3.6 TCP/IP coupling (CP5100)

Introduction

This application software example is intended to show how a design engineer should proceed when implementing a SIMATIC TDC subrack coupling via TCP/IP or UDP.

The configuration described here includes the basic hardware equipment as well as function blocks and shows how it is used. A conscious decision was made to keep the functional scope of this application software extremely low, so that those reading it can get up to speed quickly on the subject. The functionality and/or the hardware components can be expanded at any time. However, the information in the applicable function block documentation must be observed.

All of the names used in this application software have been randomly selected and are only binding for this configuration example.

The structure of these configuring instructions essentially determines the sequence that the various steps are made with which the complete application software can be generated. However, this should only be considered as a recommendation and does not have to be rigidly maintained.

Application cases

CP5100 can be used for the following applications:

- Exchanging process data with other CP5100 and SIMATIC Industrial Ethernet modules (e.g. CP443-1)
- Visualizing process data using WinCC
- Visualizing messages using WinCC
- Exchanging process data with third-party systems (e.g. process computer)

NOTE When integrating new modules into an existing TCP/IP network, you should always ask your network administrator about the IP, sub-network and router addresses as well as the port numbers for applications.

Literature on this subject:

- TCP/IP basics
  (e.g. W. Richard Stevens, TCP/IP Illustrated, Volume 1, Addison-Wesley Verlag)

CP5100 hardware documentation

3.6.1 Comparison between TCP/IP and UDP

TCP/IP

The TCP/IP protocol type is connection-oriented. This means that data can only be sent if the coupling partner can also be addressed. As long as there is a connection to the coupling partner, the TCP/IP protocol ensures that the data which is sent also arrives at the coupling partner. If a fault condition develops, the data are, if necessary, transferred several times.

If a connection is only established at one end, **data could be lost** in the TCP/IP stack of the CP5100 or the coupling partner (e.g. PC) as a result of the protocol.

NOTE For important information, data reception must be monitored at the **application level**.

UDP

The UDP protocol type is not connection-oriented. This means that data is also sent even if the coupling partner is temporarily not addressable. **Data is then lost**. However, received data is correct as a result of the data security mechanisms.

When compared to TCP/IP, UDP can achieve a higher communications speed between the coupling partners. However, this is only possible with a somewhat restricted data transfer security.

Channel modes

The following channel modes are possible using the TCP/IP and UDP protocols:

- Handshake
- Refresh
- Multiple
- Select

Typical applications

The **Refresh** and **Multiple** channel modes are overwriting modes (i.e. older data can be overwritten by more recent data). Thus, for these two channel modes, **UDP** is especially suitable as data transfer protocol.

The **Handshake** and **Select** channel modes are not overwriting; in this case **TCP/IP** should be recommended as data transfer protocol.

Client or server

An overview as to when SIMATIC TDC operates as TCP/IP client or server is provided in the following table:

SIMATIC TDC (TCP/IP) process data		Communications partner
Function block CTV (client)	Connect → ← accept Send data →	Receiver
Function block CRV (server)	← Connect accept → ← Send data	Sender

3.6.2 Typical configuration

The following hardware components are required per station as a minimum for a coupling from SIMATIC TDC via TCP/IP:

- **UR5213** subrack
- **CPU550** CPU module with **MC5xx** memory module
- **CP5100** communications module (here at slot 18)

NOTE

CP5100 may only be configured at **slots 18 to 21**. This means that a maximum of four CP5100 modules can be operated in parallel in a subrack.

Depending on the selected slot, BCD coding switches S1 to S3 (refer to Fig. 3-19 for their location on the module) must be set on the module according to the table below. The combination “C-8-0” is obtained for the typical configuration.

Slot	S1	S2	S3
18	C	8	0
19	C	8	4
20	C	8	8
21	C	8	C

((Fig. below: Coding switches to adapt to the slot in the subrack))

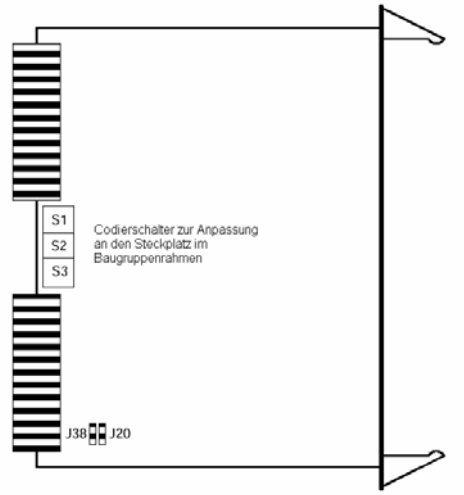


Fig. 3-19: Position of the coding switches on the CP5100 (side view)

3.6.3 Configuring steps

Only the configuring steps for one station are shown in the following. All of the other stations can be essentially handled in the same way.

3.6.3.1 Configuring in HWConfig

Parameterizing the CP5100

The CP5100 is parameterized via the appropriate tab in the object property dialog box.

To parameterize the Ethernet interface, the following relevant settings must be specified in the sub-tab “Properties of the CP5100 Ethernet interface”:

- **IP address ("Parameter" tab)**
  IP address of the module in the "dot" notation, in this case:
  141.20.135.197
- **Sub-network mask ("Parameter" tab)**
  Sub-network mask to designate the network segment,
  in this case: **255.255.0.0** for a Class B network
- **IP address of the default router ("Parameter" tab)**,
  in this case: **"Use no router"**

NOTE

It must be ensured that the IP address and sub-network mask are harmonized. For example, for a Class C IP address (193.x.y.z), the Class B sub-network mask 255.255.0.0 is not permissible. Detailed information on how to select the IP address and its value ranges is provided in the online help for CP5100.

The following diagram shows the settings, specified above under the "Parameter" tab:

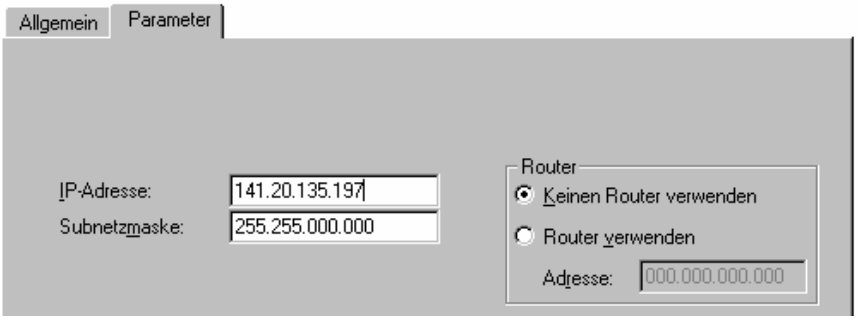


Fig. 3-20: CP5100 network settings

3.6.3.2 Configuring with CFC

Only the most important I/O are handled in the following. Deviations, which are obtained when configuring a UDP coupling, are described at the appropriate locations.

FBs required

The following function blocks are required for this example:

- **@TCPIP** central block to initialize and monitor the CP5100 module **(this is always required)**
- **CRV** receive block (in this case, optional, for the process data coupling)
- **CTV** send block (in this case, optional, for the process data coupling)

3.6.3.2.1 Central block @TCPIP

Configuring

Connection	Connection assignment (significance)
CTS	**D1800C** (module name of the configured CP5100)

CFC chart

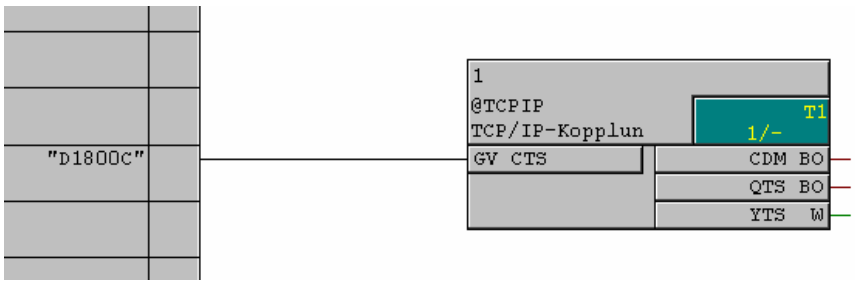


Fig. 3-21 : Connection assignment of the @TCPIP

3.6.3.2.2 Receive block CRV

"AR" connection

Initialization connection to enter an address. **Address stage 1** must be specified in addition to the channel name.

Rules for address stage 1:

- The first character (letter) defines the required protocol (“**T**“ = TCP/IP, “**U**” = UDP) (in this case, "**T**").
- The second character must be a “**-**”.
- The channel port number is defined by the next **5 digits**, whereby leading zeros must be specified (in this case: "**01024**" for port number 1024).

Only ports 1024 to 65535 should be used as port number. This should be harmonized with the system administrator. Port numbers up to and including 1023 are generally reserved for “well known services” and “unix-specific services”.

Configuring

Connection	Connection assignment (significance)
CTS	**D1800C** (module name of the configured CP5100)
AR	**RXKAN1.T-01024** (address parameter, receive)
MOD	**H** (receive mode)
EN	**1** (enable)

CFC chart

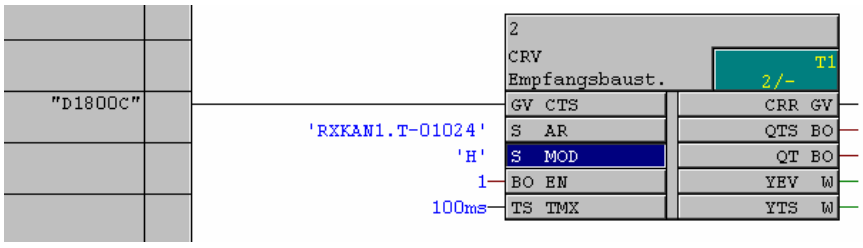


Fig. 3-22: Connection assignment of the CRV

3.6.3.2.3 Send block CTV

"AR" connection

Initialization I/O for address data. **Address stage 1** and **address stage 2** must be specified here, in addition to the channel names.

The two address stages are separated in the notation by a "**.**".

Rules for address stage 1:

- The first character (letter) defines the required protocol (“**T**” = TCP/IP, “**U**” = UDP) (in this case, "**T**").
- The second character must be a “**-**“.
- The channel port number is defined by the next **5 digits**, whereby leading zeros must be specified (in this case: "**01024**" for port number 1024).

Only ports from 1024 to 65535 should be used as port number. Port numbers up to and including 1023 are generally reserved for “well known services” and “unix-specific services”.

Rules for address stage 2:

- The first **12 digits** define the IP address of the remote coupling partner. This is entered in the so-called “dot" notation; however without specifying the separating point. Leading zeros should be specified (in this case: "**141020135198**" for IP address 141.20.135.198).
- The 13th character must be a “**-**”.
- The port number of the remote coupling partner is defined by the next **5 digits**; leading zeros should be specified (in this case: "**01024**" for port number 1024).

Configuring

Connection	Connection assignment (significance)
CTS	**D1800C** (module name of the configured CP5100)
AR	**TXKAN1.T-01024.141020135198-01024** (address parameter, send)
MOD	**H** (send mode)
EN	**1** (enable)

CFC chart

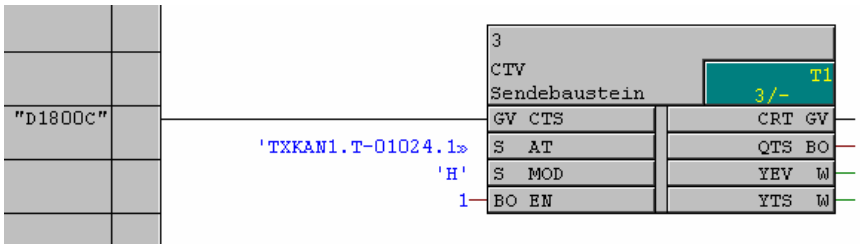


Fig. 3-23 Connection assignment of the CTV

3.6.4 Application information

3.6.4.1 Channel number

A maximum of **256 channels** can be set-up on the TCP/IP module using send and receive blocks (e.g. CTV and CRV).

The actual possible number of channels depends on the size and the number of net data and the access mechanism (handshake, refresh). 254 Kbytes of RAM on the CP5100 are available for the data interface.

Calculation

The number of channels can be roughly calculated according to the following formula

- **per refresh/multiple channel:** 150 bytes + 2 * number of net data bytes (size of the virtual connection)
- **per handshake/select channel:** 150 bytes + 1 * number of net data bytes

3.6.4.2 Telegram length

The telegram length is restricted as follows:

- To 55759 bytes for receiving
- 65535 bytes for sending

3.6.4.3 "Ping" on CP5100

Communications with the CP5100 can be checked using a "Ping". The module only responds to a "Ping" if the communications partner is located within the particular network or a default router (within the particular network) can establish the coupling.

3.6.4.4 Performance

The performance of the TCP/IP coupling depends on the configuration used. The following data indicate the performance values for a typical configuration.

Prerequisites

- For telegram lengths 200 and 1000 bytes, 32 send and 32 receive UDP connections are configured for each CPU (refresh).
- For a telegram length of 4000 bytes, 8 send and 8 receive connections are configured.

Results

The number of send/receive tasks per CP5100 is approx. **1600**.

The maximum data transfer rate is approx. 1.2 Mbyte/s (telegram length of 1000 bytes); this cannot be increased, in spite of the higher channel lengths.

Data transfer rate [kbytes/s]	Tasks/s	Send/receive cycle [ms] / connections	Telegram length [bytes]
320.000	1600	64 / 40	200
1142.857	1143	64 / 56	1000
1000.000	250	64 / 16	4000