Installation and Start-Up Manual Edition 12/2004

sinamics

SINAMICS S120
CANopen interface

SIEMENS

SIEMENS

SINAMICS

SINAMICS S120
CANopen interface

Commissioning Manual

Inner title infofield

12.2004 Edition

6SL3097-2AA00-0BP1

Safety Guidelines

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring to property damage only have no safety alert symbol. These notices shown below are graded according to the degree of danger.

Danger

indicates that death or severe personal injury **will** result if proper precautions are not taken.

Warning

indicates that death or severe personal injury **may** result if proper precautions are not taken.

Caution

with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.

Caution

without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.

Notice

indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The device/system may only be set up and used in conjunction with this documentation. Commissioning and operation of a device/system may only be performed by **qualified personnel**. Within the context of the safety notes in this documentation qualified persons are defined as persons who are authorized to commission, ground and label devices, systems and circuits in accordance with established safety practices and standards.

Prescribed Usage

Note the following:

Warning

This device may only be used for the applications described in the catalog or the technical description and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens. Correct, reliable operation of the product requires proper transport, storage, positioning and assembly as well as careful operation and maintenance.

Trademarks

All names identified by ® are registered trademarks of the Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Copyright Siemens AG 2004. All rights reserved.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Automation and Drives
Postfach 4848, 90327 Nuremberg, Germany

Siemens AG 2005
Technical data subject to change

Preface

Information on the SINAMICS S documentation

The SINAMICS S documentation is divided into the following areas:

- General documentation/catalogs
- Manufacturer/service documentation
- Electronic documentation

This documentation is an integral part of the manufacturer/service documentation developed for SINAMICS. All documents can be obtained separately.

You can obtain detailed information about the documents named in the documentation overview and other documents available for SINAMICS from your local Siemens office.

For the sake of simplicity, this documentation does not contain all detailed information about all types of the product and cannot cover every conceivable case of installation, operation, or maintenance.

The contents of this documentation are not part of an earlier or existing agreement, a promise, or a legal agreement, nor do they change this. All obligations entered into by Siemens result from the respective contract of sale that contains the complete and sole valid warranty arrangements. These contractual warranty provisions are neither extended nor curbed as a result of the statements made in this documentation.

Audience

This documentation is aimed at machine and plant builders, commissioning engineers, and service personnel who use SINAMICS.

Objective

Note

"SINAMICS S120 CANopen Interface" describes the steps involved in commissioning a **CANopen interface** in the SINAMICS S120 drive line-up.

This Installation and Start-Up Manual extends the description of "Initial commissioning using servo as an example" to include a description of the initial commissioning procedure for the CANopen communication interface with the CBC10 Communication Board.

Detailed instructions on commissioning the entire SINAMICS S120 drive line-up are available in the SINAMICS S120 /IH1/ Installation and Start-Up Manual.

Definition: qualified personnel

With reference to this manual and the warning labels on the product, a "qualified person" is someone who is familiar with the installation, mounting, start-up, and operation of the equipment and who has certified qualifications for the type of responsibility involved, such as:

- Training and instruction, i.e. authority to switch on and off, to earth and to label circuits and equipment according to safety regulations.
- Trained in the proper care and use of protective equipment in accordance with established safety procedures.
- First aid training.

Finding information

To help you find information more easily, the following sections have been included in the appendix in addition to the table of contents:

1. References
2. Index

Technical information

Hotline

If you have any further questions, please call our hotline:

A&D Technical Support

Tel.: +49 (0) 180 5050 - 222

Fax: +49 (0) 180 5050 - 223

http://www.siemens.de/automation/supportrequest

Please send any questions about the documentation (suggestions for improvement, corrections, and so on) to the following fax number or e-mail address:

Fax: +49 (0) 9131 98 - 2176

Fax form: See feedback page at the end of this publication

E-mail: motioncontrol.docu@erlf.siemens.de

Internet Address

Up-to-date information about our products can be found on the Internet at the following address:

http://www.siemens.com/motioncontrol

ESD notices

Caution

Electrostatic sensitive devices (ESDs) are individual components, integrated circuits, or boards that may be damaged by either electrostatic fields or electrostatic discharge.

Regulations for handling ESD components:

When handling components, make sure that personnel, workplaces, and packaging are well earthed.

Personnel in ESD areas with conductive flooring may only handle electronic components if:

They are grounded with an ESD wrist band

They are wearing ESD shoes or ESD shoe grounding straps

Electronic boards should only be touched if absolutely necessary. They must only be handled on the front panel or, in the case of printed circuit boards, at the edge.

Electronic boards must not come into contact with plastics or items of clothing containing synthetic fibers.

Boards must only be placed on conductive surfaces (work surfaces with ESD surface, conductive ESD foam, ESD packing bag, ESD transport container).

Do not place boards near display units, monitors, or television sets (minimum distance from screen: 10 cm).

Measurements must only be taken on boards when:

The measuring instrument is grounded (via protective conductors, for example).

The measuring probe is briefly discharged before measurements are taken with an isolated measuring device (for example, touching a bare metal housing).

Safety guidelines

Danger

Commissioning must not start until you have ensured that the machine in which the components described here are to be installed complies with Directive 98/37/EC.

SINAMICS S equipment must only be commissioned by suitably qualified personnel.

Personnel must take into account the information provided in the technical customer documentation for the product, and be familiar with and observe the specified danger and warning notices.

When electrical equipment and motors are operated, the electrical circuits automatically conduct a dangerous voltage.

Dangerous mechanical movements may occur in the system during operation.

All work on the electrical system must be performed after the system has been switched off and disconnected from the power supply.

SINAMICS S equipment with three-phase motors may only be connected to the line system via residual current devices (RCDs) if compatibility of the SINAMICS equipment with the RCD has been ensured as specified in EN 50178, Subsection 5.2.11.2.

Warning

Correct and safe operation of SINAMICS S equipment assumes correct transportation, storage, setup, and installation, as well as careful operation and maintenance.

The details in the catalogs and proposals also apply to the design of special equipment versions.

In addition to the danger and warning information provided in the technical customer documentation, the applicable national, local, and system-specific regulations and requirements must be taken into account.

Only protective extra-low voltages (PELVs) that comply with EN60204-1 must be connected to all connections and terminals between 0 and 48 V.

Caution

As part of routine tests, SINAMICS equipment with three-phase motors will undergo a voltage test in accordance with EN 50178. Before the voltage test is performed on the electrical equipment of industrial machines to EN 60204-1, Section 19.4, all connectors of SINAMICS equipment must be disconnected/unplugged to prevent the equipment from being damaged.

Motors must be connected in accordance with the circuit diagram provided. They must not be connected directly to the three-phase supply because this will damage them.

Note

When operated in dry operating areas, SINAMICS equipment with three-phase motors conforms to low-voltage Directive 73/23/EEC.

SINAMICS equipment with three-phase motors conforms to EMC Directive 89/336/EEC in the configurations specified in the associated EC Certificate of Conformity.

Caution

Operating the equipment in the immediate vicinity (< 1.5 m) of mobile telephones with a transmitter power of > 1 W may lead to incorrect operation.

Table of contents

Tables

Add infeed

1

1.1 Previous knowledge

Introduction

To fully understand this Installation and Start-Up Manual, you must be familiar with CANopen terminology.

This section includes:

- An overview of the most important terms and abbreviations
- A breakdown of the communication objects in the CANopen object directory in the CANopen slave software

You must be familiar with the following standards:

Note

SINAMICS S120 with CANopen complies with the following standards:

- CiA DS-301 V4.01 (Application Layer and Communication Profile)

- CiA DS-402 V2.0 (Device Profile for Drives and Motion Control)

- CiA DR-303-3 V1.0 (Indicator Specification)

1.2 Commissioning options

Introduction

In the SINAMICS S120 drive line-up, the STARTER commissioning tool offers two options for commissioning a CANopen interface:

- Via predefined message frames ("predefined connection set")
- Via free PDO mapping (user-defined message frames)

1.3 Terminology

When using a CANopen profile via the CAN bus, you will encounter the following common terms and abbreviations:

CAL (CAN application layer)

Communication layer above the CAN bus designed for CAN bus applications in open communication systems. It comprises NMT, DBT, LMT, and CMS elements. Since CAL is very extensive and highly flexible, a subset of CAL functions for automation applications has been defined with the CANopen communication profile CiA DS 301.

CAN (controller area network)

A serial bus system (also known as CAN bus) that was originally designed for use in vehicles but is now also used in automation technology. CANopen (see below) extends the CAN bus protocols to include additional layers.

CAN controller

An electronic module whose hardware processes the CAN bus protocols.

CAN identifier

With the assignment of CAN identifiers to CAN messages (CANopen: PDOs, SDOs), the relative priority of the CAN messages over one another is specified.

CANopen

A CiA-defined communication model based on the CAN bus and CAL. To make it easier to use devices produced by different manufacturers on a bus, a subset of CAL functions for automation applications has been defined with the CANopen communication profile CiA DS 301. Other profiles are also defined for certain device types (e.g. drives).

CiA (CAN in Automation international users and manufacturers group)

Association of manufacturers and users of devices with a CAN interface.

CMS (CAN message specification)

A part of the CAL that defines different mechanisms for transferring data.

COB (communication object)

On the CAN bus, data is transferred in packages known as communication objects (COB) or CAN messages.

Devices connected to the CAN bus can transmit and receive COBs.

COB-ID (COB identifier)

Each COB can be uniquely identified by means of an identifier, which is part of the COB. CAN specification 2.0A supports up to 2048 COBs, which are identified by means of 11-bit identifiers. In this documentation, COB IDs are always specified as hexadecimal values.

A list of COB identifiers, which contains all the COBs that can be accessed via CAN, is available in the object directory for the relevant drive unit.

DBT (distributor)

A part of the CAL that controls the distribution of COB IDs. Like most CANopen devices, the digital servo amplifiers use more straightforward means of assigning COB IDs to a device: they are selected via default values on the basis of the node ID and can, if necessary, be changed via SDO.

DRIVECOM

Association of drive manufacturers that has developed standards for networking drives (profiles). DRIVECOM profile 22 for positioning drives, which is implemented in the servo amplifier, was used by CiA as a basis for developing CANopen drive profile CiA DSP 402.

EMCY (Emergency)

SINAMICS S120 features an emergency object to inform other nodes on the CANopen bus of internal device faults or CAN bus faults. It is assigned a high priority and provides important information about the status of the drive unit.

Channel

With the SINAMICS S120 drive line-up, up to 24 receive PDOs can be received.

One channel in the CAN controller is assigned to each activated receive PDO. Transmit PDOs are transmitted via two predefined channels.

Send PDOs always use two predefined channels.

NMT (network management)

A part of CAL used for initialization, configuration, and troubleshooting purposes.

Node ID (node identification)

Uniquely identifies a device in the CANopen network. For this reason, all the devices must have a unique node ID (bus address). The default distribution (standard setting) of the COB IDs is derived from the node ID. In this documentation, node IDs are always specified as hexadecimal values.

OD (object directory)

A "database" – or object directory – containing all the objects supported by a drive is defined for each drive unit. The object directory contains:

- Type, description, and serial number of the device
- Name, format, description + index for each object
- Lists of PDOs and SDOs
- The data that is assigned to the PDOs
- The time at which the PDOs are transmitted (SYNC, change in object, etc.)
- The time at which emergency messages are transmitted
- ...

All the drive unit variables are accessed via objects. The SDO and PDO communication services access the object directory of the drive unit.

PDO (process data object)

Used for accessing selected data rapidly and in real time. Mappings of certain PDOs are preconfigured for certain variables or groups of variables.

The SDO is used to access all the other variables.

Profile

In the case of communication with bus systems, profiles are documents used for device standardization purposes, whereby communication functions (in a communication profile), device functions (in a device profile), or drive functions (in a drive profile) are described from the point of view of the communication interface.

RPDO (receive PDO)

PDO is received by the device (contains the final position, for example).

SDO (service data object)

The SDO provides access to all variables in a CANopen device (in the case of drives: drive and CANopen variables).

The SDO is generally used for configuration purposes. PDOs provide fast, real-time access to selected variables.

SYNC (synchronization)

SYNC is a special message frame that synchronizes the CAN devices with each other. This message frame has a very high priority.

TPDO (transmit PDO)

PDO transmitted by the drive (contains the actual position value, for example).

Variable

All the drive and CANopen functions can be accessed via variables.

Variables can be accessed via SDOs or PDOs.

1.4 CAN bus structure for SINAMICS

The following diagram shows an example of how the hardware and software are arranged when a CANopen interface is commissioned.

The diagram shows the following:

- How a master application of a CANopen user is connected to a SINAMICS S120 drive line-up.
- The CAN bus interface of the CBC10 Communication Board.
- The associated CANopen slave software on the CU320 Control Unit and the meaning of the terms "transmit" and "receive", which are used for the transmit and receive message frames during commissioning.
- How a PC on which the STARTER commissioning tool has been installed can be connected via PROFIBUS.

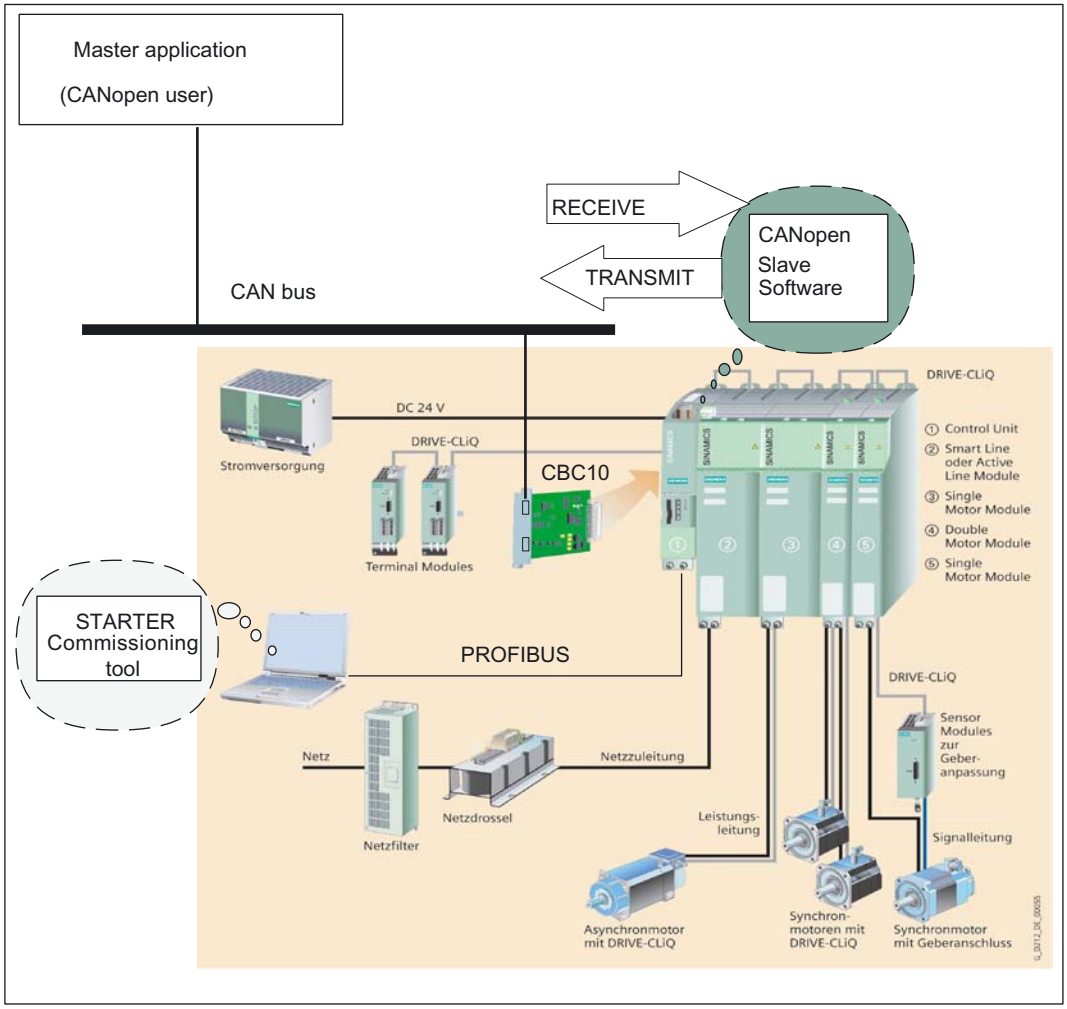


Figure 1-1 SINAMICS S120 drive line-up with CAN bus

1.5 What does the CANopen slave software describe?

Introduction

In the CANopen slave software, a separate expert list is created – among other things – for each module in the SINAMICS S120 drive line-up. The expert list reflects – among other things – the CANopen object directory.

Example

The data value for the control word of the object directory index from the CANopen object directory is entered in parameter r8795 in the expert list.

Depending on how the Motor Modules are arranged in the drive line-up, they are counted as a drive object from the point of view of CANopen.

See the following diagram:

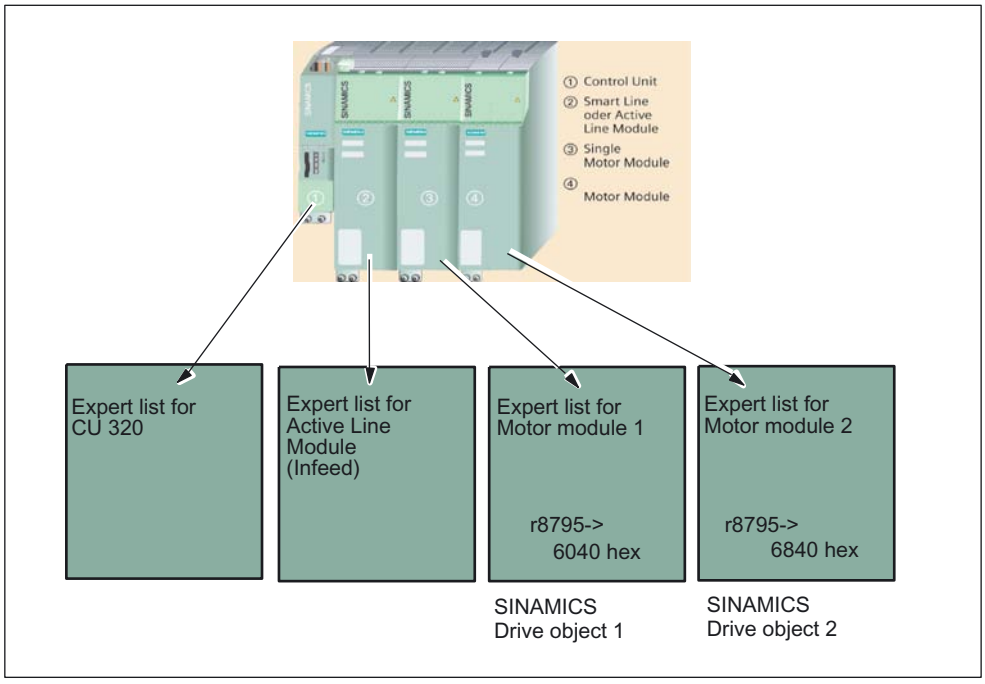


Figure 1-2 Breakdown of drive objects from the point of view of CANopen

Motor Module 1 is the first drive object for CANopen. In the description in the drive profile, drive object 2 begins with an offset of 800 hex.

If, for example, you want to read a value from parameter **r8795** (control word for Motor Module 1, that is, drive object 1), the value from object **6040** hex is displayed.

For Motor Module 2 (drive object 2), therefore, this value is **6840** hex.

Note

Each module has a separate receive and transmit buffer for transferring message frames.

1.6 CANopen object directory

Note

When the drive objects are initialized, the CANopen objects are initialized in the object directory for the SINAMICS S120 drive line-up (CANopen slave software).

The following diagram shows the breakdown of communication objects in the CANopen object directory in the CANopen slave software using three SINAMICS drive objects as an example (values are hexadecimal).

Note

In the section "Communication objects", a table is provided listing the communication objects of CANopen and SINAMICS S120 parameters that are used in SINAMICS S120 for communication via the CANopen interface. These are:

- Drive-independent CU320 communication objects
- Drive-dependent communication objects
- Objects in drive profile DSP402

See also

Communication objects (Page 5-1)

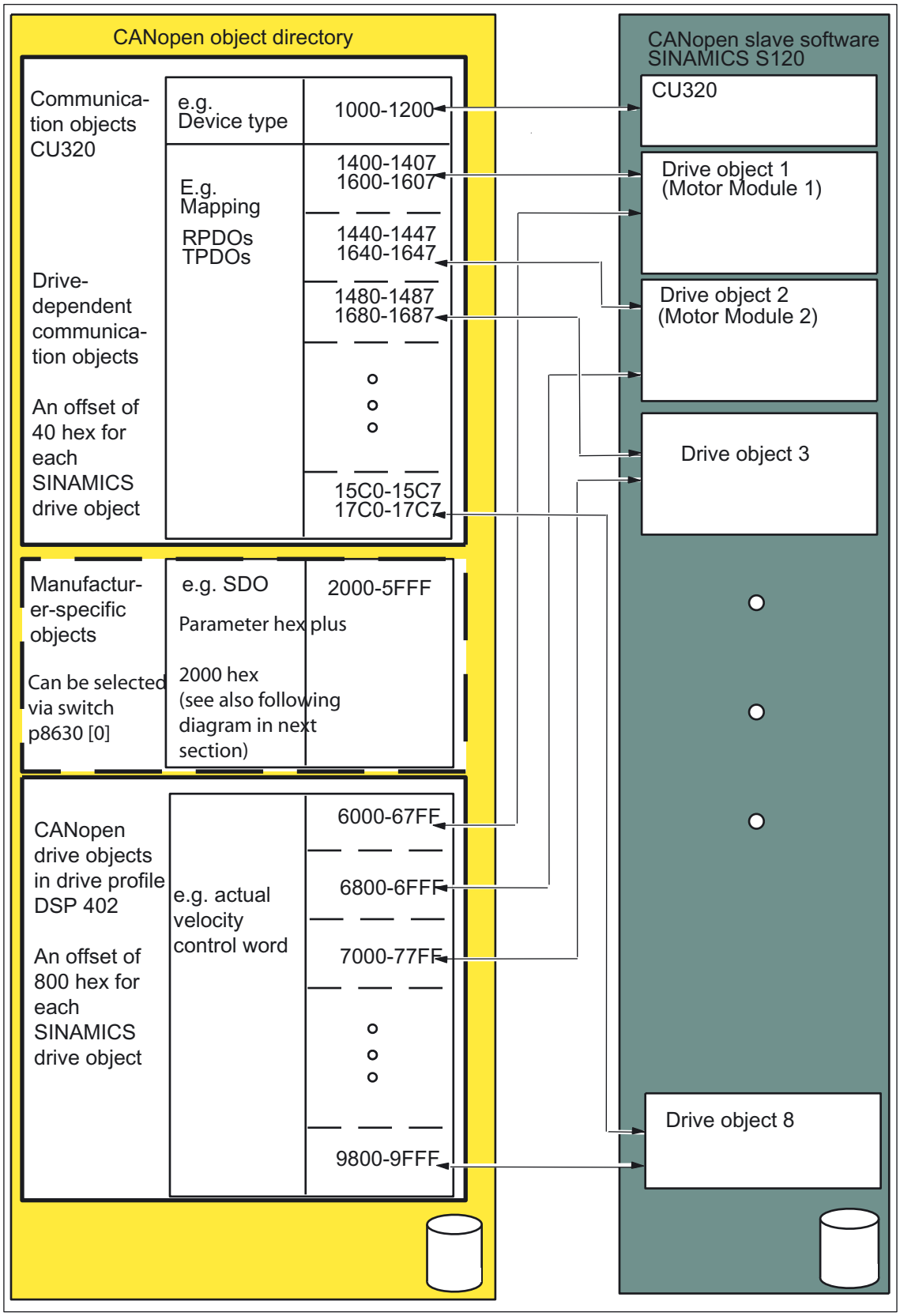


Figure 1-3 CANopen object directory

1.7 Manufacturer-specific objects

Introduction

Manufacturer-specific objects contain the data values for the SINAMICS S120 parameters.

Example

The following diagram shows the breakdown of manufacturer-specific objects in the object directory.

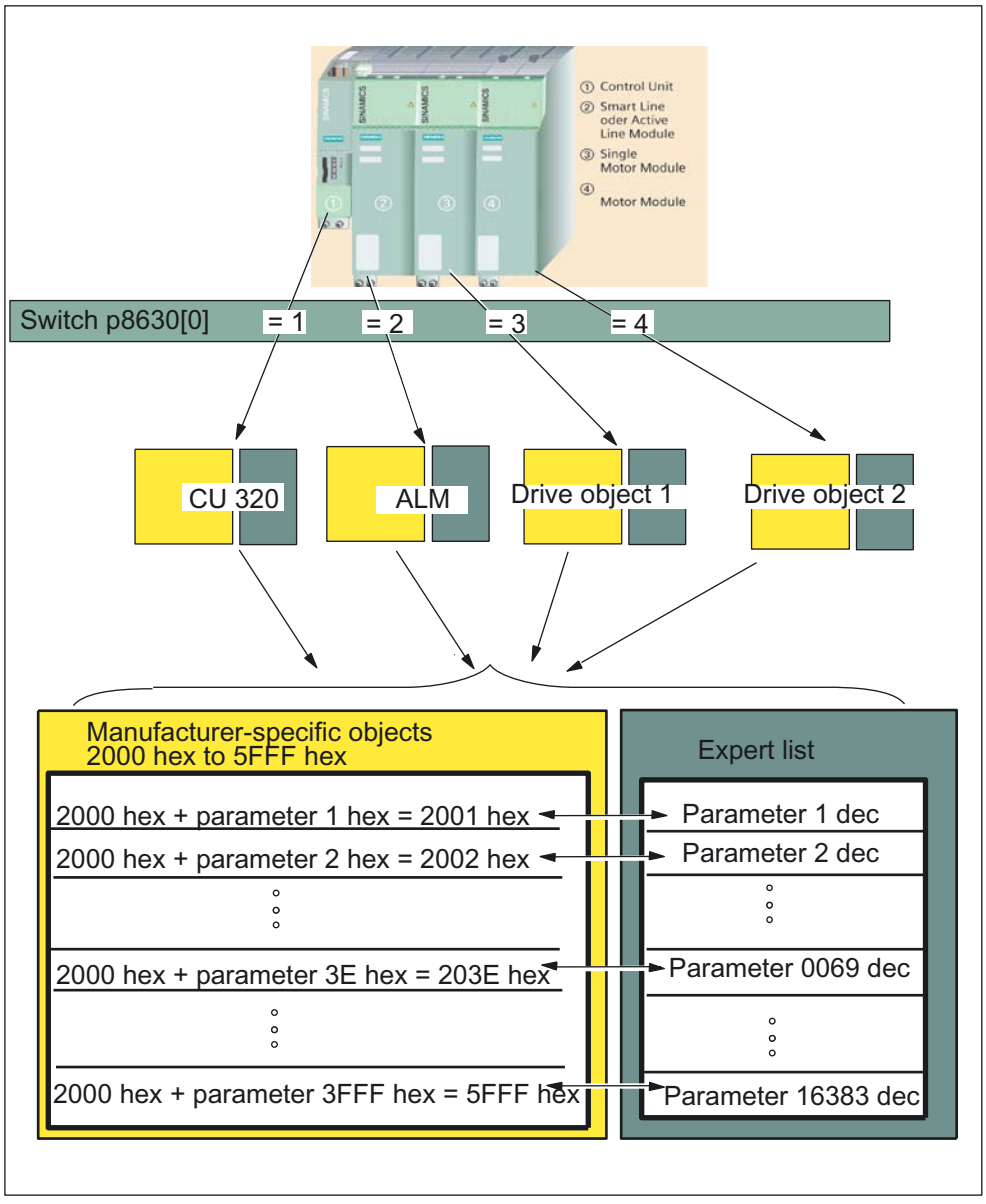


Figure 1-4 Manufacturer-specific objects

Principle

All SINAMICS S120 parameters can be addressed via the SDO->parameter channel.

This functions as follows:

The SDO->parameter channel operates entirely within the manufacturer-specific area of the CANopen profile.

This means that all SINAMICS S120 parameters can be addressed via the objects 2000 hex to 5FFF hex.

The SDO->parameter channel converts manufacturer-specific objects to parameters internally. 2000 hex is added to the parameter number converted to a hexadecimal value. This number is the object number in the SDO request required to access the SINAMICS S120 parameter. Since the parameter range of a SINAMICS S120 drive object already covers the entire manufacturer-specific object space, the drive object that is to be written is selected in parameter **p8630[0]** in SINAMICS S120.

A "SINAMICS S120 parameter" can be an "r" or "p" parameter. The manufacturer-specific objects contain the data values for these parameters.

Depending on the switch position of parameter **p8630[0]**, the data values for the modules can be read or written.

If, for example, parameter r0062 (set velocity) is to be read by drive object 1, then:

- The switch must be switched to "3" in parameter **p8630 [0]** (see previous diagram)
- Parameter **r0062** must be converted to a hexadecimal value and 2000 hex added. This hexadecimal number 203E hex, which corresponds to the object number, can be used to access the parameter via an SDO request.

Note

The switch in parameter **p8630 [0]** determines the structure of the drive line-up. If an Active Line Module (ALM) is not installed, the count for the first drive object starts with "2".

1.8 What is the maximum number of PDOs (channels) that can be created?

Introduction

PDOs (process data objects) are process data that allow fast, real-time access to selected data. Mappings of certain PDOs are preconfigured for certain variables or groups of variables.

PDOs include, for example, the control word, set velocity, and actual velocity.

Principle

Note

CANopen can monitor up to eight SINAMICS drive objects in the object directory. Up to eight PDOs can be defined in a SINAMICS drive object.

Note

The maximum number of receive and transmit PDOs (channels) on the CAN bus in the SINAMICS S120 drive line-up is 25.

Since a maximum of eight PDOs can be defined for each SINAMICS drive object, this would mean that with maximum utilization of the PDOs created, three SINAMICS drive objects could potentially be utilized. An additional PDO could also be created.

Principle

Eight PDOs (plus one PDO in one drive object) multiplied by three SINAMICS drive objects = max. 25 PDOs (channels).

1.9 Which transmit and receive message frames for process data are required?

Introduction

In the SINAMICS S120 drive line-up, the STARTER commissioning tool offers two options for commissioning a CANopen interface:

- Via predefined message frames ("predefined connection set") and COB IDs
- Via free PDO mapping (user-defined message frames)

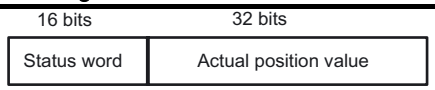
In the first step, you are advised to activate the "predefined connection set" and then update the preconfigured parameters in accordance with your application in the free PDO mapping process.

Predefined message frames for SINAMICS S120

The following process data objects are predefined and mapped in the "predefined connection set" in the receive and transmit message frames for the relevant drive objects. The table contains the hexadecimal object values for drive object 1. Each additional drive object begins with an offset of 800 hex.

Table 1-1 Process data objects in the predefined connection set

Type	Process data	Message frame
Receive message frame	RPDO1	16 bits: Control word 6040 hex
	RPDO2	16 bits: Control word; 32 bits: Set velocity 6040 hex+6077 hex
	RPDO3	16 bits: Control word; 16 bits: Set torque 6040 hex+6071 hex
	RPDO4	16 bits: Control word; 32 bits: Set velocity; 16 bits: Set torque 6040 hex+60FF hex+6071 hex
Transmit message frame	TPDO1	16 bits: Status word 6041 hex
	TPDO2	16 bits: Status word; 32 bits: Actual velocity 6041 hex+6062 hex
	TPDO3	16 bits: Status word; 16 bits: Actual torque 6041 hex+6074 hex

Type	Process data	Message frame
	TPDO4	 6041 hex+6083 hex

Note

For each drive object in the expert list, the process data objects for mapping the message frames begin as follows:

For receive message frames: as of parameter **p8710**

For transmit message frames: as of parameter **p8730**

1.10 What is PDO mapping?

Introduction

PDO mapping is used to map drive objects (process data (e.g. setpoints or actual values)) from the object directory for each PDO service as a message frame.

The PDO itself transfers the data values for these objects.

Once detected, each drive object is assigned up to 8 receive and 8 transmit PDOs.

A CAN message frame can transfer up to 8 bytes of user data. The user can decide which data is to be transferred in a PDO.

Example

The following diagram uses an example to illustrate PDO mapping (values are hexadecimal (e.g. object size 20 hex = 32 bits)):

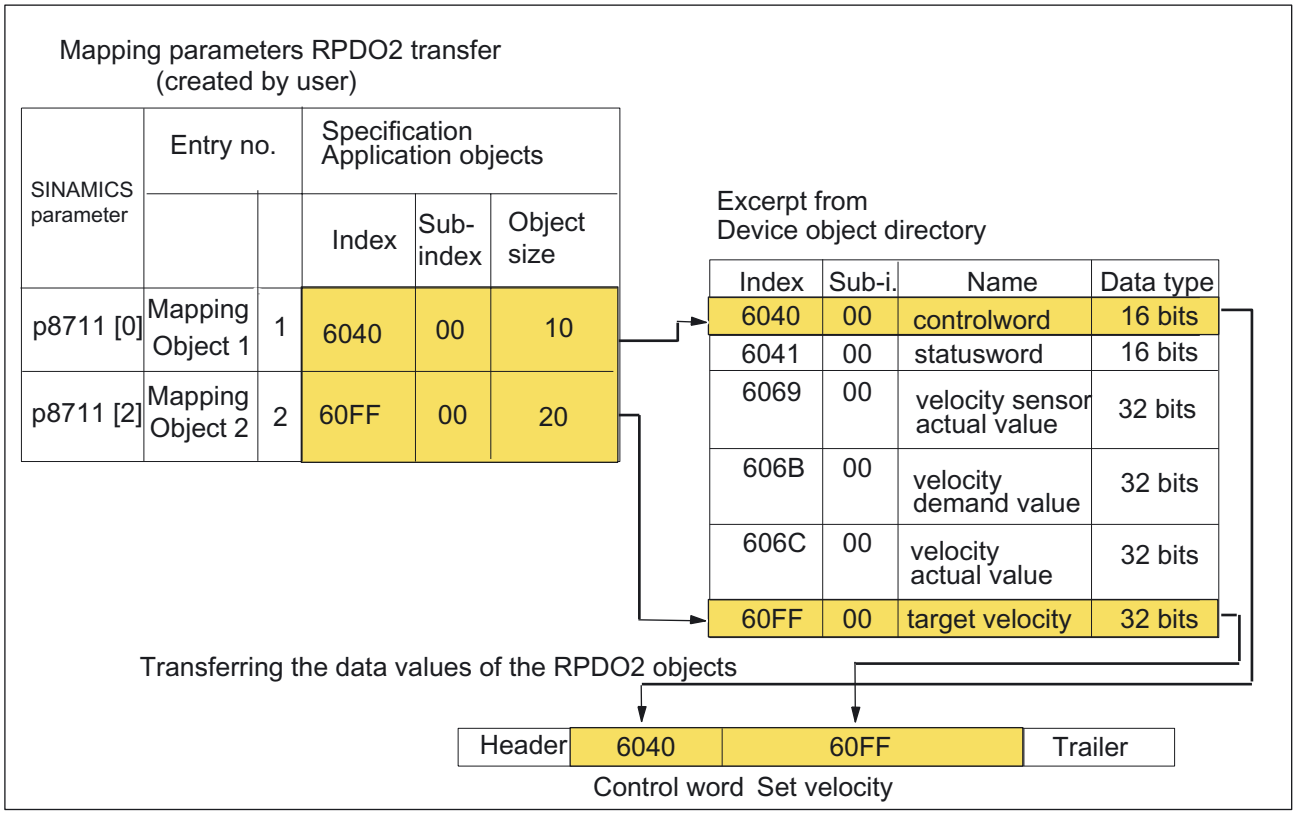


Figure 1-5 PDO mapping

1.11 What is a COB ID?

Introduction

Each COB (communication object) can be uniquely identified by means of an identifier, which is a part of the COB. CAN specification 2.0A supports up to 2048 COBs, which are identified by means of 11-bit identifiers.

A list of COB identifiers, which contains all the COBs that can be accessed via CAN, is available in the object directory of the relevant SINAMICS S120 drive unit.

The COB ID prioritizes the communication objects by means of an identifier assignment.

Principle

The COB ID is user defined.

CANopen defines a preset identifier assignment ("predefined connection set") for the communication objects. The following table shows how the preset identifier assignment is structured.

Table 1-2 Identifier assignment

Communication objects	Function code		Resulting COB ID		OD index (hex) 1)
	dec	bin	hex	Explanation	
NMT commands (NMT) 2)	0	0000	0	0 dec	–
Sync message (SYNC)	1	0001	80	128 dec	1005,1006,1007
Alarm objects (EMERGENCY)	1	0001	81–FF	129–255 dec	1014, 1015
Tx–PDO1	3	0011	181–1FF	180 hex + node ID	1800
Rx–PDO1	4	0100	201–27F	200 hex + node ID	1400
Tx–PDO2	5	0101	281–2FF	280 hex + node ID	1801
Rx–PDO2	6	0110	301–37F	300 hex + node ID	1401
Tx–PDO3	7	0111	381–3FF	380 hex + node ID	1802
Rx–PDO3	8	1000	401–47F	400 hex + node ID	1402
Tx–PDO4	9	1001	481–4FF	480 hex + node ID	1803
Rx–PDO4	10	1010	501–57F	500 hex + node ID	1403
Tx–SDO2)	11	1011	581–5FF	580 hex + node ID	1200
Rx–SDO2)	12	1100	601–67F	600 hex + node ID	1200
Node monitoring (NMT error control) 2)	14	1110	701–77F	700 hex + node ID	100C, 100D

Footnotes for above table

- 1) The OD index for the Tx and Rx PDOs depends on the number of drive objects in the drive line-up. Each additional drive object begins with the description in an offset of 40 hex (with Tx/Rx, the x stands for the relevant SINAMICS S120 drive object, T = transmit, R = receive).

Example: for TPDOs, the OD index for the first drive object begins with 1800 hex and, for RPDOs, with 1400 hex; each additional drive object begins with an offset of 40 hex –> 1840 hex and 1440 hex for the second drive object, and so on.
With a predefined connection set, the COB IDs are incremented by +1 for each additional SINAMICS S120 drive object.

- 2) These COB IDs are set by default.

COB IDs for SINAMICS S120

With SINAMICS S120, the following COB IDs for receive and transmit message frames are predefined for the drive objects in the predefined connection set:

Table 1-3 Identifier assignment

Communication objects	Function code		Resulting COB ID		OD index (hex)
	dec	bin	hex	Explanation	
TPDO1	3	0011	181–1FF	180 hex + node ID	1800
RPDO1	4	0100	201–27F	200 hex + node ID	1400
TPDO2	5	0101	281–2FF	280 hex + node ID	1801
RPDO2	6	0110	301–37F	300 hex + node ID	1401
TPDO3	7	0111	381–3FF	380 hex + node ID	1802
RPDO3	8	1000	401–47F	400 hex + node ID	1402
TPDO4	9	1001	481–4FF	480 hex + node ID	1803
RPDO4	10	1010	501–57F	500 hex + node ID	1403

Note

In the predefined connection set, the COB IDs are incremented by +1 for each additional SINAMICS S120 drive object.

Example:

for the **first** SINAMICS S120 drive object,
180 hex + node ID applies for TPDO1;

for the **second** SINAMICS S120 drive object,
180 hex + node ID **plus 1** applies for TPDO1.

Note

You assign COB IDs for each drive object in the STARTER expert list as of **p8700** for receive message frames and **p8720** for transmit message frames.

1.12 What are SDO services?

Introduction

SDO services allow you to access the object directory for the connected drive unit. An SDO connection is a peer-to-peer connection between an SDO client and a server.

The drive unit and its object directory is an SDO server.

The identifiers for the first SDO channel of a drive unit are defined according to CANopen.

For communication between:

- Receive: server <= client, COB ID 600 hex + node ID applies
- Transmit: server => client, COB ID 580 hex + node ID applies

Properties

The SDOs have the following properties:

- Confirmed transfer of objects
- The transfer procedure is always asynchronous
- Transfer of values greater than 4 bytes (normal transfer)
- Transfer of values with no more than 4 bytes (expedited transfer)
- Corresponds with the acyclic PROFIBUS parameter channel
- All drive unit variables can be addressed via SDO

1.13 What is a BICO interconnection in conjunction with CANopen?

Introduction

Each drive unit contains a large number of interconnectable input and output variables as well as internal control variables.

BICO technology (binector connector technology) allows the drive to be adapted to a wide variety of conditions.

Digital and analog signals, which can be connected freely by means of BICO parameters, are identified by the prefix BI, BO, CI, or CO in their parameter name.

These parameters are identified accordingly in the parameter list or in the function diagrams.

There are:

- Binectors, with
  BI: binector input, BO: binector output
- Connectors, with
  CI: connector input, CO: connector output

Interconnecting signals using BICO technology

To interconnect two signals, a BICO input parameter (signal sink) must be assigned to the required BICO output parameter (signal source).

Note

See also the Installation and Start-Up Manual /IH1/ or the List Manual /LH1/.

BICO interconnection for CANopen parameters

The drive objects created during PDO mapping are interconnected in a receive and transmit buffer for CANopen. Each process data object only appears once in each buffer.

For an explanation of how the process data objects for the receive and transmit buffer are interconnected, see "Interconnecting the receive and transmit buffer".

See also

BICO interconnection procedure in STARTER (Page 2-9)

Interconnecting the receive buffer (Page 3-42)

Interconnecting the transmit buffer (Page 3-45)

1.14 How does SINAMICS S120 behave during ramp-up with the CANopen Communication Board?

Introduction

SINAMICS S120 only ramps up as a CANopen slave when the CANopen Communication Board is inserted.

Ramp-up

During the ramp-up, the CAN software first queries the hardware address switch. If it is set to 0 or 127, the parameter of the CAN bus address can be written.

The address can be set via parameter **p8620**. If the address switch is set to 1-126, this address is copied to the CAN bus address parameters and displayed. The parameter is then read only.

Profile velocity mode

Profile velocity mode is supported for servo drives.

This mode allows velocity setpoints to be defined and set according to variable velocity profiles.

See also

CAN interface (Page 3-14)

1.15 What does network management (NMT service) mean?

Introduction

Network management is node oriented and has a master-slave structure.

The NMT services can be used to initialize, start, monitor, reset, or stop nodes.

SINAMICS S120 is an NMT slave.

Overview

The following diagram shows a status diagram of a CANopen node with a SINAMICS S120 interface. This is followed by a table listing the NMT services that are available for controlling the status transitions.

A detailed description of the NMT service is available in CANopen standard "CiA DS-301 V4.01 (Application Layer and Communication Profile)".

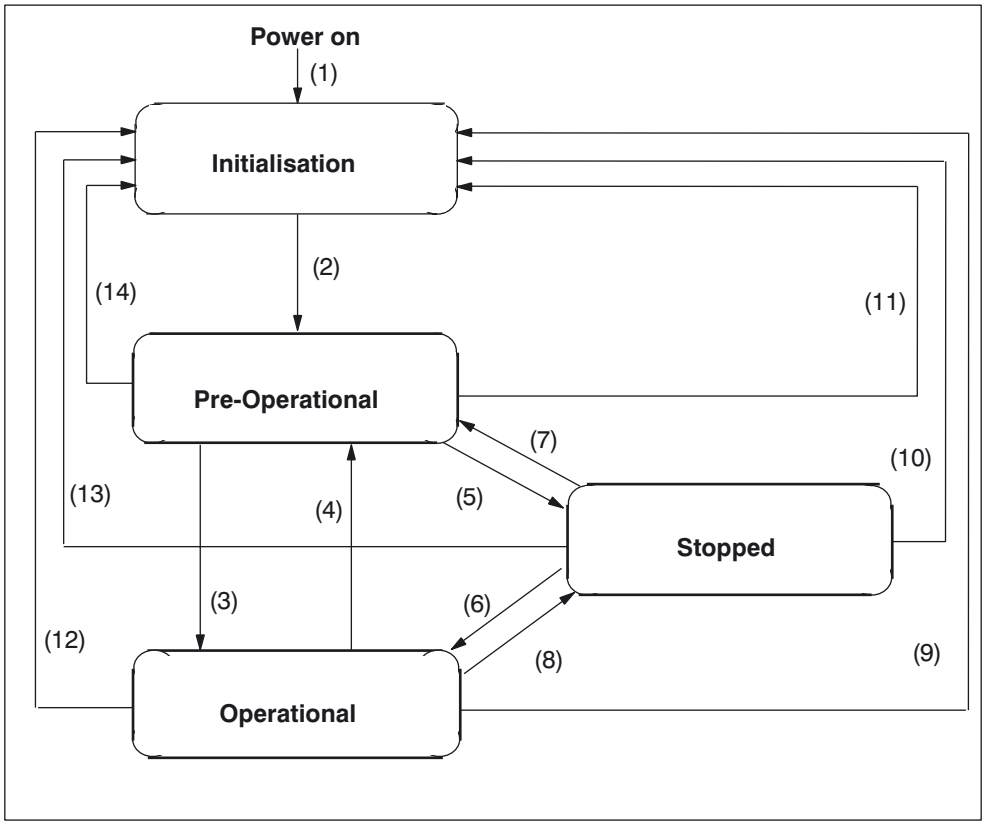


Figure 1-6 Status diagram of a CANopen node

Note

In the "Pre-operational" status, only SDO communication is possible; PDO communication is inhibited.

In the "Operational" status, PDO communication also takes place.

The diagnostics LED -> green (CANopen RUN LED) indicates the status.

NMT services

The following table lists the NMT services that are available for controlling the status transitions.

Table 1-4 Transitions in the status diagram

Transitions	Services
(1)	After power-on, the Control Unit automatically switches to "Initialization".
(2)	After initialization, it switches to "Pre-Operational".
(3), (6)	Start_Remote_Node command (CS = 128)
(4), (7)	Enter_Pre-Operational_State command (CS = 128)
(5), (8)	Stop_Remote_Node command (CS = 2)
(9), (10), (11)	Reset_Node command (CS = 129)
(12), (13), (14)	Reset_Communication command (CS = 130)

The NMT services have the following functions:

- Start Remote Node:
  command for switching from the "Pre-Operational" communication status to "Operational". The drive can only transmit and receive process data in the "Operational" status.
- Stop Remote Node:
  command for switching from "Pre-Operational" to "Stopped" or from "Operational" to "Stopped". The node can only process NMT commands in the "Stopped" status.
- Enter Pre-Operational:
  command for switching from "Operational" or "Stopped" to "Pre-Operational". In the "Pre-Operational" status, the node cannot process any PDOs. It can, however, be parameterized or operated via SDOs, which means that setpoints can also be specified.
- Reset Node:
  command for switching from "Operational", "Pre-Operational", or "Stopped" to "Initialization". When the Reset Node command is issued, all the objects (1000 hex - 9FFF hex) are reset to the status that was present after "Power On".
- Reset Communication:
  command for switching from "Operational", "Pre-Operational", or "Stopped" to "Initialization". When the Reset Communication command is issued, all the communication objects (1000 hex - 1FFF hex) are reset to their original status.

Prerequisites for commissioning

2

Section content

This section describes the commissioning prerequisites:

- CBC10 Communication Board
- STARTER commissioning tool

Note

All the parameters, faults, alarms, and function diagrams for CANopen in the SINAMICS S120 drive line-up are described in the List Manual SINAMICS S120 /LH1/.

2.1 Prerequisites for commissioning

Overview

To commission a CAN bus in a SINAMICS S120 drive line-up, the following hardware and software components are required:

- CBC10 Option Board (CAN Communication Board)
- CompactFlash card with firmware
- Connecting the Control Unit PROFIBUS interface to a PC/PG with PROFIBUS interface
- The STARTER commissioning tool must be installed on your PC/PG

Note

For descriptions of the components in a SINAMICS S120 drive line-up and instructions on wiring the components, connecting the PROFIBUS interface to a PC/PG, and installing the STARTER commissioning tool, see the Equipment Manuals /GH1/ and /GH2/ and the Installation and Start-Up Manual /IH1/.

2.2 CBC10 Communication Board for CAN bus

Introduction

The CBC10 Option Board (CAN Communication Board) is used to connect drives in the SINAMICS S120 drive system to higher-level automation systems with a CAN bus.

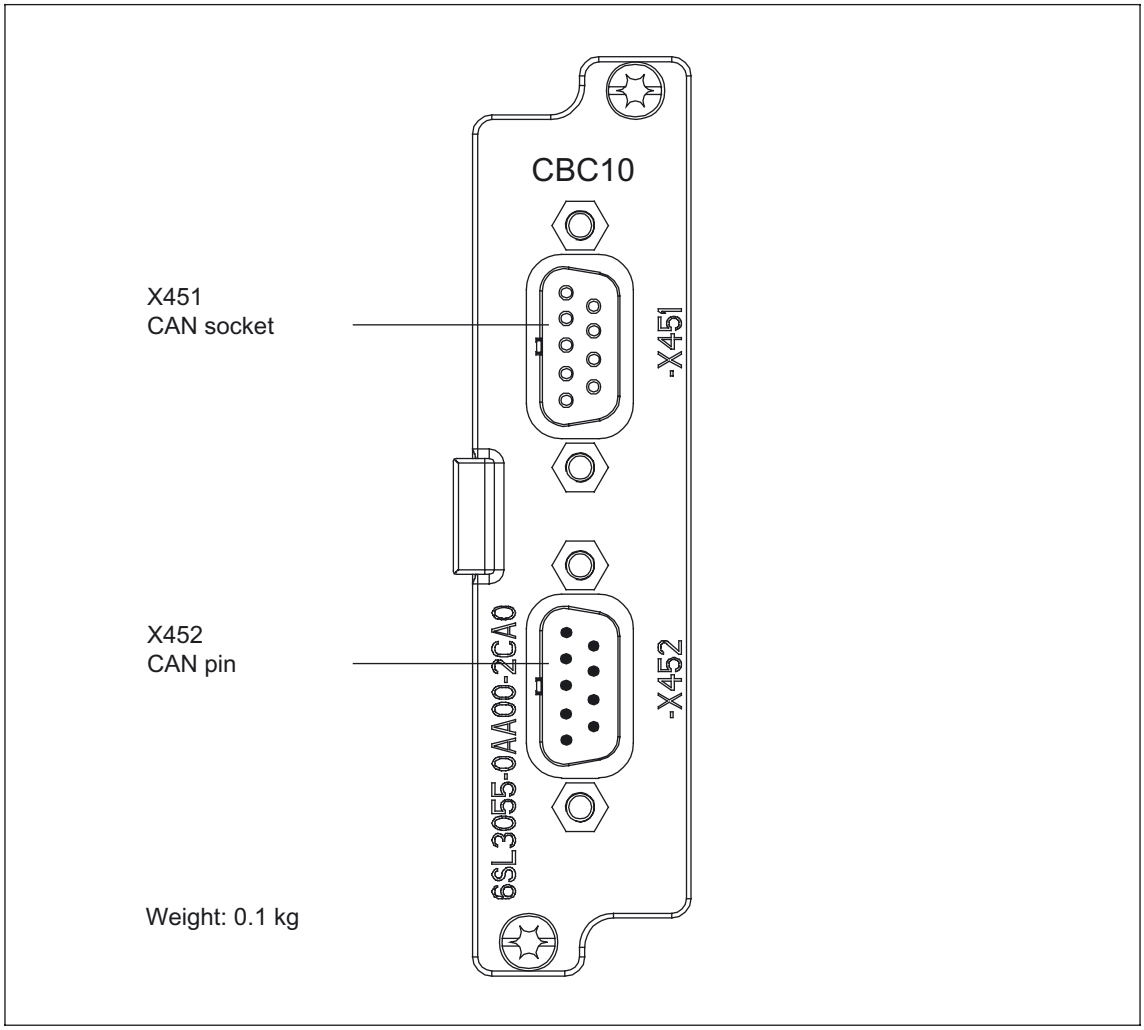


Figure 2-1 View of the CBC10 Option Board

The CANopen Option Board uses two 9-pin Sub D connectors for the connection to the CAN bus system.

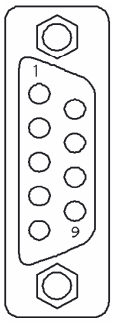
The connectors can be used as inputs or outputs. Unused pins are plated through.

The following baud rates (among others) are supported: 10, 20, 50, 125, 250, 500, 800 kBaud, and 1 MBaud.

CAN bus interface X451

The X451 CAN bus interface has the following socket assignments:

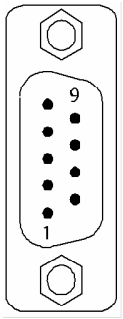
Table 2-1 CAN bus interface X451

	Pin	Designation	Technical specifications
	1	Reserved	
	2	CAN_L	CAN signal (dominant low)
	3	CAN_GND	CAN ground
	4	Reserved	
	5	CAN_SHLD	Optional shield
	6	GND	CAN ground
	7	CAN_H	CAN signal
	8	Reserved	
	9	Reserved	
Type: 9-pin SUB-D female			

CAN bus interface X452

The X452 CAN bus interface has the following socket assignments:

Table 2-2 CAN bus interface X452

	Pin	Designation	Technical specifications
	1	Reserved	
	2	CAN_L	CAN signal (dominant low)
	3	CAN_GND	CAN ground
	4	Reserved	
	5	CAN_SHLD	Optional shield
	6	GND	CAN ground
	7	CAN_H	CAN signal
	8	Reserved	
	9	Reserved	
Type: 9-pin SUB-D male			

2.2.1 CBC10 Installation

Mounting and installation steps

The CBC10 Communication Board is installed on the option slot on Control Unit CU320 as follows (see diagram below):

1. Unscrew and remove the protective cover.
2. Insert the Communication Board.
3. Secure the Communication Board.

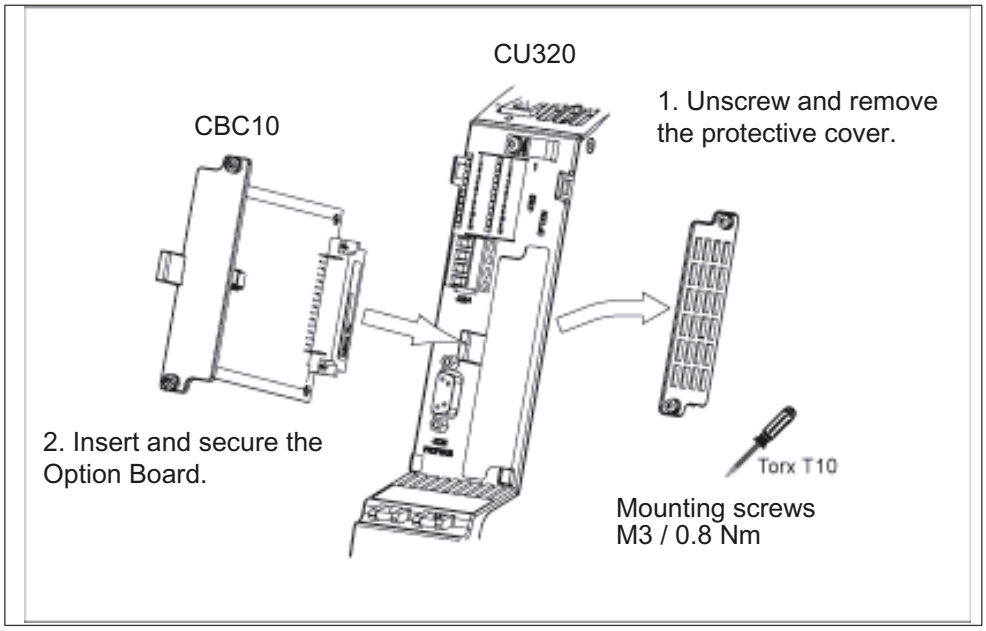


Figure 2-2 Installing the CBC10 Communication Board

2.2.2 CANopen functionality

Introduction

The CBC10 supports the CANopen transfer types with SDOs (service data objects) and PDOs (process data objects).

The CBC10 also supports free PDO mapping.

The CBC10 supports CANopen communication profile DS 301 version 4.0, device profile DSP 402 (drives and motion control) version 2.0, and indicator profile DR303-3 version 1.0.

For communication monitoring purposes, the CBC10 supports node guarding and the heartbeat protocol (heartbeat producer).

The CBC10 features an SDO->parameter channel that can be used to read or write all the SINAMICS S120 parameters.

The CBC10 firmware supports profile velocity mode.

Node guarding

SINAMICS S120 waits a certain time (node life time) for message frames from the master application and permits a specific number (life time factor) of failures within a specified time interval (node guard time).

The node life time is calculated by multiplying the node guard time by the life time factor.

Heartbeat protocol

SINAMICS S120 (producer) cyclically transmits (heartbeat time) its communication status (sign of life) on the CAN bus to the master application.

Profile velocity mode

This mode allows velocity setpoints to be defined and set according to variable velocity profiles.

2.2.3 Diagnostics LED "OPT"

Overview

The diagnostics LED "OPT" on the Control Unit CU320 indicates the status of the CANopen node on the device.

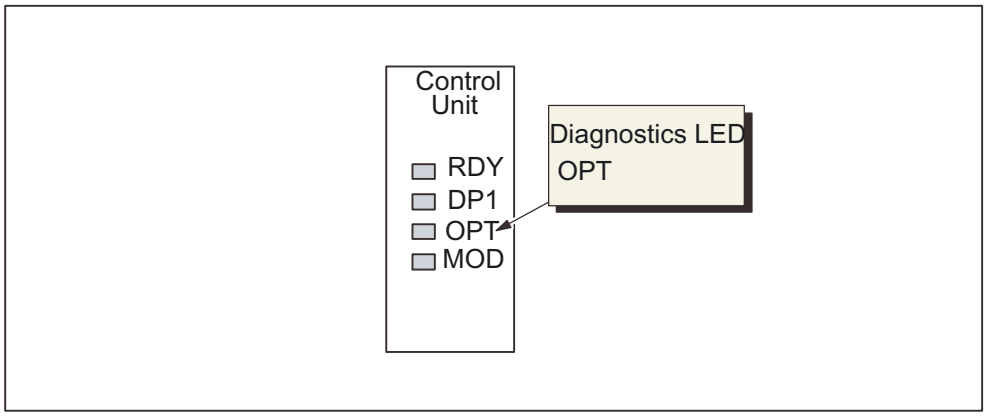


Figure 2-3 Overview of the LEDs on the Control Unit

The diagnostics LED "OPT" on the Control Unit, which displays both the module and communication status, provides users with all the required information about the current status of the CBC10.

Principle

The different flashing frequencies indicate the following:

- Diagnostics LED OPT -> red (see the appropriate table below) indicates whether or not an error is present.
- Diagnostics LED OPT -> green (see the appropriate table below) indicates the status of the nodes in the communication state machine.

Diagnostics LED OPT -> red

Table 2-3 Diagnostics LED OPT -> red (CANopen error LED)

ERROR LED flashing frequency	Status	Meaning
Off	No error	Ready to operate
Single flash	Warning limit reached	At least one of the error counters in the CAN controller has reached the warning threshold "Error passive" (too many message frames with errors).
Double flash	Error control event	A guard event has occurred.
On	Bus off	The CAN controller is "Bus off".

Diagnostics LED OPT -> green

Table 2-4 Diagnostics LED -> green (CANopen RUN LED)

ERROR LED flashing frequency	Status	Meaning
Single flash	Stopped	The node is in the STOPPED status.
Flashing	PRE-OPERATIONAL	The node is in the PRE-OPERATIONAL status.
On	OPERATIONAL	The node is in the OPERATIONAL status.

2.3 STARTER commissioning tool

Start

To launch STARTER, click the STARTER icon or choose Start > SIMATIC >STARTER in the Windows Start menu.

2.3.1 The STARTER user interface

You can use STARTER to create the sample project. The different areas of the user interface are used for different configuration tasks (see diagram below):

- Project navigator: this area displays the elements and objects that can be added to your project.
- Working area: you create the project in this area:
  - When you are configuring the drive, this area contains the Wizards that help you configure the drive objects.
  - You can configure the parameters for the speed setpoint filter, for example.
  - When you call up the expert list, the system displays a list of all the parameters that you can view or change.
- Detail view: this area contains detailed information on faults and alarms, for example.

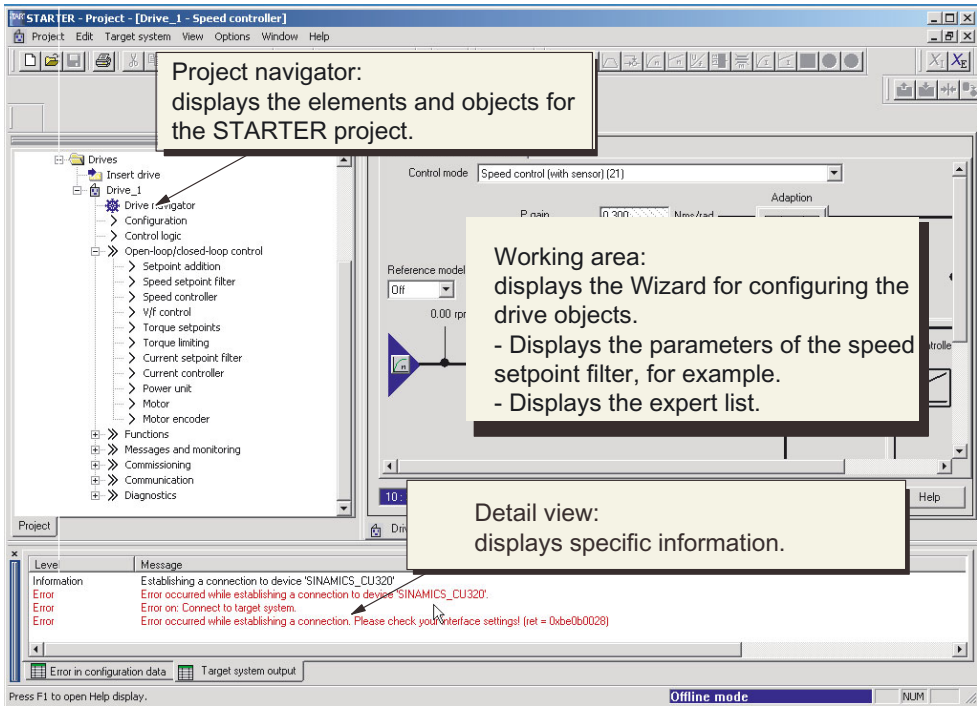


Figure 2-4 The different areas of the STARTER user interface

2.3.2 BICO interconnection procedure in STARTER

Introduction

You can parameterize the drive settings on the Motor Module in OFFLINE mode via STARTER by means of BICO interconnection. Parameterization can be carried out via the following means:

- Expert list
- Graphical screen interface

The steps described below explain the BICO interconnection procedure in STARTER.

Expert list

When carrying out BICO interconnection via the expert list, proceed as follows:

You want to interconnect parameter **p0840** of the control word with r parameter **r8890[0]**, for example.

1. In the project navigator, call up the expert list as follows: **Drive_1 > right-click > Expert > Expert list**.
2. Search for parameter **p0840**.

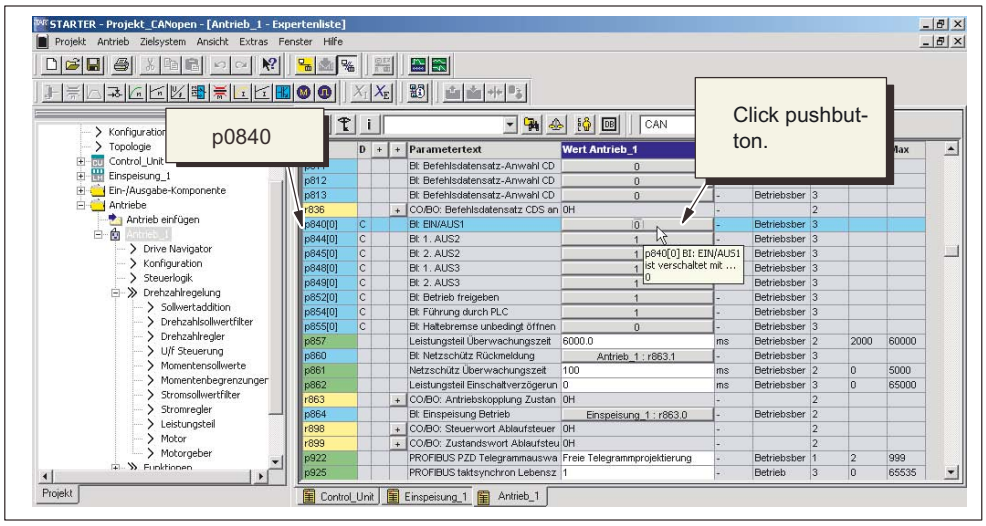


Figure 2-5 Interconnect 1

3. Click the pushbutton to interconnect with an r parameter.

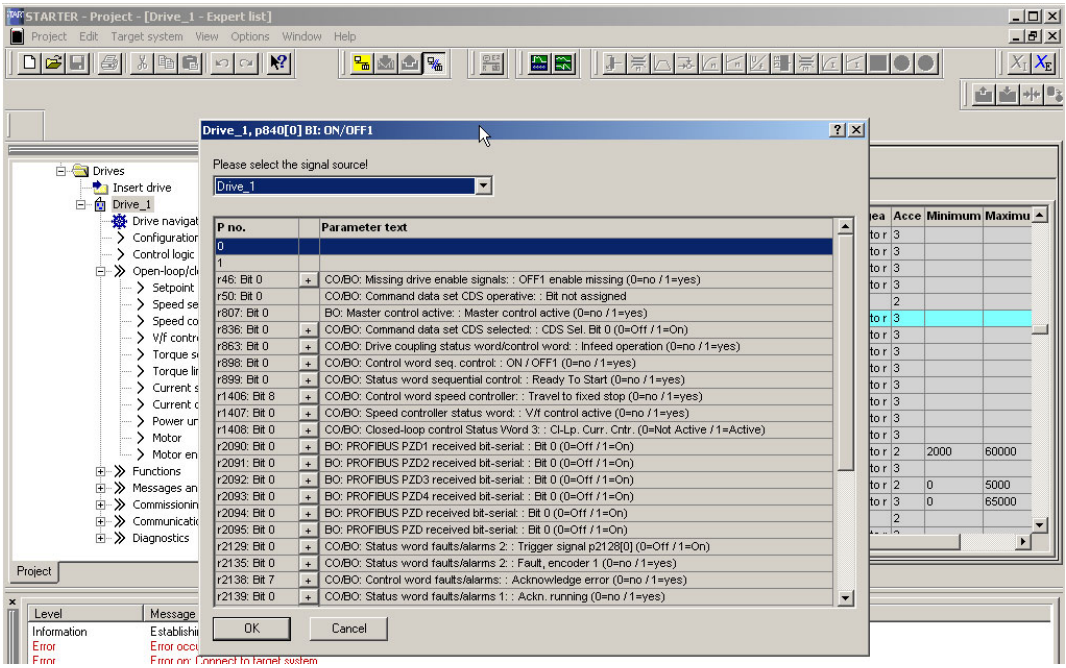


Figure 2-6 Interconnect 2

4. A list from which you can select the available r parameters is now displayed.

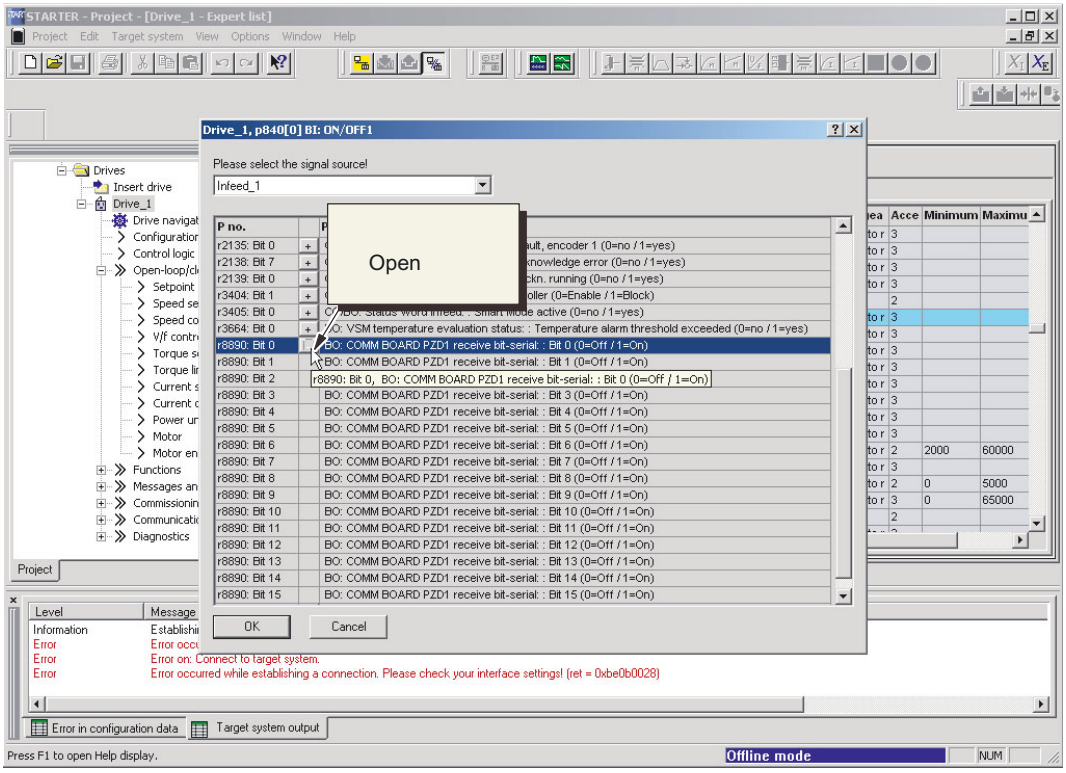


Figure 2-7 Interconnect 3

5. Open the 16 bit of r parameter **r8890**.

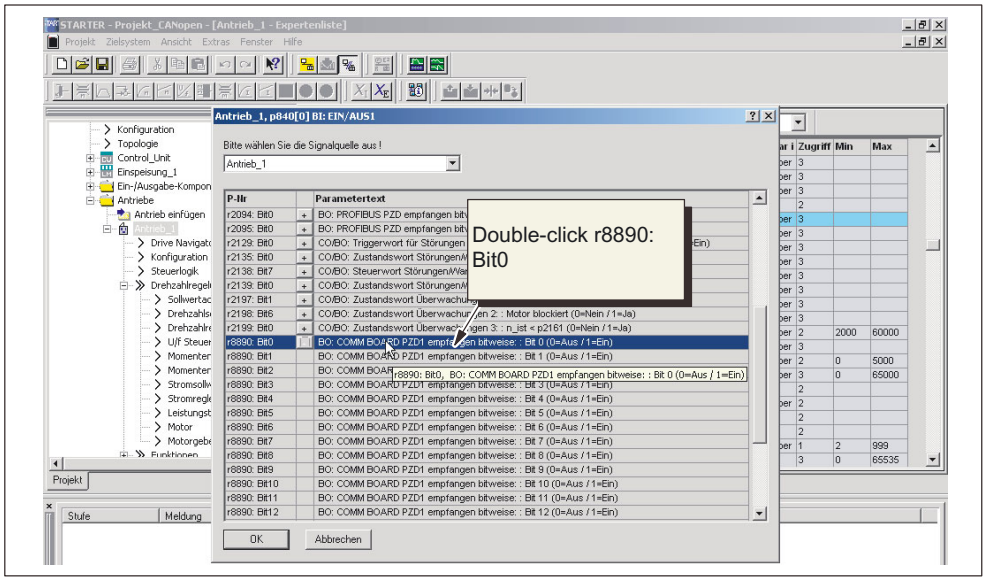


Figure 2-8 Interconnect 4

6. Double-click **r8890: Bit0.**

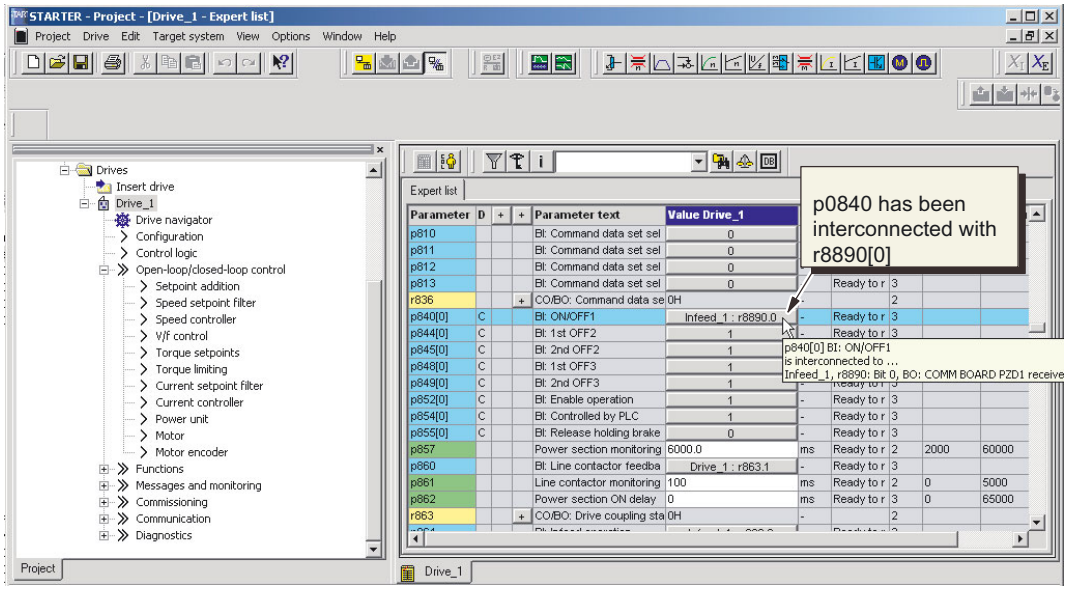


Figure 2-9 Interconnect 5

7. In the expert list, you can now see that **p0840** has been interconnected with r parameter **r8890[0]**.

Graphical screen interface

When carrying out BICO interconnection via the graphical screen interface, proceed as follows:

For the set velocity, which is a 32-bit data type, you want to interconnect p parameter **p1155 [0]** for "Speed setpoint 1" with r parameter **r8860 [1]**, for example.

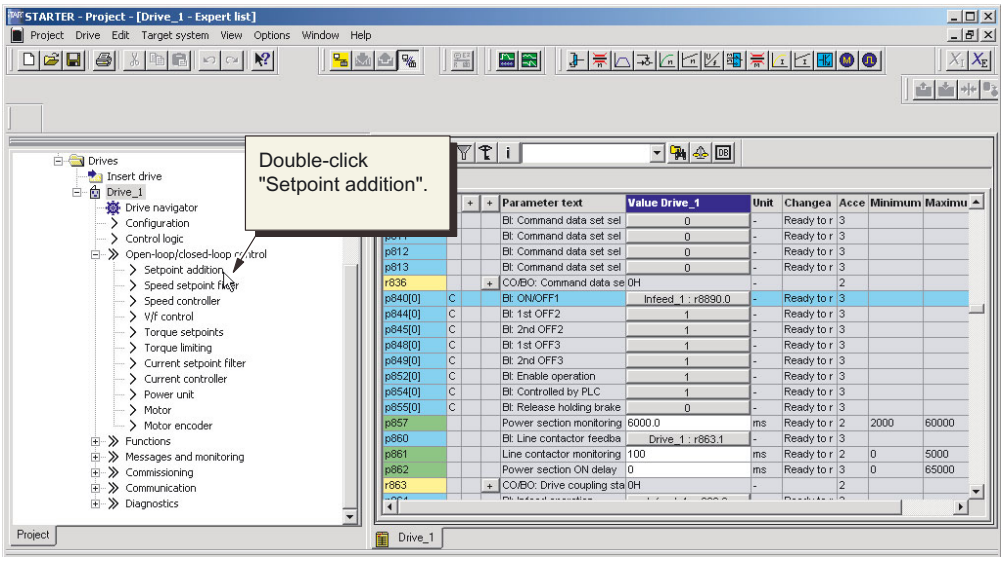


Figure 2-10 Interconnection via graphical screen interface 1

1. In the project navigator under **Drive_1 > Open-loop/closed-loop control**, double-click the selection **Setpoint addition**.

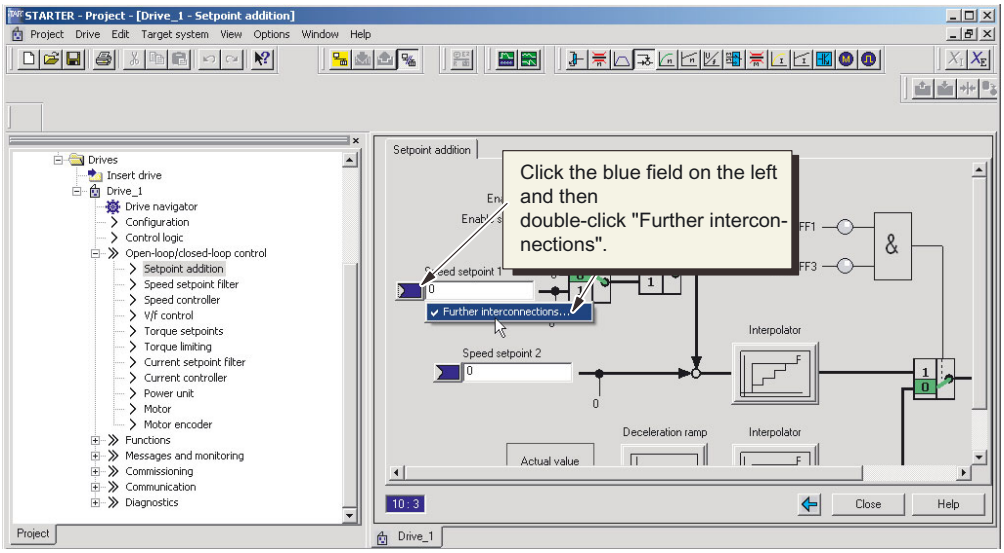


Figure 2-11 Interconnection via graphical screen interface 2

2. Click the **blue** field to the left of the field for **Speed setpoint 1** and then double-click the selection **Further interconnections**, which is now displayed.

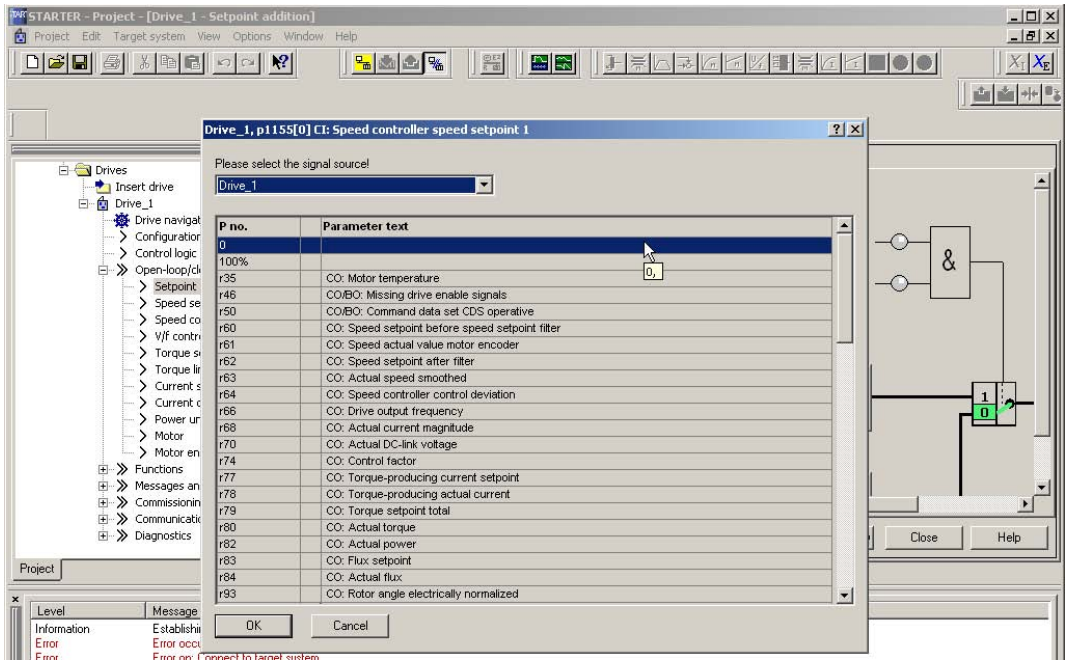


Figure 2-12 Interconnection via graphical screen interface 3

3. A list from which you can select the available r parameters is now displayed.

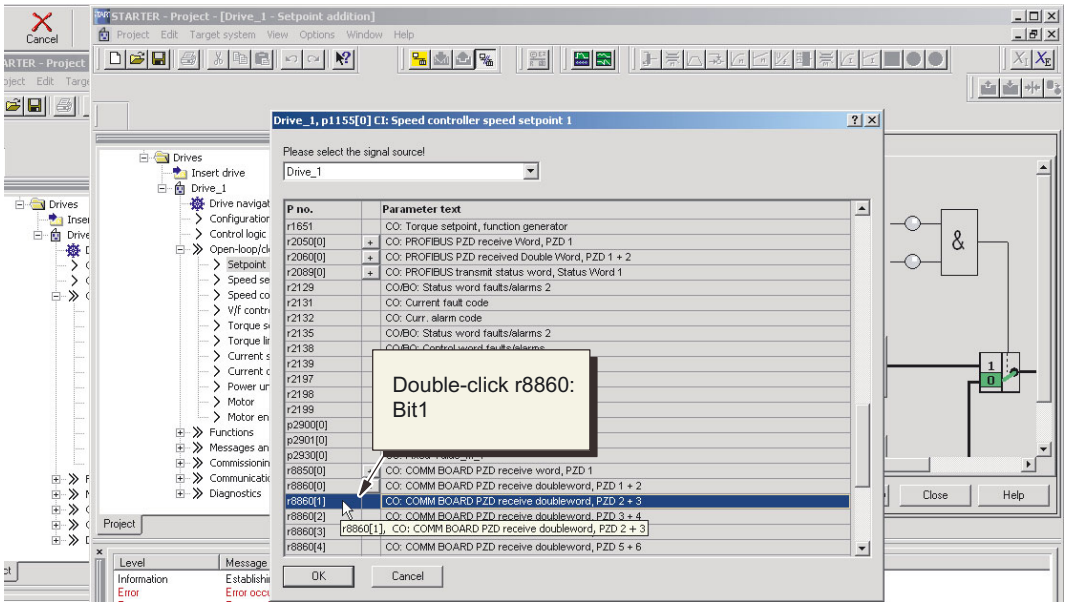


Figure 2-13 Interconnection via graphical screen interface 4

4. Double-click **r8860: Bit1**.

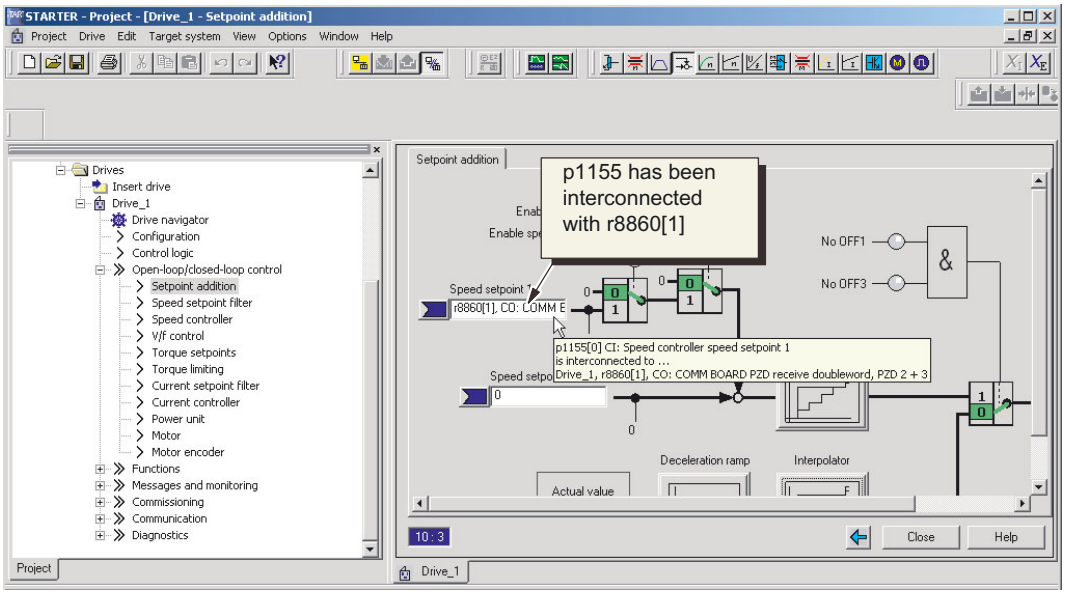


Figure 2-14 Interconnection via graphical screen interface 5

5. In the graphical screen interface, you can now see that **p1155** has been interconnected with r parameter **r8860[1]**.

See also

What is a BICO interconnection in conjunction with CANopen? (Page 1-19)

Interconnecting the receive and transmit buffers (overview) (Page 3-37)

2.3.3 Entering parameter values in the expert list

Introduction

Note

Parameters can be entered and displayed in the expert list as **decimal** or hexadecimal values.

Example

If you want to process decimal values as **hexadecimal** or **binary** values, use the calculator on your PC and carry out the following:

1. Copy the decimal value in the expert list to your calculator.
2. Display this decimal value on your calculator as either a hexadecimal or binary value.
3. Change the value.
4. Use your calculator to convert the value back to a decimal value.
5. Copy this decimal value to the expert list.

Commissioning

3

Section content

This section shows you how to carry out initial commissioning for the CANopen interface in the SINAMICS S120 drive line-up using the STARTER commissioning tool.

This section first looks at the initial commissioning procedure and describes a sample configuration.

A table is provided showing the individual commissioning steps in which the current step is highlighted in **bold**.

The initial commissioning procedure described here takes place with STARTER in ONLINE mode. At the end of each step, notes are included (if necessary) to explain how the procedure differs in OFFLINE mode.

Requirements

Before carrying out the commissioning steps described in this section, make sure you have read the section "Prerequisites for commissioning".

3.1 Initial commissioning procedure

Initial commissioning

The following steps must be carried out during the initial commissioning procedure for the CANopen interface in the SINAMICS S120 drive line-up:

Table 3-1 CANopen initial commissioning

Step	Procedure
1	Make the hardware settings on CBC10.
2	Configure the drive unit using the STARTER commissioning tool in ONLINE mode.
3	Configure the COB IDs and process data objects for the receive and transmit message frames.
4	Interconnect the receive and transmit buffers.
5	In ONLINE mode, load the projects from the drive unit to the PC/PG and save.

See also

Making the hardware settings on CBC10 (Page 3-4)

Configuring the drive unit with STARTER (overview) (Page 3-6)

Configuring the COB IDs and process data objects for the receive and transmit message frames (overview) (Page 3-25)

Interconnecting the receive and transmit buffers (overview) (Page 3-37)

Load the projects from the drive unit to the PC/PG in ONLINE mode and save them (Page 3-49)

3.2 Sample configuration

Overview

Note

The commissioning procedure described here is based on the sample configuration (as illustrated in the diagram below) in a SINAMICS S120 drive line-up for:

- One infeed (Active Line Module)

- Two drives (Single Motor Module)

For a more detailed description of this configuration, see the section "Initial commissioning using servo as an example" in the Installation and Start-Up Manual SINAMICS S120 /IH1/.

This Installation and Start-Up Manual extends the description of "Initial commissioning using servo as an example" to include a description of the initial commissioning procedure for the CANopen communication interface with the CBC10 Communication Board.

The following diagram illustrates the drive objects. From the point of view of CANopen and with this arrangement:

- Single Motor Module 1 is the first SINAMICS drive object
- Single Motor Module 2 is the second SINAMICS drive object

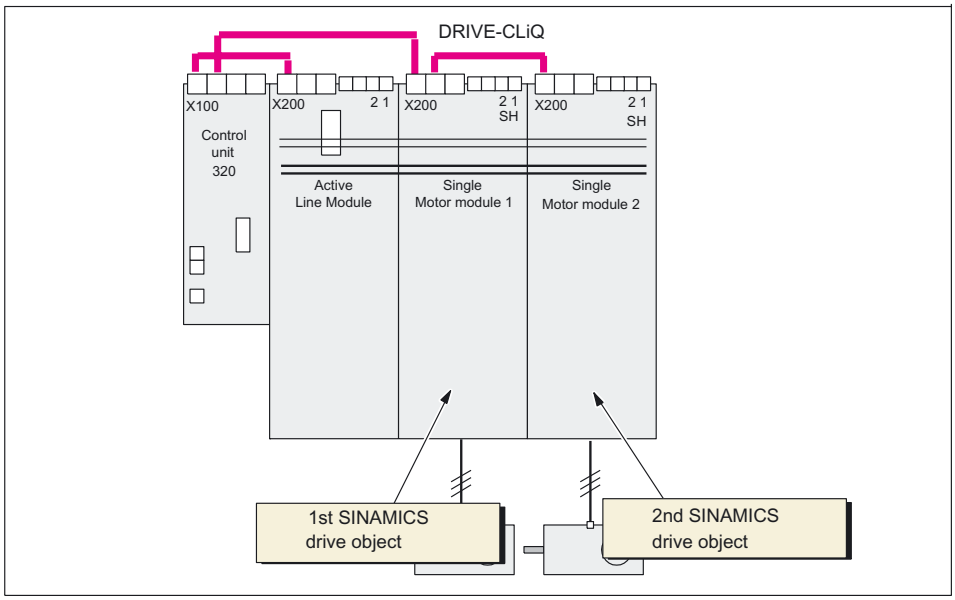


Figure 3-1 Components (example)

3.3 Making the hardware settings on CBC10

Initial commissioning: procedure

In the table below, the current commissioning step is highlighted in **bold**:

Table 3-2 CANopen initial commissioning

Step	Procedure
1	**Make the hardware settings on CBC10.**
2	Configure the drive unit using the STARTER commissioning tool in ONLINE mode.
3	Configure the COB IDs and process data objects for the receive and transmit message frames.
4	Interconnect the receive and transmit buffers.
5	In ONLINE mode, load the projects from the drive unit to the PC/PG and save.

Carrying out the commissioning step

To ensure that data can be transferred reliably via the CAN bus, switches S1/S2 on the CBC10 Communication Board must be set accordingly (see table below). Set the following:

- Bus terminating resistor
- Operation with/without ground

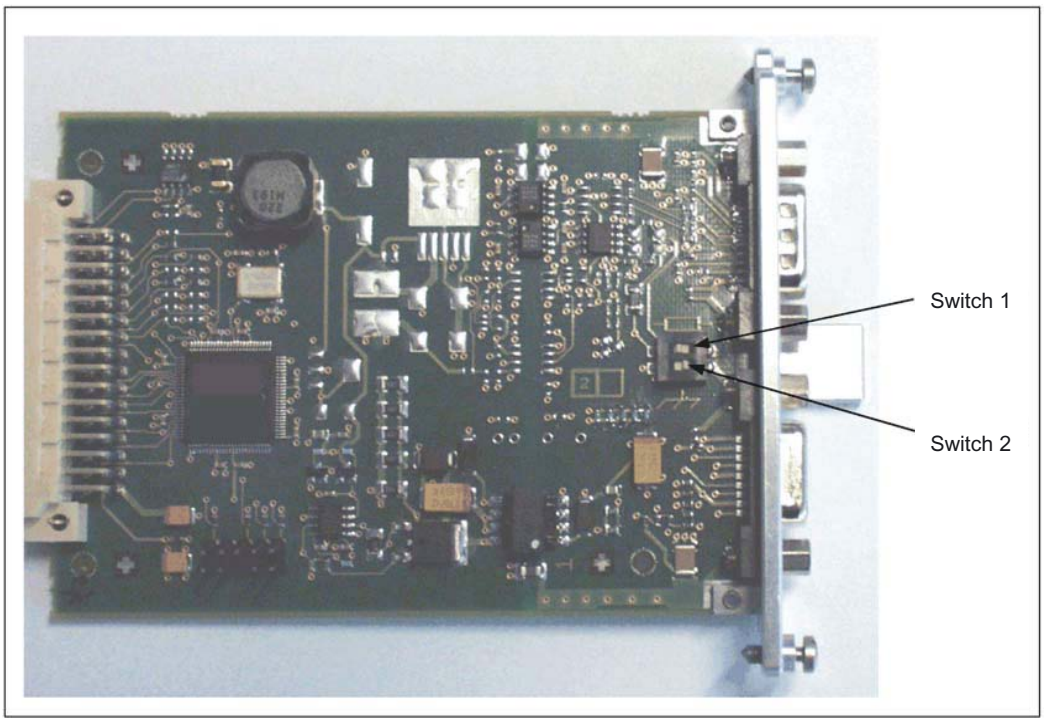
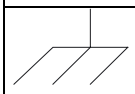


Figure 3-2 Switch S1/S2

Table 3-3 2-pin SMD DIL switch

ID on the board	Switch	Function	Switch position		Default
	S1	Bus terminating resistor (120Ω)	OFF	Inactive	OFF
			ON	Active	
	S2	Operation with/without ground	OFF	Ground-free operation	OFF
			ON	Operation with ground	

Note

In a line-up comprising more than one drive unit (bus nodes), switch S1 must be set to **On** for the last bus node and **Off** for the preceding bus node.

3.4 Configuring the drive unit using the STARTER commissioning tool

Initial commissioning: procedure

In the table below, the current commissioning step is highlighted in **bold**:

Table 3-4 CANopen initial commissioning

Step	Procedure
1	Make the hardware settings on CBC10.
2	**Configure the drive unit using the STARTER commissioning tool in ONLINE mode.**
3	Configure the COB IDs and process data objects for the receive and transmit message frames.
4	Interconnect the receive and transmit buffers.
5	In ONLINE mode, load the projects from the drive unit to the PC/PG and save.

Carrying out the commissioning step

Configure the drive unit in STARTER by carrying out the following steps:

- Search for the drive unit ONLINE.
- The component topology and configuration of the drive unit are determined automatically.
- Configure the motor.
- Configure the interface for the CBC10 Option Board on the Control Unit.
  - CAN interface
  - PDO message frame
  - Monitoring
- Load the project to the drive unit.

Note

When carrying out these steps, refer to the Installation and Start-Up Manual /IH1/ for SINAMICS S120.

3.4.1 Searching for the drive unit ONLINE

Introduction

The SINAMICS firmware is able to recognize the actual topology automatically and store it in the appropriate parameters.

Steps

To ensure that the drive unit configuration is identified automatically, open a new project in STARTER: Proceed as follows:

1. Click the STARTER icon or choose **Start > Programs > STARTER** in the Windows Start menu to launch STARTER.
   The STARTER Project Wizard is displayed.
2. Choose **Find drive units online....**

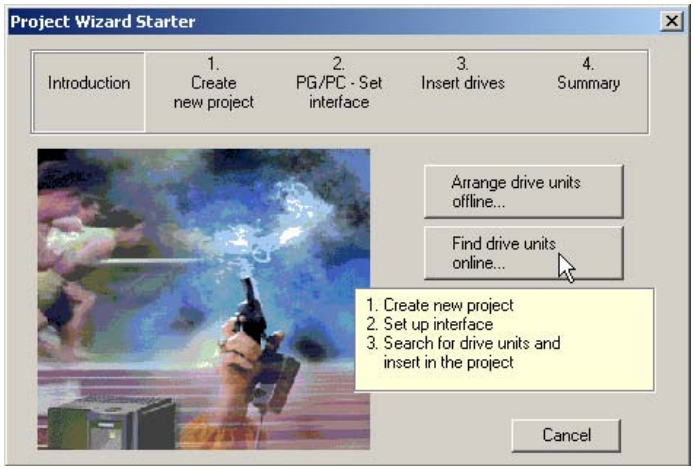


Figure 3-3 Find drive units online...

3. The Wizard guides you through the procedure for creating a new project. In the next dialog box, enter a name for the project (e.g. **Project_CANopen_0** and click **Continue >.**
4. The Project Wizard searches for the drive unit ONLINE and inserts it in the project. Click **Continue >.** The Wizard displays a summary of the project.
5. Choose **Fertig stellen (Complete)**. The new project and drive unit are displayed in STARTER.

Note

The system searches for drive units or, more precisely, Control Units; in other words, if more than one Control Unit exists in the system, more than one drive unit is found. The peripheral components of a drive unit (Control Unit, Active Line Module, and so on) are not displayed until you carry out **Automatic configuration**.

3.4.2 Determining the component topology and configuration of the drive unit automatically

Introduction

Once you have created the project and entered the drive unit with its bus address (e.g. 126) ONLINE, you have to enter the associated component topology and drive unit configuration ONLINE.

Steps

1. Select the drive unit **Drive_Unit_Adr126** in the project navigator.
2. Choose **Connect to target system**. An ONLINE connection is established and the dialog box **Online/OFFLINE comparison** is displayed.
3. Click **Close** and, if necessary, restore the factory settings.
4. Select the drive unit **Drive_Unit_Adr126** in the project navigator.
5. Choose **Restore factory settings** (see screenshot below).

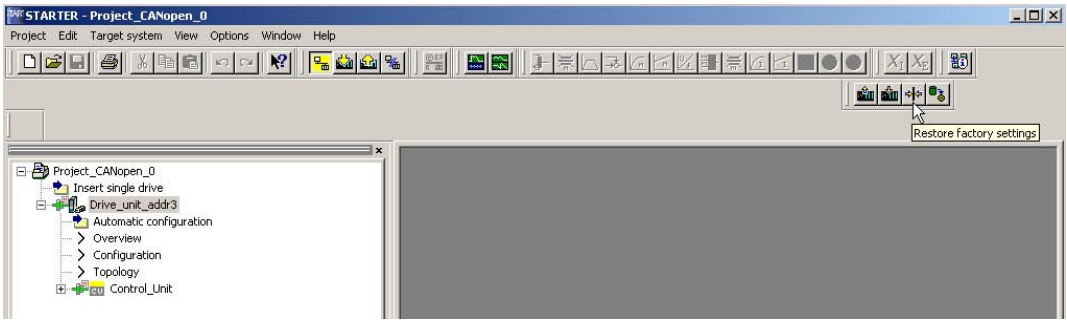


Figure 3-4 Restore factory settings

6. Confirm the following queries and messages by choosing **OK**:
   - "Restore factory settings?" dialog box
   - "The factory settings have been restored" dialog box
   - "The data has been successfully copied from RAM to ROM" dialog box

7. In the project navigator, double-click **Automatic configuration** under the drive unit.

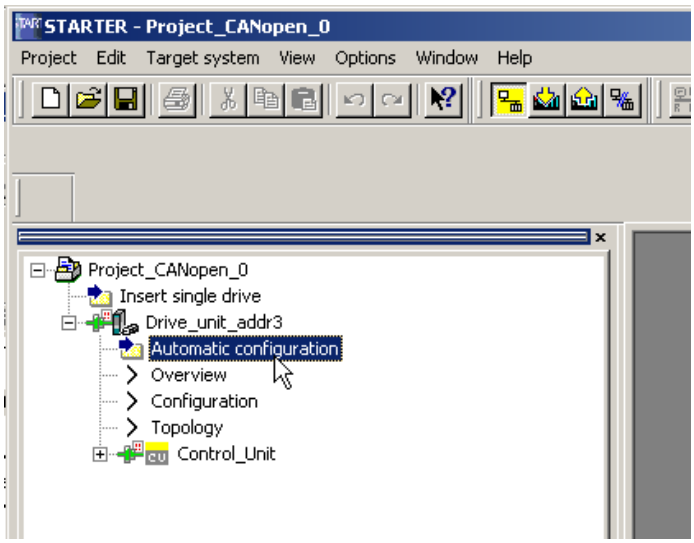


Figure 3-5 Automatic configuration

8. In the "Automatic configuration" dialog box, click **Start automatic configuration**. STARTER automatically searches for all drive unit components that are connected properly and then uploads them. In this case, it has recognized two drive objects.
9. In the "Drive object type" dialog box, choose **Servo**.

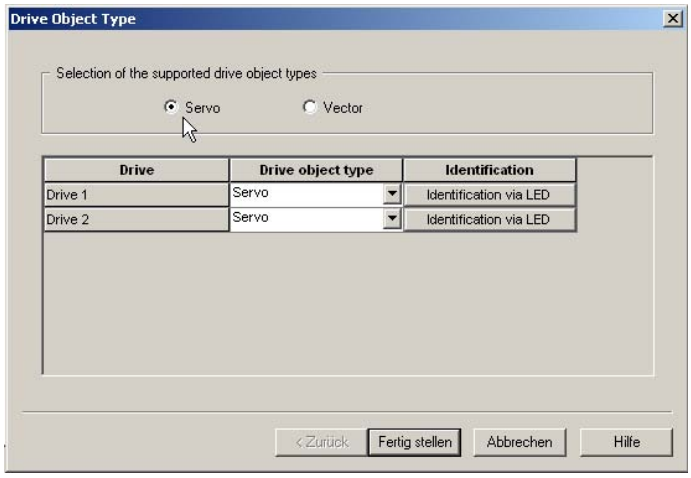


Figure 3-6 Drive object type

10. Click **Fertig stellen (Complete)**. The system loads the data from RAM to ROM and to the PG.

11. Another message is output to inform you that the motors also have to be configured. Confirm this message by choosing **OK.**

Figure 3-7 Message prompting you to configure the motor

12. Initialization is complete. Click **Close** in the "Automatic configuration" dialog box.

Automatic configuration is complete

Automatic configuration is complete. In the project navigator, all the drive unit components that have been found (e.g. **Control Unit, CBC10, infeed**, and **drive**) are displayed.

The following screenshot shows a configured drive in STARTER. In the work area, the view showing the set and actual topology has been selected. Among other things, you can see that the CBC10 Communication Board in the drive line-up has been recognized.

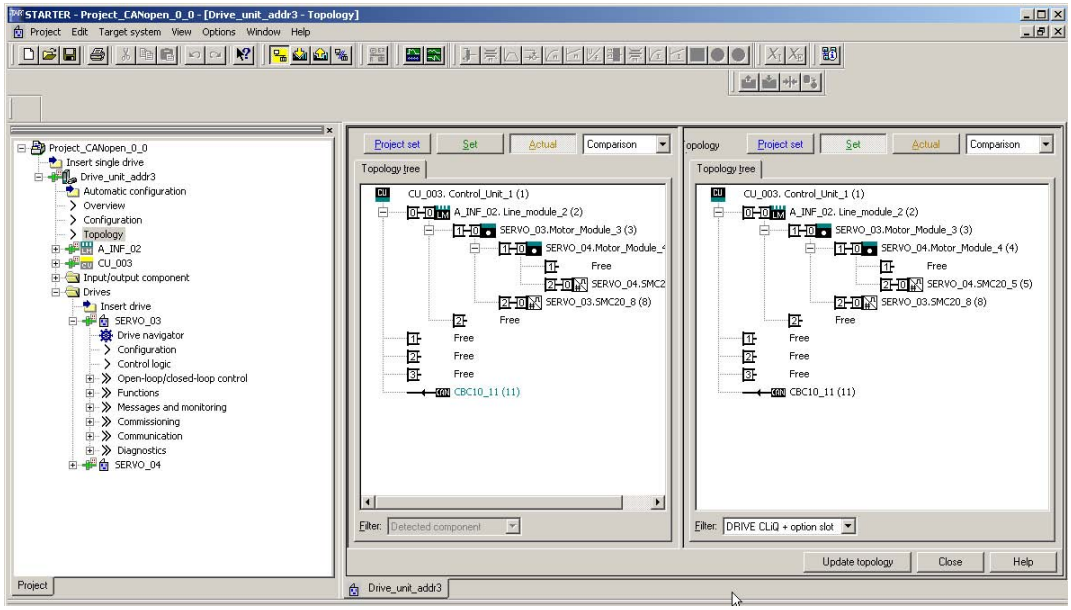


Figure 3-8 Set and actual topology drive configuration

You will now be shown how to configure the motor OFFLINE in STARTER.

3.4.3 Configuring the motor

Introduction

In the previous steps, you automatically determined the component topology and configuration of the drive unit and integrated the data in the STARTER project.

Steps

In the following steps, you will be shown how to configure the drive motor and encoder.

Note

You only have to change the motor and encoder configuration; leave the infeed and so on as it is.

1. Choose **Disconnect from target system...**. The modified data is loaded from RAM to ROM and to the PG.
   The motors are configured in OFFLINE mode and are then loaded to the target system in ONLINE mode.
2. In the project navigator, choose the Drives folder and double-click **Configuration** under the relevant drive (see screenshot below).

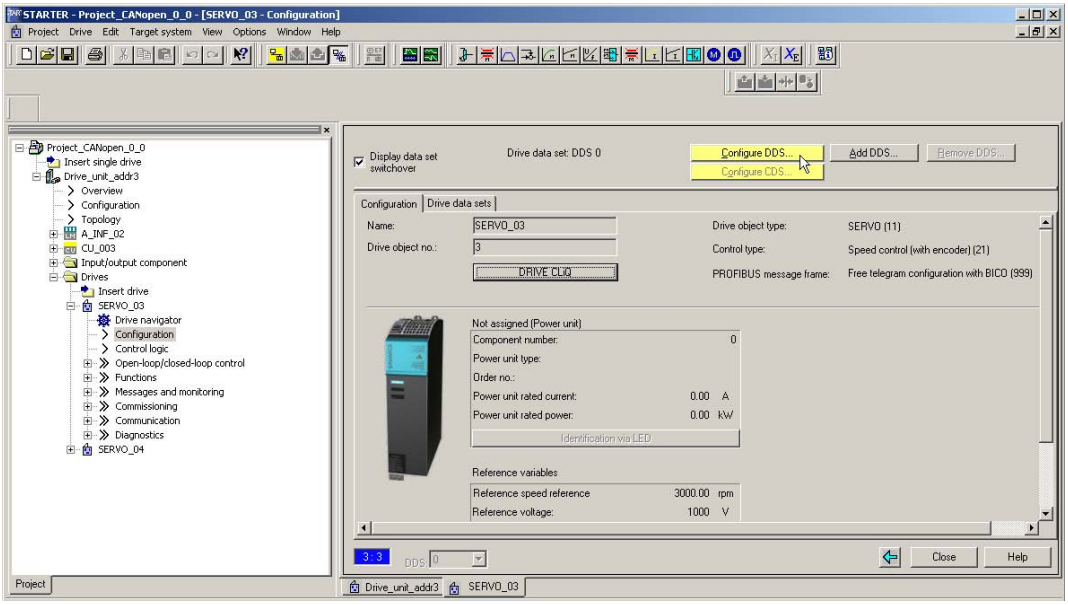


Figure 3-9 Configure the drive (parameterize the motor)

3. In the dialog box that is displayed, choose **Configure DDS...**

4. In the dialog box that is displayed (see screenshot below), you can define whether or not the drive object (function module) is operated with or without an extended setpoint channel. The commissioning procedure described here is carried out **without** an extended setpoint channel (ramp-function generator). The field for the extended setpoint channel must be deactivated (as shown below).

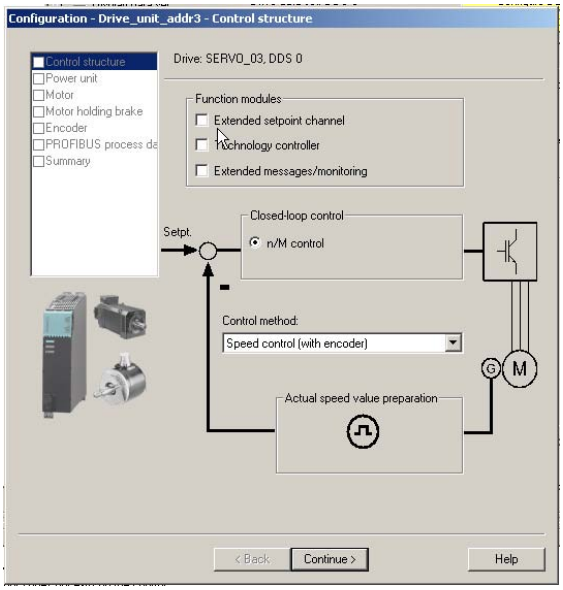


Figure 3-10 Extended setpoint channel

5. You only configure the motor and the encoder. Work through the Wizard by choosing **Continue >** until you reach the point at which you configure the motor (see screenshot below).

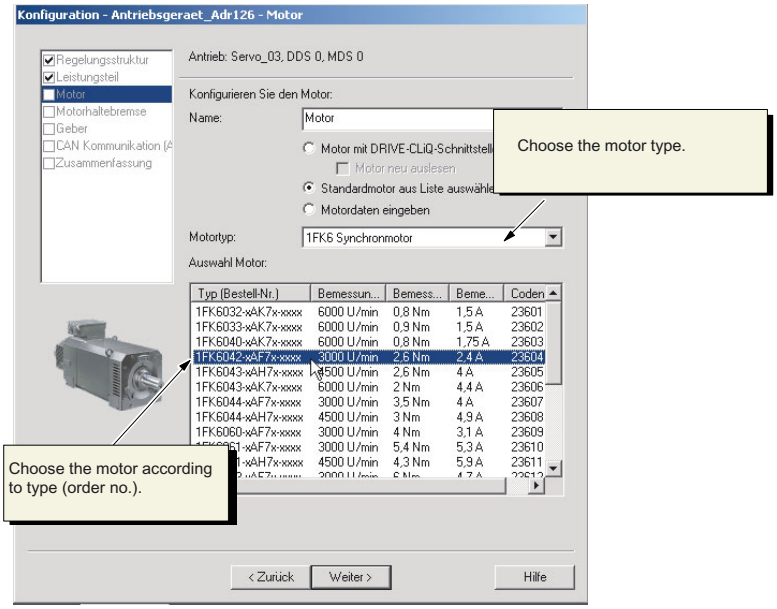


Figure 3-11 Configure the motor

6. Choose the motor type and the motor according to the type (order no.) (see the type plate).

7. Click **Continue >** until you reach the point at which you configure the encoder.
8. Choose the motor encoder and work through the Wizard by choosing **Continue >** until the dialog box containing a summary is displayed.
9. Click **Fertig stellen (Complete)**.
10. Once you have configured the motor and the encoder for the first drive, repeat these steps for the second drive.

When you have completed configuring the drive unit OFFLINE, carry out the following steps to parameterize the CANopen interface on the Control Unit.

3.4.4 Configuring the interface for the CBC10 Option Board on the Control Unit

Prerequisite

You have configured the drive unit with the CBC10 Option Board in STARTER and connected the drive OFFLINE with STARTER.

In the project navigator, double-click **Drive unit_Adr_126 > Control_Unit > CAN Option Board > Configuration**.

You can now configure the CBC10 Option Board in the dialog box that is displayed.

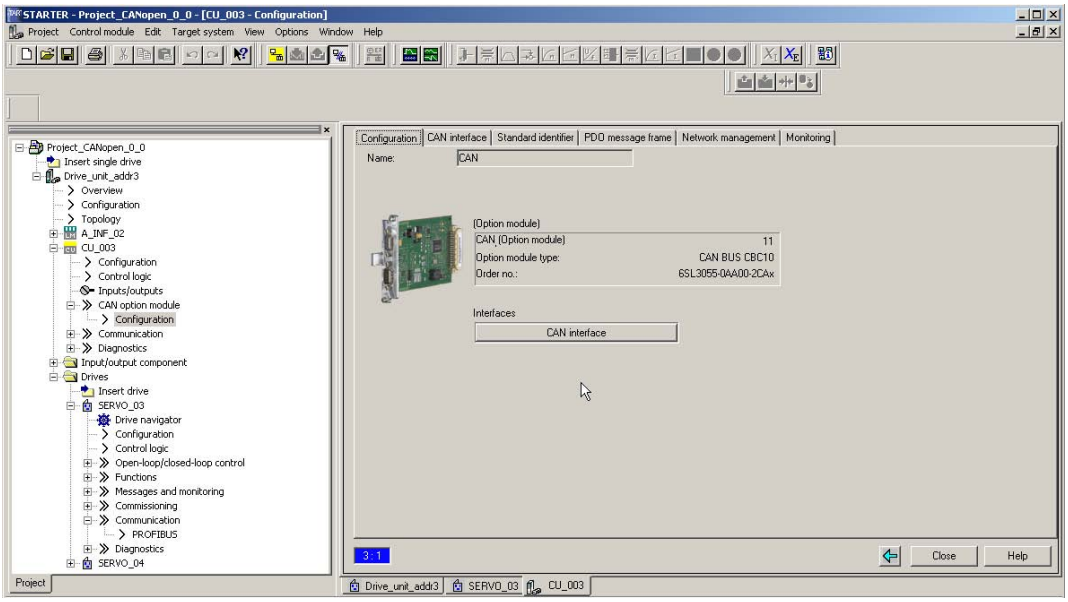


Figure 3-12 Configuring the CBC10 Option Board

Carrying out the commissioning step

On the Control Unit CU320, configure the following transmission properties for the CBC10 Option Board:

- Transmission rate
- CAN bus address (node ID)
- Number of PDO message frames
- Node monitoring (heartbeat, node guarding)

3.4.4.1 CAN interface

Steps

On the **CAN interface** tab, enter the transmission rate and the CAN bus address/node ID.

1. Select the **CAN interface** tab (see screenshot below).

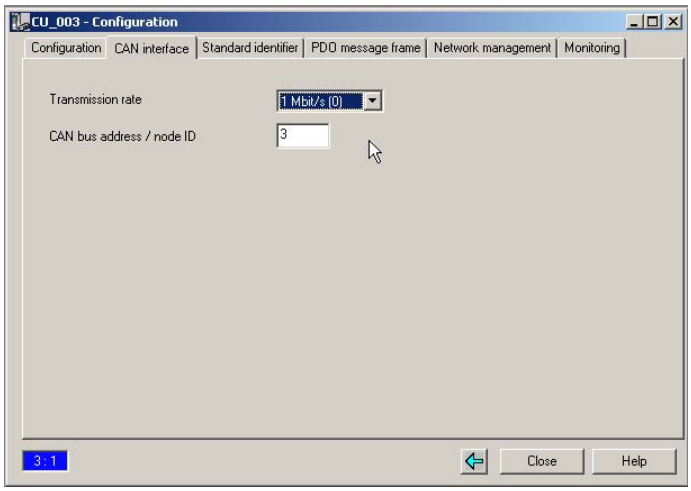


Figure 3-13 CAN interface

2. Enter a transmission rate of **1 MBit/s** for commissioning, for example.
   The factory setting is 20 kBit/s.
3. Two options are available for the bus address/node ID:
   - In this dialog box, you can set a value of between 1 and 126 if the address switch on the Control Unit CU320 ("DP address") is set to 0 or 127.

Note

If the address switch is set to 1...126, values that were entered here in OFFLINE mode are not downloaded.

   - You can set a value directly via the address switch on the Control Unit CU320.
     The following diagram shows an example for address **5**.

Example

Setting the bus address via the address switch on the Control Unit.

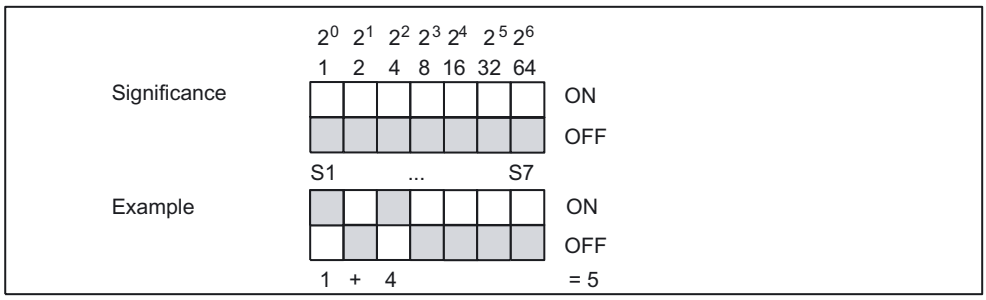


Figure 3-14 Example: bus address via the address switch on the Control Unit

The following information must be taken into account!

Note

Permissible CAN bus address: 1...126.

The address setting on the switch is displayed in **p8620.0**.

If the switch setting is changed, the new setting does not become effective until POWER ON.

The factory settings are "ON" or "OFF" for all switches.

While the SINAMICS S120 is being ramped up, the address switch is queried first in order to set the bus address. If the switch setting is 0 or 127, the address can be set via parameter **p8620.0**.

If the address is set to a valid node address (1...126), this is copied to parameter **p8620.0** and displayed.

3.4.4.2 PDO message frame

Introduction

This description of the initial commissioning procedure uses a predefined setting ("predefined connection set") with a fixed assignment of SDOs and PDOs as an example, whereby the device features the following for each SINAMICS drive object (in this case, the Single Line Module):

- 4 receive PDOs (channels)
- 4 transmit PDOs (channels)
- An SDO pair for reading and writing the objects (acyclic configuration) is available with a "predefined connection set" and for free PDO mapping.

As a result, the device (Control Unit) is to be equipped with 4 receive and 4 transmit channels for each Motor Module.

Steps

In the **PDO message frame**, enter the number of transmit and receive PDO message frames (channels) for each drive object.

1. Select the **PDO message frame** tab (see screenshot below).

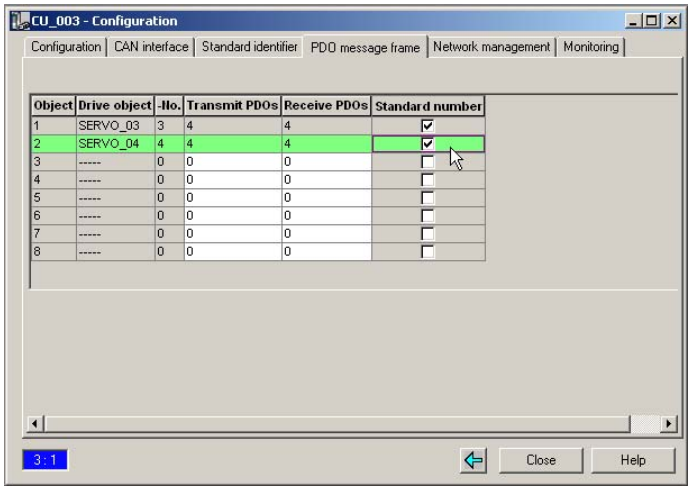


Figure 3-15 Standard number

2. Unless already defaulted, enter **4** transmit PDOs and **4** receive PDOs for both **Drive_1** and **Drive_2**.

Note

The standard number (predefined connection set) is **4**. You can also enter this value for each drive via the **Standard number** field.

Commissioning in ONLINE mode

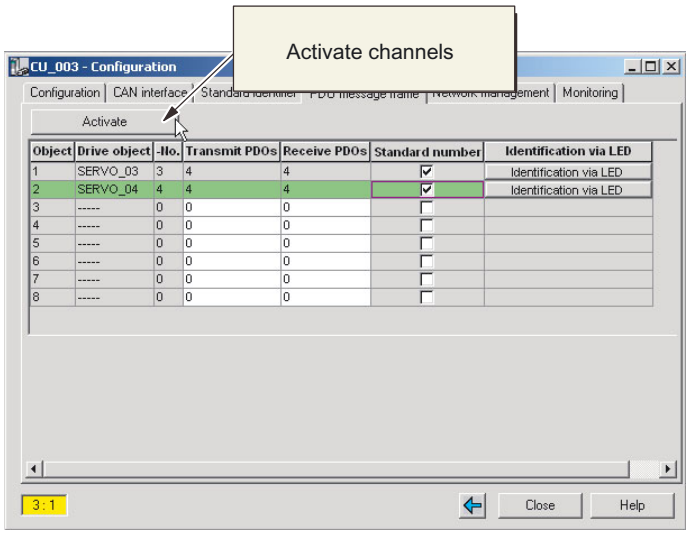


Figure 3-16 Activating the channels

Note

Once you have entered the channel assignments in ONLINE mode, confirm them in this dialog box by choosing **Channel assignment** (see screenshot above).

You can also confirm them in the expert list in parameter **p8741** by choosing **1** (see screenshot below).

In the project navigator, open the expert list for the Control Unit as follows: **Control Unit > right-click > Expert > Expert list**.

Search for parameter **p8741** for confirming the channel assignment and enter **1**.

In parameter **p8742**, you can see how many channels are still available (max. 25).

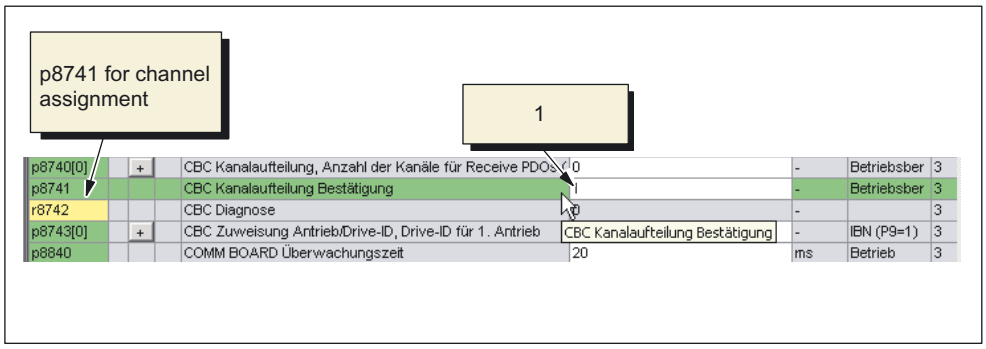


Figure 3-17 Confirm channel assignment

3.4.4.3 Monitoring

Introduction

SINAMICS S120 supports the following two optional monitoring services to ensure that the CANopen network nodes function properly:

- Heartbeat:
  SINAMICS S120 (producer) cyclically transmits (heartbeat time) its communication status on the CAN bus to the master application.
- Node guarding:
  SINAMICS S120 waits a certain time (node life time) for message frames from the master application and permits a specific number (life time factor) of failures within a specified time interval (node guard time).
  The node life time is calculated by multiplying the node guard time by the life time factor.

Note

Only one node monitoring service can be activated at any one time (either heartbeat or node guarding).

If both monitoring services are activated, node guarding is effective.

Steps

On the **Monitoring** tab, enter the required monitoring service (heartbeat or node guarding).

1. Select the **Monitoring** tab (see screenshot below).

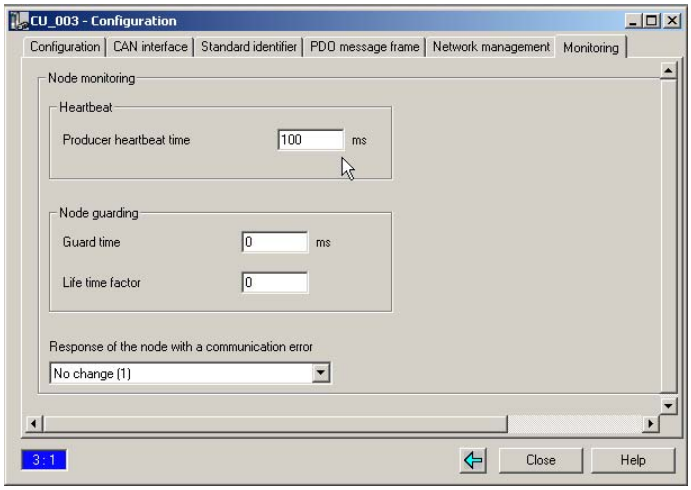


Figure 3-18 Monitoring

2. The default commissioning value for the **heartbeat** monitoring service could be **100 ms**, for example. Enter this value (unless it has already been entered).
3. The default commissioning values for the **node guarding** monitoring service could be:
   - Time interval (guard time): **100 ms**
   - Number of failures (life time factor): **3**
     Enter these values (unless they have already been entered).

The CANopen interface is now parameterized. To load the project to the target system in ONLINE mode, carry out the following steps.

Note

SINAMICS S120 parameters **p8609** and **p8641** govern the behavior of the drive/CAN node in the event of a CAN communication error or device malfunction.

These parameters have the following factory settings:

p8609 = 1, => no change

p8641 = 0, => no response

3.4.5 Activating the "predefined connection set"

Introduction

Note

The STARTER commissioning tool is in OFFLINE mode.

Activating the "predefined connection set"

To activate the "predefined connection set", carry out the following steps:

1. In the project navigator, open the expert list for the drive as follows: **Drive_1 > right-click > Expert > Expert list**.

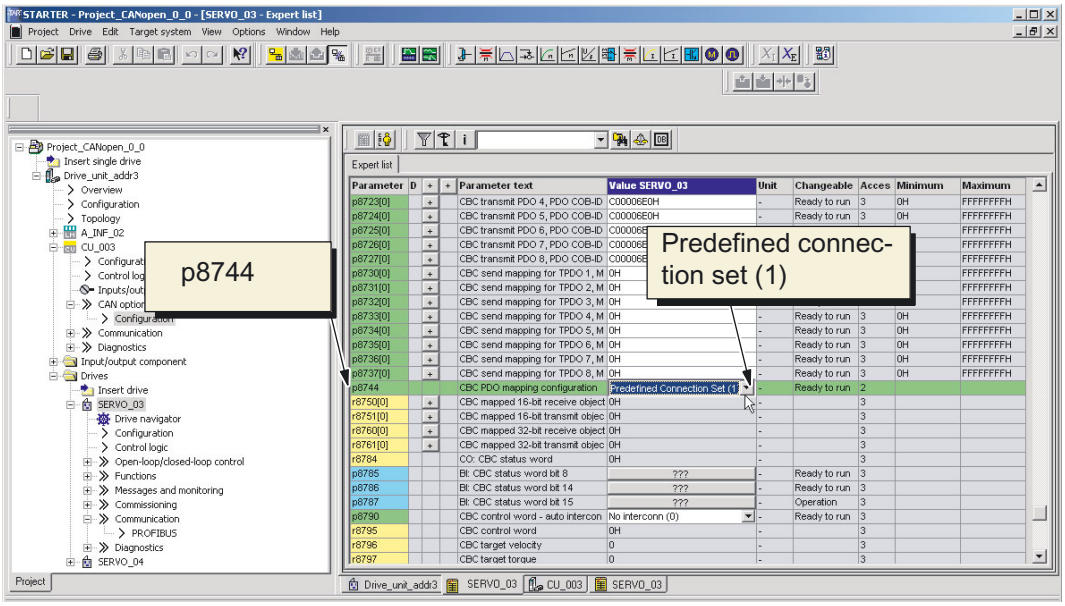


Figure 3-19 Predefined connection set

2. Find parameter **p8744** (see screenshot above).
3. Choose "Predefined connection set (1)" to set the "predefined connection set".
4. Repeat steps 1 to 3 for the next SINAMICS drive object, **Drive_2**.

3.4.6 Loading the project to the drive unit

Introduction

To load the project to the drive unit, proceed as follows:

Steps

1. Click **Connect to target system**. An ONLINE connection is established and an ONLINE/OFFLINE comparison takes place. If any discrepancies are identified, they are displayed (see screenshot below).

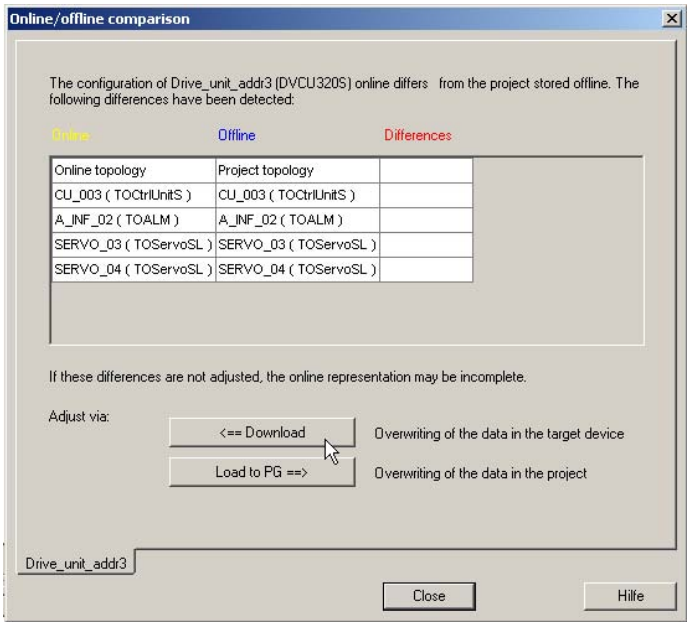


Figure 3-20 ONLINE/OFFLINE comparison

2. You changed the data OFFLINE and now have to load it to the target system. Carry out the following:
   - **<-- Download** in the "ONLINE/OFFLINE comparison" dialog box
   - When the system asks "Are you sure?", click **Yes**. The system now starts loading the data.
   - When the system informs you that the data was successfully loaded to the target system, click **OK**.
   - Click **OK** for "Load from RAM to ROM".

3. Discrepancies were identified again during the ONLINE/OFFLINE comparison. Now click **Load to PG -->** (see screenshot below).

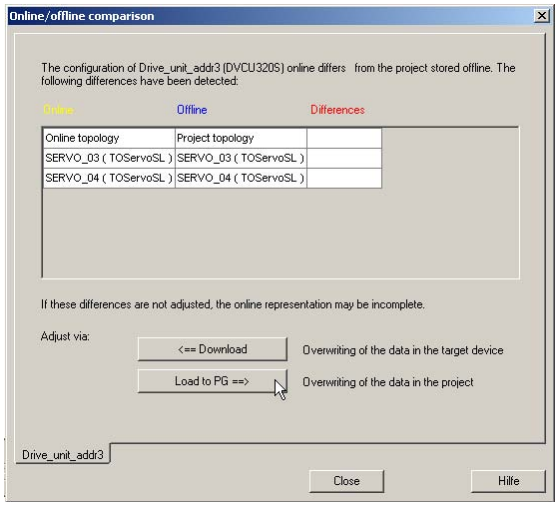


Figure 3-21 Load to PG

4. Load the new data from the drive unit to the PG. Carry out the following:
   - When the system asks "Are you sure?", click **Yes**. The system now starts loading the data.
   - When the system informs you that the data was successfully loaded to the PG, click **OK**.
5. No further discrepancies are displayed in the ONLINE/OFFLINE comparison dialog box. Click **Close** (see screenshot below).

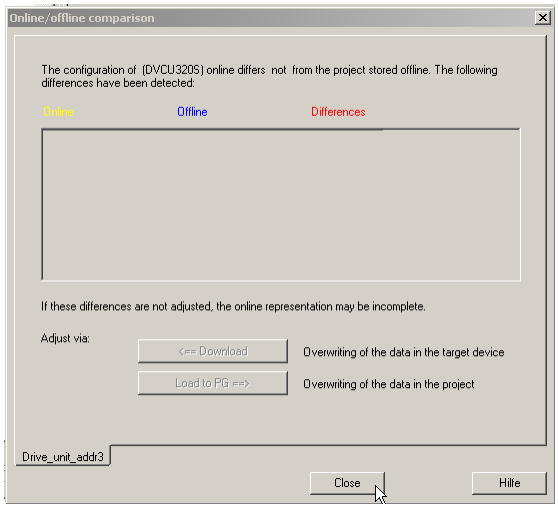


Figure 3-22 ONLINE/OFFLINE comparison OK

This completes the procedure for configuring the drive unit hardware with the CANopen interface. Before you start configuring the COB IDs and the process data objects for the receive and transmit message frames, however, you have to interconnect the infeed.

3.4.7 Interconnecting the infeed

Carrying out the commissioning step

In the current software version, the infeed (Active Line Module) cannot be addressed directly via the PDOs.

The DC link can still be activated via a signal source by means of Active Line Module parameter **p0840** (ON/OFF1), however, using the following BICO interconnection options:

- You can interconnect parameter **p0840** with a free bit at the digital input of the CU320 (e.g. r722 bit 15).
- You can interconnect parameter **p0840** with a free bit of the control word from Motor Module 1 (e.g. r8890 bit 15).

Note

In the sample configuration described here, the control word is present at the receive buffer in PZD receive word 1 (see also the function diagram in the appendix).

Note

The BICO interconnection procedure in STARTER is described in the section "BICO interconnection procedure in STARTER".

Steps

To interconnect parameter **p0840** (ON/OFF1) with the signal source (e.g. from the control word of Motor Module 1 (Drive_1)) from the infeed, carry out the following:

1. In the project navigator, choose parameter **p0840** ("search"): **Infeed > right-click > Expert > Expert list**.
2. Interconnect the p parameter with r parameter **r8890 bit 15** = PZD 1 from Drive_1 (SERVO_3).

Note

Please also refer to the commissioning instructions in the Installation and Start-Up Manual SINAMICS S120 /IH1/

(e.g. the DC link must be activated before the Motor Module is switched on).

3.5 Configuring the COB IDs and process data objects for the receive and transmit message frames.

Initial commissioning: procedure

In the table below, the current commissioning step is highlighted in **bold**:

Table 3-5 CANopen initial commissioning

Step	Procedure
1	Make the hardware settings on CBC10.
2	Configure the drive unit using the STARTER commissioning tool in ONLINE mode.
3	**Configure the COB IDs and process data objects for the receive and transmit message frames.**
4	Interconnect the receive and transmit buffers.
5	In ONLINE mode, load the projects from the drive unit to the PC/PG and save.

Carrying out the commissioning step

You need to commission the communication and mapping parameters for the individual drive objects. STARTER is connected to the drive unit ONLINE. For each drive object, you can freely configure (free PDO mapping) the following for the receive and transmit message frames:

- COB IDs
- Mapping the process data objects

Note

As already described in the introduction, the communication parameters, mapping parameters, and COB IDs are automatically predefined ("predefined connection set") for each drive object when the CANopen interface is commissioned for the first time. The following initial commissioning procedure with free PDO mapping assumes that the parameters have been predefined and describes how they can be changed. If you do not want to parameterize free PDO mapping, you can omit the steps described below and continue commissioning with the section "Interconnecting the receive and transmit buffers".

3.5.1 Predefined COB IDs and process data objects for the drive objects

Introduction

Before you start commissioning free PDO mapping, open the expert lists for the individual drive objects as follows:

1. In the project navigator, call up the expert list for drive object 1: **Drives > Drive_1 > right-click > Expert > Expert list**.
2. In the project navigator, call up the expert list for drive object 2: **Drives > Drive_2 > right-click > Expert > Expert list**.

The expert lists for both drive objects are displayed in STARTER. You can now see the automatically predefined parameters in the "predefined connection set".

Note

You assign COB IDs for each drive object in the expert list as of **p8700** for receive message frames and as of **p8720** for transmit message frames.

For each drive object in the expert list, the process data objects for mapping the message frames begin as of parameter **p8710** for receive message frames and as of parameter **p8730** for transmit message frames.

COB IDs and mapped process data objects for drive object 1

The following screenshots show the predefined COB IDs and mapped process data objects for transmitting and receiving drive object 1 in the expert list in STARTER.

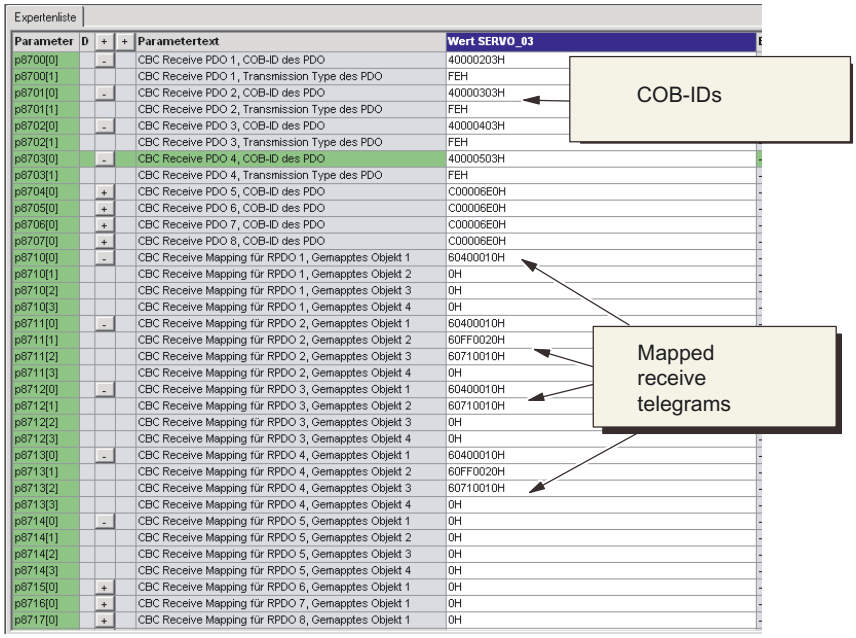


Figure 3-23 COB IDs and mapped process data objects for receive message frame: drive object 1

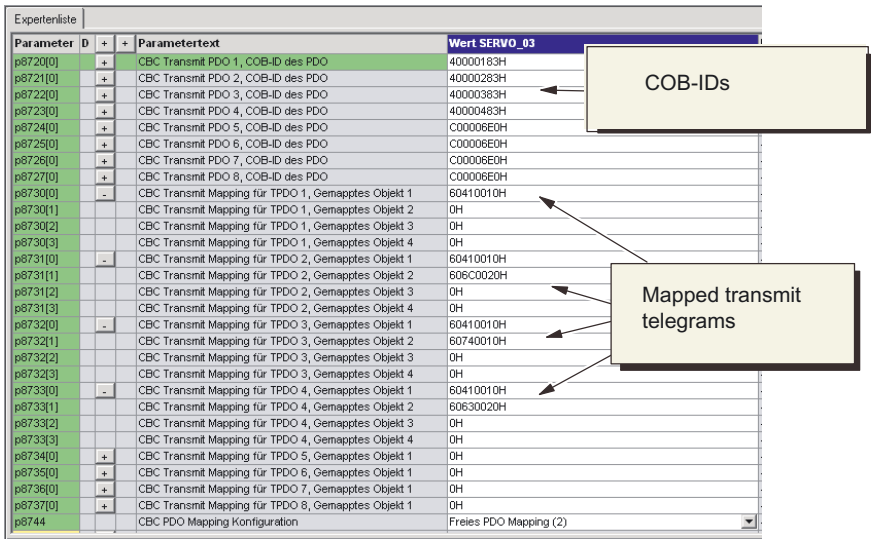
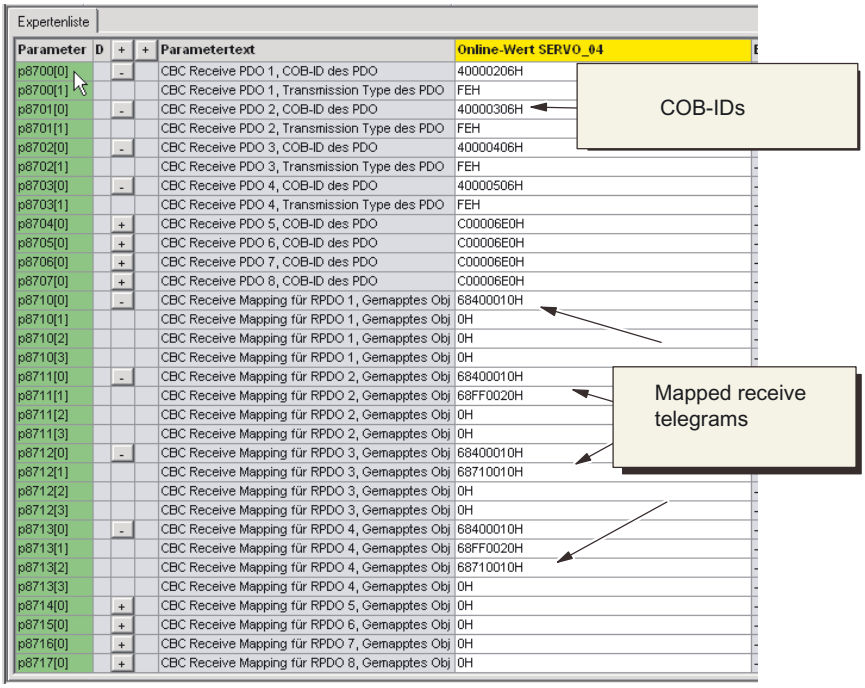


Figure 3-24 COB IDs and mapped process data objects for transmit message frame: drive object 1

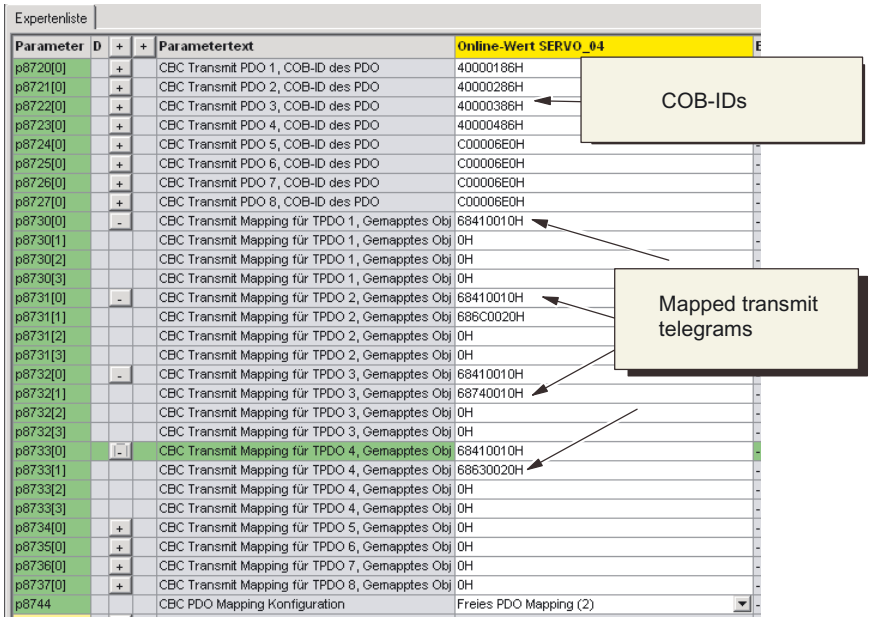
COB IDs and mapped process data objects for drive object 2

The following screenshots show the predefined COB IDs and mapped process data objects for transmitting and receiving drive object 2 in the expert list in STARTER. The process data objects are parameterized with an offset of 800 hex for the second drive object.



Parameter	D	+	+	Parametertext	Online-Wert SERVO_04
p8700[0]		-		CBC Receive PDO 1, COB-ID des PDO	40000206H
p8700[1]				CBC Receive PDO 1, Transmission Type des PDO	FEH
p8701[0]		-		CBC Receive PDO 2, COB-ID des PDO	40000306H
p8701[1]				CBC Receive PDO 2, Transmission Type des PDO	FEH
p8702[0]		-		CBC Receive PDO 3, COB-ID des PDO	40000406H
p8702[1]				CBC Receive PDO 3, Transmission Type des PDO	FEH
p8703[0]		-		CBC Receive PDO 4, COB-ID des PDO	40000506H
p8703[1]				CBC Receive PDO 4, Transmission Type des PDO	FEH
p8704[0]		+		CBC Receive PDO 5, COB-ID des PDO	C00006E0H
p8705[0]		+		CBC Receive PDO 6, COB-ID des PDO	C00006E0H
p8706[0]		+		CBC Receive PDO 7, COB-ID des PDO	C00006E0H
p8707[0]		+		CBC Receive PDO 8, COB-ID des PDO	C00006E0H
p8710[0]		-		CBC Receive Mapping für RPDO 1, Gemapptes Obj	68400010H
p8710[1]				CBC Receive Mapping für RPDO 1, Gemapptes Obj	0H
p8710[2]				CBC Receive Mapping für RPDO 1, Gemapptes Obj	0H
p8710[3]				CBC Receive Mapping für RPDO 1, Gemapptes Obj	0H
p8711[0]		-		CBC Receive Mapping für RPDO 2, Gemapptes Obj	68400010H
p8711[1]				CBC Receive Mapping für RPDO 2, Gemapptes Obj	68FF0020H
p8711[2]				CBC Receive Mapping für RPDO 2, Gemapptes Obj	0H
p8711[3]				CBC Receive Mapping für RPDO 2, Gemapptes Obj	0H
p8712[0]		-		CBC Receive Mapping für RPDO 3, Gemapptes Obj	68400010H
p8712[1]				CBC Receive Mapping für RPDO 3, Gemapptes Obj	68710010H
p8712[2]				CBC Receive Mapping für RPDO 3, Gemapptes Obj	0H
p8712[3]				CBC Receive Mapping für RPDO 3, Gemapptes Obj	0H
p8713[0]		-		CBC Receive Mapping für RPDO 4, Gemapptes Obj	68400010H
p8713[1]				CBC Receive Mapping für RPDO 4, Gemapptes Obj	68FF0020H
p8713[2]				CBC Receive Mapping für RPDO 4, Gemapptes Obj	68710010H
p8713[3]				CBC Receive Mapping für RPDO 4, Gemapptes Obj	0H
p8714[0]		+		CBC Receive Mapping für RPDO 5, Gemapptes Obj	0H
p8715[0]		+		CBC Receive Mapping für RPDO 6, Gemapptes Obj	0H
p8716[0]		+		CBC Receive Mapping für RPDO 7, Gemapptes Obj	0H
p8717[0]		+		CBC Receive Mapping für RPDO 8, Gemapptes Obj	0H

Figure 3-25 COB IDs and mapped process data objects for receive message frames: drive object 2



Parameter	D	+	+	Parametertext	Online-Wert SERVO_04
p8720[0]		+		CBC Transmit PDO 1, COB-ID des PDO	40000186H
p8721[0]		+		CBC Transmit PDO 2, COB-ID des PDO	40000286H
p8722[0]		+		CBC Transmit PDO 3, COB-ID des PDO	40000386H
p8723[0]		+		CBC Transmit PDO 4, COB-ID des PDO	40000486H
p8724[0]		+		CBC Transmit PDO 5, COB-ID des PDO	C00006E0H
p8725[0]		+		CBC Transmit PDO 6, COB-ID des PDO	C00006E0H
p8726[0]		+		CBC Transmit PDO 7, COB-ID des PDO	C00006E0H
p8727[0]		+		CBC Transmit PDO 8, COB-ID des PDO	C00006E0H
p8730[0]		-		CBC Transmit Mapping für TPDO 1, Gemapptes Obj	68410010H
p8730[1]				CBC Transmit Mapping für TPDO 1, Gemapptes Obj	0H
p8730[2]				CBC Transmit Mapping für TPDO 1, Gemapptes Obj	0H
p8730[3]				CBC Transmit Mapping für TPDO 1, Gemapptes Obj	0H
p8731[0]		-		CBC Transmit Mapping für TPDO 2, Gemapptes Obj	68410010H
p8731[1]				CBC Transmit Mapping für TPDO 2, Gemapptes Obj	686C0020H
p8731[2]				CBC Transmit Mapping für TPDO 2, Gemapptes Obj	0H
p8731[3]				CBC Transmit Mapping für TPDO 2, Gemapptes Obj	0H
p8732[0]		-		CBC Transmit Mapping für TPDO 3, Gemapptes Obj	68410010H
p8732[1]				CBC Transmit Mapping für TPDO 3, Gemapptes Obj	68740010H
p8732[2]				CBC Transmit Mapping für TPDO 3, Gemapptes Obj	0H
p8732[3]				CBC Transmit Mapping für TPDO 3, Gemapptes Obj	0H
p8733[0]		-		CBC Transmit Mapping für TPDO 4, Gemapptes Obj	68410010H
p8733[1]				CBC Transmit Mapping für TPDO 4, Gemapptes Obj	68630020H
p8733[2]				CBC Transmit Mapping für TPDO 4, Gemapptes Obj	0H
p8733[3]				CBC Transmit Mapping für TPDO 4, Gemapptes Obj	0H
p8734[0]		+		CBC Transmit Mapping für TPDO 5, Gemapptes Obj	0H
p8735[0]		+		CBC Transmit Mapping für TPDO 6, Gemapptes Obj	0H
p8736[0]		+		CBC Transmit Mapping für TPDO 7, Gemapptes Obj	0H
p8737[0]		+		CBC Transmit Mapping für TPDO 8, Gemapptes Obj	0H
p8744				CBC PDO Mapping Konfiguration	Freies PDO Mapping (2)

Figure 3-26 COB IDs and mapped process data objects for transmit message frames: drive object 2

3.5.2 Activating "free PDO mapping"

Steps

To activate "free PDO mapping" in SINAMICS drive object 2 (Single Motor Module 2), carry out the following:

1. In the project navigator, open the expert list for the drive as follows: **Drive_2 > right-click > Expert > Expert list**.
2. Search for parameter **p8744** (see screenshot below) to activate free PDO mapping.

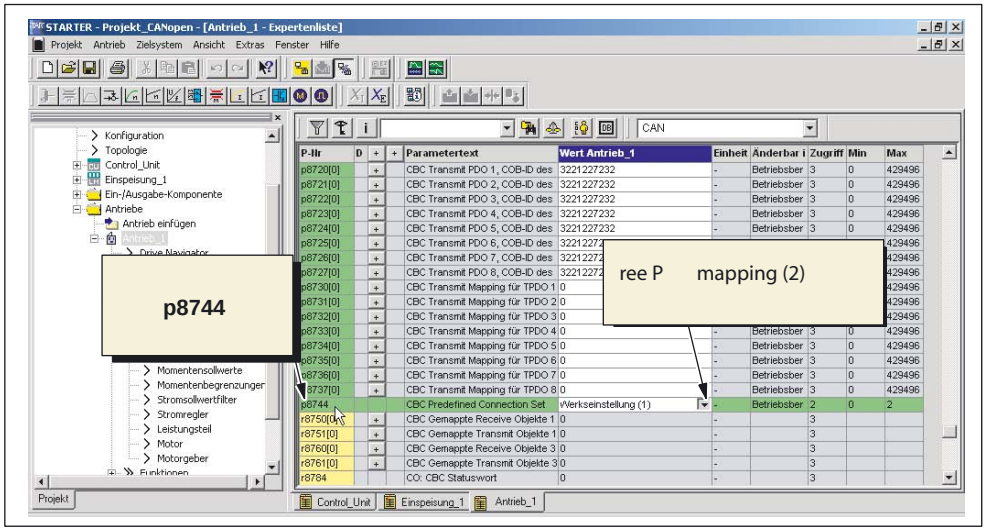


Figure 3-27 Free PDO mapping

3. Choose **Free PDO mapping (2)**.

3.5.3 Assigning COB IDs and mapping parameters for free PDO mapping

Introduction

If you are in ONLINE mode and want to assign free COB IDs and mapping parameters, you have the following options:

- COB IDs:
  You can assign a COB as required in the following parameters:
  - Receive (RPDOs) in parameters **p8700** to **p8707**
  - Transmit (TPDOs) in parameters **p8720** to **p8727**
- Mapping parameters:
  You can enter the process data objects for mapping the transmission message frames in the following parameters:
  - Receive message frames (RPDOs) as of parameter **p8710**
  - Transmit message frames (TPDOs) as of parameter **p8730**

Note

If mapping parameters are changed in STARTER in ONLINE mode, the COB ID for the PDO in question must first be set to "invalid" in the communication parameters and, once the parameters have been changed, reset to "valid".

COB ID, bit 31 = 0 -> COB ID valid,

bit 31 = 1 -> COB ID invalid

Procedure

To carry out free PDO mapping in ONLINE mode, proceed as follows:

1. Set the COB ID of the RPDO or TPDO in question to "invalid" (e.g.: **p8700[0]** CBC: receive PDO1, COB ID of PDO).
2. Enter the process data objects as mapping parameters in the RPDO or TPDO in question (e.g.: as of **p8710[0]** CBC receive mapping for RPDO1, mapped object).
3. Set the COB ID of the RPDO or TPDO in question to "valid".

Steps

An example is used to illustrate the procedure described below (in this example: RPDO1 for drive object 2).

Setting the COB ID to "invalid"

1. In the project navigator, call up the expert list for Single Motor Module 2 (drive_2): **Drives > Drive_2 > right-click > Expert > Expert list**.
2. Search for COB ID parameter **p8700 [0]** for communication parameter RPDO1.
3. Copy the hexadecimal value from STARTER to a calculator and enter **Or** 8000 0000 hex. Copy the result to the STARTER parameter. As a result, bit 31 has now been set to "invalid".
4. You can now adjust the mapping parameters.

Mapping parameters

1. Choose the process data object to be mapped
   (e.g. RPDO1, control word = 6040 hex) from the "Objects in drive profile DSP402" table (see "Communication objects") .
2. Add a suitable offset for the SINAMICS drive object (e.g. plus 800 hex as of drive object 2).
   See the column "Values from table" OD index (hex) (e.g. 6840 hex) in the parameterization table in the following section.
3. Calculate the sum of "OD index + sub-index + object size" as a hexadecimal value.
   See the column "Resulting hexadecimal value" (e.g. 6840 0010 hex) in the parameterization table in the following section.
4. Enter this value in STARTER in the corresponding mapping parameter.
   See the column "Mapping parameter in STARTER" (e.g. **p8710 [0]**) in the parameterization table in the following section.

Setting the COB ID to "valid"

1. In the project navigator, call up the expert list for Single Motor Module 2: **Drives > Drive_2 > right-click > Expert > Expert list**.
2. Search for COB ID parameter **p8700 [0]** for communication parameter RPDO1.
3. Copy the hexadecimal value from STARTER to a calculator and enter **And** 7FFF FFFF hex. Copy the result to the STARTER parameter. As a result, bit 31 has now been set to "valid".
4. Copy the value to STARTER.

Note

The RPDOs and TPDOs described below serve to illustrate the free PDO mapping process. You can decide for yourself how you want to map your own process data objects for transmission message frames.

See also

Objects in drive profile DSP402 (Page 5-17)

Mapping RPDO1

When you choose the process data object for the control word from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8710 [0]** to **p8710 [3]** for RPDO1 mapping (see the "Resulting hexadecimal value" column in the following table):

Note

Since it is SINAMICS drive object 2, 800 hex is added to the index.

Table 3-6 Values for RPDO1

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter RPDO1
Control word	6840	00	10 hex (16 bit)	6840 0010	p8710 [0]
No object				0	p8710 [1] to [3]

Mapping RPDO2

When you choose the process data object for the control word and the set velocity from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8711 [0]** to **p8711 [3]** for RPDO2 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-7 Values for RPDO2

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter RPDO2
Control word	6840	00	10 hex (16 bit)	6840 0010	p8711 [0]
Set velocity	68FF	00	20 hex (32 bit)	68FF 0020	p8711 [1]
No object				0	p8711 [2] to [3]

Mapping RPDO3

When you choose the process data object for the control word and the set torque from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8712 [0]** to **p8712 [3]** for RPDO3 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-8 Values for RPDO3

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter RPDO3
Control word	6840	00	10 hex (16 bit)	6840 0010	p8712 [0]
Set torque	6871	00	10 hex (16 bit)	6871 0010	p8712 [1]
No object				0	p8712 [2] to [3]

Mapping RPDO4

When you choose the process data object for the control word, the set velocity, and the set torque from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8713 [0]** to **p8713 [3]** for RPDO4 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-9 Values for RPDO4

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter RPDO4
Control word	6840	00	10 hex (16 bit)	6840 0010	p8713 [0]
Set velocity	68FF	00	20 hex (32 bit)	6840 0020	p8713 [1]
Set torque	6871	00	10 hex (16 bit)	6871 0010	p8713 [2]
No object				0	P8713 [3]

Mapping TPDO1

When you choose the process data object for the CBC status word from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8730 [0]** to **p8730 [3]** for TPDO1 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-10 Values for TPDO1

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter TPDO1
CBC status word	6841	00	10 hex (16 bit)	6841 0010	p8730 [0]
No object				0	P8730 [1] to [3]

Mapping TPDO2

When you choose the process data object for the CBC status word and the set velocity from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8731 [0]** to **p8731 [3]** for TPDO2 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-11 Values for TPDO2

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter TPDO2
CBC status word	6841	00	10 hex (16 bit)	6841 0010	p8731 [0]
Actual velocity	686C	00	20 hex (32 bit)	686C 0020	p8731 [1]
No object				0	P8731 [2] to [3]

Mapping TPDO3

When you choose the process data object for the CBC status word and the actual torque from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8732 [0]** to **p8732 [3]** for TPDO3 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-12 Values for TPDO3

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter TPDO3
CBC status word	6841	00	10 hex (16 bit)	6841 0010	p8732 [0]
Actual torque	6874	00	10 hex (16 bit)	6874 0010	p8732 [1]
No object				0	P8732 [2] to [3]

Mapping TPDO4

When you choose the process data object for the CBC status word and the encoder actual position value from the "Objects in drive profile DSP402" table, this yields the following values, which you enter in parameters **p8733 [0]** to **p8733 [3]** for TPDO4 mapping (see the "Resulting hexadecimal value" column in the following table):

Table 3-13 Values for TPDO4

Object name	Values from "Objects in drive profile DSP402" table			Mapping parameter in STARTER	
	OD index (hex)	Sub-index (hex)	Object size	Resulting hexa-decimal value	Parameter TPDO4
CBC status word	6841	00	10 hex (16 bit)	6841 0010	p8733 [0]
Actual position value	6863	00	20 hex (32 bit)	6863 0020	p8733 [1]
No object				0	p8733 [2] to [3]

3.5.4 Interconnecting the receive and transmit buffers

Steps

Once you have mapped receive and transmit PDOs in ONLINE mode in free PDO mapping, you have to interconnect the image of the mapped process data objects with the receive and transmit buffers.

Note

To interconnect the receive and transmit buffers in ONLINE mode, carry out the steps described in the following section ("Interconnecting receive and transmit buffers").

3.6 Interconnecting process data in the receive and transmit buffers

Initial commissioning: procedure

In the table below, the current commissioning step is highlighted in **bold**:

Table 3-14 CANopen initial commissioning

Step	Procedure
1	Make the hardware settings on CBC10.
2	Configure the drive unit using the STARTER commissioning tool in ONLINE mode.
3	Configure the COB IDs and process data objects for the receive and transmit message frames.
4	**Interconnect the receive and transmit buffers.**
5	In ONLINE mode, load the projects from the drive unit to the PC/PG and save.

Carrying out the commissioning step

The process data objects for the transmission message frames must be interconnected by means of BICO interconnection for the interface between SINAMICS S120 and the CAN bus. During initial commissioning, proceed as follows:

- Read the image of the process data objects to the PZD receive and PZD transmit words for the receive and transmit buffers.
- Interconnect the source parameter of the PZD receive word on the receive buffer with the SINAMICS target parameter of the process data object.
- Interconnect the SINAMICS source parameter of the process data object with the target parameter of the PZD transmit word on the transmit buffer.

Note

The STARTER commissioning tool is in ONLINE mode.

See also

BICO interconnection procedure in STARTER (Page 2-9)

Function diagrams (Page A-2)

Objects in drive profile DSP402 (Page 5-17)

3.6.1 Reading the image of the individual process data objects for the receive and transmit buffers

Introduction

Once you have parameterized PDO mapping, STARTER automatically recognizes how the individual process data objects have to be distributed to the PZD receive and PZD transmit words.

To read the image of the individual process data objects to the PZD receive and PZD transmit words for the receive and transmit buffers, carry out the following steps.

Example

Note

Each drive object has a separate receive and transmit buffer for transferring message frames.

The expert list for each of the drive objects contains the images for the:

- 16-bit process data objects for
  - Receive, as of parameter **r8750[0]**
  - Transmit, as of parameter **r8751[0]**
- 32-bit process data objects for
  - Receive, as of parameter **r8760[0]**
  - Transmit, as of parameter **r8761[0]**

Steps (using drive object 2 as an example)

1. In the project navigator, call up the expert list for drive object 2: **Drives > Drive_2 > right-click > Expert > Expert list**.
2. Search for r parameter **r8750[0]** with the image of the 16-bit process data objects in the receive buffer. In this example, the following are mapped (see also screenshot below):
   - Control word (6840 hex) in PZD 1
   - Set torque (6871 hex) in PZD 4
3. Search for r parameter **r8760[0]** with the image of the 32-bit process data objects in the receive buffer. In this example, the following is mapped (see also screenshot below):
   - Set velocity (68FF hex) in PZD 2+3

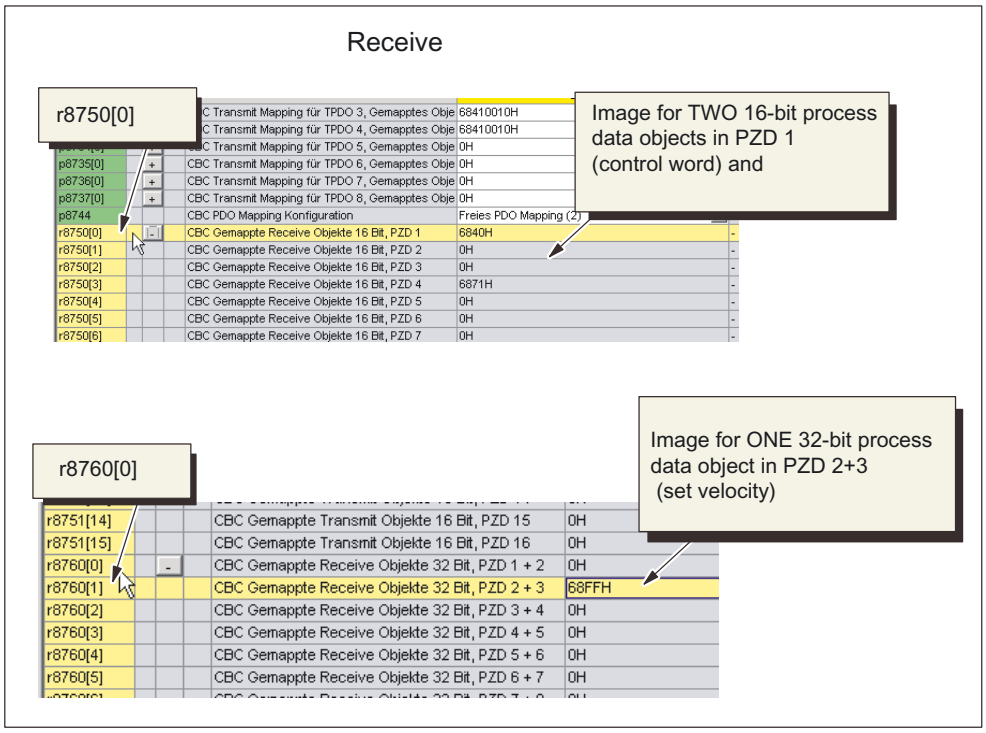


Figure 3-28 Image of process data objects in receive buffer

4. Search for r parameter **r8751[0]** with the image of the 16-bit process data objects in the transmit buffer. In this example, the following are mapped (see also screenshot below):
   - CBC status word (6841 hex) in PZD 1
   - Actual torque (6874 hex) in PZD 4
5. Search for r parameter **r8761[0]** with the image of the 32-bit process data objects in the transmit buffer. In this example, the following are mapped (see also screenshot below):
   - Actual velocity (686C hex) in PZD 2+3
   - Actual position value (6863 hex) in PZD 5+6

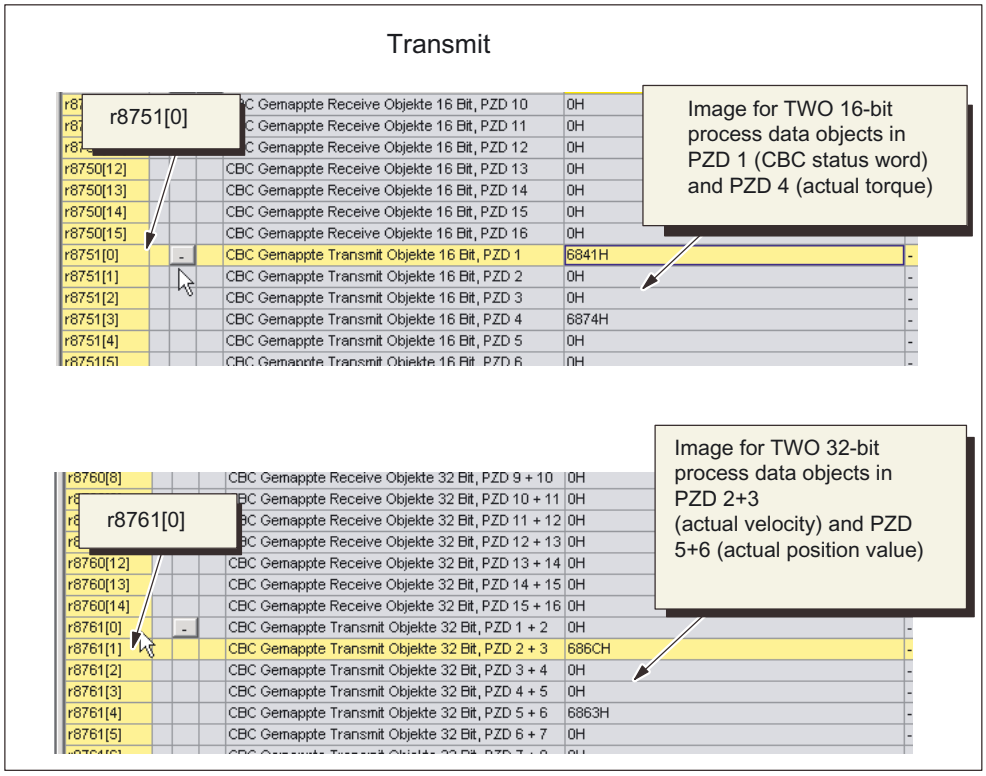


Figure 3-29 Image of process data objects in transmit buffer

6. In the project navigator, call up the expert list for drive object 1: **Drives > Drive_1 > right-click > Expert > Expert list**.
   For receiving purposes, the 16-bit process data objects as of parameter **r8750[0]** and the 32-bit process data objects as of parameter **r8760[0]** are also available here.
   Likewise, the 16-bit process data objects as of parameter **r8751[0]** and the 32-bit process data objects as of parameter **r8761[0]** can be read for transmission purposes.
7. Once you have read the images for both drive objects, you can now interconnect the process data objects in the receive and transmit buffers (see below).

3.6.2 Interconnecting process data for transmission message frames

Prerequisite

On the basis of the image read in the previous section, note the following:

The following are interconnected:

- The SINAMICS source and target parameters for the process data objects
- The receive/transmit words in the receive/transmit buffer

Note

The excerpts of function diagrams for the receive/transmit buffer, which are shown in the following sections, illustrate:

- How the process data objects in the receive and transmit buffers are distributed to the receive and transmit words.

- Which associated target and source parameters for the receive and transmit words have to be interconnected (**highlighted**).

Note

The SINAMICS source and target parameters, which are interconnected with the receive and transmit words, are listed in the "SINAMICS parameter" column in the "Objects in drive profile DSP 402" table.

See also

Objects in drive profile DSP402 (Page 5-17)

Function diagrams (Page A-2)

3.6.2.1 Interconnecting the receive buffer

Introduction

In the receive buffer, the following are interconnected for the transmission message frames:

- Control word (PZD 1)
- Set velocity (PZD 2+3)
- Set torque (PZD 4)

Steps

To interconnect the set velocity process data object in PZD receive word 2+3 (32 bit), for example, interconnect the following target and source parameters:

- SINAMICS target parameter for the set velocity (**p1155[0]** => 32 bit, see "Objects in drive profile DSP402" table)

Table 3-15 Excerpt of "Objects in drive profile DSP402"

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
...	...	...	...	...	...	...	...
60FF		Target velocity Set velocity	Without ramp-function generator -> p1155[0] With ramp-function generator -> p1070	SDO/PDO	Integer32	-	rw

- Source parameter **r8860[1]** => 32 bit in the receive buffer (see the following diagram; the source parameter for the set velocity is **highlighted** accordingly).

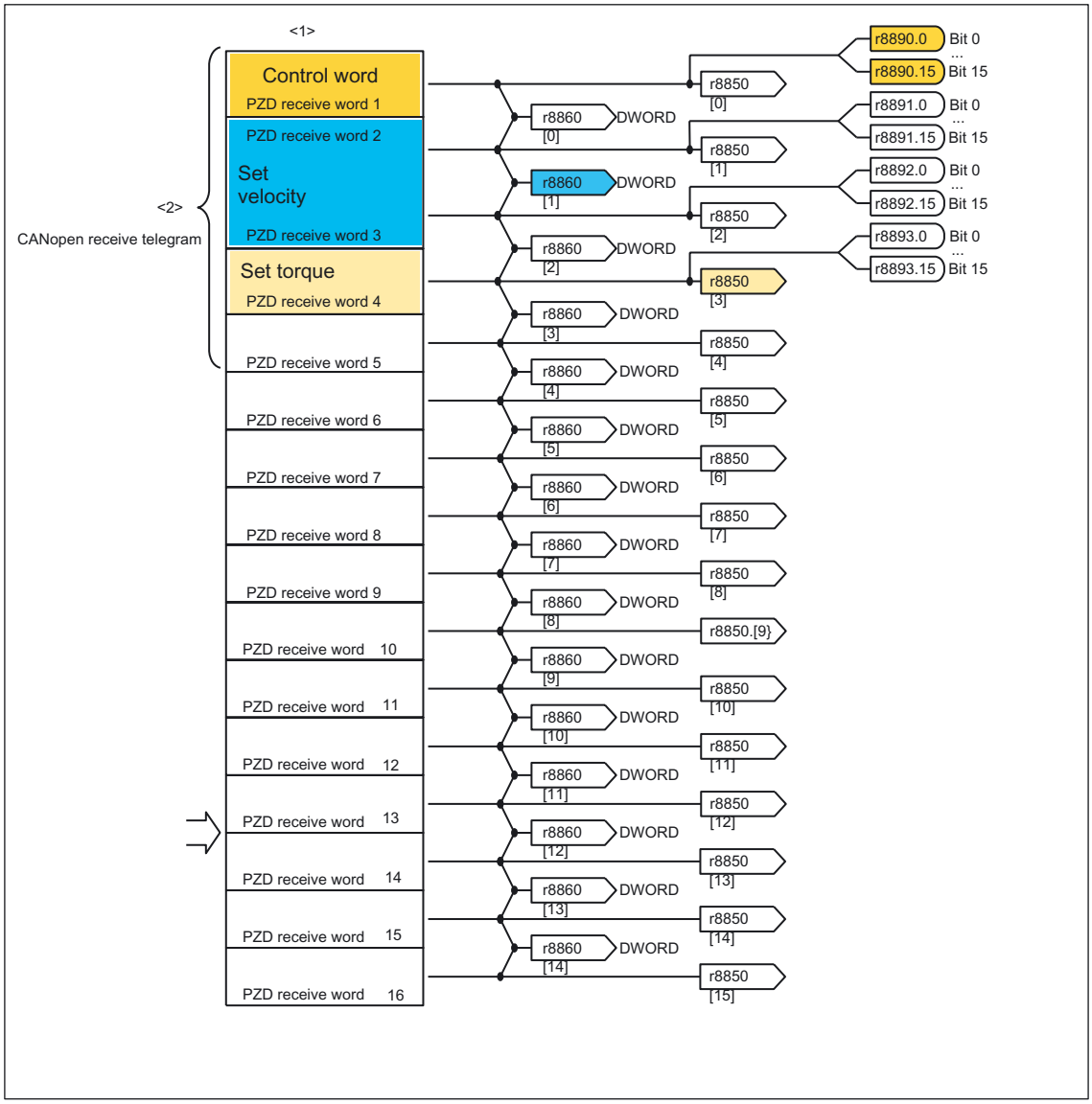


Figure 3-30 Excerpt of function diagram: receive buffer

Note

You can now interconnect the process data objects listed below.

Control word (PZD 1 16 bit)

The control word is interconnected automatically with source parameter **p8890** as a SINAMICS target parameter during initial commissioning.

You do not have to interconnect the control word.

Set velocity (PZD 2+3 32 bit)

The following table shows the p parameter for the set velocity, which has to be interconnected with the corresponding source.

Table 3-16 Interconnect set velocity

Target (sink)	Source PZD 2+3	Meaning
p1155[0]	r8860 [1]	Speed setpoint 1

Interconnect p parameter **p1155 [0]** with r parameter **r8860 [1]** for the setpoint velocity (32-bit data type) by carrying out the following:

1. In the project navigator, choose **Drive_1 > Open-loop/closed-loop control** and double-click **Setpoint addition**.
2. Choose **Speed setpoint1 > Further interconnections**.
3. Interconnect parameter **p1155 [0]** with **r8860 [1]** = PZD 2+3.

Set torque (PZD 4 16 bit)

The following table shows the p parameter for the set torque, which has to be interconnected with the corresponding source.

Table 3-17 Interconnect the set torque

Target (sink)	Source PZD 4	Meaning
p1513[0]	r8850 [3]	Additional torque

Interconnect p parameter **p1513 [0]** with r parameter **r8850 [3]** for the set torque by carrying out the following:

1. In the project navigator, choose **Drive_1 > Open-loop/closed-loop control** and double-click **Torque setpoints**.
2. Choose **Supplementary torque 2 > Further interconnections**.
3. Interconnect parameter **p1513 [0]** with **r8850 [3]** = PZD 4.

Note

To ensure that the set torque becomes active, it still has to be interconnected with a free bit of the control word. This can be carried out in this dialog box by choosing **Speed/torque control > Further interconnections** (**p1051 [0]** with **bit 14** from **r8890**).

3.6.2.2 Interconnecting the transmit buffer

Introduction

In the transmit buffer, interconnect the following for the transmission message frames:

- CBC status word (PZD 1)
- Actual velocity (PZD 2+3)
- Actual torque (PZD 4)

Steps

To interconnect the CBC status word process data object in PZD transmit word 1 (16 bit), for example, proceed as follows. Interconnect the following source and target parameters:

- SINAMICS source parameters for the CBC status word (**r8784 =>** 16 bit, see "Objects in drive profile DSP402" table)

Table 3-18 Excerpt of "Objects in drive profile DSP402"

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
...	...	...	...	...	...	...	...
6041		statusword	r8784	PDO/SDO	Unsigned16	-	ro

- Target parameter **p8851[0] =>** 16 bit in the transmit buffer (see the following diagram; the target parameter for the CBC status word is **highlighted** accordingly).

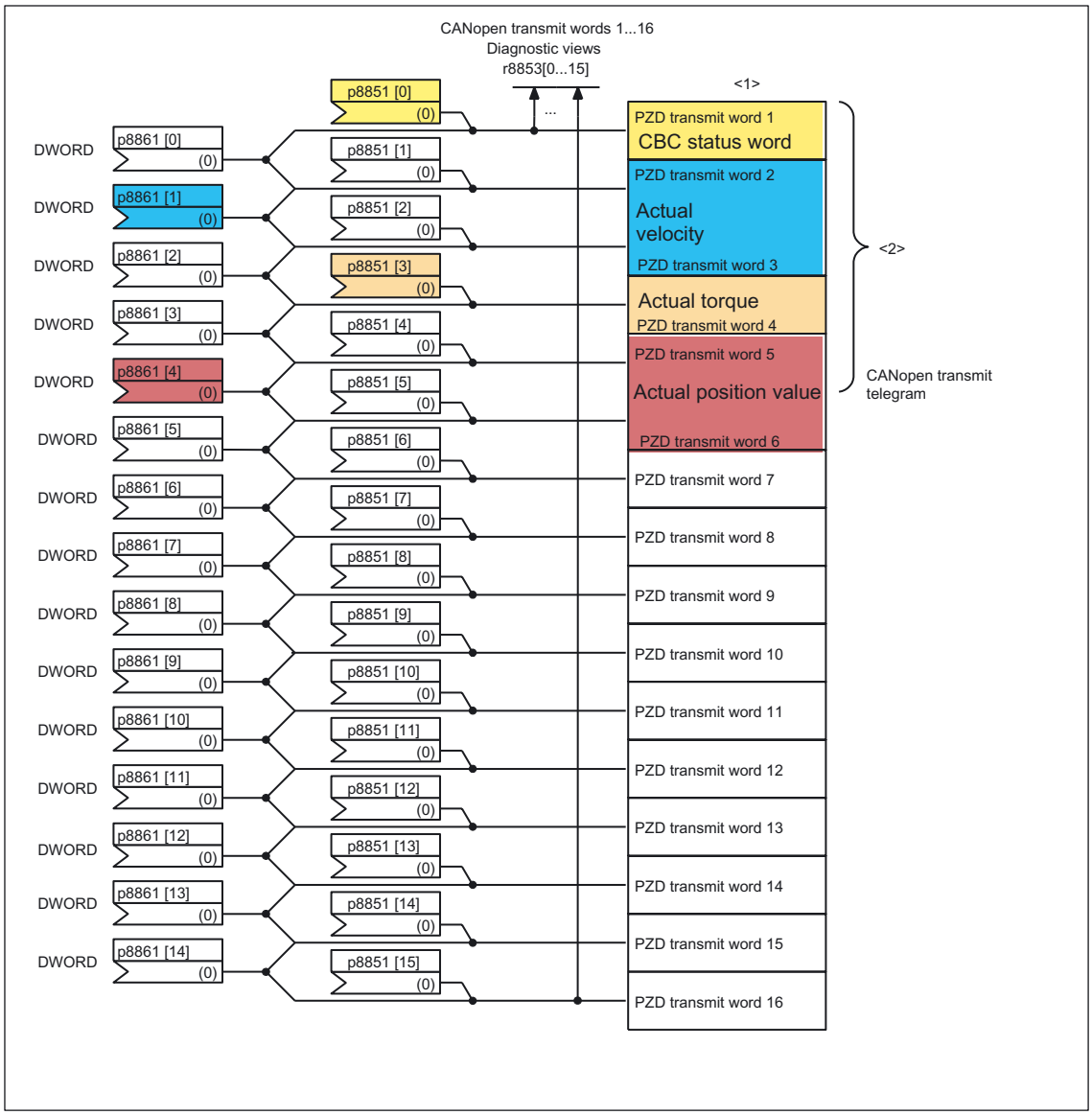


Figure 3-31 Excerpt of function diagram: transmit buffer

Note

You can now interconnect the process data objects listed below.

CBC status word (PZD1 16 bit)

The following table shows the p parameter for the CBC status word, which has to be interconnected with the corresponding source.

Table 3-19 Interconnect the CBC status word

Target (sink) PZD1	Source	Meaning
p8851 [0]	r8784	CBC status word

Interconnect p parameter **p8851 [0]** with r parameter **r8784** for the CBC status word by carrying out the following:

1. In the project navigator, search for parameter **p8851 [0]** by choosing **Drive_1 > right-click > Expert > Expert list**.
2. Interconnect parameter **p8851 [0]** = PZD 1 with the associated r parameters (**r8784**).

Actual velocity (PZD 2+3 32 bit)

The following table shows the p parameter for the actual velocity, which has to be interconnected with the corresponding source.

Table 3-20 Interconnect the actual velocity

Target (sink) PZD 2+3	Source	Meaning
p8861 [1]	r0063	Actual speed value

Interconnect p parameter **p8861 [1]** with r parameter **r0063** for the actual velocity by carrying out the following:

1. In the project navigator, search for parameter **p8861 [1]** by choosing **Drive_1 > right-click > Expert > Expert list**.
2. Interconnect parameter **p8861 [1]** = PZD 2 + 3 with the associated r parameters (**r0063**).

Actual torque (PZD 4 16 bit)

The following table shows the p parameter for the actual torque, which has to be interconnected with the corresponding source.

Table 3-21 Interconnect the actual torque

Target (sink) PZD 4	Source	Meaning
p8851 [3]	r0080	Actual torque value

Interconnect p parameter **p8851 [3]** with r parameter **r0080** for the actual velocity by carrying out the following:

1. In the project navigator, search for parameter **p8851 [3]** by choosing **Drive_1 > right-click > Expert > Expert list**.
2. Interconnect parameter **p8851 [3]** = PZD 4 with the associated r parameters (**r0080**).

Actual position value (PZD 5+6 32 bit)

The following table shows the p parameter for the actual position value, which has to be interconnected with the corresponding source.

Table 3-22 Interconnect the actual position value

Target (sink) PZD 5+6	Source	Meaning
p8861 [4]	r0482	Actual position value

Interconnect p parameter p8851 [3] with r parameter r0482 for the actual position value by carrying out the following:

1. In the project navigator, search for parameter **p8861 [4]** by choosing **Drive_1 > right-click > Expert > Expert list**.
2. Interconnect parameter **p8861 [4]** = PZD 5 + 6 with the associated r parameters (**r0482**).

3.6.2.3 Interconnecting an additional drive object

Steps

To interconnect the next drive object (Single Motor Module 2), repeat the steps described in "Interconnecting transmission message frames in the receive buffer" and "Interconnecting transmission message frames in the transmit buffer".

When carrying out the steps, choose **Drive_2** in the project navigator.

3.7 Load the projects from the drive unit to the PC/PG in ONLINE mode and save them

Initial commissioning: procedure

In the table below, the current commissioning step is highlighted in **bold**:

Table 3-23 CANopen initial commissioning

Step	Procedure
1	Make the hardware settings on CBC10.
2	Configure the drive unit using the STARTER commissioning tool in ONLINE mode.
3	Configure the COB IDs and process data objects for the receive and transmit message frames.
4	Interconnect the receive and transmit buffers.
5	**In ONLINE mode, load the projects from the drive unit to the PC/PG and save.**

Prerequisite

You are in ONLINE mode in STARTER and have completed the initial commissioning procedure.

Steps

To store the data configured ONLINE in the STARTER project on the PG/PC, proceed as follows:

1. In the project navigator, select **Drive unit_126**. Click the **Load to PG** function key (see screenshot below).

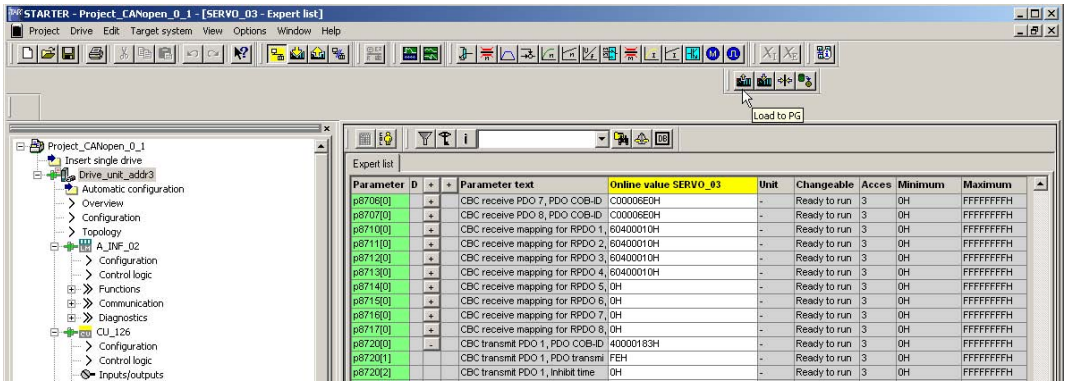


Figure 3-32 "Load to PG" function key

2. Carry out the following:
   - When the system asks "Are you sure?", click **Yes**. The system now starts loading the data.
   - When the system informs you that the data was successfully loaded, click **OK**.
3. Click the **Disconnect from target system** function key.
4. Carry out the following:
   - **Changes in the drive unit...**
   - **Save data,** for SERVO_3
   - When the system informs you that the data was successfully copied from RAM to ROM, click **OK**.
   - When the system prompts you to confirm that you want the data to be loaded to the PG, click **Yes**.
   - When the system informs you that the data was successfully loaded to the PG, click **OK**.
   - **Save data,** for SERVO_4
   - When the system informs you that the data was successfully copied from RAM to ROM, click **OK**.
   - When the system prompts you to confirm that you want the data to be loaded to the PG, click **Yes**.
   - When the system informs you that the data was successfully loaded to the PG, click **OK**.
5. STARTER is in OFFLINE mode.
6. Click **Project > Save as...**

Note

This completes initial commissioning for the CANopen interface.

Diagnosis

4

Section content

CANopen supports a standardized system for detecting, describing, and signaling device errors with the following equipment:

- One emergency object per drive unit
- A device-internal error list ("predefined error field")
- An error register

Note

See also the following standards:

- CiA DS-301 V4.01 (Application Layer and Communication Profile)

- CiA DS-402 V2.0 (Device Profile for Drives and Motion Control)

4.1 Emergency object

Introduction

Error statuses for each drive unit are signaled via the high-priority 8-byte emergency object (error message).

The relevant parameters are located:

- In the object directory index 1014 hex (COB ID EMCY) and 1015 hex (inhibit time EMCY)
- With SINAMICS S120 in parameter p8603 of the CU320

When an error occurs, an error message frame is sent to the identifier set in object **1014 hex**.

In CANopen, errors are assigned error codes which, in turn, are sub-divided into current errors, voltage errors, and so on. SINAMICS S120 outputs the same code for each error (generat error 1000).

Emergency message frame

When an error occurs, the CANopen drive unit automatically transmits an emergency message asynchronously. The emergency message is structured as follows (see table below):

Table 4-1 Structure of the emergency message frame

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Error code		Error register	Drive object number	Fault number		Reserved	Reserved

The CANopen error code is in byte 0 and 1. In SINAMICS S120, this is always generic error 1000 hex.

Byte 3 contains the number of the drive object from which the error originated.

Alarms that do not result in a shutdown are indicated only by the alarm bit or fault bit in the status word and do not trigger an emergency message. Faults trigger emergency messages and cause the drives to be shut down. The master can then read the alarm or fault during a period of free bus time. The fault message frame can be suppressed by setting bit 31 in object 1014 hex.

4.2 Drive-unit-internal error list ("predefined error field")

Introduction

The drive-unit-internal error list ("predefined error field") can be read via:

- Object directory index 1003 hex
- SINAMICS S120 parameter **p8611** in the CU320

This list contains the individual faults diagnosed in a drive unit. The faults are listed in the order in which they occur, along with the fault code and additional, device-specific information.

4.3 Error register

Introduction

The 1-byte error register can be read via:

- Object directory index 1001 hex
- SINAMICS S120 parameter **p8601** in the CU320

The register displays any drive unit errors that have occurred and their type.

Note

CANopen SINAMICS 120 can only display "Generic Error 1000 hex" errors.

Communication objects

5

Section content

This section contains a table of the objects (data values) that are used in SINAMICS S120 for communication via the CANopen interface. This includes:

- Drive-independent CU320 communication objects
- Drive-dependent communication objects
- Objects in drive profile DSP402

The objects are stored in an object directory.

See also

CANopen object directory (Page 1-8)

5.1 Drive-independent CU320 communication objects

Overview

The following table lists the object directory with the index of the individual drive-independent CU320 communication objects. The column "SINAMICS parameter" shows the parameter range in which they are located for SINAMICS S120.

Table 5-1 Drive-independent CU320 communication objects

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
1000		Device type	r8600	SDO	Unsigned32	–	ro
1001		Error register	r8601	SDO	Unsigned8	–	ro
1003		Predefined error field	p8611 [0...82]	SDO	Unsigned32	0	rw
	0	Number of errors	p8611.0	SDO	Unsigned32	0	rw
	1	Number of module	p8611.1	SDO	Unsigned32	0	rw
	2	Number of errors: module 1	p8611.2	SDO	Unsigned32	0	rw
	3-9	Standard error field: module 1	p8611.3-p8611.9	SDO	Unsigned32	0	rw
	A	Number of errors: module 2	p8611.10	SDO	Unsigned32	0	rw
	B-11	Standard error field: module 2	p8611.11-p8611.17	SDO	Unsigned32	0	rw
	12	Number of errors: module 3	p8611.18	SDO	Unsigned32	0	rw
	12-19	Standard error field: module 3	p8611.19-p8611.25	SDO	Unsigned32	0	rw
	1 A	Number of errors: module 4	p8611.26	SDO	Unsigned32	0	rw
	1B-21	Standard error field: module 4	p8611.27-p8611.33	SDO	Unsigned32	0	rw
	22	Number of errors: module 5	p8611.34	SDO	Unsigned32	0	rw
	23-29	Standard error field: module 5	p8611.35-p8611.41	SDO	Unsigned32	0	rw
	2A	Number of errors: module 6	p8611.42	SDO	Unsigned32	0	rw
	2B-31	Standard error field: module 6	p8611.43-p8611.49	SDO	Unsigned32	0	rw
	32	Number of errors: module 7	p8611.50	SDO	Unsigned32	0	rw
	33-39	Standard error field:	p8611.51-	SDO	Unsigned32	0	rw

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
		module 7	p8611.57				
	3A	Number of errors: module 8	p8611.58	SDO	Unsigned32	0	rw
	3B-41	Standard error field: module 8	p8611.59-p8611.65	SDO	Unsigned32	0	rw
1005		SYNCH COB ID	p8602.0	SDO	Unsigned32	128	rw
1006		Communication cycle period	p8602.1	SDO	Unsigned32	128	rw
1008		Manufacturer device name		SDO			
100A		Manufacturer software version	r0018	SDO	Unsigned32	–	ro
100C		Guard time	p8604.0	SDO	Unsigned16	0	rw
100D		Life time factor	p8604.1	SDO	Unsigned16	0	rw
1010		Store parameters	p0977	SDO	Unsigned16	0	rw
	0	Largest subindex supported		SDO			
	1	Save all parameters	p0977	SDO	Unsigned16	0	rw
	2	Save communication parameters (0x1000-0x1fff)	p0977	SDO	Unsigned16	0	rw
	3	Save application-related parameters (0x6000-0x9fff)	p0977	SDO	Unsigned16	0	rw
1011		Restore default parameters	p0976	SDO	Unsigned16	0	rw
	0	Largest subindex supported		SDO			
	1	Restore all default parameters	p0976	SDO	Unsigned16	0	rw
	2	Restore communication default parameters (0x1000-0x1fff)	p0976	SDO	Unsigned16	0	rw
	3	Restore application default parameters (0x6000-0x9fff)	p0976	SDO	Unsigned16	0	rw
1014		COB ID emergency	p8603.0	SDO	Unsigned32	0	rw
1017		Producer heartbeat time	p8606	SDO	Unsigned16	0	rw
1018		Identy Object	r8607[0...3]		Unsigned32	–	ro
	0	Number of entries		SDO			
	1	Vendor ID	r8607.0	SDO	Unsigned32	–	ro
	2	Product code	r8607.1	SDO	Unsigned32	–	ro
	3	Revision number	r8607.2	SDO	Unsigned32	–	ro
	4	Serial number	r8607.3	SDO	Unsigned32	0	ro

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
1027		Module list					
	0	Number of entries	r0102	SDO	Unsigned16	–	ro
	1-8	Module ID	p0107 [0...15]	SDO	Integer16	0	rw
1029		Error behavior					
	0	No. of error classes		SDO			
	1	Communication Error	p8609.0	SDO	Unsigned32	1	rw
	2	Device profile or manufacturer-specific error	p8609.1	SDO	Unsigned32	1	rw
1200		1st server SDO parameter					
	0	Number of entries		SDO			
	1	COB ID client -> server (rx)	r8610.0	SDO	Unsigned32	–	ro
	2	COB ID server -> client (tx)	r8610.1	SDO	Unsigned32	–	ro
41AE		Baud rate selection	p8622	SDO	Integer16	6 (20kBit/s)	rw
41B6		Virtual objects	p8630 [0...2]	SDO	Unsigned16		rw
	0	Axe number	p8630.0		Unsigned16	1	rw
	1	Sub-index range	p8630.1		Unsigned16	0	rw
	2	Parameter range	p8630.2		Unsigned16	0	rw

5.2 Drive-dependent communication objects

Introduction

Eight transmit/receive PDOs can be parameterized for each drive, whereby a total of 25 PDOs (channels) must not be exceeded.

Each PDO contains:

- Communication parameters
- Mapping parameters (max. 8 bytes/4 words/64 bits)

Rule

In the following tables, the first PDO is highlighted in **bold** to indicate that the communication and mapping parameters for one PDO are related.

The "Predefined connection set" column contains the predefined values for the "predefined connection set".

5.2.1 Table: communication objects for receive PDOs

Overview

The following table lists the object directory with the index of the individual drive-dependent communication objects for the receive PDOs of the first drive object:

Table 5-2 Drive-dependent communication objects for receive PDOs

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
1400		**Receive PDO 1 communication parameter**					
	0	**Largest sub-index supported**		**SDO**	**Unsigned8**	**2**	**ro**
	1	**COB ID used by PDO**	**p8700.0**	**SDO**	**Unsigned32**	**200 hex + node ID**	**rw**
	2	**Transmission type**	**p8700.1**	**SDO**	**Unsigned8**	**FE hex**	**rw**
1401		Receive PDO 2 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8701.0	SDO	Unsigned32	300 hex + node ID	rw
	2	Transmission type	p8701.1	SDO	Unsigned8	FE hex	rw
1402		Receive PDO 3 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8702.0	SDO	Unsigned32	400 hex + node ID	rw
	2	Transmission type	p8702.1	SDO	Unsigned8	FE hex	rw
1403		Receive PDO 4 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8703.0	SDO	Unsigned32	500 hex + node ID	rw
	2	Transmission type	p8703.1	SDO	Unsigned8	FE hex	rw
1404		Receive PDO 5 communication parameter					

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8704.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8704.1	SDO	Unsigned8	FE hex	rw
1405		Receive PDO 6 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8705.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8705.1	SDO	Unsigned8	FE hex	rw
1406		Receive PDO 7 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8706.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8706.1	SDO	Unsigned8	FE hex	rw
1407		Receive PDO 8 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	2	ro
	1	COB ID used by PDO	p8707.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8707.1	SDO	Unsigned8	FE hex	rw
1600		**Receive PDO 1 mapping parameter**					
	0	**Number of mapped application objects in PDO**		**SDO**	**Unsigned8**	**1**	**ro**
	1	**PDO mapping for the first application object to be mapped**	**p8710.0**	**SDO**	**Unsigned32**	**6040 hex**	**rw**
	2	**PDO mapping for the second application object to be mapped**	**p8710.1**	**SDO**	**Unsigned32**	**0**	**rw**
	3	**PDO mapping for the third application object to be mapped**	**p8710.2**	**SDO**	**Unsigned32**	**0**	**rw**
	4	**PDO mapping for the fourth application object to**	**p8710.3**	**SDO**	**Unsigned32**	**0**	**rw**

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
		be mapped					
1601		Receive PDO 2 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	2	ro
	1	PDO mapping for the first application object to be mapped	p8711.0	SDO	Unsigned32	6040 hex	rw
	2	PDO mapping for the second application object to be mapped	p8711.1	SDO	Unsigned32	60FF hex	rw
	3	PDO mapping for the third application object to be mapped	p8711.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8711.3	SDO	Unsigned32	0	rw
1602		Receive PDO 3 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	2	ro
	1	PDO mapping for the first application object to be mapped	p8712.0	SDO	Unsigned32	6040 hex	rw
	2	PDO mapping for the second application object to be mapped	p8712.1	SDO	Unsigned32	6071 hex	rw
	3	PDO mapping for the third application object to be mapped	p8712.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8712.3	SDO	Unsigned32	0	rw
1603		Receive PDO 4 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	3	ro
	1	PDO mapping for the first application object to be mapped	p8713.0	SDO	Unsigned32	6040 hex	rw
	2	PDO mapping for the second	p8713.1	SDO	Unsigned32	60FF hex	rw

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
		application object to be mapped					
	3	PDO mapping for the third application object to be mapped	p8713.2	SDO	Unsigned32	6071 hex	rw
	4	PDO mapping for the fourth application object to be mapped	p8713.3	SDO	Unsigned32	0	rw
1604		Receive PDO 5 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8714.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8714.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8714.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8714.3	SDO	Unsigned32	0	rw
1605		Receive PDO 6 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8715.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8715.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8715.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8715.3	SDO	Unsigned32	0	rw
1606		Receive PDO 7 mapping parameter					

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8716.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8716.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8716.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8716.3	SDO	Unsigned32	0	rw
1607		Receive PDO 8 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8717.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8717.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8717.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8717.3	SDO	Unsigned32	0	rw

Note

Each additional drive begins with the description in an offset of 40 hex.

5.2.2 Table: communication objects for transmit PDOs

Overview

The following table lists the object directory with the index of the individual drive-dependent communication objects for the transmit PDOs of the first drive object:

Table 5-3 Drive-dependent communication objects for transmit PDOs

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
1800		**Transmit PDO 1 communication parameter**					
	0	**Largest sub-index supported**		**SDO**	**Unsigned8**	**5**	**ro**
	1	**COB ID used by PDO**	**p8720.0**	**SDO**	**Unsigned32**	**180 hex + node ID**	**rw**
	2	**Transmission type**	**p8720.1**	**SDO**	**Unsigned8**	**FE hex**	**rw**
	3	**Inhibit time**	**p8720.2**	**SDO**	**Unsigned16**	**0**	**rw**
	4	**Compatibility entry**	**p8720.3**	**SDO**	**Unsigned8**	**3**	**rw**
	5	**Event timer**	**p8720.4**	**SDO**	**Unsigned16**	**0**	**rw**
1801		Transmit PDO 2 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	5	ro
	1	COB ID used by PDO	p8721.0	SDO	Unsigned32	280 hex + node ID	rw
	2	Transmission type	p8721.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8721.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8721.3	SDO	Unsigned8	0	rw
	5	Event timer	p8721.4	SDO	Unsigned16	0	rw
1802		Transmit PDO 3 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	5	
	1	COB ID used by PDO	p8722.0	SDO	Unsigned32	380 hex + node ID	rw
	2	Transmission type	p8722.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8722.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8722.3	SDO	Unsigned8	0	rw
	5	Event timer	p8722.4	SDO	Unsigned16	0	rw
1803		Transmit PDO 4 communication parameter					

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
	0	Largest sub-index supported		SDO	Unsigned8	5	ro
	1	COB ID used by PDO	p8723.0	SDO	Unsigned32	480 hex + node ID	rw
	2	Transmission type	p8723.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8723.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8723.3	SDO	Unsigned8	0	rw
	5	Event timer	p8723.4	SDO	Unsigned16	0	rw
1804		Transmit PDO 5 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	5	ro
	1	COB ID used by PDO	p8724.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8724.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8724.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8724.3	SDO	Unsigned8	0	rw
	5	Event timer	p8724.4	SDO	Unsigned16	0	rw
1805		Transmit PDO 6 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	5	ro
	1	COB ID used by PDO	p8725.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8725.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8725.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8725.3	SDO	Unsigned8	0	rw
	5	Event timer	p8725.4	SDO	Unsigned16	0	rw
1806		Transmit PDO 7 communication parameter					
	0	Largest sub-index supported		SDO	Unsigned8	5	ro
	1	COB ID used by PDO	p8726.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8726.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8726.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8726.3	SDO	Unsigned8	0	rw
	5	Event timer	p8726.4	SDO	Unsigned16	0	rw
1807		Transmit PDO 8 communication parameter					
	0	Largest sub-index		SDO	Unsigned8	5	ro

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
		supported					
	1	COB ID used by PDO	p8727.0	SDO	Unsigned32	C000 06E0 hex	rw
	2	Transmission type	p8727.1	SDO	Unsigned8	FE hex	rw
	3	Inhibit time	p8727.2	SDO	Unsigned16	0	rw
	4	Compatibility entry	p8727.3	SDO	Unsigned8	0	rw
	5	Event timer	p8727.4	SDO	Unsigned16	0	rw
1A00		**Transmit PDO 1 mapping parameter**					
	0	**Number of mapped application objects in PDO**		**SDO**	**Unsigned8**	**1**	**ro**
	1	**PDO mapping for the first application object to be mapped**	**p8730.0**	**SDO**	**Unsigned32**	**6041 hex**	**rw**
	2	**PDO mapping for the second application object to be mapped**	**p8730.1**	**SDO**	**Unsigned32**	**0**	**rw**
	3	**PDO mapping for the third application object to be mapped**	**p8730.2**	**SDO**	**Unsigned32**	**0**	**rw**
	4	**PDO mapping for the fourth application object to be mapped**	**p8730.3**	**SDO**	**Unsigned32**	**0**	**rw**
1A01		Transmit PDO 2 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	2	ro
	1	PDO mapping for the first application object to be mapped	p8731.0	SDO	Unsigned32	6041 hex	rw
	2	PDO mapping for the second application object to be mapped	p8731.1	SDO	Unsigned32	606C hex	rw
	3	PDO mapping for the third application object to be mapped	p8731.2	SDO	Unsigned32	0	rw

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
	4	PDO mapping for the fourth application object to be mapped	p8731.3	SDO	Unsigned32	0	rw
1A02		Transmit PDO 3 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	2	ro
	1	PDO mapping for the first application object to be mapped	p8732.0	SDO	Unsigned32	6041 hex	rw
	2	PDO mapping for the second application object to be mapped	p8732.1	SDO	Unsigned32	6074 hex	rw
	3	PDO mapping for the third application object to be mapped	p8732.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8732.3	SDO	Unsigned32	0	rw
1A03		Transmit PDO 4 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	2	ro
	1	PDO mapping for the first application object to be mapped	p8733.0	SDO	Unsigned32	6041 hex	rw
	2	PDO mapping for the second application object to be mapped	p8733.1	SDO	Unsigned32	6063 hex	rw
	3	PDO mapping for the third application object to be mapped	p8733.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8733.3	SDO	Unsigned32	0	rw
1A04		Transmit PDO 5					

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
		mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8742.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8742.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8742.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8742.3	SDO	Unsigned32	0	rw
1A05		Transmit PDO 6 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8752.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8752.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8752.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8752.3	SDO	Unsigned32	0	rw
1A06		Transmit PDO 7 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Predefined connection set	read/ write
	1	PDO mapping for the first application object to be mapped	p8752.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8752.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8752.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8752.3	SDO	Unsigned32	0	rw
1A07		Transmit PDO 8 mapping parameter					
	0	Number of mapped application objects in PDO		SDO	Unsigned8	0	ro
	1	PDO mapping for the first application object to be mapped	p8752.0	SDO	Unsigned32	0	rw
	2	PDO mapping for the second application object to be mapped	p8752.1	SDO	Unsigned32	0	rw
	3	PDO mapping for the third application object to be mapped	p8752.2	SDO	Unsigned32	0	rw
	4	PDO mapping for the fourth application object to be mapped	p8752.3	SDO	Unsigned32	0	rw

Note

Each additional drive begins with the description in an offset of 40 hex.

5.3 Objects in drive profile DSP402

Overview

The following table lists the object directory with the index of the individual objects for the drives. The column "SINAMICS parameter" shows the parameter range in which they are located for SINAMICS S120.

CANopen currently supports profile velocity mode for SINAMICS S120.

Table 5-4 Objects in drive profile DSP402

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
Predefinitions							
67FF		Single device type		SDO			
Common entries in the object dictionary							
6007		Abort connection option code	p8641	SDO	Integer32	0	rw
6502		Supported drive modes		SDO	Integer32		
6504		Drive manufacturer		SDO	String		
Device control							
6040		controlword	p8890	PDO/SDO	Unsigned16	–	rw
6041		statusword	r8784	PDO/SDO	Unsigned16	–	ro
6060		Modes of operation	p1300	SDO	Integer16	-	rw
6061		Modes of operation display	p1300	SDO	Integer16	-	rw
Factor group							
6094		Velocity encoder factor					
	01	Velocity encoder factor numerator		SDO	Integer32		
	02	Velocity encoder factor denumerator		SDO	Integer32		
Profile velocity mode							
6069		Velocity sensor actual value	r0061	SDO/PDO	Integer32	–	ro
6063		Actual position value	r0482	SDO/PDO	Integer32	-	ro
606B		Velocity demand value	r1170	SDO/PDO	Integer32	–	ro
606C		Velocity actual value Actual velocity	r0063		Integer32	–	ro
6071		Target torque Set torque	p1513[0]		Integer16	–	rw
6074		Torque demand value	r0080		Integer16	–	ro

OD index (hex)	Sub-index (hex)	Object name	SINAMICS parameter	Transmission	Data type	Default values	read/ write
		Actual torque					
60FF		Target velocity Set velocity	Without ramp-function generator -> p1155[0] With ramp-function generator -> p1070	SDO/PDO	Integer32	–	rw

Note

The drive objects for further SINAMICS drive objects in the drive profile are described with an offset of 800 hex.

Parameters, faults and alarms, function diagrams, terminology

A

Section content

Note

All the parameters, faults, alarms, and function diagrams for CANopen in the SINAMICS S120 drive line-up are described in the List Manual SINAMICS S120 /LH1/.

This section describes the parameters, faults, alarms, and function diagrams for CANopen in the SINAMICS S120 line-up that are either not covered or are mentioned only briefly in the List Manual.

The relevant description will be removed from this section as soon as the parameter, fault, alarm, or function diagram is included in the List Manual.

A.1 Parameters

The CANopen parameters are described in the List Manual SINAMICS S120 /LH1/.

A.2 Faults and alarms

The CANopen faults and alarms are described in the List Manual SINAMICS S120 /LH1/.

A.3 Function diagrams

Overview

The following function diagrams are relevant for CANopen (see the diagrams on the following pages):

- "Predefined connection set"
  - CAN 10 Communication Board (CBC10) - receive telegram for predefined connection set
  - CAN 10 Communication Board (CBC10) - transmit telegram for predefined connection set
- "Free PDO mapping"
  - CAN 10 Communication Board (CBC10) - receive telegram for free PDO mapping
  - CAN 10 Communication Board (CBC10) - transmit telegram for free PDO mapping

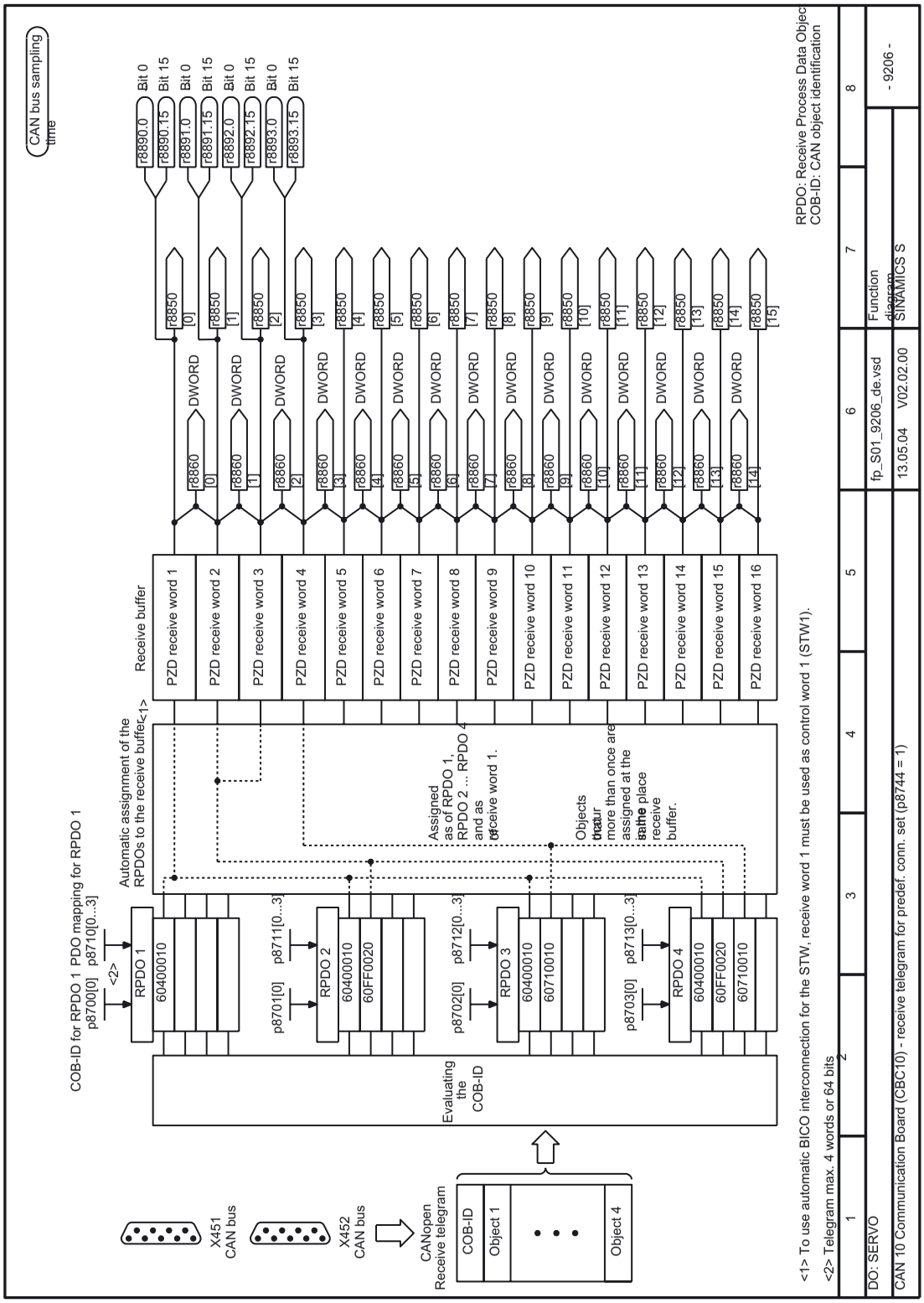


Figure A-1 CAN 10 Communication Board (CBC10) - receive telegram for predefined connection set

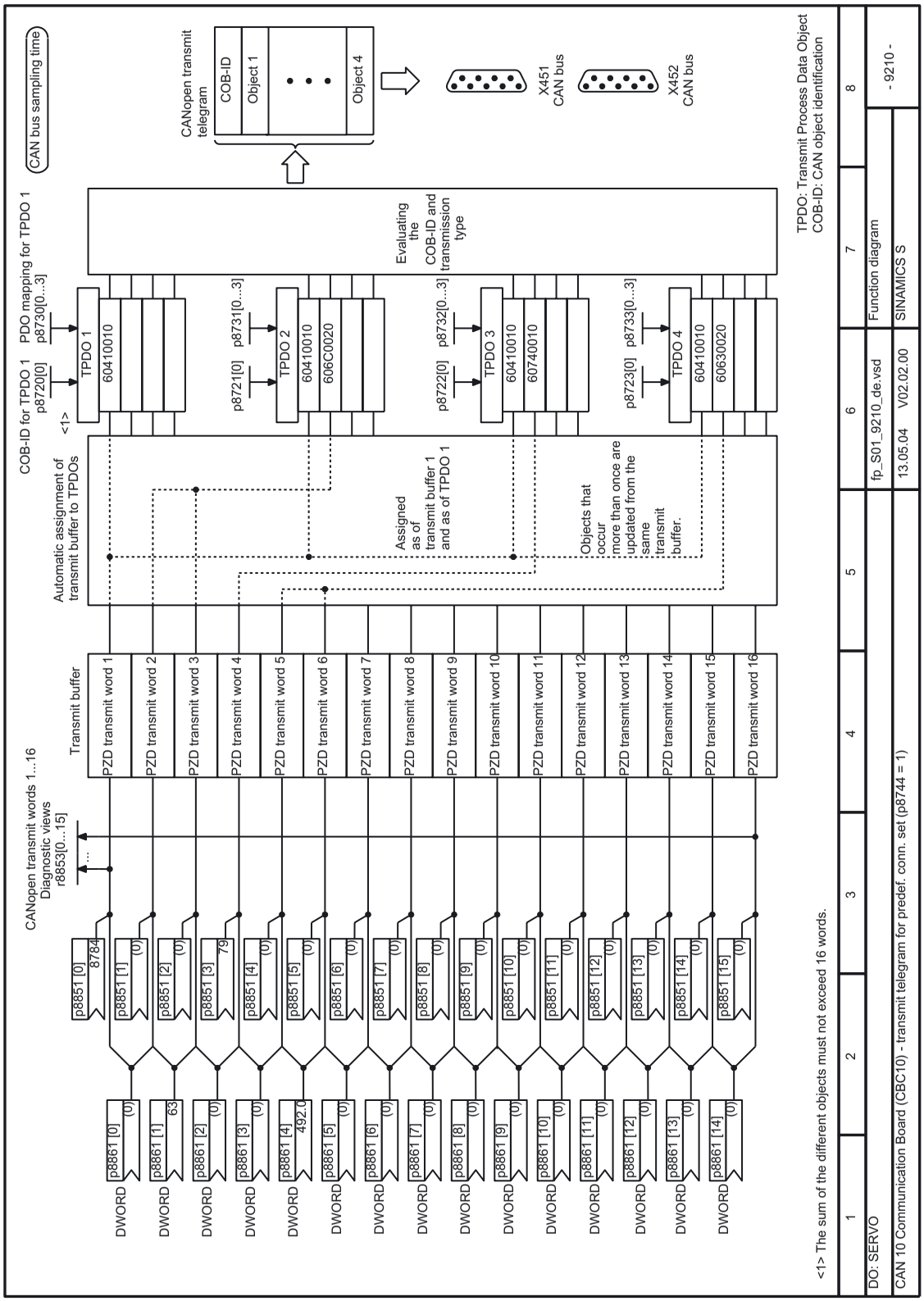


Figure A-2 CAN 10 Communication Board (CBC10) - transmit telegram for predefined connection set

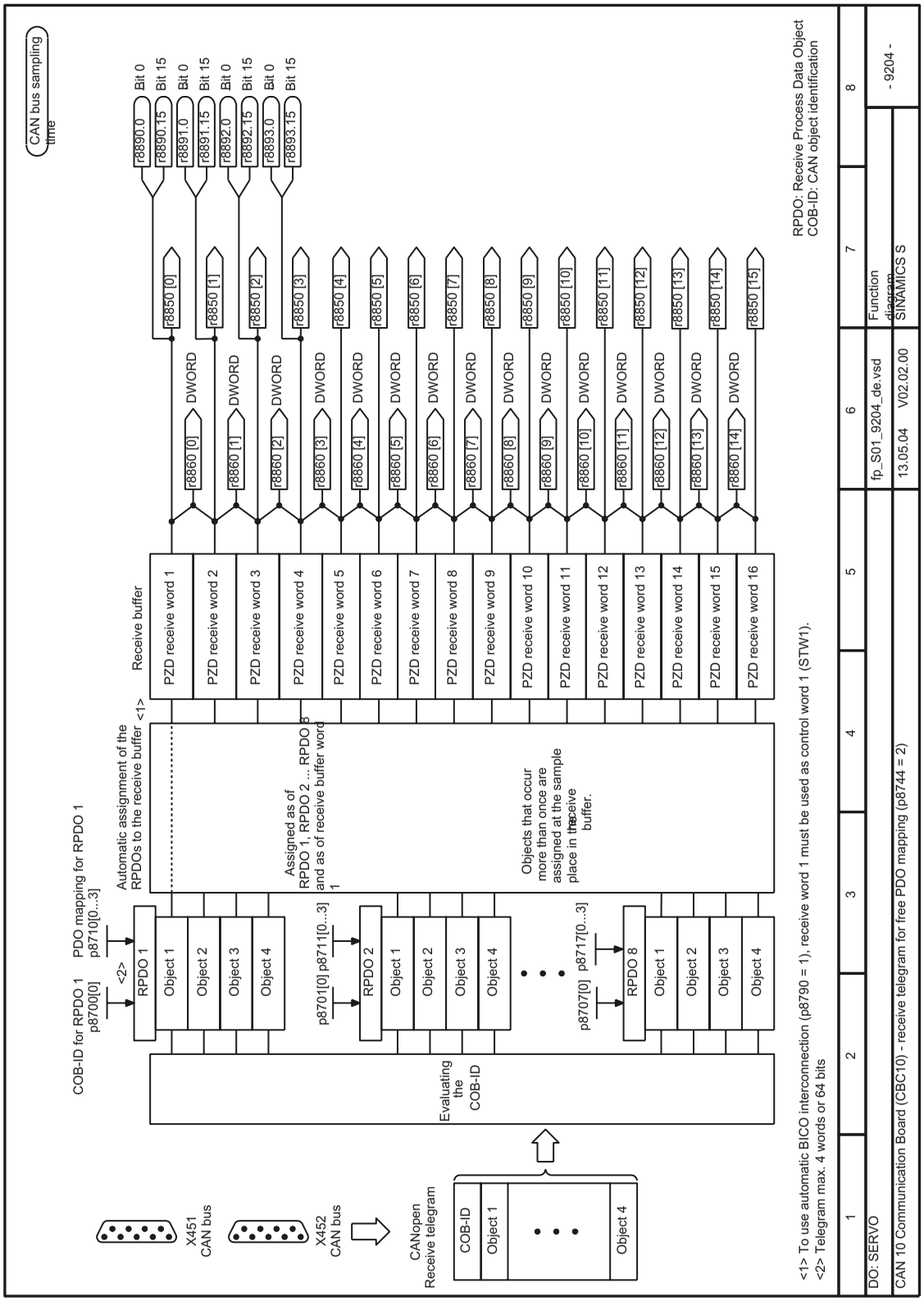


Figure A-3 CAN 10 Communication Board (CBC10) - receive telegram for free PDO mapping

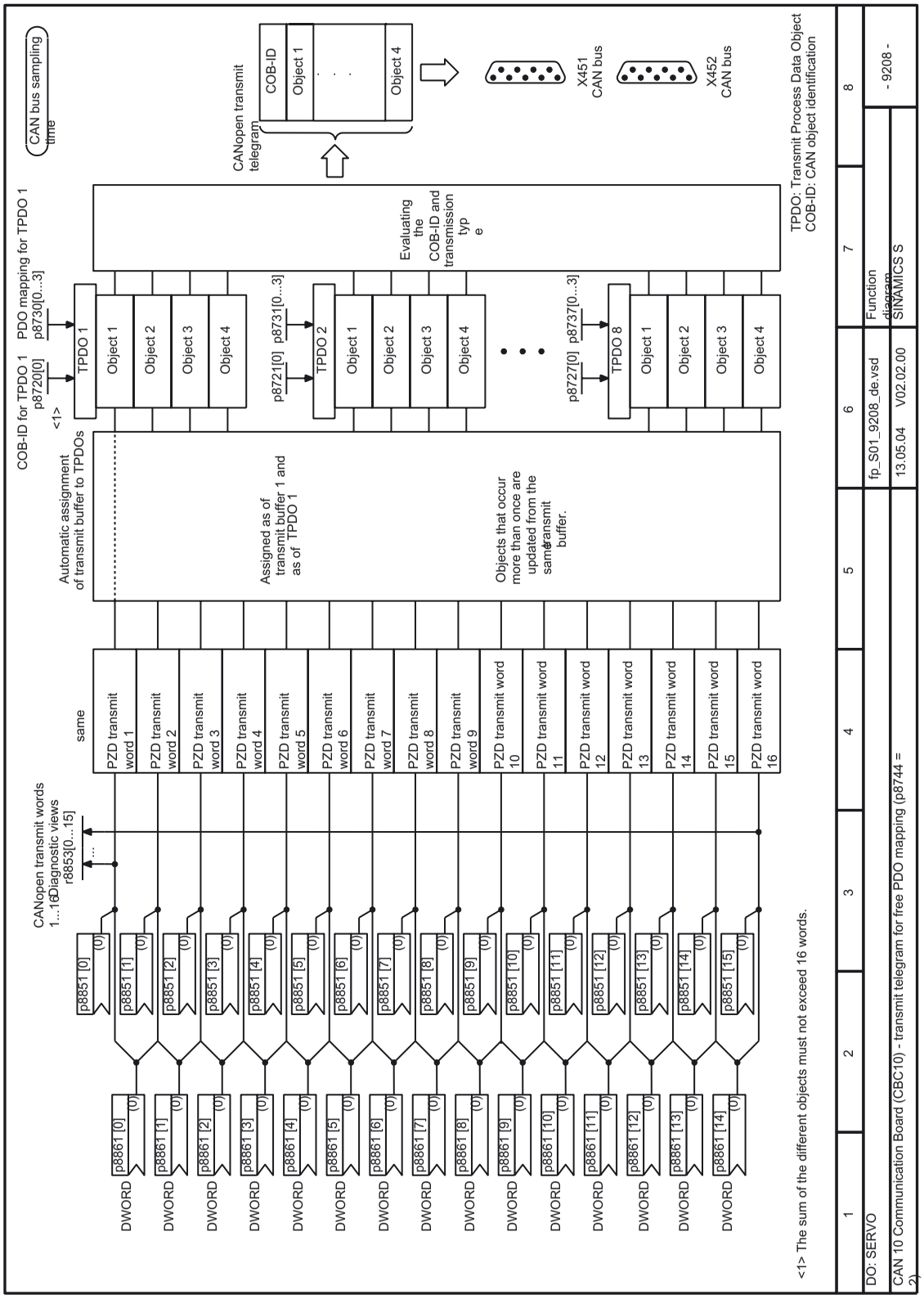


Figure A-4 CAN 10 Communication Board (CBC10) - transmit telegram for free PDO mapping

A.4 Glossary

---------- Explanation of this glossary (abridged) ----------

Term in German Term in English 1) Abbrev.1)

Definition of the term in English 1) -> if available

---------- Explanation of this glossary ----------

Active Line Module Active Line Module none

Controlled, self-commutating feed/feedback unit (with –> "IGBT"s in feed/feedback device), which supplies the DC link voltage for the –> "Motor Module"s.

Antrieb Drive none

The drive includes the motor (electric or hydraulic), the actuator (converter, valve), the control unit, measuring system, and supply components (line infeed module, pressure reservoir).
For electric drives, a distinction is made between a converter system and an inverter system. With a converter system (e.g. –> "MICROMASTER 4"), the line infeed, the actuator, and the control component form a single device from the point of view of the user. With an inverter system (e.g. –> "SINAMICS S"), the supply is ensured by means of –> "Line Module"s, thereby realizing a DC line to which the –> "Inverter"s (– "Motor Module"s) are connected. The –> "Control unit" is implemented as a separate device and connected to the other components by means of –> "DRIVE-CLiQ".

Antriebsgerät Drive Unit none

The drive unit includes all the components connected via –> "DRIVE-CLiQ" that are required for carrying out drive tasks: -> "Motor Module" –> "Control Unit" –> "Line Module", and the required –> "Firmware" and –> "Motor"s, but not additional components, such as filters or reactors.
Several –> "Drive"s can be implemented in a drive unit.
See –> "Drive System".

Antriebskomponente Drive Component none

Hardware component connected to a –> "Control Unit" via –> "DRIVE-CLiQ", for example. –> "Motor Module"s, –> "Line Module"s, –> "Motor"s, –> "Sensor Module"s, and –> "Terminal Module"s.
The overall arrangement of a Control Unit including the connected drive components is called a –> "Drive Unit".

Antriebsobjekt Drive Object DO

A drive object is a self-contained software function with its own –> "Parameter"s and, if necessary, its own –> "Fault"s and –> "Alarm"s. The drive objects may exist by default (e.g. On Board I/O) and may be easy to create (e.g. –> "Terminal Board" 30, TB30). It may also be possible to create them more than once (e.g. –> "Servo Control"). As a rule, each drive object has its own –> "STARTER" window for parameterization and diagnostic purposes.

Antriebs-Parameter Drive Parameter none

Parameters of a drive axis that include, for example, the parameters of the corresponding controllers, as well as the motor and encoder data. The parameters of the higher-level technology functions (positioning, ramp-function generator), however, are called –> "Application Parameters".
See –> "Basic Unit System".

Antriebssystem Drive system none

The drive system includes all the components in a product family (e.g. SINAMICS) that belong to a drive. A drive system comprises, for example, –> "Line Module"s, –> "Motor Module"s, –> "Encoder"s, –> "Motor"s, –> "Terminal Module"s, and –> "Sensor Module"s, as well as additional components (reactors, filters, cables, etc.).
See –> "Drive Unit".

Antriebsverband Drive line-up none

A drive line-up comprises a –> "Control Unit" as well as the –> "Motor Module"s and –> "Line Module"s connected via -DRIVE-CLiQ.

Basic Infeed Basic Infeed none

Overall functionality of an infeed with –> "Basic Line Module", including the required additional components (filters, switching devices, etc.).

Basic Line Module Basic Line Module none

Unregulated line infeed unit (diode bridge or thyristor bridge, without feedback) for rectifying the line voltage of the –> "DC Link".

CompactFlash Card CompactFlash Card none

Memory card for non-volatile storage of the drive software and corresponding –> "Parameter"s. The memory card can be plugged into the –> "Control Unit" from outside.

Control Unit Control Unit CUxxx

Central control module in which the closed-loop and open-loop control functions for one or more –> "SINAMICS" –> "Line Module"s and/or –> "Motor Module"s are implemented.
There are three types of Control Unit:

- SINAMICS Control Units (e.g. –> "CU320"
- SIMOTION Control Units (e.g. –> "D425" and –> "D435"
- SINUMERIK Control Units (e.g. NCU710, NCU720, and NCU730)

CU320 CU320 none

SINAMICS –> "Control Unit" with 4 –> "DRIVE-CLiQ socket"s and 16 digital inputs/outputs.

Double Motor Module Double Motor Module keine

Two motors can be connected to and operated with a Double Motor Module.
See –> "Motor Module" –> "Single Motor Module"
Former term: –> "Double-axis module"

DRIVE-CLiQ DRIVE-CLiQ none

Abbreviation for "Drive Component Link with IQ".
Communication system for connecting the different components of SINAMICS drive system (e.g. –> "Control Unit" –> "Line Module"s –> "Motor Module"s –> "Motor"s and speed/position encoders.
The DRIVE-CLiQ hardware is based on the Industrial Ethernet standard and uses twisted-pair lines. The DRIVE-CLiQ line provides the transmit and receive signals, as well as the +24 V power supply.

Einspeisung Feeding Section none

Input component of a converter system for generating a DC link voltage to supply one or more –> "Motor Module"s, including all the required components (e.g. –> "Line Module"s, fuses, reactors, line filters, and firmware, as well as proportional computing power (if required) in a –> "Control Unit".

externer Geber External encoder none

Position encoder that is not built in or mounted on the –> "Motor", but fitted instead via a mechanical transmission element or mechanical intermediate element.
The external encoder (see –> "Externally-Mounted Encoder") is used for –> "Direct Position Detection".

Geber Encoder none

An encoder is a measuring system that captures actual values for the speed and/or angular/position values and makes them available for electronic processing. Depending on the mechanical construction, encoders can be integrated in the –> "Motor" (–> "Motor Encoder") or mounted on the external mechanics (– "External Encoder"). Depending on the type of movement, a distinction is made between rotary encoders ("rotary transducers") and translatory encoders (e.g. –> "Linear Encoder"s). In terms of measured value provision, a distinction is made between –> "Absolute Encoder"s (code sensors) and –> "Incremental Encoder"s.
See –> "Incremental Encoder TTL/HTL" –> "Incremental Encoder sin/cos 1 Vpp" –> "Resolver".

Line Module Line Module none

A Line Module is a power component that generates the DC link voltage for one or more –> "Motor Module"s from a 3-phase mains voltage.
In SINAMICS, three types of Line Module are available:
–> "Basic Line Module", –> "Smart Line Module" –> "Active Line Module".
The overall function of an infeed, including the required additional components (–> "Line Reactor", proportional computing power in a –> "Control Unit", switching devices, etc.) is called –> "Basic Infeed", –> "Smart Infeed", and –> "Active Infeed".

Motor Motor none

For the electric motors that can be driven by –> "SINAMICS", a basic distinction is made between rotary and linear motors with regard to their direction of motion, and between synchronous and induction motors with regard to their electromagnetic operating principle. In SINAMICS, the motors are connected to a –> "Motor Module".
See –> "Synchronous Motor" –> "Induction Motor" –> "Built-In Motor" –> "Motor Encoder" –> "External Encoder" –> "Third-Party Motor".

Motor Module Motor Module none

A Motor Module is a power component (DC-AC inverter) that supplies the power for the connected motor(s).
Power is supplied through the –> "DC Link" of the –> "Drive Unit".
A Motor Module must be connected to a –> "Control Unit" via –> "DRIVE-CLiQ". The open-loop and closed-loop control functions for the Motor Module are stored in the Control Unit.
–> "Single Motor Module"s and –> "Double Motor Module"s are available.

Motorgeber Motor Encoder none

An –> "Encoder" (e.g. –> "Resolver", –> "Incremental Encoder TTL/HTL", or –> "Incremental Encoder sin/cos 1 Vpp" that is integrated in or attached to the motor.
The encoder detects the motor speed. In the case of synchronous motors, it can also detect the rotor position angle (of the commutation angle for the motor currents).
For drives without an additional –> "Direct Position Measuring System", it is also used as a –> "Position Encoder" for position control.
In addition to the motor encoders, –> "External Encoder"s for –> "Direct Position Sensing" are available.

Option Board Option Board

PC board inserted in the –> "Control Unit" (e.g. a –> "Terminal Board" 30, TB30).

Option Slot Option Slot none

Slot for an optional module (e.g. in the –> "Control Unit").

Parameter Parameter keine

Variable quantity within the drive system that the user can read and, in some cases, write.
For –> "SINAMICS", all specifications defined in the –> "PROFIdrive" profile are defined by a

parameter.
See –> "Visualization Parameter"s and –> "Adjustable Parameter"s.

PROFIBUS PROFIBUS none

Field bus to IEC 61158, Sections 2 to 6.
The abbreviation "DP" is no longer included because PROFIBUS FMS is not standardized and PROFIBUS PA (for Process Automation) is now part of the "general" –> "PROFIBUS".

Sensor Module Sensor Module SMCxx
SMExx
SMIxx

Hardware module for evaluating speed/position encoder signals and providing detected actual values as numerical values at a –> "DRIVE-CLiQ Socket".
Three mechanical Sensor Module variants are available:
- SMCxx = Sensor Module Cabinet-Mounted
- SME = Sensor Module Externally Mounted (with a high degree of protection)
- SMI = Sensor Module Internal (integrated in the motor flange outlet).

Servoantrieb Servo Drive none

An electric servo drive comprises a motor, a –> "Motor Module", a –> "Servo Control" and, in most cases, a speed and position –> "Encoder"
Electric servo drives are normally extremely precise and have a high dynamic response. They are designed for cycle times to less than 100 ms, and often have a short-time overload capacity, which enables quick acceleration. Servo drives are available as rotary and linear drives and are used for machine tools, handling robots, and packaging machines.

Servoregelung Servo Control none

This type of control enables operation with an extremely high –> "Dynamic Response" and –> "Precision" for –> "Motor"s with a –> "Motor Encoder".
In addition to speed control, position control can be implemented.

SITOP power SITOP Power none

–> "Electronics Power Supply" component.
Example: 24 V DC

Smart Line Module Smart Line Module none

Unregulated line infeed/feedback with a diode bridge for the infeed and stall-protected, line-commutated feedback via –> "IGBT"s.
The Smart Line Module provides the DC link voltage for the –> "Motor Module"s.

STARTER STARTER none

STARTER assists with the startup and parameterization of the drive units. This tool can also be used to execute the diagnostic functions required during servicing (e.g. PROFIBUS diagnostics, function generator, trace).
See –> "SIZER" –> "Engineering System".

Steuerwort Control Word STW

Bit-coded –> "Process data" word. –> "PROFIdrive" transmits this word at cyclic intervals to control the drive states.

Terminal Board Terminal Board TBxx

Terminal expansion board that can be plugged in to the –> "Control Unit".
In –> "SINAMICS", for example, Terminal Board 30 (TB30) is available with analog and digital I/O terminals.

Terminal Module Terminal Module TMxx

Terminal expansion board that can be snapped onto the installation rail for installation in a cabinet unit.
In –> "SINAMICS", the following Terminal Modules are available:
- TM3x = Terminal Modules with digital and analog I/O terminals
- TM4x = Terminal Modules with encoder emulation

Vektorregelung Vector Control

Vector control (field-oriented control) is a high-performance control type for induction machines. It is based on an exact model calculation of the motor and two current components that simulate and accurately control the flux and torque by means of software algorithms, thereby enabling predefined speeds and torques to be observed and limited accurately and with a good dynamic response.
Two vector control types exist:
Frequency control (–> "Sensorless Vector Control") and speed-torque control with speed feedback (–> "Encoder").

Zustandswort Status Word ZSW

Bit-coded –> "Process Data" word. –> "PROFIdrive" transmits this word at cyclic intervals to control the drive states.

Index

O

P

R

S

T

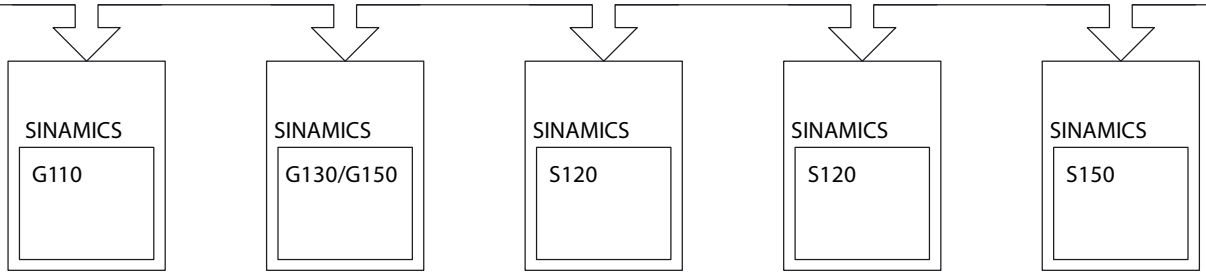
U

V

W

Overview of SINAMICS Documentation (12.2004)

General Documentation/Catalogs



D 11.1
Converter 0.12–3 kW
Chassis Units

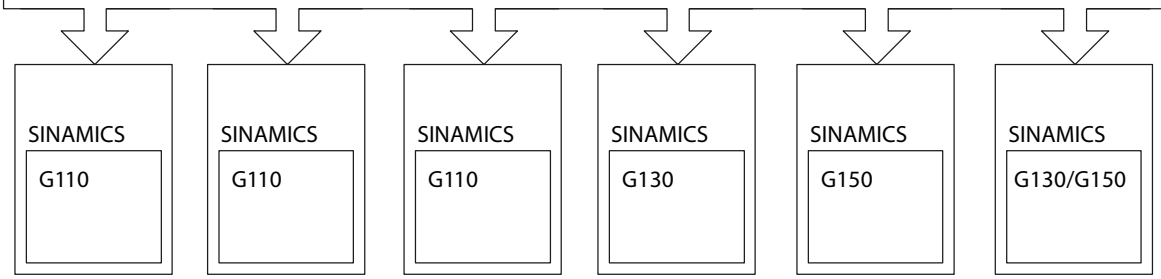
D11
Converter 75–800 kW
Chassis/Cabinet Units

D 21.1
Vector Control
Drive System
(incl. Motors)

D 21.2
Servo Control
Drive System
(incl. Motors)

D 21.3
Vector Control
75–1200 kW
Cabinet Units

Manufacturer/Service Documentation



Operating
Instructions

List Manual
Parameters,
Function Diagrams

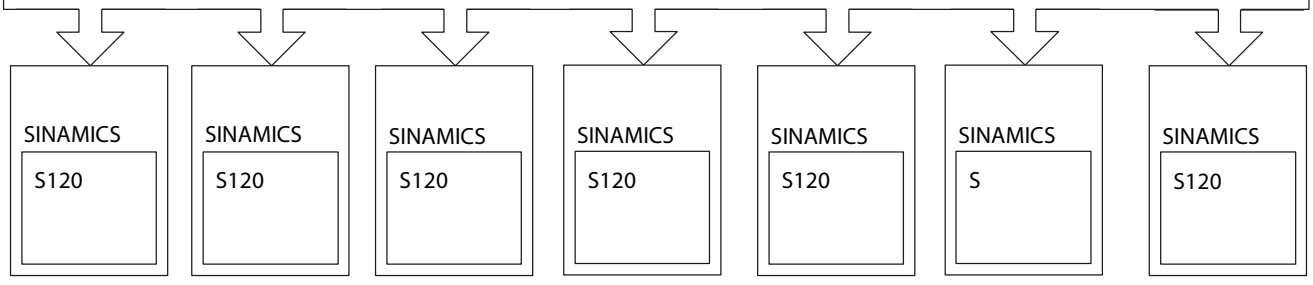
Getting Started

Operating
Instructions

Operating
Instructions

List Manual
Parameters,
Function Diagrams

Manufacturer/Service Documentation



Equipment
Manual
Control Unit,
Additional System
Components

Equipment
Manual
Booksize Power
Sections

Equipment
Manual
Chassis Power
Sections

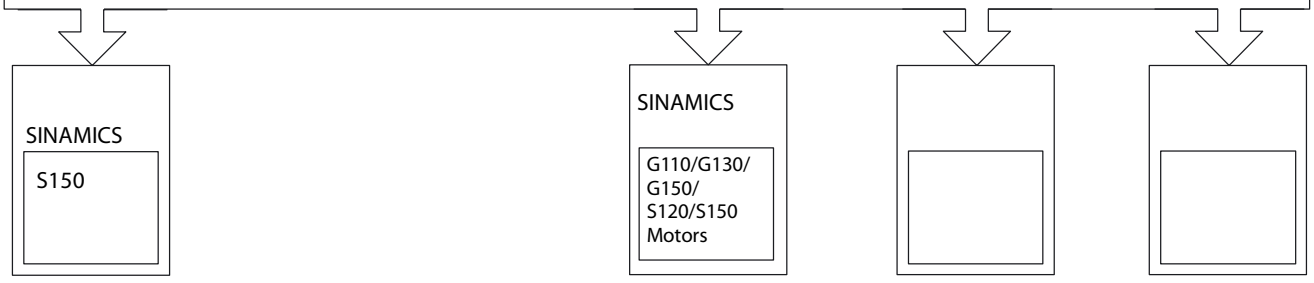
Installation &
Start-Up Manual

Getting Started

List Manual
Parameters,
Alarms
Function Diagrams

Function
Manual

Manufacturer/Service Documentation



Operating
Instructions

DOCONCD

Planning Guide
Motors
1FT6; 1FK7;
1PH7; 1PH4; 1PH2

EMC
Installation
Guideline

Siemens AG
Automation & Drives
Motion Control Systems
P.O. Box 3180, D -91050 Erlangen
Germany

www.siemens.com/motioncontrol

© Siemens AG 2004
Subject to change without prior notice
Order no.: 6SL3097-2AA00-0BP1

Printed in Germany