay it on there again!

Please regard at installation!

At potential differences between the grounding points, there may be a compensation current via the isolation connected at both sides.

Remedy: Potential compensation line

Chapter 3 Hardware description

Outline

The CPUs 31x are available in different versions that are described in this chapter. In addition to the hardware description the chapter also contains installation and commissioning instructions and applications for the memory modules of the System 300V family.

The technical data conclude the chapter.

Content

Introduction

General

A CPU is an intelligent module. Here your driver software is processed. Depending on your performance needs, you are able to choose between 4 different CPUs. The more power the chosen CPU has, the more user memory it has. This is composed of load memory for the project engineering and work memory for the program compiled from the project engineering.

These CPUs are conceived for middle and large applications with integrated 24V-Power supply. Additionally every CPU has a MMC slot at the frontside. Every CPU has a MPI slot and is instruction compatible to STEP$^{®}$7 from Siemens.

The performance of the CPU series 314...317 corresponds to the S7-300 CPU-series from Siemens (except CPU 318).

This CPUs series allows you the access to the peripheral modules of the System 300V. Via standardized commands and programs you may request sensors and control actors. The CPU is parameterized by means of the integrated MP^{2}I-Interface.

Types

The CPU 31x family is available in 4 different CPU versions with each 4 variants:

- **CPU 31x** SPS-CPU with DP slave
- **CPU 31xDPM** SPS-CPU with DP slave/master
- **CPU 31xNET** SPS-CPU with DP slave, Ethernet TCP/IP-Interface, 4port Switch
- **CPU 31xNET** SPS-CPU with DP slave/master, Ethernet TCP/IP-Interface, 4port Switch

Versions

The following CPU versions are available:

CPU	Work memory in kBytes	Load memory in kBytes
CPU 314	96	144
CPU 315	192	256
CPU 316	256	512
CPU 317	512	1024

Note!

The further description in this manual refers to the complete CPU family CPU 31x. The CPUs 314...317 are functionally identical and they only differ in their range of memory.

Operating structure of a CPU

General

A CPU supports the following modes of operation:

- cyclic operation
- timer processing
- alarm controlled operation
- priority based processing

Cyclic processing

Cyclic processing represents the major portion of all the processes that are executed in the CPU. Identical sequences of operations are repeated in a never ending cycle.

Timer processing

Where a process requires control signals at constant intervals you can initiate certain operations based upon a **timer**, e.g. not critical monitoring functions at one-second intervals.

Alarm controlled processing

If a process signal requires a quick response you would allocate this signal to an **alarm controlled** procedure. An alarm can activate a procedure in your program.

Priority based processing

The above processes are handled by the CPU in accordance with their **priority**. Since a timer or an alarm event requires a quick reaction, the CPU will interrupt the cyclic processing when these high-priority events occur to react to the event. Cyclic processing will resume, once the reaction has been processed. This means that cyclic processing has the lowest priority.

CPU 31x Applications

The program that is present in every CPU is divided as follows:

- System routine
- User application

System routine

The system routine organizes all those functions and procedures of the CPU that are not related to a specific control application.

User application

This consists of all the functions that are required for the processing of a specific control application. The operating modules provide the interfaces to the system routines.

Operands of the CPU 31x

The following series of operands is available for programming the CPU 31x:

- Process image and periphery
- Bit memory
- Timers and counters
- Data blocks

Process image and periphery

The user application can quickly access the process image of the inputs and outputs PAA/PAE. You may manipulate the following types of data:

- individual Bits
- Bytes
- Words
- Double words

You may also gain direct access to peripheral modules via the bus from user application. The following types of data are available:

- Bytes
- Words
- Blocks

Bit memory

The bit memory is an area of memory that is accessible by means of certain operations. Bit memory is intended to store frequently used working data.

You may access the following types of data:

- individual Bits
- Bytes
- Words
- Double words

Timers and counters

In your program you may load cells of the timer with a value between 10ms and 9990s. As soon as the user application executes a start-operation, the value of this timer is decremented by the interval that you have specified until it reaches zero.

You may load counter cells with an initial value (max. 999) and increment or decrement these when required.

Data blocks

A data block contains constants or variables in the form of bytes, words or double words. You may always access the current data block by means of operands.

You may access the following types of data:

- individual Bits
- Bytes
- Words
- Double words

Function security of the VIPA CPUs

Security mechanisms

The CPUs include security mechanisms like a Watchdog (100ms) and a parameterizable cycle time surveillance (parameterizable min. 1ms) that stop res. execute a RESET at the CPU in case of an error and set it into a defined STOP state.

The VIPA CPUs are developed function secure and have the following system properties:

Event	concerns	Effect
RUN → STOP	general	BASP (**B**efehls-**A**usgabe-**Sp**erre, i.e. command output lock) is set.
	central digital outputs	The outputs are set to 0V.
	central analog outputs	The voltage supply for the output channels is switched off.
	decentral outputs	The outputs are set to 0V.
	decentral inputs	The inputs are read constantly from the slave and the recent values are put at disposal.
STOP → RUN res. Power on	general	First the PII is deleted, the call of the OB 100 follows. After the execution of the OB, the BASP is set back and the cycle starts with: Delete PIO → Read PII → OB 1.
	central analog outputs	The behavior of the outputs at restart can be preset.
	decentral inputs	The inputs are read constantly from the slave and the recent values are put at disposal.
RUN	general	The program execution happens cyclically and can therefore be foreseen: Read PII → OB 1 → Write PIQ.

PII: = Process image inputs

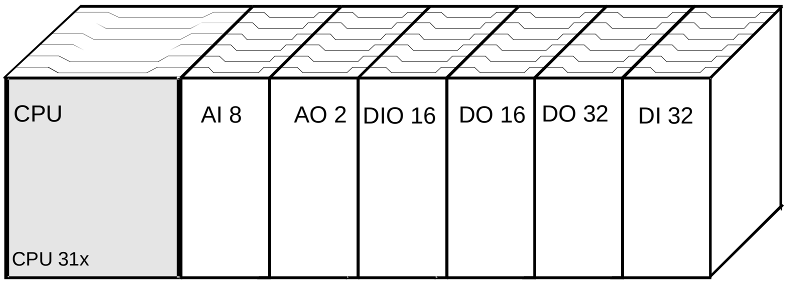
PIQ: = Process image outputs

Application fields

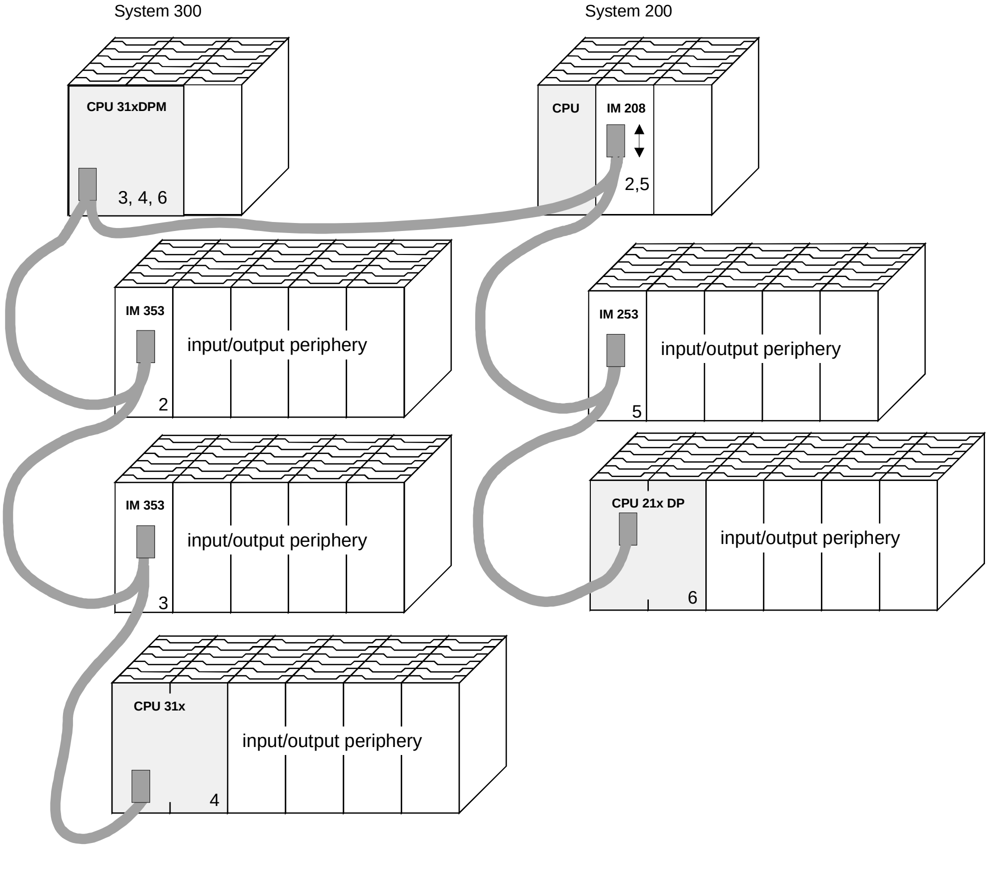
With the help of this CPU series you can access the peripheral modules of the System 300V from VIPA. You may request sensors and control actors via standardized commands and programs. One CPU is able to manage a maximum of 32 modules in one line.

Application example

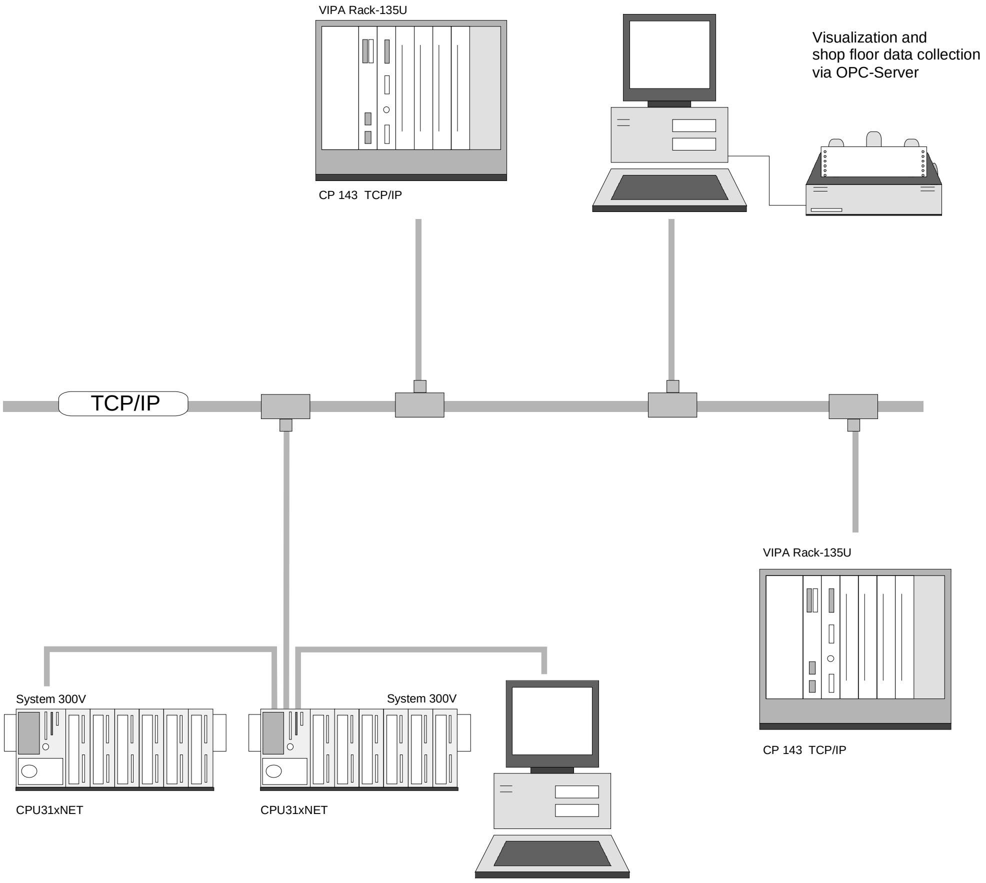
Central compact solution



Decentral system via Profibus



Operation via TCP/IP

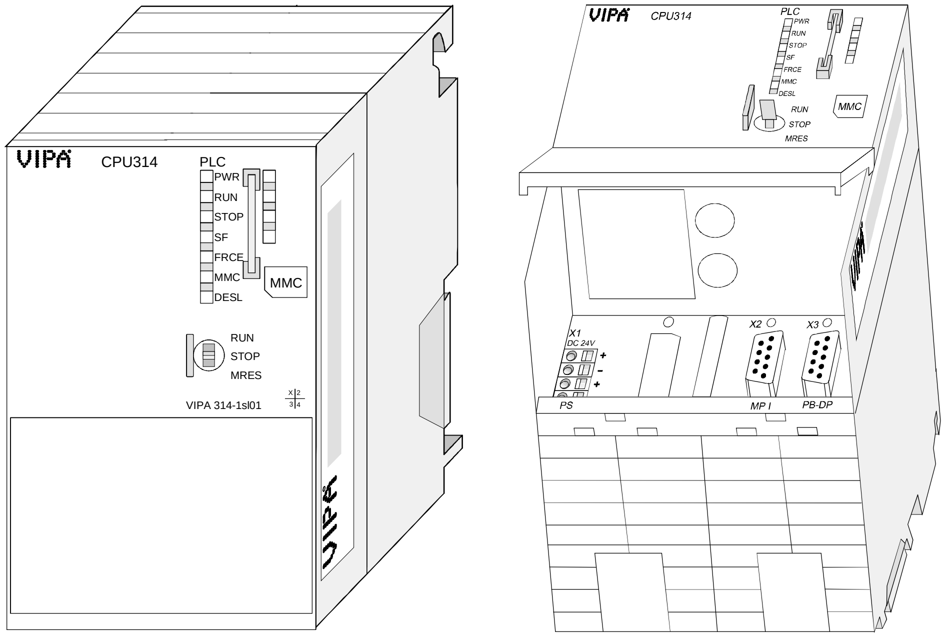


System overview

CPU 31x

- Instruction compatible to STEP®7 by Siemens
- Configuration via the Siemens SIMATIC Manager
- Integrated 24V Power supply
- MP^{2}Interface: MPI (4 static / 8 dynamic connections) or serial point-to-point coupling to the PC
- LEDs for status and diagnostics
- Battery buffer for RAM and clock
- Slot for MMC as external storage medium for programs and data
- Internal Flash ROM for saving the project engineering at battery failure
- Possibility of firmware update via MMC
- Slot for CompactFlash® Type II (only at CP version)
- Integrated controller for the System 300V peripheral modules
- Integrated Profibus-DP-Slave
- E/A address range digital/analog 1024Byte
- Work memory 96 ... 512kByte
- Load memory 144 ... 1024kByte
- 256 Timer
- 256 Counter
- 8192 Bit memories

CPU 31x



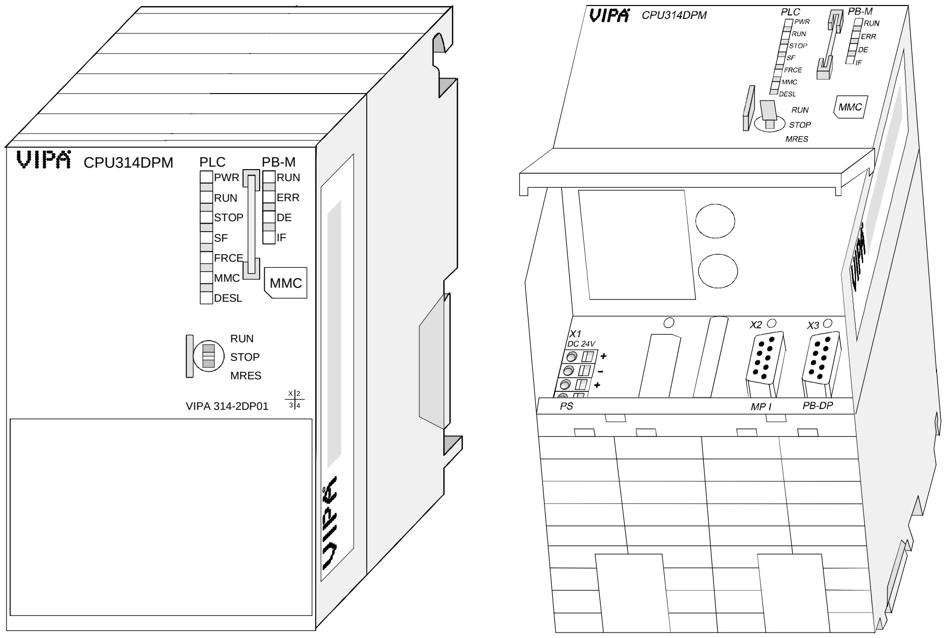
Ordering data CPU 31x

Type	Order number	Description
CPU 314	VIPA 314-1SL01	SPS CPU 314 with DP slave 96kB Work / 144kB Load memory
CPU 315	VIPA 315-1SL01	SPS CPU 315 with DP slave 192kB Work / 256kB Load memory
CPU 316	VIPA 316-1SL01	SPS CPU 316 with DP slave 256kB Work / 512kB Load memory
CPU 317	VIPA 317-1SL01	SPS CPU 317 with DP-Slave 512kB Work-/ 1MB Load memory

CPU 31xDPM

Like CPU 31x additionally with:

- integrated Profibus-DP master with Auto RUN instead of DP slave
- status display for Profibus-DP master



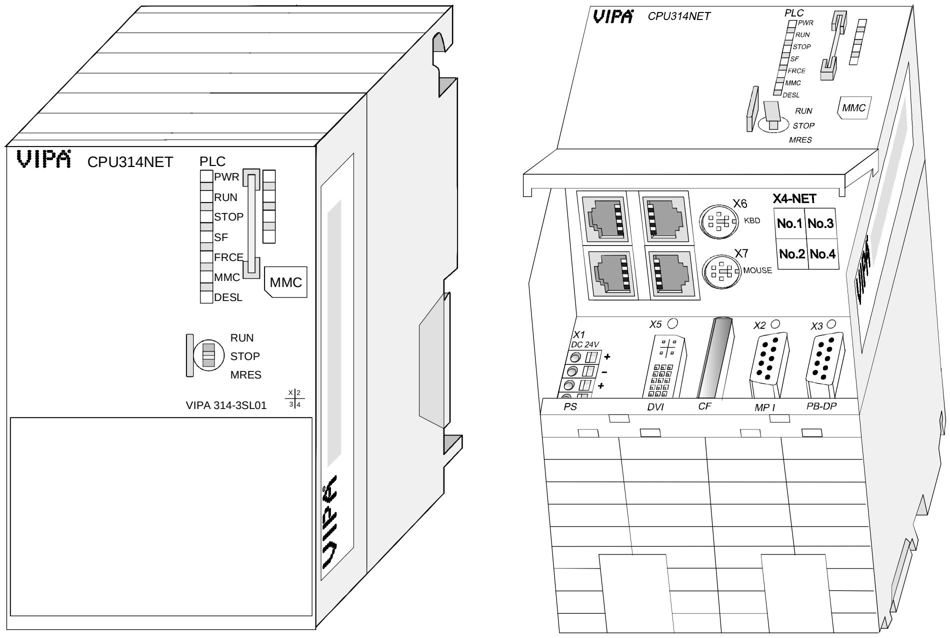
Ordering data CPU 31xDPM

Type	Order number	Description
CPU 314	VIPA 314-2DP01	SPS CPU 314 with DP master 96kB Work / 144kB Load memory
CPU 315	VIPA 315-2DP01	SPS CPU 315 with DP master 192kB Work / 256kB Load memory
CPU 316	VIPA 316-2DP01	SPS CPU 316 with DP master 256kB Work / 512kB Load memory
CPU 317	VIPA 317-2DP01	SPS CPU 317 with DP master 512kB Work / 1MB Load memory

CPU 31xNET

Like CPU 31x additionally with:

- integrated TCP/IP-CP with Auto RUN
- integrated 4port Switch



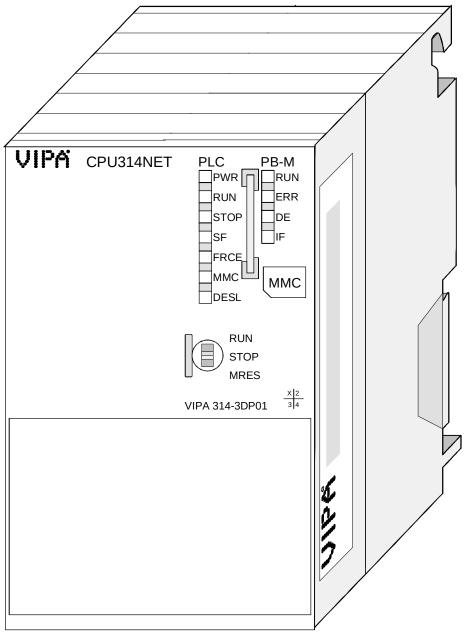
Ordering data CPU 31xNET

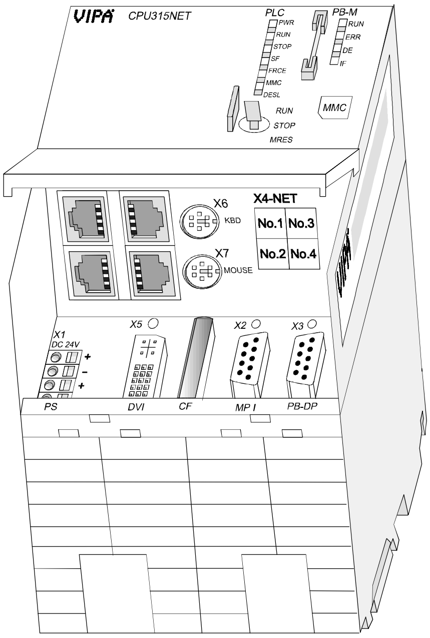
Type	Order number	Description
CPU 314	VIPA 314-3SL01	SPS CPU 314 with DP slave, Ethernet TCP/IP-Interface, 4port Switch, 96kB Work / 144kB Load memory
CPU 315	VIPA 315-3SL01	SPS CPU 315 with DP slave, Ethernet TCP/IP-Interface, 4port Switch, 192kB Work / 256kB Load memory
CPU 316	VIPA 316-3SL01	SPS CPU 316 with DP slave, Ethernet TCP/IP-Interface, 4port Switch, 256kB Work / 512kB Load memory
CPU 317	VIPA 317-3SL01	SPS CPU 317 with DP slave, Ethernet TCP/IP-Interface, 4port Switch, 512kB Work / 1MB Load memory

CPU 31xNET with DP master

Like CPU 31x additionally with:

- integrated TCP/IP-CP with Auto RUN
- integrated 4port Switch
- integrated Profibus-DP master with Auto RUN instead of DP slave
- status display for Profibus-DP master



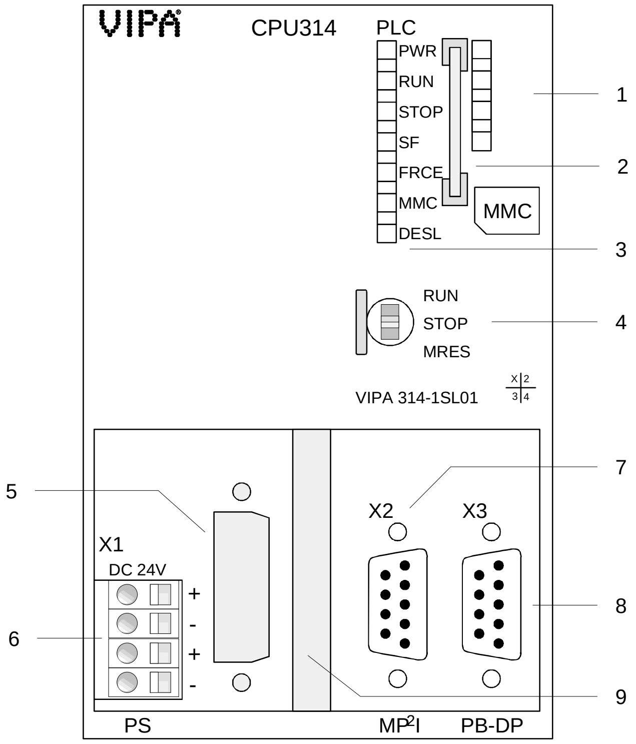


Ordering data CPU 31xNET with DP master

Type	Order number	Description
CPU 314	VIPA 314-3DP01	SPS CPU 314 with DP master, Ethernet TCP/IP-Interface, 4port Switch, 96kB Work / 144kB Load memory
CPU 315	VIPA 315-3DP01	SPS CPU 315 with DP master, Ethernet TCP/IP-Interface, 4port Switch, 192kB Work / 256kB Load memory
CPU 316	VIPA 316-3DP01	SPS CPU 316 with DP master, Ethernet TCP/IP-Interface, 4port Switch, 256kB Work / 512kB Load memory
CPU 317	VIPA 317-3DP01	SPS CPU 317 with DP master, Ethernet TCP/IP-Interface, 4port Switch, 512kB Work / 1MB Load memory

Structure

CPU 31x
31x-1SL01

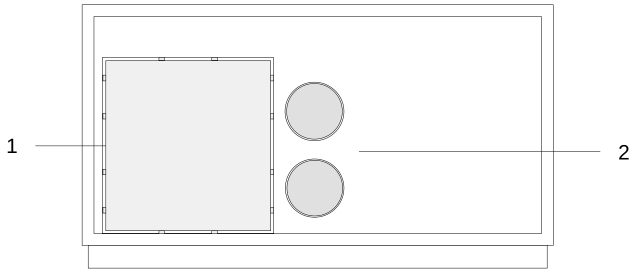


[1] deactivated LEDs
[2] MMC slot
[3] LEDs of the CPU part
[4] Operating mode switch CPU

The following components are under the front flap

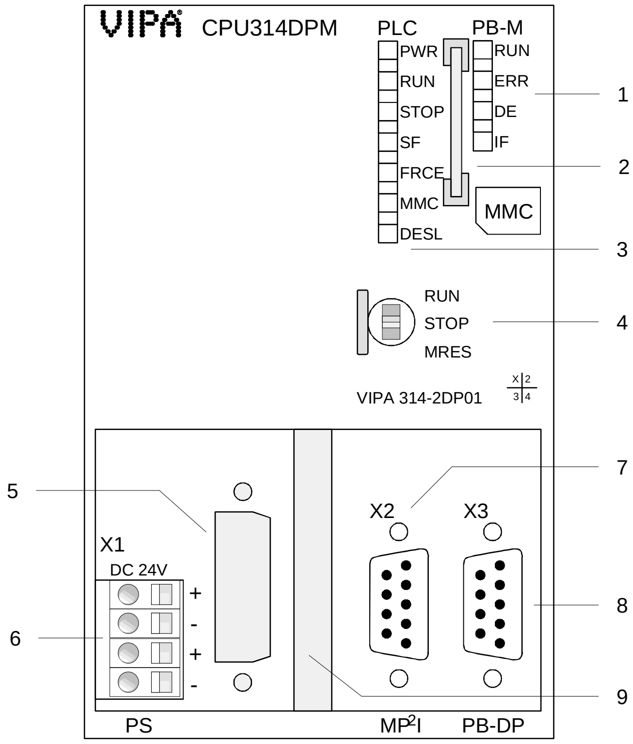
[5] Clipping for DVI slot
[6] Slot for DC 24V Power supply
[7] MP^{2}I interface
[8] RS485 Profibus interface
[9] Clipping for CompactFlash slot

Underside view For later expansions there are places for an Ethernet slot res. switch slot, keyboard and mouse (as PS2-Connection) on the underside of the front flap:



[1] Clipping for Ethernet slot res. Switch
[2] Clipping for keyboard res. mouse PS2 slots

CPU 31xDPM
31x-2DP01



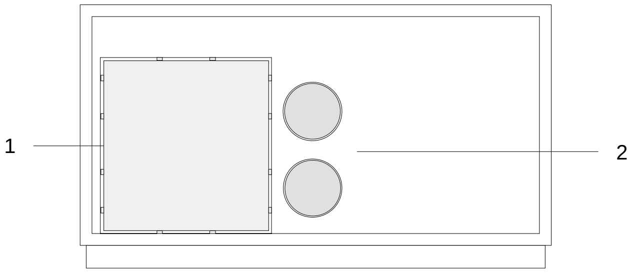
[1] LEDs of the integrated Profibus DI
[2] MMC slot
[3] LEDs of the CPU part
[4] Operating mode switch CPU

The following components are under the front flap

[5] Clipping for DVI slot
[6] Slot for DC 24V Power supply
[7] MP^{2}I interface
[8] RS485 Profibus interface
[9] Clipping for CompactFlash slot

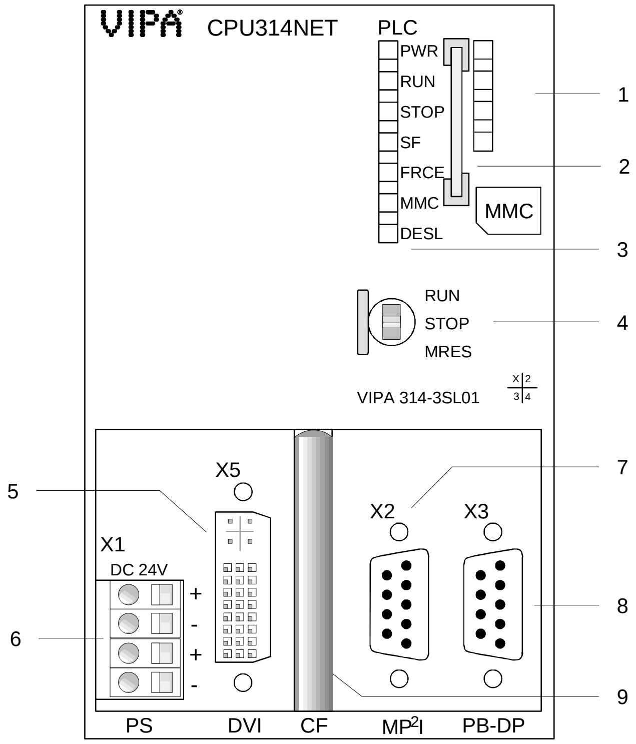
Underside view

For later expansions there are places for an Ethernet slot res. Switch slot, keyboard and mouse (as PS2-Connection) on the underside of the front flap:



[1] Clipping for Ethernet slot res. Switch
[2] Clipping for keyboard res. mouse PS2 slots

CPU 31xNET
31x-3SL01

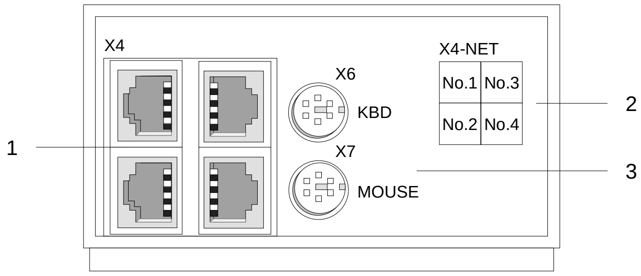


[1] deactivated LEDs
[2] MMC slot
[3] LEDs of the CPU part
[4] Operating mode switch CPU

The following components are under the front flap

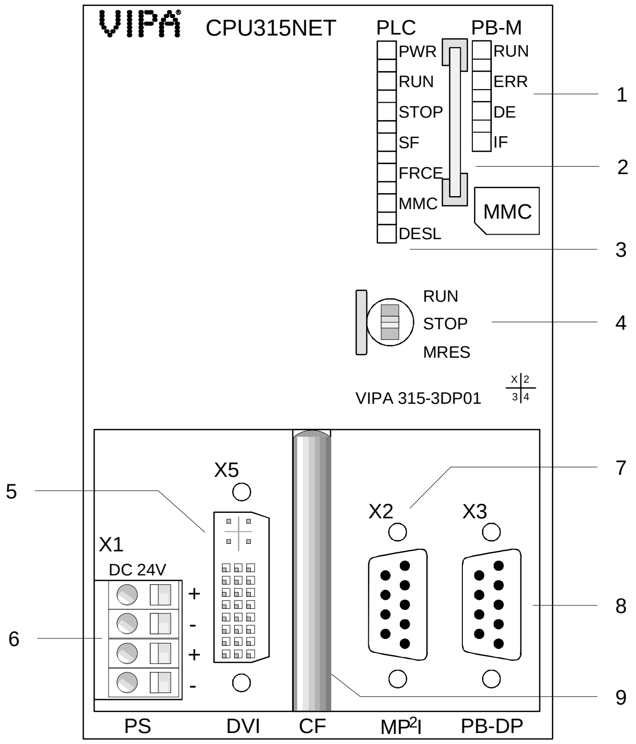
[5] DVI slot
[6] Slot for DC 24V Power supply
[7] MP^{2}I interface
[8] RS485 Profibus interface
[9] CompactFlash slot

Underside view The connections for Ethernet, keyboard and mouse are under a front flap that you may raise.



[1] 4port Switch for twisted pair connection to the Ethernet
[2] Assignment of the 4port Switch
[3] Jacks for keyboard res. mouse

CPU 31xNET
31x-3DP01

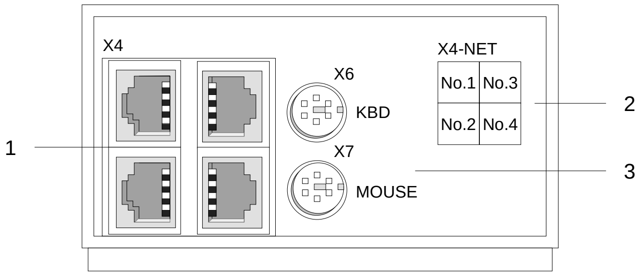


[1] LEDs of the integrated Profibus DP master
[2] MMC slot
[3] LEDs of the CPU part
[4] Operating mode switch CPU

The following components are under the front flap

[5] DVI slot
[6] Slot for DC 24V Power supply
[7] MP^{2}I interface
[8] RS485 Profibus interface
[9] CompactFlash slot

Underside view The connections for Ethernet, keyboard and mouse are beneath a front flap that you may raise.



[1] 4port Switch for twisted pair connection to the Ethernet
[2] Assignment of the 4port Switch
[3] Jacks for keyboard res. mouse

Components

LEDs

To display the status the CPU has two rows of LEDs on the frontside, one for the CPU and one for the Profibus-DP master.

The following tables show you the usage of the LEDs and the according colors:

CPU-Part

Label	Color	Meaning
PWR	yellow	CPU-Part is provided with internal 5V
RUN	green	CPU is in the operating mode RUN
STOP	yellow	CPU is in the operating mode STOP
SF	red	blinks at system errors(Hardware-defect)
FRCE	yellow	blinks as soon as variables are forced (fixed)
MMC	yellow	blinks at MMC access
DESL	yellow	shows Profibus DP slave activity, if the integrated Profibus master is deactivated

Note!

All LEDs of the CPU part are blinking three times, when accessing an invalid MMC or when the MMC is pulled out during the reading process.

Profibus DP master part

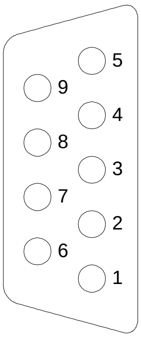
Name	Color	Meaning
RUN	green	Only RUN blinks: the DP master is in RUN. The DP-Slaves are called and the outputs are 0 ("clear"-Status). RUN+DE are blinking: the DP master is in "operate"-Status. It Exchanges data with the DP-Slaves.
ERR	red	blinks at a slave failure
DE	yellow	DE (Data exchange) shows communication via Profibus-DP.
IF	red	Init error at wrong parametrization.

Jacks and Plugs

The recent CPU includes the following jacks res. plugs:

Profibus-interface PB DP

Via the 9pin RS485-interface you link up the integrated Profibus-DP master to your Profibus network. The RS485 jack has the following pin assignment:



9pin jack

Pin	Occupation
1	Shield
2	n.c.
3	RxD/TxD-P (Line B)
4	RTS
5	M5V
6	P5V
7	n.c.
8	RxD/TxD-N (Line A)
9	n.c.

Note!

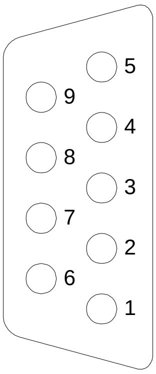
Please make sure to activate the terminating resistors at the bus ends!

MP^{2}Interface

The MP^{2}I interface handles the data exchange between CPU and PC. Via a bus communication you may transfer applications and data between the CPUs that are connected via MPI.

For a serial transfer from your PC you normally need a MPI transducer. Alternatively you may use the "Green Cable" (Order No.: 950-0KB00). from VIPA, that allows a serial connection without transducer. The "Green Cable" may only be used directly and exclusively at CPUs with MP^{2}I interface. Please also regard the hints in the chapter "Deployment CPU 31x"!

The MP^{2}I jack has the following pin assignment:



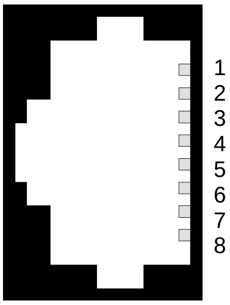
9pin jack

Pin	Assignment
1	reserved (do not connect)
2	M24V
3	RxD/TxD-P (Line B)
4	RTS
5	M5V
6	P5V
7	P24V
8	RxD/TxD-N (Line A)
9	n.c.

Ethernet-Connection

The twisted-pair jacks are used to build-up a twisted-pair network in star topology. This allows you to connect up to 4 Ethernet components, where 1 connection has to be deployed as uplink port to the ongoing network. The uplink port is detected automatically.

This jack has the following pin assignment:

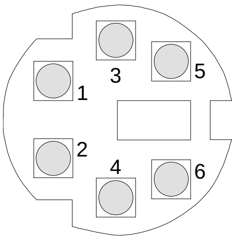


8pin RJ45 jack:

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

PS2 jack KBD/MOUSE

The pin assignment of the two PS2 jacks is identical. Connect your keyboard to the jack marked with "KBD" and your mouse to the jack marked with "MOUSE".

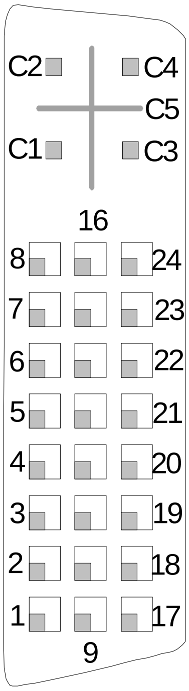


Pin	Assignment
1	+ KBD-Data (I/O)
2	reserved
3	GND
4	+5V
5	+ KBD-Clock (I/O)
6	reserved

DVI jack

The DVI jack is suitable for analog and digital displays and monitors with a max. resolution of 1280x1024pixel.

The jack has the following pin assignment:



Pin	Signal
C1	Analog Red
C2	Analog Green
C3	Analog Blue
C4	Analog Horizontal Sync
C5	Analog RGB Return
1	T.M.D.S Data2-
2	T.M.D.S Data2+
3	T.M.D.S Data2/4 Shield
4	T.M.D.S Data4-
5	T.M.D.S Data4-
6	DDC Clock
7	DDC Data
8	Analog Vertical Sync
9	T.M.D.S Data1-
10	T.M.D.S Data1+
11	T.M.D.S Data1/3 Shield
12	T.M.D.S Data3-
13	T.M.D.S Data3+
14	+5V Power
15	Ground (return for +5V, HSync and VSync)
16	Hot Plug Detect
17	T.M.D.S Data0-
18	T.M.D.S Data0+
19	T.M.D.S Data0/5 Shield
20	T.M.D.S Data5-
21	T.M.D.S Data5+
22	T.M.D.S Clock Shield
23	T.M.D.S Clock+
24	T.M.D.S Clock-

Memory of the CPU

The following storage technologies are deployed at the CPU 31x:

Memory	Type	Description
Load memory	battery buffered RAM	Here is the battery buffered program code and data blocks also in off-state.
Work memory	RAM	During program execution, the program code and data blocks are stored in the work memory.
Internal Flash	Flash-ROM	With a copy command you may store the program code and data blocks in the internal Flash-ROM. A read access happens only at empty buffer battery.
MMC	Flash-ROM	External memory card for project engineering of CPU and DP master and for firmware update of CPU, DP master and CP. With a copy command you may store the program code and data blocks from the CPU to MMC.
CompactFlash®	Flash-ROM	The CompactFlash® is only used with the CPU 31xNET and serves as external data memory.

Load memory

The load memory is a battery buffered RAM where the program code and data blocks are stored during project engineering.

Work memory

After power-on of the CPU, it transfers the program code and data blocks from the load memory in the unbuffered work memory.

Internal Flash

In addition to the load memory, every CPU 31x has an internal Flash-ROM. With the Copy-RAM-to-ROM command in the STEP®7 Manager from Siemens, the program code and data blocks are transferred to the internal Flash-ROM and stored permanently.

An access to the data in the internal Flash-ROM happens only at empty buffer battery after boot-up when no MMC is plugged-in.

MultiMediaCard "MMC"

The Multi Media Card, in short MMC, is an external memory module and can be ordered via the order no.: VIPA 953-0KX10.

CPU, CP and Profibus-DP master may use the CPU simultaneously for project engineering and firmware update.

The MMCs are preformatted with the file system FAT16. With every common MMC reading device, the MMC is linked-up to your PC as external drive. This allows to create programs at the PC, to copy them onto MMC and transfer them to the VIPA CPU by plugging-in the MMC.

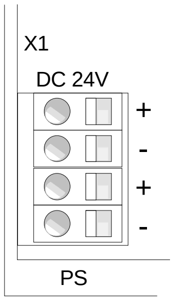
More detailed information is to find in the chapter "Deployment CPU 31x"

CompactFlash® Type II

As soon as a PC (CPU 31xNET) is integrated, it uses a CompactFlash® of the type I or II for the external data storage.

This slot for the CompactFlash® is marked with "CF".

Power supply



The CPU has an integrated power supply. The power supply has to be provided with DC 24V (20,4 ... 28,8V). For this serves the double DC 24V slot, that is underneath the flap.

Due to the double slot you may directly deduct DC 24V from the not used connection pair for front provided modules.

Via the power supply not only the internal electronic is provided with voltage, but by means of the backplane bus also the connected System 300V modules.

The internal power supply may be strained with maximum 3.5A.

The internal electronic is galvanically connected with the supply voltage.

The power supply is protected against polarity inversion and overvoltage.

Battery backup for clock and RAM

A rechargeable battery is installed on every CPU 31x to safeguard the contents of the RAM when power is removed. This battery is also used to buffer the internal clock.

The rechargeable battery is maintained by a charging circuit that receives its power from the internal power supply and that maintain the clock and RAM for a max. period of 30 days.

Operating mode switch RUN/STOP/MRES

With the operating mode switch you may switch the CPU between STOP and RUN. The operating mode START is driven automatically from the CPU between STOP and RUN.

Placing the switch to Memory Reset (MRES), you request an overall reset.

Technical Data

CPU 31x

General

Electrical Data	VIPA 314 ... VIPA 317
Power supply Current consumption Dissipation power Status display (LEDs) Integrated Profibus-interface	DC 24V max. 1.1A max. 2W via LEDs on the frontside DP slave
Slots / Interfaces: MP^{2}I Profibus	MP^{2}I-interface for data transmission: 4 static / 8 dynamic connections(187kBaud) RS232: 38,4kBaud RS485 Profibus-DP slave
Battery buffer / clock Output voltage to backplane bus Bit memory Timers / Counters Modules	yes / Lithium-Battery, 30 Days Buffer max. 3.5A 8192 256 / 256 FBs 1024, FCs 1024 DBs 2047
Combination with peripheral modules	
Max. number of modules Max. digital E/A (I/O) Max. analog E/A (I/O) Addressable E/A (I/O) digital channels analog channels	32 32 16 1024 1024
Measurements and Weight	
Measurements (HxWxL) in mm Weight in g	80x125x120 200g

Module Specific

	CPU 314	CPU 315	CPU 316	CPU 317
Program memory	96kByte	192kByte	256kByte	512kByte
Load memory	144kByte	256kByte	512kByte	1MB
Typical processing time Bit/Word (ms/k commands)	0.18 / 0.78	0.18 / 0.78	0.18 / 0.78	0.18 / 0.78
Order No.:	VIPA 314-1SL01	VIPA 315-1SL01	VIPA 316-1SL01	VIPA 317-1SL01

CPU 31xDPM

Electrical Data	VIPA 314 ... 317
Like CPU 31x additionally with DP master Status display (LEDs)	 like CPU 31x additionally with LEDs for the Profibus-DP master
Profibus interface	
Connection Network topology Medium Transfer rate Total length Max. number of partners	9-pin SubD-slot linear bus, active bus termination at both ends, tap line possible isolated drilled twisted pair-cable, isolation may be left depending on environmental conditions. 9.6kBaud to 12MBaud without repeater 100m at 12Mbaud, with repeater up to 1000m 32 stations in every segment without repeater; with Repeater expandable to 126.
Measurements and Weight	
Measurements (WxHxL) in mm Weight	80x125x120 200g

Module Specific

	CPU 314DPM	CPU 315DPM	CPU 316DPM	CPU 317DPM
Program memory	96kByte	192kByte	256kByte	512kByte
Load memory	144kByte	256kByte	512kByte	1MB
Typical processing time Bit/Word (ms/k commands)	0.18 / 0.78	0.18 / 0.78	0.18 / 0.78	0.18 / 0.78
Order No.: with DP slave	VIPA 314-2DP01	VIPA 315-2DP01	VIPA 316-2DP01	VIPA 317-2DP01

CPU 31xNET

Electrical Data	VIPA 314 ... 317
Like CPU 31x additionally: Ethernet TCP/IP Interface Memory	 4port Switch DVI-slot when expanded to Ethernet CompactFlash®
Ethernet Interface	
Network topology Medium Transfer Rate Total length	4port Switch, RJ45 for Twisted-Pair-Ethernet Twisted Pair 10/100MBit max. 100m per Segment
Measurements and Weight	
Measurements (WxHxL) in mm Weight	80x125x120 200g

Module Specific

	CPU 314NET	CPU 315NET	CPU 316NET	CPU 317NET
Program memory	96kByte	192kByte	256kByte	512kByte
Load memory	144kByte	256kByte	512kByte	1MB
Typical processing time Bit/Word (ms/k commands)	0.18 / 0.78	0.18 / 0.78	0.18 / 0.78	0.18 / 0.78
Order number with DP-Slave	VIPA 314-3SL01	VIPA 315-3SL01	VIPA 316-3SL01	VIPA 317-3SL01
Order No. with DP-Slave/Master	VIPA 314-3DP01	VIPA 315-3DP01	VIPA 316-3DP01	VIPA 317-3DP01

Chapter 4 Deployment CPU 31x

Outline

This chapter describes the deployment of a CPU 31x together with the peripheral modules of the System 300V. The description applies to those modules that have been inserted together with the CPU on one profile rail and are connected via the backplane bus.

Besides commissioning and start-up behavior you will find here also a description of the project engineering, parameterization, operating modes and test functions.

This information also basically applies to the deployment of the CPU 31x with integrated communication part.

Content

Note!

This information is valid for all the CPUs described in this manual, since the back panel communication between the CPU and the peripheral modules is the same for all models of CPU!

Fast introduction

Compatibility

The System 300V CPU is instruction compatible to the programming language STEP®7 from Siemens and may be programmed via the SIMATIC manager from Siemens.

A large function library is delivered in this package.

Note!

For programming of a System 300V CPU from VIPA please use always the **CPU 315-2DP (6ES7 315-2AF03)** from Siemens in the hardware catalog.

For the project engineering, a thorough knowledge of the hardware configurator from Siemens is required!

Connection to Profibus-DP

When using the System 300V as intelligent slave, the configuration works via the hardware configurator by Siemens by means of "configured stations" as CPU 31x.

The configuration of the optional integrated Profibus master takes place via the hardware configurator from Siemens. The project may be saved on the MMC.

Deployment of the CP part of the CPU 31xNET

The CP integrated in the CPU 31xNET is projected by WinNCS. The project engineering may be executed online via Ethernet or offline by means of a MMC.

The CP supports PU/OP functionality for up to 64 participants and is accessible by the "PLC" functions via Ethernet.

WinNCS allows you to execute a firmware update online and password protected or to reboot the CP part independently from the CPU.

The access rights to the CP are restricted by means of IP list that are transferred at project engineering.

With an invalid CP project engineering, the project at the CP is deleted and the CP reboots with the default IP address.

Firmware and project engineering are stored at a DOC (DiscOnChip).

MultiMediaCard as external storage medium

CPU, CP and Profibus master use the MultiMediaCard (MMC) as external storage medium. The MMC is available from VIPA.

With an Overall reset, it is automatically read from MMC.

Green Cable

For project engineering of your DP slave you may transfer your projects from your PC to the CPU serial via MPI by using the "Green Cable". Please also regard the hints to the Green Cable in this chapter.

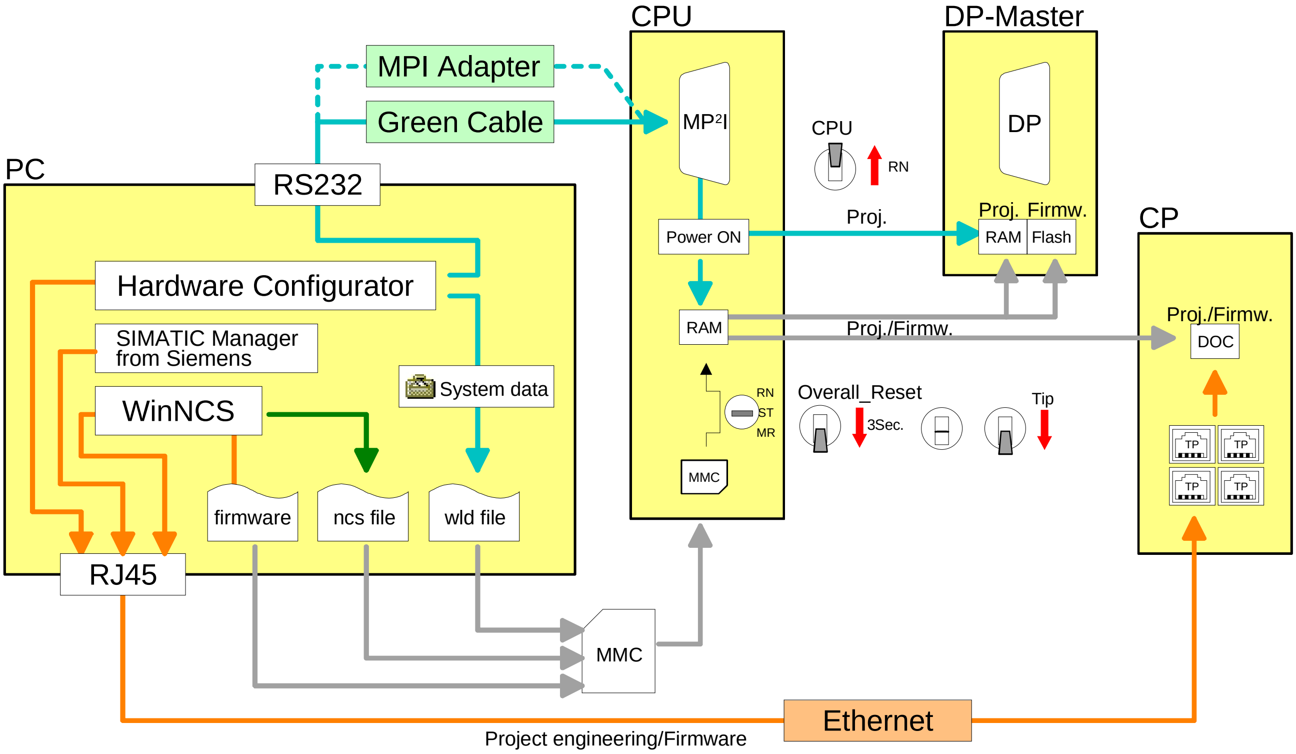
Integrated power supply

Every CPU res. bus coupler comes with an integrated power supply. The power supply has to be supplied with DC 24V. By means of the supply voltage, the bus coupler electronic is supplied as well as the connected modules via backplane bus. Please regard that the integrated power supply may supply the backplane bus with max. 3.5A.

The power supply is protected against inverse polarity and overcurrent.

Access options for project engineering and firmware update

The following overview shows all access options for project engineering and firmware update.



Hints for the deployment of the MPI interface

What is MP^{2}I?

The MP^{2}I jack combines 2 interfaces in 1:

- MPI interface
- RS232 interface

Please regard that the RS232 functionality is only available by using the Green Cable from VIPA.

Deployment as MPI interface

The MP interface provides the data transfer between CPUs and PCs. In a bus communication you may transfer programs and data between the CPUs interconnected via MPI.

Connecting a common MPI cable, the MPI jack supports the full MPI functionality.

⚠

Important notes for the deployment of MPI cables!

Deploying MPI cables at the CPUs from VIPA, you have to make sure that Pin 1 is not connected. This may cause transfer problems and in some cases damage the CPU!

Especially Profibus cables from Siemens, like e.g. the 6XV1 830-1CH30, must not be deployed at MP^{2}I jack.

For damages caused by nonobservance of these notes and at improper deployment, VIPA does not take liability!

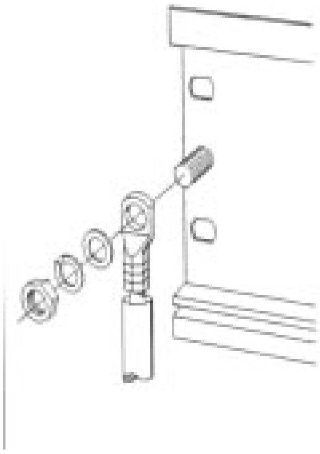
Deployment as RS232 interface only via "Green Cable"

For the serial data transfer from your PC, you normally need a MPI transducer. Fortunately you may also use the "Green Cable" from VIPA. You can order this under the order no. VIPA 950-0KB00.

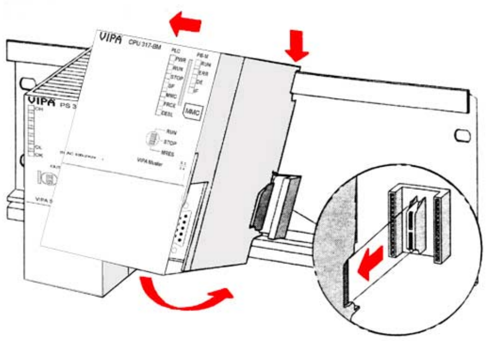
The "Green Cable" supports a serial point-to-point connection for data transfer via the MP^{2}I jack exclusively for VIPA CPUs.

Installation

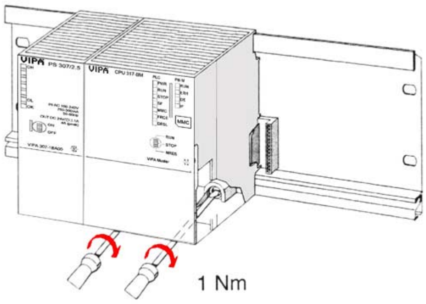
Approach



- Bolt the profile rail with the background (screw size: M6), so that you still have minimum 65mm space above and 40mm below the profile rail.
- If the background is a grounded metal or device plate, please look for a low-impedance connection between profile rail and background.
- Connect the profile rail with the protected earth conductor. For this purpose there is a bolt with M6-thread.
- The minimum cross-section of the cable to the protected earth conductor has to be 10mm^2.



- Stick the power supply to the profile rail and pull it to the left side to the grounding bolt of the profile rail.
- Fix the power supply by screwing.
- Take a bus coupler and click it at the CPU from behind like shown in the picture.
- Stick the CPU to the profile rail right from the power supply and pull it to the power supply.



- Click the CPU downwards and bolt it like shown.
- Repeat this procedure with the peripheral modules, by clicking a backplane bus coupler, stick the module right from the modules you've already fixed, click it downwards and connect it with the backplane bus coupler of the last module and bolt it.

More details see chapter "Cabling and installation specifications"

Danger!

- Before installing or overhauling the System 300V, the power supplies must be disconnected from voltage (pull the plug or remove the fuse)!
- Installation and modifications only by properly trained personnel!

Start-up behavior

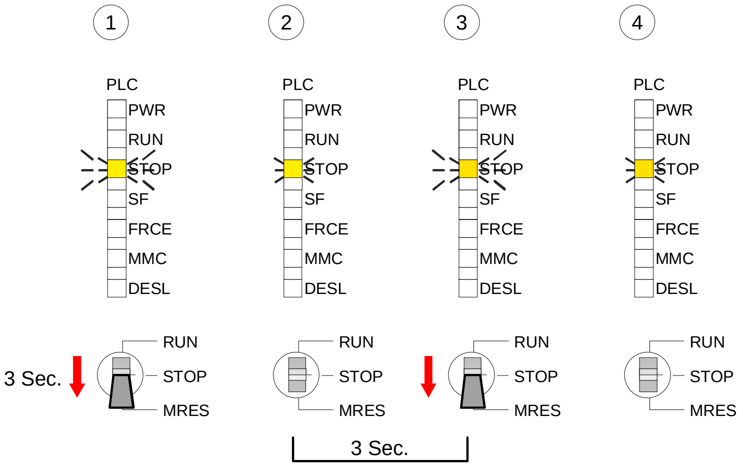
Turn on power supply

After the power supply has been switched on, the CPU changes to the operating mode the operating mode lever shows.

Now you may transfer your project to the CPU via MPI from your configuration tool res. plug in a MMC with your project and run an Overall reset.

Overall reset

The following picture shows the approach once more:



Note!

The transfer of the application program from the MMC into the CPU takes always place after an Overall reset!

Default boot procedure, as delivered

When the CPU is delivered it has been reset. After a STOP→RUN transition the CPU switches to RUN without program.

Boot procedure with valid data in the CPU

The CPU switches to RUN with the program stored in the battery buffered RAM.

Boot procedure with empty battery

The accumulator/battery is automatically loaded via the integrated power supply and guarantees a buffer for max. 30 days. If this time is exceeded, the accu may be totally discharged. This means that the battery buffered RAM is deleted.

In this state the CPU executes an Overall reset. If a MMC is plugged in, the application on the MMC is transferred into the RAM. Otherwise the CPU accesses the application in the internal Flash and transfers this one to the RAM.

Depending on the chosen operating mode, the CPU switches to RUN res. remains in STOP.

This event is stored in the diagnostic buffer as: "Start Overall reset automatically (unbuffered POWER_ON)".

Addressing

Automatic addressing

To provide specific addressing of the installed peripheral modules, certain addresses must be allocated in the CPU.

The CPU contains a peripheral area (addresses 0 ... 1023) and a process image of the inputs and the outputs (for every address 0 ... 127).

When the CPU is initialized, it automatically assigns peripheral addresses to the digital input/output modules starting from 0.

When no hardware project engineering is available, analog modules are stored at the automatic address allocation on even addresses starting with 128.

Up to 32 modules in one row

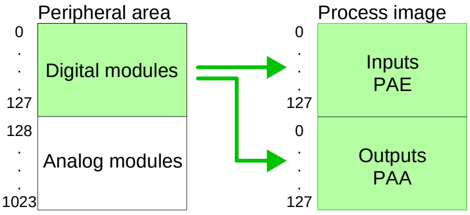
In the hardware configurator from Siemens you may parameterize maximum up to 8 modules per row. When using the System 300V CPUs from VIPA you may control up to 32 modules by distributing the modules you want to parameterize on the first 8 Plug-ins. Although the modules that are behind on the profile rail are not visible, they are included into the addressing range of the CPU by means of the automatic addressing.

Signal states in the process image

The signal states of the lower addresses (0 ... 127) are additionally saved in a special memory area called the *process image*.

The process image this divided into two parts:

- process image to the inputs (PAE)
- process image to the outputs (PAA)



The process image is updated automatically when a cycle has been completed.

Read/write access

You may access the modules by means of read or write operations on the peripheral bytes or on the process image.

Note!

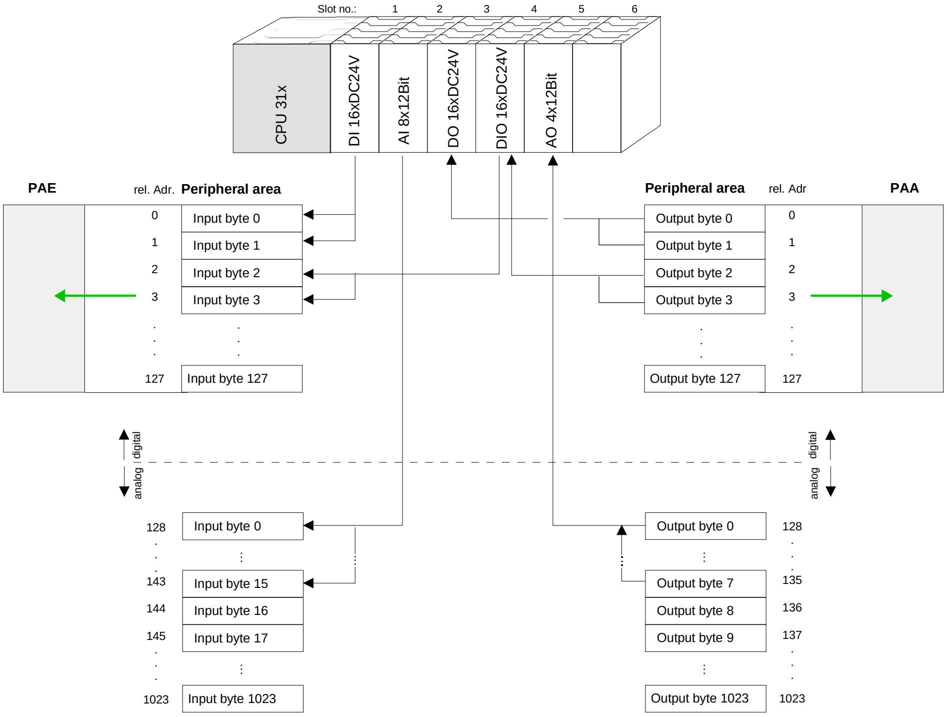
Please remember that you can access different modules by means of read and write operations on the same address.

Digital and analog modules have separate address ranges at the automatic address allocation.

Digital modules: 0 ... 127, Analog modules: 128 ... 1023

Example for the automatic address allocation

The following figure illustrates the automatic address allocation:



Change address allocation by project engineering

You may change the address allocation with the help of the SIMATIC manager from Siemens at any time. This allows you to get also analog modules into the process image area (0 ... 127) and digital modules above address 127.

The preparation for the project engineering and the approach are described on the following pages.

Project engineering

Overview

The description below refers to modules that are at the same bus together with the CPU 31x from VIPA.

In order to address the installed peripheral modules individually, specific addresses in the CPU have to be assigned to them.

The allocation of addresses and the configuration of the installed modules is a function of the Siemens SIMATIC manager.

The project engineering of the CPU 31x, the Profibus components and System 300V modules from VIPA happens in the Siemens SIMATIC manager.

The parameterization of the Ethernet components takes place via the VIPA parameterization tool WinNCS.

With the help of the VIPA Green Cable you may transfer your project from the PC into the CPU via the MPI interface.

You may also copy your project with a common MMC reading device to a MMC and transfer it into the CPU by means of an Overall reset.

To be compatible to the Siemens SIMATIC manager, you have to configure the System 300V CPUs from VIPA as

CPU 315-2DP (6ES7 315-2AF03 V1.2)!

Profibus address 1 is reserved

Please regard that the Profibus address 1 is reserved for the system. You should not use the address 1!

Fast introduction

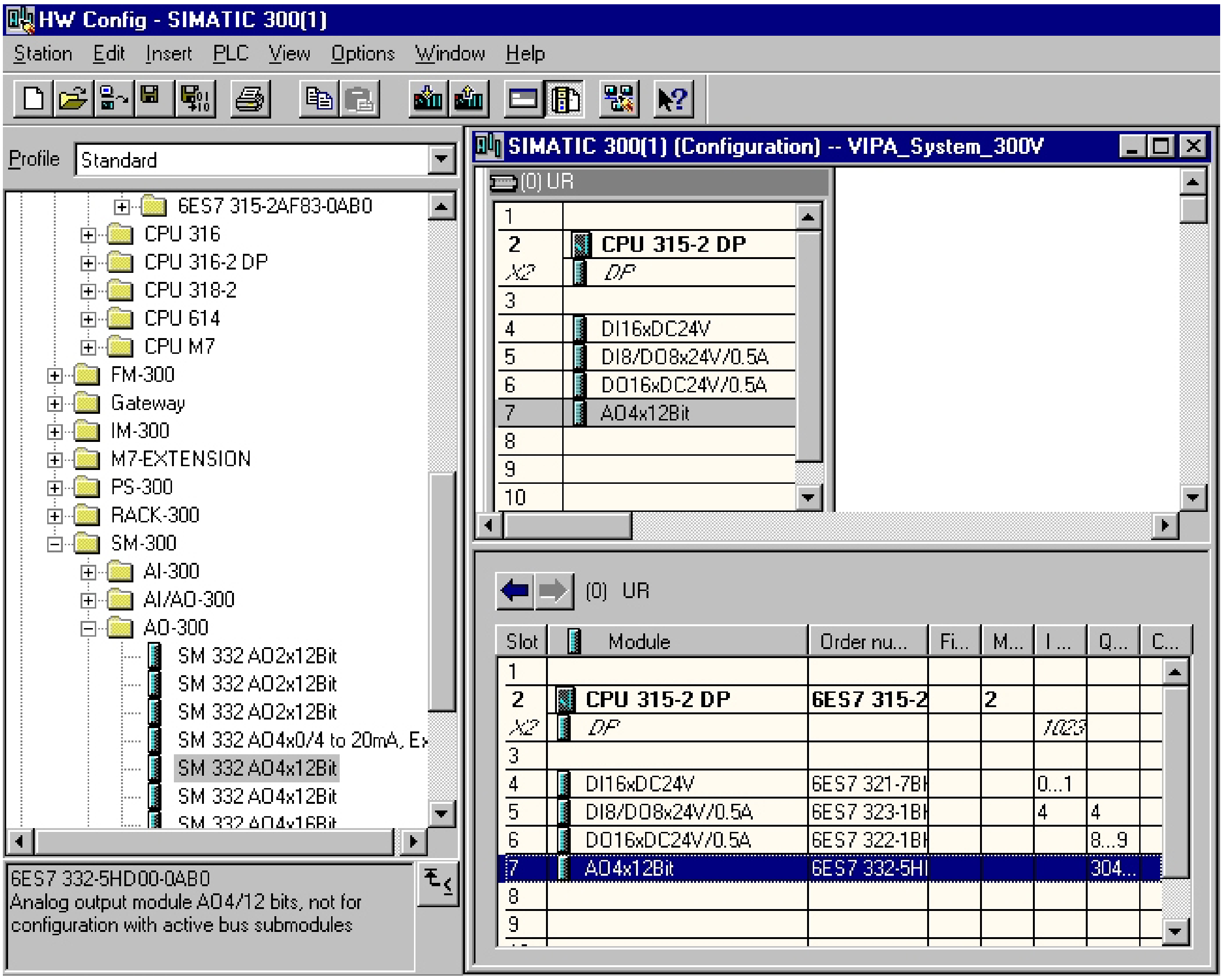
- Start Siemens SIMATIC manager with new project and configure Siemens CPU 315-2DP (6ES7 315-2AF03 V1.2). The CPUs 31x from VIPA have to be included into the hardware configurator from Siemens as **CPU 315-2DP (6ES7 315-2AF03 V1.2)**.
- The modules are linked-up in the plugged-in sequence starting with slot 4.
- Via double click on the CPU or the modules you may parameterize them.
- When deploying a CPU 31xDP with a master system the CPU can be connected as slave system CPU 31x by means of "configured stations". A condition is that the slave system was projected before.
- Save and transfer your project.

In the following these points are more near described.

Approach of the project engineering

This is the approach to configure the modules that are on the bus beside the CPU:

- Create a new project.
- Add the CPU 315-2DP (6ES7 315-2AF03-0AB0 V1.2) from the hardware catalog.
  You will find this at *Simatic 300 > CPU 300 > CPU 315-2 DP*.
- Link-up your System 300V modules in the plugged-in sequence starting with slot 4.
- If needed, parameterize the CPU res. the modules. The parameter window appears as soon as you double click on the according module.
- Save your project.



6ES7 332-5HD00-0AB0
Analog output module AO4/12 bits, not for configuration with active bus submodules

1	
2	CPU 315-2 DP
X2	DP
3	
4	DI16xDC24V
5	DI8/DO8x24V/0.5A
6	DO16xDC24V/0.5A
7	AO4x12Bit
8	
9	
10	

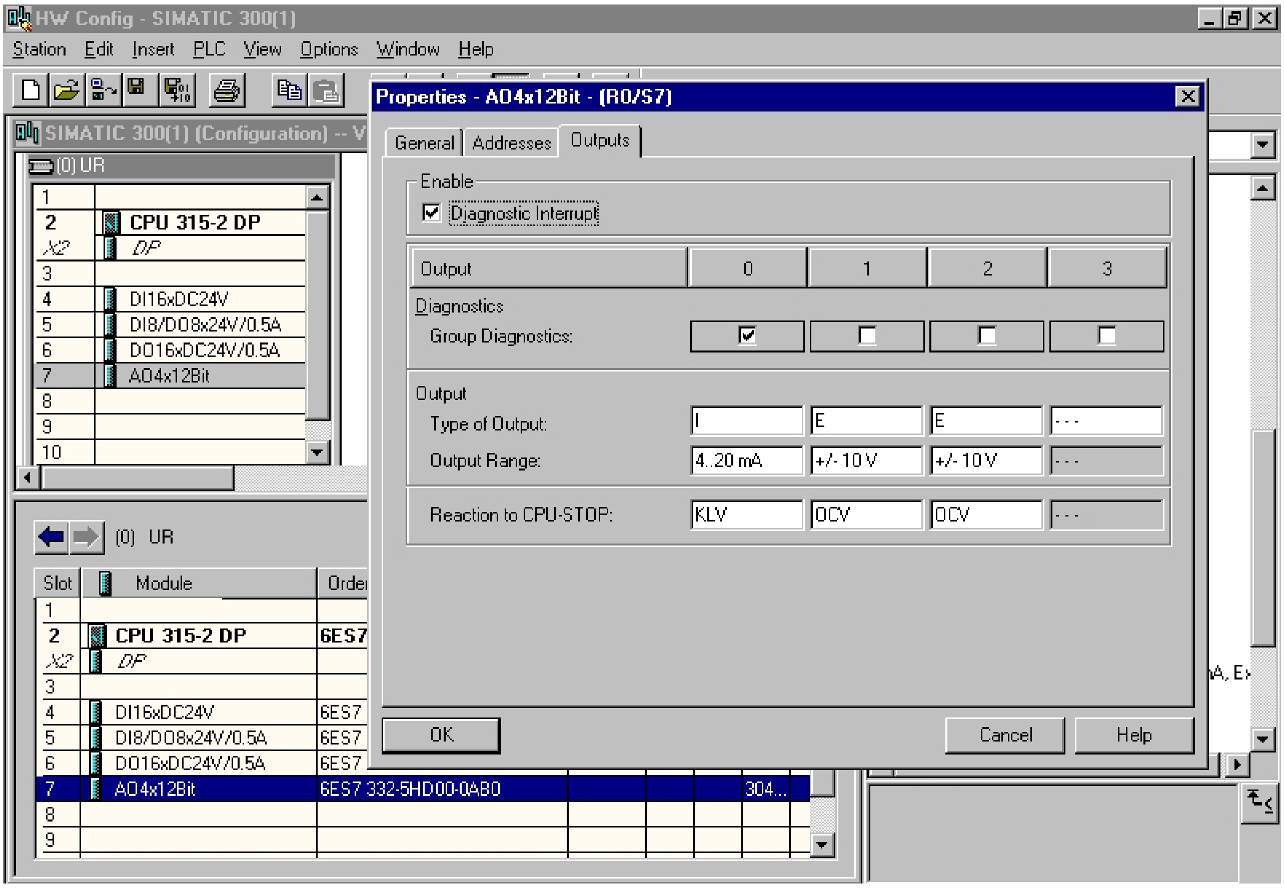
(0) UR

Slot	Module	Order nu...	Fi...	M...	I ...	Q...	C...
1							
2	CPU 315-2 DP	6ES7 315-2		2			
X2	DP				1023		
3							
4	DI16xDC24V	6ES7 321-7BH			0...1		
5	DI8/DO8x24V/0.5A	6ES7 323-1BH			4	4	
6	DO16xDC24V/0.5A	6ES7 322-1BH				8...9	
7	AO4x12Bit	6ES7 332-5HI				304...	
8							
9							

Parameterization of modules

System 300V modules are able to receive parameter data from the CPU. Via deployment of the SIMATIC manager from Siemens you may define parameters for parameterizable System 300V modules at any time.

Therefore you double-click at the placed module you want to parameterize. In the now opening dialog window the parameters can be changed.


HW Config - SIMATIC 300(1)
Station Edit Insert PLC View Options Window Help
SIMATIC 300(1) (Configuration) -- V
(0) UR
(0) UR
OK
Cancel
Help

Properties - AO4x12Bit - (R0/S7)

General | Addresses | Outputs

Enable

☑ Diagnostic Interrupt

Output	0	1	2	3
Diagnostics				
Group Diagnostics:	☑	□	□	□
Output				
Type of Output:	I	E	E	- - -
Output Range:	4..20 mA	+/- 10 V	+/- 10 V	- - -
Reaction to CPU-STOP:	KLV	OCV	OCV	- - -

Slot	Module	Order
1		
2	CPU 315-2 DP	6ES7
X2	DP	
3		
4	DI16xDC24V	6ES7
5	DI8/DO8x24V/0.5A	6ES7
6	DO16xDC24V/0.5A	6ES7
7	AO4x12Bit	6ES7 332-5HD00-0AB0
8		
9		

Transfer project

The data transmission e.g. takes place via MPI. If your programming system don't has a MPInterface, you may use the Green cable from VIPA to establish a serial point-to-point connection.

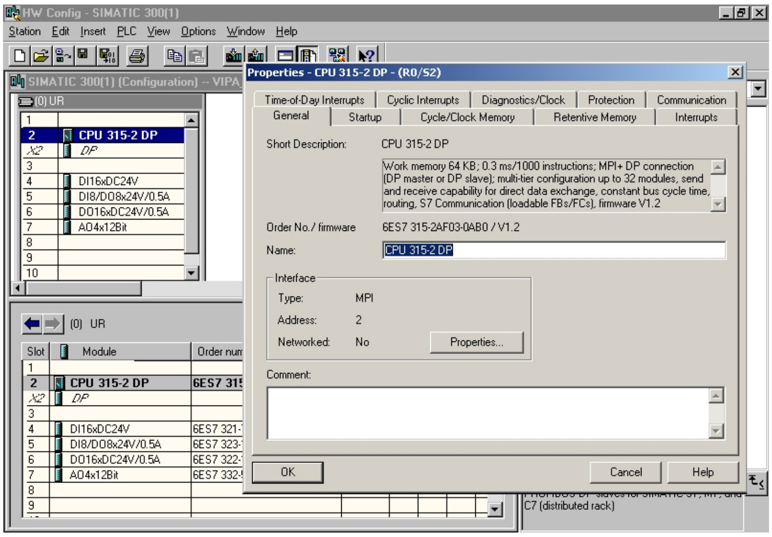
The "Green Cable" has the order number VIPA 950-0KB00 and may only be used with VIPA CPUs with MP^{2}I interface.

- Connect your PG with the CPU
- With **PLC** > *Upload Station* in your configuration tool you transfer your project into the CPU.
- Plug in a MMC and transfer your application program to the MMC with **PLC** > *Copy RAM to ROM*.
- During the write process, the "MMC"-LED at the CPU is blinking. Due to system reasons, the ready write process is announced a little bit too soon. The write process is only finished when the LED extinguishes.

Parameter adjustment CPU

Outline

The general parameters concerning the CPU section of your System 300V have to be configured in the hardware configurator from Siemens under the properties of the CPU 315-2 DP.



(0) UR

Slot	Module
1	
2	CPU 315-2 DP
X2	DP
3	
4	DI16xDC24V
5	DI8/DO8x24V/0.5A
6	DO16xDC24V/0.5A
7	AO4x12Bit
8	
9	
10	

(0) UR

Slot	Module	Order num
1		
2	CPU 315-2 DP	6ES7 315
X2	DP	
3		
4	DI16xDC24V	6ES7 321-
5	DI8/DO8x24V/0.5A	6ES7 323-
6	DO16xDC24V/0.5A	6ES7 322-
7	AO4x12Bit	6ES7 332-
8		
9		

Short Description: CPU 315-2 DP

Work memory 64 KB; 0.3 ms/1000 instructions; MPI+ DP connection (DP master or DP slave); multi-tier configuration up to 32 modules, send and receive capability for direct data exchange, constant bus cycle time, routing, S7 Communication (loadable FBs/FCs), firmware V1.2

Order No./ firmware: 6ES7 315-2AF03-0AB0 / V1.2

Name: CPU 315-2 DP

Interface
Type: MPI
Address: 2
Networked: No

Comment:

Approach

With a double click at the CPU 315-2 DP you reach the parameterization window for your CPU. Via the register tabs you may access all parameters of your System 300V CPU.

Please regard, that at this time not all parameters are supported.

Supported parameters

The following parameters are evaluated by the CPU at this time:

General:

MPI address of the CPU
Baud rate (19.2kBaud, 187kBaud)
maximum MPI address

Startup:

Startup when expected/actual configuration differ

"Finished" message by modules
Transfer of parameters to modules

Retentive memory:

Number of Memory bytes starting with MB0
Number of S7 Timers starting with T0
Number of S7 Counters starting with C0

Protection:

Level of protection ...password

Time-of-Day Interrupts:

OB10: active
execution
start date
time-of-day

Cyclic Interrupts:

OB35: execution

Cycle Clock Memory:

Scan Cycle monitoring time
Scan Cycle load from communication
OB85: Call up at I/O access error
Clock bit memory with memory byte no.

Project transfer

Overview

There are 3 possibilities for the transfer of your project into the CPU:

- Transfer via MP^{2}I
- Transfer via Profibus (not for startup)
- Transfer via MMC by using a card reader
- Transfer via Ethernet (only CPU 31xNET)

Transfer via MP^{2}I resp. Profibus

Depending upon CPU 31x you have the following transfer options:

- Transfer via MPI Programming cable (MPI/Profibus Communication)
- Transfer via Green Cable (Serial Communication via MP^{2}I - not Profibus)

Transfer with MPI Programming cable via MP^{2}I resp. Profibus

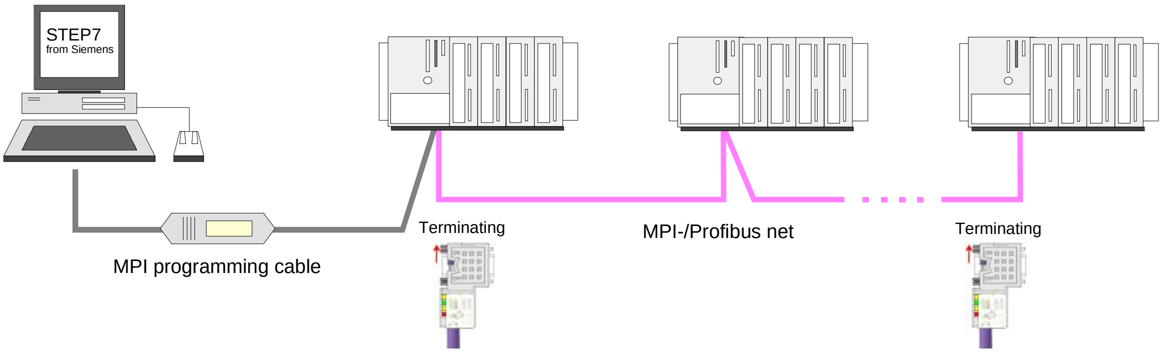
The MPI programming cables are available at VIPA in different variants. The employment of the cables is identical. The cables provide a bus enabled RS485 plug for the MP^{2}I jack of the CPU and a RS232 res. USB plug for the PC.

Due to the RS485 connection you may plug the MPI programming cables directly to a already plugged MPI plug on the MP^{2}I jack. Every bus participant identifies itself at the bus with an unique MPI address, in the course of which the address 0 is reserved for programming devices. The structure of a MPI net is in the principal identical with the structure of a 1.5MBaud Profibus net. I.e. the same rules are valid and you use the same components for the build-up. The single participants are connected with each other via bus interface plugs and Profibus cables. Your CPU 31x supports transfer rates of up to 1.5MBaud. Per default the MPI net runs with 187.5kBaud. VIPA CPUs are delivered with MPI address 2.

Terminating resistor

A cable has to be terminated with its surge impedance. For this you switch on the terminating resistor at the first and the last participant of a network or a segment.

Please make sure that the participants with the activated terminating resistors are always provided with voltage during start-up and operation.



Approach transfer via MP^{2}I

A maximum of 32 PG/OP connections is supported by MPI. The transfer via MPI takes place with the following proceeding:

- Connect your PC to the MPI jack of your CPU via a MPI programming cable.
- Load your project in the SIMATIC Manager from Siemens.
- Choose in the menu **Options** > *Set PG/PC interface*
- Select in the according list the "PC Adapter (MPI)"; if appropriate you have to add it first, then click on [Properties].
- Set in the register *MPI* the transfer parameters of your MPI net and type a valid *address*.
- Switch to the register *Local connection*
- Set the COM port of the PCs and the transfer rate 38400Baud for the MPI programming cable from VIPA.
- Via **PLC** > *Load to module* you may transfer your project via MPI to the CPU and save it on a MMC via **PLC** > *Copy RAM to ROM* if one is plugged.

Approach transfer via Profibus

Transfer via Profibus is only available by DP master, if projected as master and assigned with a Profibus address before. A maximum of 31 PG/OP connections is supported by Profibus. The transfer via MPI takes place with the following proceeding:

- Connect your PC to the MPI jack of your CPU via a MPI programming cable.
- Load your project in the Siemens SIMATIC Manager.
- Choose in the menu **Options** > *Set PG/PC interface*
- Select in the according list the "PC Adapter (Profibus)"; if appropriate you have to add it first, then click on [Properties].
- Set in the register *Profibus* the transfer parameters of your Profibus net and type a valid *Profibus address*. The *Profibus address* must be assigned to the DP master by a project before.
- Switch to the register *Local connection*
- Set the COM port of the PCs and the transfer rate 38400Baud for the MPI programming cable from VIPA.
- Via **PLC** > *Load to module* you may transfer your project via Profibus to the CPU and save it on a MMC via **PLC** > *Copy RAM to ROM* if one is plugged.

Transfer with Green Cable only at MP^{2}I

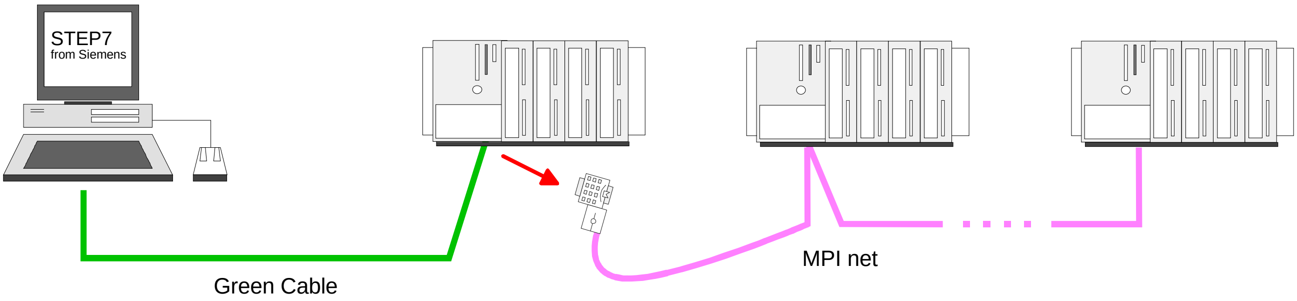
The "Green Cable" is a programming and download cable that may exclusively be plugged directly to VIPA components with MP^{2}I jack. By plugging the Green Cable to a MP^{2}I jack you may establish a serial connection between the COM interface of your PC and the MP^{2}I interface of your CPU.

Attention!

Please regard that you may plug the "Green Cable" exclusively directly to the MP^{2}I interfaces of VIPA-CPUs!

Approach

- Connect the PC and CPU with the Green Cable.
- Load your project in the SIMATIC Manager from Siemens.
- Choose in the menu **Options** > *Set PG/PC interface*
- Select in the according list the "PC Adapter (MPI)"; if appropriate you have to add it first, then click on [Properties].
- Switch to the register *Local connection*
- Set the COM port of the PCs and the transfer rate 38400Baud for the MPI programming cable from VIPA. The settings in the register *MPI* are ignored at the usage of the Green Cable.
- Via **PLC** > *Load to module* you may transfer your project to the CPU and save it on a MMC via **PLC** > *Copy RAM to ROM* if one is plugged.



Hints for the Green Cable

The Green Cable is a green connection cable, manufactured exclusively for the deployment at VIPA System components.

The Green Cable is a programming and download cable for VIPA CPUs MP^{2}I jack and VIPA fieldbus masters. The Green Cable from VIPA is available under the order no. VIPA 950-0KB00.

The Green Cable allows you to:

- *transfer projects serial*
  Avoiding high hardware needs (MPI transducer, etc.) you may realize a serial point-to-point connection via the Green Cable and the MP^{2}I jack. This allows you to connect components to your VIPA-CPU that are able to communicate serial via an MPI adapter like e.g. a visualization system.
- *execute firmware updates of the CPUs and fieldbus masters*
  Via the Green Cable and an upload application you may update the firmware of all recent VIPA CPUs with MP^{2}I jack and certain fieldbus masters (see Note).

Important notes for the deployment of the Green Cable

Nonobservance of the following notes may cause damages on system components.

For damages caused by nonobservance of the following notes and at improper deployment, VIPA does not take liability!

Note to the application area

The Green Cable may exclusively deployed directly at the concerning jacks of the VIPA components (in between plugs are not permitted). E.g. a MPI cable has to be disconnected if you want to connect a Green Cable.

At this time, the following components support Green Cable:

VIPA CPUs with MP^{2}I jack and fieldbus masters from VIPA.

Note to the lengthening

The lengthening of the Green Cable with another Green Cable res. The combination with further MPI cables is not permitted and causes damages of the connected components!

The Green Cable may only be lengthened with a 1:1 cable (all 9 Pins are connected 1:1).

Transfer via MMC

As external storage medium a MMC is deployed. The MMC (**Me**mory **C**ard) serves as external transfer medium for programs and firmware for, among others, it provides the PC compatible FAT16 file system. With an overall reset or PowerON the MMC is automatically read. There may be stored several projects and sub-directories on a MMC storage module. Please consider that the current project is stored in the root directory and has one of the in the following described file names.

Transfer MMC→RAM→ROM

Always after overall reset and PowerON the CPU tries to load a user program from the MMC into the battery-buffered RAM or in the Flash memory. Here the following file names may be assigned to the project depending upon the desired functionality:

- **S7PROG.WLD**
  After overall reset the user program S7PROG.WLD is transferred into the battery-buffered RAM.
- **S7PROGF.WLD (starting with Firmware-Version V. 3.8.6)**
  After overall reset the user program S7PROG.WLD is transferred into the battery-buffered RAM and additionally into the Flash memory. An access to the Flash memory takes only place at empty battery of the buffer and when no MMC with user program is plugged-in.
- **AUTOLOAD.WLD**
  After PowerON the user program AUTOLOAD.WLD is transferred into the battery-buffered RAM.

Transfer RAM→MMC→ROM

When the MMC has been plugged-in, the write command stores the content of the battery-buffered RAM as **S7PROG.WLD** at the MMC. The write command is controlled by means of the Siemens hardware configurator via **PLC** > *Copy RAM to ROM*. During the write process the "MMC"-LED of the CPU is blinking. When the LED expires the write process is finished. Simultaneously a write process into the internal Flash memory of the CPU takes place.

Transfer control

After a write process onto the MMC, an according ID event is written into the diagnostic buffer of the CPU. To monitor the diagnosis entries, you select **PLC** > *Module Information* in the Siemens SIMATIC Manager. Via the register "Diagnostic Buffer" you reach the diagnosis window.

The following events may occur:

Event-ID	Meaning
0xE100	MMC access error
0xE101	MMC error file system
0xE102	MMC error FAT
0xE200	MMC writing finished
0xE300	Internal Flash writing finished

More information to the event IDs may be found at the end of this chapter.

Note!

If the size of the user application exceeds the user memory of the CPU, the content of the MMC is not transferred to the CPU.
Execute a compression before the transfer, for this does not happen automatically.

Transfer via Ethernet

Besides of the transfer via MPI res. MMC you may configure the CPU via the CP portion when using the CPU 31xNET.

Here all block functions are supported like e.g. load, alter, monitor and save blocks.

All "PLC" functions except of "Hardware diagnosis" are available.

Up to 64 participants may access simultaneously.

Preconditions

The following software packages are installed:

- SIMATIC manager from Siemens V.5.1 or V.5.2 with SP1
- SIMATIC NET

Approach

- Start the SIMATIC manager from Siemens, load your project and select the wanted interface for the online driver under **Options** > *PG/PC Interface*.
- Choose an interface parameterization that refers to your network PCI card like e.g.:
  TCP/IP->3Com EtherLink 10/100 PCI
- Transfer the wanted blocks and system data via the PLC functions into your CPU.

More detailed information is to be found at chapter "Deployment CPU 31x with TCP/IP" at "PG/OP communication".

Operating modes

Outline

The CPU can be in one of 3 operating modes:

- Operating mode STOP
- Operating mode START-UP
- Operating mode RUN

Certain conditions in the operating modes START-UP and RUN require a specific reaction from the system program. In this case the application interface is often provided by a call to an organization block that was included specifically for this event.

Operating mode STOP

- The application program is not processed.
- If there has been a processing before, the values of counters, timers, flags and the process image are retained during the transition to the STOP mode.
- Outputs are inhibited, i.e. all digital outputs are disabled.
- RUN-LED off
- STOP-LED on

Operating mode START-UP

- During the transition from STOP to RUN a call is issued to the start-up organization block OB 100. The length of this OB is not limited. The processing time for this OB is not monitored. The start-up OB may issue calls to other blocks.
- All digital outputs are disabled during the start-up, i.e. outputs are inhibited.
- RUN-LED blinks
- STOP-LED off

When the CPU has completed the start-up OB, it assumes the operating mode RUN.

Operating mode RUN

- The application program in OB 1 is processed in a cycle. Under the control of alarms other program sections can be included in the cycle.
- All timers and counters being started by the program are active and the process image is updated with every cycle.
- The BASP-signal (outputs inhibited) is deactivated, i.e. all digital outputs are enabled.
- RUN-LED on
- STOP-LED off

Overall Reset

Outline

During the overall reset the entire user memory (RAM) is erased. Data located in the memory card is not affected.

You have 2 options to initiate an overall reset:

- initiate the overall reset by means of the function selector switch
- initiate the overall reset by means of the Siemens SIMATIC manager

Note!

You should always issue an overall reset to your CPU before loading an application program into your CPU to ensure that all blocks have been cleared from the CPU.

Overall reset by means of the function selector

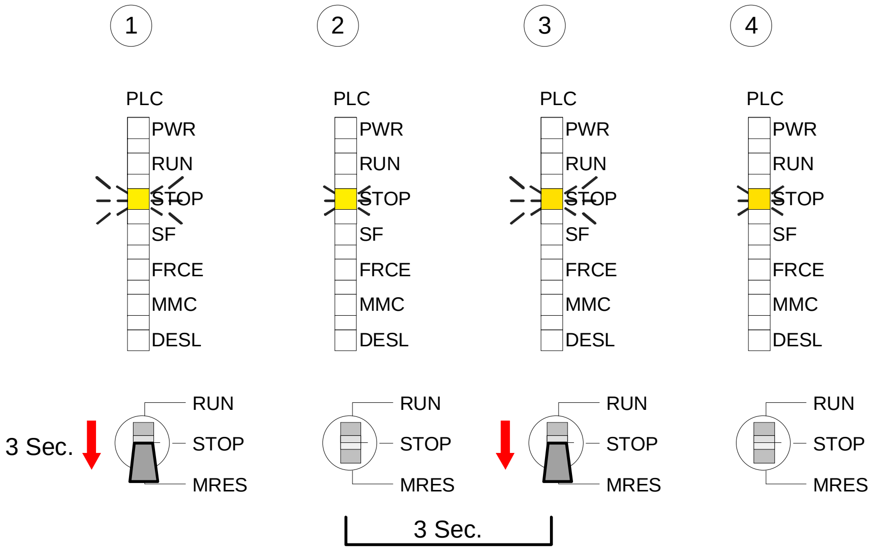
Condition

The operating mode of the CPU is STOP. Place the function selector on the CPU in position "STOP" → the STOP-LED is on.

Overall reset

- Place the function selector in the position MRES and hold it in this position for app. 3 seconds. → The STOP-LED changes from blinking to permanently on.
- Place the function selector in the position STOP and switch it to MRES and quickly back to STOP within a period of less than 3 seconds. → The STOP-LED blinks (overall reset procedure).
- The overall reset has been completed when the STOP-LED is on permanently. → The STOP-LED is on.

The following figure illustrates the above procedure:



Overall reset by means of the Siemens SIMATIC manager

Condition

The operating mode of the CPU must be STOP.
You may place the CPU in STOP mode by the menu command **PLC** > *Operating mode*.

Overall reset

You may request the overall reset by means of the menu command **PLC** > *Clean/Reset*.

In the dialog window you may place your CPU in STOP mode and start the overall reset if this has not been done as yet.

The STOP-LED blinks during the overall reset procedure.

When the STOP-LED is on permanently the overall reset procedure has been completed.

Automatic reload

At this point the CPU attempts to reload the parameters and the program from the memory card. → The MMC-LED blinks.

When the reload has been completed, the LED is extinguished. The operating mode of the CPU will be STOP or RUN, depending on the position of the function selector

Reset to factory setting

The following approach deletes the internal RAM of the CPU completely and sets it back to the delivery state.

Please regard that the MPI address is also set back to default 2!

- Push down the reset lever for app. 30 seconds. The ST-LED blinks. After a few seconds the LED turns to static light. Count the number of static light phases because now the LED switches between static light and blinking.
- After the 6th static light you release the reset lever and push it down again shortly. Now the green RUN-LED is on once. This means that the RAM is totally deleted.
- Turn the power supply off and on again.

Firmware update

Outline

At all CPUs of the System 300V you may execute a firmware update via MMC starting with firmware version 3.3.0.

The latest 2 firmware versions are to find in the service area at www.vipa.de.

By means of reserved file names in the CPU 31x you may transfer the firmware updates per MMC for the following components:

Component	Possible with CPU version	File name at server	New file name at MMC
CPU	V. 3.3.3	order no._release**.**version	firmware.bin
Profibus master	V. 3.0.5	Bb000089.version	dpm00.bin *)
CP	V. 2.0.6	Bb000125.version	Bb000125.rtb

*) When deploying the CPU firmware version V 3.3.3, the file name of the firmware update has to be dpm.bin!

Attention!

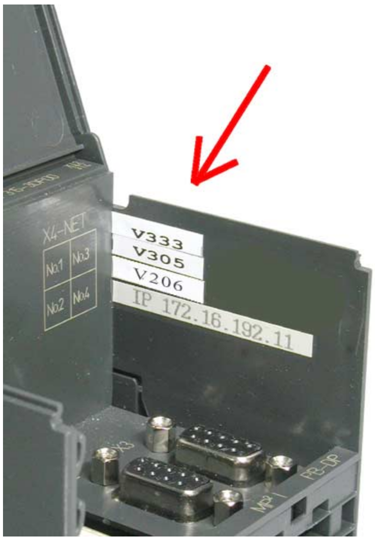
When installing a new firmware you have to be extremely careful. Under certain circumstances you may destroy the CPU, for example if the voltage supply is interrupted during transfer or if the firmware file is defective.

In this case, please call the VIPA-Hotline!

Please regard that the version of the update firmware has to be different from the existing firmware otherwise no update is executed.

Seek firmware version

If you didn't execute a firmware update yet, there are labels beneath the front flap at the right side showing the according firmware versions in following sequence:



CPU firmware version

DP-Master firmware version

CP firmware version

Beneath the CP firmware version is a label with the original IP address of the CP.

You may display the current firmware version of your CPU via the SIMATIC manager from Siemens.

To display the firmware version, you go online with the CPU via your PU res. PC and start the SIMATIC manager from Siemens.

Via **PLC** > *Module status,* register "General", the current firmware version is evaluated and displayed.

Load firmware and transfer it to MMC with reserved name

- Go to www.vipa.de/support/firmware
- Navigate to System 300V. Here the according firmware can be found.
- Download the firmware file. The according structure of the file name can be found in the table above for renaming.
- Rename the file and transfer this file onto a MMC.

Transfer firmware from MMC into CPU

- Get the RUN-STOP lever of your CPU in position STOP.
- Turn off the voltage supply.
- Plug the MMC with the firmware into the CPU. Please take care of the correct plug-in direction of the MMC.
- Turn on the voltage supply.
- After a short boot-up time, the alternate blinking of the LEDs SF and FRCE shows that the firmware file has been found on the MMC.
- You start the transfer of the firmware as soon as you tip the RUN/STOP lever downwards to MRES within 10s. The CPU shows the transfer via a LED blink line.
- During the update process, the LEDs SF, FRCE and MMC are alternately blinking. This may last several minutes.
- The update is successful finished when all CPU-LEDs are on. If they are blinking fast, an error occurred.

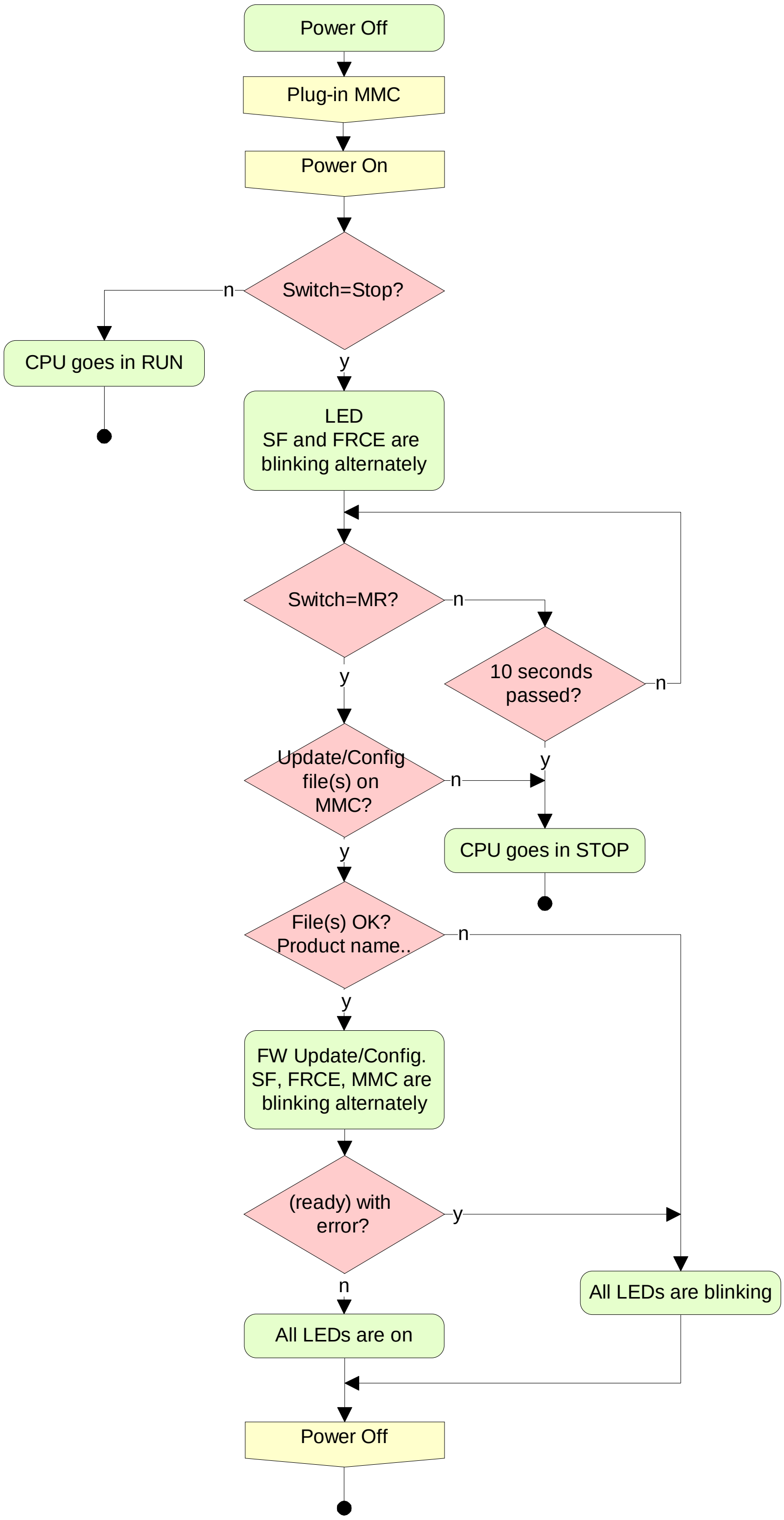
Note!

Starting with the WinNCS version 3.1.1 you may update the firmware of the CP online via the parameterization window. This function is supported by starting with CP firmware version 2.1.2.

For more detailed information see chapter "Deployment CPU 31x under TCP/IP".

Flowchart for firmware update

The following flowchart illustrates the CPU behavior at firmware update:



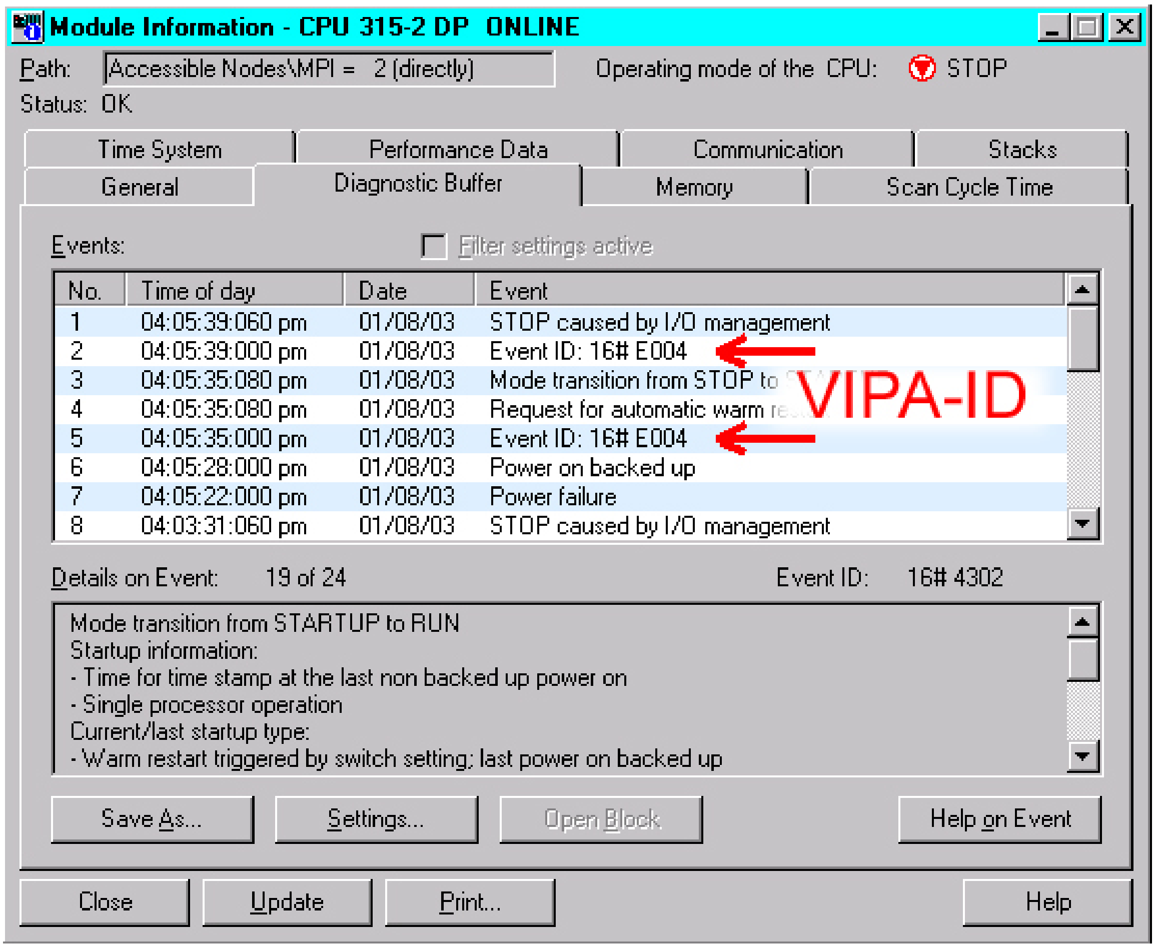
VIPA specific diagnostic entries

Entries in the diagnostic buffer

You may read the diagnostic buffer of the CPU via the Siemens SIMATIC Manager. Besides of the standard entries in the diagnostic buffer, the VIPA CPUs support some additional specific entries in form of event-IDs.

Monitoring the diagnostic entries

To monitor the diagnostic entries you choose the option **PLC** > *Module Information* in the Siemens SIMATIC Manager. Via the register "Diagnostic Buffer" you reach the diagnostic window:



Module Information - CPU 315-2 DP ONLINE

Path: Accessible Nodes\MPI = 2 (directly)

Operating mode of the CPU: STOP

Status: OK

Events: □ Filter settings active

No.	Time of day	Date	Event
1	04:05:39:060 pm	01/08/03	STOP caused by I/O management
2	04:05:39:000 pm	01/08/03	Event ID: 16# E004
3	04:05:35:080 pm	01/08/03	Mode transition from STOP to
4	04:05:35:080 pm	01/08/03	Request for automatic warm re
5	04:05:35:000 pm	01/08/03	Event ID: 16# E004
6	04:05:28:000 pm	01/08/03	Power on backed up
7	04:05:22:000 pm	01/08/03	Power failure
8	04:03:31:060 pm	01/08/03	STOP caused by I/O management

Details on Event: 19 of 24 Event ID: 16# 4302

Mode transition from STARTUP to RUN
Startup information:
- Time for time stamp at the last non backed up power on
- Single processor operation
Current/last startup type:
- Warm restart triggered by switch setting; last power on backed up

The diagnosis is independent from the operating mode of the CPU. You may store a max. of 100 diagnostic entries in the CPU.

The following page shows an overview of the VIPA specific Event-IDs.

Overview of the Event-IDs

Event-ID	Description
0xE003	Error at access to I/O devices Zinfo1: I/O address Zinfo2: Slot
0xE004	Multiple parameterization of a I/O address Zinfo1: I/O address Zinfo2: Slot
0xE005	Internal error - Please contact the VIPA-Hotline!
0xE006	Internal error - Please contact the VIPA-Hotline!
0xE007	Configured in-/output bytes do not fit into I/O area
0xE008	Internal error - Please contact the VIPA-Hotline!
0xE009	Error at access to standard back plane bus
0xE010	Not defined module group at backplane bus recognized Zinfo2: Slot Zinfo3: Type ID
0xE011	Master project engineering at Slave-CPU not possible or wrong slave configuration
0xE012	Error at parameterization
0xE013	Error at shift register access to VBUS digital modules
0xE014	Error at Check_Sys
0xE015	Error at access to the master Zinfo2: Slot of the master (32=page frame master)
0xE016	Maximum block size at master transfer exceeded Zinfo1: I/O address Zinfo2: Slot
0xE017	Error at access to integrated slave
0xE018	Error at mapping of the master I/O devices
0xE019	Error at standard back plane bus system recognition
0xE01A	Error at recognition of the operating mode (8 / 9 Bit)
0xE0CC	Communication error MPI / Serial
0xE100	MMC access error
0xE101	MMC error file system
0xE102	MMC error FAT
0xE104	MMC error at saving
0xE200	MMC writing finished (Copy Ram to Rom)
0xE210	MMC reading finished (reload after overall reset)
0xE300	Internal Flash writing ready (Copy RAM to ROM)

Using test functions for the control and monitoring of variables

Outline

For troubleshooting purposes and to display the status of certain variables you can access certain test functions via the menu item **Test** of the Siemens SIMATIC manager.

The status of the operands and the VKE can be displayed by means of the test function **Debug** > *Monitor*.

You can modify and/or display the status of variables by means of the test function **PLC** > *Monitor/Modify Variables*.

Debug > *Monitor*

This test function displays the current status and the VKE of the different operands while the program is being executed.

It is also possible to enter corrections to the program.

Note!

When using the test function “Monitor” the PLC must be in RUN mode!

The processing of statuses can be interrupted by means of jump commands or by timer and process-related alarms. At the breakpoint the CPU stops collecting data for the status display and instead of the required data it only provides the PG with data containing the value 0.

For this reason, jumps or time and process alarms can result in the value displayed during program execution remaining at 0 for the items below:

- the result of the logical operation VKE
- Status / AKKU 1
- AKKU 2
- Condition byte
- absolute memory address SAZ. In this case SAZ is followed by a "?".

The interruption of the processing of statuses does not change the execution of the program. It only shows that the data displayed is no longer valid from the point on, where the interrupt occurred.

PLC > *Monitor/Modify Variables*

This test function returns the condition of a selected operand (inputs, outputs, flags, data word, counters or timers) at the end of program-execution.

This information is obtained from the process image of the selected operands. During the "processing check" or in operating mode STOP the periphery is read directly from the inputs. Otherwise only the process image of the selected operands is displayed.

Control of outputs

It is possible to check the wiring and proper operation of output-modules.

You can set outputs to any desired status with or without a control program. The process image is not modified but outputs are no longer inhibited.

Control of variables

The following variables may be modified:

I, Q, M, T, C and D.

The process image of binary and digital operands is modified independently of the operating mode of the CPU 31x.

When the operating mode is RUN the program is executed with the modified process variable. When the program continues they may, however, be modified again without notification.

Process variables are controlled asynchronously to the execution sequence of the program.

Chapter 5 Deployment CPU 31x with Profibus

Overview

Content of this chapter is the deployment of the CPU 31x with Profibus. After a short overview the project engineering and parameterization of a CPU 31x with integrated Profibus part VIPA is shown.

Here you will also get information how the Profibus part can be used as DP master and DP slave.

The chapter is finished by hints for commissioning and start-up behavior.

Content

Topic **Page**

Overview

Profibus DP

Profibus is an international standard applicable to an open and serial fieldbus for building, manufacturing and process automation that can be used to create a low (sensor-/actuator level) or medium (process level) performance network of programmable logic controllers.

Profibus comprises an assortment of compatible versions. The following details refer to Profibus DP.

Profibus DP is a special protocol intended mainly for automation tasks in a manufacturing environment. DP is very fast, offers Plug'n'Play facilities and provides a cost-effective alternative to parallel cabling between PLC and remote I/O. Profibus DP was designed for high-speed data communication on the sensor-actuator level.

The data transfer referred to as "Data Exchange" is cyclical. During one bus cycle, the master reads input values from the slaves and writes output information to the slave.

CPU with DP master

The Profibus DP master is to configure via the hardware configurator from Siemens. Therefore you have to choose the Siemens-CPU 315-2AF03 in the hardware configurator from Siemens.

The transmission of your project engineering into the CPU takes place by means of MPI. This is internally passing on your project data to the Profibus master part. During the start-up the DP master automatically mappes his data areas into the address area of the CPU. A configuration of the CPU part is not required. As external storage medium the Profibus DP master uses the MultiMediaCard (MMC) together with the CPU.

Deployment of the DP-Master with CPU

Via the Profibus DP master up to 125 Profibus DP slaves may be coupled to the CPU. The DP master communicates with the DP slaves and links up its data areas with the address area of the CPU. There may be created maximal 1024Byte Input and 1024Byte Output data.

At every POWER ON res. OVERALL RESET the CPU fetches the I/O mapping data from the master. At DP slave failure, the ER-LED is on and the OB 86 is requested. If this is not available, the CPU switches to STOP and BASP is set.

As soon as the BASP signal comes from the CPU, the DP master is setting the outputs of the connected periphery to zero. The DP master remains in the operating mode RUN independent from the CPU.

Profibus address 1 is reserved

Please regard that the Profibus address 1 is reserved for the system. The address 1 should not be used!

DP slave operation

For the deployment in an super-ordinated master system you first have to project your slave system as CPU 315-2DP (6ES7 315-2AF03-0AB0/V1.2) in *Slave* operation with configured in-/output areas. Afterwards you configure your master system. Assign your slave system to your master system. The assignment to your master systems happens for VIPA CPUs 31x with DP slave/master via *Configured stations* and for VIPA CPUs 31x with DP slave via VIPA GSD file. Here drag the slave system from the hardware catalog and drop it on the master system, onto the master system, choose your slave system and connect it.

Project engineering CPU with integrated Profibus master

Outline

For the project engineering of the Profibus DP master you have to use the hardware manager from Siemens. Your Profibus projects are transferred via MPI to the CPU 31xDPM by means of the "PLC" functions. The CPU passes the data on to the Profibus DP master.

Preconditions

For the project engineering of the Profibus DP master in the CPU 31xDPM the following preconditions have to be fulfilled:

- Hardware-Manager from Siemens has to be installed.
- With deployment of a DP Slave the appropriate GSD is to be installed. If the slave system is a CPU 31x from VIPA no GSD is necessary.
- There is a transfer possibility between configuration tool and CPU 31x.

Note!

For the project engineering of the CPU and the Profibus DP master a thorough knowledge of the STEP®7 manager and the hardware configurator from Siemens is required!

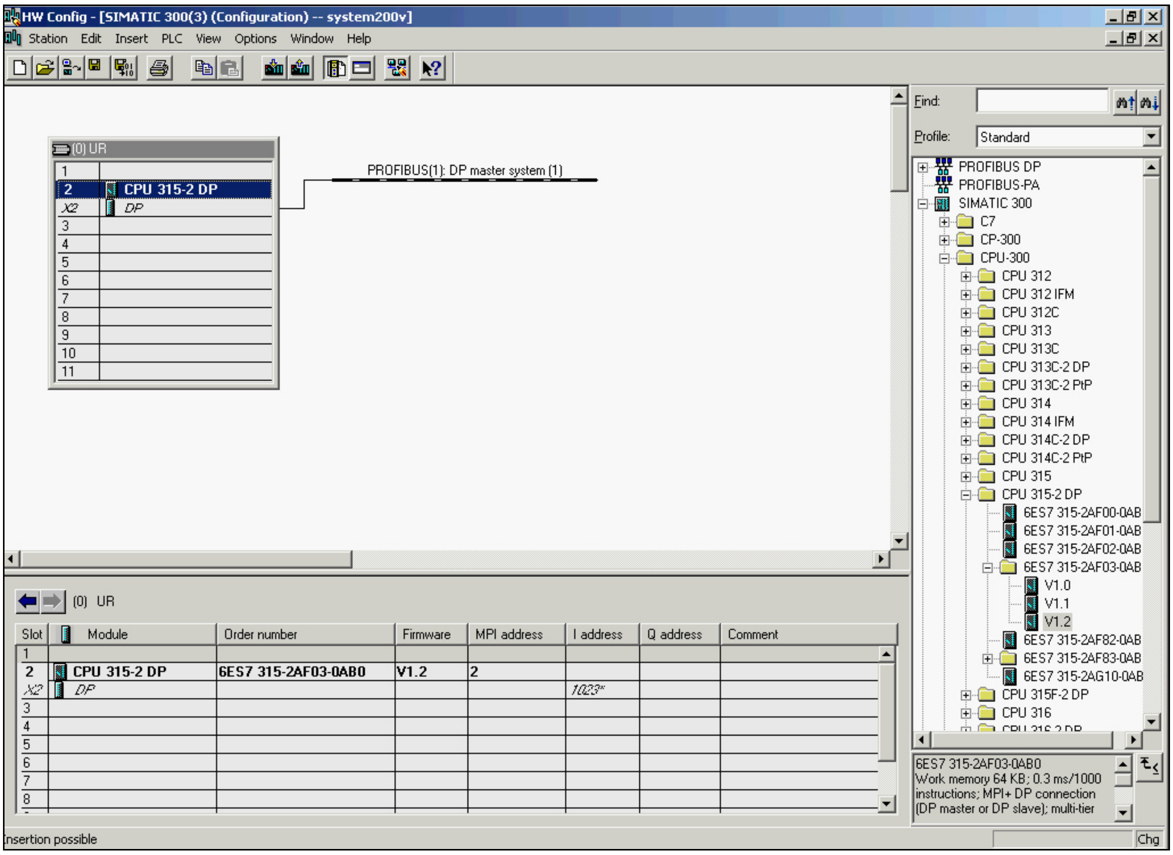
Install Siemens Hardware configurator

The hardware configurator is a part of the STEP®7 projecting tool from Siemens. It serves the project engineering. The modules that may be configured here, are listed in the hardware catalog.

With deployment of a DP Slave the appropriate GSD is to be installed. If the slave system is a CPU 31x from VIPA no GSD is necessary.

Configure DP master

- Create a new project System 300.
- Add a profile rail from the hardware catalog.
- In the hardware catalog the CPU with Profibus master is listed as: `Simatic300/CPU-300/CPU315-2DP/`**`6ES7 315-2AF03-0AB0`**
- Insert the CPU 315-2DP (**6ES7 315-2AF03-0AB0**).
- Type the Profibus address of your master (e.g. 2).
- Click on DP, choose the operating mode "DP master" at *object properties* and confirm your entry with OK.
- Click on "DP" with the right mouse button and choose "add master system".
- Create a new Profibus subnet via NEW.



(0) UR

Slot	Module	Order number	Firmware	MPI address	I address	Q address	Comment
1							
2	CPU 315-2 DP	6ES7 315-2AF03-0AB0	V1.2	2			
X2	DP				1023*		
3							
4							
5							
6							
7							
8							

6ES7 315-2AF03-0AB0
Work memory 64 KB; 0.3 ms/1000 instructions; MPI+ DP connection (DP master or DP slave); multi-tier

Now the project engineering of your Profibus DP master is finished. Please link up now your DP slaves with periphery to your DP master.

- For the project engineering of Profibus DP slaves you search the concerning Profibus DP slave in the *hardware catalog* and drag&drop it in the subnet of your master.
- Assign a valid Profibus address to the DP slave.
- Link up the modules of your DP slave system in the plugged sequence and add the addresses that should be used by the modules.
- If needed, parameterize the modules.

Note!

If you deploy an IM153 from Siemens with a VIPA CPU 31xDPM, please use the "compatible" DP slave modules.

These are listed in the hardware catalog at `PROFIBUS DP/ Additional Field Devices/Compatible Profibus DP-Slaves.`

Slave operation with firmware 3.4.0 or higher

Starting with the CPU firmware 3.4.0, you may deploy your DP master also as DP slave. The approach is described on the following page at Project engineering of the CPU with integrated Profibus slave".

Transfer project

Details about project transfer can be found at chapter "Deployment CPU 31x" at "Project transfer".

Deployment as Profibus DP slave

Fast introduction

The deployment of the Profibus section as "intelligent" DP slave happens exclusively at master systems that may be configured in the Siemens SIMATIC manager. The following steps are required:

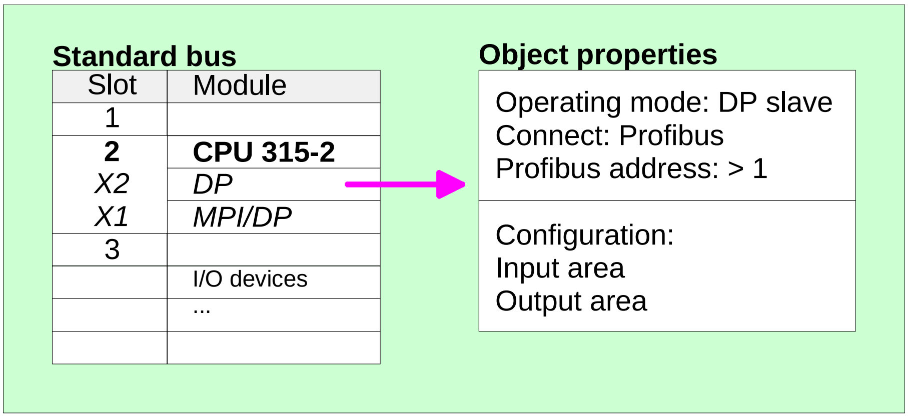
- Start the Siemens SIMATIC manager and configure a CPU 315-2DP with the operating mode *DP slave*.
- Connect to Profibus and configure the in-/output area for the slave section.
- Save and compile your project.
- Configure another station as CPU 315-2DP with operating mode *DP master.*
- Connect to *Profibus* and configure the in-/output ranges for the master section
- Save and compile your project.

In the following these steps are more detailed.

Project engineering of the slave section

- Start the Siemens SIMATIC manager with a new project.
- Insert a *SIMATIC 300 station* and name it as "...DP slave"
- Open the hardware configurator and insert a profile rail from the hardware catalog.
- Place the following Siemens CPU at slot 2:
  CPU 315-2DP (6ES7 315-2AF03-0AB0 V1.2)
- Add your modules according to the real hardware assembly.
- Connect the CPU to *Profibus*, set a Profibus address >1 (preferably 3) and switch the Profibus section via *operating mode* to "slave operation".
- Via *Configuration* you define the in-/output address area of the slave CPU that shall be assigned to the DP slave.
- Save, compile and transfer your project.

Slave section



Standard bus

Slot	Module
1	
2	**CPU 315-2**
X2	*DP*
X1	*MPI/DP*
3	
	I/O devices
	...

Object properties

Operating mode: DP slave
Connect: Profibus
Profibus address: > 1

Configuration:
Input area
Output area

Project engineering of the master section

- Insert another *SIMATIC 300 station* and name it as "...DP master".
- Open the hardware configurator and insert a profile rail from the hardware catalog.
- Place the following Siemens CPU at slot 2:
  CPU 315-2DP (6ES7 315-2AF03-0AB0 V1.2)
- Add your modules according to the real hardware assembly.
- Connect the CPU to *Profibus*, set a Profibus address >1 (preferably 2) and switch the Profibus section via *operating mode* to "master operation".

Assignment

Dependent on the used VIPA CPU 31x of the slave part the assignment happens either via Configured stations or by installing the VIPA GSD file VIPA802F.gsd.

Assignment VIPA CPU 31x with DP slave/master

- Connect your slave system to the master system by dragging the "CPU 31x" from the hardware catalog at *configured stations* onto the master system and select your slave system.

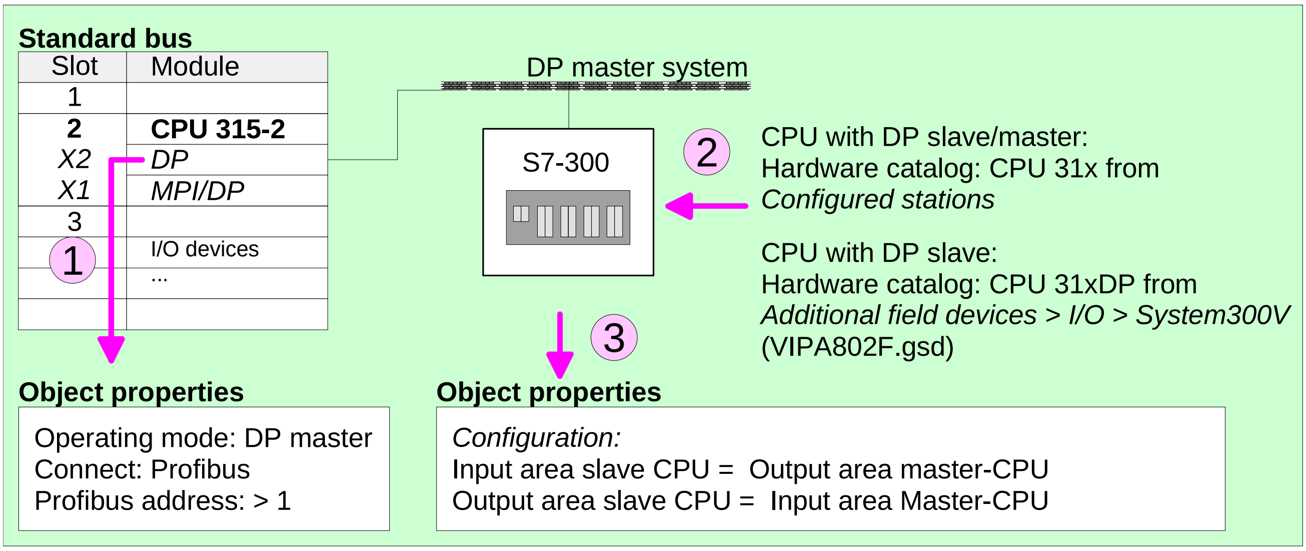
Assignment VIPA CPU 31x with DP slave

- In the service area of www.vipa.de the GSD file VIPA802F.gsd will be found. Install this GSD at your hardware catalog.
- Connect your slave system to the master system by dragging the "CPU 31xDP" of the hardware catalog from *Additional field devices > I/O > VIPA_System_300V* onto the master system.

Connection

- Open the *Configuration* at *Object properties* of your slave system.
- Via double click to the according configuration line you connect the according input address area on the master CPU to the slave output data and the output address area to the slave input data. Data consistency can only be guaranteed for one unit! The choice "Data consistency by length" is not supported.
- Save and compile your project.

Master section



Standard bus

Slot	Module
1	
2	**CPU 315-2**
X2	*DP*
X1	*MPI/DP*
3	
	I/O devices
	...

CPU with DP slave/master:
Hardware catalog: CPU 31x from
Configured stations

CPU with DP slave:
Hardware catalog: CPU 31xDP from
Additional field devices > I/O > System300V
(VIPA802F.gsd)

Object properties

Operating mode: DP master
Connect: Profibus
Profibus address: > 1

Object properties

Configuration:
Input area slave CPU = Output area master-CPU
Output area slave CPU = Input area Master-CPU

Profibus installation guidelines

Profibus in general

- A Profibus DP network may only be built up in linear structure.
- Profibus DP consists of minimum one segment with at least one master and one slave.
- A master has always be deployed together with a CPU.
- Profibus supports max. 126 participants.
- Per segment a max. of 32 participants is permitted.
- The max. segment length depends on the baud rate:

9.6 ... 187.5kBaud	→	1000m
500kBaud	→	400m
1.5MBaud	→	200m
3 ... 12MBaud	→	100m

- Max. 10 segments may be built up. The segments are connected via repeaters. Every repeater counts for one participant.
- All participants are communicating with the same baudrate. The slaves adjust themselves automatically on the baudrate.
- The bus has to be terminated at both ends.
- Master and slaves are free combinable.

Transfer medium

As transfer medium Profibus uses an isolated twisted-pair cable based upon the RS485 interface.

The RS485 interface is working with voltage differences. Though it is less irritable from influences than a voltage or a current interface. You are able to configure the network as well linear as in a tree structure. Your VIPA CPU 31x includes a 9pin slot where you connect the Profibus coupler into the Profibus network as a slave.

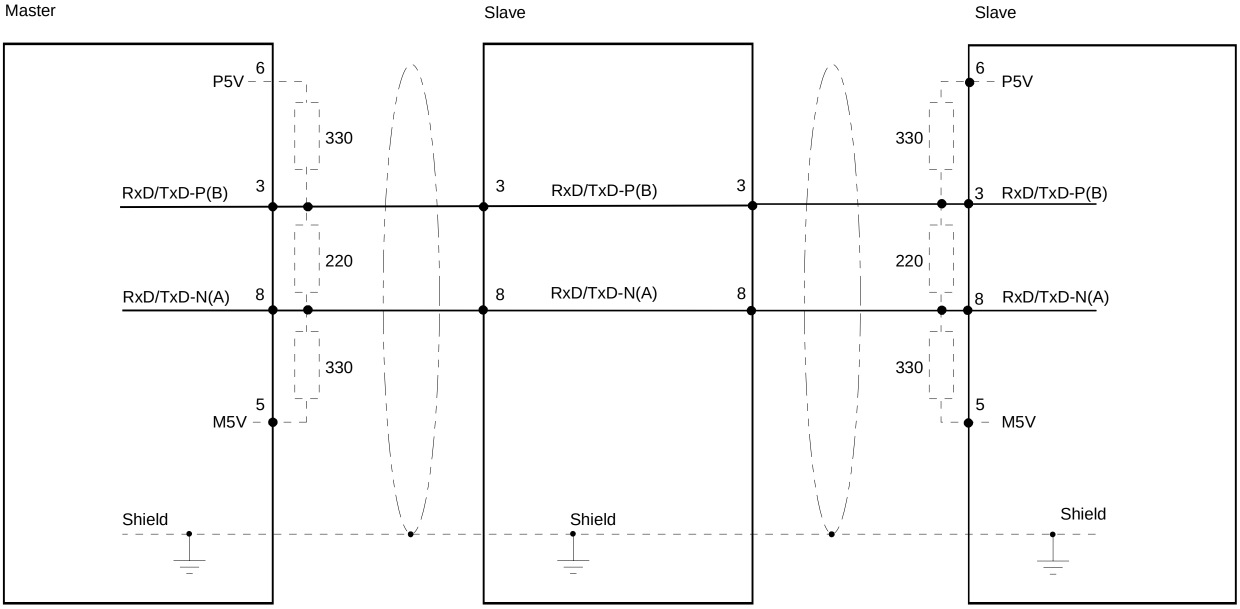
Max. 32 participants per segment are permitted. The segments are connected via repeaters. The maximum segment length depends on the transfer rate.

Profibus DP uses a transfer rate between 9.6kBaud and 12MBaud, the slaves are following automatically. All participants are communicating with the same baudrate.

The bus structure of RS485 allows an easy connection res. disconnection of stations as well as starting the system step by step. Later expansions don't have any influence on stations that are already integrated. The system realizes automatically if one partner had a fail down or is new in the network.

Bus connection

The following picture illustrates the terminating resistors of the respective start and end station.



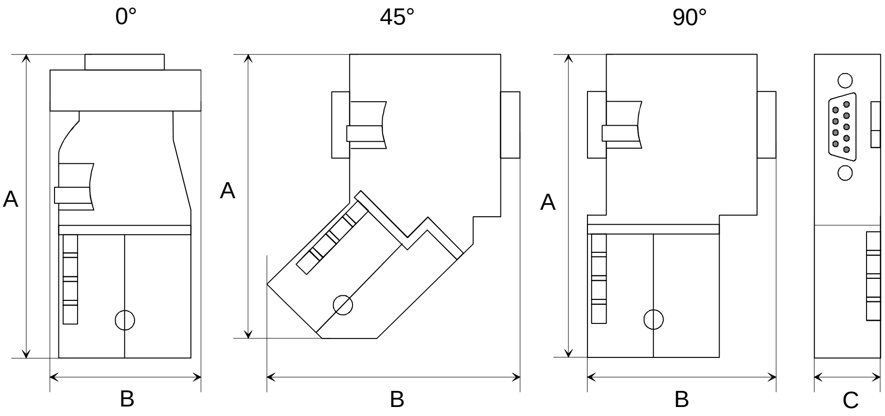
Note!

The Profibus line has to be terminated with its ripple resistor. Please make sure to terminate the last participants on the bus at both end by activating the terminating resistor.

"EasyConn" Bus connector

In systems with more than two stations all partners are wired in parallel. For that purpose, the bus cable must be feed-through uninterrupted.

Via the order number VIPA 972-0DP10 you may order the bus connector "EasyConn". This is a bus connector with switchable terminating resistor and integrated bus diagnostic.



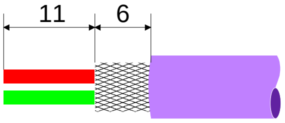
	0°	45°	90°
A	64	61	66
B	34	53	40
C	15,8	15,8	15,8

all in mm

Note!

To connect this plug, please use the standard Profibus cable type A (EN50170). Starting with release 5 you also can use highly flexible bus cable: Lapp Kabel order no.: 2170222, 2170822, 2170322.

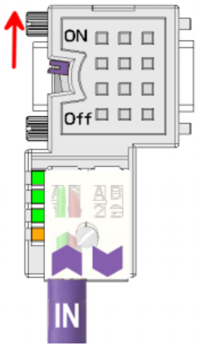
Under the order no. 905-6AA00 VIPA offers the "EasyStrip" de-isolating tool, that makes the connection of the EasyConn much easier.

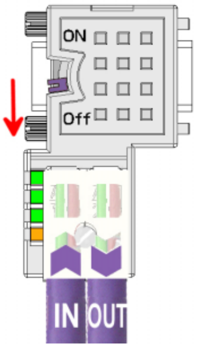


Dimensions in mm

Termination with "EasyConn"

The "EasyConn" bus connector is provided with a switch that is used to activate a terminating resistor.





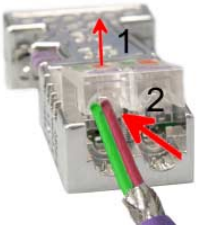
Attention!

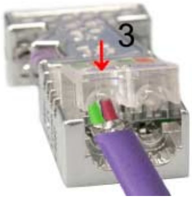
The terminating resistor is only effective, if the connector is installed at a slave and the slave is connected to a power supply.

Note!

A complete description of installation and deployment of the terminating resistors is delivered with the connector.

Assembly





- Loosen the screw.
- Lift contact-cover.
- Insert both wires into the ducts provided (watch for the correct line color as below!)
- Please take care not to cause a short circuit between screen and data lines!
- Close the contact cover.
- Tighten screw
(max. tightening torque 4Nm).

Please note:

The green line must be connected to A, the red line to B!

Commissioning and Start-up behavior

Start-up on delivery

In delivery the CPU is overall reset. The Profibus part is deactivated and its LEDs are off after Power ON.

Online with bus parameter without slave project

The DP master can be served with bus parameters by means of a hardware configuration. As soon as these are transferred the DP master goes online with his bus parameter. This is shown by the RUN LED. Now the DP master can be contacted via Profibus by means of his Profibus address. In this state the CPU can be accessed via Profibus to get configuration and DP slave project.

Slave configuration

If the master has received valid configuration data, he switches to *Data Exchange* with the DP Slaves. This is indicated by the DE-LED.

CPU state controls DP master

After Power ON respectively a receipt of a new hardware configuration the configuration data and bus parameter were transferred to the DP master.

The DP master does not have any operation switch. His state is controlled by the RUN/STOP state of the CPU.

Dependent on the CPU state the following behavior is shown by the DP master:

Master behavior at CPU RUN

- The global control command "Operate" is sent to the slaves by the master. Here the DE-LED is ON.
- Every connected DP slave is cyclically attended with an output telegram containing recent output data.
- The input data of the DP slaves were cyclically transferred to the input area of the CPU.

Master behavior at CPU RUN

- The global control command "Clear" is sent to the slaves by the master. Here the DE-LED is blinking.
- DP slaves with f*ail safe mode* were provided with output telegram length "0".
- DP slaves without *fail safe mode* were provided with the whole output telegram but with output data = 0.
- The input data of the DP slaves were further cyclically transferred to the input area of the CPU.

Chapter 6 Deployment CPU 31x with TCP/IP

Outline

The following chapter describes the deployment of the CPU 31xNET and the communication with TCP/IP.

Content

Topic **Page**

Principles

Network (LAN)

A network res. LAN (**L**ocal **A**rea **N**etwork) provides a link between different stations that enables them to communicate with each other.

Network stations can consist of PCs, IPCs, TCP/IP adapters, etc.

Network stations are separated by a minimum distance and connected by means of a network cable. The combination of network stations and the network cable represent a complete segment.

All the segments of an network form the Ethernet (physics of a network).

Twisted Pair

In the early days of networking the Triaxial- (yellow cable) or thin Ethernet cable (Cheapernet) was used as communication medium. This has been superseded by the twisted pair network cable due to its immunity to interference. The CPU 31xNET module has a twisted-pair connector.

The twisted pair cable consists of 4 cores that are twisted together in pairs. Due to these twists, this system provides an increased level of immunity to electrical interference.

Where the coaxial Ethernet networks are based on a bus topology the twisted pair network is based on a point-to-point scheme.

The network that may be established by means of this cable has a star topology. Every station is connected to the hub/switch by means of a separate cable. The hub/switch provides the interface to the Ethernet.

Hub

The hub is the central element that is required to implement a twisted pair Ethernet network.

It is the job of the hub to regenerate and to amplify the signals in both directions. At the same time it must have the facility to detect and process segment wide collisions and to relay this information. The hub is not accessible by means of a separate network address since it is not visible to the stations on the network.

A hub has provisions to interface with thin- and/or thick-Ethernet or to another hub res. switch.

Switch

A switch also is a central element for realizing Ethernet on Twisted Pair. Several stations res. hubs are connected via a switch. Afterwards they are able to communicate with each other via the switch without interfering the network. An intelligent hardware analyzes the incoming telegrams of every port of the switch and passes them collision free on to the destination stations of the switch. A switch optimizes the bandwidth in every connected segment of a network. Switches enable exclusive connections between the segments of a network changing at request.

Access Control

Ethernet supports the principle of random bus accesses: every station on the network accesses the bus independently as and when required. These accesses are coordinated by a CSMA/CD (Carrier Sense Multiple Access/Collision Detection) scheme: every station "listens" on the bus cable and receives communication messages that are addressed to it.

Stations only initiate a transmission when the line is unoccupied. In the event that to participants should start transmitting simultaneously, they will detect this and stop transmitting, to restart after a random delay time has expired.

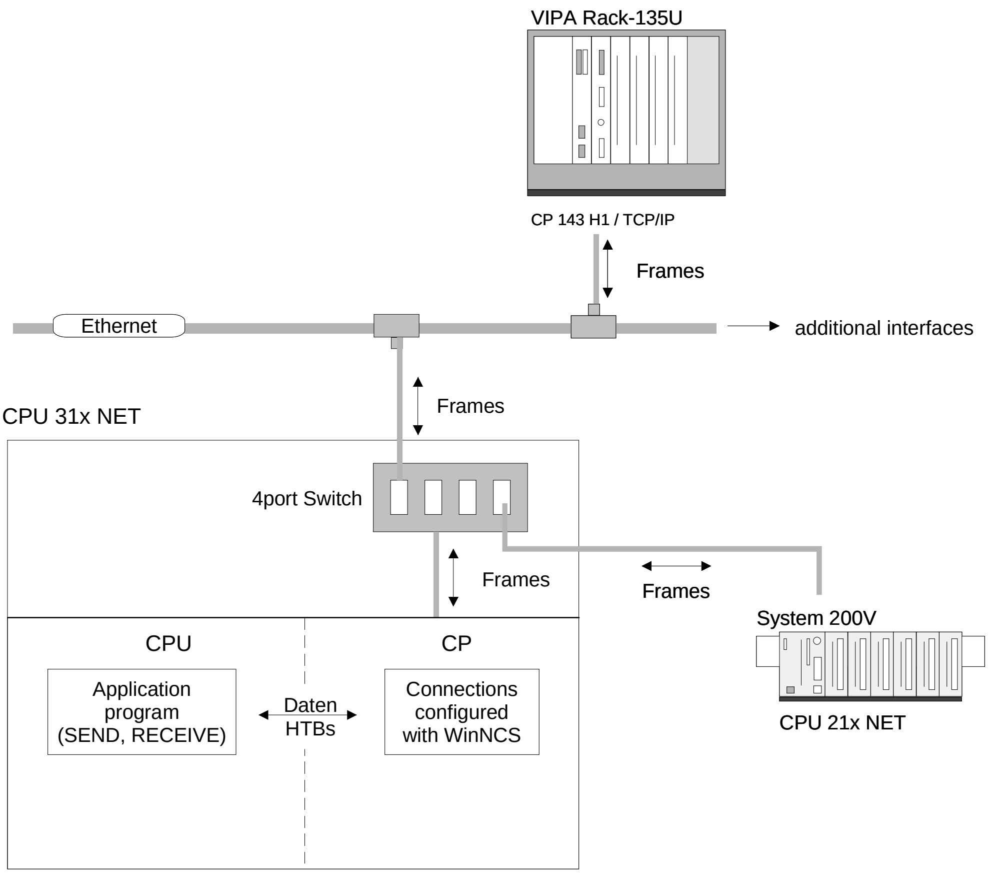
Communication

The internal CP of the CPU 31xNET is directly connected to the CPU 31x by means of a Dual-Port-RAM. The Dual-Port-RAM is divided into 4 equal segments called page frames.

These 4 page frames are available at the CPU as Standard-CP-Interface. Data is exchanged by means of standard handler blocks (SEND and RECEIVE).

The communication under TCP/IP is controlled by means of connections. These are defined by means of the VIPA configuration tool WinNCS and transferred into the CPU via the Twisted-Pair connector.

For details on the configuration, please refer to the WinNCS manual (HB91)



Protocols

Outline

Protocols define a set of instructions or standards that enable computer to establish communication connections and exchange information as error free as possible.

A commonly established protocol for the standardization of the complete computer communication is the so called ISO/OSI layer model, a model based upon seven layers with rules for the usage of hardware and software.

More detailed information about the ISO/OSI layer model is to find in the chapter "Principles".

The CPU 31xNET from VIPA uses the following protocols that are described in the following:

- TCP/IP
- UDP
- RFC1006

TCP/IP

TCP/IP protocols are available on all major systems. At the bottom end this applies to simple PCs, through to the typical mini-computer up to mainframes (TCP/IP implementations also exist for IBM-systems) and special processors like vector processors and parallel computers.

For the wide spread of internet accesses and connections, TCP/IP is often used to assemble heterogeneous system pools.

TCP/IP, standing for **T**ransmission **C**ontrol **P**rotocol and **I**nternet **P**rotocol, collects a various range of protocols and functions.

TCP and IP are only two of the protocols required for the assembly of a complete architecture. The application layer provides programs like "FTP" and "Telnet" for the PC.

The application layer of the Ethernet portion of the CPU 31xNET is defined with the user application using the standard handling blocks.

These user applications use the transport layer with the protocols TCP and UDP for the data transfer which themselves communicate via the IP protocol with the internet layer.

IP

The internet protocol covers the network layer (Layer 3) of the ISO/OSI layer model.

The purpose of IP is to send data packages from on PC to another passing several other PCs. These data packages are referred to as datagrams. The IP doesn't guarantee the correct sequence of the datagrams nor the delivery at the receiver.

For the unambiguous identification between sender and receiver 32Bit addresses (IP addresses) are used that are normally written as four octets (exactly 8Bit), e.g. 172.16.192.11.

These internet addresses are defined and assigned worldwide from the DDN network (Defense Department Network), thus every user may communicate with all other TCP/IP users.

One part of the address specifies the network, the rest serves the identification of the participants inside the network. The boarder between the network and the host portion is variable and depends on the size of the network.

To save IP addresses, so called gateways are used that have one official IP address and cover the network. Then the network can use any IP address.

TCP

The TCP (**T**ransmission **C**ontrol **P**rotocol) bases directly on the IP and thus covers the transport layer (layer 4) of the OSI layer model. TCP is a connection orientated end-to-end protocol and serves the logic connection between two partners.

TCP guarantees the correct sequence and reliability of the data transfer. Therefore you need a relative large protocol overhead that slows down the transfer speed.

Every datagram gets a header of at least 20 octets. This header also contains a sequence number identifying the series. This has the consequence that the single datagrams may reach the destination on different ways through the network.

UDP

The UDP (**U**ser **D**atagram **P**rotocol) is a connection free transport protocol. It has been defined in the RFC768 (Request for Comment). Compared to TCP, it has much fewer characteristics.

The addressing happens via port numbers.

UDP is a fast unsafe protocol for it doesn't care about missing data packages nor about their sequence.

RFC1006

The TCP transport service works stream orientated. This means that data packages assembled by the user not necessarily have to receive the partner in the same packaging. Packages may reach the partner and the package borders are not recognized.

However, most applications require a message orientation.

This means that you need another protocol above TCP. This purpose is defined in the protocol RFC1006. The protocol definition describes the function of an ISO transport interface (ISO 8072) basing upon the transport interface TCP (RFC793).

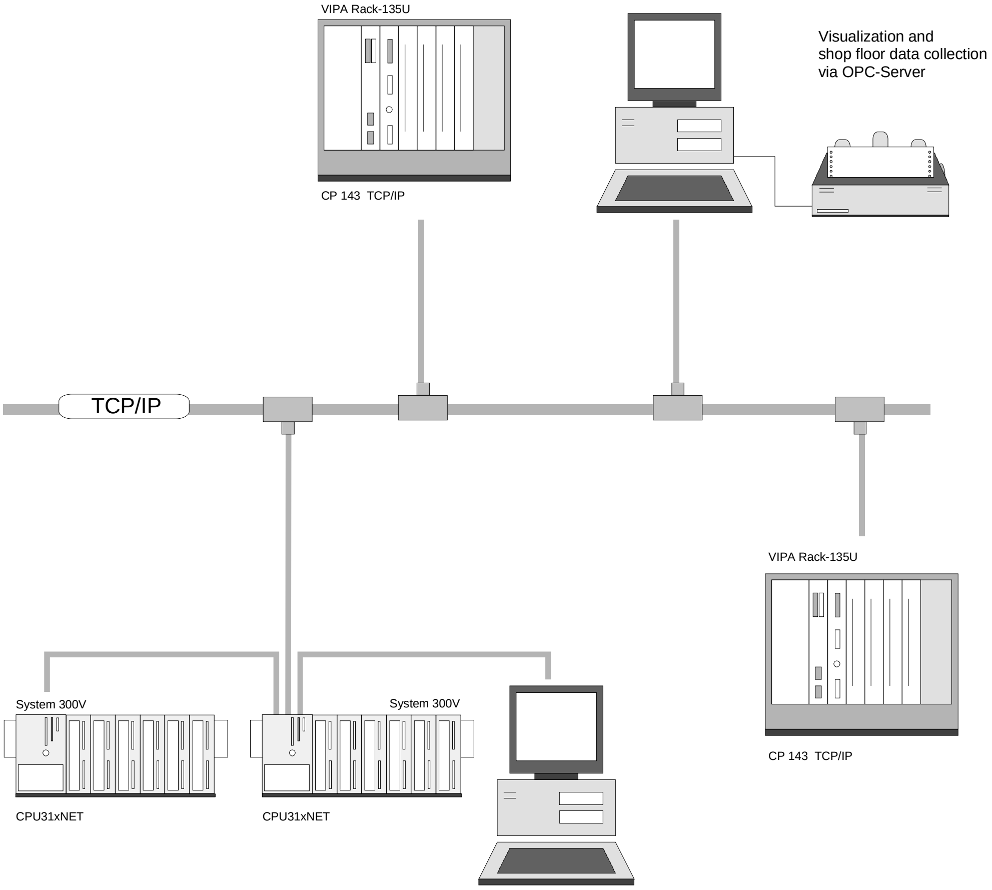
The basic protocol of RFC1006 is nearly identical to TP0 (Transport Protocol, Class 0) in ISO 8073.

For RFC1006 is run as protocol for TCP, the decoding takes place in the data section of the TCP package.

Properties

- The project engineering happens via VIPA WinNCS V 3.02 or higher.
- With RFC1006 only SEND and RECEIVE commands are permissible.
- If the CP should act as an active partner, a SEND order must be projected, should it act as an passive partner you have to project a RECEIVE order.
- For every SEND order, the CP automatically creates a RECEIVE order and vice versa. The assignment of order numbers for the automatically created orders is following this rules:
    - SEND commands may only have odd order numbers. The automatically generated RECEIVE order gets the next higher even order number. (Example: SEND, ANR:5 → RECEIVE, ANR:6)
    - RECEIVE commands may only have even order numbers. The automatically created SEND order gets the next lower odd order number. (Example: RECEIVE, ANR:2 → SEND, ANR:1)
- Automatically created orders are not monitored in WinNCS.
- For the addressing TSAPs are used instead of ports besides of the IP address.
- The TSAP definition differentiates between small letters and capitals.
- The maximum length of TSAPs is 8 signs. The length has to be preset in WinNCS.
- Independently from the used protocol the standard SEND and RECEIVE commands of the PLC are necessary for data transfer.

Example of a TCP/IP application



Network planning

General

The main property of the bus structure is that it consists of a single physical connection. The physical communication medium consists of:

- one or more electrical cables (twisted pair cable)
- coaxial cable (Triaxial cable)
- fiber optic cables

The applicable rules and regulations must be satisfied in order to establish reliable communications between the different stations.

These agreements define the form of the data protocol, the method of access to the bus and other principles that are important for reliable communications.

The VIPA CPU 31xNET was developed in accordance with the standards defined by ISO.

Standards and guidelines

International and national committees have defined the following standards and guidelines for networking technologies:

ANSI American National Standards Institute
The ANSI X3T9.5 standard currently defines the provisions for high speed LAN’s (100 MB/s) based on fiber optic technology.
(FDDI) Fiber Distributed Data Interface.

CCITT Committee Consultative Internationale de Telephone et Telegraph.
Amongst others, this advisory committee has produced the provisions for the connection of industrial networks (MAP) to office networks (TOP) on Wide Area networks (WAN).

ECMA European Computer Manufacturers Association.
Has produced various MAP and TOP standards.

EIA Electrical Industries Association (USA)
This committee has issued standard definitions like RS-232 (V.24) and RS-511.

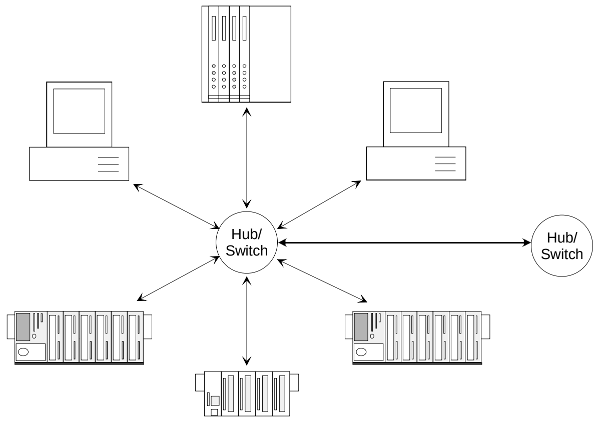
IEC International Electrotechnical Commission.
Defines certain special standards, e.g. for the Field bus.

ISO International Organization for Standardization.
This association of national standards organizations developed the OSI-model (ISO/TC97/SC16). It provides the framework for the standardization of data communications. ISO standards are included in different national standards like for example UL and DIN.

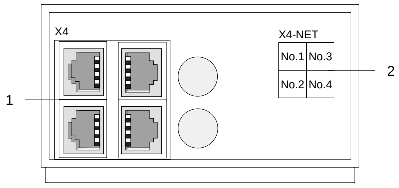
IEEE Institute of Electrical and Electronic Engineers (USA).
The project-group 802 determines LAN-standards for transfer rates of 1 to 20MB/s. IEEE standards often form the basis for ISO-standards, e.g. IEEE 802.3 = ISO 8802.3.

Overview over the components

A twisted pair network can only be constructed with a star topology.



CPU 31xNET

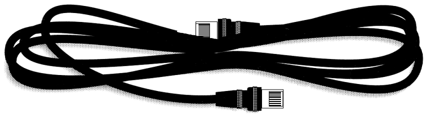


[1] 4port Switch

[2] Switch assignment

Twisted pair cable

At twisted pair cable has four conductors twisted together in pairs. The different conductors have a diameter of 0.4 to 0.6mm.



Restrictions

There are a few restrictions and rules that must be taken into account using twisted pair:

- Maximum number of hub elements per segment 2
- Maximum length of one segment 100m

Analyzing the requirements

- What is the size of the area that must be served by the network?
- How many network segments provide the best solution for the physical (space, interference related) conditions encountered on site?
- How many network stations (SPS, IPC, PC, transceiver, bridges if required) must be connected to the cable?
- What is the distance between the different stations on the network?
- What is the expected “growth rate” and the expected number of connections that must be catered for by the system?
- What data amount has to be handled (band width, accesses/sec.)?

Drawing a network diagram

Draw a diagram of the network. Identify every hardware item (i.e. station cable, hub, switch). Observe the applicable rules and restrictions.

Measure the distance between all components to ensure that the maximum length is not exceeded.

IP address

IP address structure

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

The structure of an IP address is as follows: XXX.XXX.XXX.XXX

Range: 000.000.000.000 to 255.255.255.255

The network administrator also defines IP addresses.

The broadcast address (transmit a message to all stations) is always: 255.255.255.255

Attention!

Certain IP addresses are restricted e.g. 000.000.000.000 These addresses are reserved for special services!

Initial Address

When the CPU 31xNET is first turned on, the module has an predefined initial IP address.

This address is available from a label that has been attached to the side of the module res. in the protocol added to your CPU.

At the first commissioning, the CP portion has the following default settings for IP address and subnet mask:

IP Address:	**172.16.192.11**
Subnet Mask:	**255.255.255.0**

Starting with WinNCS version V 3.12 you have the comfortable possibility to search stations in the Ethernet, to import their project engineering an to change their IP addresses without changing the IP circuit.

Attention!

Please regard! If you change the IP Address of the CP by using [Change IP], the configuration inside the CP is cleared.

Project Engineering of the CPU 31xNET

Outline

The project engineering procedure for TCP/IP consists of two parts:

- **CP-configuration** by means of VIPA WinNCS (Ethernet connection).
- **Hardware configuration** (including the CP into the CPU)
- **PLC programming** by means of an application program (PLC connect).

Fast introduction

Precondition

CP is connected to Ethernet, powered an running.

CP project engineering under WinNCS

- Start WinNCS.
- By choosing 🖧 you set the functionality "Ethernet".
- Create a project with the function group "Ethernet" via **File** > *Project set-up/open*.
- Click in "Parameter"-window on [Search stations] → The available VIPA CPs are listed by their IP address.
- If your target CP is inside your IP circle, the CP can online be projected. Otherwise you have to change the IP address by using [Change IP]. After that click to [Search stations]. Please regard! If you change the IP Address of the CP by using [Change IP], the configuration inside the CP is cleared.
- Via double click on the wanted station, the according project is imported and monitored in the "network" window for further parameterization.
- Type the *station name, IP address* and *subnet mask* and click on [Apply].

If needed, you get the IP address and the subnet mask from your system operator. The rest of the entries remain in default.

- For the project engineering of the READ/WRITE connection click at 🌐. Type the according parameters for *connection name, order type, order model, order no., priority, IP addresses and ports* and click on [Apply]:

... continue fast introduction

Transfer CP project

In the "network" window mark the station to be transfered.

- Activate the online functions via Online.
- If there is still an online connection to the CP, set the CP into software STOP via Stop an start transfer with CP PC.

  Otherwise set "IP protocol" under Init and type the IP address 172.16.192.11 (delivery address).
- Establish a connection via Conn.
- Set the CP into software STOP via Stop.
- With CP PC, the transfer into the CP starts. If a request for a NCS file appears, you forgot to choose the "CPU 31xNET" in the network window! Return to the last step, choose the correct setting and start the transfer again.
- As soon as the transfer has finished, reboot the CP via Start.

Now the CP is on the network with the assigned IP address.

... continue fast introduction

Hardware configuration

Precondition: SIMATIC manager from Siemens V. 5.1 or V.5.2 with SP1 and SIMATIC NET

- Start the SIMATIC manager from Siemens with a new project.
- Add a new System300 station via **Insert** > *Station* > *SIMATIC 300-Station*.
- Activate the station "SIMATIC 300" and open the hardware configurator by clicking on "Hardware".
- Configure a rack (Simatic300 > Rack-300 > Profile rail).
- For all CPUs 31x from VIPA are configured as CPU 315-2DP, you select the CPU 315-2DP with the order no. 6ES7 315-2AF03-0AB0 in the hardware catalog.
  This is to find under Simatic300 > CPU-300 > CPU 315-2 DP.
- Place your modules starting at plug-in location 4. After that place your virtual CP343-1 TCP(Simatic300 > CP-300 > Industrial Ethernet > CP 343-1 TCP).
- If needed parameterize the CPU res. the modules. The parameter window is opened at double click on the depending module.
- Via a double click on the CP 343-1 TCP you open the properties window and you type that IP address and that Subnet mask into properties that you have defined in WinNCS.
- Save your project.

The transfer of the hardware configuration happens together with the user application.

User application

- Create a communication channel between CPU and CP by means of the SYNCHRON block.
- Program the according SEND and RECEIVE blocks for initializing send and receive orders.
- Program the blocks SEND_ALL res. RECV_ALL for data transfer.

Transfer of user application and hardware configuration

To transfer your user application and the hardware configuration you have the following possibilities:

- Transfer via MPI respectively Profibus
- Transfer via MMC
- Transfer via Ethernet

Details about project transfer can be found at chapter "Deployment CPU 31x" at "Project transfer".

Here the fast introduction ends. The following pages contain a detailed description of the project engineering.

CP-configuration with WinNCS

The CP section of the CPU 31xNET can only be configured by means of WinNCS from VIPA and consists of the following 3 parts:

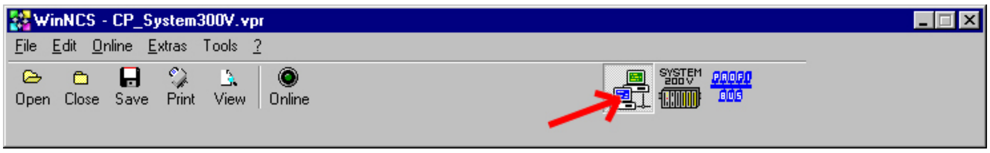
- the initial CP configuration,
- configuration of connection modules
- transfer configuration data into the CP.

Initial CP configuration

This is where the address and other identification parameters of a station are defined. This defines the behavior of the station in the network.

Functionality Ethernet

Start WinNCS.



Click on to set the functionality "Ethernet".

Create a project with the function group "Ethernet" via **File** > *Project set-up/open*. You can either online search stations or offline add a new station.

Online search and import stations

A click on [Search stations] starts the search for all available stations in the network.

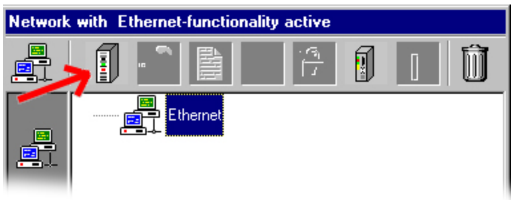
The available CPs are listed by IP address. Via double click you import the project of the current station and monitor it in the "network" window for further configuration.

The online search is for the complete network up to the gateway.

Note!

Please regard, only CPs inside the same IP circle as your PC can be projected. By using [Change IP] you can online change IP address and subnet mask of your CP.

Offline add new station



Add a new station with .

Parameter window

Now the "parameter" window contains the basic initialization of your CP. The following registers are supported:

The "parameter" window contains the following registers:

- CP-Init (Basic initialization of the CP)
- Firmware (Online firmware update)
- Reboot (Start online reboot at the CP)
- Password (Password for online engineering and firmware update)
- IP-List (Access protection for IP addresses)
- Clock-Init (Parameterization as time master)
- Parameter-IP (System parameter for the IP protocol)
- Syskonfig (Parameter for IPK)

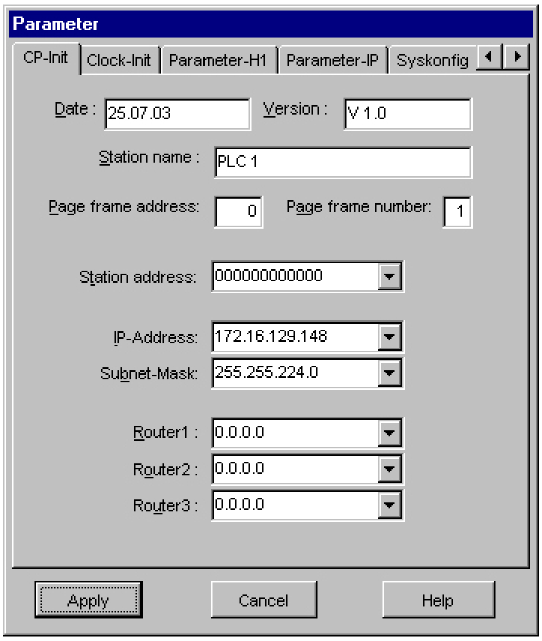
CP basic configuration

The register "CP-Init" contains the basic initialization of your CP. Here you parameterize addresses and other identification parameter of the CP.

Insert values for the following parameters under CP-Init and click on [Apply]:

Station name, *page frame basic address* (0 fix), *page frame amount* (1 fix), *station address* (000000000000 fix), *IP address* and *subnet mask*.

If needed, you get the IP address and the subnet mask from your system operator, the rest of the settings remain unchanged.



Date : 25.07.03 Version : V 1.0

Station name : PLC 1

Page frame address: 0 Page frame number: 1

Station address: 000000000000

IP-Address: 172.16.129.148

Subnet-Mask: 255.255.224.0

Router1 : 0.0.0.0

Router2 : 0.0.0.0

Router3 : 0.0.0.0

Date, Version

The entry of date and version serves your own documentation. It always shows the last parameterization alteration.

Station name

You may assign a name for the station (max. 19 characters). Station names serve the overview and eases the parameterization of connections. Please take care with the symbolic addresses that the symbolic name of a station res. its address and the station name of the parameterization here are identical.

This allows a stringent name assignment.

Page frame basic address

CP is fixed to a page frame basic address of 0.

Page frame number

CP is fixed to a page frame number of 1.

Station address

CP is fixed to Ethernet address (station address) 000000000000.

IP address

The IP address is a 32Bit address that must be unique within the network. The address is a combination of 4 numbers separated by a dot.

To build-up private IP networks within the internet, RFC 1597/1918 reserves the following address ranges:

Class A:	10.0.0.0	...	10.255.255.255
Class B:	172.16.0.0	...	172.31.255.255
Class C:	192.168.0.0	...	192.168.255.255

At first start-up the following address and subnet mask is preset in the CP part:

IP address:	**172.16.192.11**
Subnet mask:	**255.255.255.0**

Subnet mask

The IP subnet mask is a 32Bit filter mask for IP addresses. Deploying subnet masks, subnets are build-up by definition of IP number circles. Subnets only communicate with each other via router.

The address entered here has the same structure and value range like the IP address. For the project engineering of a CP with the PC, you have to make sure that PC and CP have the same subnet mask.

Router

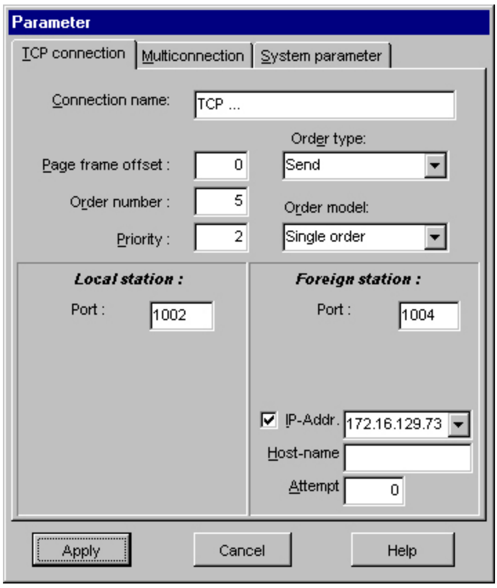
A router is able to connect different network topologies with each other like e.g. Ethernet with token ring. It also serves the connection of subnets. Router have one IP address per network that connect the networks logical. A connection of 2 networks via router has 2 addresses.

If you don’t want to use router IP addresses you enter 0.0.0.0 at router.

Configuration of a Connection Block

A connection block contains both remote parameters, i.e. parameters that are oriented towards the partner on the network, and local parameters, i.e. parameters that apply to the PLC-program of a connection.

You may configure TCP/IP-connections by selecting the symbol of the station and inserting/configuring the respective connection.



Connection name: TCP ...

Order type: Send

Page frame offset : 0

Order number : 5

Priority : 2

Order model: Single order

Local station : Port : 1002

Foreign station : Port : 1004

☑ IP-Addr. 172.16.129.73

Host-name

Attempt 0

Host name is not supported

Configure connections

🌐

To configure connections, click on 🌐. Give values for the following parameters and click on [Apply]:

Connection name, *page frame offset* (0 fix), *order no.*, order type, order model, *priority*, *IP address* and p*ort* for the stations.

Typing an IP address for the foreign station means that the CP may only communicate with this IP address. If you configure a test environment, you should not set an IP address here.

Transfer of the CP project

When all the required connections have been configured, you have to transfer the parameters into the CP. Due to the hardware a serial transfer from the PC into the CP is not possible.

To transfer your WinNCS project you have the following two possibilities:

a) Transfer directly via Ethernet

b) Transfer via MMC as webcp.ncs

to a)
Transfer directly via Ethernet

Every CPU 31x with TCP/IP is delivered with the following addresses:

IP address: **172.16.192.11**

Subnet mask: **255.255.255.0**

This IP address is only valid for the first configuration. For control purposes you find this address on a label under the flap and in the added test protocol.

Note!

Please regard that every IP address must be unique within the network an CP an PC have to be connected to the same IP circle. To rearrange the IP circle you can either change the CP IP address by using [Change IP] or the IP address of your PC by click on "network" with the right mouse button. Via *Properties* you will reach a dialog window. Here you may type a new IP address and subnet mask depending on the operating system.

Before making changes, you should note the recent values in the networks properties.

For the access on the address 172.16.192.11 via Ethernet, you change the IP address of the network card of your projecting PC to:

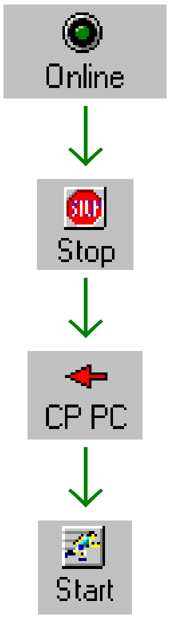
172.16.192.X, where "X" can get all values between 0 and 255 except 11.

For the subnet mask you choose: 255.255.255.0

Transfer the configuration data with online functions

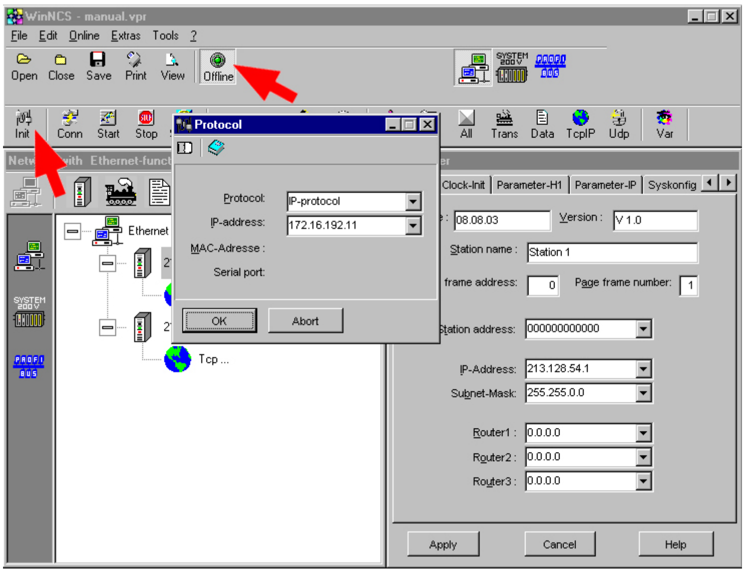
For your projecting PC is in the same IP circle like the CP, you may access the CP via the Ethernet.

In the "network" window mark the station to be transfered.



- Activate the online functions via Online.
- If there is still an online connection to the CP, set the CP into software STOP via Stop an start transfer with CP PC.
- As soon as the transfer has finished, reboot the CP via Start.

... continue to a)
Transfer directly via Ethernet

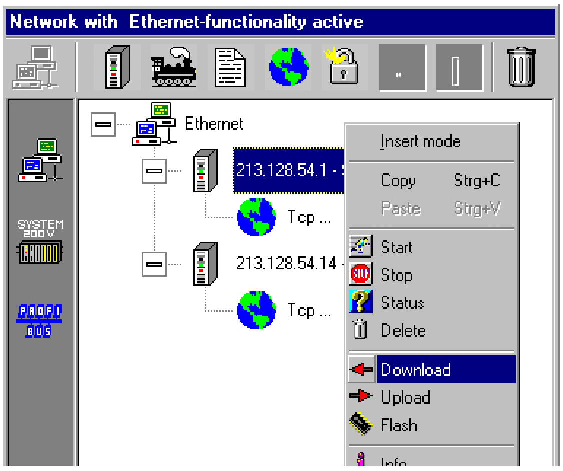

WinNCS - manual.vpr
File Edit Online Extras Tools ?
Open Close Save Print View Offline
Init Conn Start Stop
All Trans Data TcpIP Udp Var
Protocol
Network with Ethernet-functionality active
Ethernet
Tcp ...
Clock-Init Parameter-H1 Parameter-IP Syskonfig
OK Abort
Apply Cancel Help

Protocol: IP-protocol
IP-address: 172.16.192.11
MAC-Adresse :
Serial port:

08.08.03 Version : V 1.0
Station name : Station 1
frame address: 0 Page frame number: 1
Station address: 000000000000
IP-Address: 213.128.54.1
Subnet-Mask: 255.255.0.0
Router1 : 0.0.0.0
Router2 : 0.0.0.0
Router3 : 0.0.0.0

- Otherwise set "IP protocol" under Init and type the IP address 172.16.192.11 (delivery address).
- Establish a connection via Conn.
- Set the CP into software STOP via Stop and select the station "CPU 31xNET" in the network window.
- With CP PC, the transfer into the CP starts. If a request for a NCS file appears, you forgot to choose the "CPU 31xNET" in the network window! Return to the last step, choose the correct setting and start the transfer again.
- As soon as the transfer has finished, reboot the CP via Start.

There is also the possibility to transfer the project to the according CP by means of the right-click context menu.



Establish as shown above a online connection an set the CP into software STOP.

Now switch to the window *Network* and click on the according station. Use the right mouse button and choose "Download".

Your project will now directly be transferred into the CP.

to b)
Transfer via MMC
as webcp.ncs

WinNCS gives you the option to export your project as *.NCS-file. Herefore, you highlight the station a choose Export via the right mouse button. Assign the file name webcp.ncs and transfer this file to the MMC.

The following transfer process bases upon the assumption that the CP project is recognized via the unambiguous file name and transferred via into the CP:

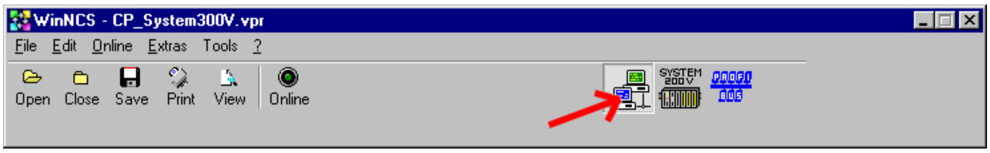
- Set the RUN-STOP switch of your CPU in position STOP.
- Turn off the power supply.
- Plug the MMC with the CP file into the CPU. Please takes care of the correct plug-in direction of the MMC.
- Turn on the power supply.
- After a short start-up time, the alternate blinking of the LEDs SF and FRCE shows that the CP file has been found on the MMC.
- You start the transfer of the file by tipping the RUN/STOP switch for a short time to MR within 10s.
- During data transfer, the LEDs SF, FRCE and MMC are blinking alternately.
- The update has been finished successful, when all CPU-LEDs are on. If they are blinking fast, an error occurred.
- Turn off the power supply, take the MMC out and turn on the power supply again. CPU and CP are rebooted. Now the CP is at the network with the configured IP address.

Control the
WinNCS project

Starting with the WinNCS version 3.1.1 you may get online access to your CP. There you may choose the wanted from all listed stations and import the according connections.

Approach

- Start WinNCS.


WinNCS - CP_System300V.vpr
File Edit Online Extras Tools ?
Open Close Save Print View Online
SYSTEM 200V
PROFI BUS

- Choosing you set the functionality "Ethernet".
- Create a project with the function group "Ethernet" via **File** > *Project set-up/open*.
- Via [Search stations] you may list all CP station with according IP address in the parameter window.
- As soon as you select the wanted station via double click, their connection parameters are imported into the network window. Changes that you are doing here are transferred into the CP with the WinNCS methods described further above. This is not yet possible as online function.

Hardware configuration

For the hardware configuration you use the hardware configurator from Siemens. Among others, you assign the IP address of the CP and configure the hardware components of your PLC.

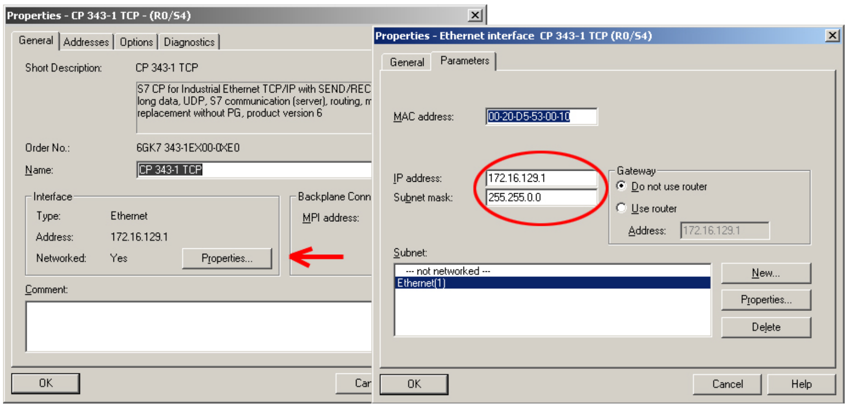
Preconditions

Please regard that the following software packages have to be installed for the hardware configuration:

- SIMATIC manager from Siemens V. 5.1 or V.5.2 with SP1
- SIMATIC NET

Approach

- Start the SIMATIC manager from Siemens and create a new project.
- Insert a new System 300 station via **Insert** > *Station* > *SIMATIC 300 station*.
- Activate the station "SIMATIC 300" and open the hardware configurator by clicking on "Hardware".
- Configure a rack (Simatic300 > Rack-300 > Rail).
- For all CPUs 31x from VIPA are configured as CPU 315-2DP, you select the CPU 315-2DP with the order no. 6ES7 315-2AF03-0AB0 V1.2 in the hardware catalog.
  This is to find under Simatic300 > CPU-300 > CPU 315-2 DP.
- Place your modules starting at plug-in location 4. After that place your virtual CP343-1 TCP(Simatic300 > CP-300 > Industrial Ethernet > CP 343-1 TCP).
- If needed parameterize the CPU res. the modules. The parameter window is opened at double click on the depending module.
- Via a double click on the CP 343-1 TCP you open the properties window and you type that IP address and that Subnet mask into properties that you have defined in WinNCS.



Properties - CP 343-1 TCP - (R0/S4)

Short Description: CP 343-1 TCP

S7 CP for Industrial Ethernet TCP/IP with SEND/REC
long data, UDP, S7 communication (server), routing, m
replacement without PG, product version 6

Order No.: 6GK7 343-1EX00-0XE0

Name: CP 343-1 TCP

Interface
Type: Ethernet
Address: 172.16.129.1
Networked: Yes

Properties - Ethernet interface CP 343-1 TCP (R0/S4)

MAC address: 00-20-D5-53-00-10

IP address: 172.16.129.1

Subnet mask: 255.255.0.0

Gateway
● Do not use router
○ Use router
Address: 172.16.129.1

Subnet:
--- not networked ---
Ethernet(1)

- Save your project.

Now the hardware configuration is ready. The transfer of the hardware configuration happens together with the PLC user application and is described on the following pages under "Online-Transfer of program and hardware configuration".

PLC application programming

To enable the PLC to process connection requests, it requires an active PLC application program on the CPU. This uses the handling blocks (SEND, RECEIVE, ...) that are included in the CPU 31xNET amongst others.

The PLC program also requires that a communication channel between the CPU and the CP ("synchronization") is specified first. This function is performed by the SYNCHRON block.

Transmission and reception is initiated by means of SEND and RECEIVE. A data transfer is initiated by means of SEND_ALL or RECV_ALL.

Error messages will appear in the indicator word.

Synchronization

The used interface of the CP has to be synchronized in the start-up OB OB 100 by means of the handling block SYNCHRON.

After power is turned on, the CPU31xNET requires app. 15s for the boot procedure. If the PLC should issue a request for synchronization during this time, an error is returned in the configuration error byte PAFE. This message is removed when the CP module has completed the boot process.

The timer in this block is initially set to 20s. Processing will be stopped if the synchronization is not completed properly within this period.

Block sizes

The following table shows the available block sizes.

Block size	CP Block size in Byte
0	Default
1	16
2	32
3	64
4	128
5	256
6	512
255	512

Cycle

The sending and receiving blocks SEND and RECEIVE which initiate the send and receive operations must be configured in the cycle program OB1. The blocks SEND_ALL and RECV_ALL perform the actual data-transfer.

Purely passive connections only require the components SEND_ALL or RECV_ALL.

To protect the data transfer you should integrate various checkpoints that evaluate the indicator word.

Handling blocks

The following table lists the required handling blocks. More detailed information is to find in the chapter "Integrated OBs, SFBs and SFCs" .

SFC	Label	Description
SFC 228	RW_frame	Read/Write page frame
SFC 230	Send	Send to CP via page frame
SFC 231	Receive	Receive from CP via page frame
SFC 232	Fetch	Fetch starts the data request via page frame. FETCH is only permissible with RW identifier and delivers the order initialization for read.
SFC 233	Control	The CONTROL block is used for the status request of an order that means that the ANZW of a defined order is updated.
SFC 234	Reset	The RESET block initializes the clearing of am order of the defined connection.
SFC 235	Synchron	The SYNCHRON block serves the synchronization of the CPU and the CP during start-up. At the same time, the page frame is cleared and the block size between CPU and CP is calculated. Active data communication may only be executed between synchronized page frames.
SFC 236	Send_All	Initialization of the data transfer from the CPU to the CP.
SFC 237	Recv_All	Initialization of the data reception from the CP to the CPU.
SFC 238	Control1	Control for page frame communication with type ANZW: Pointer and parameter IND.

Transfer project

Details about project transfer can be found at chapter "Deployment CPU 31x" at "Project transfer".

Control project engineering

- Activate the online functions in WinNCS via Online.
- Select "IP protocol" under Init and type the new IP address.
- Establish a connection via Conn. Now you communicate via the IP address given under CP-Init.
- The CP has to be in RUN. Ensure this via Status. If the CP is in *Idle*-Mode, the synchronization with the CPU has failed. Please control the SYNCHRON block in OB 100.
- For control you may monitor the summary status of the TCP connections via TcpIP.

Now the project engineering of the CPU and the CP is finished.

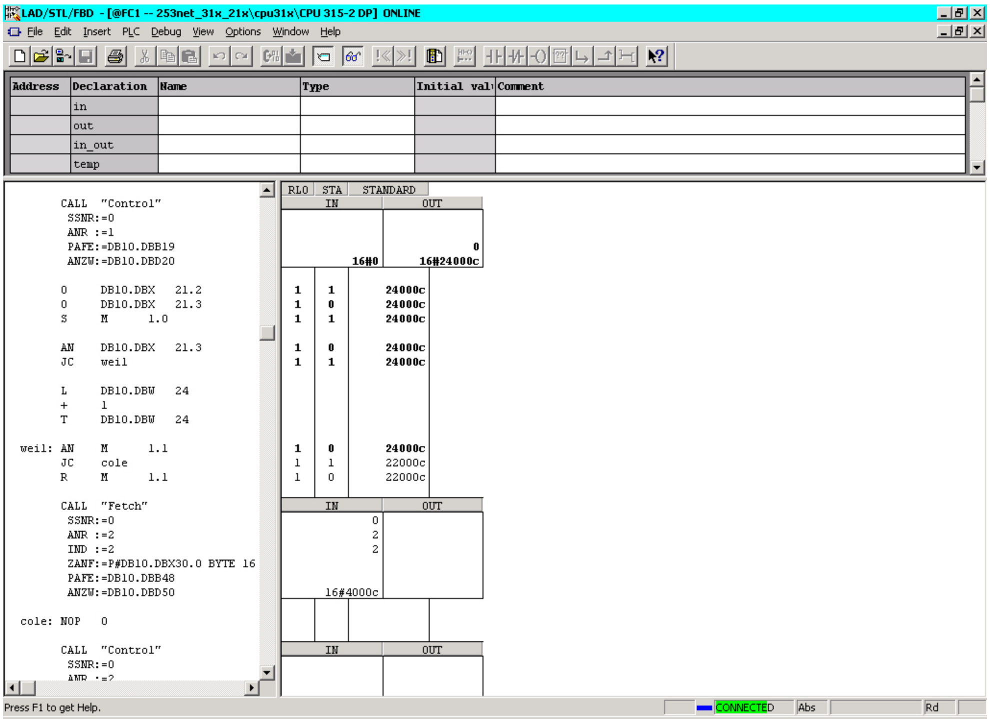
PG/OP communication

Outline

When using a CPU 31xNET you may gain online access to the CPU via the CP regarding some system restrictions. All block functions are supported like for example load, alter, monitor and save blocks.

All "PLC" functions except of "Hardware diagnosis" are available.

Up to 64 participants may access simultaneously.


LAD/STL/FBD - [@FC1 -- 253net_31x_21x\cpu31x\CPU 315-2 DP] ONLINE
File Edit Insert PLC Debug View Options Window Help
Press F1 to get Help.
CONNECTED
Abs
Rd

Address	Declaration	Name	Type	Initial val	Comment
	in				
	out				
	in_out				
	temp				

```
                                        RLO  STA  STANDARD
                                             IN           OUT
      CALL  "Control"
       SSNR:=0
       ANR :=1
       PAFE:=DB10.DBB19                                     0
       ANZW:=DB10.DBD20                          16#0  16#24000c

      O     DB10.DBX   21.2              1    1     24000c
      O     DB10.DBX   21.3              1    0     24000c
      S     M      1.0                   1    1     24000c

      AN    DB10.DBX   21.3              1    0     24000c
      JC    weil                         1    1     24000c

      L     DB10.DBW   24
      +     1
      T     DB10.DBW   24

weil: AN    M      1.1                   1    0     24000c
      JC    cole                         1    1     22000c
      R     M      1.1                   1    0     22000c

                                             IN           OUT
      CALL  "Fetch"
       SSNR:=0                                    0
       ANR :=2                                    2
       IND :=2                                    2
       ZANF:=P#DB10.DBX30.0 BYTE 16
       PAFE:=DB10.DBB48
       ANZW:=DB10.DBD50                    16#4000c

cole: NOP   0
                                             IN           OUT
      CALL  "Control"
       SSNR:=0
       ANR :=2
```

Difference to the PU functionality of a CPU 31x with TCP/IP from Siemens

- Instead of "NetPro" from Siemens, the VIPA-CPU uses the VIPA tool WinNCS V3.0 for connection project engineering. You should regard that the IP address that allows you to access the VIPA-CP via Ethernet may only be set or altered via the VIPA project engineering tool WinNCS. An IP address that you preset in the hardware manager from Siemens will be ignored from the VIPA-CP.

  Vice versa you have to set the IP address of WinNCS also in the "Properties" of the VIPA-CP in the hardware configurator from Siemens so that the Siemens-PU knows the IP address of the VIPA-CP.

- The PLC system functions are only permissible for the CPU. Used for other modules, it causes an communication error at the PG.

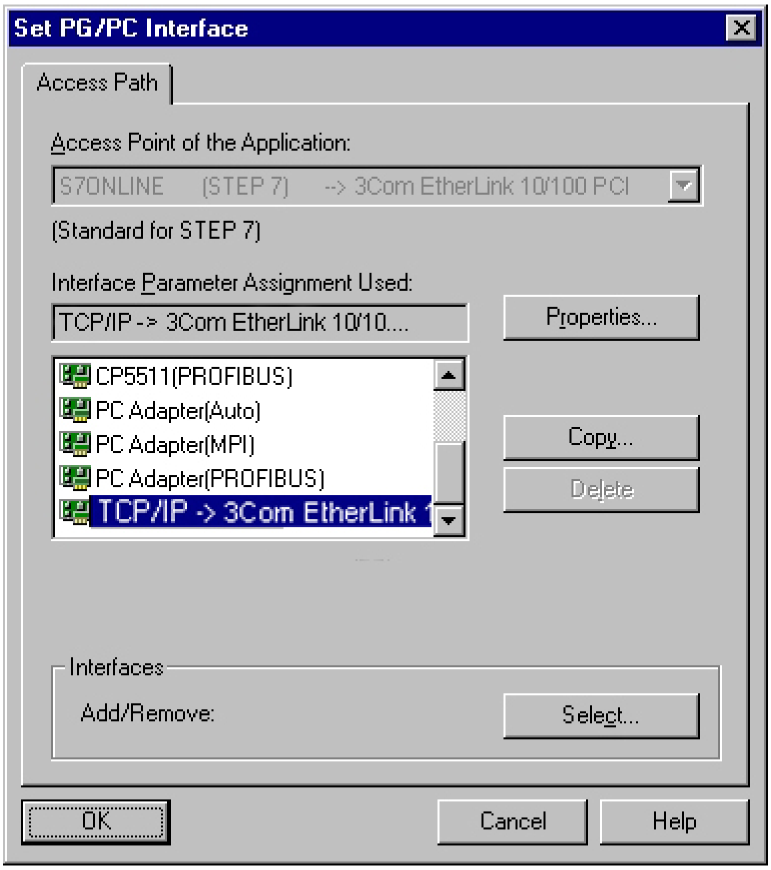
Online with the SIMATIC manager from Siemens

Precondition

The CP portion of the CPU 31xNET is projected under WinNCS and has a valid IP address.

Approach

Start the SIMATIC manager from Siemens, load your project and open the following dialog window via **Options** > *Set PG/PC Interface*:



Set PG/PC Interface

Access Path

Access Point of the Application:

S7ONLINE (STEP 7) --> 3Com EtherLink 10/100 PCI

(Standard for STEP 7)

Interface Parameter Assignment Used:

TCP/IP -> 3Com EtherLink 10/10....

- CP5511(PROFIBUS)
- PC Adapter(Auto)
- PC Adapter(MPI)
- PC Adapter(PROFIBUS)
- TCP/IP -> 3Com EtherLink 1

Interfaces

Add/Remove:

Choose an interface parameterization that refers to your network like shown, for example:

TCP/IP -> 3Com EtherLink 10/100 PCI

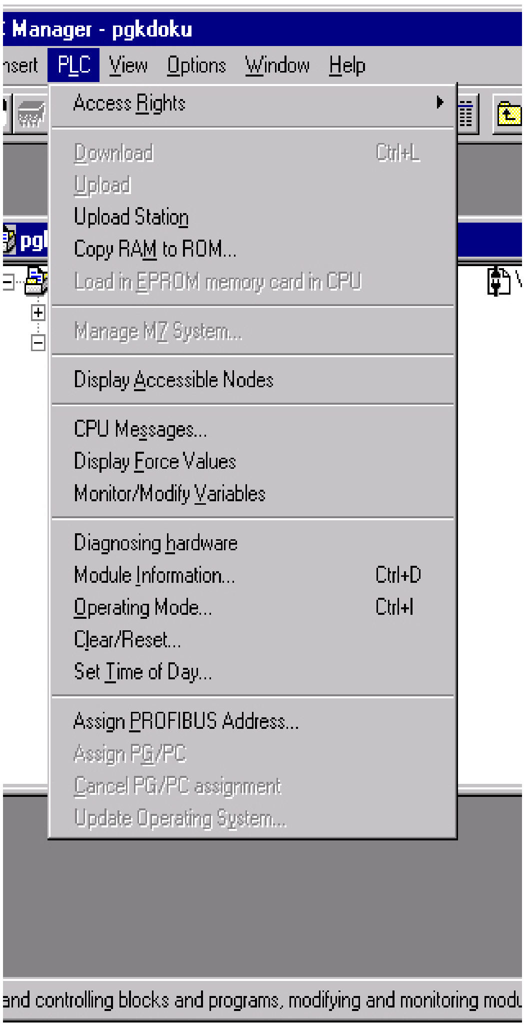
and confirm your entry with [OK].

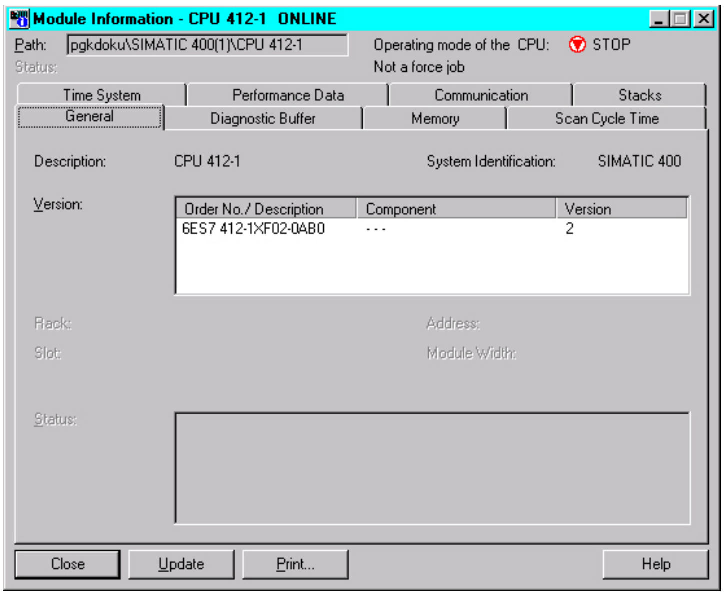
Note!

You may raise the performance when you activate "Fast Acknowledge" at the properties.

Online destination system functions

The VIPA-CP supports all the destination system functions except "Diagnosing Hardware". This function is in preparation.





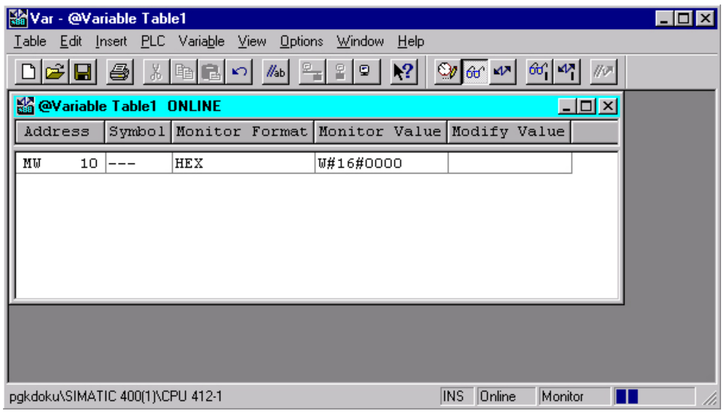
Path: pgkdoku\SIMATIC 400(1)\CPU 412-1
Status:
Operating mode of the CPU: STOP
Not a force job

Description: CPU 412-1
System Identification: SIMATIC 400

Version:

Order No./ Description	Component	Version
6ES7 412-1XF02-0AB0	- - -	2

Rack:
Slot:
Address:
Module Width:
Status:



Address	Symbol	Monitor Format	Monitor Value	Modify Value
MW 10	---	HEX	W#16#0000	

pgkdoku\SIMATIC 400(1)\CPU 412-1 INS Online Monitor

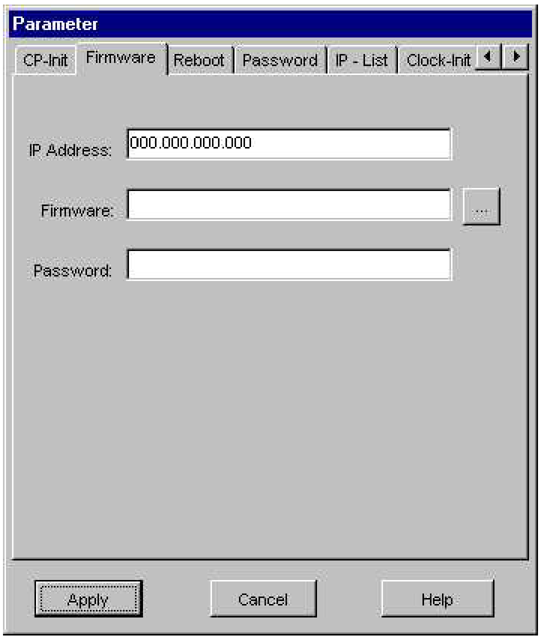
Firmware update and reboot

Register Firmware

Depending on the used firmware on the CP, this register allows you to execute an online firmware update.

As soon as you have given a firmware file and a valid password and click upon [Apply], the firmware is transferred online to the CP defined by the IP address and activated.

This function is password protected. The password is preset in the register Password.



IP Address: 000.000.000.000

Firmware:

Password:

IP address

This field contains the IP address of the destination station for the firmware update. Per default, the IP address of the currently engineered station is entered. The given IP address can be changed.

Firmware

Click on ... and choose a valid firmware file. The firmware may only be transferred with a valid password.

Password

Type the valid password for the CP portion The password in delivery state is: 00000000 (8xzero).

The alteration happens in the register Password.

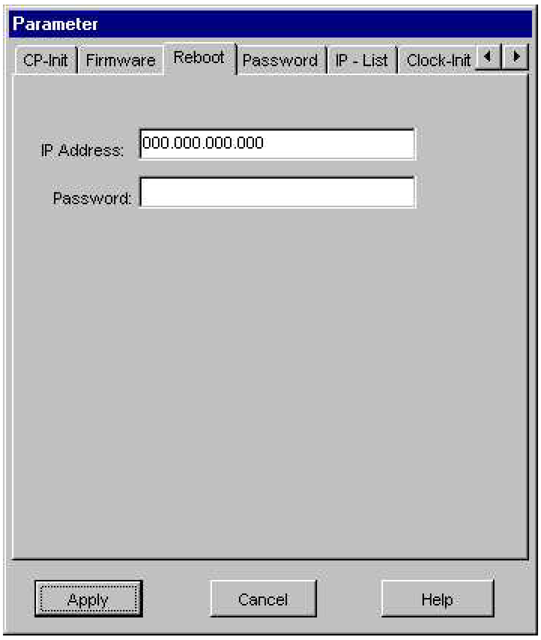
[Apply] firmware update

As soon as you entered IP address, firmware and password, the new firmware is transferred to the CP with [Apply] and the CP is rebooted.

The progress of the reboot and the firmware update is shown via a progress indicator. When the reboot is ready, there is an acknowledgement.

Register Reboot

Depending on the firmware of the CP, you may execute a reboot online. As soon as you entered a destination IP address and a valid password, the CP is rebooted with [Apply]. This function is password protected. The password is preset in the register Password.



IP Address: 000.000.000.000

Password:

IP address

This field contains the IP address of the destination station for the reboot. Per default, the IP address of the currently engineered station is entered. The given IP address can be changed.

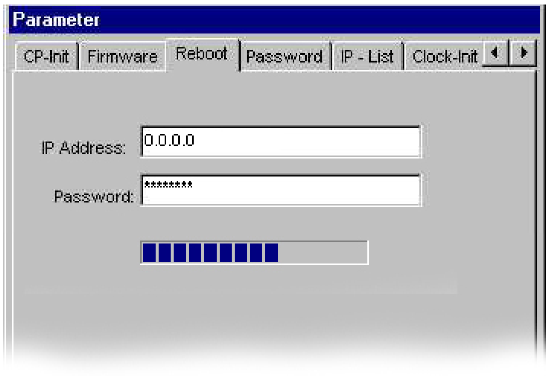
Password

Enter the password valid for the CP. The password in delivery state is to find in the description of your CP.

The alteration of the password happens in the register Password.

Reboot via [Apply]

As soon as you click on [Apply], a reboot is executed at the CP. Reboot is only possible with valid password. The progress of the reboot is shown via a progress indicator. When the reboot is ready, there is an acknowledgement.



IP Address: 0.0.0.0

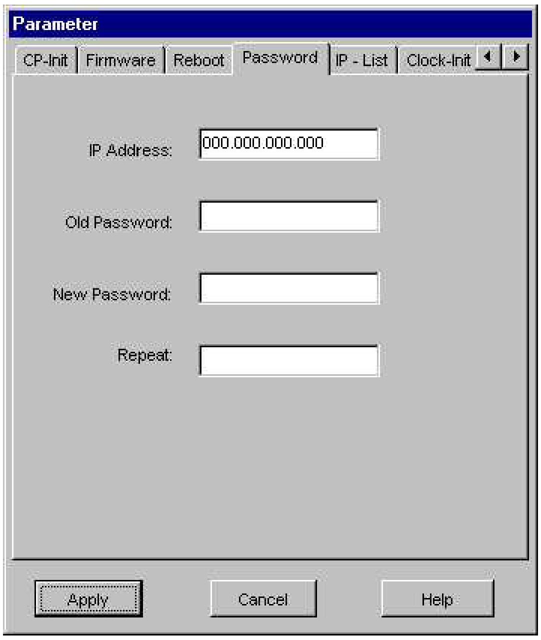
Password: ********

Register Password

Depending on the used firmware you may assign a password for the online firmware update and reboot.

This dialog window allows you to change the password. Enter the recent (old) password and twice the new one.

A click on [Apply] transfers the new password online to the CP defined by the IP address.



IP Address: 000.000.000.000

Old Password:

New Password:

Repeat:

Note!

Please regard that the alteration of the password only happens online. Thus you have to make sure that your PC is able to communicate with the CP via the given IP address.

IP address

This field contains the IP address of the destination station for the reboot. Per default, the IP address of the currently engineered station is entered. The given IP address can be changed.

Old Password

Enter the recent valid password. Every CP is delivered with the password 00000000 (8xzero).

New Password

Type the new password for the CP. The password has to consist of at least 8 characters.

Repeat

For confirmation of the password, type it a second time.

Access control

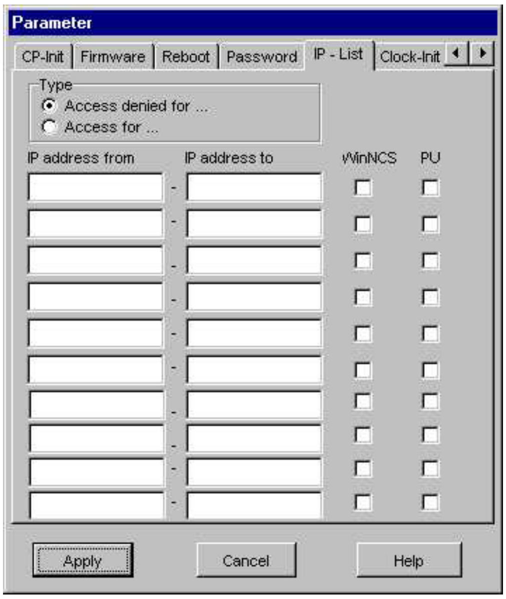
Register IP-List

Starting with version 3.0.8, WinNCS gives you the option to control the access to a CP via IP-List. The access control only influences the transfer of project data via WinNCS res. from PG into CP and CPU.

The normal data transfer between CPs and PGs is not influenced.

By means of the IP-List you may exclude or include IP addresses res. address ranges for the project engineering. The access control is only active when you choose "WinNCS" or "PG" or both.

The IP-List is transferred together with the CP project engineering.



Type

Type defines the type of the access list:

Access denied for ...

IP-List contains IP addresses that are not allowed to access the CP via WinNCS res. PG. The not listed addresses have access rights.

Access for ...

The list contains IP addresses that have access on the CP via WinNCS or PG. The not listed addresses have no access rights.

IP address fields

The IP address fields can be filled with single IP addresses or complete address ranges. A single IP field has to be entered in the left IP field. After [Apply] it is taken over into the right field.

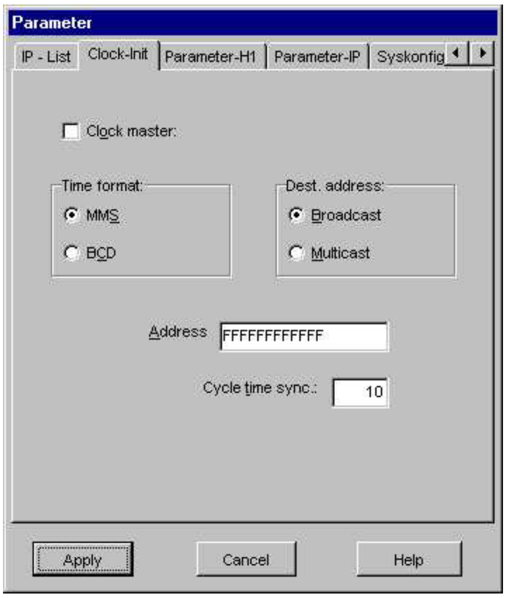
WinNCS, PG

Depending on the chosen list type, you may here release or restrict the access for WinNCS res. PG.

Activate clock master

Register Clock-Init

Every CP is able to work as master if parameterized accordingly.



☐ Clock master:

Time format:
● MMS
○ BCD

Dest. address:
● Broadcast
○ Multicast

Address FFFFFFFFFFFF

Cycle time sync.: 10

Note!

At mixed deployment with Siemens CPs you have to make sure to use the MMS time format because this is the only one supported by Siemens.

Clock master

When the superordinated master (Clock transmitter) fails, the CP parameterized here can get a dynamic master. For this, the CP has to be highest priority and you have to use it as clock master.

For clock master mode you have to enter the destination address (Broadcast or Multicast) and at Multicast the destination Ethernet address. Time format and cycle time must be conform.

As slave, only time format and cycle time must be conform.

Destination address

Broadcast

When choosing broadcast, the address is set to FFFFFFFFFFFFh. It is not changeable.

The clock telegrams are sent to all bus participants.

Multicast

The clock telegrams are only sent to the bus participants that are members of the multicast circle given by the address.

Address

At the setting Broadcast this address is at FFFFFFFFFFFFh. At Multicast you type the address for the multicast circle.

Cycle time for SYNC telegrams

You may choose the cycle time for sending a synchronization telegram in a range of 1...60 seconds. Default are 10s.

The synchronization telegram is transferred in MMS or BCD format.

Attention!

Please regard that time format and cycle time must be conformly parameterized within the system! The last byte of the Ethernet address of every Cp has to be different!

MMS format

For the synchronization of foreign modules you may only use the MMS format. The clock telegram has the following structure:

----	tttt	Byte 0	t: time in ms starting at 0:00 o'clock
tttt	tttt	Byte 1	
tttt	tttt	Byte 2	
tttt	tttt	Byte 3	
dddd	dddd	Byte 4	d: Date in days starting from the 1.1.1984
dddd	dddd	Byte 5	
vkkk	kk--	Byte 6	v: Sign 0=+, 1=-; k: time change in 0,5 hours
----	--e-	Byte 7	e: 0=time of DCF; 1=time of replacement master

BCD format

The BCD format is a simpler alternative. Please regard that the BCD format is not useable with all CPs.

1/10s	1/100s	Byte 0	Parts of seconds
10s	1s	Byte 1	Seconds
10m	1m	Byte 2	Minutes

ab10h	1h	Byte 3	a: Modus: 0=12h, 1=24h;
			b: 0=AM, 1=PM;
			Hours

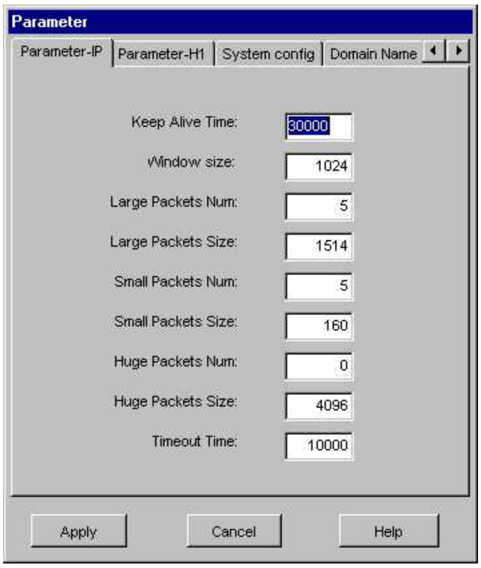
wwww	----	Byte 4	0=Monday ... 6=Sunday
10T	1T	Byte 5	Day of month
10M	1M	Byte 6	Month
10J	1J	Byte 7	Year
v--k	kkkk	Byte 8	v: Sign: 0=+, 1=- k: time change in 0,5 hours
----	1/1000s	Byte 9	1/1000s
ssss	--em	Byte 10	s: Time status: (Click status in ANZW); e: Flag: 0=time of DCF; 1=time of rep. mast. m: Module valid: 0=valid, 1=invalid

Module valid means that the recent master-CP could be parameterized and does not work with default parameters.

IP parameters

Register Parameter-IP

The IP system parameters show the operating parameter of the TCP/IP communication on the network. The values should only be changed for very important reasons.



Keep Alive Time: 30000

Window size: 1024

Large Packets Num: 5

Large Packets Size: 1514

Small Packets Num: 5

Small Packets Size: 160

Huge Packets Num: 0

Huge Packets Size: 4096

Timeout Time: 10000

Keep Alive Time

This parameter shows the dissipation watch time of the TCP connection in ms. The dissipation watch time is the watch time for a connection when no telegram traffic occurs.

Value range: 0 to 999999 ms (Default: 30000 ms)

Window Size

Size of the data window.

Value range: 1 to 65535 (Default: 1024)

Large Packets Num

Number of send and receive buffer.

Value range: 1 to 65535 (Default: 5)

Large Packets Size

This setting is fix on 1514 for Ethernet.

Small Packets Num

Number of acknowledgement telegrams.

Value range: 1 to 65535 (Default: 5)

Small Packets Size

Size of the small packages.

Value range: 1 to 65535 (Default: 160)

Timeout Time

System internal CP parameter; this value shouldn’t be changed.

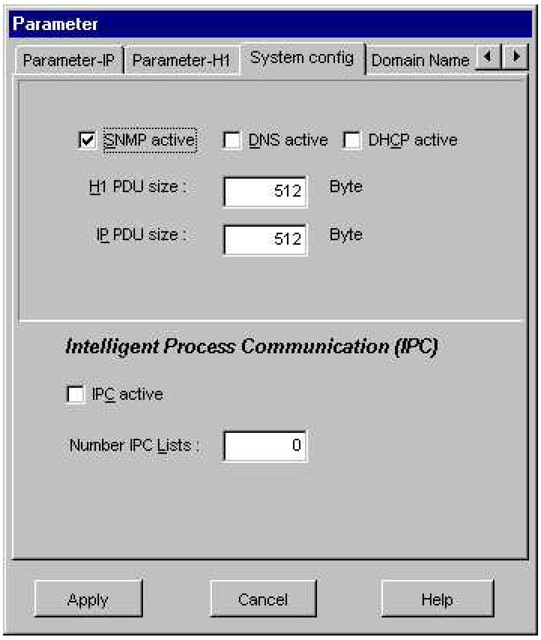
Value range: 1 to 999999 (Default: 10000)

Activate IPC

Register Sys config

With system config, a dialog window for general settings for the system to speed-up the performance opens.

Only the IPK-Part is supported by the CPU 31xNET



☑ SNMP active ☐ DNS active ☐ DHCP active

H1 PDU size : 512 Byte

IP PDU size : 512 Byte

Intelligent Process Communication (IPC)

☐ IPC active

Number IPC Lists : 0

IPC

At the **I**ntelligent **P**rocess **C**ommunikation IPC, also called spontanous operation, data is send to the host for one time on CP request. Afterwards only data that has been changed since the last transfer is transferred. The IPC is used for communication between host and several CPs.

This acyclic transfer method is faster and causes significant less netload than the cyclic transfer.

This allows the data aquisition of several hundred PLC systems via TCP/IP.

IPC is only permissible with FETCH order types and is supported by other visualization systems.

As default IPC is active.

Number IPC Lists

Type the number of IPC orders.

Start-up behavior

Start-up

When the power supply is turned on, the CPU and the CP execute the respective BIOS routine (hardware and driver initialization and memory test).

While the CPU detects which modules have been installed on the back panel bus and loads the application program, the CP starts the page frame administration routine.

After app. 15s the CP waits for the synchronization request from the CPU. In this condition data communication with the PLC is inhibited and it is only enabled after synchronization has taken place.

The boot time of the CPU 31xNET including the CP section amounts app. 18s.

Status after CP start-up

With every status change from STOP to RUN and vice versa back to RUN, the CPU31xNET performs a cold-/warm start. All connections are cleared and reestablished after the CP job boot.

Three different reasons can cause these status change requests:

- Resynchronization of at CP by the SYNCHRON-HTBs of the CPU (warm start) after it has already been synchronized,
- STOP/START function of the configuration tool WinNCS (warm start),
- Reset (warm start).

Note!

When a defective project engineering is transferred to the CP so that is not able to start, the project engineering is deleted and a default setting is used. Here the CP gets its original IP address again.

This allows you to clear the CP independently from the CPU.

System properties of the CPU 31xNET

Note

System properties of a CP are not restrictions or failure reactions. Certain functions can not be provided or are not desired when the overall system is taken into account.

Specials under TCP/IP

- The boot time of the CP section of the CPU 31xNET is app. 18 seconds. The integrated SYNCHRON block caters for this boot time (delay time 30s).
- Under the communication type TCP/IP the joker length (-1, FFFFh) for the ORG format length definition is not permissible, i.e. that the user has to specify a defined length of receiving data.
  If you use a RCF1006 connection instead, you may set the joker length at RECEIVE jobs (-1, FFFFh).
- RECEIVE jobs that are mapped to communication type UDP cannot receive all data messages from a fast cyclic station. Messages that have not been received are discarded.
- The protocol stack TCP/IP has a global buffer pool, where the receive and transmit buffers are located. This is where system collisions can occur if:
  - data for a receive job is not collected. After a period of time a lack of resources will occur and the other connections will terminate after the defined time. A complete communication may started again when the receive buffer of the one connection part is released again (connection termination) res. the data has been requested with the RECEIVE-HTBs.
  - one or more cyclic stations place a load on a CP. When resource bottlenecks are encountered, the CP can also initiate the termination of connections.
  - A station transmits two or more messages and the receiver did not have a chance to accept them, then the reception of the unknown data type would cause collisions in the receiver. However, the CP prevents this. The PLC application requires are defined size for the reception of data and the default or wildcard length is not permitted.
  - VIPA recommends the use of acknowledgment messages on the user level to ensure that data transfers are 100% safe.

Communication to other systems

ORG-Format

The organization format is the abbreviated description of a data source or a data destination in a PLC environment. The available ORG-Formats are listed in the following table.

In the case of READ and WRITE the ORG-Block is optional.

The ERW-identifier is used for the addressing of data blocks. In this case the data block number is entered into this identifier. The start address and quantity provide the address for the memory area and they are stored in HIGH-/LOW- format (Motorola-formatted addresses)

Description	Type	Range
ORG-identifier	BYTE	1..x
ERW-identifier	BYTE	1..255
Start address	HILOWORD	0..y
Quantities	HILOWORD	1..z

The following table contains a list of available ORG-formats. The "length" must not be entered as -1 (FFFFh).

ORG-identifier 01h-04h

CPU-area	DB	MB	EB	AB
ORG-identifier	01h	02h	03h	04h
Description	Source/destination data from/into data Block in Main Memory.	Source/destination data from/into flag area	Source/destination data from/into process image of the inputs (PAE).	Source/destination data from/into process image of the outputs (PAA).
DBNO valid range:	DB, from where the source data is retrieved or to where the destination data is transferred. 1...255	irrelevant	irrelevant	irrelevant
Start address significance valid Range:	DW-No., from where the data is retrieved or where the data is saved. 1...2047	MB-No., from where the data is retrieved or where the data is saved. 0...255	IB-No., from where the data is retrieved or where the data is saved. 0...127	QB-No., from where the data is retrieved or where the data is saved. 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

ORG-identifier 05h-0Ah

CPU-area	PB	ZB	TB	DI
ORG-identifier	05h	06h	07h	0Ah
Description	source/destination data from/into peripheral modules. Input module for source data, output module for destination data.	source/destination data from/into counter cells.	Source/destination data from/into timer cells.	Source/destination data from/into extended data block
DBNO valid range:	irrelevant	irrelevant	irrelevant	DI, from where the data can be retrieved or where it is saved 1...255
Start address Significance valid range:	PB-No., from where the data can be retrieved or where it is saved. 0...127 digital Periph. 128...255 anal. Periph.	ZB-No., from where the data can be retrieved or where it is saved. 0...255	TB-No., from where the data can be retrieved or where it is saved. 0...255	DW-No. from where the data can be retrieved or where it is saved. 0...2047
Quantity Significance valid range:	Length of the source/destination data block in bytes. 1...256	Length of the source/destination data block in words (counter cell = 1 word). 1...256	Length of the source/destination data block in words (counter cell = 1 word). 1...256	Length of the source/destination data block in words 1...2048

Transfer of blocks with numbers >255

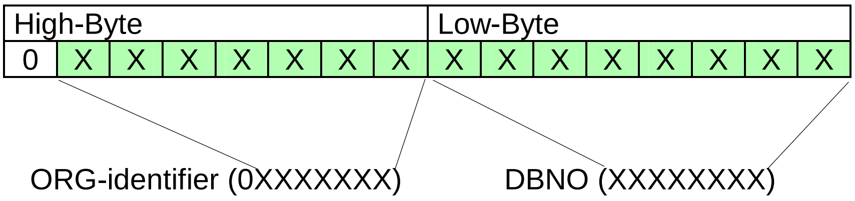
ORG-identifier 81h-FFh

To transfer data blocks of the number range 256 ... 32768 you may use the ORG identifier 81h-FFh.

For the setting of a DB No. >255 needs a length of one word, the DBNO$_{new}$ is assembled from the content of the ORG identifier and the DBNO.

DBNO$_{new}$ is created as word as follows:

DBNO$_{new}$



High-Byte								Low-Byte							
0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

If the highest Bit of the ORG identifier is set, the Low-Byte of DBNO$_{new}$ is defined via DBNO and the High-Byte of DBNO$_{new}$ via ORG identifier, where the highest Bit of the ORG-identifier is eliminated.

The following formula illustrates this:

DBNO$_{new}$ =256 x (ORG-identifier AND 7Fh) + DBNO

Structure of PLC-Header

For every READ and WRITE the CP generates PLC header for request and acknowledgment messages. Normally the length of these headers is 16Bytes and have the following structure:

for WRITE

Request message

System identifier	="S"
	="5"
Length of header	=16d
Ident. OP-code	=01
Length of OP-code	=03
OP-Code	**=03**
ORG-block	=03
Length of ORG-block	=08
ORG identifier	
DBNR	
Start address	H
	L
Length	H
	L
Dummy block	=FFh
Length of dummy bl.	=02
64K data only if error no.=0	

Acknowledgment message

System identifier	="S"
	="5"
Length of header	=16d
Ident. OP-code	=01
Length of OP-code	=03
OP-Code	**=04**
Ack. block	=0Fh
Length of Ack-block	=03
Error No.	=No.
Dummy block	=FFh
Length of dummy bl.	=07
not used	

for READ

Request message

System identifier	="S"
	="5"
Length of header	=16d
Ident. OP-code	=01
Length of OP-code	=03
OP-Code	**=05**
ORG-block	=03
Length of ORG-block	=08
ORG identifier	
DBNR	
Start address	H
	L
Length	H
	L
Dummy block	=FFh
Length dummy block	=02

Acknowledgement message

System identifier	="S"
	="5"
Length of header	=16d
Ident. OP-code	=01
Length of OP-code	=03
OP-Code	**=06**
Ack. block	=0Fh
Length of Ack-block	=03
Error No.	=No.
Dummy block	=FFh
Length Dummy block	=07
not used	
64K data only if error no.=0	

Note!

Please regard that the pointers for the start address of the CPU 31x from VIPA are -not like the S5 from Siemens- pointing to a Byte area. The length assignment is via a word!

Test program for TCP/IP connections

Overview

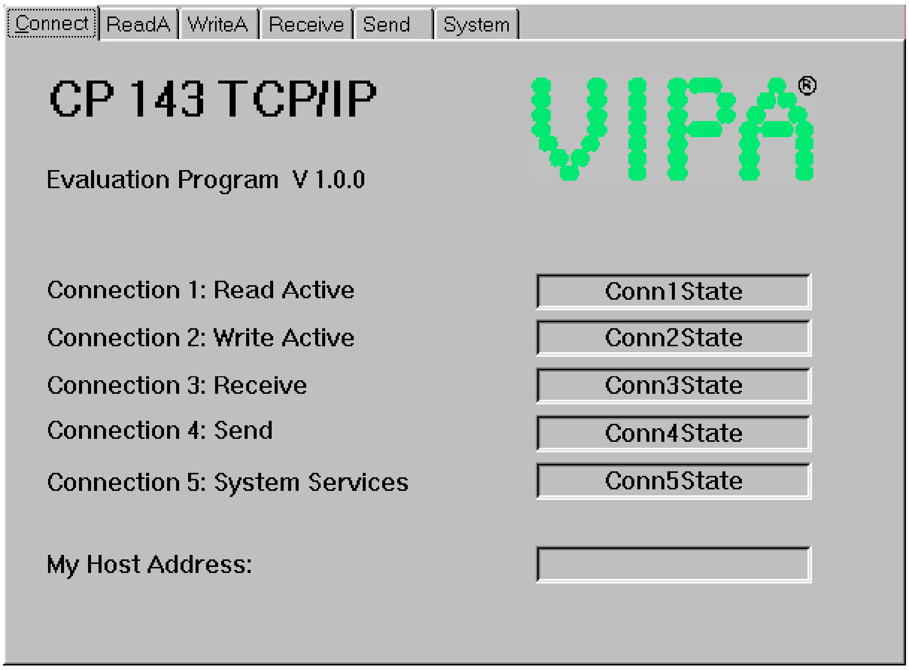
TCPTest.exe is to be found at the service area using the link www.vipa.de/support/software. You can use this test program to create simple TCP/IP connections and analyze them.

TCPTest requires no further installation, is executable at usual operating systems and communicates via Ethernet.

The following section provides a short introduction to the test-program.

For this purpose please start TCPTEST.EXE. The test program is executed and displays the following window:

Initial display



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:

Tab sheets

The menu has the appearance of tab sheets. The respective dialog window can be displayed by left clicking with the mouse.

Tab sheets	
Connect	window containing the status of the connections and the local IP-address.
ReadA	configuration window for READ AKTIV connections (FETCH).
WriteA	configuration window for WRITE AKTIV connections.
Receive	configuration window for RECEIVE connections.
Send	configuration window for SEND connections.
System	control windows for status requests and toggling between RUN/STOP of the CP.

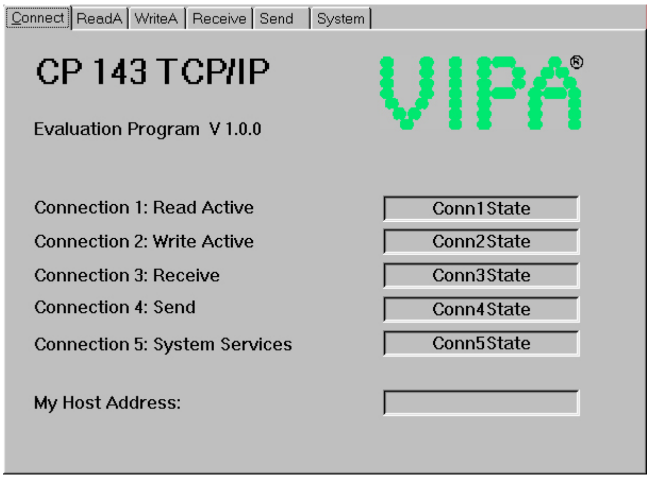
Context menu (right mouse key)

You can activate a context menu in each tab sheet. This is activated by means of the right mouse key or button.

You can always access the context menu by clicking the right mouse key. This menu offers the following selection:

Save All	save all parameters.
Save Conn1 to *Save Conn5*	saves the respective connection.
Save Win Pos	saves the current window position.
Show Hints	When you place the cursor on an input field or on a button, a hint is displayed if you have selected "Show Hints".

Connect tab (Status)



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

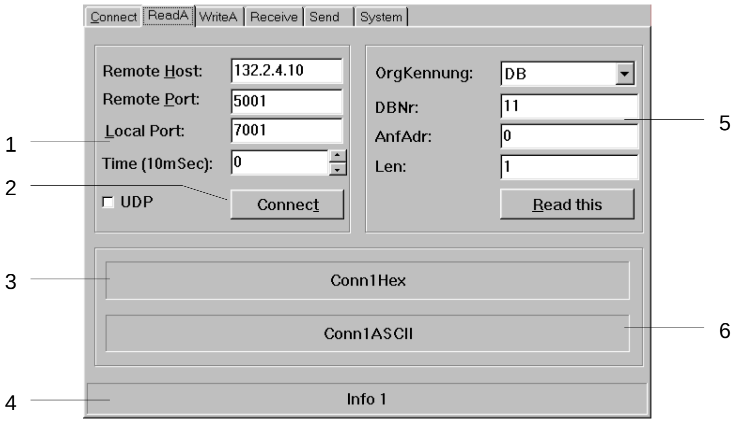
Connection 5: System Services Conn5State

My Host Address:

This window displays the status of all the connections that can be configured in this program. Here you can recognize in one screen, which connections are stable and which are unstable. When a status changes in a register the change is displayed in this window.

For reference, your own IP address is also displayed in the window.

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

[1] port data
[2] establish a connection
[3] hexadecimal number
[4] information window for the status of the connection
[5] source data
[6] ASCII formatted display of the data received

Here you can configure an active read connection.

In addition to the data required to establish the connection you must also specify the source from where the data should be read.

Input fields

Remote Host IP-address of the station from which you wish to read data.

Remote Port port address of the remote station.

Local Port port address of your own (local) station. To simplify matters you can specify the same port address for remote and local.

Time (10mSec) definable interval for cyclic read operations.

OrgKennung type of the source block.

DBNr number of the source block.

AnfAdr start address of the source block.

Len word length of the source block.

Tick-box

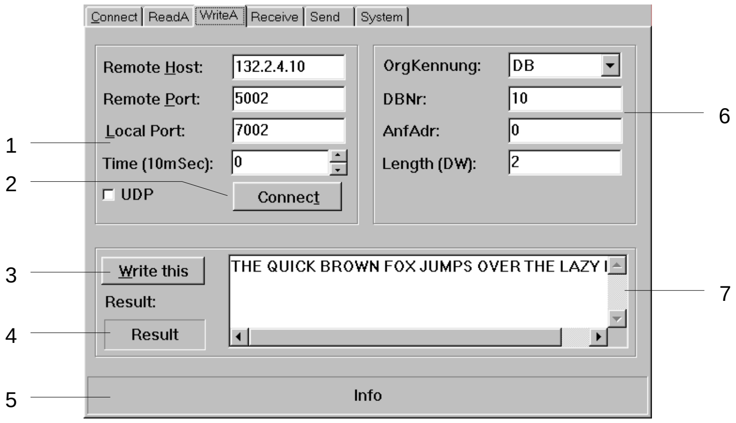
UDP This tick mark selects unsecured communications. No virtual connections are used by unsecured communication links. In this manner you can only display UDP messages.

Buttons

Connect the connection is established and prepared for the read operation.

Read this the data requested is read via this connection.

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: Result
THE QUICK BROWN FOX JUMPS OVER THE LAZY I

[1] port data
[2] establish a connection
[3] transfer the data via the connection
[4] result-code of the write job

[5] information window for the status of the connection
[6] source data
[7] ASCII-text that must be transferred to the CP

This is where you activate an active write connection.

In the same way as for the READ active command you declare the destination block where the data must be transferred in addition to the data required for establishing the connection.

Input fields

Remote Host IP-address of the station where to which you wish to write the data.

Remote Port port address of the remote station.

Local Port port address of your own (local) station. To simplify matters you can specify the same port address for remote and local.

Time (10mSec) definable interval for cyclic write operations. The minimum timer value for cyclic writes is 5.

OrgKennung type of destination block.

DBNr number of the destination block.

AnfAdr start address of destination block.

Len word length of destination block.

Tick-box

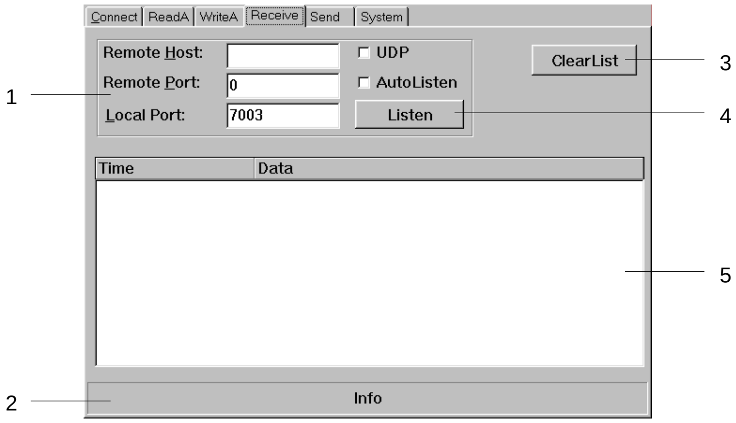
UDP this tick mark selects unsecured communications. No virtual connections are used by unsecured communication links. In this manner you can only display UDP messages.

Buttons

Connect the connection is established and prepared for the write operation.

Write this data entered into the ASCII field is transferred to the CP via the connection that was established by means of *Connect*.

Receive tab



Remote Host:
Remote Port: 0
Local Port: 7003

[1] port data
[2] connection status information bar
[3] clear received list
[4] list the messages
[5] list of received messages

In this dialog window you can configure the reception of messages from a specific host.

Input fields

Remote Host IP-address of the station where the data must be saved.

Remote Port port address of the remote station.

Local Port port address of own (local) station. To simplify matters you can specify the same port address for remote and local.

Tick-box

UDP this tick mark selects unsecured communications. No virtual connections are used by unsecured communication links. In this manner you can only display UDP messages.

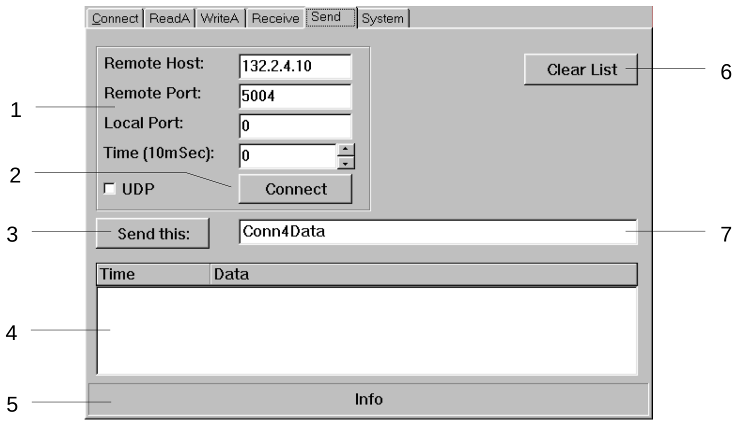
AutoListen If you select "AutoListen" the program goes to receive mode. Every message received from the remote CP is displayed in the list. Interruptions of the connection are detected and displayed, however, the program remains ready to receive data. As soon as the connection is reestablished messages will again be listed.

Buttons

Listen any received messages are entered into the list. The listing is stopped when you click the "STOP" button or the connection is interrupted. You can also stop the listing by entering a new set of connection parameters.

ClearList clears the received list, new entries will appear at the top of the list.

Send tab



Remote Host: 132.2.4.10
Remote Port: 5004
Local Port: 0
Time (10mSec): 0
□ UDP
Send this: Conn4Data

Time	Data

[1] port data
[2] establish a connection
[3] transfer data via the connection
[4] list of transmitted messages
[5] information bar for the connection status
[6] clear the list of messages
[7] ASCII-text that must be transferred to the CP

You can use this dialog window to send a message to a specific host.

Input fields

Remote Host IP-address of the station where the data must be saved.

Remote Port port address of the remote station.

Local Port port address of own (local) station. To simplify matters you can specify the same port address for remote and local.

Time (10mSec) definable interval for cyclic write operations. The minimum timer value for cyclic writes is 5.

Tick-box

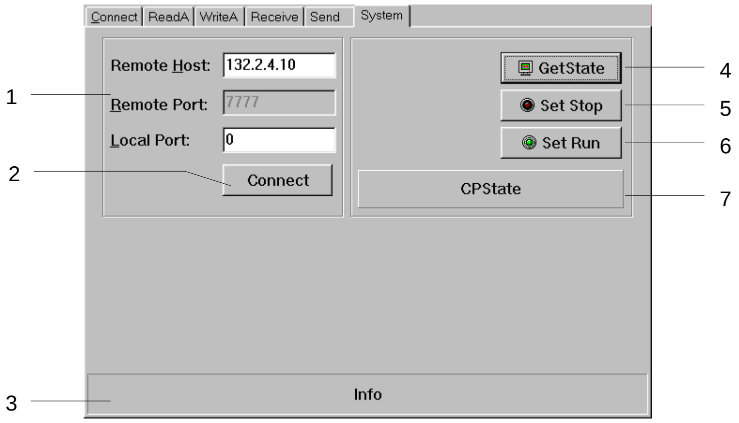
UDP this tick mark selects unsecured communications. No virtual connections are used by unsecured communication links. In this manner you can only display UDP messages.

Buttons

Connect the connection is established and prepared for the write operation.

Send this data entered into the ASCII field is transferred to the CP via the connection that was established by means of *Connect*.

System tab



[1] port data
[2] establish a connection
[3] information bar for the connection status
[4] CP status request
[5] CP in STOP
[6] CP in RUN
[7] status monitor, requested with GetState

This dialog window gives you information about the specified host-CP.

Input fields

Remote Host IP address of the station, where the data is stored to.

Remote Port Connection address of the foreign station.

Local Port Connection address of the own station. To simplify the process, you may use the same address for remote and local connections.

Buttons

Connect The connection is established and prepared for communication.

GetState Via the connection, established by means of *Connect*, the status of the CP is transferred and monitored in the status window. Monitored may be:

- Hardware-Stop (Run/Stop-switch at the CP is in stop position)
  The CP must not be remoted via the test program.
- Hardware-Run (Run/Stop-switch at the CP is in run position)
  The CP may be remoted via the test program.
- Software-Stop (Run/Stop-switch at the CP has to be in run position)
  The CP has been set to stop by means of *SetStop*.
- Software-Run (Run/Stop-switch at the CP has to be in run position)
  The CP has been set to start by means of *SetRun*.

SetStop The CP is set to stop. This function is only available, if the CP-switch is in run position.

SetRun The CP is set to run. This function is only available, if the CP-switch is in run position.

Sample communication TCPTest - CPU 31xNET

Outline

This chapter provides an introduction to the communication between a CPU 31xNET and the software TCPTest from VIPA. Here data is exchanged via 2 connections. The sample shows the principles of project engineering and will help you to use WinNCS.

Preconditions

The minimum technical equipment required for the examples is as follows:

Hardware

- 1 CPU 31xNET from VIPA
- 1 PC or PG with Twisted Pair Ethernet connection

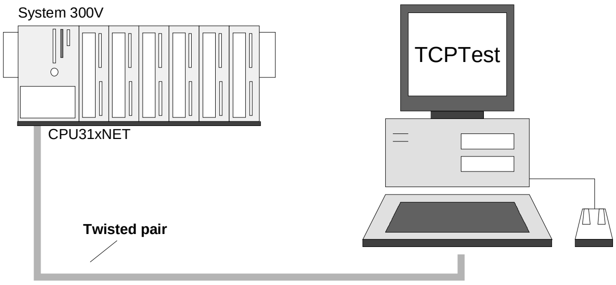
Communication line

- 1 bus cable (crossed at point-to-point connection)

Software packages

- Parameterization software WinNCS and TCPTest from VIPA
- SIMATIC manager from Siemens V. 5.1 or V.5.2 with SP1
- SIMATIC NET

Structure



Problem

The following communication task is the basis of the sample project:

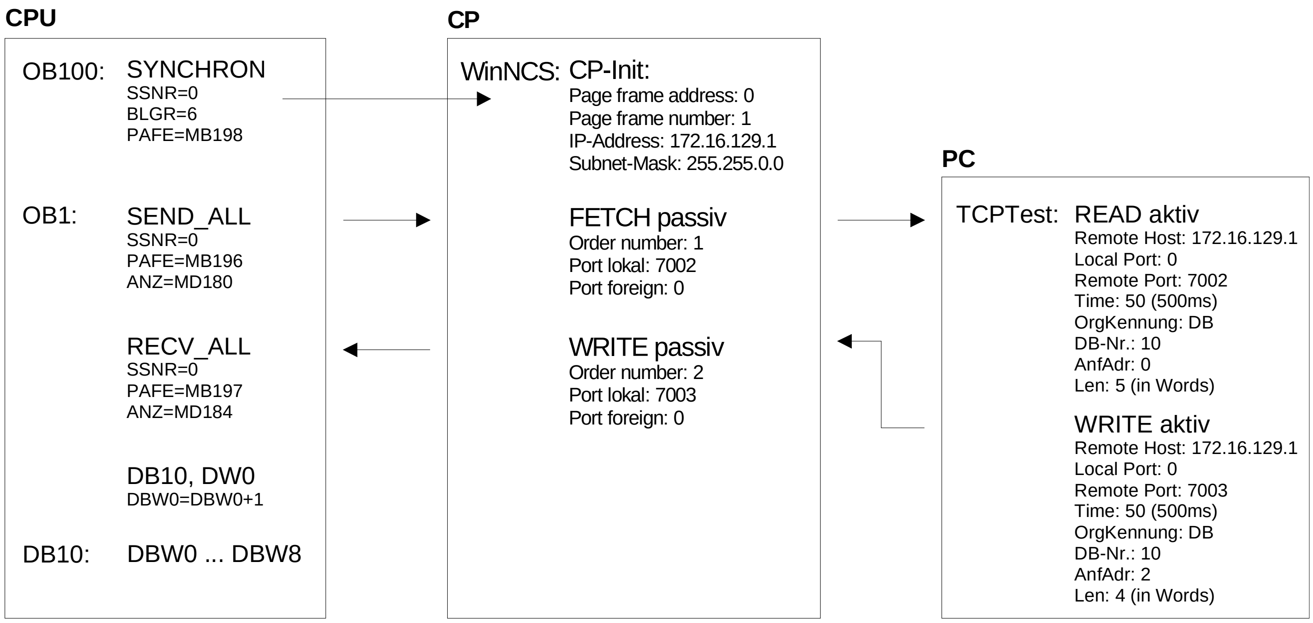
DW0 of DB10 is incremented in OB 1 of the CPU. Deploying TCPTest, the DB10 of the CPU (DBW2 ... DBW8) shall be given a value via WRITE_ACTIVE. For control purposes we want to read DBW0 ... DBW8 from the CPU via READ_ACTIVE.

The CP gets the sample IP address 172.16.129.1 and communicates via the ports: read port: 7002, write port: 7003

At PC side, the IP address and ports are not specified.

Approach

- *Configure CP*
  READ_PASSIV and FETCH_PASSIV connection to the PC in WinNCS
- *Configure CPU*
  SYNCHRON in OB 100 and SEND_ALL and RECV_ALL in OB1, Data block DB10 with DW0 ... DW4.
  To create dynamic data, the content of DW0 has to be incremented cyclically
- *Configure PC*
  WRITE_AKTIV and READ_AKTIV connection to the CP in TCPTest



CPU

OB100: SYNCHRON
SSNR=0
BLGR=6
PAFE=MB198

OB1: SEND_ALL
SSNR=0
PAFE=MB196
ANZ=MD180

RECV_ALL
SSNR=0
PAFE=MB197
ANZ=MD184

DB10, DW0
DBW0=DBW0+1

DB10: DBW0 ... DBW8

CP

WinNCS: CP-Init:
Page frame address: 0
Page frame number: 1
IP-Address: 172.16.129.1
Subnet-Mask: 255.255.0.0

FETCH passiv
Order number: 1
Port lokal: 7002
Port foreign: 0

WRITE passiv
Order number: 2
Port lokal: 7003
Port foreign: 0

PC

TCPTest: READ aktiv
Remote Host: 172.16.129.1
Local Port: 0
Remote Port: 7002
Time: 50 (500ms)
OrgKennung: DB
DB-Nr.: 10
AnfAdr: 0
Len: 5 (in Words)

WRITE aktiv
Remote Host: 172.16.129.1
Local Port: 0
Remote Port: 7003
Time: 50 (500ms)
OrgKennung: DB
DB-Nr.: 10
AnfAdr: 2
Len: 4 (in Words)

At this point the purpose and the required settings have been outlined. The programs provide additional details of the configuration of the handler blocks. A detailed description of a suitable configuration of the CPs under control of TCP/IP is also included.

Configuration under WinNCS

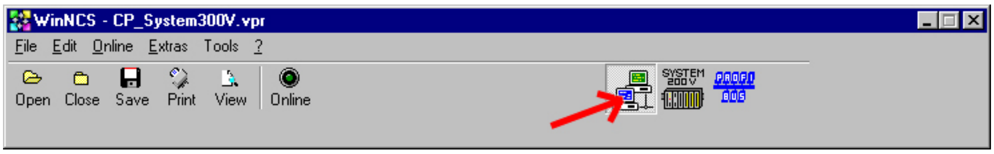
The project engineering of the CP happens via WinNCS. It is divided into 3 parts:

- basic CP configuration
- Configuration of the connection blocks
- transfer of configuration data into the CP

The following text describes the steps of the WinNCS project engineering.

Functionality Ethernet

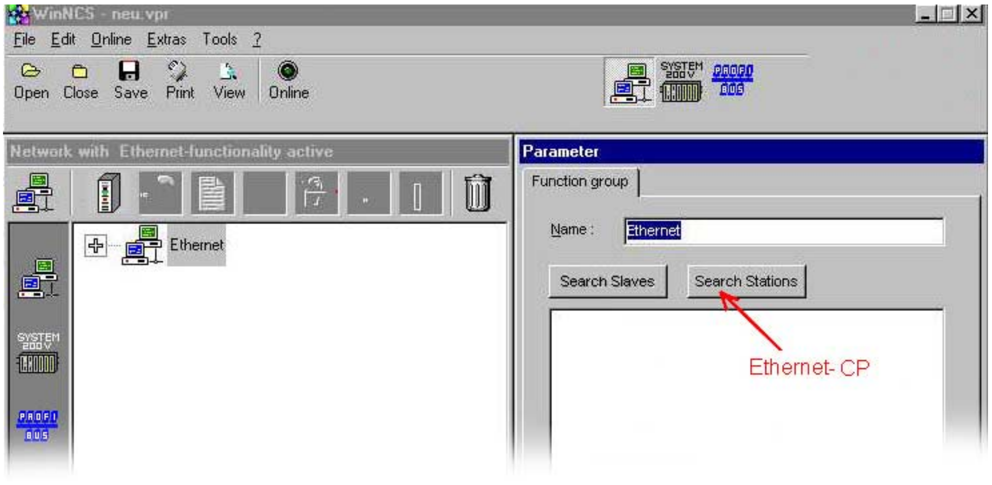
Start WinNCS.


WinNCS - CP_System300V.vpr
File Edit Online Extras Tools ?
Open Close Save Print View Online
SYSTEM 200 V
PROFI BUS

Activate the functionality "Ethernet" via .

Create a new project with the function group "Ethernet" via **Project** > *Project set-up/open*.

Search CP



Click in "Parameter"-window on [Search stations] → The available VIPA CPs are listed by their IP address.

If your target CP is inside your IP circle, the CP can online be projected. Otherwise you have to change the IP address by using [Change IP]. After that click to [Search stations]. Please regard! If you change the IP Address of the CP by using [Change IP], the configuration inside the CP is cleared.

Via double click on the wanted station, the according project is imported and monitored in the "network" window for further parameterization.

Basic CP configuration

Insert the following values and click on [Apply]:

Station name: CPU 31xNET (free chooseable)

Page frame basic address: 0 (fix)

Page frame number: 1 (fix)

Station address: 000000000000 (fix)

IP address (here 172.16.129.1) and *Subnet mask* (here 255.255.0.0): The IP address and the subnet mask is available at your system operator. The rest of the entries remain unchanged.

Configure connection blocks

🌐

FETCH-passiv connection (Read)

To configure the FETCH connection, you click on 🌐. Insert the following parameters and click on [Apply]:

Connection name: PC Fetch

Page frame offset: 0 (fix) *Order type*: FETCH passive

Order no: 1 *Order model*: Single order

Priority: 2

Port local: 7002 *Port external*: 7013

By setting an IP address for the foreign station you fix that the CP may only contact this IP address. For a test environment you should leave this field empty.

WRITE-passive connection

The project engineering of the WRITE connection is identical. Click on 🌐, insert the following parameters and click on [Apply]:

Connection name: Receive from PC

Page frame offset: 0 (fix) *Order type*: Write passive

Order no: 2 *Order model*: Single order

Priority: 2

Port local: 7003 *Port external*: 7012

Transfer the WinNCS project

To transfer your WinNCS project you have the following two possibilities:

- Transfer directly via Ethernet
- Transfer via MMC

Transfer directly via Ethernet

Online

The transfer via Ethernet is proceeded in the online function as follows when this is the first project:

- Connect the CPU with the Ethernet.
- Turn on the power supply of your CPU. After app. 30s, the CP has been booted.
- Activate the online functions via Online.
- Set "IP-Protocol" under Init and give the IP address 172.16.192.11. Every CPU 31xNET from VIPA is delivered with this IP address.
- Establish a connection via Conn.
- Set your CP in software STOP via Stop.
- Choose the station "CPU 31xNET" *network* window.
- With CP PC the transfer into the CP starts. If you get a NCS file request, you missed the last step! Select it now and start the transfer again.
- As soon as the transfer is finished, you reboot the CP with Start. This takes about 20s.

Now the CP is on the Network, with the assigned IP address.

Transfer as *webcp.ncs* via MMC

WinNCS allows you to export your project as *.NCS file. Mark the station and select Export via the right mouse button. Assign the name webcp.ncs to the file and transfer it to the MMC.

In the following description, the CP project is recognized due to the unambiguous file name and transferred into the CP:

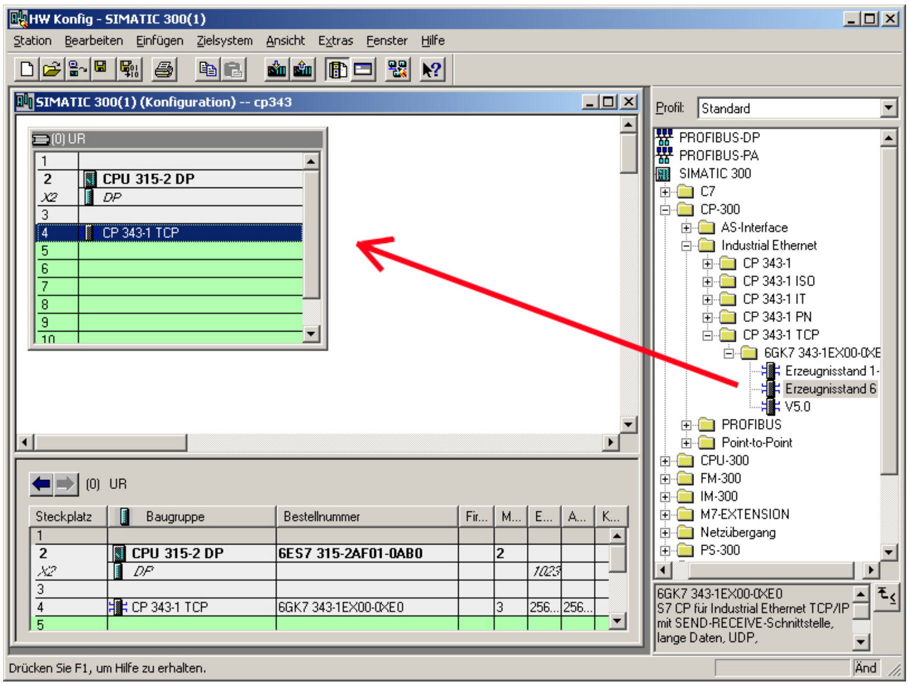
- Set the RUN-STOP switch of your CPU in position STOP.
- Turn off the power supply.
- Plug the MMC with the CP file into the CPU. Please take care of the correct plug-in direction.
- Turn on the power supply.
- After a short start-up time, the alternately blinking of the LEDs SF and FRCE shows that the CP file has been recognized on the MMC.
- You start the transfer of the file by tipping the RUN/STOP switch shortly to MR within 10s.
- During data transfer, the LEDs SF, FRCE and MMC are blinking alternately.
- The update has been completed successful when all CPU-LEDs are on. At fast blinking, an error has occurred.
- Turn off the power supply, dismantle the MMC and switch on the power supply again. CPU and CP are rebooting. Now the CP is at the network with the configured IP address.

Control the WinNCS project

- Activate the online functions in WinNCS via Online.
- Select "IP protocol" under Init and type the new IP address 172.16.129.1.
- Establish a connection with Conn. Now you communicate with the IP address given under CP-Init.
- The CP has to be in RUN. Proof this with Status. If the CP is in *Idle*-Mode, the synchronization with the CPU has failed. Please control the SYNCHRON block in OB 100.
- For control you may monitor the summary status of the TCP connections via TcpIP.

CPU and CP project engineering in the SIMATIC manager from Siemens

- Start the SIMATIC manager from Siemens and create a new project.
- Insert a new System 300 station via **Insert** > *Station* > *SIMATIC 300 Station*.
- Activate the "SIMATIC 300" station and open the hardware configurator by clicking on "Hardware".
- Configure a rack (Simatic300 > Rack-300 > Rail).
- For all CPUs 31x from VIPA are configured as CPU 315-2DP, you select the CPU 315-2DP with the order no. 6ES7 315-2AF03-0AB0 in the hardware catalog.
  This is to find under Simatic300 > CPU-300 > CPU 315-2 DP.
- Insert the CP343-1 TCP at the plug-in location 4 (Simatic300 > CP-300 > Industrial Ethernet > CP 343-1 TCP).



(0) UR	
1	
2	CPU 315-2 DP
X2	DP
3	
4	CP 343-1 TCP
5	
6	
7	
8	
9	
10	

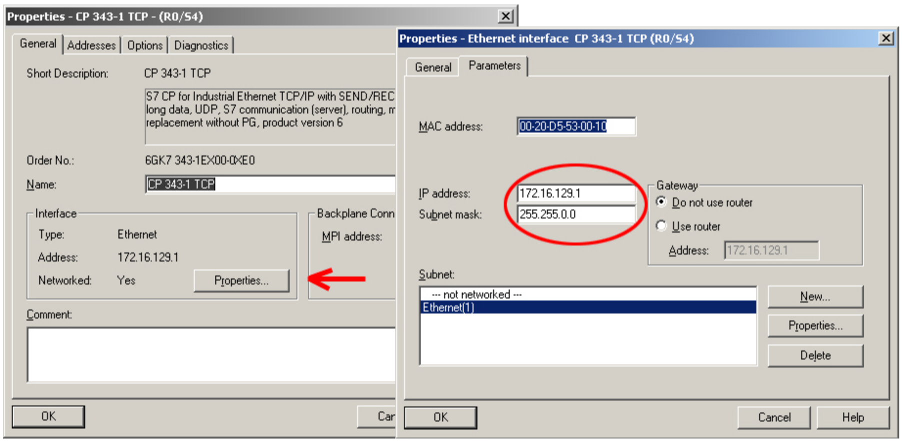
(0) UR

Steckplatz	Baugruppe	Bestellnummer	Fir...	M...	E...	A...	K...
1							
2	CPU 315-2 DP	6ES7 315-2AF01-0AB0		2			
X2	DP				1023		
3							
4	CP 343-1 TCP	6GK7 343-1EX00-0XE0		3	256...	256...	
5							

6GK7 343-1EX00-0XE0
S7 CP für Industrial Ethernet TCP/IP mit SEND-RECEIVE-Schnittstelle, lange Daten, UDP,

Drücken Sie F1, um Hilfe zu erhalten.

- Via a double click on the CP 343-1 TCP you open the properties window and choose the valid IP address.


Properties - CP 343-1 TCP - (R0/S4)
General Addresses Options Diagnostics
Properties...
OK
Car
Properties - Ethernet interface CP 343-1 TCP (R0/S4)
General Parameters
New...
Properties...
Delete
OK
Cancel
Help

Short Description: CP 343-1 TCP

S7 CP for Industrial Ethernet TCP/IP with SEND/REC
long data, UDP, S7 communication (server), routing, m
replacement without PG, product version 6

Order No.: 6GK7 343-1EX00-0XE0

Name: CP 343-1 TCP

Interface
Type: Ethernet
Address: 172.16.129.1
Networked: Yes

Backplane Conn
MPI address:

Comment:

MAC address: 00-20-D5-53-00-10

IP address: 172.16.129.1
Subnet mask: 255.255.0.0

Gateway
● Do not use router
○ Use router
Address: 172.16.129.1

Subnet:
--- not networked ---
Ethernet(1)

- If needed parameterize the CPU res. the modules. The parameter window is opened at double click on the depending module.
- Save your project.

CPU program

Synchronization of the CP interface

The used interface of the CP has to be synchronized in the CPU by means of the start-up OB 100 and the handling block SYNCHRON.

The block SYNCHRON proofs if the synchronization has been executed error free. In case of an error there is an error output in MB198.

OB 100:

```
CALL "Synchron"
  SSNR:=0
  BLGR:=6
  PAFE:=MB198
```

Data transfer between CPU and CP

Via the cycle OB 1, the sending and reception of data between CP and CPU is controlled:

OB 1:

```
CALL "Recv_ALL" //SFC 237
  SSNR:=0
  PAFE:=MB197
  ANZW:=MD184

CALL "Send_All"       //SFC 236
  SSNR:=0
  PAFE:=MB196
  ANZW:=MD180

AUF     DB 10         // Content of DW0 is cyclically
L       DBW 0         // incremented
+       1
T       DBW 0
```

Create the DB10 with the data words DBW0 ... DBW8:

DBW0 of type Word for the counter value

DBW2 ... DBW8 of type Byte for the data area

Transfer PLC project

The data transfer happens via MPI. If your programming device has no MPI interface you may establish a serial point-to-point connection from the PC to the MPI via the VIPA "Green Cable".

The "Green Cable" has the order no. VIPA 950-0KB00 and may only be deployed with VIPA CPUs of the Systems 100V, 200V and 300V.

- Connect your PG with the CPU.
- With **PLC** > *Download* in your projecting tool you transfer your project into the CPU.

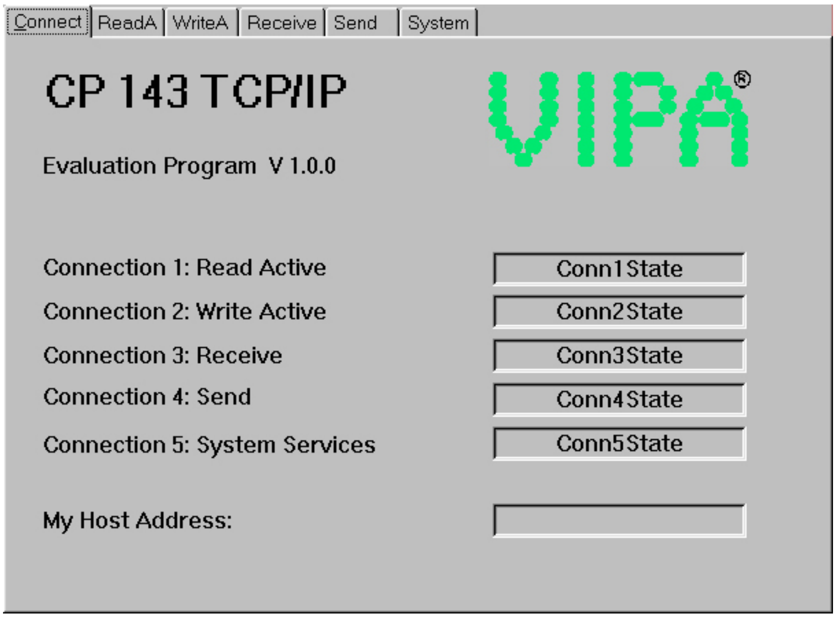
Now the project engineering of the CPU and the CP is finished. The following page shows the deployment of TCPTest:

Deployment of TCPTest

TCPTest.exe is to be found at the service area using the link www.vipa.de/support/software. This test program that runs on all common Windows operating systems allows you to establish and analyze simple TCP/IP connections via Ethernet.

Here follows a short description of the deployment of this test program together with the already configured CPU 31xNET.

Start TCPTEST.EXE. The test program starts with the following window:



The menu has the form of register tabs. By clicking with the left mouse button, the chosen dialog window comes into view.

Preconditions

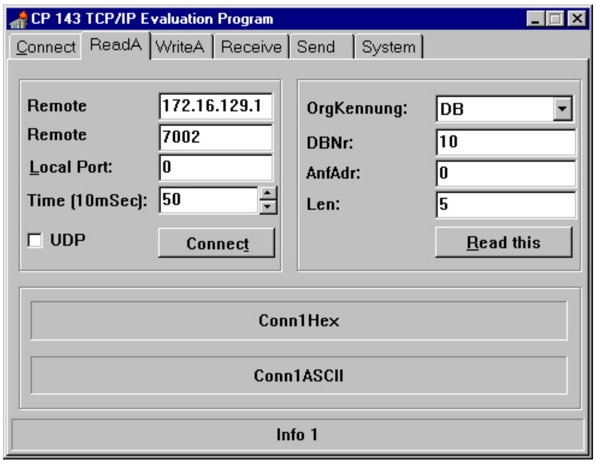
The following preconditions have to be fulfilled:

- PC and CPU 31xNET are connected to Ethernet.
- PC and CPU 31xNET need to be inside the same IP address circle (see note for „Transfer "WinNCS project").
- The PLC program and the CP project engineering are in the CPU 31xNET.
- The CPU has been booted and is in RUN.
- The CP has been configured and is in RUN.

Read data

After getting the preconditions you may establish a communication to the CP.

For this, you click on the *ReadA* register, type the following values and click on [Connect]:



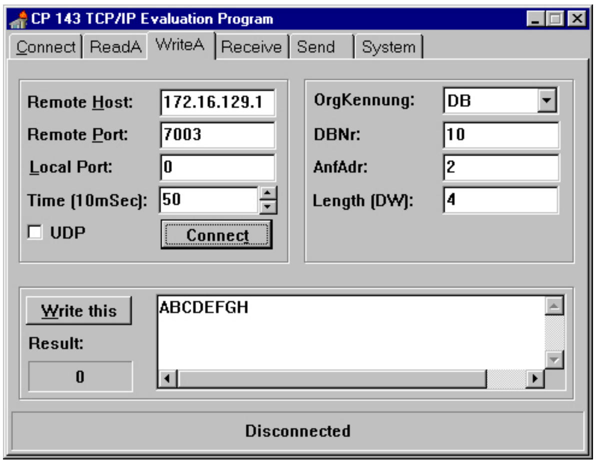
Remote: 172.16.129.1
Remote: 7002
Local Port: 0
Time (10mSec): 50
□ UDP
OrgKennung: DB
DBNr: 10
AnfAdr: 0
Len: 5

A connection to your CP is established and shown under "Info 1". As soon as you click on [Read this], the specified content of DB10 is written all 500ms into "Conn1Hex" and "Conn1ASCII" in Hex res. ASCII format.

When your entries are valid, you see the decrementation of DW0. Leave the connection running.

Write data

For a write access to the CPU 31xNET click on the *WriteA* register. Type the following values:



Remote Host: 172.16.129.1
Remote Port: 7003
Local Port: 0
Time (10mSec): 50
□ UDP
OrgKennung: DB
DBNr: 10
AnfAdr: 2
Length (DW): 4
ABCDEFGH
Result: 0
Disconnected

[Write this] sends the data to the CP every 500ms. You may alter the data during runtime and monitor this in the *ReadA* register.

Now you have realized a small communication example. The following example shows the communication between 2 CPs.

Sample communication CPU 31xNET - CPU 31xNET

Outline

This chapter provides an introduction to use the TCP/IP bus system for the System 300V. The object of this chapter is to create a small communication system between two VIPA CPUs 31xNET that provides a simple approach to the control of the communication processes.

Preconditions

You should have executed the "Sample communication TCPTest - CPU 31xNET" to be comfort with the basic handling of WinNCS.

Knowledge of the CP handling blocks is required. CP handling blocks are standard function blocks. These provide the options required to utilize the communication functions in the programs of the programmable logic controllers.

The minimum technical equipment required for the examples is as follows:

Hardware

- 2 CPUs 31xNET from VIPA
- 1 PC or PG with Twisted Pair Ethernet connection

Communication line

- 2 bus cables

Software package

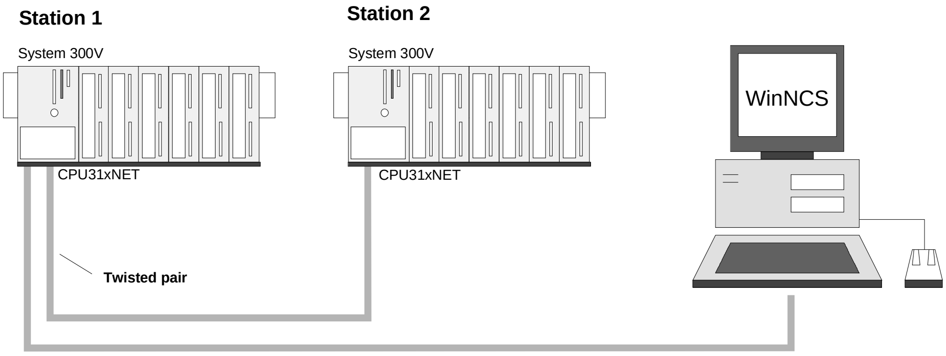
- Configuration software WinNCS from VIPA
- Programming package WinPLC7 from VIPA res. SIMATIC-Manager from Siemens for CPU 31xNET

The implementation of the example requires that the two programmable logic controllers be programmed as well as the configuration of the communication processors by means of WinNCS.

Note!

The complete example is saved on the attached disk. You may transfer the PLC program directly to both CPUs.

Structure



Station tasks

The example for the application is based upon a communication task that is described in detail in the following passage:

Both of the CPUs contain the same PLC program, only the configuration of the CPs have to be adjusted.

Both stations are sending and receiving 16 data words per second.

- Data block DB11 transfers the data words DW 0 to DW 15 at an interval of 1s. data word DW 0 in DB 11 is used as message counter. It is only incremented if the preceding transmit command was processed correctly (completed without error). The remaining data words (DW 1 to DW 15) can be used for the transfer of user data.
- The receiving station stores the data in DB12 (DW 1 to DW 15).
- SEND is configured with job number A-No. = 1 and with a page frame offset SSNR = 0.
- RECEIVE is configured with job number A-No. = 11 and a page frame offset SSNR = 0.
- The source and destination parameters must be configured directly.

At this point the purpose and the required settings have been outlined. The programs provide additional details of the configuration of the handler blocks. A detailed description follows.

Configuration at WinNCS

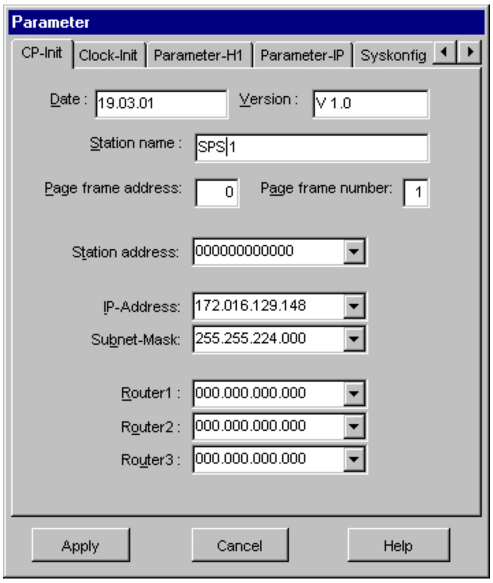
The two CPs are configured by means of WinNCS. Start WinNCS and create a project containing the function group "Ethernet". The procedure is the same for both stations. It differs only in the parameters that must be defined and is divided into the following 3 parts:

- basic CP configuration,
- configuration of connection blocks,
- transfer of configuration data into the CP.

Basic CP configuration

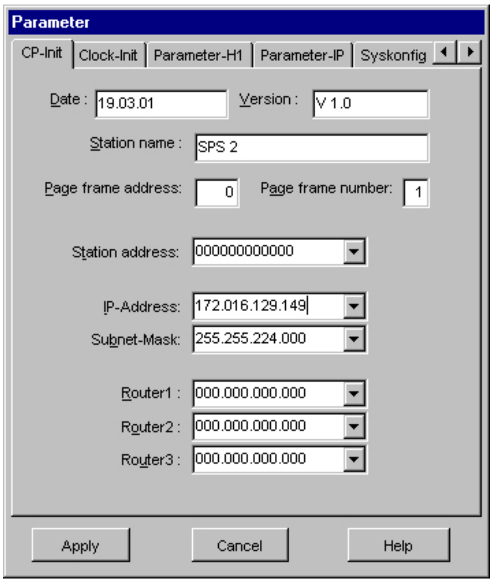
Insert two stations and select the following settings:

Station 1



Date : 19.03.01
Version : V 1.0
Station name : SPS 1
Page frame address: 0
Page frame number: 1
Station address: 000000000000
IP-Address: 172.016.129.148
Subnet-Mask: 255.255.224.000
Router1 : 000.000.000.000
Router2 : 000.000.000.000
Router3 : 000.000.000.000

Station 2



Date : 19.03.01
Version : V 1.0
Station name : SPS 2
Page frame address: 0
Page frame number: 1
Station address: 000000000000
IP-Address: 172.016.129.149
Subnet-Mask: 255.255.224.000
Router1 : 000.000.000.000
Router2 : 000.000.000.000
Router3 : 000.000.000.000

Request the required station addresses from your system administrator.

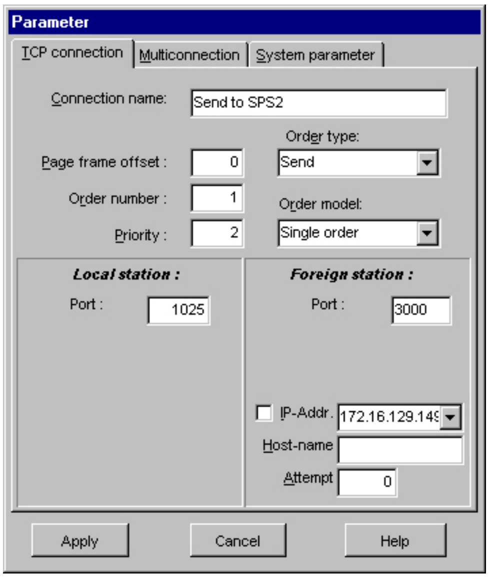
If necessary, you can enter additional settings into the configuration windows.

TCP/IP-connections

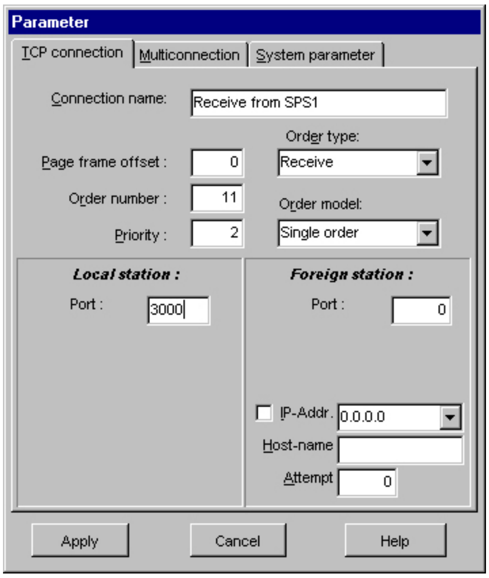
You configure your TCP/IP connection by inserting a TCP connection below the stations by means of TcpIP and entering the following parameters for the stations:

TCP/IP-connections

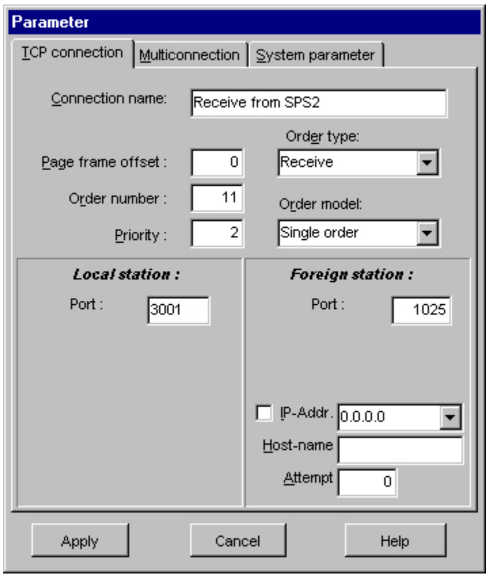
Station 1 Station 2



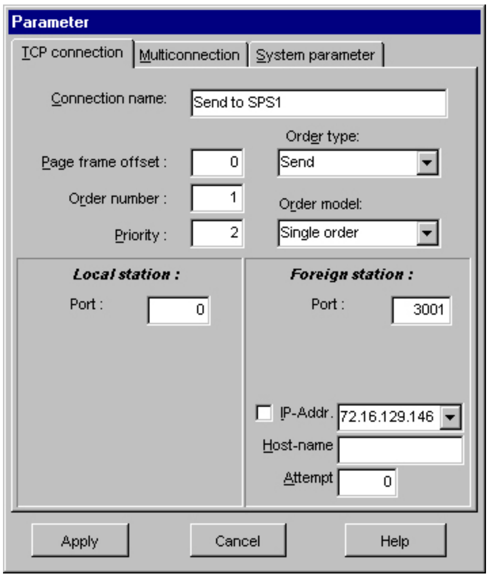
Connection name: Send to SPS2
Order type: Send
Page frame offset : 0
Order number : 1
Order model: Single order
Priority : 2
Local station : Port : 1025
Foreign station : Port : 3000
□ IP-Addr. 172.16.129.14(
Host-name
Attempt 0



Connection name: Receive from SPS1
Order type: Receive
Page frame offset : 0
Order number : 11
Order model: Single order
Priority : 2
Local station : Port : 3000
Foreign station : Port : 0
□ IP-Addr. 0.0.0.0
Host-name
Attempt 0



Connection name: Receive from SPS2
Order type: Receive
Page frame offset : 0
Order number : 11
Order model: Single order
Priority : 2
Local station : Port : 3001
Foreign station : Port : 1025
□ IP-Addr. 0.0.0.0
Host-name
Attempt 0

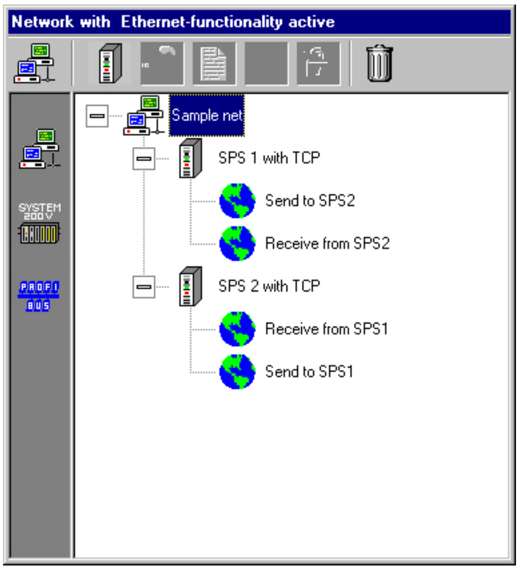


Connection name: Send to SPS1
Order type: Send
Page frame offset : 0
Order number : 1
Order model: Single order
Priority : 2
Local station : Port : 0
Foreign station : Port : 3001
□ IP-Addr. 72.16.129.146
Host-name
Attempt 0

Save your project!

Network window

Your Network window should have the following content:



Network with Ethernet-functionality active

- Sample net
  - SPS 1 with TCP
    - Send to SPS2
    - Receive from SPS2
  - SPS 2 with TCP
    - Receive from SPS1
    - Send to SPS1

Transferring the configuration data into the CPUs

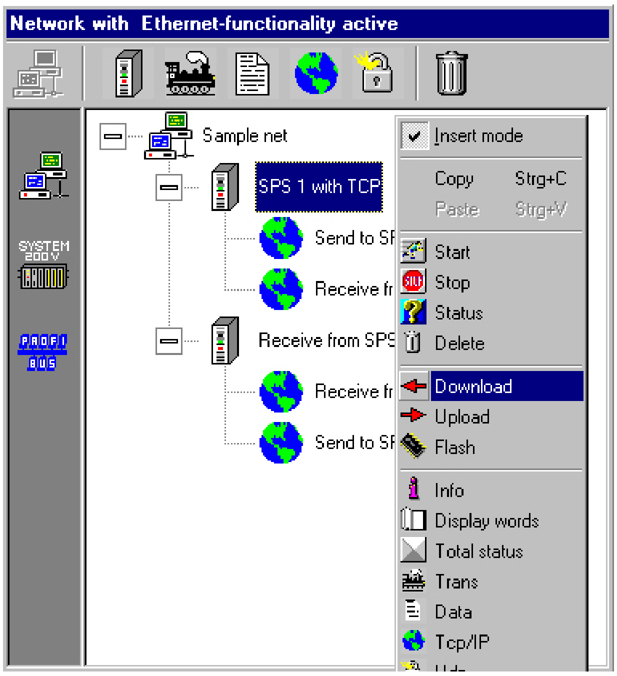
You may transfer your configuration data online via the network into the respective CPUs. Create the system structure as shown above and start both CPUs.

Note!
Please regard, that both modules have the same IP address at the initial start. For the initial project engineering you should activate the CPUs one after the other.

Approach

- Switch on the power supply of CPU 1 and wait app. 15 sec until the CPU has started.
- Click in "Parameter"-window on [Search stations] → The available VIPA CPs are listed by their IP address. If your target CP is inside your IP circle, the CP can online be projected. Otherwise you have to change the IP address by using [Change IP]. After that click to [Search stations]. Please regard! If you change the IP Address of the CP by using [Change IP], the configuration inside the CP is cleared.
- Double click on the wanted CP, the CP gets online. Set the CP into software STOP.

- Click on the first station in the *Network*-window. Click the right mouse button and choose the "Download" command → the project is now transferred into the Flash-ROM of the first CP.



- Switch your 1. CPU off and on again. → After the start-up, the 2. CP is included in the network with the configured IP address.
- Now connect the 1. CP with your network and start the CP.
- Click in "Parameter"-window on [Search stations]
- Otherwise you have to change the IP address by using [Change IP]. After that click to [Search stations].
- Via double click on the wanted CP, the CP gets online. Set the CP into software STOP.
- Click in the *Network*-window of WinNCS on the 2. station. Use the right mouse button and choose the command "Download". → your project is now transferred into the Flash-ROM of the 2. CP.
- Switch your 2. CPU off and on again. → After the start-up, the 2. CP is included in the network with the configured IP address.

If no error occurs, both CPs are switching automatically in RUN after PowerON.

The configuration of the CP section is now ready. The following page describes the programming of the PLC section.

PLC programs for the CPUs

Both CPUs use the same PLC program.

OB100 Synchronization of the interfaces

Synchronization of the interfaces

In the start-up operating module OB 100 of the CPU the interface used on the CP must be synchronized by means of the handler block SYNCHRON.

OB 100 verifies that the synchronization procedure was completed without errors. If an error is detected, the error number is entered into MB 200.

OB 100:

```
CALL "Synchron"
  SSNR:=0
  BLGR:=6
  PAFE:=MB200
```

OB 1 Cycle

Via the cycle OB OB1 the sending and receiving of data is controlled. The initiation of transmission in station 1 is issued by means of a SEND of the FC1. The partner station responses with RECEIVE (FC2). After that the data is transmitted via SEND_ALL and received in the partner station by means RECV_ALL.

OB 1:

```
CALL "RECV_ALL"       //Data transfer CP to CPU
  SSNR:=0
  PAFE:=MB197
  ANZW:=MD184               //Only at RECV_ALL ANZW
                            //contains the number of
                            //the current order

CALL "Send_All"             //Data transfer CPU to CP
  SSNR:=0
  PAFE:=MB196
  ANZW:=MD180               //Only at Send_All ANZW
                            //contains the number of
                            //the current order

CALL FC 1                   //Execute Send
CALL FC 2                   //Execute Receive
```

FC 1 - SEND
FC 2 - RECEIVE

FC 1:

```
CALL "Control"                     //Control Send order
  SSNR:=0
  ANR:=1
  PAFE:=MB195
  ANZW:=MD174
O       M       175.1              //Order in process or
O       T       1                  //timer on
BEB

L       S5T#1s                     //Set timer value at 1s
CLR
SV      T       1
SET
SV      T       1                  //Start timer

CALL "Send"                        //Send order
  SSNR:=0
  ANR:=1
  IND:=0
  QANF:=P#DB11.DBX 0.0 BYTE 16     //Source: DB11 from DW0, 16Byte
  PAFE:=MB194
  ANZW:=MD174

L       DB11.DBW   0               //Telegram counter
+       1                          //in DB11,
T       DB11.DBW   0               //increment DW0
```

FC 2:

```
CALL "Control"                     //Control Receive order
  SSNR:=0
  ANR:=1
  PAFE:=MB195
  ANZW:=MD184
UN      M       185.0              //if handshake not
BEB                                //convenient, end

CALL "Receive"                     //Receive order
  SSNR:=0
  ANR:=11
  IND:=0
  ZANF:=P#DB12.DBX 0.0 BYTE 16     //Dest: DB12 from DW0, 16Byte
  PAFE:=MB193
  ANZW:=MD184

L       MB      188                //Increment telegram
+       1                          //counter in MB188
T       MB      188
```

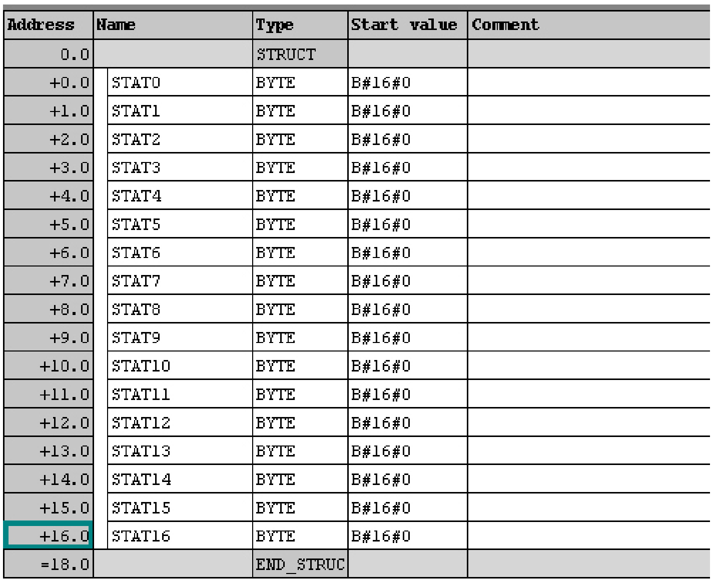
Data blocks DB 11, DB 12

The frequency, with which a SEND job is issued, depends on the time that was configured for the FC1 call. The timer in this example is configured for 1000ms. The sample program initiates the SEND job at a rate of once every 1000ms.

Data Word DW 0 of the data Block DB 11 is incremented ahead of the SEND call, that actually transmits a message. This occurs in function block FC 1. All together 16bytes of data are transmitted. The partner station receives the data and stores it in DB 12.

Together with DW 0, 15bytes of additional user data may be transferred.

The data blocks DB 11 and DB 12 have the same structure:



Address	Name	Type	Start value	Comment
0.0		STRUCT		
+0.0	STAT0	BYTE	B#16#0	
+1.0	STAT1	BYTE	B#16#0	
+2.0	STAT2	BYTE	B#16#0	
+3.0	STAT3	BYTE	B#16#0	
+4.0	STAT4	BYTE	B#16#0	
+5.0	STAT5	BYTE	B#16#0	
+6.0	STAT6	BYTE	B#16#0	
+7.0	STAT7	BYTE	B#16#0	
+8.0	STAT8	BYTE	B#16#0	
+9.0	STAT9	BYTE	B#16#0	
+10.0	STAT10	BYTE	B#16#0	
+11.0	STAT11	BYTE	B#16#0	
+12.0	STAT12	BYTE	B#16#0	
+13.0	STAT13	BYTE	B#16#0	
+14.0	STAT14	BYTE	B#16#0	
+15.0	STAT15	BYTE	B#16#0	
+16.0	STAT16	BYTE	B#16#0	
=18.0		END_STRUC		

Program transfer

The data transmission takes place via MPI. If your programming device (PG) has no MPI adapter, you may use the "Green Cable" from VIPA (VIPA 950-0KB00) to establish a serial point-to-point connection between the PC and the CPU.

The "Green Cable" may only be used with the VIPA CPUs of the Systems 100V, 200V and 300V.

- Connect your PG with the CPU.
- Via **PLC** > *Download* in your projecting tool you transfer the project into the CPU.
- Plug in a MMC and transfer your user application to the MMC via **PLC** > *Copy RAM to ROM*. During the write operation the "MMC"-LED on the CPU is blinking. For internal reasons the message signaling completion of the write operation arrives too soon. The write operation is only completed, when the LED has been extinguished.
- Switch both CPUs into RUN.

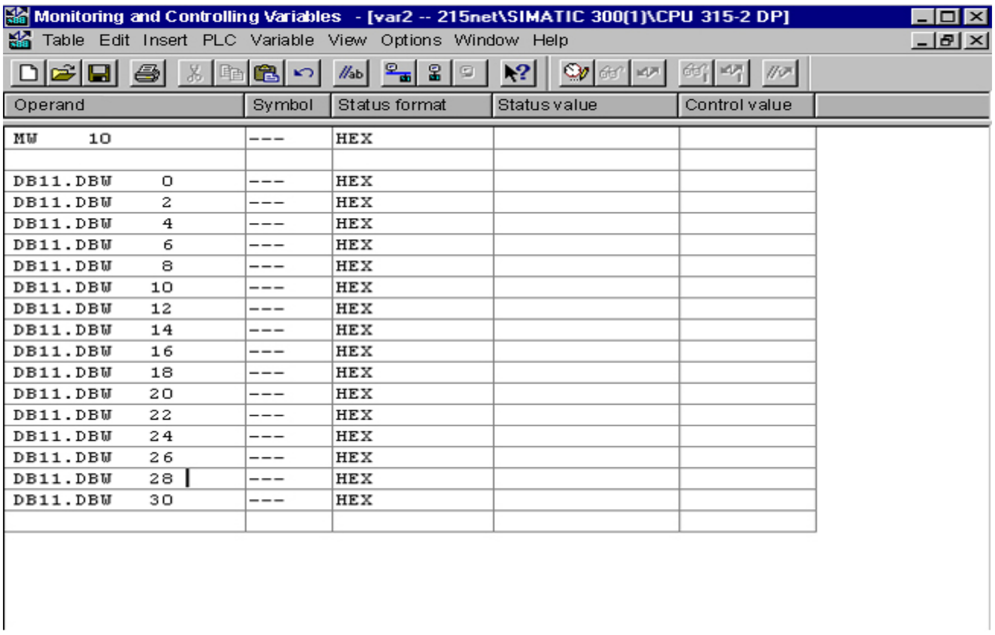
Monitoring the data transfer in Siemens SIMATIC Manager

It is assumed, that the CPs are programmed and that an overall reset was issued to the CPUs, where the RUN/STOP switch must be located in STOP position.

Now load the above PLC programs into both CPUs and switch them into RUN.

Start the Siemens SIMATIC Manager and execute the following steps to monitor the transmit job:

- **PLC** > *Monitor/Modify Variables*
- In the column "Operand" you have to enter the respective data block number and the data word (DB11.DW0-15).
- Establish a connection and click "monitor" .


Monitoring and Controlling Variables - [var2 -- 215net\SIMATIC 300(1)\CPU 315-2 DP]
Table Edit Insert PLC Variable View Options Window Help

Operand	Symbol	Status format	Status value	Control value
MW 10	---	HEX		
DB11.DBW 0	---	HEX		
DB11.DBW 2	---	HEX		
DB11.DBW 4	---	HEX		
DB11.DBW 6	---	HEX		
DB11.DBW 8	---	HEX		
DB11.DBW 10	---	HEX		
DB11.DBW 12	---	HEX		
DB11.DBW 14	---	HEX		
DB11.DBW 16	---	HEX		
DB11.DBW 18	---	HEX		
DB11.DBW 20	---	HEX		
DB11.DBW 22	---	HEX		
DB11.DBW 24	---	HEX		
DB11.DBW 26	---	HEX		
DB11.DBW 28	---	HEX		
DB11.DBW 30	---	HEX		

Entering User data

You may enter user data starting with DW1. Place the cursor on *modify value* and enter the value you wish to transfer, e.g. W#16#1111.

The button transfers the modify value in every cycle and the button initiates a single transfer.

Support for error search

This page shall help you with fast error localization. Due to the complexity of TCP/IP errors may easily occur but are hard to identify.

Question	Remedy at "no"
CPU in Run?	Control DC 24Vpower supply. Set RUN/STOP switch into RUN. Control PLC program and transfer it again.
SEND_ALL, RECV_ALL in OB 1?	For data transfer between CP and CPU these 2 blocks are required.
CP in RUN?	Control status in WinNCS and switch the CP to Run if necessary. Control SYNCHRON block in OB 100
Is the CP able to establish a connection?	Control the Ethernet core (at a point-to-point connection you have to use a twisted pair Ethernet cable). Control the IP address. Please regard that TCPTest doesn’t show the complete IP address!
May data be transferred?	Control port no. for read and write. Control source and destination area.

Appendix

A Index

M.Stich