applications & TOOLS

Connection between PROFIBUS Networks and DeviceNet by Anybus X-Gateways

SIEMENS

Configuration example

Note The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are correctly used. These Application Examples do not relieve you of the responsibility of safely and professionally using, installing, operating and servicing equipment. When using these Application Examples, you recognize that Siemens cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice. If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications – e.g. Catalogs – then the contents of the other documents have priority.

Warranty, liability and support

We do not accept any liability for the information contained in this document.

Any claims against us - based on whatever legal reason - resulting from the use of the examples, information, programs, engineering and performance data etc., described in this ... shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health The above provisions does not imply a change in the burden of proof to your detriment.

For questions about this document please use the following e-mail-address:

mailto:csweb@ad.siemens.de

Foreword

Objective of the application

The demand for the connection of networks of different manufacturers increases worldwide. For economic and technological reasons, interoperability is of greatest importance.

In this context, the products manufactured by Allen-Bradley/Rockwell are of particular interest. Together with a large number of partner companies, Allen-Bradley sells a broad portfolio of controllers, I/O devices and network components, which hold a considerable market share particularly in the USA.

This application uses an example to show how Allen-Bradley “DeviceNet” networks[1] can be connected to PROFIBUS networks supported by SIMATIC. An “Anybus X-gateway” module manufactured by HMS is used.

Main contents of this application

The widely ramified product range of both Allen-Bradley and Siemens makes it impossible to comprehensively explain all possible combinations in one single application. For this reason, the focus of the example described in this document is the use of the Anybus X-gateways. (Alternatives are listed in chapter 2.5)

Delimitation

This Configuration does not include in-depth descriptions of

- principles of programming Allen-Bradley controllers,
- Allen-Bradley networks.

This Configuration does not contain code examples; instead, the shown step tables combined with the guide of the individual component manufacturers enable the reader to create his/her own application.

Additional information

This selection aid is intended as an extension of the application *Communication with Allen-Bradley “ControlLogix” Controllers via PROFIBUS Scanner* (ID Number 23809864, see \3\) and the configuration *INAT Echochange Gateway between Allen-Bradley EtherNet/IP Networks and SIMATIC Industrial Ethernet Networks* (ID Number 23901499, see \4\). The use of the PROFIBUS scanner or the Echochange module is an alternative to using the Anybus gateway described in this document.

[1] For the designation of DeviceNet as “Allen-Bradley network”, see also the note on page 6.

In addition, document 23809864 includes essential background information on Allen-Bradley controllers and network technologies. We recommend reading this document.

Structure of the document

The documentation of this application is divided into the following main parts.

Part	Description
Application Description	You are provided with a general overview of the contents. You are informed on the used components (standard hardware and software components).
Principles of Operation and Program Structures	This part describes the detailed functional sequences of the involved hardware and software components, the solution structures and – where useful – the specific implementation of this application. It is only required to read this part if you want to familiarize with the interaction of the solution components to use these components, e.g., as a basis for own developments.
Structure, Configuration and Operation of the Application	This part takes you step by step through structure, important configuration steps, startup and operation of the application.
Appendix	This part of the documentation includes further information, e.g. bibliographic references, glossaries, etc.

Reference to Automation and Drives Service & Support

This entry is from the internet application portal of Automation and Drives Service & Support. Clicking the link below directly displays the download page of this document.

http://support.automation.siemens.com/WW/view/en/23902276

Table of Contents

Application Description

Contents

You are provided with an overview of Allen-Bradley components and technologies and connection options to SIMATIC networks. The main focus is on the connection of PROFIBUS to DeviceNet (Allen-Bradley).

1 Automation Problem

You are provided with information on…

... the classes of controllers and networks that are sold by Allen-Bradley and the problems that occur when connecting to SIMATIC networks.

1.1 Overview

Note From a legal point of view, Allen-Bradley is a subsidiary of Rockwell Automation, which deals with the development and the application of programmable controllers.

To avoid confusion, the term “Allen-Bradley” designates all products sold by Rockwell and Allen-Bradley in this application.

Note In this document, *DeviceNet* is referred to as “Allen-Bradley network”, although it has in the meantime developed into an open standard.

The reason for this is that DeviceNet was developed by Allen-Bradley (see chapter 3.1) and that the concerns of Allen-Bradley controllers have priority in this documentation.

Introduction

Allen-Bradley currently has a significant market share particularly in the United States. To serve the combination of different networks and the increasingly desired interoperability, this application uses an example to present an option to operate a network with SIMATIC network nodes in conjunction with an Allen-Bradley network.

Allen-Bradley controllers and networks

Over the years, Allen-Bradley has developed a number of different controller classes – PLC 5, SLC 500, ControlLogix, etc. – and network types – DH+, DH 485, ControlNet, EtherNet/IP, etc. – which clearly differ in their performance and their field of application. For this reason, it is not possible to provide a universally applicable solution for connecting the “SIMATIC world” to Allen-Bradley.

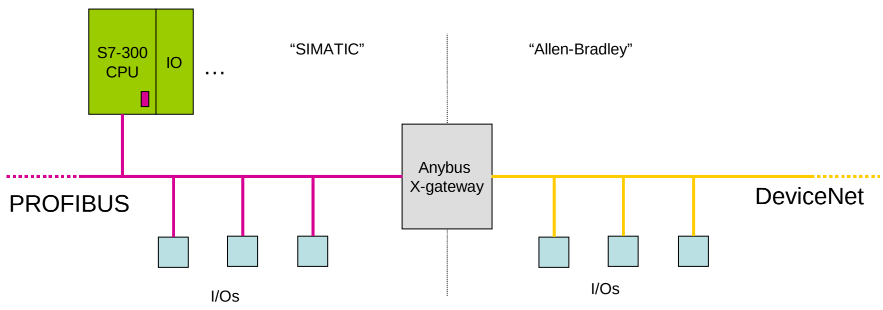
For a detailed description of the Allen-Bradley product range, please refer to \3\.

Our example presents the use of a **PROFIBUS**-networked controller in conjunction with a **DeviceNet network** on the Allen-Bradley side.

Overview of the automation problem

The figure below provides an overview of the automation problem.

Figure 1-1



The automation problem consists of combining two subnetworks, i.e.,

- a PROFIBUS network with at least one SIMATIC CPU and several I/O slaves and
- a DeviceNet with several I/O modules.

Description of the automation problem

DeviceNet is based on an open, serial master/slave protocol which is based on the CAN bus standard. It is primarily intended for networks at field level and enables to connect simple I/O devices such as I/O modules to a central controller and to transmit process and control values. For further information, please refer to chapter 3.1.

In the example of a solution described here, a Flex I/O head module with one input and one output module as a DeviceNet slave is set up in DeviceNet. This Flex I/O module exchanges data with the S7-300 CPU controlling the PROFIBUS as a master.

An Anybus X-gateway forms the interface between the two networks; it acts as a slave on the PROFIBUS side, but as a master on the DeviceNet side.

2 Automation Solution

You are provided with information on…

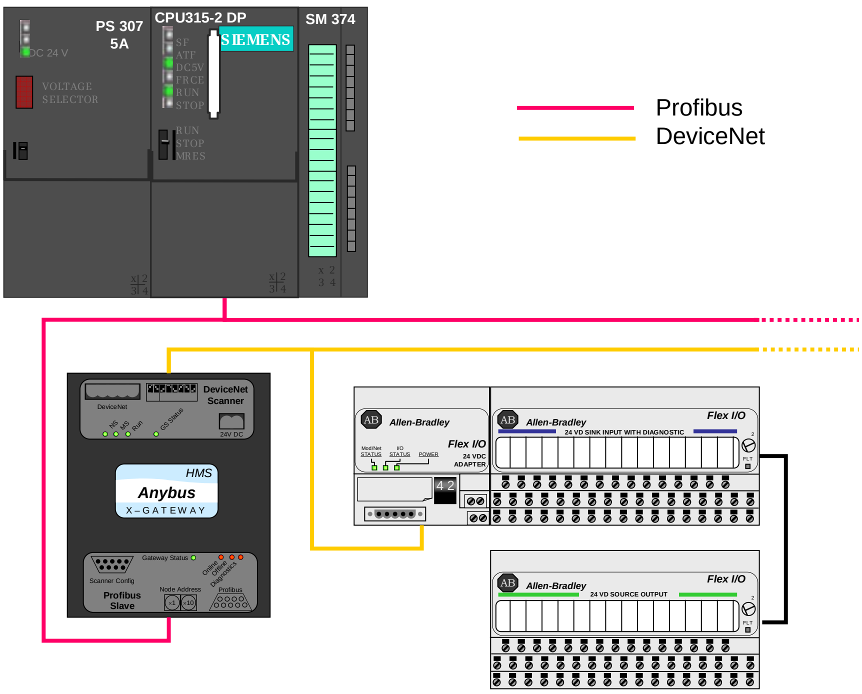
the solution selected for the automation problem.

2.1 Overview of the overall solution

Diagrammatic representation

The following figure schematically shows the most important components of the solution:

Figure 2-1



Configuration

The configuration presented in our example of a solution consists of a CPU 315-2DP with an SM374 I/O module acting as a master on a PROFIBUS segment. The other subnetwork is a DeviceNet segment with a Flex I/O head module and one input and one output module. (For more information on DeviceNet, please refer to chapter 3.1)

An Anybus X-gateway with two interfaces connects the two network segments. One of the interfaces is connected to PROFIBUS, the other interface is connected to DeviceNet. (For more information on the principle of operation of the Anybus gateways, please refer to chapter 3.2) Correspondingly, the selected Anybus product variant is a *PROFIBUS DP slave, DeviceNet scanner (master)*. To PROFIBUS, the gateway thus acts as a DPV1 slave which provides Class 1 and Class 2 services, the gateway acts as a scanner on DeviceNet.

The indirect path via the gateway thus enables the SIMATIC S7 CPU to communicate with the DeviceNet Flex I/O modules via PROFIBUS DP.

2.2 Description of the core functionality

The Anybus X-gateway allows the data exchange between two subnetworks. In this configuration, the S7-300 CPU communicates with the DeviceNet stations in both directions. From the point of view of the PROFIBUS master, the data of the DeviceNet stations are mapped to the input/output memory areas of the slave interface on the gateway.

For a detailed description of the principle of operation, please refer to 4.1.

Advantages of this solution

The use of Anybus X-gateways offers several advantages.

These advantages include in particular the reduced additional cabling and configuration overhead since only one individual component has to be fit to connect two already existing networks.

In addition, the Anybus X-gateway is very flexible and its configuration can meet a large number of requirements.

Finally, the use of gateways is economic and once you have learned the procedure, it is simple to configure also other Anybus gateway versions for the use in different networks or configurations.

2.3 Required hardware and software components

Note In the tables below, components not sold by Siemens are silhouetted in gray. The respectively specified sales sources are responsible for their procurement (see also page 11).

Hardware components

Table 2-1: Required hardware components

Component	No.	MLFB / Order number	Note
SIMATIC Field PG M Standard	1	6ES7712-0AA0.-0XXX	Or comparable PC with MPI
PS 307 load power supply	1	6ES7307-1BA00-0AA0	Or comparable power source
SIMATIC S7-300 CPU 315-2DP	1	6ES7315-2AG10-0AB0	Or S7-400
SM 374 simulator module	1	6ES7374-2XH01-0AA0	Or DI8/DO8 module with digital inputs/outputs
Power supply and head module	1	1794-ADN	Procurement via 1
Flex I/O input module with diagnostic function	1	1794-IB16D	Procurement via 1
Flex I/O output module	1	1794-OB16	Procurement via 1
Flex I/O backplane module (input)	1	1794-TB32	Procurement via 1
Flex I/O backplane module (output)	1	1794-TB2	Procurement via 1
DeviceNet PCMCIA adapter (PC card connector to 5-pos open-style)	1	1784-PCD-1	Procurement via 1; for the configuration
Anybus X-gateway DeviceNet scanner (master), PROFIBUS slave	1	AB7663	Procurement via 2

Standard software components

Table 2-2: Required software components

Component	No.	MLFB / Order number	Note
Simatic S7, Step 7 V5.4 (or higher)	1	6ES7810-4CC08-0YA5	
RSLogix 5000 Standard Edition, V13.03 (or higher)	1	9324-RLD300DEE	(Optional, only for diagnostics) German version, procurement via 1
RSNetWorx for DeviceNet	1	9357-DNETL3	Version 5.0 or higher,[2] procurement via 1
NetTool-DN configuration tool hardware and software for configuring the DeviceNet scanner interface[3]	1	018020	Version 3.0 or higher. Also includes an adapter serial → DeviceNet.[4] procurement via 2

Sources of supply for Germany:

1. **Rockwell Automation** regional office of Rockwell Int'l GmbH
   Düsselbergerstrasse 15
   42781 Gruiten
   Germany
   Tel: +49 2104 9600
   Fax: +49 2104 960 121
   (see also \8\)
2. **HMS Industrial Networks GmbH**
   Haid-und-Neu-Str. 7
   76131 Karlsruhe
   Germany
   Tel: +49 721 96472-0
   Fax: +49 721 96472-10
   (see also \5\)

[2] A demo version which is also available is limited to the configuration of devices with DeviceNet addresses between 0 and 6. This is basically sufficient for the application presented by us.
[3] The PROFIBUS interface can be configured using Simatic S7.
[4] The demo version of the configuration software to be downloaded from the web free of charge is limited to 2 nodes and does not include an adapter to DeviceNet. Thus, it does not meet the requirements of our configuration.

Example files and projects

Configurations or code examples are not delivered with this Configuration, i.e., no further files are available except for this document.

Table 2-3: Files/documents included in the delivery

Component	Note
23902276_Anybus-Gateway_DOKU_V10_e.pdf	This document.

For links for the download of necessary configuration files (GSD, EDS files), please refer to the respective sections of this document or the installation instructions of the corresponding modules.

2.4 Performance data

In this configuration (PROFIBUS slave, DeviceNet scanner), the Anybus X-gateway has the following features:

- Complete PROFIBUS-DPV1 slave functionality (Class 1 and Class 2 services)
- Maximum number of managed DeviceNet slaves: 63
- Maximum data rate:
  - DeviceNet: 500 kbps (settable via hardware)
  - PROFIBUS: 12 Mbps (automatic detection)
- Data areas:
  - Process data: Maximum of 244 bytes of input and output data each, however, not more than a total of 416 bytes
  - Parameter and diagnostic data: Up to 512 bytes of acyclic PROFIBUS DPV1 parameter and diagnostic data

Depending on the quantity of configured diagnostic data, the actually usable areas can be insignificantly smaller.

2.5 Alternative solutions

The large number of networks supported by Allen-Bradley and the wide range of used hardware simply makes it impossible to provide a universal solution for all possible network combinations and applications. (A total of approx. 170 product variants exist for the Anybus X-gateways alone.)

Connections from PROFIBUS to other Allen-Bradley networks

Examples of other options to connect the “Allen-Bradley world” to SIMATIC are –

- An SST PROFIBUS scanner which integrates an Allen-Bradley CPU into a SIMATIC PROFIBUS installation as a master. This configuration is particularly suitable for PLC-PLC communication and suited for use in a large number of hardware variants. See \3\; this document describes the integration of a MicroLogix CPU in the medium performance range.
- INAT GmbH sells “Echochange” gateways which connect Allen-Bradley EtherNet/IP networks to SIMATIC Industrial Ethernet networks. See \4\. This solution is of interest when using high-performance controllers for demanding tasks.

These are only some possibilities of establishing a cross-network communication. The optimum solution will always depend on the respective application and the used hardware configuration and these documents can only provide aids.

Alternatives to the Anybus X-gateway for connecting PROFIBUS to DeviceNet

Aside from the Anybus X-gateway sold by HMS GmbH, there are a number of products of other manufacturers with basically the same functionality, i.e., the connection of a PROFIBUS network to DeviceNet.

Examples:

- *DN-CBM-DP* PROFIBUS-DP / DeviceNet gateway of the “esd electronic system design” company (overview \14\, product description \15\, company home page \16\),
- *Bridgeway* of the “Pyramid Solutions” company (overview \17\, product description \18\, company home page \19\),
- Gateways of the Woodhead company (the module presented here is an adapter between a DeviceNet slave and an ET200S and the ET200S is connected to the controller via PROFIBUS; overview \20\, product description \21\, company home page \22\).

In general, the ODVA[5] (\10\) home page provides a good and current overview of the currently available product range.

[5] “Open DeviceNet Vendor Association”, an interest club of the manufacturers and retailers of DeviceNet products.

Principles of Operation and Program Structures

Contents

This part describes the detailed functional sequences of the involved hardware and software components, the solution structures and – where useful – the specific implementation of this application.

It is only required to read this part if you are interested in the interaction of the solution components.

3 Functional Mechanisms

You are provided with information on…

... basics of the principle of operation of DeviceNet networks and Anybus products which can be used to connect different networks.

3.1 DeviceNet

DeviceNet is a field bus system which was initiated by Rockwell and then transferred to the ODVA for further development.

Basics

Like ControlNet and EtherNet/IP DeviceNet belongs to the family of the CIP-based networks. CIP[6] forms the common application layer of these 3 industrial networks. DeviceNet uses CIP at the upper, application-oriented layers of the OSI-reference model (5 to 7) while basically the CAN specifications with some additional restrictions/extensions are used at the lower, transport-oriented layers (1 to 4).

DeviceNet is thus the implementation of CIP based on CAN[7].

[6] "Common Industrial Protocol", a standard defined by the ODVA
[7] "Controller Area Network", a field bus system originally developed for networking control devices in automobiles

Table 3-1

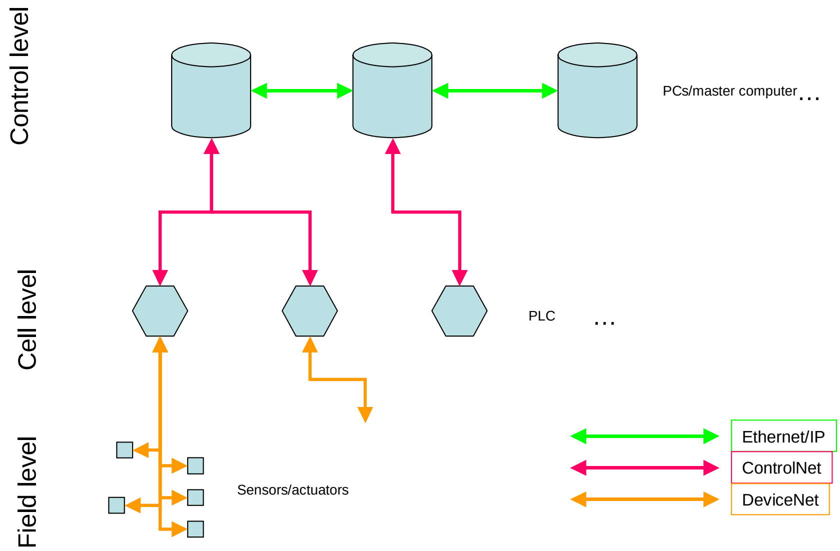
<table>
<tr><th colspan="2">Layer according to OSI</th><th>DeviceNet</th><th>ControlNet</th><th>EtherNet/IP</th></tr>
<tr><td>7</td><td rowspan="3">Application-oriented</td><td rowspan="3">CIP</td><td rowspan="3">CIP</td><td rowspan="3">CIP</td></tr>
<tr><td>6</td></tr>
<tr><td>5</td></tr>
<tr><td>4</td><td rowspan="4">Transport-oriented</td><td rowspan="4">CAN</td><td rowspan="4">...</td><td rowspan="4">TCP/IP</td></tr>
<tr><td>3</td></tr>
<tr><td>2</td></tr>
<tr><td>1</td></tr>
</table>

Relation to ControlNet and EtherNet/IP

DeviceNet, ControlNet and EtherNet/IP are tuned to one another and provide a graded communications system for control level (EtherNet/IP), cell level (ControlNet) and field level (DeviceNet) to the user.

The figure shows typical fields of application of the different networks.

Figure 3-1



However, it is, of course, also possible to use the networks at other levels by means of gateways.

Connection-oriented producer/consumer model

DeviceNet is an object-oriented bus system and operates according to the producer/consumer method.

When using “classic” sender/receiver protocols, messages are exchanged between one sender and one receiver. If data are to be transmitted to a large number of receivers (e.g. during the clock synchronization via the network), a corresponding number of messages has to be sent; this may have negative consequences for data throughput and consistency.

However, the producer/consumer model identifies messages with the aid of a connection ID and several receivers can “subscribe” to the same connection. (This connection is maintained for the entire duration of the communication.) The receivers thus become “consumers” of the data to be generated by the “producer”. This results in two effects:

- Only one message is required to send a data record to a group of receivers of any size (i.e., a type of multicast takes place),
- all receivers must request a “subscription” to a connection only once to receive all updates of the corresponding data in the future.

Both measures increase the security and the determination of the data communication.

On DeviceNet each station can basically appear as producer, consumer or both.

Data sources can either be configured in such a way that they are polled if required or that they send cyclic messages or “change-of-state” messages. In the last case, messages for data update are not sent at defined intervals but only if the respective values have changed. The use of change-of-state messages contributes to an additional considerable reduction of the data volume in DeviceNet.

Network properties

In a DeviceNet network, up to 64 bus nodes can communicate with one another via baud rates of 125, 250 or 500 kbaud. Aside from the two signals for the data transmission, the DeviceNet cable provides 2 lines for the supply of the DeviceNet bus nodes with 24 volt operating voltage, i.e., the nodes can be supplied via the bus or externally.

The installation is performed in a bus topology with or without branches and uses terminating resistors at both ends.

The most remarkable characteristic of CAN, on which DeviceNet is based, is a non-destructive method for the elimination of collisions, while simultaneously transmitting messages of several stations. Each message has its own priority in CAN and if several stations simultaneously send one message, always the station whose message has the highest priority prevails. 0-8 bytes of user data can be transmitted in a CAN message.

For further information on DeviceNet and its use, please refer, for example, to \9\.

3.2 Anybus X-gateways and other Anybus products

The “Anybus” products sold by HMS GmbH (see \5\, \13\) do not represent a separate field bus standard; they are protocol converters which are to ensure a type of “universal bus connection”.

The products are divided into different categories:

Embedded products and PCI cards:

These products are modules for the direct installation into field devices or PC cards for the PCI bus.

On the network side, the modules provide an interface to one of the supported networks acting as a master or slave (compare list on page 17) and a chip which is responsible for handling the corresponding protocol.

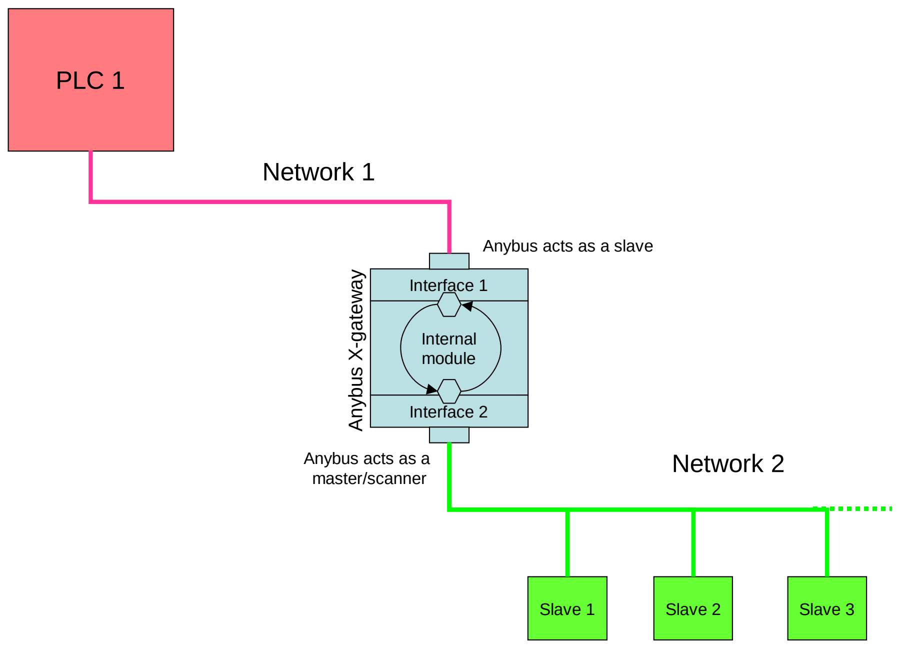
On the device or PC side, the corresponding module provides access to a dual-port RAM to its host. This enables the field device to exchange data with other network stations without knowing the details of the used network; the device “sees” only the respective RAM.

This means that also reconfigurations of the network do not require manipulations on the field device, but only on the used Anybus modules.

Networking

Anybus gateways used to connect two different field busses consist – in simplified terms – of two of the network modules as used for the Embedded Anybus products (see above) and a switching module between both interface modules. All three are combined in one single housing.

Figure 3-2



The gateway is typically configured in such a way that it appears as a master or scanner to one network and as a slave to the other network. This enables it to cyclically or acyclically receive data from a number of slaves located on one network; it then transfers these data to the master on the other network. For details on the principle of operation, please refer to chapter 4.1.

Supported bus systems

The busses which can be connected to one another or to individual terminals via Anybus include:

- PROFIBUS
- AS-Interface
- CANopen
- DeviceNet
- ControlNet
- LonWorks
- Modbus
- FIPIO
- Interbus and CC-Link

- Industrial Ethernet variants:
  - TCP/IP
  - Profinet IO
  - Modbus-TCP
  - EtherNet/IP protocol

These different networks can be connected to one another by Anybus products in multiple ways; the modular design of the Anybus products currently allows (winter 2006/07), for instance, approx. 170 different gateway variants.

4 Functional Mechanisms of this Application

You are provided with information on…

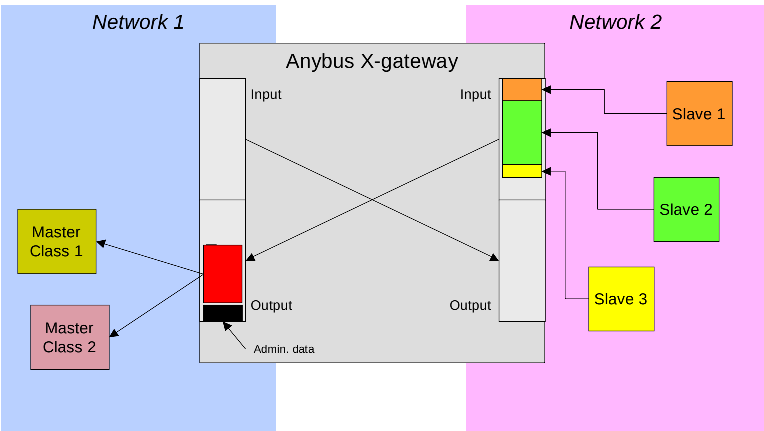
the functionalities of the used components and how their cooperation has been realized.

4.1 Anybus X-gateway functionality

The Anybus X-gateway features two interfaces which can communicate with two different networks independently of one another.

For this purpose, each interface is equipped with one input and one output buffer, which both have a gross capacity of 256 bytes.[8] In simplified terms, the input buffer of one interface is transferred to the output buffer of the other interface. The allocation of the individual areas in the buffers to the communication partners is – aside from the correct setting of the network parameters – the actual task of the gateway configuration.

Figure 4-1



The above representation schematically shows the mapping of the memory areas to one another:[9] The allocation of the memory areas between an input and the associated output is not defined. For example, the buffer area of the gateway presents itself as one single continuous memory area to the master in network 1. The copying process of the data of the input buffer located at interface 2 is not transparent to the clients on network 1, i.e., the master communicates only directly with the gateway, but it is not informed on the way the gateway used to collect the data of the slaves from network 2. The same applies analogously to the reverse data path.

[8] Since administrative data such as a “LiveList” with the data of active slaves can be stored in the buffers, the usable data buffer is normally smaller.
[9] For greater clarity, only the connection of the inputs of interface 2 to the outputs of interface 1 is shown. The reverse channel is designed analogously.

On the PROFIBUS side, the Anybus X-gateway implements the complete PROFIBUS-DPV1 functionality and can operate a Class 2 master in addition to a Class 1 master.

Anybus Gateway for DeviceNet ID Number: 23902276

Structure, Configuration and Operation of the Application

Contents

This part takes you step by step through structure, important configuration steps, startup and operation of the application.

5 Installation and Startup

You are provided with information on…

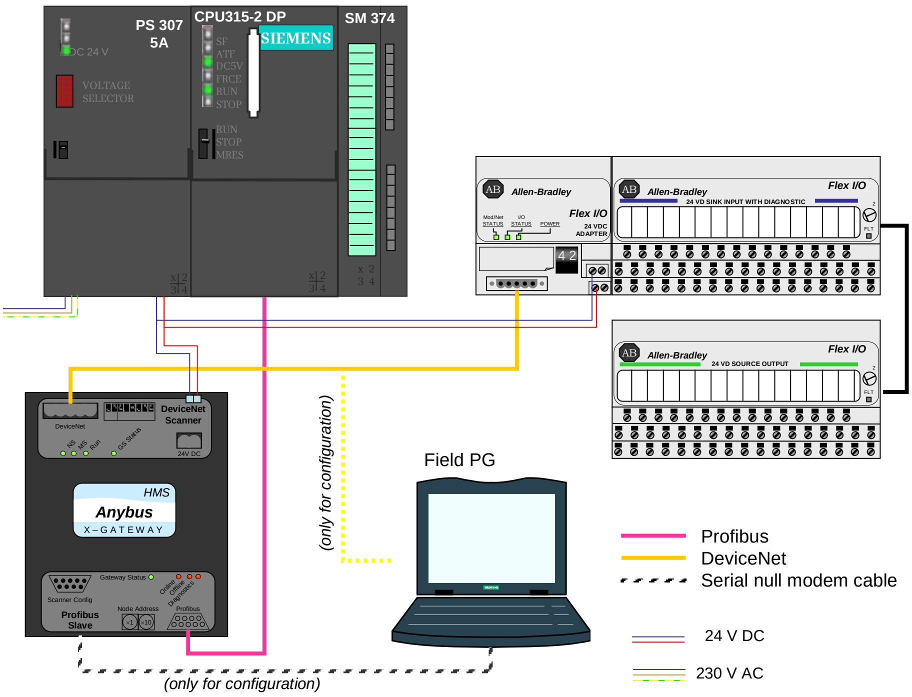
the hardware and software you have to install and the steps necessary to start up the example.

5.1 Installation of hardware and software

This chapter describes which hardware and software components have to be installed. The descriptions and manuals as well as delivery information included in the delivery of the respective products should be observed in any case.

Installation of the hardware

Figure 5-1



Note This is the hardware configuration for the **operation** of the gateway. A serial null modem cable is additionally required for the **configuration** of the gateway, please compare with chapter 6.4!

The required hardware components are listed in chapter 2.3, table 2-1. For the hardware configuration, please follow the instructions listed in the table below:

Table 5-1

No.	Instruction	Comment
1.	Mount the SIMATIC components (power supply, CPU, I/O module) on your rack.	CPU and I/O module have to be connected by a backplane bus connector.
2.	Connect the Flex I/O backplane modules to each other and to the head module.	
3.	Mount the Flex I/O input/output modules on the backplane modules.	Make sure that the keyswitch of the backplane module for both Flex I/O modules is in the “2” position.
4.	Attach the Flex I/O modules with their backplane modules on a suitable DIN rail.	
5.	Mount the Anybus X-gateway also on a DIN rail.	
6.	Use a PROFIBUS cable to connect the PROFIBUS interface of the CPU to the “PROFIBUS Slave” interface of the Anybus X-gateway on the *bottom* side of the housing.	Ensure that the terminating resistors of the PROFIBUS cable are set correctly.
7.	Connect the DeviceNet interface of the Flex I/O head module to the DeviceNet scanner interface of the Anybus X-gateway on the *top* side of the housing.	Make sure that the DeviceNet cable is terminated correctly. For information on the preparation and interface assignment of a DeviceNet cable, please refer to /6/.
8.	Supply both the Anybus X-gateway and the Allen-Bradley head module with a 24 V direct voltage source.	The PS 307 can be used as a source. Ensure the correct connection to the voltage source.
9.	Supply the Flex I/O modules with 24 V direct voltage and wire the connections correspondingly. (These connections are *not* shown in figure 5-1)	For details, consult the Allen-Bradley installation instructions, compare /2/ and /3/.

No.	Instruction	Comment
10.	For the phase of the configuration, use a serial null modem cable[10] to connect the “Scanner Config” interface to the serial interface of your PG.	
11.	To configure the SIMATIC CPU, use an MPI cable to connect your PG to the MPI on the CPU. Alternatively, you can also configure the SIMATIC CPU by connecting the PG to the PROFIBUS network .	
12.	During the configuration of the DeviceNet interface of the Anybus X-gateway connect the DeviceNet interface of the NetTOOL adapter to DeviceNet.	
13.	After completing the configuration, the connections made in step 10 to 12 can be disconnected.	

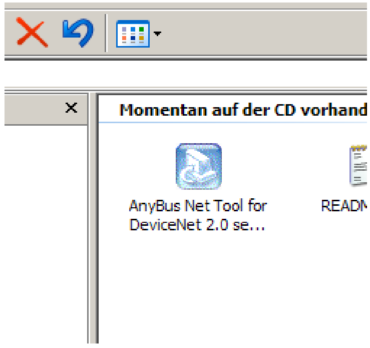
Note In general, the installation guidelines of all components have to be observed.

Note For details on the installation of the non-Siemens components, also observe the Allen-Bradley or HMS manuals! (Compare /2/, /3/ and /4/ to \9\)

[10] Included in the delivery of the gateway

Installation of the standard software

Table 5-2

No.	Instruction	Comment
1.	Install STEP7 on your PG.	Follow the instructions of the installation program.
2.	Install the RSNetWorx configuration software for DeviceNet. Make sure that a current RSLinx version is set up during the installation.	
3.	Install “AnyBus Net Tool for DeviceNet 3.0” from the CD included in the delivery of your DeviceNet adapter. (Observe the following note!) Double-click the Setup icon (shown on the right) and follow the installation instructions.	

Note The demo version of the AnyBus configuration software that can be downloaded from the web free of charge does *not* satisfy the requirements of our application since it is limited to a maximum of two network stations.
In addition, versions older than 3.0 of the configuration software cannot be operated with the EDS files provided by Allen-Bradley for the Flex I/O adapters.

Note For the basic configuration of the Anybus gateway, you additionally require a terminal program which is included, for example, in the delivery of Windows in the form of `HyperTerminal`.

However, you can also use other terminal programs for this purpose.

6 Configuration

You are provided with information on…

how the DeviceNet-PROFIBUS gateway has to be configured and the essential configuration steps that are necessary to establish a communication between an S7-300 CPU on PROFIBUS and the DeviceNet nodes.

Note The following tables only show you the basic steps necessary to establish a communication between the two networks. In any case, please observe the instructions of the Anybus components, which provide detailed and exact information and which may also consider future changes in the functional scope of the gateway.

The following configuration steps have to be performed:

Table 6-1

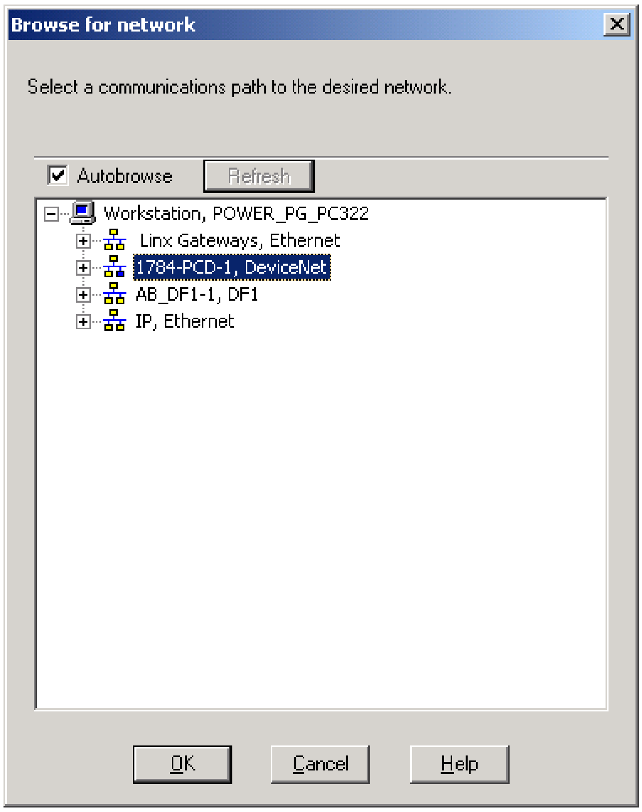
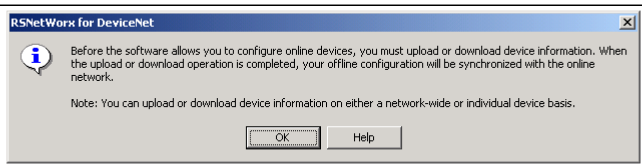
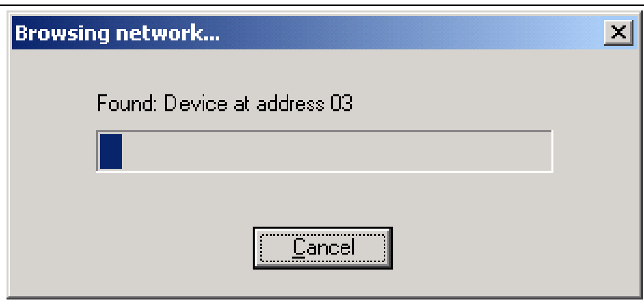
	Step	Chapter
1.	Configuration of the 1794 ADN Flex I/O adapter for DeviceNet	6.1
2.	Inserting the Anybus X-gateway into the SIMATIC hardware catalogs	6.2.1
3.	Configuration of the PROFIBUS network	6.2.2
4.	Configuration of the PROFIBUS communication between S7-300 CPU and Anybus X-gateway	6.3
5.	Hardware configuration of the Anybus X-gateway	6.4.1
6.	Basic configuration of the Anybus X-gateway	6.4.2
7.	Configuration of the Anybus X-gateway for the communication on DeviceNet	6.4.3

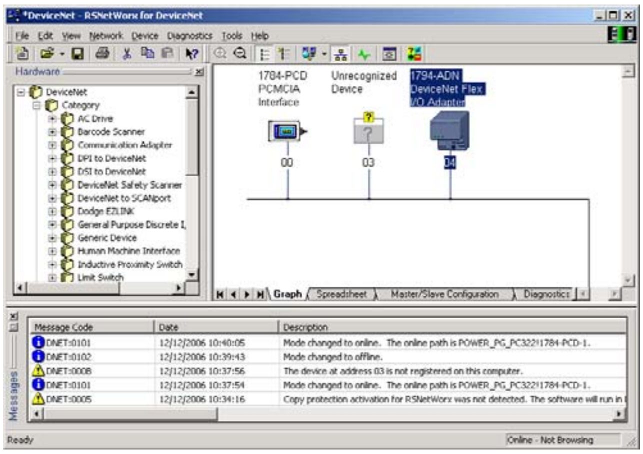
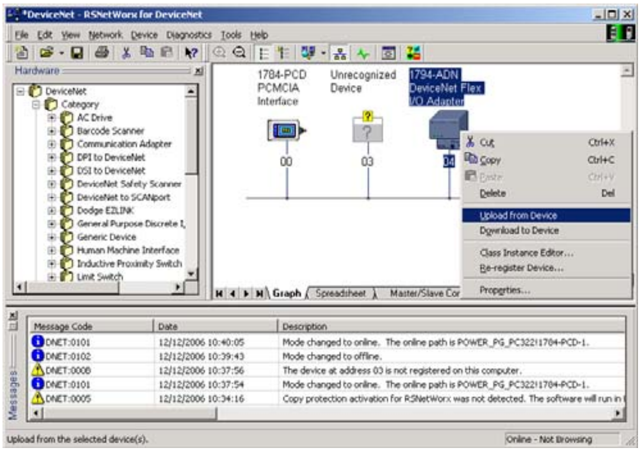
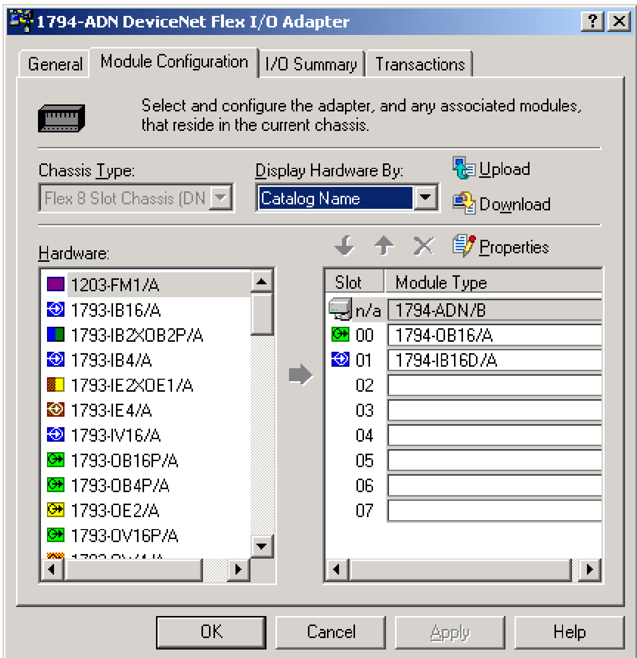
We assume that the hardware configuration and software installation have already been performed according to chapter 5.

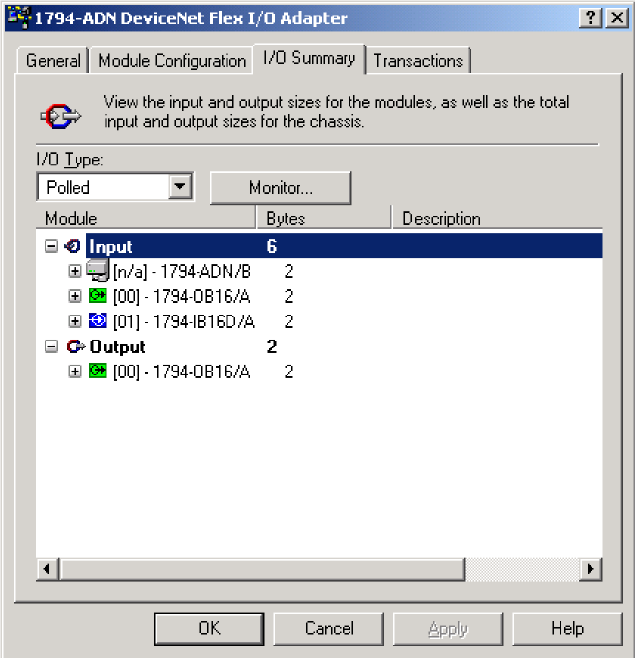
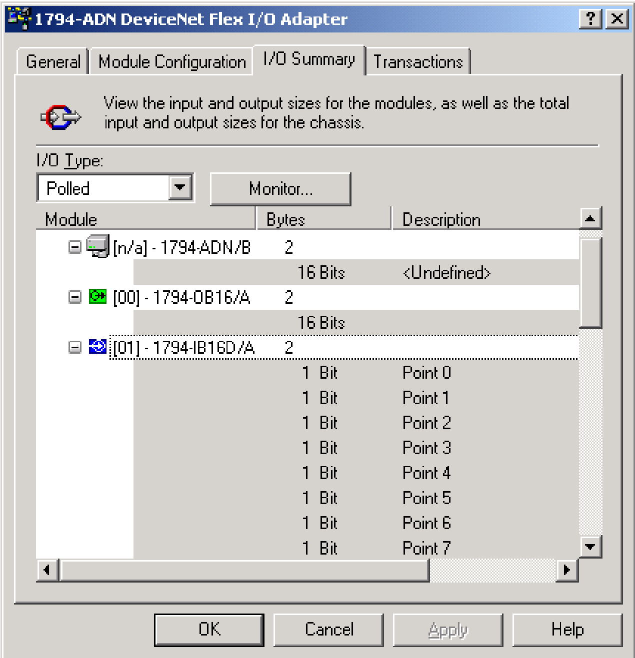
6.1 Configuration of the 1794 ADN Flex I/O adapter

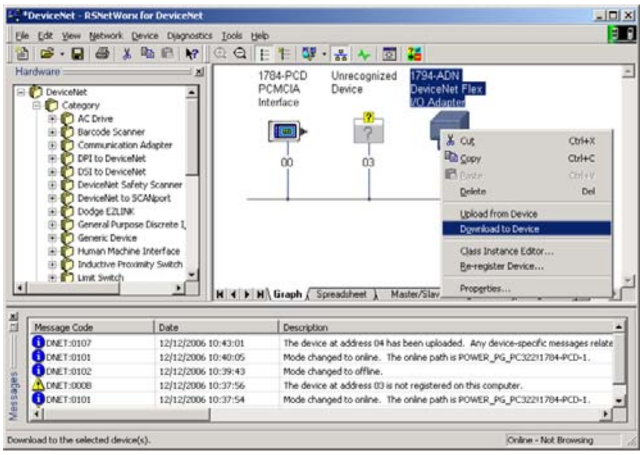
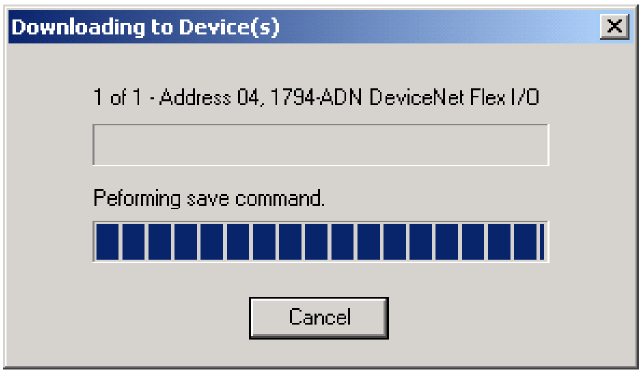
After mounting the I/O modules to the head module, the head module independently *detects* the added input/output modules, but the *activation* has to be performed in a separate step. The table below shows the steps required to determine the performed configuration and to activate it.

Table 6-2

<table>
<tr><th>No.</th><th>Instruction</th><th>Comment</th></tr>
<tr><td>1.</td><td>Use the 1784-PCD-1 adapter (this is not the Anybus NetTOOl adapter!) to connect the PCMCIA interface of your PG to DeviceNet.
Then start the RSNetWorx for DeviceNet application.</td><td>For this process also observe the RSNetWorx software manuals and comply with the usual procedures.</td></tr>
<tr><td>2.</td><td>Press the “<code>F10</code>” key or select the “<code>Online</code>” main menu command.
A dialog box is displayed with which you select the interface via which the connection to DeviceNet is to be established.
Select the 1784-PDC-1 DeviceNet interface card.</td><td>
</td></tr>
<tr><td>3.</td><td>You are informed that the configuration data have to be synchronized later.
Confirm the query with “<code>OK</code>”.</td><td>
</td></tr>
<tr><td>4.</td><td>The NetWorx software starts browsing DeviceNet for accessible nodes.
This process takes several moments during which the progress bar keeps you informed.</td><td>
</td></tr>
</table>

No.	Instruction	Comment
5.	After completing the network scan, the NetWorx main screen appears. All determined bus nodes are displayed in the top right corner, including the Flex I/O head module. The actual I/O modules are *not* displayed since they are not independent bus nodes.	
6.	Select the head module, use the right mouse button to open the context menu and select the "`Upload from Device`" command to load the configuration data of the head module to the PC.	
7.	After the data upload has been completed, the Properties dialog box shown on the right is displayed. Select the "`Module Configuration`" tab. Aside from a hardware catalog of known modules (left list), an overview of the modules connected to the head module is displayed (right list). The input and output modules connected to the head module are displayed in addition to the actual head module. In our case this tab is only for information.	

No.	Instruction	Comment
8.	Select the "`I/O Summary`" tab. The number of input and output bytes generated or required by the individual modules connected to the head module is displayed here.	
9.	Open the "+" signs in front of the module icons to display detailed configuration information. In contrast to the first two modules in the right screen shot which only exchange configuration data, the elements designated with "Point..." are actual I/O data. The data are exchanged between head module and Anybus gateway in this volume and with this structure, i.e., they require this information to be able to reconstruct the data structure on the PROFIBUS side!	

No.	Instruction	Comment
10.	Close the dialog box and again save the configuration (even if you have not changed it!) on the head module by selecting the "`Download to Device`" command in the context menu of the module.	
11.	The updated data are written to the head module.	
12.	You can now close the RSNetWorx software.	

The module now uses the correct information for the communication with its I/O modules.

Note Repeat this process each time the hardware configuration of the Flex I/O modules changes. Otherwise, the communication between the network segments cannot occur.

Note In the standard configuration – one 1794-OB16 output module and one 1794-IB16D input module with diagnostic function each – for this document, the head module provides

- 6 input bytes and
- 2 output bytes

after the configuration process.
In the input byte range, the user data are located in the **4th** and **5th** transferred byte.

6.2 Inserting the Anybus X-gateway into the PROFIBUS configuration

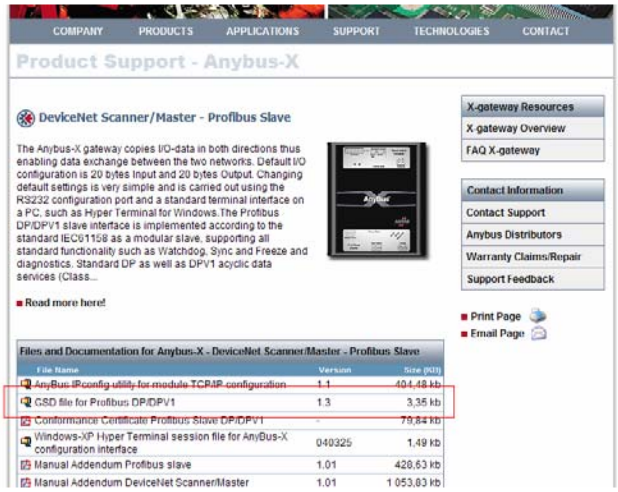
To enable the communication between the S7-300 CPU and the Anybus X-gateway, a PROFIBUS network has to be configured in which the CPU acts as a master and the Anybus gateway as a slave.[11]

6.2.1 Installation of the GSD file

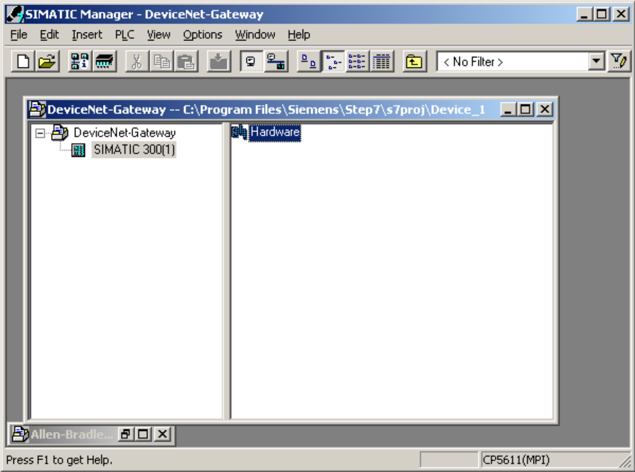
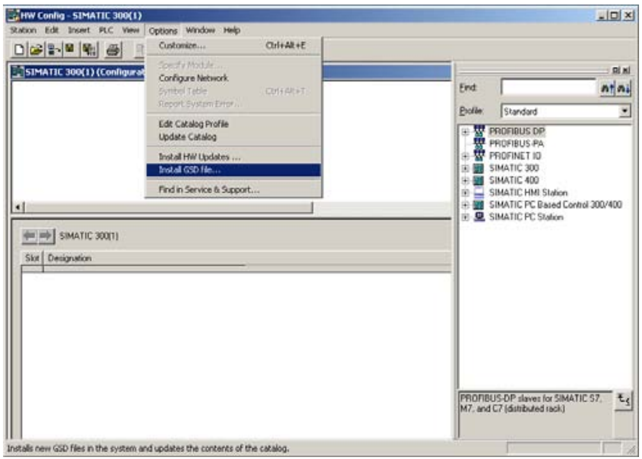
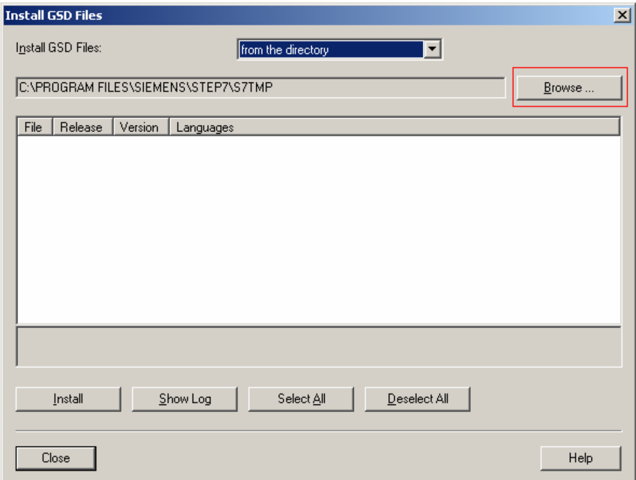
Introduce the Anybus X-gateway to the S7 Manager by inserting the GSD file into the hardware manager.

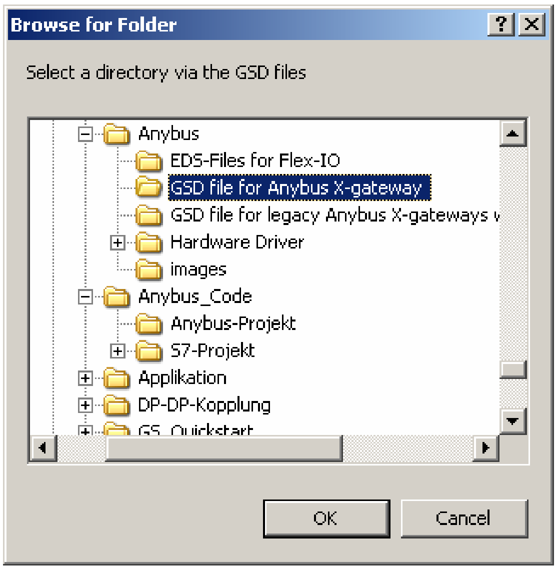
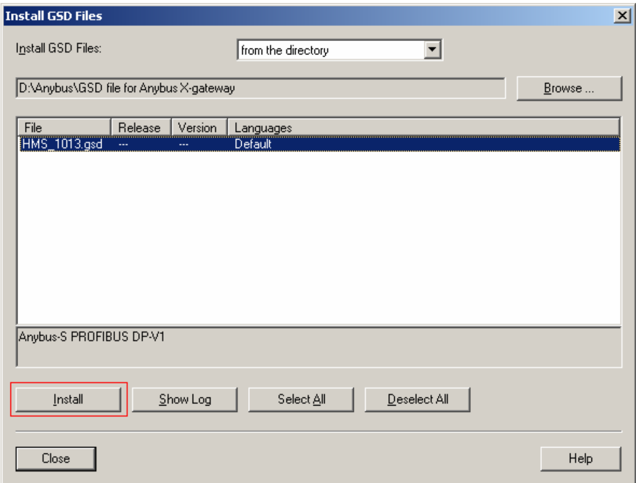
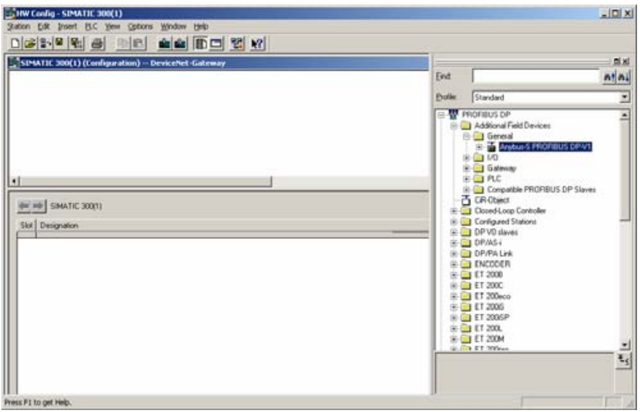
GSD files include configuration data which enable a cross-vendor configuration and the data exchange between the modules of different manufacturers.

Table 6-3

No.	Instruction	Comment
1.	For this purpose, visit the HMS support download pages which provide the GSD file for the gateway. (\12\)	
2.	Download the file, use a suitable tool (e.g. `Winzip`) to extract it and save the GSD file to a location of your choice.	

[11] Basically, the reverse configuration would also be possible with another Gateway variant.

No.	Instruction	Comment
3.	Start the S7 Manager and create a new project with an S7-300 CPU or open an existing project. Open the hardware configuration by double-clicking the corresponding icon.	
4.	In HW Config, select the "`Options` → `Install GSD file...`" command from the main menu.	
5.	In the opening dialog box, select the "`Browse...`" command and navigate to the directory in which you have saved the GSD file.	

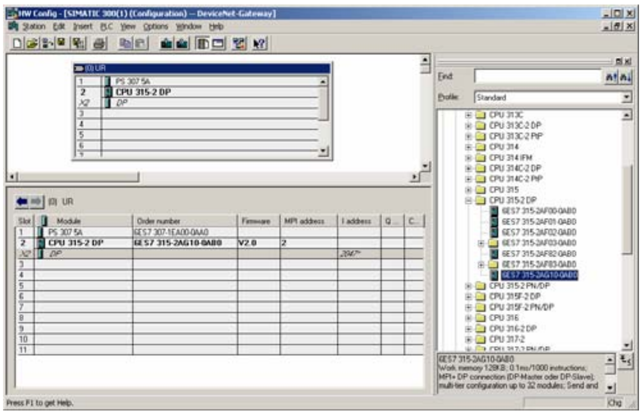
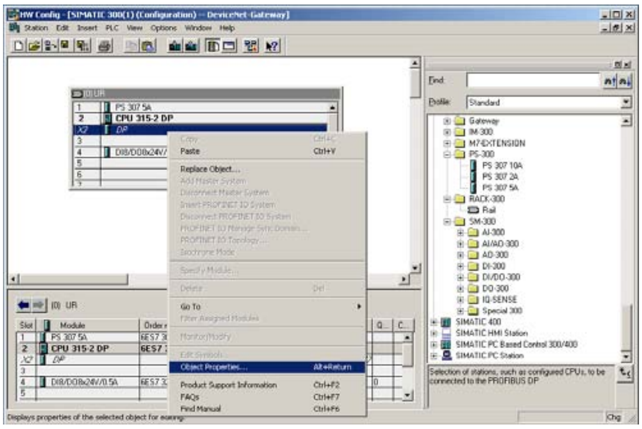
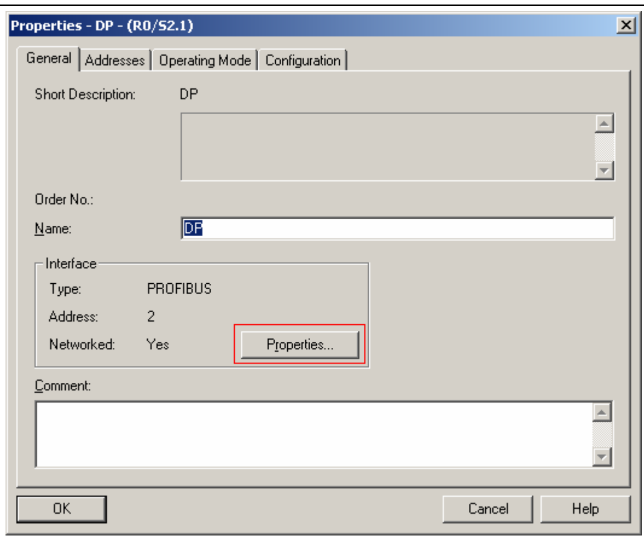
No.	Instruction	Comment
6.	Click "OK". (Your directory tree will probably differ from the shown example.)	
7.	The opening dialog box displays all GSD files which were found in the corresponding directory. Make sure that "from the directory" is selected in the combo box at the top. In the table, select the GSD file of the gateway and select the "Install" command.	
8.	The GSD file is inserted and the gateway is available in the catalog (right window pane) in the "Additional Field Devices" directory after a short period of time.	

The PROFIBUS Slave interface of the gateway can now be applied to the hardware configuration of your projects.

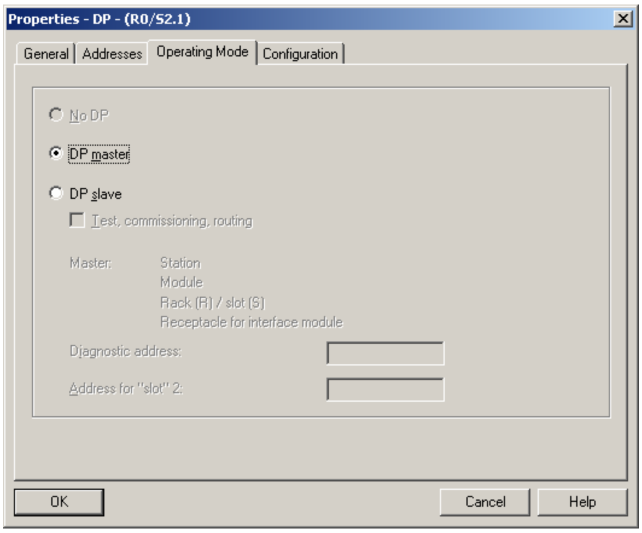
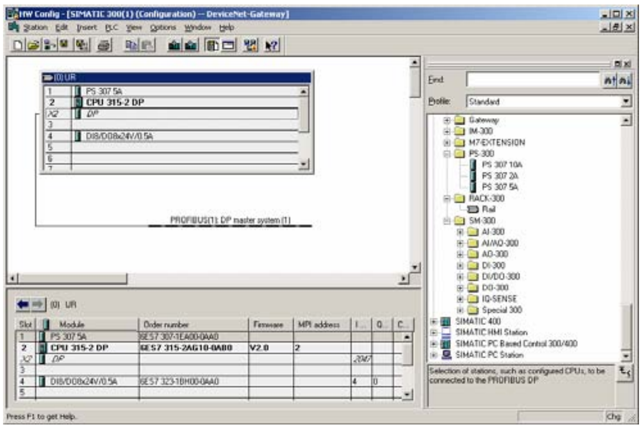
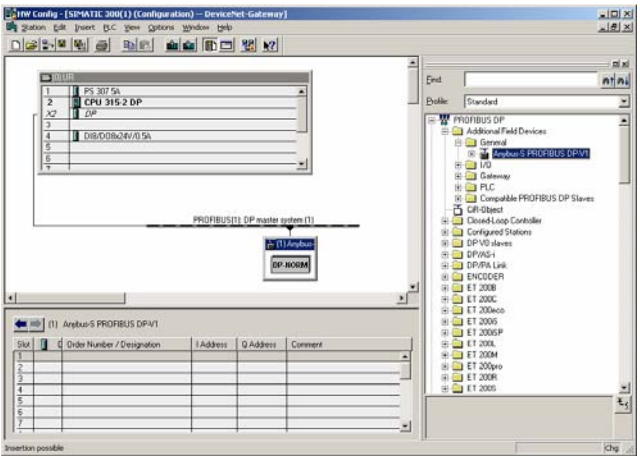
6.2.2 Configuration of the PROFIBUS network

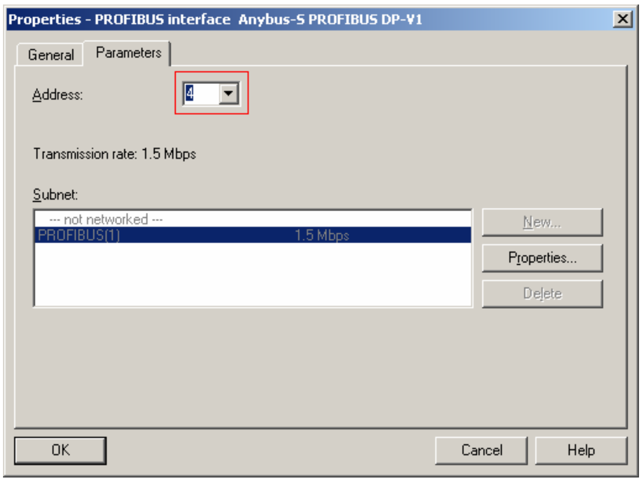
In the next step you have to configure a PROFIBUS network. In this step you define the design of the bus, the network stations and other boundary conditions.

Table 6-4

No.	Instruction	Comment
1.	Create your project in Step 7 and in HW Config, insert the SIMATIC components into the rack according to your configuration.	
2.	Select the DP interface entry in the rack, open the context menu and use the “`Edit → Object Properties...`” command.	
3.	In the Properties dialog box which now opens, enter the corresponding network parameters. Use the “`Properties...`” button in the “`Interface`” segment to configure the PROFIBUS communication. If required, consult the PROFIBUS documentation.	

Anybus Gateway for DeviceNet ID Number: 23902276

No.	Instruction	Comment
4.	In the “Operating Mode” tab of the Properties dialog box, ensure that the CPU acts as a “Master” on the bus.	
5.	Close the dialog boxes with “OK” and the PROFIBUS segment is displayed in HW Config as part of the DP master system.	
6.	In the hardware catalog, select the Anybus module whose GSD file you have installed in the previous section and use drag & drop to move the icon to the PROFIBUS segment in the Hardware view. The gateway is inserted into PROFIBUS under the name “DP-NORM”.	

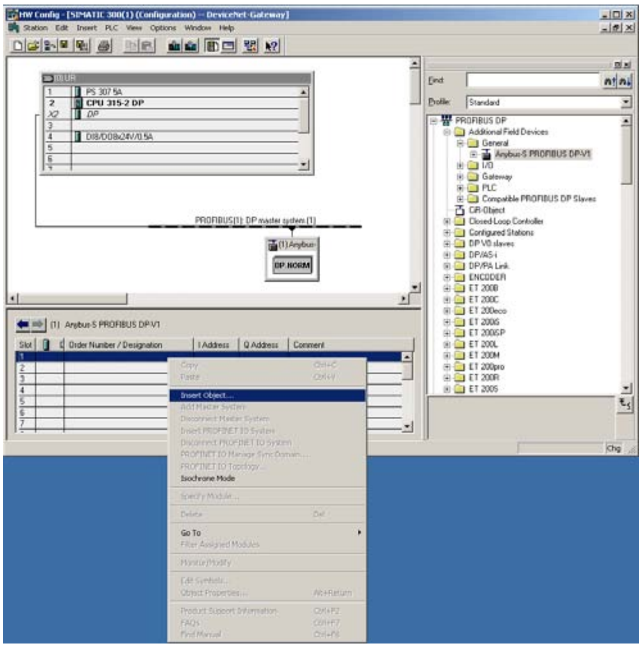
No.	Instruction	Comment
7.	Parameterize the Anybus gateway consistently with the master station by opening the Object Properties dialog box from the context menu with the right mouse button. Ensure that particularly the setting of the PROFIBUS address for the gateway is correct. (See also chapter 6.4.1)	

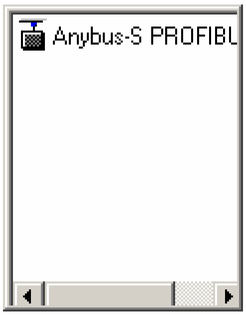
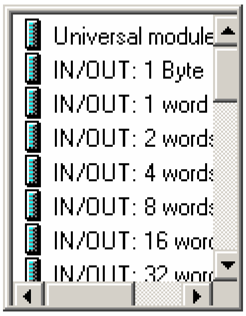
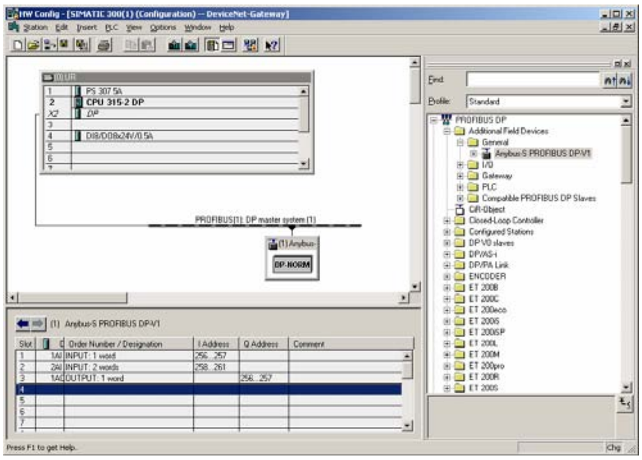
As a PROFIBUS slave, the Anybus X-gateway is now part of the PROFIBUS communication which is controlled by the S7-300 CPU in the role of the master.

6.3 Configuration of the PROFIBUS communication

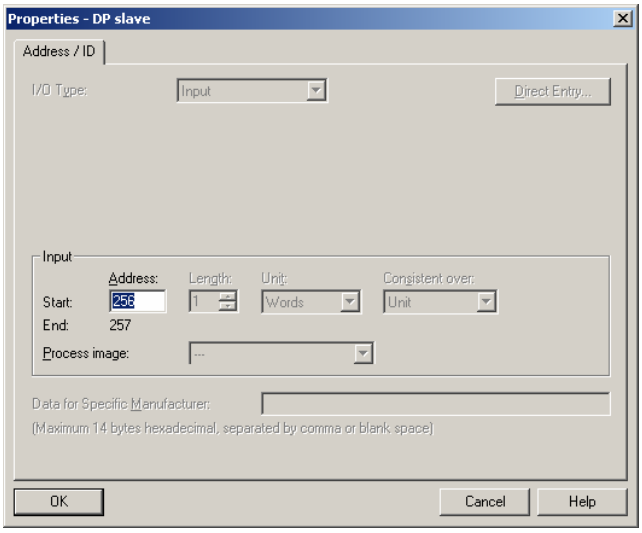
The next configuration step is to configure the data modules exchanged between master and slave.

Table 6-5

No.	Instruction	Comment
1.	Open HW Config of the PROFIBUS network created in chapter 6.2.2. Select the Anybus X-gateway ("DP-NORM") icon, select a blank line in the "rack" displayed in the lower part of the window and open the context menu with the right mouse button. Select the "`Insert Object...`" command	

No.	Instruction	Comment
2.	A box with the Anybus gateway as the only element is displayed. Select the entry.	
3.	The now opening box displays all I/O modules via which the gateway can communicate with the CPU. Successively select the input and output modules as described in table 6-2 and in the note on page 31.[12] Remark: The configured modules must not exceed the size of the buffers provided by the gateway (compare chapter 6.4.2); otherwise, communication errors occur.	
4.	Select the entries of the modules and use the right mouse button to open the context menu in which you execute the “`Object Properties...`” command.	

[12]Observe that the master’s view is shown to advantage despite the configuration in the slave. This means that “input” modules are mapped to the inputs of the master, ”output” modules are mapped to its outputs.

No.	Instruction	Comment
5.	In the opening Properties dialog box, you can enter the address at which the received data or the data to be sent are stored. In the example, the slave values are written to the input bytes 256 and 257.	
6.	Confirm your changes with "OK", save, compile and download your project to apply the changes.	

The CPU is now prepared for the communication with the Anybus X-gateway.

6.4 Configuration of the Anybus X-gateway

After configuring the hardware of the Anybus X-gateway (table 6-6), the software is configured in two additional steps:

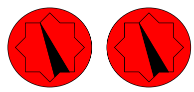
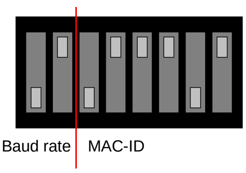
1. First, the actual gateway is configured for the network, i.e, fundamental parameters are set on the gateway by the serial cable via the configuration interface: Table 6-7.
2. Subsequently, the Anybus gateway is configured as a master or scanner in DeviceNet with the aid of the DeviceNet adapter (which is connected to the PG with a serial cable): Table 6-9.

Note Except for setting the bus address, a separate configuration of the Anybus X-gateway on the PROFIBUS side is not required since the gateway only acts as a slave on this network.

Note For the configuration, always consult the references provided by HMS (see /4/ -- /9/)!

6.4.1 Hardware configuration of the Anybus X-gateway

Table 6-6

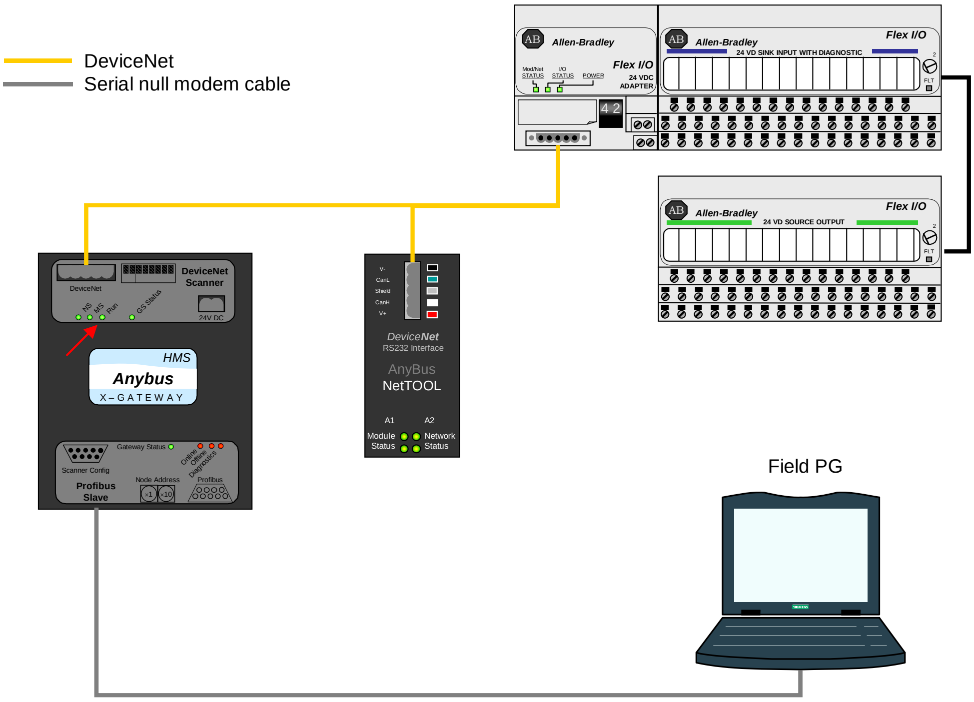
No.	Instruction	Comment
1.	Two rotary selector switches marked in red are located on the *bottom side* of the gateway with which you can set the PROFIBUS address of the gateway. Turn the two switches to the desired position with the aid of a screw driver or a similar tool; this position must correspond to the setting made in table 6-4, step 7 and must not be in conflict with the address of the CPU or other network stations. For this process, please also observe /5/ and /8/.	
2.	A set of eight DIP switches is located on the top side of the gateway. The combination of the left two DIP switches (“1”, “2”) determines the data rate on the DeviceNet bus. The slaves on the bus follow the rate specified by the master, however, possibly not all slaves are capable of following the highest rates. For this and the next step, please observe /6/ and /9/	
3.	The six right DIP switches (“3”-“8”) determine the “Mac-ID”, i.e., the address of the gateway on DeviceNet. Select an address which is not in conflict with other bus nodes.	
4.	After implementing the changes, restart the gateway.	

The Anybus X-gateway is now physically prepared for the network traffic.

6.4.2 Anybus X-gateway: Basic parameterization

Install the devices as described in chapter 5.1. Connect the field PG directly to the Anybus X-gateway as shown below to perform the basic configuration of the device:

Figure 6-1



The Anybus X-gateway must be connected to the field PG via a serial null modem cable which leads to the gateway interface designated with “Gateway Config”. Make sure that the gateway is supplied with current (24V DC). The gateway should be connected to DeviceNet.

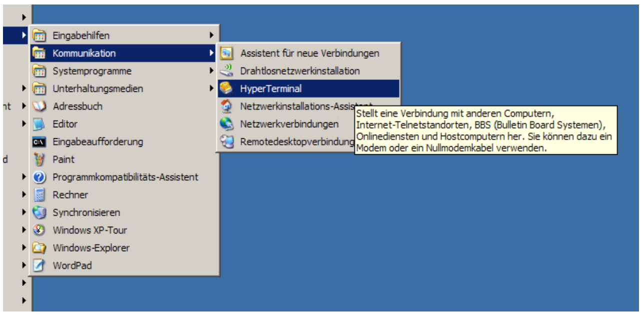
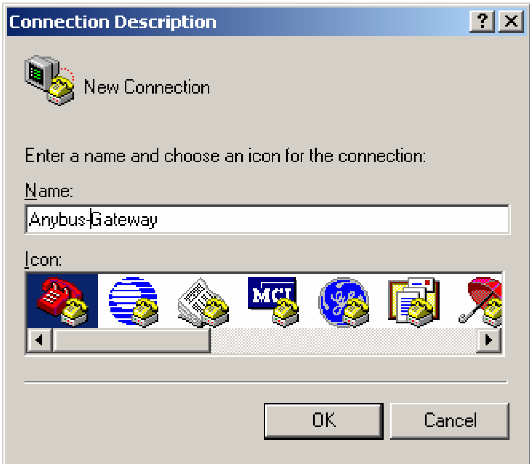
Note For these configuration steps, also consult chapter 5 of document /4/.

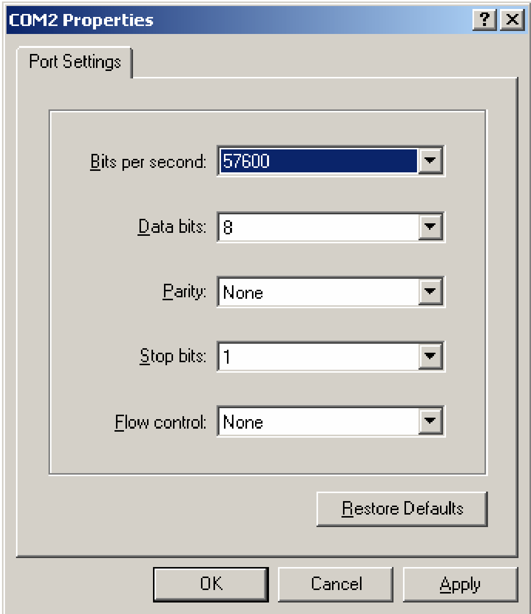
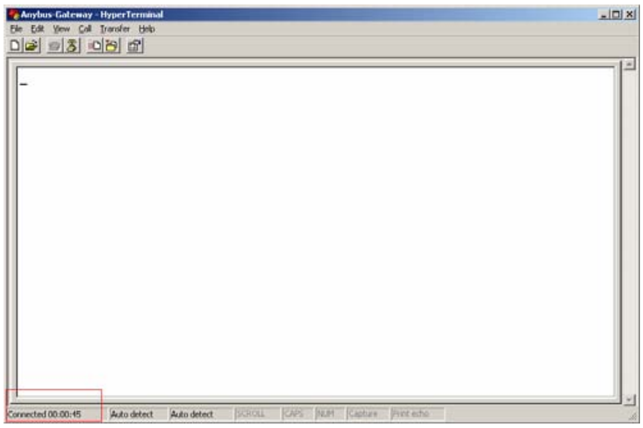
Setting the gateway to “Run” mode

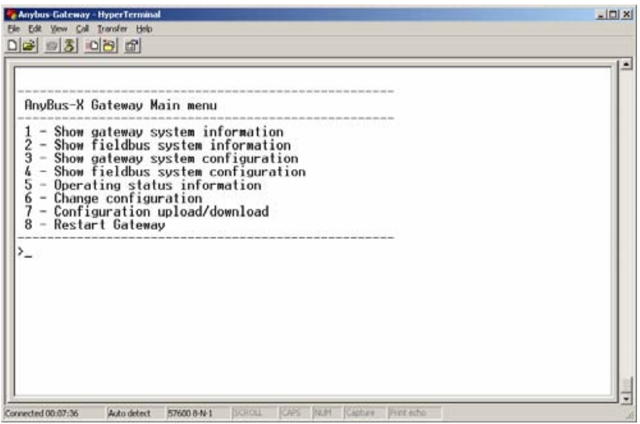
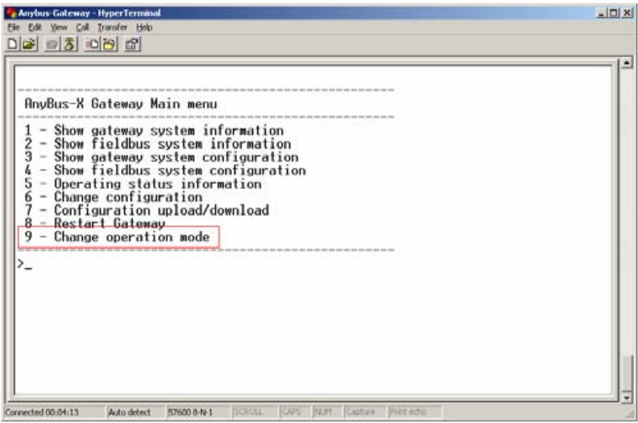
If the gateway is in “`Idle`”, there is no communication. This is indicated by the “`RUN`” LED on the DeviceNet panel of the gateway housing (see figure 6-1, red arrow) which flashes green.

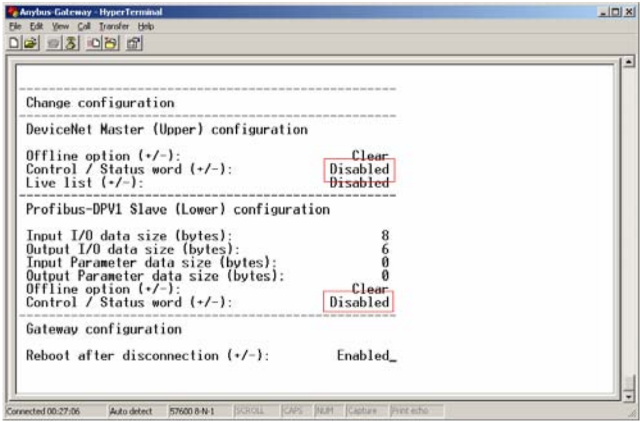
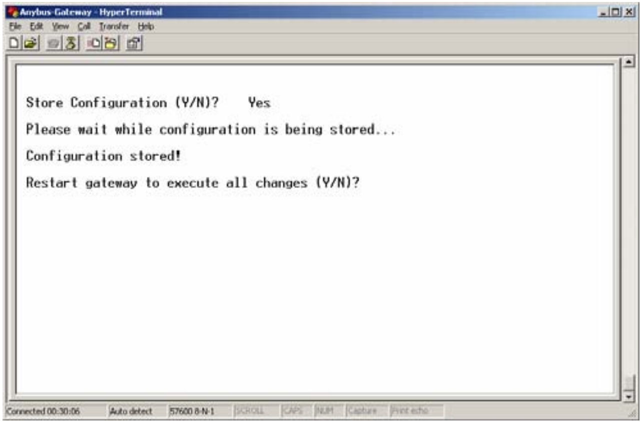
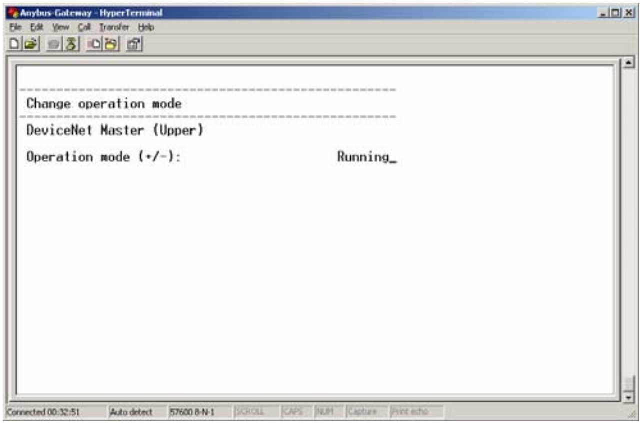
The following configuration steps set the gateway to “`Run`” mode.

Table 6-7

No.	Instruction	Comment
1.	Start the gateway by connecting it to the voltage source. The startup of the gateway takes approx. 30 to 60 seconds.	
2.	Use a null modem cable to connect a serial interface of your PG to the “Gateway-Config” interface at the bottom side of the gateway. Start a terminal program of your choice such as `HyperTerminal`. Depending on configuration and operating system, the arrangement of your start menu items may differ from the one shown here.	
3.	When prompted, enter a name for the connection and select an icon. If you have already worked with `HyperTerminal` and the Anybus gateway, you can, of course, also open an already saved connection.	

No.	Instruction	Comment
4.	Select the interface via which you are connected to the gateway.	
5.	Set the serial connection to the parameters shown on the right. The gateway only supports these values for the communication. Quit the dialog box with "`OK`".	
6.	Instantaneously, the Terminal window remains empty also if the connection has been established. If the connection has been successfully established, this is indicated by the display of "`Connected`" and the duration of the current connection in the status field in the bottom left corner of the window.	

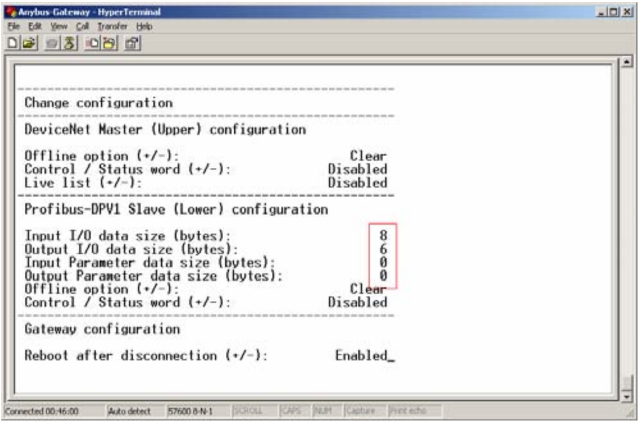
No.	Instruction	Comment
7.	You can interrupt the connection now or later by selecting the "`Call → Disconnect`" command in the main menu or the corresponding icon. The connection can always be reestablished with the "`Call → Call`" command of the main menu or the corresponding icon.	**Disconnect icon:**  **Call icon:** 
8.	After completing the startup of the gateway, press "ESC" on the PG to go to the configuration menu.	
9.	If the control of the gateway has been activated *via control/status words*, the configuration menu only displays eight entries.	
10.	If the controller is instead active *via the serial interface*, an additional 9th menu item is displayed. In this case, select "`9`" and continue with point 14.	
11.	If only 8 entries are visible, select "`6 – Change Configuration`" (i.e., simply press "`6`".)	

No.	Instruction	Comment
12.	Now allow the interruption of the network connection on request and use the `Return` key to go through the individual menu options. Use the "+" key to set the "`Control / Status word (+/-):`" entry to "*Disabled*" for *both* networks.	
13.	Save the configuration in the gateway and restart the gateway. After the restart you also see the 9th menu option "`Change operation mode`" (see point 10). Select it.	
14.	Use the "+" key to set the gateway status to "`Running`".	
15.	Have the gateway perform a restart, close the connection to the gateway with the "`Disconnect`" command and exit the `HyperTerminal` application.	

The gateway is now in Run mode and can communicate with the DeviceNet slaves when parameterized correspondingly.

Adapting the gateway buffers for the data transmission

Table 6-8

No.	Instruction	Comment
1.	Repeat steps 1 -- 8 from table 6-7.	
2.	Then select the “`6 – Change Configuration`” command by pressing the “`6`” key.	
3.	In the “`Input/Output I/O data/Parameter size (bytes)`” entries, set the size of the buffers you want to use. Please observe the notes below. The values shown in the screen shot are only examples; you obtain the correct values from the procedure performed in chapter 6.1, table 6-2. Compare with the note on page 31.	
4.	Save the configuration, restart the gateway and terminate the `HyperTerminal` connection.	

Nomenclature

The following has to be observed for the Anybus designations:

- Cyclically transmitted (fast) data are referred to as “I/O data”,
- acyclically transmitted (slow) data are referred to as “parameter data”.

Buffer sizes

The entered buffer sizes have to meet the following requirements:

- The configuration of the buffer sizes must *at least* be large enough to ensure that the configured I/O modules find space in them. In addition, it is required that the buffer provides enough space for possibly transmitted status data and LiveLists.
- All of the cyclic I/O data buffers together can comprise a *maximum* of 416 bytes; the respective inputs and outputs must not exceed 244 bytes.
- All of the acyclic data (“parameter/diagnostics”) together can have a size of 237 bytes.

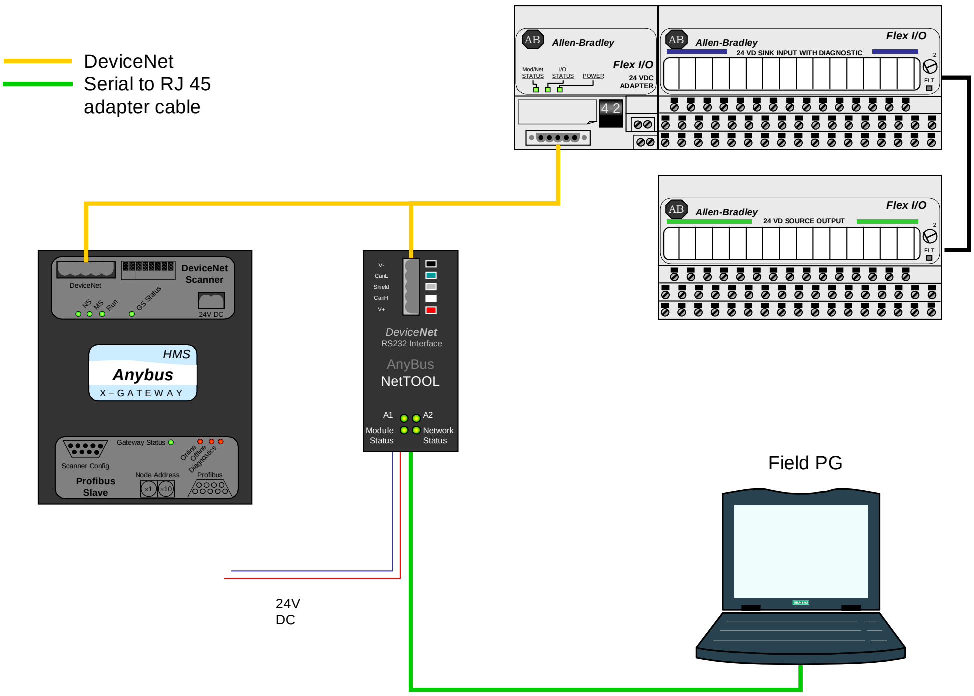
A buffer whose configured size is larger than the actually transmitted data does not cause problems.

After completing these activities, the Anybus X-gateway is prepared for DeviceNet.

6.4.3 Anybus X-gateway: Parameterization as a DeviceNet master/scanner

Install the devices as described in chapter 5.1. Connect the field PG directly to DeviceNet as shown below to configure the Anybus X-gateway:

Figure 6-2



To configure the DeviceNet interface of the gateway, connect both the gateway and the AnyBus NetTOOL adapter to DeviceNet. Make sure that both gateway and NetTOOL are supplied with 24V direct current. Then connect the serial to RJ 45 adapter cable to the serial interface of your PG and the corresponding NetTOOL interface.

Note For these configuration steps, also consult the DeviceNet Configuration Tool or NetTOOL adapter manuals (\23\ and \24\).

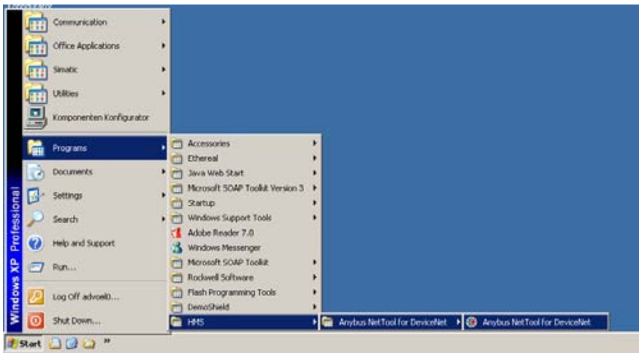
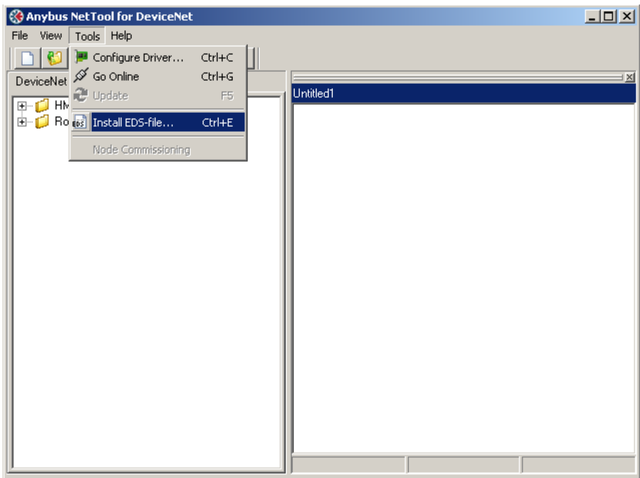
Note Please ensure that you use the latest versions of the Allen-Bradley EDS files and the latest version of the Anybus configuration software (3.0 or higher).
Older versions are normally not compatible with each other.

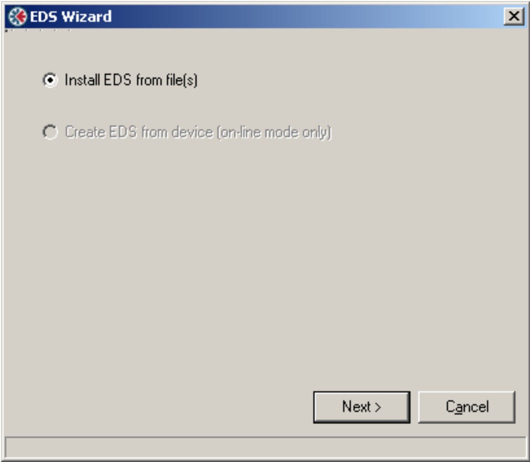
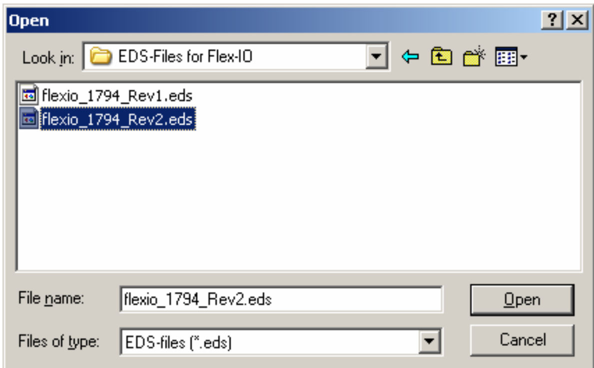
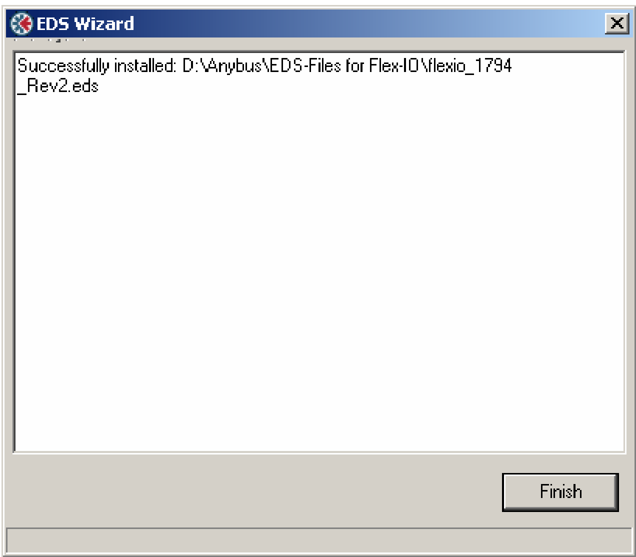
Inserting the EDS file for the Flex I/O adapter

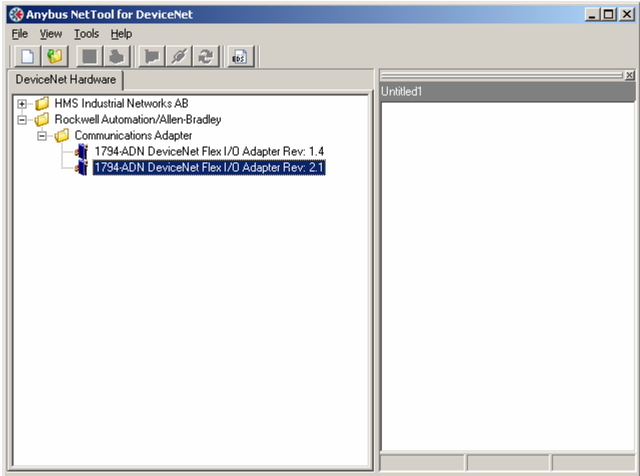
Before the Anybus X-gateway can be configured for the communication with the 1794-ADN Flex I/O adapter, the corresponding EDS file has to be entered in the database of the NetTool configuration software.

Once this has been done, the Flex I/O adapter can be used in all future projects.

Table 6-9

No.	Instruction	Comment
1.	Use your internet browser to navigate to the web site on which the EDS file for the Flex I/O adapter is offered. Download the file, extract it (if required) and save it to a directory of your choice.	Allen-Bradley provides a search screen for the download of all its EDS files on \26\. On this page, select “DeviceNet” as network, enter “1794-AND” as search word in the “Product Name” field and leave the other fields blank or keep the “Any” value. At present (winter 2006/07), the current version of the EDS file is Rev. 2.1; it can be directly downloaded at \27\.
2.	Start “Anybus NetTool for DeviceNet” that you have previously installed (see table 5-2) on your PG.	
3.	In the main menu, select “`Tools → EDS-File...`” or click the corresponding “” icon.	

No.	Instruction	Comment
4.	In the subsequently opened dialog box, select the "`Install EDS from file(s)`" option. Click the "`Next >`" button.	
5.	In the subsequently opened dialog box, navigate to the folder in which you have saved the downloaded EDS file. Select the corresponding file and select the "`Open`" button or directly double-click the entry.	
6.	If the installation was successful, a corresponding message is displayed. Click "`Finish`" to complete the installation process.	

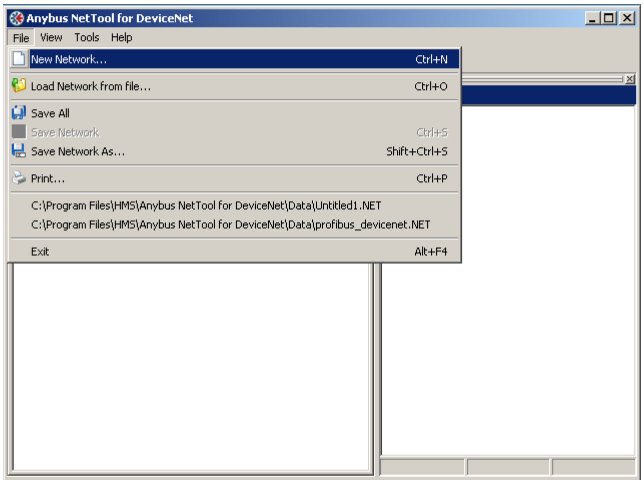
No.	Instruction	Comment
7.	In the hardware catalog in the left window pane of NetTool, you can check the successful installation by navigating to the “`Rockwell Automation/Allen-Bradley` → `Communications Adapter`” path. You will find the entry of the freshly inserted EDS file at this storage location.	

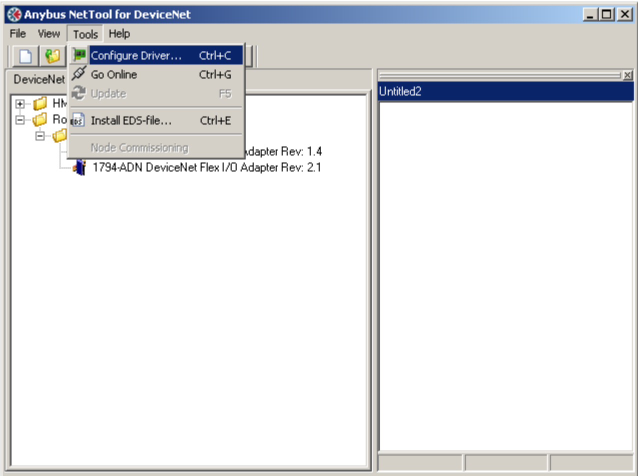
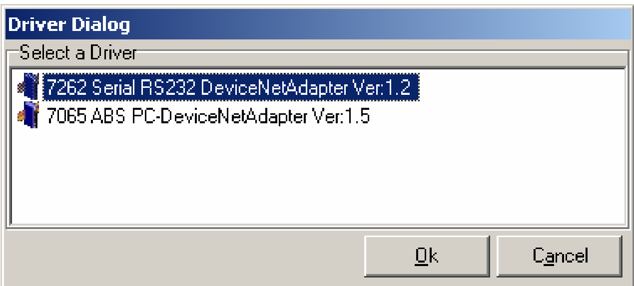
The EDS file of the Flex I/O adapter is now available for future configurations with the NetTool configuration tool.

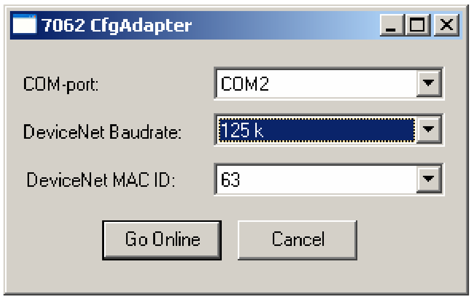
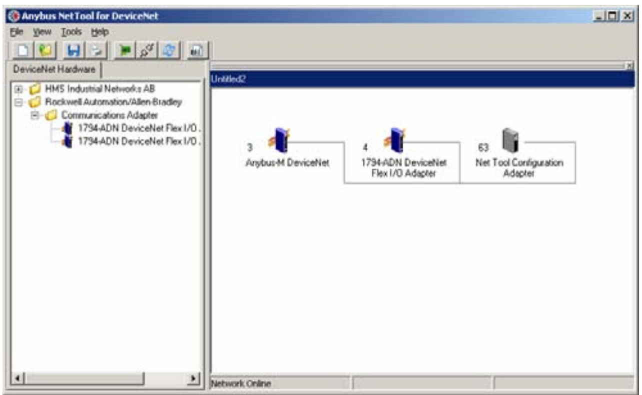
To insert other modules of third-party manufacturers into the hardware database and to use them in own projects, proceed correspondingly. Contact the hardware manufacturers for the necessary EDS files.

Configuring the data transmission

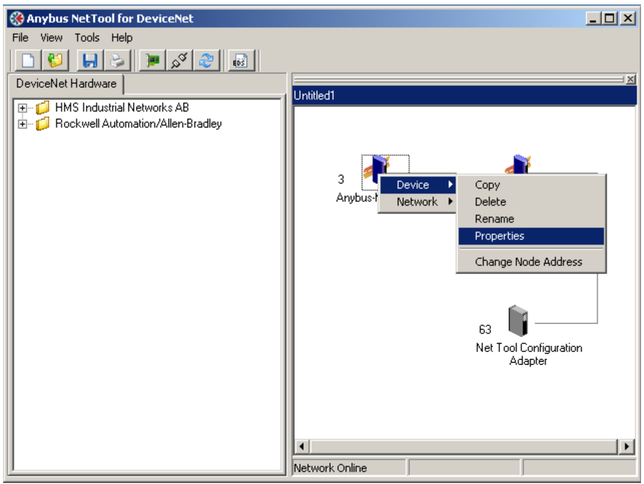
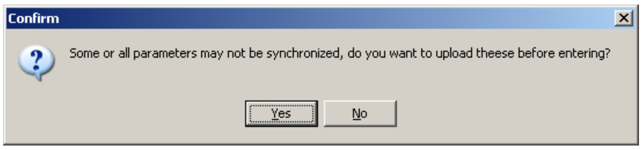
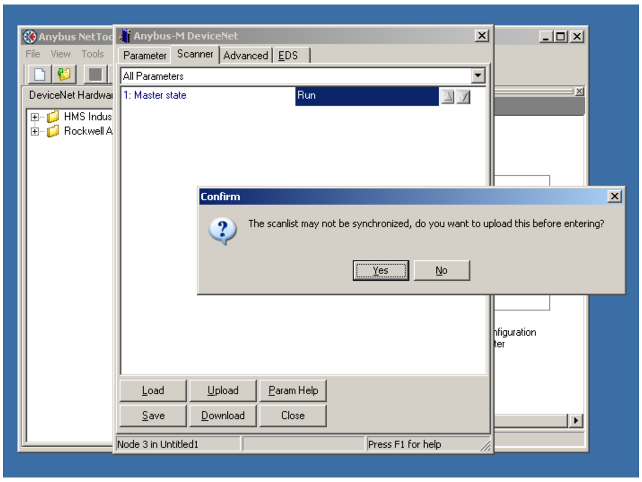
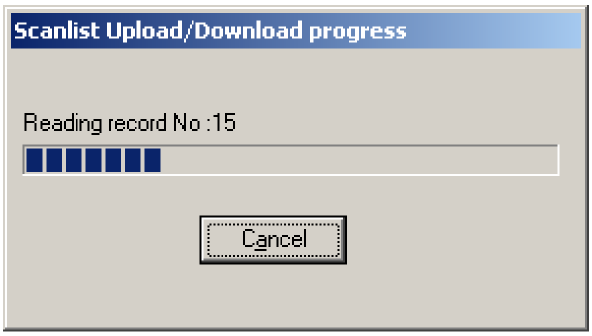
Table 6-10

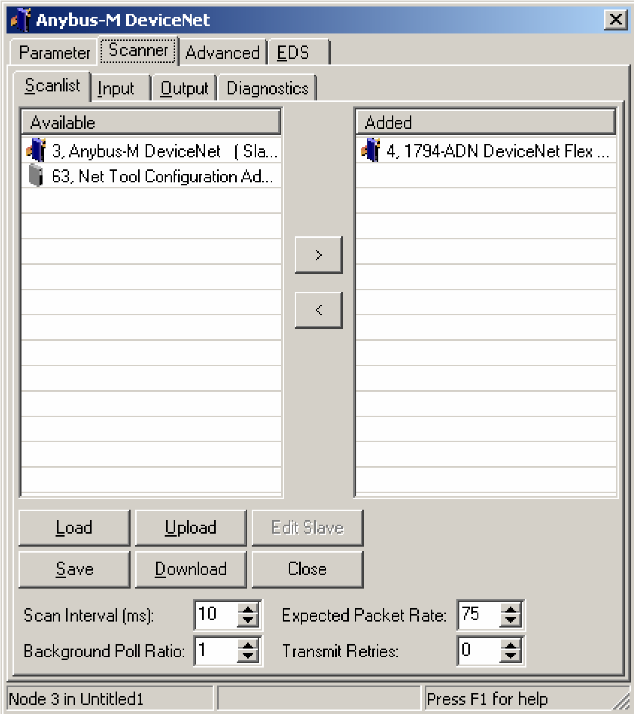
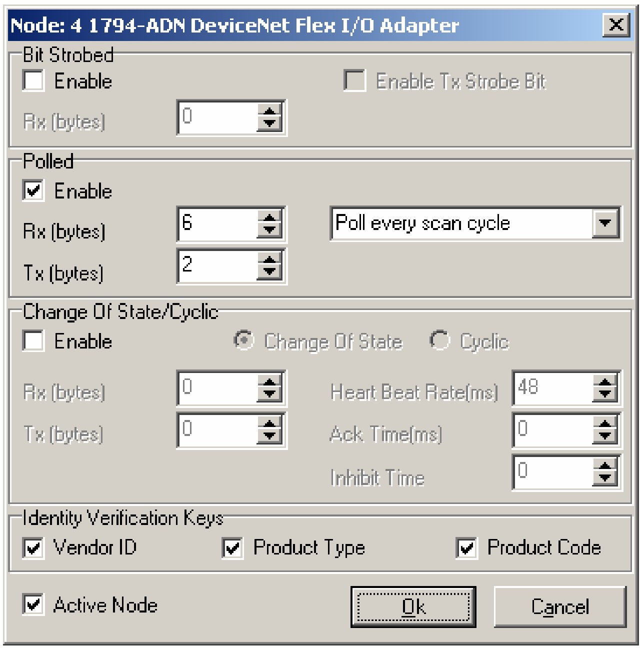
No.	Instruction	Comment
1.	Connect the PG to the NetTOOL configuration adapter as shown in figure 6-2. Use the “serial to RJ45 adapter cable” included in the delivery.	
2.	After starting the NetTool configuration software, select the “`New Network...`” command from the main menu or use “`Load Network from file...`” to open a network configured in a previous session. As an alternative to the “`New Network...`” command, you can also click the “ ” icon in the button bar.	

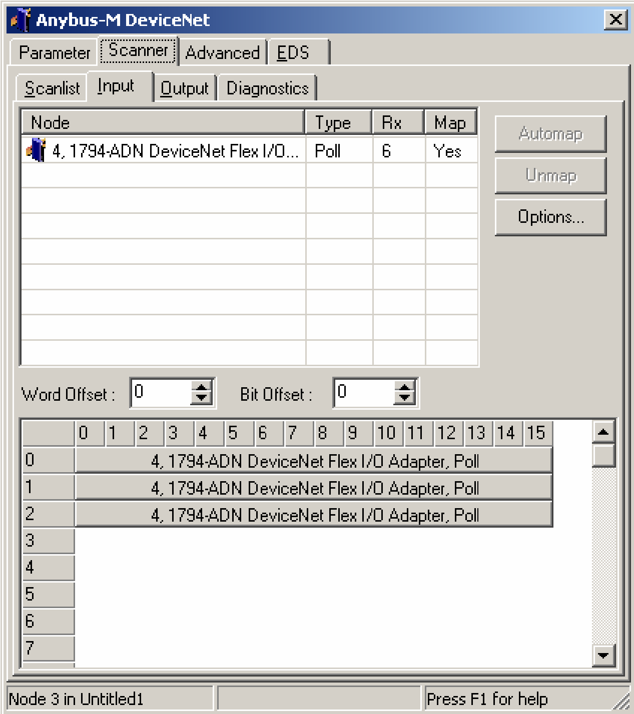
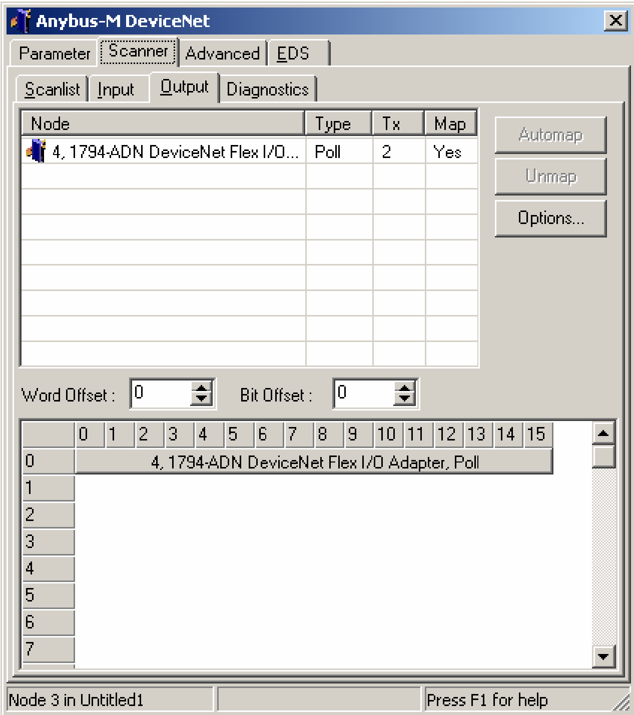
No.	Instruction	Comment
3.	If you are using NetTool for the first time or if the connection to the NetTOOL adapter has changed, use the "`Configure Driver...`" command to adapt the communication driver. Continue with step 5.	
4.	In all other cases, i.e., if your connection has not changed since the last session, use the mouse to select the entry of the "NetTOOL Configuration Adapter" in the network overview. Select the "`Go Online`" command and continue with point 7.	
5.	Select the "`7262 Serial RS232...`" entry in the opening dialog box. Click the "`Ok`" button.	

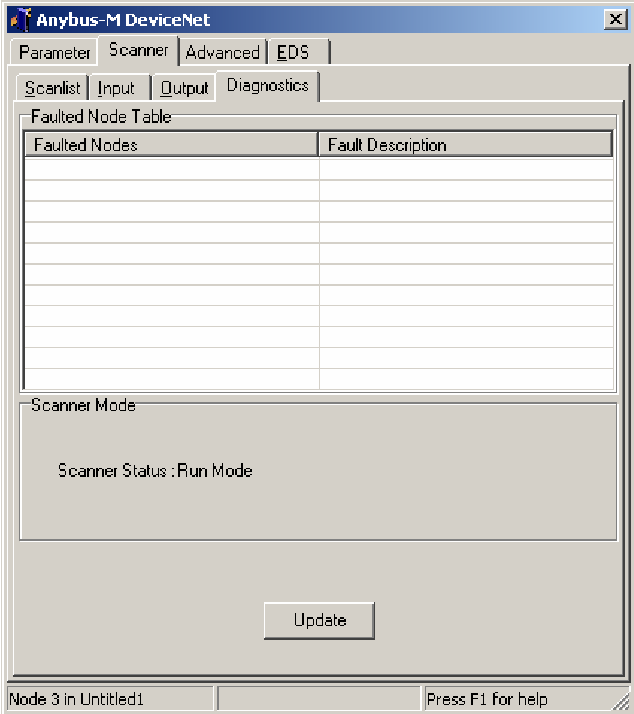
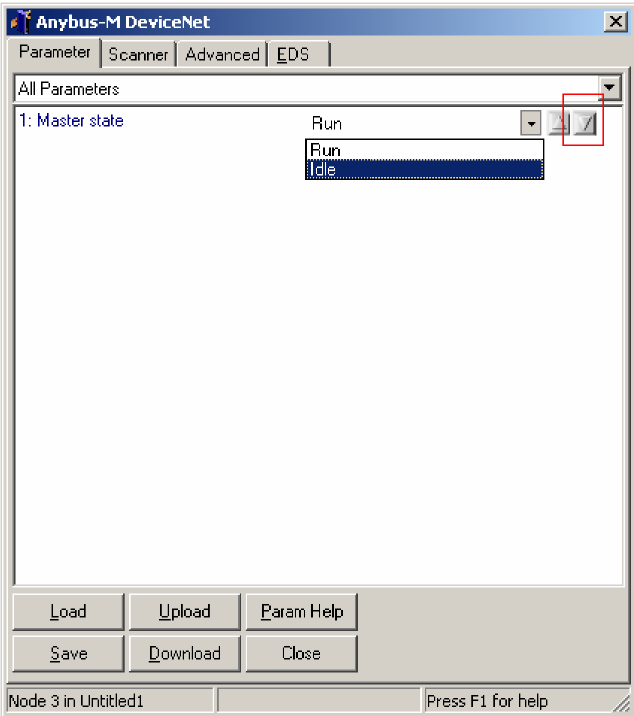
No.	Instruction	Comment
6.	In the parameterization dialog box in the "`COM-port`" box, select the interface via which you are connected to the NetTOOL adapter. In "`DeviceNet Baudrate`", select the value you have set on the DIP switches of the gateway. (See also table 6.6, step 2) In "`DeviceNet Mac ID`", assign an address under which the gateway logs on to DeviceNet. This address must not be in conflict with another bus node. Now select the "`Go Online`" button.	
7.	When the connection to DeviceNet has been established, the following dialog box is displayed. Press "`OK`" to display an overview of all detected bus nodes. When the connection has been established, this is also indicated by the flashing LED "A2" on the front of the NetTOOL adapter (compare figure 6-2) during the network update.	
8.	The right window pane of the NetTool configuration software displays the detected bus nodes. The actual hardware adapter is displayed as "`Net Tool Configuration Adapter`"; the Anybus X-gateway has the designation "`Anybus-M DeviceNet`".[13] The respective DeviceNet addresses are displayed to the left of the bus nodes.	

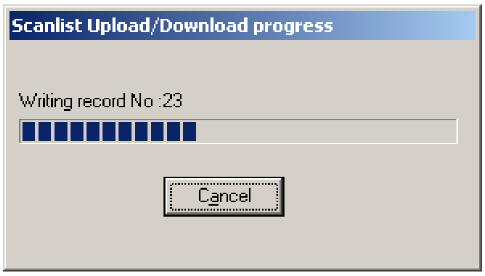
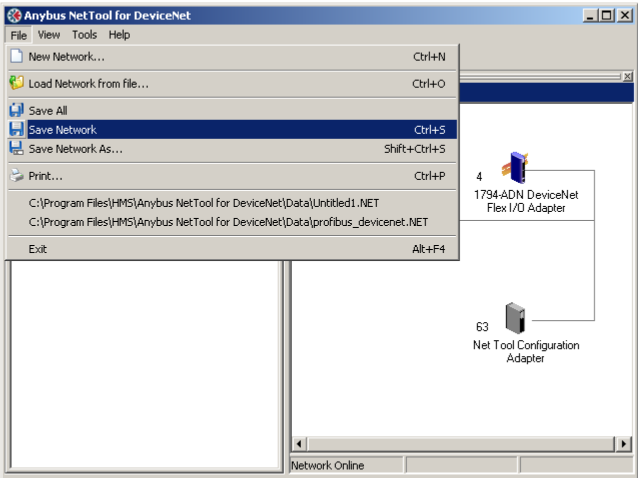
[13] "M" refers to the role as a master in DeviceNet. In the function of a slave, the designation would be "S".

No.	Instruction	Comment
9.	Select the gateway icon, open the context menu with the right mouse button and select the "`Device → Properties`" command.	
10.	A query appears asking whether you want to synchronize the data in the computer with the configuration set in the gateway. Confirm with "`Yes`" to read out the current configuration from the gateway.	
11.	The Properties dialog box of the gateway opens. Select the "`Scanner`" tab. A further query appears asking whether the current configuration data are to be taken from the gateway. Again answer with "`Yes`".	
12.	The configuration data are read from the Anybus X-gateway over several seconds.	

No.	Instruction	Comment
13.	When the upload has been completed, the "`Scanlist`" tab opens. The left "`Available`" section displays a list of accessible bus nodes, the right section shows the bus nodes with which the gateway exchanges data. ("`Added`"). The "`Added`" list is, of course, empty at the beginning of the configuration. To add a bus node to the scanlist, select it in the left list and press the " > " button. To remove a bus node from the scanlist, select it in the right list and press the " < " button.	
14.	In the "`Added`" list, double-click the entry of a slave with which you want to exchange data. A dialog box opens in which you configure the data exchange with this slave. In the three sections "`Bit Strobed`", "`Polled`" and "`Change of State/Cyclic`", configure the variables depending on the communication mode. In this example, you have to select the "`Enable`" checkbox in the "`Polled`" section and enter 6 "`RxBytes`" (receiving) and 2 "`TxBytes`" (sending). (These values must correspond to the values determined in chapter 6.1, table 6-2.) "`Poll every scan cycle`" ensures data transmission with the highest update rate. Confirm your changes with "`OK`".	

No.	Instruction	Comment
15.	Change to the “`Input`” tab. After the data volume has been determined in the exchange with the slaves, you now have to configure where the data are to be stored in the data memory of the gateway. These addresses have to be matched to the configuration of the data modules that are exchanged via the PROFIBUS interface. (See table 6-5) In the simplest case, select the data module entered in the top section of the dialog box and select the “`Automap`” button. The data are then sequentially entered in the gateway data buffer and displayed correspondingly in the bottom section of the dialog box.	
16.	Change to the “`Output`” tab and repeat the procedure analogously. Also in this tab, the positioning in the buffer memory is defined by the “`Automap`” button.	

<table>
<tr><th>No.</th><th>Instruction</th><th>Comment</th></tr>
<tr><td>17.</td><td>You can use the “<code>Diagnostics</code>” tab to obtain an analysis of the individual bus nodes in the event of connection problems.
Use the “<code>Update</code>” button to update the displayed data.</td><td>
</td></tr>
<tr><td>18.</td><td>To transfer the changed configuration to the gateway, return to the “<code>Parameter</code>” tab. In the “<code>1: Master state</code>” combo box, select the “<code>Idle</code>” entry. (The individual entries of the combo box are only displayed for selection after clicking the box twice.)
Press the down arrow to the right of the combo box (see red square in the opposite figure) to set the gateway to Idle.
The Idle mode of the gateway is indicated by the “RUN” LED on the housing front (red arrow figure 6-1) which is no longer constantly lit but flashes green.</td><td>
</td></tr>
</table>

No.	Instruction	Comment
19.	The changed configuration can only be loaded to the gateway when it is in Idle mode. For this purpose, click the "Download" button.	
20.	The download is accompanied by a progress display which remains visible for several moments. Simultaneously, the "A2" LED on the NetTOOL adapter housing flashes green.	
21.	After completing, reset the gateway to Run mode by reversing step 18 and by selecting the "Run" setting. The communication between the networks should now start.	
22.	After completing the configuration, you can save the configuration with the "Save Network" start menu command to later adapt it or to transfer copies to additional devices.	

The gateway now switches the communication between the PROFIBUS network and DeviceNet.

Appendix and Literature

7 Glossary

This section provides brief explanations of important terms and abbreviations.

CAN

"Controller Area Network", an asynchronous, serial bus system which was developed for networking control devices in automobiles. CAN is used at the lower layers of the → OSI reference model for DeviceNet.

CIP

"Common Industrial Protocol", a family of object-oriented field bus protocols which are primarily used at the upper layers of the → OSI reference model for a protocol. → ControlNet, → DeviceNet and → EtherNet/IP are examples of field busses which use CIP.

ControlNet

A bus system developed by Allen-Bradley/Rockwell for use at cell level. In the upper protocol layers (compare → OSI reference model), ControlNet is based on → CIP. ControlNet uses the → "producer/consumer" model for data transmission.

DeviceNet

An open bus system developed by Allen-Bradley/Rockwell and further supported by the → ODVA for use at field level. In the lower protocol layers (compare → OSI reference model), DeviceNet is based on → CAN, in the upper layers it is based on → CIP. DeviceNet uses the → "producer/consumer" model for data transmission. See chapter 3.1.

EtherNet/IP

An open bus system developed by Allen-Bradley/Rockwell and further supported by the → ODVA for use at control level. In the upper protocol layers (compare → OSI reference model), EtherNet/IP is based on → CIP, in the lower layers it is based on Ethernet.

Gateway

A protocol converter which connects two networks. Unlike a router, a gateway can also establish a connection between *different* networks with different protocols. To do this, all information of a message except for the mere user data are compiled from the source protocol to the target protocol.

ISO

"International Organization for Standardization", the international umbrella organization of different national standards committees (such as DIN).

ODVA

"Open DeviceNet Vendors Association", an interest club consisting of manufacturers and retailers of → DeviceNet and other → CIP-based communications systems, in particular → Ethernet/IP and → ControlNet. See \10\

OSI reference model

"Open Systems Interconnection Reference Model", a layer model for the communication of open information processing systems standardized by the → ISO, which nowadays is regarded as the standard for the design and the implementation of protocols.
The communication requirements are divided into seven tasks with increasing complexity, each individual task can be performed by associated driver modules.

"Producer/consumer" model

A data communication model implemented by → DeviceNet and → ControlNet, see "Connection-oriented producer/consumer model" page 16.

Scanner

In conjunction with→ DeviceNet similar to a master (on the bus).

8 Literature

8.1 Bibliographic references

This list is by no means complete and only provides a selection of appropriate sources.

Table 8-1

	Topic	Title
/1/	STEP7	Automating with STEP7 in STL and SCL Hans Berger Publicis MCD Verlag ISBN 3-89578-113-4
/2/	DeviceNet adapter installation	“FLEX I/O DeviceNet Adapter Module Installation Instructions”, AB publication no. 1794-IN099B-EN-P, can be obtained from \11\
/3/	DeviceNet Flex I/O module installation instructions	“Flex I/O 16 Input and 16 Output w/Diagnostics Module Installation Instructions”, AB publication no. 1794-IN096B-EN-P , can be obtained from \11\
/4/	Anybus gateway manual	“ABX Generic Gateway User Manual 1_02.pdf”, can be obtained from \12\
/5/	Addendum for the Anybus manual for PROFIBUS Slave interface	“PROFIBUS Slave Network Interface Addendum 1_01.pdf”, can be obtained from \12\
/6/	Addendum for the Anybus manual for DeviceNet Scanner interface	“DeviceNet Scanner Interface Addendum 1_01.pdf”, can be obtained from \12\
/7/	Anybus installation instructions	“SP0736 Gateway Installation Sheet 1_00.pdf”, can be obtained from \12\
/8/	Anybus installation instructions, addendum for PROFIBUS Slave interface	“SP0741 PROFIBUS Slave Installation Sheet 1_00.pdf”, can be obtained from \12\

	Topic	Title
/9/	Anybus installation instructions, addendum for DeviceNet Scanner interface	"SP0747 DeviceNet Scanner Installation Sheet 1_00.pdf", can be obtained from \12\

8.2 Internet links

This list is by no means complete and only provides a selection of appropriate web sites.

	Topic	Title
\1\	Reference to the entry	http://support.automation.siemens.com/WW/view/en/23902276
\2\	Siemens A&D Customer Support	http://www.ad.siemens.de/support
\3\	Entry in the automation portal "Allen-Bradley communication with PROFIBUS scanner"	http://support.automation.siemens.com/WW/view/en/23809864
\4\	Entry in the automation portal "Allen-Bradley-communication with Echochange gateway"	http://support.automation.siemens.com/WW/view/en/23901499
\5\	HMS Networks web site (German)	http://www.hms-networks.de
\6\	Web sites for the discussion of Anybus basics	http://www.hms-networks.de/products/abx_overview.shtml
\7\	Alternative web site for Anybus basics	http://www.hms-networks.de/Technologies/whatisanybus.shtml
\8\	"Rockwell Automation" web site	http://www.rockwellautomation.com
\9\	"DeviceNet" web site	http://www.feldbusse.de/DeviceNet/devicenet.htm
\10\	"ODVA" web site	http://www.odva.org/
\11\	Rockwell web site for technical references	http://literature.rockwellautomation.com/

	Topic	Title
\12\	HMS web site for Anybus gateway references	http://www.anybus.com/eng/support/support.asp?PID=329&ProdType=Anybus-X
\13\	Anybus web site (English)	http://www.anybus.com/
\14\	DN-CBM-DP gateway, product brief	http://www.odva.org/10_2/04_products/400-1ED01.htm
\15\	DN-CBM-DP gateway, data sheet	http://www.esd-electronics.com/german/PDF-file/CAN/Englisch/dncbmdp_e.pdf
\16\	“esd” web site	http://www.esd-electronics.com/
\17\	“Bridgeway” product brief	http://www.odva.org/10_2/04_products/776-1PY01.htm
\18\	“Bridgeway” technical overview	http://www.pyramidsolutions.com/Expertise_DataCom_Products_BridgeWay_AB7605.htm
\19\	“Pyramid Solutions” web site	http://www.pyramidsolutions.com/
\20\	Woodhead gateways product brief	http://www.odva.org/10_2/04_products/711-1SS01.htm
\21\	Woodhead gateways data sheet	http://www.woodhead.com/data/current2105/edns200s.pdf
\22\	“Woodhead” web site	http://www.woodhead.com/
\23\	“DeviceNet Configuration Tool” manual	http://www.anybus.com/eng/upload/98-7256-NetTool%20DN%20User%20Manual%201_0.pdf
\24\	“DeviceNet Configuration Tool” application notes	http://www.anybus.com/eng/upload/98-9977-SCD-7231-017%20Application%20Note%20NetTool%20DeviceNet%201_00.pdf
\25\	RSNetWorx configuration software, home page	http://www.rockwellautomation.com/rockwellsoftware/design/rsnetworx/
\26\	Download page for Allen-Bradley EDS files	http://www.ab.com/networks/eds.html
\27\	Download for EDS file of the 1794-ADN Flex I/O adapter, Rev 2.1	http://www.ab.com/networks/eds/cgi-bin/detail.pl?file=DN/0001000C00010200.eds

9 History

Table 9-1 History

Version	Date	Modification
V1.0	01/01/07	First edition