User Interface and Bus Systems

NORDAC SK 1000E

Servo- Controller

SK 1000E-101-340-A ... SK 1000E-102-340-A

T.-Nr. 0604 1292

BU 1200 GB
12/ 2003

Getriebebau NORD
GmbH & Co. KG

1 General

1.1 NORDAC SK 1000E Servo- Controller

The dynamic, intelligent model from NORD
The SK 1000E is the best choice where rapid acceleration, fast movement and high standstill moment, coupled with high speed and positioning precision, are essential. Combined with the integrated PLC, technology functions and CANopen field bus connection, it is the solution for demanding drive problems.

Rated current: 3.5-20Aeff, 3 AC380-460V 50-60Hz
Speed range: Servo- controller with resolver +/- 22500 rpm,
Servo-controller with incremental encoder is proportionally resolution-dependent e.g. at 4096 lines +/- 18500 rpm (**note:** observe the threshold frequency for the encoder, this is often only 250kHz!)

NORDAC SK 1000E servo controllers are designed using the latest fully digital signal processor technology (DSP) with field bus connections for moment, speed and positioning regulation of synchronous and asynchronous motors.

Due to the field-orientated control concept with current, speed and positioning regulation as well as operational profile connection with just 50 ms sampling time, this system has extremely high dynamics and full motor moment even when at a standstill. Extremely high, load-independent speed and positioning accuracy is achieved by the rotor position feedback. Demanding drive tasks can therefore be resolved. Implemented technology functions emphasis customer benefits.

An integrated mini-PLC rounds off the controller's function spectrum.

The controller is commissioned with control software, which runs on Windows 95/98, 2000, NT and XP. Set-up has been significantly simplified, due to the motor / controller database, oscilloscope and numerous test functions, which are supplied, and so can be carried out by anyone.

Some specifications are still in preparation. Further technical developments may necessitate deviation from this product description and may be collected as appropriate, in order to be incorporated in this product description at a later point in time.

1.2 Variants and Options

Variant	Description	Data
R for Resolver	Feedback = Resolver	12 bit resolution
E for Encoder	Feedback = Incremental Encoder Hall addition recommended	Free line number selection from 500 increment resolution
RS	Resolver with 1 Mbaud CAN option and absolute value encoder, no stepper motor interface	
ES	Encoder with 1 Mbaud CAN option and absolute value encoder, no stepper motor interface	
RT	Resolver with technology functions option	See Nord Applications company document
ET	Encoder with technology functions option	See Nord Applications company document
RST	Resolver with 1 Mbaud CAN option and absolute value encoder, no stepper motor interface, with technology functions option	
EST	Encoder with 1 Mbaud CAN option and absolute value encoder, no stepper motor interface, with technology functions option	

All servo controllers can operate synchronous and asynchronous motors. Only the dataset of the motor connected is loaded from the database during Set-up.

1.3 Device Characteristics

- Operation of synchronous and asynchronous motors
- High dynamics, motors with up to 0.5ms electrical and mechanical time constants can be operated.
- Full torque at standstill
- Sinusoidal commutation
- Completely digital control concept
- Integrated mains filter A
- Integrated brake chopper with external brake resistor
- Moment, speed and position regulation in a regulation cycle time of 50µs
- Ramp generator / operational profile generator
- New positions, movement speeds and ramps can be set at any time during movement in the positioning regulation
- Stepper motor interface with 5 MHz - input frequency
- Technology functions, e.g. electronic gears, flying saw, winders, disk cams and camshaft

  Note -> request the document NORD applications
- Encoder, resolver or absolute value encoder (SSI)
- Incremental encoder emulation output
- CAN field bus (up to 1 MBit/s), RS 232 and RS 485 (up to 56000 Bit/s)
- CAN- Open protocol DS301V4.01&DS402V1.1, EDS-configuration file is supplied
- Variable PDO mapping
- ±10V analogue set value interface
- 6 freely programmable inputs
- 6 freely programmable short-circuit proof outputs
- 1 freely programmable relay e.g. for integrated brake control
- Integrated PLC with input assistant for particularly simple intuitive operation
- Carry out logical and arithmetic operations with variables and constants
- Integrated reference switch logic
- Integrated end switch monitoring
- Fast and simple Set-up using control software with database and oscilloscope functions
- May be mounted adjacently, without additional clearance
- Permissible ambient temperature up to 40°C
- Suitable for high speeds (encoder-dependent)

2 PC User Interface (NORD SERV)

Note: Further information on all dialogs can be found by clicking the relevant Help buttons! For this reason, not all dialogs are explained in detail in this document.

2.1 Hard and Software Requirements

- IBM-PC/AT and compatible from 486 processor
- 16 MB RAM
- 6 MB free disk space on the hard drive
- Available serial port to connect Servo-controller to PC with NORD SERV (for cable see NORD cable set)
- Mouse or similar system
- Min. resolution 800*600, a resolution of 1024*768 pixels is recommended for the oscilloscope function
- Running Windows 95, 98, Windows NT, Windows ME, Windows2000 or Windows XP

2.2 Terms

NORD SERV = Human-Machine User Interface
Online = the user interface is communicating with the controller via a bus or the RS232
Offline = the user interface is not communicating with a controller

2.3 Installation

The installation assistant starts automatically with the CD supplied. Please select the components required.

2.3.1 Updating the Manufacturer Database

NORD's manufacturer database for the various motor-controller combinations is installed when the NORD SERV program is installed. Because NORD continuously calibrate new motors, users can update the manufacturer database at any time. The source of the new manufacturer database can be a diskette or CD, as well as an Internet download. NORD always makes the latest manufacturer database available on the Internet, or NORD supplies an updated diskette with the new drive set together with the program CD. After installing the NORD SERV CD, select the **Database \ Update** command to update. A dialog box then opens for you to select the directory, in which the new manufacturer database files are located. Just select one of the following files: "Amplifier.dat", "NordDataBase.dat“ or "Motor.dat“ and press Enter. The program then installs the new manufacturer database (all three files) in NORD SERV's working directory. These are now available in NORD SERV and can be loaded into the controller and then saved in the servo controller.
The user database and all PLC programs in NORD SERV are unaffected by this update. In the same way, for example, if you install a new version of NORD SERV in an existing directory, the manufacturer database will be automatically updated, but the user database and PLC programs will be unaffected.

2.4 Functions

NORD SERV is the PC program to parameterise, optimise, program and control servo-controllers manufactured by NORD Gear.

NORD SERV can communicate with up to 16 servo controllers concurrently, via the integrated RS485 interface. Communication takes place via the PC's serial interface and optionally-available RS232/RS485 converter.

For Set-up or test procedures, the connected servo-controllers can be controlled by PC. It is also possible to record the course of control variables, e.g. current, speed, positioning etc. in the servo-controller, load it onto the PC and observe it via the NORD SERV oscilloscope function.

NORD SERV enables creation, documentation and storage of parameter settings. All of the servo-controller's parameter settings can be read out for this purpose. After connecting with the attached servo-controllers, NORD SERV recognises the type (hard and software version or variant) and displays the relevant datasets to the user. Parameter sets can also be created or edited offline, i.e. without servo-controllers being connected.

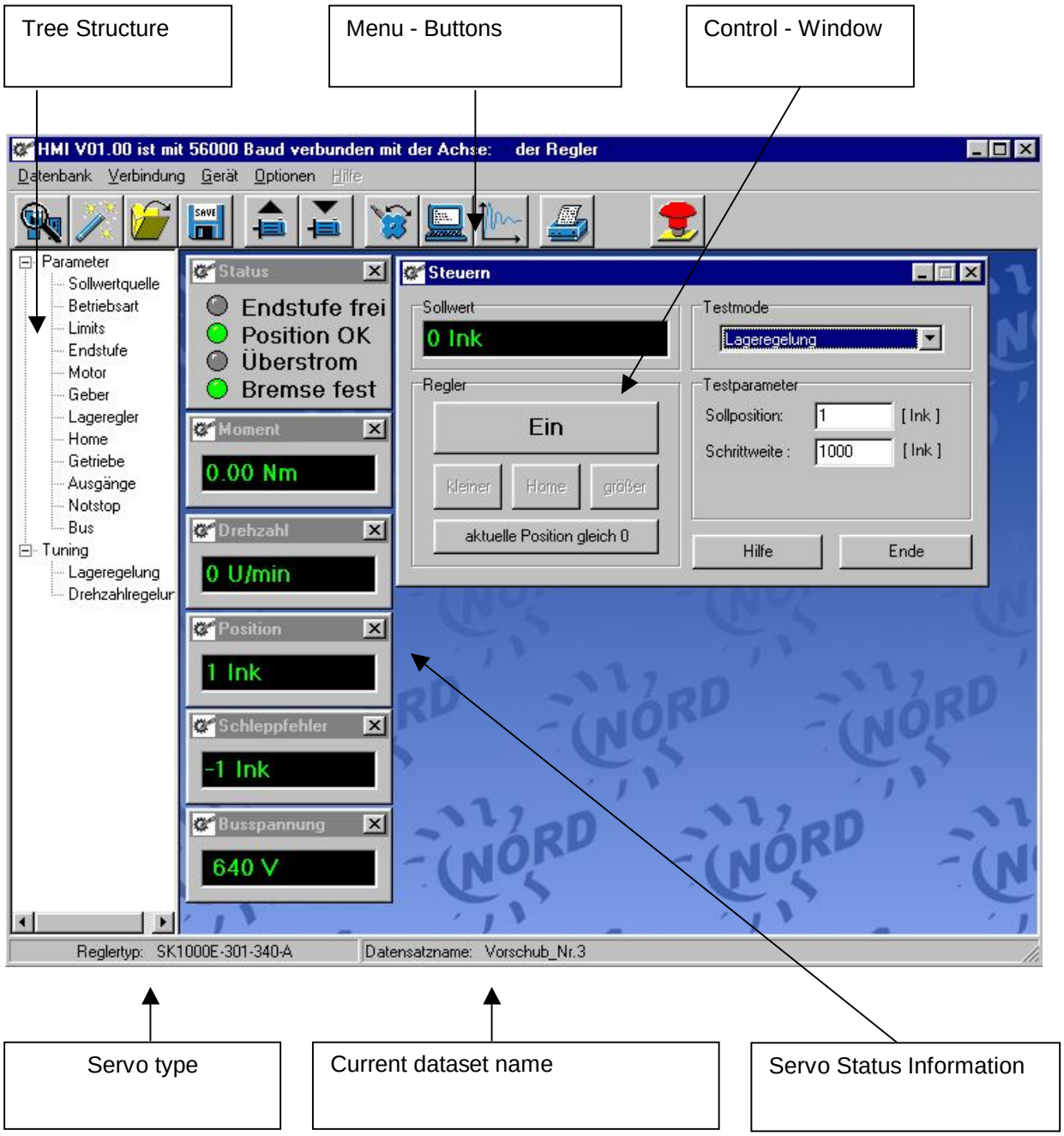
Programming and checking of the PLC contained in the servo-controller also takes place via NORD SERV.

2.4.1 With and without Technology Function

Technology functions are software modules, which can be optionally added to the standard software. Technology functions include e.g. the functions "gears", "winder", "compensator" and "flying saw". Each servo-controller is activated separately. The enable status of technology functions can be viewed under the "Option / Status Technology Functions" command.
NORD Gear will carry out subsequent activation at any time. To do this, you must inform NORD of the device code ("Option / Status Technology Functions"). NORD will then identify the relevant activation code and inform you of it. You can then enter this code in the aforementioned dialog box. Your servo-controller has now been activated and you can now use all of the technology functions in this device as you require.

2.5 Main Elements

The following image shows the user interface as it appears, when a servo-controller is connected with the PC via the RS232 interface.



NORD SERV when a servo-controller is connected to the PC via RS232

Tree Structure
Change the parameters of the servo-controllers with this element; they are sorted into numerous dialog boxes and windows under the headers *"Parameter"* and *"Tuning".* The parameters under the *"Tuning"* header affect optimisation of the servo-controller under the respective working conditions and can also be changed during switched-on state (motor has a current). Changes may only be made to the entries under *"Parameter"* when the servo-controllers are switched off.

Menu Buttons
Important program functions can be called up using the menu buttons. They can also be called up using normal menu commands.

Control Window
The following functions can be carried out via this window:
- Switching the servo-controller on and off,
- Selection of operating mode (e.g. positioning control, speed control, …)
- Specification of axle movement setpoint

Servo-controller Status Information
The 5 windows shown display the servo-controller's most important current values.

The symbols in the *"Status"* window at the top show the current status of the servo-controller.

- "*Output stage available*": Display is lit, when the servo-controller is switched on, it can accept setpoints and the motor has a current.

- "*Position OK*": The axle is within the area of set position ± tolerance, the tolerance is set in the *"Position Controller"* parameter dialog box.

- *"Overcurrent"*: Display is lit, if the servo-controller has been operated in the overcurrent for too long, it will only then work with rated current.

- *"Brake Secure"*: Display is lit, if the brake relay is not applied, a brake may be connected, to secure the axle. If this display is grey, then the servo-controller's braking function has not been configured, this can be carried out in the "Motor" parameter dialog box.

2.6 Communication

Communication with the servo-controller can take place via RS232, RS485 and CAN interfaces. Communication can occur simultaneously via the CAN – Bus and RS232 or RS485.

2.6.1 Scan and Single Mode

The NORD SERV user interface can be operated in two function states.

Single – Mode
In this mode, NORD SERV can only communicate with one servo-controller. This applies both to operation via an RS232 connection and a Bus connection via RS485. The complete functionality of the user interface is available in this mode.
With a bus connection, you can switch between individual servo-controllers, by clicking on the servo-controller required in the tree diagram.

Scan – Mode
In this mode, NORD SERV can recognise and display all servo-controllers connected to the bus simultaneously. The status and current value of the currently active servo-controller mode are display for each servo-controller. No further functionality is available in this mode. However, you can switch to Single – Mode at any time, by clicking on the appropriate servo-controller in the tree diagram or clicking on the servo-controller window directly. Click on the  button, to return to Scan – Mode.

2.6.2 Operation via RS232

NORD SERV is in Single – Mode when operating via the RS232 interface. All of NORD SERV's functions are available in this mode.

2.6.3 Operation via RS485 - Bus

An RS232/RS485 converter is required to operate the RS485 bus via the user interface. NORD Gear recommends the I – 7520 converter. The converter must have automatic direction switching. The *"Controller dead time"* parameter in the *"Setpoint specification"* dialog box may also have to be altered for the converter. This parameter comprises the minimum time period the controller must wait, before responding to a request. This provides a slow RS232/RS485 converter with the opportunity to switch the direction of transfer. Otherwise, the response to a request takes place within 100-200µs.
We recommend you consult NORD if you intend using other converters.
A further condition for operating multiple servo-controllers from one bus is, that all the servo-controllers have different addresses (DIP switch on servo-controller) and that the same baud rate is set.
Once the servo-controllers and PC are connected to the bus, then a scan procedure must be carried out in NORD SERV by clicking the  button. The bus will then be scanned for available servo-controllers. Servo-controllers with the same address and differing baud rates cannot be recognised.
After the bus scan, NORD SCAN will switch to Scan Mode, i.e. only the servo-controllers found can be observed or operated. The servo-controllers are then displayed in the tree diagram on the left of the screen and individually as small status windows. The status of each servo-controller is displayed in these windows by LED symbols. The *"Enable"* LED signifies when lit, that the servo-controller is switched on (PWM operational, motor under current). On the other hand, if the *"Fault"* LED is lit, then the servo-controller has a fault and is unavailable for operation. In addition, the current value of the current servo-controller mode is displayed in the status window, e.g. current speed in speed mode.
However, if you want to parameterise, optimise or control one of the servo-controllers displayed with NORD SERV, click on the servo-controller in the tree diagram or on the respective status window. The interface switches to Single Mode, as when operating via the RS232 interface. All of NORD SERV's functions are available in this mode. If you want to switch to another servo-controller when in Single Mode, click on the respective servo-controller in the tree diagram.

Switch back to the bus mode by clicking on the  button.

2.7 Languages

NORD SERV can be displayed in two languages, German and English. Further languages are being prepared. Switch between languages at any time by clicking on the *"Options/Languages"* command.

2.8 Set-up Assistant

After the NORD SERV program is started, the Set-up Assistant appears with various options.

If the program does not start in the correct language (German or English), close the dialog box and select the language required under the *"Option/Language"* or "*Optionen/Sprachen"* command. Switch back to the Set-up Assistant by clicking on the  button.

The Assistant will now lead you through all the dialog boxes required for set-up:

1. The first dialog box asks, which serial interface on the PC the program should use. This interface may not be occupied by another active program!

2. In the following step, the connection to the servo-controller is made. In the event of any problems, please check the following points:
   - Was the correct interface used?
   - Does the serial connection to the servo-controller exist?
   - Was the correct cable used?
   - IS the servo-controller's supply voltage switched on?

3. After a successful connection has been made, the load dialog box opens automatically, whereyou can select a suitable dataset, which is stored on the PC (from the database). Only the datasets suitable for the servo-controllers connected are displayed in this dialog box. Select the correct motor from the *"Motor Selection"* list field. Only datasets with the suitable motor/servo-controller combination will be displayed in the dialog box.

Be particularly careful on this point, since incorrect motor selection can lead to the motor you use being damaged!

4. Now select a dataset from the *"Manufacturer-defined Datasets"* field. "Reference Sets" created by NORD Gear are displayed in this field. The datasets have been optimised for a specific motor/servo-controller combination, for an unencumbered motor axle. Datasets optimised by clients are stored in the *"User-defined Datasets"* field.

5. After the dataset is selected, a number of dialog windows open, allowing the control parameters of factors relevant to the application to be adjusted. For questions relating to these dialog boxes, please use the online help.

6. After this procedure, the Assistant opens the online screen. On the left side, you can click on the most important current values and see the servo-controller status. Use the *"Control"* window to operate an axle, see *"Controlling a Drive".*
Subsequently, parameters may be altered at any time via the tree diagram on the left of the screen.

2.9 Controlling a Drive

It is possible to operate and test the servo-controller on site with the user interface. This happens using the *"Control"* window, which is called up by clicking the  button.

2.9.1 Switching the servo-controller on and off,

Use the *"On"* button to switch the servo-controller on. When it is switched on, the text on the button changes to *"Off"* and clicking again will switch it off.

Control window

2.9.2 Change Setpoints and send to servo-controller

Use the *"Smaller"* and *"Larger"* buttons to change the setpoints. Each change is immediately sent to the servo-controller and displayed in the "Setpoint" display. The setpoint increments are 5% of the maximum value displayed. The value is transferred from the *"Limits"* dialog box. Setpoints above this maximum value are not possible. Smaller setpoint changes can be made by manually reducing the maximum setpoint.
In the moment and speed modes, click on the Stop button to immediately set the setpoint to 0. Click again to send the setpoint displayed to the servo-controller. This enables setpoint changes to be created, to analyse a change response with the oscilloscope function.

2.9.3 Set Test and Regulation Mode

The following operating types can be set for a switched-off servo-controller in the *"Test Mode"* box:

- Positioning Regulation
- Speed Regulation
- Moment Regulation
- Speed - Reverse
- Moment - Reverse

In the Positioning Regulation mode, the *"Stop"* button becomes *"Home"*, to start a reference run. During this run, the button is unavailable and setpoints cannot be sent. Furthermore, when the motor is switched-off, click on *"Current Position 0"* to set the servo-controller's current position to zero. For test purposes, a reference run like this can be bypassed. When reversing, after the time entered in *"Pulse Width"* has run out, the current setpoint will be negated.

2.10 Trace and Oscilloscope Function

The servo-controller has a trace function. This enables changes to control variables and variables to be recorded over a set time period with a variable sample rate. The tracer's sample frequency can be set from 1 second to a sample frequency of the current controller loop of 50µs. It is possible to record three variables in the servo-controller simultaneously and load them onto the PC, so that they can be displayed and analysed in the NORD SERV oscilloscope function.

Call up the functions with the  button.
The Tracer and Oscilloscope functions are available in the Online mode, when the relevant windows are open. Oscilloscope functions are also available offline.

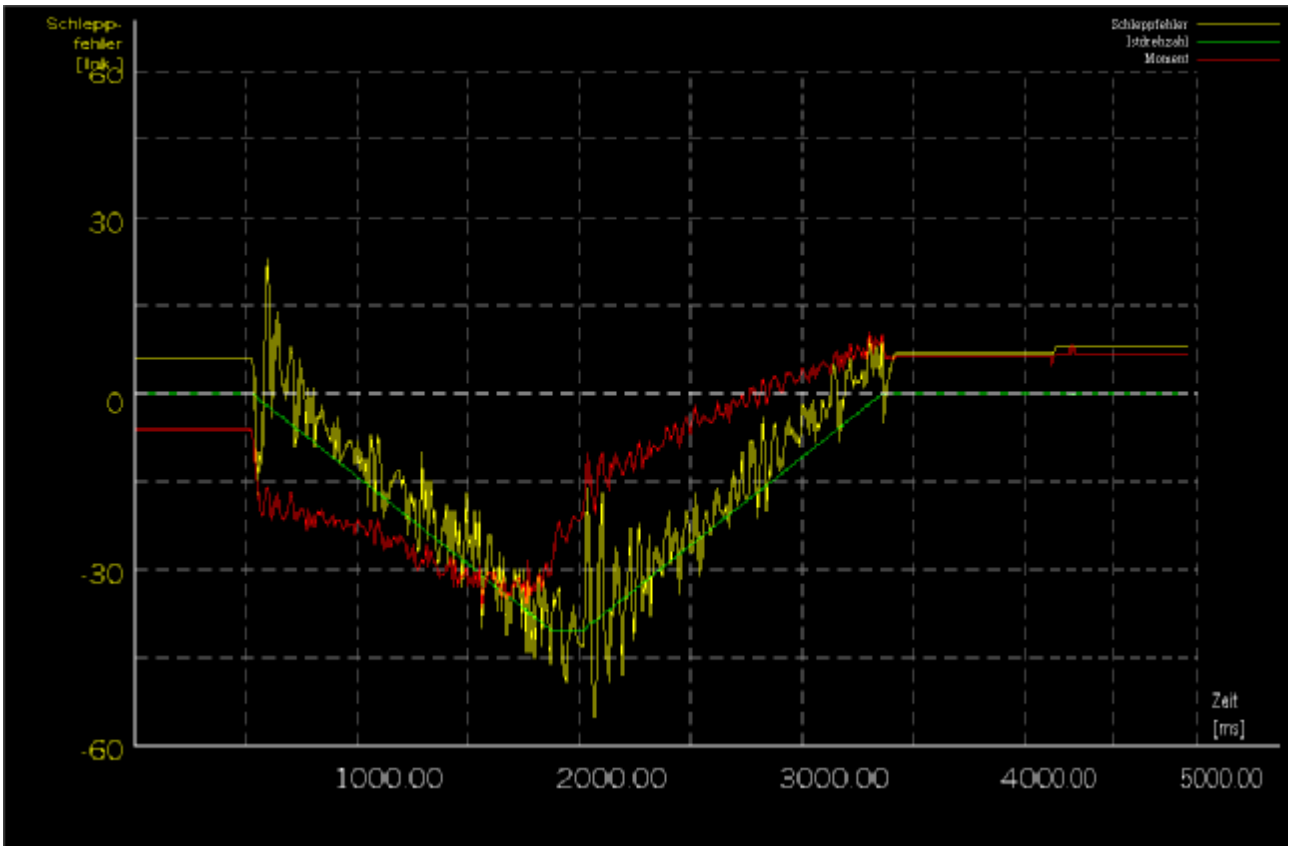
2.10.1 Tracer

Tracer window

Firstly, the tracer in the servo-controller must be configured. Click the "Settings" button. In the dialog box, which now opens, set the number of channels to be measured, the respective values, the sampling time and the duration of measurement. This is explained in greater detail in the Online Help of the dialog box.
Then click the *"Start"* button to start measurement. After measurement has finished, click the *"Readout"* button to load the tracer content onto the PC. When the tracer is restarted, the entire old content is overwritten, unless measurement is interrupted before the set time has elapsed by clicking the *"Readout"* button. In this event, only the area, which has been recorded on up to this point, is overwritten.

2.10.2 Oscilloscope

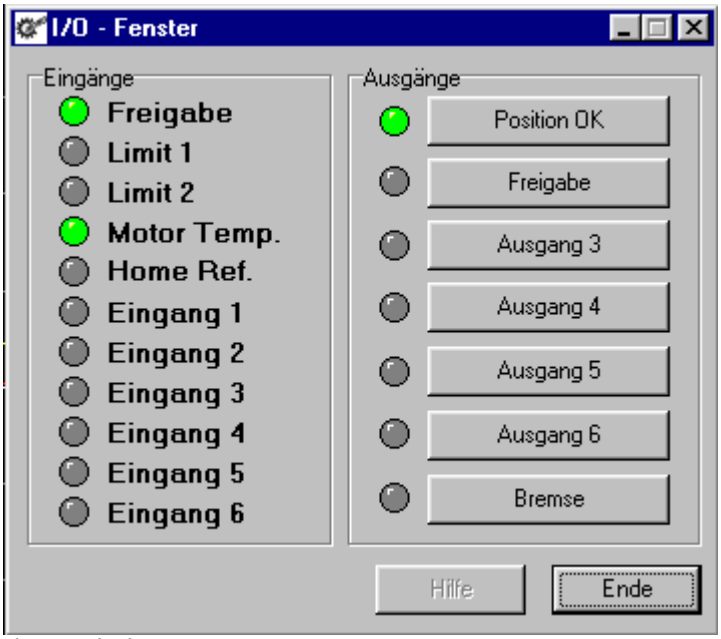
After the Tracer data is read out, it is then displayed by the NORD SERV oscilloscope function. This window can display three value gradients simultaneously. The time scale is the same for all of the gradients, the value scale (Y-axis) on the other hand, can be different for multiple gradients. For this reason, only the value scale for one gradient is displayed on the Y-axis. However, on the right side of the window, the y and x values for all three gradients are shown for the current mouse position. Furthermore, you can switch between which gradient determines the value scale for the Y-axis. Right-click in the oscilloscope window and select the relevant gradient from the menu. The exact function of all the commands in the oscilloscope menu are explained in greater detail in the "Help" command.



2.11 Input and Output Settings

Before starting testing and optimisation, with linear systems, you should check the function of the final position switch.

Use the *"I/O - Window"* to check and set the status of the inputs and outputs on the servo-controller. Call up the window with the *"Device / I/O – Window"* command.



I/O – Window

A green "LED" indicates high level at the respective input or output. The input "LED's" only mirror the status display. Use the assigned buttons to set or reset the outputs. Note, that the outputs can be overlaid with specific functions (enable, position OK etc.). These are then automatically set or reset by the servo-controller software. The settings carried out "by hand" are then overwritten.

2.12 Send and Load Datasets to Servo-controller

Click on the  button to load NORD SERV's current dataset directly into the controller. This is only possible, when the servo-controller is switched off. It is therefore only loaded in the RAM (volatile)! Datasets sent to the device can be saved permanently with *"Device/ Save"*. Note that the PLC program must be saved separately (right-click in PLC window). Be careful with assignments, when working with multiple servo-controllers simultaneously via the bus.

Use the  button at any time, to load the current dataset in the servo-controller into NORD SERV, even with a moving axle.

2.13 Load and Save PC Datasets

Click on the  button, to load a dataset from the PC hard drive or a diskette to NORD SERV. For a more detailed description of this dialog box, please see the Online Help (*"Help"* button) or in the chapter "Guided Set-up".

Click the 

button, to save the current dataset in NORD SERV on to the PC's hard drive. If you want to save the dataset under a different name, click on the *"Database / Save as"* command.

If the dataset is loaded from the dataset into the servo-commander, or changed there, it is located in the RAM. When the current to the servo-controller is switched off, this dataset is lost. For this reason, the dataset in the servo-controller was be saved permanently before switching off. Select the *"Device/Save"* command to do this. Since the servo-controller does not react to any form of communication or input level change, the motor must be switched without current beforehand. Saving takes about 2 seconds.

2.14 Optimising Servo-controller Parameters

The datasets provided by NORD Gear are already optimised for the unencumbered motor moment of inertia for the motor selected. If the motor is connected with the system and the motor axle's mass moment of inertia has been significantly altered, then the speed and positioning controller should be optimised for optimal controller behaviour.
Both control loops are optimised, when the controller is switched on. Begin by optimising the speed controller.

2.14.1 Optimising the Speed Controller

In order to optimise the speed controller, it is recommended that you send a speed setpoint change to the motor axle. Various test functions in the *"Control"* window are intended for this, e.g. *"Reverse Speed"*, these are explained in greater detail in the Online Help or in the chapter *"Controlling a Drive".*

Speed changes are always carried out with the full moment of the set current limit. Speed ramps only work in Position Mode. Set the mechanical limits of your system with the current and moment limits in the *"Limits"* parameter dialog box (under tree diagram/Parameter) beforehand! Otherwise, there is a risk that you will damage your machine by mechanical overloading.

Open the *"Control"*  and *"Speed Regulation"* window (tree diagram/Tuning). The *"Reverse Speed"* test function in the Control window is a very simple option for creating speed test point changes. In this mode, after the time entered in "Pulse Width" has run out, the speed setpoint will be negated. After the controller is switched on, the speed amplitude can be changed with the "smaller" or "larger" buttons.

Firstly, set the *"Boost"* in the *"Speed Regulation"* window as follows:

1. Increase the *"Boost"* in small steps.
2. Listen for the drive oscillating. If this is the case, then continue to the next point, if not, continue to increase the *"Boost"* in small steps.
3. When you reach the oscillation limit, reduce the value deduced by approximately one third.

Next, set the *"I – Share"* as follows:

1. Increase the *"I - Share"* in small steps.
2. Listen for the drive oscillating. If this is the case, then continue to the next point, if not, continue to increase the *"I - Share"* in small steps.
3. When you reach the oscillation limit, reduce the value deduced by approximately one third.

The speed controller has now been set. With the help of the Trace and Oscilloscope functions, the result can now be observed visually and "corrected", if necessary. This is explained further in the "Trace and Oscilloscope function" section.

2.14.2 Optimising the Positioning Controller

Its efficiency can be determined by tracking errors (status window or scope image). Positioning setpoint changes should be carried out to optimise the positioning controller. These can be created by a superior PLC, the internal servo-controller PLC or the *"Control"* window. The *"Simple Move.sps"* program and the internal PLC present a very simple option to have the axle reversed. Operation is described in chapter *"SPS".*
Before starting optimisation, all relevant parameters for the positioning controller must be adapted to your application. In the *"Limits"* dialog box, this affects at least the parameters for acceleration, braking and final speed. Positioning controller tracking errors should be switched off during optimisation, or set very generously, because controllers, which are not optimised, cause very large tracking errors during the acceleration and braking phases.

In the event of a fault, an emergency stop will take place at the servo-controller current limit, (i.e. with full moment.)

The axle tracking error can be optimised, by using the trace and oscilloscope function, operation of this function is explained in the "Trace and Oscilloscope Function" section. In addition, "Tracking Error" and "Moment" must be selected for display in the Tracer.

Steps for optimisation:
1. Start the tracer, the time set should encompass a complete movement cycle.
2. Start the "movement" of the axle
3. After a trace cycle, load the tracer's values onto the PC
4. If the tracking error meets your expectations, then the settings are suitable for your application; go to step 8
5. Increase the positioning controller boost
6. Observe the "moment" If it starts to vibrate, then the "I-Share" in the speed controller must be reduced a little, in this case, step 8
7. Repeat steps 5 to 7, until the tracking error meets your expectations.
   if yes, step 8
8. A small, but enduring tracking error of the motor axle is typical for a pure P-controller (only P-share). This can be avoided by an I-Share. Taking extreme care, increase the "I-Share" in the positioning controller for this. In general, the I-Share should not be used.

2.15 Calibrating New Motors

It is possible for experienced users to incorporate new and even third-party motors in the motor database.

After selecting the motor type, only the following motor data is required.
Synchronous motors: max. speed, peak current, extended standstill current, number of poles, torque constants, EMF constants
Asynchronous motors: max. speed, rated speed, peak current, extended standstill current, power, frequency, number of poles, cos phi,

Firstly, the *"Complex"* option must be selected in the *"Options / NORD SERV – Functionality"* menu. Next, click on the  button, to call up the Open dialog box and load a dataset with the correct controller. You can access the Motor dialog box via the *"Motor"* parameter dialog box in the tree structure. Enter the name of the motor with its parameters in this dialog box. All motors, which have been altered, or a re newly created, have a "U_" before their names. Finally, enter the motor index number. NORD SERV uses this number to recognise the motor type when uploading (loading dataset from servo-controller to NORD SERV). Click on *"Save to Database"* to save the new motor in the PC NORD SERV database and close the dialog box by clicking *"OK"*. The new dataset may possibly also need to be saved permanently in the servo-controller. Finally, at least the moment controllers and possibly the ASM pre-control/model values may also need to be set.
Datasets with self-created asynchronous motors can only be loaded from the servo-controller to PC, if these motors are in the database.

Only the motor index number is stored in the servo-controller. If these numbers are assigned to different motors on different PC's, it is possible that the uploaded dataset might be associated with an incorrect motor and will not function.

2.16 Fault Reports

Should a fault occur in the servo-controller during operation, the operator will be informed by a report box. After confirmation of the NORD SERV report, the fault window will open, in which the causative fault is displayed in a list. Confirm this window with *"OK"*, to return the servo-controller from fault status; it can now be switched on again. To receive a fault description and possible cause, highlight the relevant fault and click on the *"Details"* button. Faults lead to the servo-controller being switched off. The CAN-Open-compatible status machine often causes a Quick Stop. In the case, the bi-coloured LED's on the front display the fault or fault group by flash code (red/green, flashes x-times…pause…flashes x-times…pause etc.)

Flash codes:

Index	Flash Code	Description	Reason
2000h	flashes 1x	Controller immediately switches output stage off, due to overcurrent.	- Motor short-circuit - Motor earth fault - Output stage overtemperature
2001h 2002h	flashes 2x	Controller immediately switches output stage off, due to overcurrent.	- incorrect motor - very poor controller settings
3100h	flashes 3x	One or more mains phases switched off at controller.	- Emergency off activated Supply voltage insufficiently fused - FI circuit breakers in circuit - Faulty power line connection to servo
3210h	flashes 4x	DC link is too high, this fault leads to the motor current being switched off immediately.	- Braking resistor not connected - Braking resistor faulty
3220h	flashes 5x	The DC-link for the servo-controller is below the permissible value	- Incorrect input voltage - An input voltage phase is missing
4300h	flashes 6x	Motor has exceeded category temperature	- Motor temperature sensor not connected - Dirt preventing motor from emitting heat via

			enclosure - Current controller poorly set with motor customer datasets
5400h	flashes 7x	Output stage reports permanent fault	If this fault is displayed, despite repeated switching on, then the SERVO-CONTROLLER is faulty.
5441h	flashes 8x	The servo-controller's Enable input is at Low – level.	The servo-controller can only be switched on by software, if the Enable input has a High – level, for safety.
5442h	flashes 9x	Limit input 1 has a Low – level.	The machine slide has entered Limit 1 or the limit input has not been interconnected or disconnected.
5443h	flashes 10x	Limit input 2 has a Low – level.	The machine slide has entered Limit 2 or the limit input has not been interconnected or disconnected.
6210h to 6271h	flashes 11x	Fault in servo-controller PLC program	For a detailed description, see CAN 5.3. Servo-controller Fault Reports.
6290h	flashes 11x	An incorrect address in the internal PLC program was accessed via index 2C01h.	- There is a different PLC program in the servo-controller The addresses in the PLC program have been changed by inserted or deleted commands
6291h	flashes 11x	The Home command is ignored.	The motor axle was moving, when Home was started
6300h	flashes 11x	A fault occurred via Index 2010 whilst a dataset was being downloaded,	An attempt was made to load a dataset with a different output stage type into the controller.
7310h	flashes 12x	The axle was outside the speed deviation fault window for longer than the set time period.	- Axle blocked by obstruction in the machine or by gear fault, - too small tracking error selected - Encoder incorrectly connected (interconnection) Encoder offset incorrectly selected (interconnection)
8120h	flashes 13x	The controller has no communication with control for the time set in switched on state.	- Safety time period too small - Connector cable faulty or fallen out
8130h	flashes 13x	The servo-controller's CAN driver has gone into "Bus off" state.	- Connector cable unsuitable, faulty or fallen out - Connector cable not properly screened, not proper 120 Ohm terminating resistor - Devices with differing baud rates on bus - strong EMC effects
8410h	flashes 14x	The speed setpoint sent exceeds the upper "Limit" set in the Parameter dialog box.	The encoder is changed, by a different speed constant being produced (NORD SERV menu *"Devices/Constants"*) and so different binary values.
8611h	flashes 15x	The max. tracking error entered in the Parameterise/Limits dialog box (Positioning deviation) has been exceeded.	- Axle blocked by obstruction in the machine or by gear fault, - Axle sluggish, due to insufficient maintenance (lubrication) - Incorrect parameter set or optimisation parameter (often incorrect selection of positioning controller boost or pre-control) - too small tracking error selected
8710h	flashes 16x	Slave has run out of the tracking error window set to the master, during gear function.	- Slave axle blocked by obstruction in the machine or by gear fault, - Slave axle sluggish, due to insufficient maintenance (lubrication) - Incorrect parameter set or optimisation parameter in slave servo-controller (often incorrect selection of positioning controller boost or pre-control) - Faulty electrical connection between both

			servo-controllers. - Monitoring function incorrectly set in master servo-controller (transmission, slave rotation)
8810h	flashes 17x	Flying start active.	Speed limit exceeded during moment mode (also applies for winder and compensator regulation)

2.16.1 Acknowledgement

1. by switching the mains voltage off and on again,
2. by Low-level of the *"Enable"* on the servo-controller,
3. by a bus acknowledgement, e.g. by NORD SERV.

2.16.2 Meaning and History

In accordance with CAN-Open, faults are organised by meaning with their set fault number in the servo-controller and can be read out and displayed by a Bus device or NORD SERV (in plain text). Fault classifications defined and the reaction of the status machine are located in the CiA "Device Profile:Drives and Motion Control DS 402" CAN-Open document.. All faults are collected with time entry and displayed in NORD SERV History.

2.16.3 Plain Text Solution Recommendations

The NORD SERV Help function presents the user with a detailed fault description in plain text. Highlight a fault in this list and possible **Solution Recommendations** will be made by NORD SERV. These explanations are very extensive and, therefore will not be repeated here. Please follow the instructions.

3 Drive PLC

This PLC, which is integrated in the servo-controller, can be used to solve small control tasks and create complex operational profiles, to relieve superior machine controls. You can often do without these and so save money! Not only that, you increase clarity and significantly reduce communication load.
The standard PLC can store up to 200 commands. If you do without the Trace function, which can be activated by NORD (request when ordering), up to 400 lines are possible. The cycle for this PLC is 100µs. The same time period applies for interrupts.

3.1 Start-up

Call up the programming window in NORD SERV with the  button. This window is used to create and display the program and program process.
Right click on the window currently displayed to open a menu to open, save and print PLC programs. You can also separately permanently save the PLC program in the servo-controller and on the PC hard drive. In particular, the PLC program is **not** automatically, when the parameter set is saved. Please be very careful with assignments, when working with multiple servo-controllers simultaneously via the bus. A PLC program can or even cannot then link to multiple servo-controllers.

The following buttons are located on the left side of the programming window and can be used to commission and create programs.

3.2 Program Execution

"Start"
Click this button to configure and switch on the servo-controller of the PLC operating type. The program in the programming window is sent to the servo-controller and automatically started beginning with the first line.
"Stop"
Click this button to end program execution and switch the servo-controller off.
After a program is completed in the programming window, a test should be carried out. For this to happen, the servo-controller connected must have a communication connection with NORD SERV ("online"), i.e. the current dataset must have been transferred from PC to servo-controller or from servo-controller to PC.
Click on the *"Start"* button to send the PLC program to the servo-controller and start it. The program execution is displayed continuously in the programming window, whereby the active program lines are marked respectively. However, the update time of 1 sec for the user interface is too slow to track every program step with fast program steps. It is also possible to use the Tracer in the servo-controller to monitor program execution. Click the *"Stop"* button to interrupt program execution and switch the servo-controller off.

3.3 Servo-controller under immediate PLC control

In order for the internal PLC to be able to take over permanent and immediate control of the servo-controller after switching on, the PLC program must be saved in the servo-controller. Select the *"Save program in Controller"* command (right click), to save the PLC program, which is only temporarily stored in the servo-controller's RAM, permanently in EEPROM. Furthermore, *"SPS"* must be activated in the *"Setpoint Source"* parameter dialog box and also saved permanently in the servo-controller (*"Device/Save"* menu).
After these steps, the PLC program in the servo-controller is active, i.e. enabling the "Enable" input leads to the servo-controller being switched on (motor has a current) and the immediate start of the PLC program.

3.4 Overriding the PLC and Position by Machine Control

One of the NORDAC servo-controller's exceptional functions is being able to start any command line with a command from a superior controller and even change positioning via the bus system. This enables complex processes to be started with a single command directly by the controller, thereby significantly reducing the field bus load. See the "CAN Bus Protocol" and "Setpoints" sections for the exact description of these commands.

3.5 Entry

"New"
Opens the command editor and places commands created at the last point in the program.
"Edit"

Opens the command editor and replaces the highlighted command in the program. Double-clicking on the relevant line will also open the command editor.

"Insert"

Opens the command editor and writes a new command above the highlighted command line.

"Blank Line"

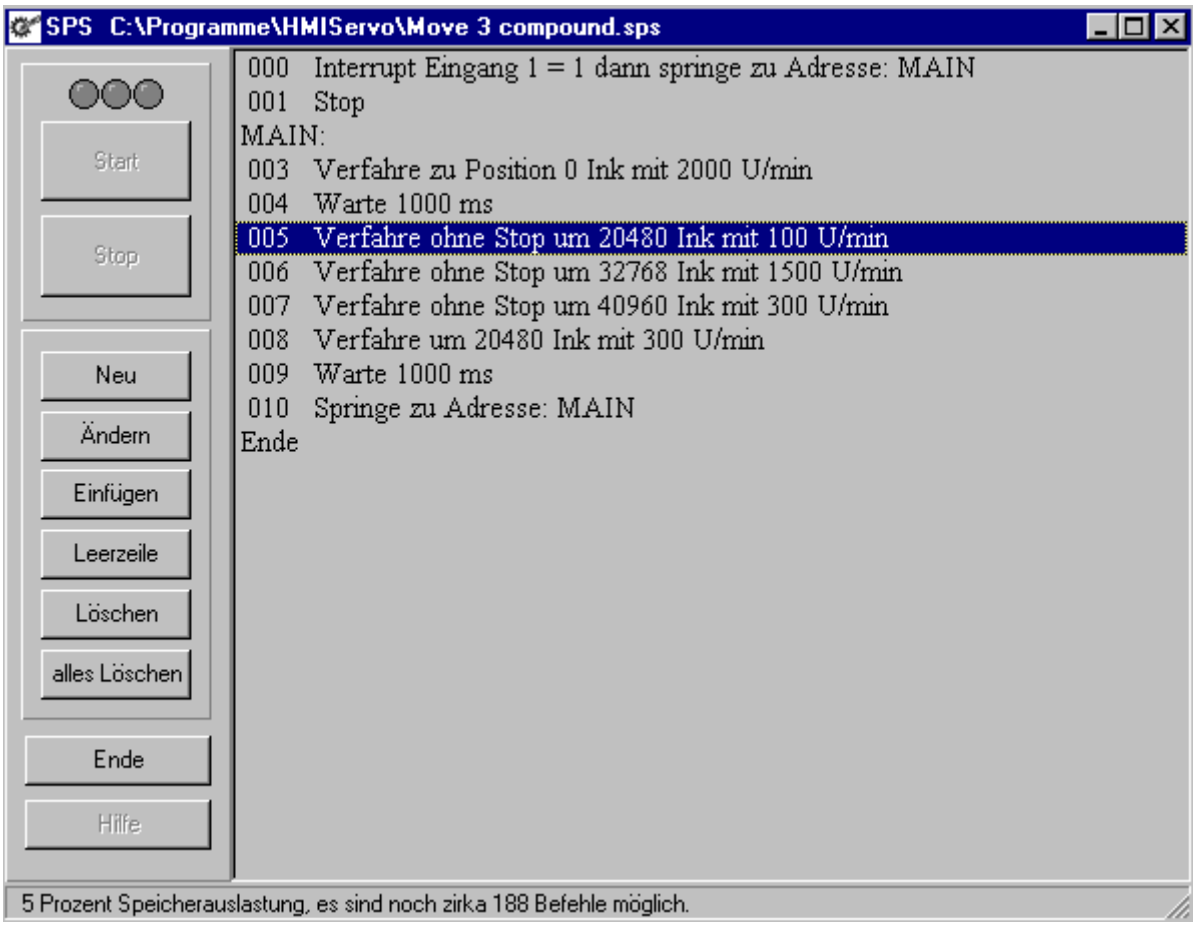
Inserts a blank line above the highlighted command line (provides a better overview).

"Delete"

Deletes the highlighted command line.

"Delete All"

Deletes the entire program.



To create a new program, first open the Programming window. If it contains a program, then click *"Delete All"* to clear. If the Programming window is empty, click *"New"*. Commands can be selected by clicking on the tree diagram in the dialog window, which now opens. Enter the respective parameters for the commands selected in the text box which appears and confirm. The dialog window closes and the command appears in the first line of the Programming window.

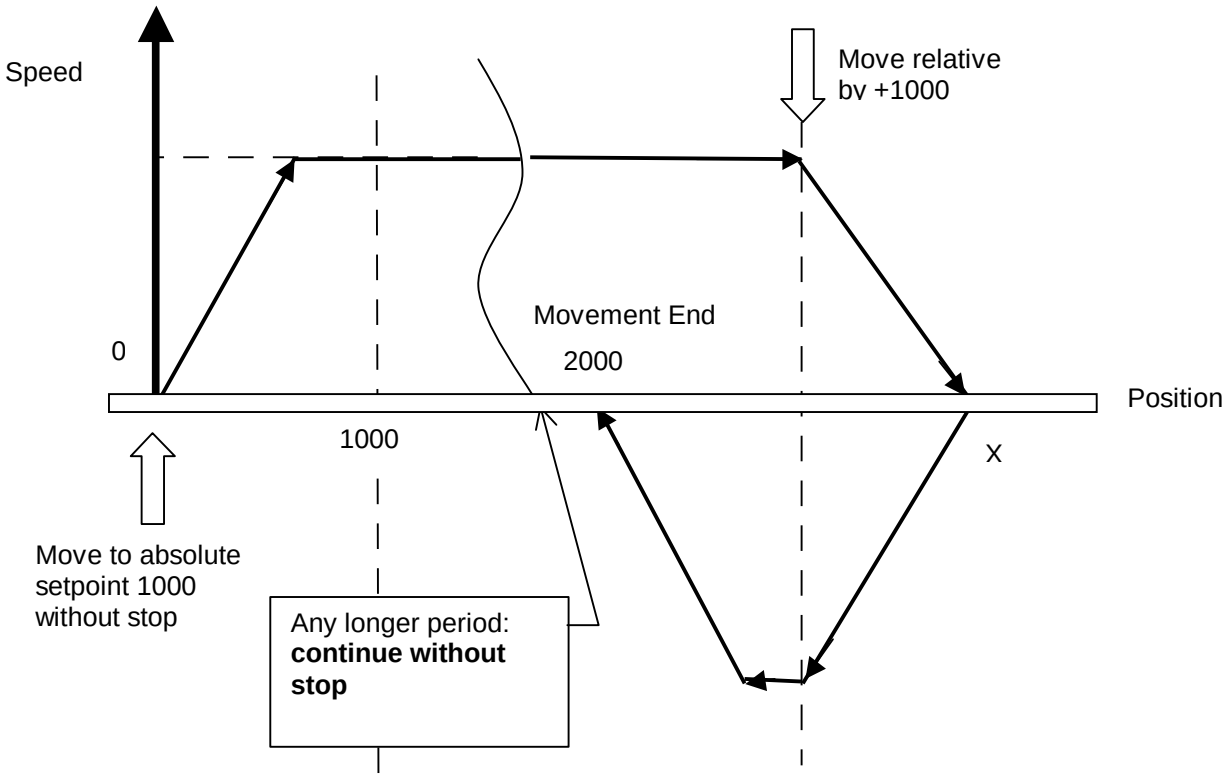
Use the *"New"* and *"Insert"* buttons to expand the program until the memory limit as been reached (see Status Line in Programming window).

3.6 Movement Commands

Incorrect entries can lead to collisions!

Operating commands can be altered and cancelled at any time. This can significantly increase the machine's cycle time. Thus, the axle does not have to carry out the old operating command, so move to a standstill if necessary, but rather, can immediately start at the newly determined position.

Fundamentally, movement commands are based on the last set position. This should be noted in particular for movement without a stop, because a potential subsequent positioning command is related to the long overrun set position. This may mean that the axle is even travelling in the opposite direction – backwards. The advantage of this structure, is that distance (incremental) commands may be linked relative to one another, without increments being lost.

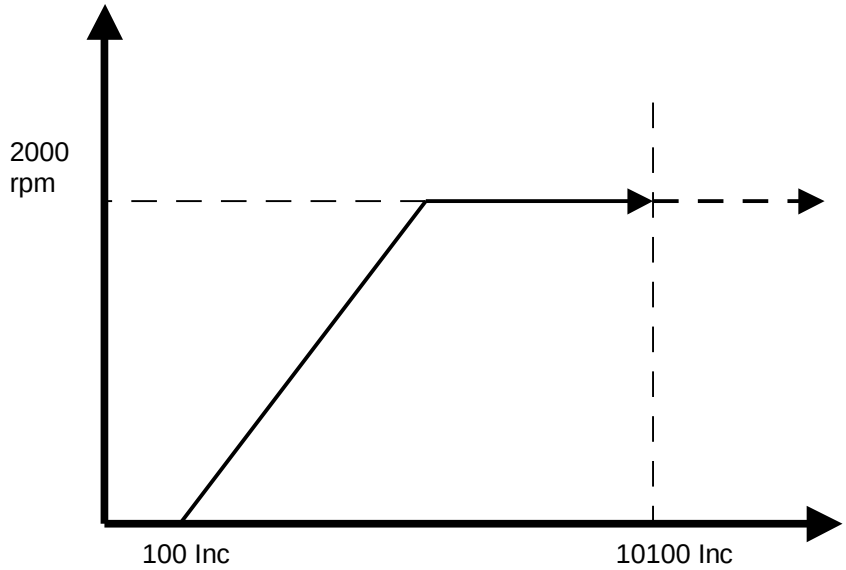


The following applies to all subsequent movement commands: Target position and distances can be set as constants or also via markers. The acceleration and brake ramping parameters are set in the *"Limits"* parameter dialog in the tree diagram. In addition, these parameters can be overwritten by the command *"Set Ramps".* Terminal velocity can be affected by the "Speed Override" function in the *"Setpoint Source"* parameter dialog box. The terminal velocity entered can be evaluated as a percentage from the current at the analogue input (0V = 0%, 10V = 100%).

Move to Position
The axle moves to the absolute position entered and accelerates to the terminal velocity entered in doing so. The following command is only called up after reaching the target position.

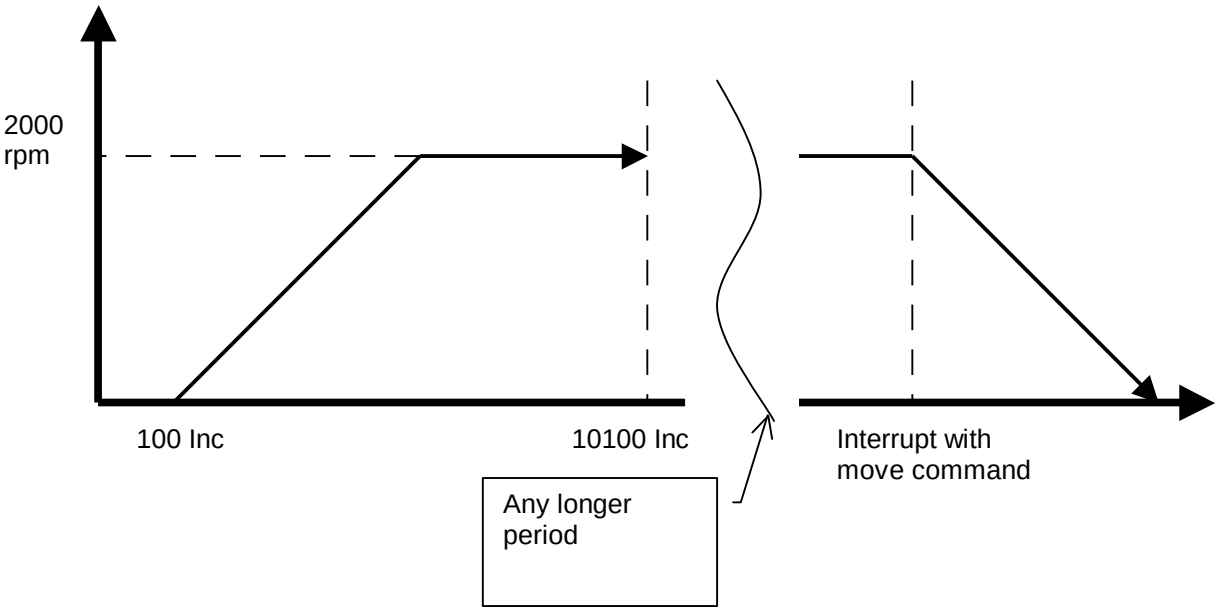
Move Distance
The axle moves the distance entered from a current position, i.e. at a current position of 2000 increments and an movement distance of 1000 increments, the axle moves to a position of 3000 increments. Entering negative distances reverses the movement direction. The following command is only called up after reaching the target position.

Move without stop
The axle brakes or accelerates immediately to the speed entered in the command and moves the distance entered in the command without braking when the new position is reached. The next command is called up after reaching the distance. If no new command follows, then the axle will maintain the last speed entered. You can combine this command with itself or any other movement command, to create complex operating profiles (speed profiles).
Examples:



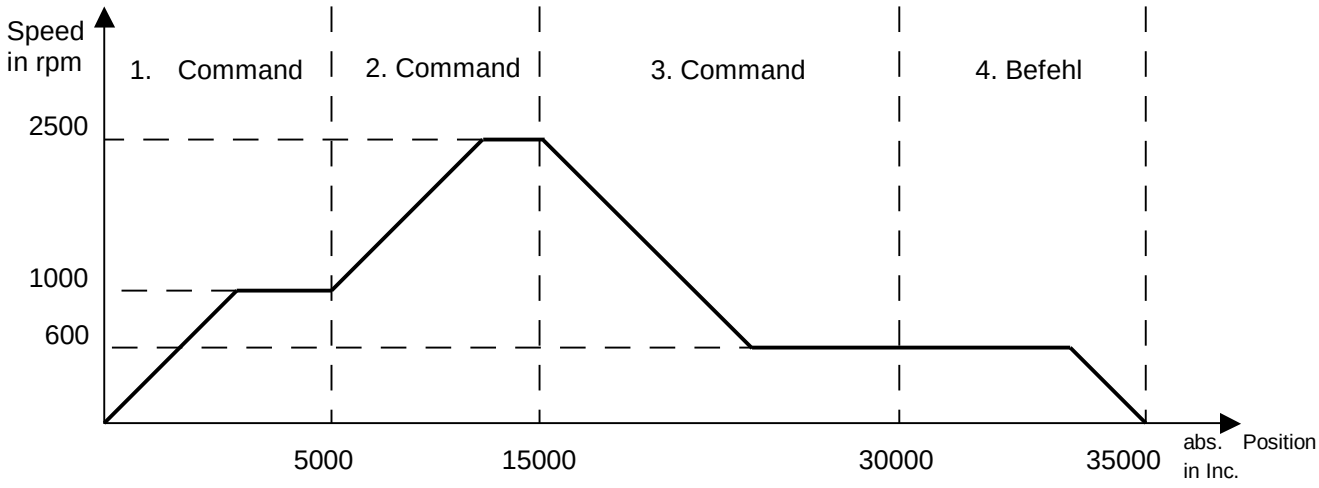
Depiction of *"Move without stop"* command

The image above depicts the command *"Move without stop by 10000 increments at 2000 rpm".* The axle starts with a position of 100 increments, accelerates with the preset ramp to 2000 rpm and moves to position 10100 increments. The command ends at this position and the subsequent command is called up.If no new command follows, then the axle continues at 2000 rpm as indicated by the dotted line.



Depiction of *"Move without stop"* command in combination with an interrupt

The above image again uses the *"Move without stop"* command. After the command is complete, the axle moves at 2000 rpm in a positive direction, until it receives a new command. So, for example, an interrupt can be activated by a freely-programmable input, which can bring the axle with a movement command to a standstill.



1. Command : Move without stop by 5000 Inc at 1000 rpm
2. Command : Move without stop by 10000 Inc at 2500 rpm
3. Command : Move without stop by 15000 Inc at 600 rpm
4. Command : Move by 5000 Inc at 600 rpm

Chaining of *"Move without stop"* command into a complicated velocity profile.

Chaining multiple *"Move without stop"* commands enables the creation of complex profiles in any form. With the 1st command, the axle moves 5000 increments. Immediately afterwards, the 2nd command starts and the axle accelerates up to 2500 rpm and the 3rd command begins at 15000 increments. With both acceleration and braking, the preset terminal velocity is achieved immediately after the command. With the 4th command, the axle moves onwards 5000 increments and remains at the position of 35000 increments.
You can chain as many *"Move without stop"* commands together as you like. The only limit is the total memory capacity.

Note:
If the 3rd command (see previous image) is started at 1000 increments, rather than 15000 increments, then the braking distance is shorter. The servo-controller thereby does not achieve the speed set in the 3rd command. The 4th command does not move to the target position at 600 rpm, but rather at the speed reached by the servo-controller, once it reached the set position for the 3rd command (16000 Inc).

Stop with Ramp
At this command, the controller brakes immediately with the last ramp set from the current speed to 0 rpm. This command may be called up at any time and interrupt other movement commands.

Home
This command starts a reference run for the servo-controller. The type and parameters of the reference run are set in the tree diagram under the *"Home"* parameter box, the various Home methods are based upon those recommended by CANopen. These are described in greater detail in the local Help.
After the reference run, the ramps are set to the values entered in the *"Limits"* parameter dialog box! If the ramps are changed by the *"Set Ramps"* beforehand, then this setting is overwritten by the *"Home"* command. For this reason, the ramps in the PLC program should always only be changed after the *"Home"* command.

3.7 Jump Commands

Goto
The program jumps to the address entered and continues running the program.

If (Input X = ...)
This command tests the logical level (High, Low) at the selected input. If the condition set is "true", then the program jumps to the address entered and continues running the program. If the condition is not true, then the next command will be executed.

If (Input XXX = ...)
This command tests the logical level (High, Low) at multiple selected inputs. If the logic at the inputs agrees with the template set, then the program jumps to the address entered and continues running the program. If the condition is not "true", then the next command is executed.

Please note that the inputs are at a steady state at the time of the query! You can check for a "validity signal", e.g. with a prior command. Otherwise, the drive may react unexpectedly.

If (Marker = ...)
This command carries out a comparison between markers and or a constant. If the condition is "true", then the program jumps to the address entered and continues running the program. If the condition is not "true", then the next command is executed.

If (Marker = Template)
This command enables individual bits and bit combinations to be tested. If the bit templates and markers agree, then the program jumps to the address entered and continues running the program. If the condition is not "true", then the next command is executed.
The symbols "0", "1" and "X" are used to set the template. The symbols "0" and "1" represent the logical values "Low" and "High" from digital technology. The symbol "X" is used when a bit in the evaluation is to be ignored.
The main application of this command is for evaluation of binary states, such as I/O's, which are imported to the servo-controller via the CAN field bus.

Interrupt Input
If the level change selected occurs at the input set, then the program continues running at the address entered. In contrast to the *"If (Input X = ...)"* command, an interrupt is always active after the first cycle (initialisation). If a level change takes place at the input, then the program continues running at the address entered, irrespective of which command is currently being carried out. An interrupt requirement will disrupt all running requirements, i.e. Wait or movement commands can also be interrupted.
Interrupts only become active after their initialisation, i.e. it makes sense to carry this out at the beginning of the program, or at a suitable place.
Reaction time to an interrupt is less than 350µs.
A maximum of 5 interrupts can be activated.
Interrupt parameters (e.g. jump address, input or edge) can be changed at any time whilst a program is running.
If multiple interrupts are initiated simultaneously, only the one with the highest priority will be carried out! All other requirements will be ignored **and** reset and will only be initiated when a new level change occurs. The interrupt input "In 1" has the highest priority an "In 5" the lowest priority. In 6 is not supported.
It is possible to use the same input for an *Interrupt* and an *"If (Input X = ...)"* command.

Interrupt Position
This command enables you to react to set positions being under or overshot, thereby enabling you to create very variable software final position switches. Two different positions can be monitored, these are set in the *"Interrupt"* box in the command window. Interrupts only become active after their initialisation, i.e. it makes sense to carry this out at the beginning of the program, or at a suitable place. However, this interrupt should be initialised last in the interrupt block. If the current position is outside the set positions when switched on, the interrupt will immediately jump to the set address, and following commands may not be carried out.
Reaction time to an interrupt is less than 250µs.
Interrupt parameters (e.g. jump address, input or edge) can be changed at any time whilst a program is running. This can be done with a new program line or markers.
By contrast with the *"Interrupt Input"* command, this interrupt is deactivated after its initiation, i.e. it must be reinitialised after the application-dependent commands are carried out (e.g. stop axle and move out of limit area).
If the interrupt position is provided by a marker, this marker must always be given in increments.

3.8 Program Process Commands

Address
Enables entry of jump addresses. Addresses are required by numerous jump commands. All symbols, including spaces, can be used. If a program with preset addresses is loaded from the servo-controller to the PC, the address names are automatically numbered by NORD SERV, because they are stored in the servo-controller in an abbreviated form. For this reason, the name length has no impact on limited storage capacity. The self-selected names only remain, when loaded from the PC's hard drive.

Comment
Enables entry of comment lines for explanation of program segments and provides a better overview. All symbols, including spaces, can be used. Comments are only stored in the PC and not on the servo-controller. (For this reason, reloading a program from the servo-controller to the PC is not possible). Since they do not occupy storage capacity in the servo-controller, the number of comment lines has no impact on possible number of commands.

Stop
This command stops program execution. The program can be restarted by an interrupt or a command via the bus from a superior controller.

3.9 Function Commands

Set Marker
This command describes the content of the marker set. The respective marker can be written with a constant in the 32Bit area, or with the servo-controller's current values.
The current position is imported in the marker in increments.
The current moment and speed are imported **without units**. For comparison operations in Nm or rpm, the constants to be compare must be converted to the controller unit. This is possible via a constant, which can be read out in the "Devices/Constants" menu in NORD SERV. The constant is dependent upon the controller and motor type used.
The second position counter only functions if an incremental encoder is connected to the "2nd Encoder In" input. The position is imported in the marker in increments.
The Bit status value contains the following information.

Bit	Meaning
Bit 0	0 = gear function is out or controller is still in flying saw function
	1 = gear synchronisation is active
Bit 1	0 = controller is not in position window
	1 = controller is in target position (only valid in position mode)

Calculate
Markers and constants can be linked together in this dialog box via the four basic calculation methods and binary functions. The calculation result is always stored in a marker.

The calculation operation must remain in the 32Bit area, otherwise the result will have a number overrun. For example: Multiplication can execute 32Bit * 32Bit, but the result is also just 32Bit! Division is defined as an integer (32Bit / 16Bit =16Bit), the result is whole-numbered, i.e. remainders or decimal places are not evaluated.

Set Output
The output entered is set High or Low. Note that only outputs without special functions (e.g. Enable, Position OK etc.) can be set. This function is set in the *"Outputs"* parameter dialog box.
Furthermore, note that it is not possible to set an out put High in one command and Low in the next. In order to switch the level of an output, a *"Wait"* command with a minimum of 1ms must be located between the two *"Set Output"* commands.

Wait
The program flow is halted for the time period entered and then starts the subsequent command afterwards. The time set can lie between 1 ms and 26 s. Unlimited wait commands may be strung together.

Report Position
If the position set in this command is overrun, then the output selected is set to High level. Outside this position, the level of the output is low. The active position is generated by the set position ± the subsequent position window. If, for example, a value of 1000 increments is entered, the selected output has a high level in the area from position -1000Inc to position +1000Inc.
This command must be declared at the start of the program and is active throughout the program duration. The parameters of the command can be changed at any time. It is possible to monitor two different positions. Differentiate in the *"Report Point"* group window.

When setting the outputs, note the description in the "Set Output" command. This applies in particular to the size of the position window, because there is insufficient time to set outputs, if movement speed is high and the window small.

Block Interrupt
The interrupt sources specified in this command can thereby be deactivated again. Of course, subsequently the interrupt sources can also be reactivated.
Note: Naturally, this takes a PLC cycle to become operative!

Zero Adjustment

Execute this command at any time to carry out a zero adjustment on the analogue input. During the execution period, there must of course be zero volts at the analogue input as a zero reference.As required, any other value can be set as base zero.
Note: During the adjustment, the input signal must remain stable. Due to filter switching, the set-up process can take a few ms.

Set Position
When executed, the current position of the second position counter will be set to the value entered in this command. The second position counter only functions when an incremental encoder is connected to the "2nd Encoder In" input.

Initialise Outputs
If the servo-controller is switched off (by a fault event, release removal etc.), then this command can be used to define the level of the outputs in switched-off state. If this is not done, then the last initial status is retained, since the internal PLC only functions in a switched on state.

Servo Out
This command immediately switched the motor without current and ends the PLC program.

3.10 Technology Function Commands

Gears
This command switches the gear function on or off. The parameters for the gear function are set in the *"Gears"* parameter dialog box. This command may not be executed during movement or home commands or servo motor operation. Furthermore, once the gear function has been switched on, this command cannot be executed again.
The "Quickstart via 'Home ref' Input" option enables the function to be started automatically via a positive flank on the "Home ref" Input. The input's reaction time is less than 50µs. The PLC remains at this command until the function is released by a positive flank.

Servo Motor
The servo motor interface is switched on and off with this command. Increment entry must be in the region between 1 and 32767 increments. The increments may be changed at any time during operation.
This command may not be executed during movement or home commands or gear mode.

Winder
This command switches the rewinder on and off.
Parameterisation is done in the *"Winder"* parameter dialog box. When the function is switched on, all subsequent movement commands are ignored.

Compensator
This command switches the compensator (pressure load cell) on and off.
Parameterisation is done in the "Winder" parameter dialog box. When the function is switched on, all subsequent movement commands are ignored.

Disk Cam
This command switches the disk cam and camshaft functions on and off. The camshaft function can only ever be switched on and off together with the disk cam function.
Select the disk required from the "Disk" box. You can enter a start off set with a marker. Select this from the "Start off set in Inc" box. This offset is related to the master encoder's increments.
In the "Disc cycle" box, you can select between the options "once" and "continuous".
With the "once" option, the servo-controller only follows the master axle with a single 360° rotation. The PLC remains on this command until the disk has been cycled through once. Therefore, chaining of disks or execution of a set number of disks is possible.
In "continuous" mode, the servo follows the master axle for an unlimited period and the PLC does not remain on this command, but rather continues with the execution of the following command.
Further movement commands may not be carried out during an active disk cam function.
You can alter a currently active disk and camshaft at any time with this command.
The camshaft can only be started, stopped and changed in conjunction with a disk cam. The camshaft required can be set and switched off with the "Camshaft" box.

Offset for disk cam
With this command, an offset to the master axle can be set for an active disk and camshaft. Beforehand, the offset must be entered in a marker. This command is only effective when a disk cam function is active.

Teach In
This command stores current positions permanently in the servo-controller.
This command must always be executed in the first program line.
Enter the number of positions to be saved in "Number of Positions".
Use the "On/Off" input to switch the function on and off. When the function is switched on, the internal counter is reset to zero, i.e. the first position is save to address zero. Furthermore, the "Confirm" input is activated and can be clicked at any time. When the function is switched off, so is the servo-controller and the positions saved are permanently saved with the current PLC controller program. This process takes approximately 5 seconds and no communication with the device is possible during this period.
The "confirm" input imports the current position for increasing flanks. Ensure that there is no chatter at the input.
If the positions are permanently stored in the controller, then a new program can be saved in the device at any time. The only condition is that the "Teach In" command is present in the first line as a memory space retainer. The "Function Out" setting under "In/Out" should be set, so that none of the inputs are blocked.

Readout Teach Position
With this command, positions saved in the servo-controller can be read out. For this to happen, the addresses required must be in "Save address" in selected markers. The first value saved has address 0. The read out position is stored in the marker selected in "Value".
In order for this function to work, the "Teach In" command must be present in the first line as a space retainer (memory reservation).

3.11 Parameter Commands

Set Regulation
Use this command to switch between moment, speed and position regulation. Switching is possible at any time. Only active technology functions must be switched off prior to switching to moment or speed regulation.
Two types of setpoint creation are possible in moment and speed mode. Via the analogue ±10V setpoint interface or via the Bus. In analogue mode, the limited moment or speed in the Limit dialog box will be reached at 10V. Reduction of the maximum moment is possible at any time with the "Set Moment" PLC command.

Set Moment
This command enables you to change the maximum possible motor moment. It should however be noted that, with a smaller moment, the servo-controller tracking error increases and the motor speed achievable reduces.

If speed and tracking error in the *"Limits"* parameter dialog box is switched off and the axle is blocked, then the servo-controller will attempt to make up the lost distance/time period when the axle is released. In doing so, the servo-controller will accelerate the motor above the speed limits entered in the *"Limits"* parameter dialog box.

Set Ramps
Use this command to set the acceleration and braking ramps for all subsequent movement commands. The entry format is [rpm/sec], i.e. how many rpm should the axle accelerate or brake per second. The setting applies until the next change or until the servo-controller processor is without current or a Homing Run is executed.

Set Gears
This command enables you to alter the gear parameters in the PLC program. It is possible to alter the gear slave's transmission and rotation.

This command should not be started, if the gear function is active. This is because parameter changes will always lead to a change in speed, with the full moment of the motor. A change during active movement commands is always possible.

3.12 CANopen – Commands

Send PDO
Enables PDO's to be sent via the PLC. General PDO settings take place via the *"CANopen"* parameter dialog box and cannot be influenced by the PLC.
The PLC markers can be sent and received via the CAN-Bus over index numbers 2C20h to 2C27h.
These index numbers can be set in the CANopen parameter dialog box.

Send NMT
Enables bus commands to be sent. These commands are required, if the servo-controller is to operate as the master on the CAN – Bus. The "Start Remote Node" command releases the transfer of PDO's on the bus and the command "Enter Pre -.Operation" stops them again.

3.13 Examples

The following list of example programs are located in the Installation directory on your PC, after installation of NORD SERV.

3.13.1 Move between two points

Simple Move.sps

No.	Command line	Command
0	Start:	Program process / address
1	If Input 1 = 0 then jump to address: Start	Jump command / If (Input X = ...)
2	Move to position 10000 Inc at 500 rpm	Movement command / Move to position
3	Wait 1000 ms	Functions / wait
4	Move to position -5000 Inc at 1000 rpm	Movement command / Move to position
5	Wait 1000 ms	Functions / wait
6	Jump to address: Start	Jump command / Goto

With this program, the axle moves cyclically to and fro between the positions –5000 and 10000 increments, provided that In1 at the servo-controller has a HIGH level. If the level at In1 input at program start is LOW, then the axle will only move, once the level changes to HIGH. Should the level change from HIGH to LOW during operation, then the axle will remain at position –5000 increments.

3.13.2 Controlling the servo-controller via inputs

Interrupt.sps

No.	Command line	Command
0	Homing Run	Movement command / Home
1	Interrupt Input 1 = 1 then jump to address: High	Jump command / Interrupt
2	Interrupt Input 2 = 1 then jump to address: Low	Jump command / Interrupt
3	Stop	Program process / Program stop
5	High:	Program process / address
6	Move to position 20000 Inc at 1000 rpm	Movement command / Move to position
7	Stop	Program process / Program stop
9	Low:	Program process / address
10	Move to position 0 Inc at 100 rpm	Movement command / Move to position
11	Stop	Program process / Program stop

This program reacts with movement actions to level changes at inputsIn1 and In2. A Homing Run is activated in line 0, the *"Home"* parameters are set under *"Parameter / Home".* Two interrupts are initialised in lines 1 and 2, the PLC will only react to the inputs after these commands. The program process ends in line 4 with a *Stop command.* The program can now only be reactivated in the event of the predefined level change at In1 and In2. After the movement command, the PLC loads a *Stop command* and ends the program process.

Note: If an Interrupt occurs during the program process, then the program execution is immediately interrupted and jumps to the new address. In this concrete example, an active movement command would not move to its final position, but rather immediately move to its new target position!

3.13.3 Creating compound operating profiles
Complex Move.sps

No.	Command line	Command
0	Start:	Program process / address
1	Move without Stop 100 cm at 2000 rpm	Movement command / Move Without Stop
2	Move without Stop 20 cm at 100 rpm	Movement command / Move Without Stop
3	Move 5 cm at 100 rpm	Movement command / Move distance
4	Wait 1000 ms	Functions / wait
5	Jump to address: Start	Jump command / Goto

This program moves continuously with a complex operating profile in a positive direction. To provide a clearer overview, the positions in increments in this program were replaced by the user-defined unit "cm", this is possible in the *"Positioning Controller"* parameter dialog. In the first cycle (line 2), the motor accelerates to 2000 rpm and moves 100 cm. Afterwards, the command in line 3 starts. The motor moves at 2000 rpm, until the bake ramp is started at 100 rpm. The brake ramp is started, so that the new speed of 100 rpm is achieved exactly at the end of 20 cm (for more information on this subject, please read the command description.) The command in line 4 moves a further 5 cm at 100 rpm, before the axle stops. The axle has now moved a total of 125 cm in a positive direction. After a wait of one second, the process is repeated, i.e. the axle moves continuously in distance segments of 125 cm in a positive direction).

3.13.4 Switching the gear function on and off
Gearing.sps

No.	Command line	Command
0	Interrupt Input 1 = 1 then jump to address: On	Jump command / Interrupt
1	Interrupt Input 2 = 1 then jump to address: Off	Jump command / Interrupt
2	Stop	Program process / Program stop
4	On:	Program process / address
5	Gear function on	Technology functions / Gears
6	Stop	Program process / Program stop
8	Off:	Program process / address
9	Gear function off	Technology functions / Gears
10	Stop	Program process / Program stop

This program enables the axle's electronic gear function to be switched on and off. The gear parameters are set in *"Parameter / Gears".*
The inputs are initialised in lines 1 an 2. Should a level change from LOW to HIGH occur at In1, then it jumps to the *"On"* address and the gear function is switched on. The axle now follows the incoming impulses at the second encoder. Program execution is ended in the following lines, this can only be restarted by an interrupt. However, further commands can follow, e.g. set outputs, If queries etc., only movement commands may not be used during active gear mode.
The gear function is switched off via In2 and line s7, 8 and 9. Movement commands are now possible again.

4 Technology Functions

In addition to the PLC, an number of standard technology functions are available in servo-controllers with option T. Standard models and Technology models only differ in the expanded software functions in the firmware, both are identical on the hardware side. The NORD SERV operating and programming program is the same for all. Technology functions are controlled by the PLC (switched on/off and combined together). A **Document : Application** is available from NORD. This contains the available technology functions with images and detailed descriptions of their respective applications. Some typical technology functions are listed here.

Time saving! **Customers can quickly solve their applications with technology functions and simple parameterisation! NORD has already done the programming.**

4.1 Flying saw

Continuous materials often need to be cut to particular lengths. For this a saw-slave axle is synchronised with running material movement. In doing so, any trim marks or lengths can be run up with the constant movement. Defined offsets can be set. Speed changes to the constant movement can be adjusted automatically. In a synchronised state, the slave axle follows the movement with angular synchronism. This function is also used for synchronous material transport or filling, where multiple machines are linked with a material influx.

4.2 Stepper Motor Interface

This is one of the simplest ways of transmitting positioning setpoints with minimal PLC to the servo-controller at the Pulse & Direction input. The servo-controller follows the number of pre-set impulses, rotation is also pre-set by the direction signal. This function enables stepper motors to be replaced by synchronous motors, without control changes. This provides the following benefits:

- all steps are made up again after the axle is blocked,
- greater motor power may be used without problem.

4.3 Indexer

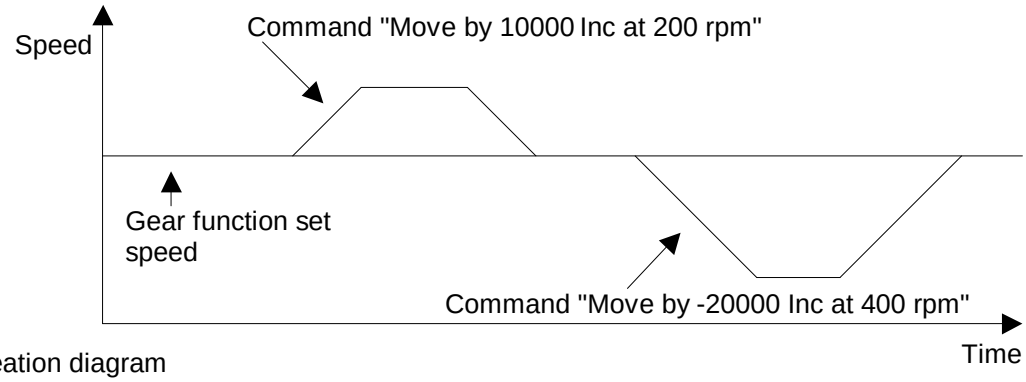
The index function can be programmed very clearly, using the inputs and outputs and the integrated PLC. Components can be measured during handling. You can react to switches and reports.

4.4 Electric Shaft / Gears, Portal

This is the electronic linkage of two or more servo-controllers via a freely programmable transmission ratio. Slave/ Master= 32767 / 1 to 1 / 32767 to angular synchronism processes. Due to a sophisticated mathematic process, rounding errors do not occur over time, nor do creeping increment losses, or angular synchronism loss, even when handling fractions in the transmission ratio.
With portal applications, where absolute angular synchronism of axle must be guaranteed without fault, the gear monitoring function must be switched on. This checks whether the slave axle is following the master axle in a freely-definable pre-set position window. If the difference is greater than set, then an emergency stop will take place in the master axle, the master axle will brake to zero and be switched off. Because the master axle is then no longer emitting any increments (setpoints), the slave axle will also stop. Further fault handling can take place via superior controls.

4.4.1 Printer's Imprint Function (Gears with Offset)

With active gear functions, the slave drive can be given a position offset. The offset can be set for spinning and standing motors.
The position offset is entered via the PLC command "Move Distance". When doing so, ensure that the active offset movement command has ended its movement, before a new offset is started. If a new offset command is entered during an offset command then it is ignored.
The resultant movement speed is derived from the slave speed preset for the gear mode **plus/minus** the speed set in the "Move Distance" command, see image below.



Speed creation diagram

Offset speeds, which are too high, are limited on execution. Later acceleration of the master can, however, enable the slave speed to achieve the limit speed defined in "Limits", a tracking error might then build up, which may cause a fault to occur.
Active offset commands are interrupted immediately when switched off, the axle stops with the active ramp.

4.4.2 Variable Gear

The transmission ratio may be altered at any time via the field bus or even the PLC command "Set Gears" (under Parameters). Furthermore, the transmission ratio may be connected to the analogue input via the "Set Gears" command. An additional parameter determines the region as a percentage, by which the transmission ratio can be changed.
If a value of 10% is entered, then –10V...0V...+10V would represent -10%...0%...+10%, for example. With a transmission ratio of 1000:1000 (i =1) and a constant master speed of 1000 rpm, the slave may be adjusted by ±100 rpm. An application example: Regulation of film thickness between two electronically-linked waves.

4.5 Winder

The winder function is utilised in applications, where continuous material must be wound. The traction force on the material for the winding process may be set, i.e. traction may remain constant, or reduce with increasing diameter. No moment jumps occur on the winding axle during the winding process.

4.6 Compensator

The compensator function allows material to be wound and unwound. Tension is regulated by a pressure load cell or a mechanical compensator. Traction remains constant during the regulation process.

4.7 Electrical Disk Cam and Camshaft

This function enables mechanical disk cams and cam shafts to be realised electronically. Use of this function leads to smaller machines, smaller set-up times, minimised maintenance costs and thereby to lower costs in direct comparison with mechanical solutions.
It is possible to create **location** or time **controlled** movement profiles. For location-controlled profiles, the main axle speed can be recorded by an incremental encoder (100Inc – 32767Inc resolution) or by other devices via the CAN-field bus. Time-controlled movement is provided by a freely-programmable time encoder (50ms – 32sec) in the servo-controller.
In total, up to 20 different disk cams and camshafts can be stored, power failure secure, in the servo. Te number of support points is limited to 2000 points. There is linear interpolation between the support points in the device, so that no jump occurs in the disk progression, even with a small amount of support points.
The camshaft allows defined switching of the 7 outputs, dependent on the main axle position of the time encoder.
Switching the disk on and off, and switching between disks during operation, takes place via the servo PLC.

4.8 Teach In

This function enables current positions to be stored in the servo-controller. This enables a number of target positions to be entered very quickly, without calculation. This function is called up via the servo PLC, with the following command groups:

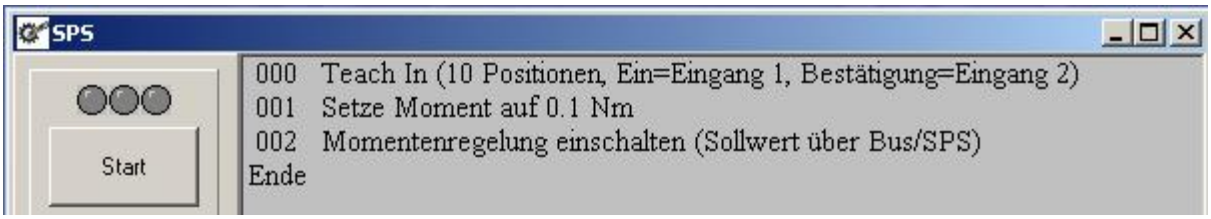
- Teach In,
- Readout Teach In,
- Move to Position.

Important
The "Teach In" command must be present in the first program line of all program lines with Teach In functionality.

The following example programs should be used for general usage. They are located on the NORD SERV CD.

4.8.1 Move to Positions

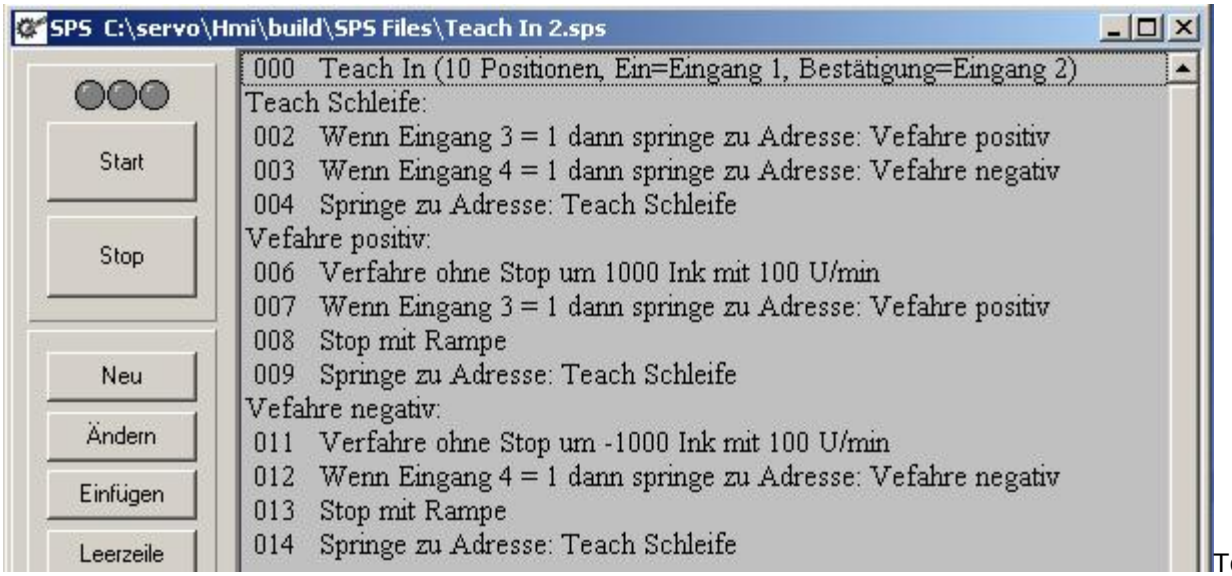
There are two different methods for moving the motor to the required position.
1. Method:
The simplest way is to move the motor axle by hand, the following program example demonstrates the implementation of this in the servo PLC.



Teach In by hand

The "Teach In" function is declared in line 0, in this example, a maximum of 10 positions can be saved. In lines 2 and 3, the motor moment is limited to a minimum and the controller is set to moment regulation. The axle is now powerless and can be moved by hand.

2. Method:
As an alternative to the previous method, the servo-controller can also move to the required position, controlled via its inputs. A program example can be seen in the image below.



Teach In by motor control

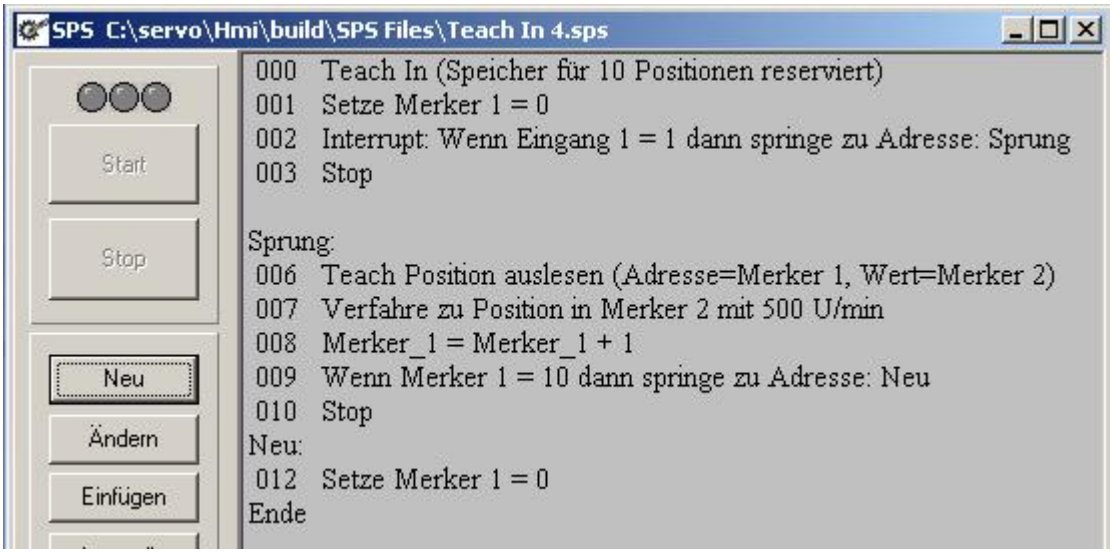
The "Teach In" function is declared in line 0, here a maximum of 10 positions can be saved.
In the subsequent lines, motor movement is programmed via inputs 3 and 4. The speed set in lines 6 and 11 can be changed or generally controlled via the "Override" input ("Setpoint Source" parameter dialog box).

4.8.2 Save Positions Permanently

This requires the "Teach In" function to be switched on. In the program above, this happens via input 1. Activation of this function will reset an internal counter to zero, so that the first position confirmed is located at address zero.
Confirmation of the position takes place via input 2 by a Low-High flank. Ensure you avoid chatter at the switch.
The positions now confirmed are located firstly in the servo's RAM and must now be saved permanently. You can do this in the NORDSERV user interface, by right-clicking and selecting the "Save program to controller" command. A second option is to switch off the "Teach In" function via the respective input (in above example, set input 1 to Low level). The servo-controller is then switched off and will save the positions and current active program. This process takes approximately 5 seconds and no communication with the controller is possible during this period. This method is suitable for applications, where no laptop is available.
A new program can be loaded into the servo-controller at any time, without the saved positions being overwritten.

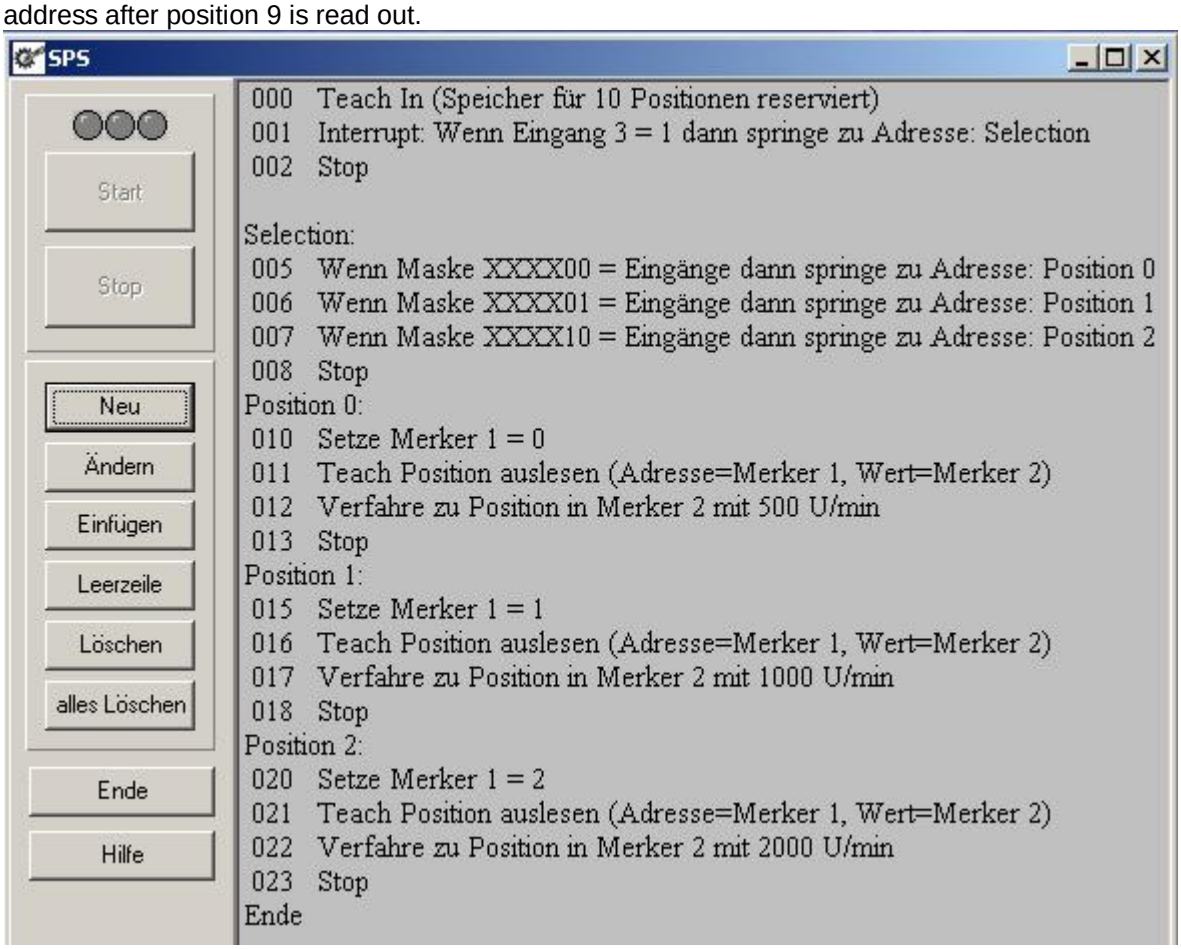
4.8.3 Leaving Saved Positions

The positions can be read out with the "Readout Teach Position" command. This requires the address of the positions to be read out to be entered in a marker. The position is then read out to another marker and can be moved to directly with the command "Move to Position".
This is demonstrated in the following two examples, which can be transferred from the NORD SERV CD.



Successive readout of positions

The "Teach In" command is entered in line 0. The function is switched off and serves as a space retainer of the required memory area.
In this example, a new position is read out for each increasing flank from input 1. Therefore, the position is increased by 1 for each addressing marker after each readout. The address marker is reset to the start address after position 9 is read out.



Readout of positions in Index mode

The "Teach In" command is entered in line 0. The function is switched off and serves as a space retainer of the required memory area.
If increasing flanks are registered at input 3, then the program jumps to line 5 and evaluates the three preset templates for the inputs. If they agree, then program execution will continue at one of the following three addresses. The address required is loaded into marker 1, to readout the position. The first position always begins with address 0. In the subsequent command "Readout Teach Position", the position in marker 2 is read and moved by the following movement command.

5 CAN - Bus - Protocol

A basic condition for operating multiple servo-controllers from one bus is, that all the servo-controllers have different addresses (DIP switch on servo-controller) and the same baud rate is set (NORD SERV : "CANopen“ parameter dialog box). Furthermore, the load resistor integrated at the Bus end of the servo-controller must be hooked up (see CAN interface section).

The CAN interface can simultaneously apply parameterisation and or control of the servo-controller with the RS232 or RS485 interface.

5.1 Controlling the servo-controller without Bus protocol

It is possible to control the servo-controller in a very simple manner, without complicated protocols or integrating it in existing protocols. Thus, up to 8 setpoints can be sent to the servo-controller and up to 8 current values received from the servo-controller. The setpoints and current values are transported by PDO's (Process Data Objects). The servo-controller has 2 send PDO's and 2 receive PDO's. Each PDO is 8 bytes large and can record a max. of 4 different values.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Available	Available	Available
00	10h	20h	30h	07	00	00	00
Set – Point = 30201000h				Outputs = 07h			

Example of a servo-controller PDO

In the example above, 2 setpoints are sent to the servo-controller. All values are transferred in Intel number format, i.e. for values larger than 8 Bits, the Low Word and then the High Word are sent. The sequence of the Bytes is displayed in the second line of the example. The set position is transferred with 32 Bits in the first four bytes of the PDO. The 8 Bit large setpoint to set outputs is sent in Byte 5. The remaining 3 bytes are unused and are not sent.
Use the "CANopen" parameter dialog box to set the number and sequence of values in the PDO. There is an explanation of the individual setpoints and current values in the CANopen– protocol section.

In this manner, one controller can position multiple servo-controllers and e.g. set outputs on the servo-controllers in parallel. Importing the send PDO's enables you to monitor the servo-controller (e.g. has it reached the target position).

The servo-controller receive PDO's process incoming messages immediately. In doing so the values are processed in sequence from Byte 1 to Byte 8.

An explanation of the most important setpoints, current values and parameters can be found in the "CANopen Protocol" section.

5.1.1 Bus Commands

If the servo-controller is switched on at the bus, the following command must first be sent, before the PDO's can function. This command can also e sent via the internal SERVO-CONTROLLER PLC.

Command "Start Remote Node Protocol“	
Identifier =	0
Byte 1 =	1
Byte 2 =	0
Number of Bytes to be sent	2

The servo-controller's Send – PDO's can be activated by the servo-controller's internal PLC or by the following command. When you send the "Sync" command, all PDO's set to synchronise ("CANopen parameter dialog box in NORD SERV) will send their content.

"Sync" Command	
Identifier =	80h
Number of Bytes to be sent	0

5.1.2 Fault Report Construction

A fault message is sent for every fault, which occurs in the servo-controller

"Emergency Object" Fault Report	
Identifier =	81h + Controller address
Number of Bytes to be sent	8

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Fault - Code		Available	Bit - Word	Available			

Fault telegram construction

Fault – Code : Contains a 16 Bit number, which specifies the fault. The exact fault meaning and cause are explained in the section "Fault Reports" under "CANopen Protocol".

Bit – Word: Only Bit 0 has any meaning, all other bits are unused. If Bit 0 = 1, then the servo-controller's I*t function was active during the fault. This means, that the servo controller only had the rated current, which leads to the conclusion that a speed or tracking error has occurred. In order to resolve this fault, the work cycles, which are in the overcurrent area, must be minimised.

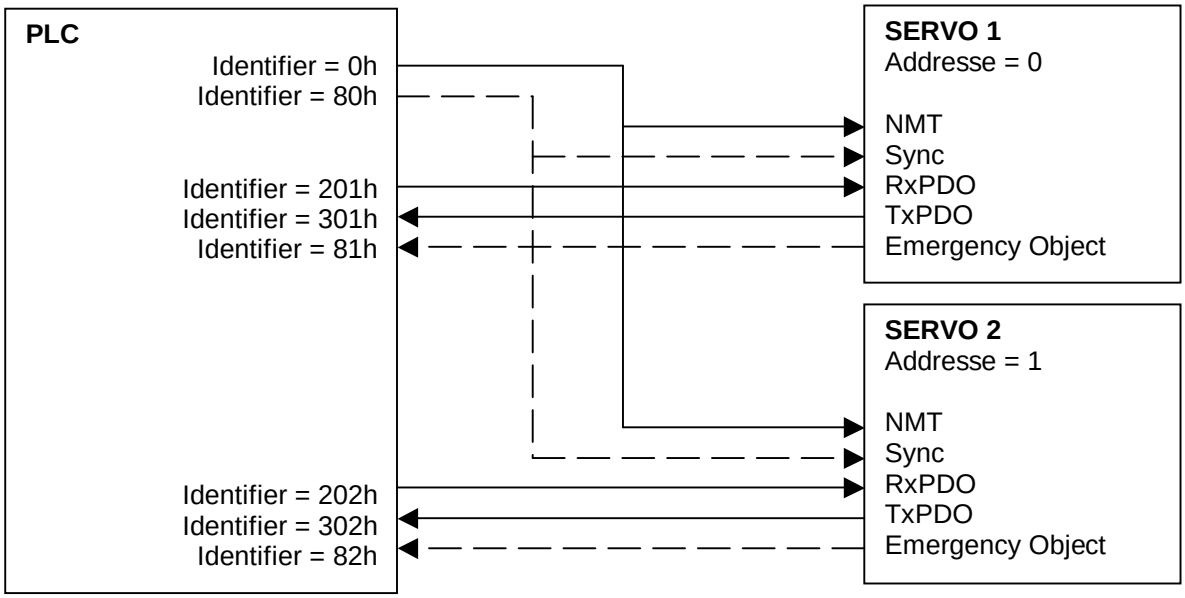
5.1.3 Integration of the Servo-controller in other Protocols

Numerous restrictions must be observed when integrating the PDO's in third-party protocols:

- The address of the receive – PDO's cannot be influenced.
- Before the PDO's can work, the "start Remote Node Protocol" must be sent to the CAN – Bus. The identifier may not be changed.
- The identifier for the "Sync" command may not be changed.
- The identifier for the fault message may not be changed.

5.1.4 Example

The following diagram shows an example of a PLC, which controls two servo-controllers via the CAN – Bus. The switching displayed shows the linking if the PDO's in principle and not the physical cabling. This would comprise a three-wire shielded connection between the three devices.



Principles of PDO connection (dotted connection is not compulsory):
The PLC in the above image starts two servo-controllers in position mode. It presets the set positions via the RxPDO's, can release and block the servo-controller output stages and set the servo-controller outputs. This occurs, assisted by the following RxPDO's:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Control word 6040h		Set position 607Ah				Outputs 2B02h

RxPDO

The RxPDO's are absolutely identical in construction and only differ in their identifier.

The PLC receives the following information concerning the servo-controller's TxPDO's:
- Output stage released
- Fault in servo-controller
- Servo-controller has reached set-point
- which inputs are set

The first three pieces of information are drawn from the status value 6041h.

Byte 0	Byte 1	Byte 2	Byte 3
Status word 6041h		Inputs 2B01h	

RxPDO

The TxPDO's are also absolutely identical in construction and only differ in their identifier.

Before the setpoints and current values can be sent via the PDO's, the "Start Remote Node Protocol" Bus command must be sent via the NMT channel. Readout of the TxPDO's to the servo-controller can take place via the "Sync" bus command or the internal servo-controller PLC.

Assisted by this PDO switching, the PLC can carry out the following functions via the Bus with both servo-controllers:
- Switching the output stage on and off
- Pre-setting the servo-controller set positions
- Monitoring reaching set positions
- Setting the servo-controller outputs

- Importing servo-controller inputs
- Recognising and resetting a fault state in the servo-controller.

The fault number can be evaluated directly by using Emergency Objects.

5.2 CANopen - Protocol

The components relevant for the servo-controller support **CANopen Specification DS301 V4.01 and DS402 V1.1.**

The **EDS Configuration File** is supplied together with the NORD SERV user interface program. This contains all data of the functions supported in accordance with CANopen specifications. Readout of this enables CANopen devices to configure themselves without problems, if they are standard. This EDS file has been successfully tested with devices from numerous renowned manufacturers.

5.2.1 Functions supported in DS301

- NMT = Slave
- Error Control = Heardbead
- Number PDO's = 2 Rx 2 Tx
- PDO Modes = asynchronous and also synchronous for T-PDO
- PDO Mapping = variable
- Number SDO's = 1 Server
- Emergency Message = yes
- Save = yes

Limitations:
- The transmission type for receive – PDO's is fixed to asynchronous (254) (1400h).
- For send PDO's, synchronous (1) and asynchronous (254) transmission types are possible (1800h).
- Only 4 entries per PDO can be mapped (1600h & 1A00h)
- Only the last 4 faults are saved in the Emergency Message Buffer (1003h).
- It is not possible to move the address for the Emergency Objects (1014h).

Note on NMT State machine:
- Use the "Reset_Node indication" command to reset the entire servo-controller, i.e. the output stage is switched off (motor without current).
- The "Reset_Communication indication" is not supported, the same action as the "Reset_Node indication" command occurs.

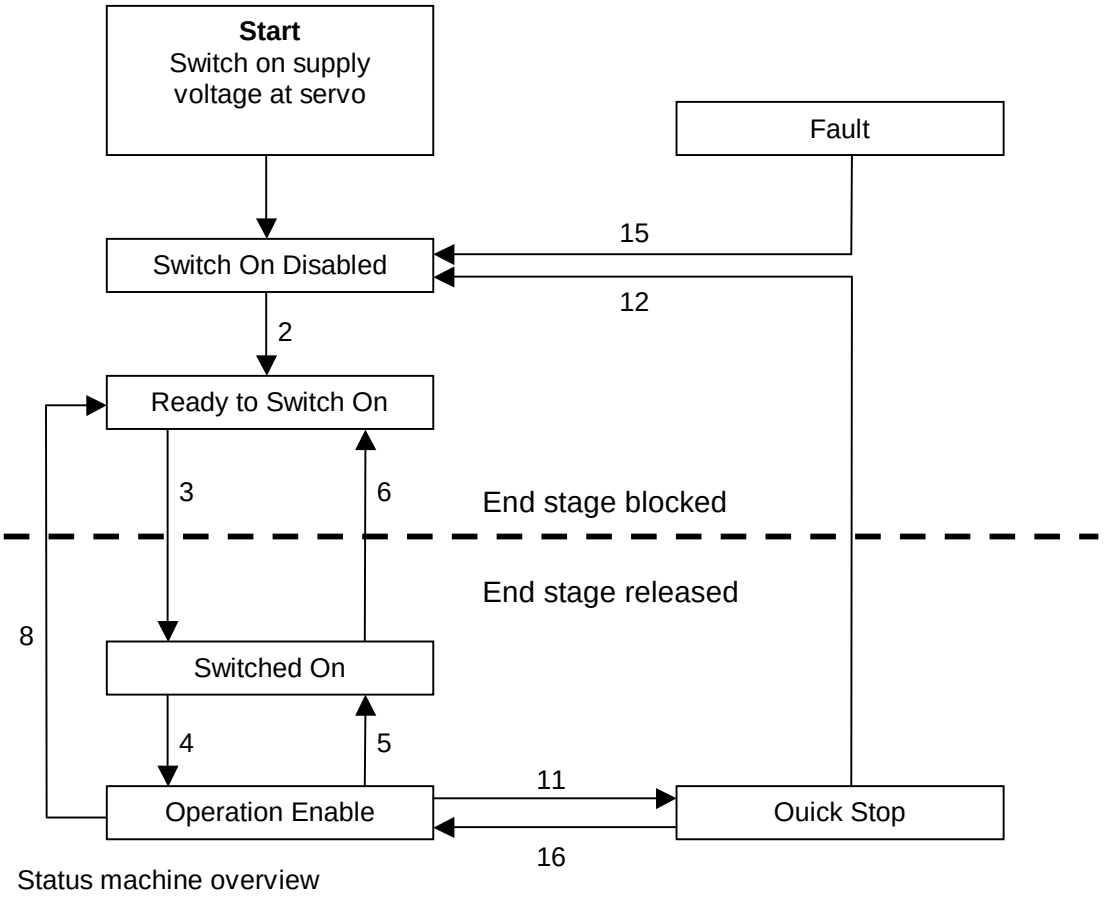
5.2.2 Functions supported in DS402
- Status machine
- Error Codes
- Modes = Profile Position Mode

Limitations:
- In the Profile Position Mode, the parameter: Motion Profile Type 6086h is fixed on linear ramps (0).

Furthermore, Velocity Mode, Profile Torque Mode and Home Mode abut the protocol.

5.2.3 Servo-controller Status Machine

The controller's individual states are shown in the blocks of the status machine. These states are described in the following section. A transition must be activated, to be able to switch between the individual states. The transitions are represented by arrows and are described in the section "Description of Transitions".



Status machine overview

Description of states:

Switch On Disabled
Initialisation of servo-controller is complete.
DC link capacitors are loaded by a resistor.
This is followed by an automatic jump to the next state.
Parameters can be read and written by Bus.
The PLC and all servo-controller movement functions are blocked.

Ready to Switch On
Servo-controller has complete operating voltage.
Parameters can be read and written by Bus.
The PLC and all servo-controller movement functions are blocked.

Switched On
Output stage is released and there is a current at the motor.
All Parameters can be read and partially written by Bus.
The PLC and all servo-controller movement functions are blocked.

Operation Enable
The PLC and all servo-controller movement functions are available.
No faults have been determined.
All Parameters can be read and partially written by Bus.

Quick Stop Active
Emergency Stop function brakes the motor to 0 rpm at the current limit.
The PLC and all servo-controller movement functions are blocked.
All Parameters can be read and partially written by Bus.

Fault
Fault determined in servo-controller.
The PLC and all servo-controller movement functions are blocked.
Speed is braked to 0 rpm at current limit and the output stage is then switched off.
In the event of current faults or overcurrent, the output stage is switched off immediately.
The mechanical brake operates.
All Parameters can be read and written by Bus.

Description of transitions:
Transition 2
Takes place automatically after the controller DC-link is loaded.

Transition 3
Event: - "Switch On" – Command from Master
- Enable input switches level from Low to High
Action: - Output stage is released and there is a current at the motor.
- Some initialisations occur

Transition 4
Event: - "Enable Operation" – command Master
- Transition 3 is started by Enable input
Action: - The PLC and all servo-controller movement functions are available.

Transition 5
Event: - "Disable Operation" – command from Master
- Enable input switches level from High to Low when braking function is switched on
Action: - Speed is braked at current limit at 0 rpm
- The PLC and all servo-controller movement functions are blocked.

Transition 6
Event: - "Shutdown" – command from Master
- Transition 5 is started by Enable input
Action: - The mechanical brake operates.
- Output stage is blocked and the motor has zero potential (not disconnected from mains!)

Transition 8
Event: - "Shutdown" – command from Master
- Enable input switches level from High to Low
Action: - Output stage is blocked and the motor has zero potential
- The PLC and all servo-controller movement functions are blocked.
- The mechanical brake operates (Note! If the motor is still rotating, brake wear will occur.)

Transition 11
Event: - "Quick Stop" – Command from Master
Action: - Speed is braked at current limit at 0 rpm
- The PLC and all servo-controller movement functions are blocked.

Transition 12
Event: - "Quick Stop" – command ended
- "Disable Voltage" – command from Master
Action: - Output stage is blocked and the motor has zero potential (not disconnected from mains!)
- The mechanical brake operates.

Transition 15
Event: - "Fault Reset" – command from Master
Action: - Fault Bit in Control word is deleted

Transition 16
Event: - "Enable Operation" – command from Master
Action: - The PLC and all servo-controller movement functions are available.

5.2.4 Object Directory Parameters

The most important parameters for the operation of the servo-controller are explained in the following section. Execution of all parameters necessary for the CANopen protocol will be waived. Consult the "Application Layer and Communication Profile" handbook of the CiA User Organisation, Draft Standard 301 Version 4.01 for these.

5.2.4.1 Control Parameters

Index	**6040h**
Sub – Index	0
Name	control word
Explanation	controls all states in servo-controller
Type	Unsigned 16
Access	write and read
Unit	none

Bit of the control word / Command	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	Transition
Shutdown	0	X	1	1	0	2, 6, 8
Switch On	0	X	1	1	1	3
Disable Voltage	0	X	X	0	X	7, 9, 10, 12
Quick Stop	0	X	0	1	X	7, 10, 11
Disable Operation	0	0	1	1	1	5
Enable Operation	0	1	1	1	1	4, 16
Fault Reset	0 ⇒ 1	X	X	X	X	15

The control commands are activated by the following bits

Bit4 = starts the preset Home – Run, only if the servo-controller is in "Homing Mode" or in "Profile Position Mode", see Index 6060h.
1 = Homing Run is started

Bit15 = Control word block / if the Control word is linked to a PDO mapping, then this bit enables the PDO to be sent without the Control word be evaluated.
1 = Control word is invalid/ignore
0 = Control word is valid

Index	**6060h**
Sub – Index	0
Name	modes of operation
Explanation	Enables regulation mode to be set
Type	Integer 8
Access	write and read
Unit	none

Modes	Number
Profile Position Mode	1
Velocity Mode	2
Torque Profile Mode	4
Homing Mode	6

Assignment of modes to Index 6060h

Index	**2C00h**
Sub – Index	0
Name	PLC Control word
Explanation	Enables PLC to be started and current execution position to be read out
Type	Unsigned 16
Access	write and read
Unit	none

Bit	Function
0	PLC Address Bit 0
1	PLC Address Bit 1
2	PLC Address Bit 2
3	PLC Address Bit 3
4	PLC Address Bit 4
5	PLC Address Bit 5
6	PLC Address Bit 6
7	PLC Address Bit 7
8	PLC Address Bit 8
9	PLC Address Bit 9
10	
11	
12	
13	
14	
15	PLC on/off Bit 15 = 1 = SPS on

Bit15 is used to switch the PLC function on and off for write access to this Index. Furthermore, with a write access where Bit 15 = 1, the PLC address is interpreted as the start address for the PLC program. This enables individual program segments to be started individually. The respective address can be seen in the PLC window in NORD SERV, they are however, only visible in Complex mode (Options/NORD SERV Functionality/Complex command). If the PLC program starts at the beginning, then the address 0 is sent.

For read access of this index, Bit 15 displays the status of the PLC and the PLC address shows the current execution position.

Warning

The address sent in Index 2C00 is not checked. An incorrect address can lead to faulty PLC program execution. For this reason, take appropriate care with addresses, which jump in the middle of the PLC program.

Index	2C01h
Sub – Index	0
Name	plc_address
Explanation	sets the address for write access in Index 2C02h
Type	Unsigned 16
Value range	0 to 496
Access	Write
Unit	none

Index	2C02h
Sub – Index	0
Name	plc_position
Explanation	overwrites the movement position of the internal PLC program addressed in Index 2C01h
Type	Integer 32
Access	Write
Unit	Increments

Index	2C03h
Sub – Index	0
Name	plc_speed
Explanation	overwrites the movement speed of the internal PLC program addressed in Index 2C01h
Type	Integer 16
Value range	0 to 7FFFh
Access	Write
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Assisted by indexes 2C01h, 2C02h and 2C03h, it is possible to overwrite the movement positions and speeds of the internal PLC, whilst operational.
To do this, the address of the movement position of interest must be set above Index 2C01h. The addresses are visible in the PLC window of NORD SERV, behind the movement commands. This requires NORD SERV to be switched to Complex mode. Do this by selecting the "Options/NORD SERV _ Functionality/Complex" command and click OK on the dialog box which appears.
The new position can now be sent direct to the servo-controller's PLC program via Index 2C02h.
The new speed can now be sent direct to the servo-controller's PLC program via Index 2C03h.

Warning

Before the position sent is written to the memory of the PLC program, a check is made, whether access to this memory line is allowed. In the event of a mistake, a fault will occur and the servo-controller will brake at the current limit to 0 rpm. This fault check is not one hundred percent certain, it is possible that incorrect memory access will not be recognised. This will lead to faulty PLC program execution. For this reason, be extremely careful when using these three commands.

Index	2C20h
Sub – Index	0
Name	PLC Marker 1
Explanation	Marker 1 of the PLC program can be written and read out directly via this Index.
Type	Integer 16
Access	write and read
Unit	none

Index	2C21h
Sub – Index	0
Name	PLC Marker 2
Explanation	Marker 2 of the PLC program can be written and read out directly via this Index.
Type	Integer 16
Access	write and read
Unit	none

Index	2C22h
Sub – Index	0
Name	PLC Marker 3
Explanation	Marker 3 of the PLC program can be written and read out directly via this Index.
Type	Integer 16
Access	write and read
Unit	none

Index	2C23h
Sub – Index	0
Name	PLC Marker 4
Explanation	Marker 4 of the PLC program can be written and read out directly via this Index.
Type	Integer 16
Access	write and read
Unit	none

Index	2C24h
Sub – Index	0
Name	PLC Marker 1
Explanation	Marker 1 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	2C25h
Sub – Index	0
Name	PLC Marker 2
Explanation	Marker 2 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**2C26h**
Sub – Index	0
Name	PLC Marker 3
Explanation	Marker 3 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**2C27h**
Sub – Index	0
Name	PLC Marker 4
Explanation	Marker 4 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**2C28h**
Sub – Index	0
Name	PLC Marker 5
Explanation	Marker 5 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**2C29h**
Sub – Index	0
Name	PLC Marker 6
Explanation	Marker 6 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**2C30h**
Sub – Index	0
Name	PLC Marker 7
Explanation	Marker 7 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**2C40h**
Sub – Index	0 to 100
Name	Teach Buffer
Explanation	The memory of the PLC program can be written and read out via this Index. The memory written can be read out and evaluated in PLC markers. Memory locations are addressed by the Sub-Index. Example: Sub- Index 0 = Memory location 0 Sub- Index 1 = Memory location 1
Type	Integer 32
Access	write and read
Unit	none

This Index can only be written if the Teach Memory (command = Technology Functions/Teach In) is declared and started in the first line of the PLC program.

Index	**2205h**
Sub – Index	0
Name	Profibus control word
Explanation	There is an explanation of the control word function in the section "The Control Word (STW)" in the Profibus section.
Type	Unsigned 16
Access	write and read
Unit	none

5.2.4.2 Status and Current Value Parameters

Index	**6041h**
Sub – Index	0
Name	Status word
Explanation	Servo-controller status word, displays current state.
Type	Unsigned 16
Access	read only
Unit	none

State	Bit 6	Bit 5	Bit 3	Bit 2	Bit 1	Bit 0
Switch On Disabled	1	X	0	0	0	0
Ready to Switch On	0	1	0	0	0	1
Switched On	0	1	0	0	1	1
Operation Enabled	0	1	0	1	1	1
Fault	0	X	1	1	1	1
Quick Stop Active	0	0	0	1	1	1

Servo-controller states

Bit8 = Stopping brake status
0 = Brake open, motor can rotate freely
1 = Brake closed

Bit9 = Remote
0 = Servo-controller ignores all set-points
0 = Servo-controller accepts set-points

Bit10 = Target Reached, in Profile Position Mode = Position OK, no meaning in all other modes
1 = Servo-controller is in position window
0 = Servo-controller is not within position window

Bit11 = Internal Limit Active, displays whether machine slide has moved to final position switch
1 = Servo-controller limit switches are connected
1 = Servo-controller limit switches are not connected

Bit12 = Homing attained
1 = Homing run successfully completed

Bit13 = Homing error
1 = Fault occurred during homing run, homing not successfully completed

Bit 14 = I*t, Surge current monitoring
1 = Servo-controller can only move with rated current
1 = Servo-controller can move with surge current

Index	**6077h**
Sub – Index	0
Name	torque actual value
Explanation	Displays the current moment of the servo-controller at the engine shaft
Type	Integer 16
Access	read only
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**6044h**
Sub – Index	0
Name	vl control effort
Explanation	Displays the current speed of the servo-controller at the engine shaft
Type	Integer 16
Access	read only
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**6064h**
Sub – Index	0
Name	Position actual value
Explanation	Displays the current position of the servo-controller
Type	Integer 32
Access	read only
Unit	Increments

Index	**2102h**
Sub – Index	0
Name	following error
Explanation	displays the current difference between setpoint and current position
Type	Integer 16
Access	read only
Unit	Increments

Index	**2103h**
Sub – Index	0
Name	dc link voltage
Explanation	Displays the current DC link of the servo-controller
Type	Unsigned 16
Access	read only
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**2B01h**
Sub – Index	0
Name	input
Explanation	Displays state of servo-controller's inputs
Type	Unsigned 16
Access	read only
Unit	none

Bit–No.	Meaning	Bit–No.	Meaning	Bit–No.	Meaning
Bit 0	Home	Bit 4	Motor – Temperature	Bit 8	Input 3
Bit 1	Limit 1	Bit 5	Enable	Bit 9	Input 4
Bit 2	Limit 2	Bit 6	Input 1	Bit 10	Input 5
Bit 3	Available	Bit 7	Input 2	Bit 11	Input 6

Assignment of Bits and Inputs

Index	2206h
Sub – Index	0
Name	Profibus status word
Explanation	There is an explanation of the control word function in the section "The Status Word (ZSW)" in the Profibus section.
Type	Unsigned 16
Access	write and read
Unit	none

5.2.4.3 Setpoints

Index	6071h
Sub – Index	0
Name	target torque
Explanation	Moment set-point
Type	Integer 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

If this set-point is set irregularly to zero in speed or positioning mode, this will acts as circuit entering and affect the control behaviour of the servo-controller. Sending this set-point during an active movement command has no effect on the internal PLC.

Index	6042h
Sub – Index	0
Name	vl target velocity
Explanation	Speed set-point
Type	Integer 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

If this set-point is set irregularly to zero in positioning mode, this will act as circuit entering and affect the control behaviour of the servo-controller.

Index	607Ah
Sub – Index	0
Name	target position
Explanation	Position set-point
Type	Integer 32
Access	write and read
Unit	Increments

If this set-point is sent during an active movement command, it will be ignored by the internal PLC.

Index	2B02h
Sub – Index	0
Name	outputs
Explanation	Sets the servo-controller's outputs
Type	Unsigned 8
Access	write and read
Unit	none

Bit–No.	Output	Bit–No.	Output
0	Output 1	4	Output 5
1	Output 2	5	Output 6
2	Output 3	6	Available
3	Output 4	7	Relay output

Assignment of Bits and Outputs for Index 2B02h

If outputs are assigned by pre-defined functions (NORD SERV, "Outputs" parameter dialog box), this will lead to the outputs written in Index 2B02h being continuously overwritten.

5.2.4.4 Parameters

All parameters listed here can be changed in the servo-controller whilst operational.

Index	**6081h**
Sub – Index	0
Name	profile velocity
Explanation	This is the speed to which maximum acceleration will be made in Profile Position Mode
Type	Unsigned 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

This value is only transferred with a new set-point. For PDO mapping, this means that "profile velocity" must be located on the front PDO – bytes, followed by the set-point, see diagram below.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Profile velocity 6081h		target position 607Ah			

Example of an RxPDO with two set-points

Index	**6083h**
Sub – Index	0
Name	profile acceleration
Explanation	Acceleration ramp
Type	Unsigned 16
Value range	2 to 7FFFh / 2 = fastest acceleration
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**6084h**
Sub – Index	0
Name	profile deceleration
Explanation	Brake ramp
Type	Unsigned 16
Value range	2 to 7FFFh / 2 = fastest braking
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**6086h**
Sub – Index	0
Name	motion profile type
Explanation	Enables ramp type for braking and acceleration ramps to be set. -2 = Acceleration S-curve and braking linear -1 = Acceleration linear and braking S-curve 0 = Acceleration and braking linear 1 = Acceleration and braking S-curve
Type	Integer 16
Value range	-2 to 1
Access	write and read
Unit	none

Index	**2800h**
Name	velocity gain factor
Explanation	Boost factor for speed controller
Type	Unsigned 16
Value range	0 to 7FFFh / 0 = smallest possible boost
Access	write and read
Unit	none

Index	**2801h**
Sub – Index	0
Name	velocity integration factor
Explanation	Integration factor for speed controller
Type	Unsigned 16
Value range	0 to 7FFFh / 0 = smallest possible I-share
Access	write and read
Unit	none

Index	**2900h**
Sub – Index	0
Name	position gain factor
Explanation	Boost factor for positioning controller
Type	Unsigned 16
Value range	0 to 7FFFh / 0 = smallest possible boost
Access	write and read
Unit	none

Index	**2903h**
Sub – Index	0
Name	position integration factor
Explanation	Integration factor for position controller
Type	Unsigned 16
Value range	0 to 100 / 0 = smallest possible I-share
Access	write and read
Unit	none

Index	**2A04h**
Sub – Index	0
Name	Gear ratio
Explanation	Contains the transmission factors for the gear function. Both factors are sent together as a string (see string construction below). The new transmission is valid immediately, even if the inverter is currently in gear mode.
Type	32Bit String
Access	write and read
Unit	none

32 Bit String			
1. Word (Unsigned 16)		2. Word (Unsigned 16)	
Gear factor Master		Gear factor Slave	
10h	00h	20h	00h

In this example, a gear transmission from 1000h/2000h or 1 to 2 is set.

Note:
Always enter the largest possible numbers in the gear factors, as this increases the precision of the function.

Index	**2A06h**
Sub – Index	0
Name	synchronizing way
Explanation	Gives the acceleration ramp for the Flying Saw in increments multiplied by 8. At the end of the ramp, the Slave moves synchronous with the Master speed. Example: 10000 sent generates an acceleration range of 80000 increments.
Type	Unsigned 16
Value range	1 – 32767
Access	write and read
Unit	Increments * 8

Note:
This parameter may not be changed whilst a gear function is active.

5.2.4.5 Data Management

Index	**1010h**
Sub – Index	1
Name	save all parameters
Explanation	Save all parameters and the PLC in the controller permanently (takes approx. 8 seconds).
Type	Unsigned 32
Value range	To save, write access of the following value must be sent: 65766173h
Access	write and read
Unit	none

Index	**2010h**
Sub – Index	0
Name	Access to data base
Explanation	The entire parameter set of a servo-controller can be read out and then re-imported with this Index. The data must be sent and read using the SDO-segment protocol.
Type	Array (minimum size 405 Byte)
Access	write and read
Unit	none

Index	**2020h**
Sub – Index	0
Name	Access to plc program
Explanation	The entire PLC program of a servo-controller can be read out and then re-imported with this Index. The data must be sent and read using the SDO-segment protocol.
Type	Array (minimum size 1010 Byte)
Access	write and read
Unit	none

5.3 Servo-controller Fault Reports

In the event of a fault state in the servo-controller, an Emergency Object is sent via the CAN-Bus. The first 2 Bytes (Byte 0 & 1) contain the fault code, as described in the CANopen specification DS402 v1.1.
Byte 2 contains the Error – Register 1001h, in the event of a fault, only Bit 0 is set to High. In the manufacturer-defined fault field (Byte 3 to 7), Bit 0 is only overlaid in Byte 3. If it is Fault event 1, then the I*t – function was active during the fault initiation. This means, that the servo controller only had the rated current available, which leads to the conclusion that a speed or tracking error occurred. In order to resolve this fault, the work cycles, which are in the overcurrent area, must be minimised.

Index	Flash Code	Description	Reason
2000h 8192D	flashes 1x	Controller immediately switches output stage off, due to overcurrent.	- Motor short-circuit - Motor earth fault - Output stage overtemperature
2001h 8193D	flashes 2x	Controller immediately switches output stage off, due to overcurrent.	- incorrect motor - very poor controller settings
2002h 8194D	flashes 2x	Controller immediately switches output stage off, due to overcurrent.	- incorrect motor - very poor controller settings
3100h	flashes 3x	One or more mains phases switched off at controller.	- Emergency off activated Supply voltage insufficiently fused - FI circuit breakers in circuit - Faulty power line connection to servo
3210h 12816D	flashes 4x	DC link is too high, this fault leads to the motor current being switched off immediately.	- Braking resistor not connected - Braking resistor faulty
3220h 12832D	flashes 5x	The DC-link for the servo-controller is below the permissible value	- Incorrect input voltage - An input voltage phase is missing
4300h 17152D	flashes 6x	Motor has exceeded category temperature	- Motor temperature sensor not connected - Dirt preventing motor from emitting heat via enclosure - Current controller poorly set with motor customer datasets
5400h 21504D	flashes 7x	Output stage reports permanent fault	If this fault is displayed, despite repeated switching on, then the SERVO-CONTROLLER is faulty.
5441h 21569D	flashes 8x	The servo-controller's Enable input is at Low – level,	The servo-controller can only be switched on by software, if the Enable input has a High – level, for safety.
5442h 21570D	flashes 9x	Limit input 1 has a Low – level.	The machine slide has entered Limit 1 or the limit input has not been interconnected or disconnected.
5443h 21571D	flashes 10x	Limit input 2 has a Low – level.	The machine slide has entered Limit 2 or the limit input has not been interconnected or disconnected.
6210h 25104D	flashes 11x	Fault in servo-controller PLC program	This fault occurs, if an movement command is started in the PLC program, with gears switched on or during stepping motor function, during the homing run.
6220h 25120D	flashes 11x	Fault in servo-controller PLC program	The value set in the "Limit Moment" command exceeds the value entered in the "Limits" parameter dialog box.
6221h 25121D	flashes 11x	Fault in servo-controller PLC program	An attempt was made to store one position more in the "Teach In" function, than was registered in the "Teach In" command.
6222h 25122D	flashes 11x	Fault in servo-controller PLC program	An attempt was made to export a "Teach In" position, which was outside the memory area reserved for it.
6223h 25123D	flashes 11x	Fault in servo-controller PLC program	An attempt was made to export a "Teach In" position, but there was no "Teach In" command in the first line.
6224h 25124D	flashes 11x	Fault in servo-controller PLC program	A multiplication overruns the 32Bit number range
6225h 25125D	flashes 11x	Fault in servo-controller PLC program	Division by zero.

6230h 25136D	flashes 11x	Fault in servo-controller PLC program	The "Gear function on" command was executed during an active movement command, in Home mode or during stepping motor operation. This fault report does not occur, if this command occurs during an active gear function.
6240h 25152D	flashes 11x	Fault in servo-controller PLC program	The "Stepping Motor Function on" command was executed during an active movement command, in Home mode or during gear operation.
6250h 25168D	flashes 11x	Fault in servo-controller PLC program	A clearly incorrect PLC command should be executed. This can lead to incorrect addresses being entered, if the PLC code is accessed directly via Indexes 2C00h, 2C01h or 2C02h.
6251h 25169D	flashes 11x	Fault in servo-controller PLC program	A negative speed was sent to the servo-controller via Index 2C03h.
6260h 25184D	flashes 11x	Fault in servo-controller PLC program	The winder processor has been incorrectly parameterised, e.g. the rewinder was parameterised, but the compensator (unwinder) was started ("Winder" parameter dialog box).
6270h 25200D	flashes 11x	Fault in servo-controller PLC program	An unparameterised disk or camshaft has been called up via the PLC.
6271h 25201D	flashes 11x	Fault in servo-controller PLC program	An attempt was made to start the disk cam function whilst the gear, winder or compensator functions were on, or during a Homing run.
6290h 25232D	flashes 11x	An incorrect address in the internal PLC program was accessed via index 2C01h.	- There is a different PLC program in the servo-controller The addresses in the PLC program have been changed by inserted or deleted commands
6291h 25233D	flashes 11x	The Home command is ignored.	The motor axle was moving, when Home was started
6300h 25344	flashes 11x	A fault occurred via Index 2010 whilst a dataset was being downloaded,	An attempt was made to load a dataset with a different output stage type into the controller.
7310h 29456D	flashes 12x	The axle was outside the speed deviation fault window for longer than the set time period.	- Axle blocked by obstruction in the machine or by gear fault, - too small tracking error selected - Encoder incorrectly connected (interconnection) Encoder offset incorrectly selected (interconnection)
8120h 33056D	flashes 13x	The controller has no communication with control for the time set in switched on state.	- Safety time period too small - Connector cable faulty or fallen out
8130h 33072D	flashes 13x	The servo-controller's CAN driver has gone into "Bus off" state.	- Connector cable unsuitable, faulty or fallen out - Connector cable not properly screened, not proper 120 Ohm terminating resistor - Devices with differing baud rates on bus - strong EMC effects
8410h 33808D	flashes 14x	The speed setpoint sent exceeds the upper "Limit" set in the Parameter dialog box.	The encoder is changed, by a different speed constant being produced (NORD SERV menu "Devices/Constants") *and so different binary values.*
8611h 34321D	flashes 15x	The max. tracking error entered in the Parameterisation/Limits dialog box (Positioning deviation) has been exceeded.	- Axle blocked by obstruction in the machine or by gear fault, - Axle sluggish, due to insufficient maintenance (lubrication) - Incorrect parameter set or optimisation parameter (often incorrect selection of

			positioning controller boost or pre-control) - too small tracking error selected
8710h 34576D	flashes 16x	Slave has run out of the tracking error window set to the master, during gear function.	- Slave axle blocked by obstruction in the machine or by gear fault, - Slave axle sluggish, due to insufficient maintenance (lubrication) - Incorrect parameter set or optimisation parameter in slave servo-controller (often incorrect selection of positioning controller boost or pre-control) - Faulty electrical connection between both servo-controllers. - Monitoring function incorrectly set in master servo-controller (transmission, slave rotation)
8810h 34832D	flashes 17x	Flying start active.	Speed limit exceeded during moment mode (also applies for winder and compensator regulation)

5.4 Example: Connection of Servo-controller to PC-PLC via CANopen

This example relates to the connection of the servo-controller to the TWINCAT PLC from Beckhoff. The integration of the servo-controller in the system manager will be explained. An example program is supplied on the installation disks. The exact sequence of the program is explained in an accompanying Readme file.

5.4.1.1 Integration of CAN card in PC

- Start system manager (see accompanying TWINCAT instruction manual).
- Start a new project with "File/New".
- Highlight the "E/A Devices" heading under "E/A Configuration" in the tree diagram and right click.
- Select the appropriate CAN card under the CANopen heading.
- Select the CAN baud rate in the newly created device.

5.4.1.2 Integration of the servo-controller

- Select the PC CAN card in the tree diagram and right click.
- Select "CANopen Node" under "General".
- Highlight the new entry and select "CAN Node" in the box, which appears on the right.
- Enter the address + 1 set at the controller under "Node ID".
- 402 under "Profile No." and 2 under Add. Information".
- Set "Guard Time“ and "Life Time Factor“ to 0.
- Select the "Auto Set" and "Auto Download" check boxes.
- Additional settings, such as PDO settings, can be entered in the "SDO" box.

5.4.1.3 Setting the TxPDO's

- Highlight the heading "TxPDO 1" in the tree diagram.
- Select the "PDO" tab in the dialog box, which appears on the right.
- You may change the "COB-Id".
- Possible settings for "Trans. Type“ are "1 (cyc / sync)“ and "254 (async)“
- all other PDO values are not supported.
- Repeat these steps for TxPDO 2.

5.4.1.4 PDO Mapping for TxPDO’s

- Highlight the "Inputs" heading under "TxPDO1" in the tree diagram and right click.
- Select "Add Variables".
- Select name and type from the dialog, which now appears.
- Select the input, which now appears.
- Select the "Variable" tab in the dialog box, which appears on the right.
- Click the "Link" button.
- Link the Input with a suitable variable.
- Repeat this process, until all variables have been entered.
- If no variables appear in the connection dialogue box, this can either be because there are no global variables of the same type, or that no SPS project is integrated.
- Repeat these steps for TxPDO 2.

5.4.1.5 Setting the RxPDO's

- It is not possible to change the settings for RxPDO's, because address and transmission type are fixed in the servo-controller.
- Highlight the heading "RxPDO 1" in the tree diagram.
- Select the "PDO" tab in the dialog box, which appears on the right.
- Check the following settings:
  Address: RxPDO 1 = 201h + controller address // RxPDO 2 = 301h + controller address
  Transmission Type = 254 (async)
- Repeat these steps for RxPDO 2.

5.4.1.6 PDO Mapping for RxPDO’s

- This is the same process as under the heading of the same name for TxPDO's.

Finally, save the settings in the registry, by selecting "Action/Save in Registry".

6 Connection of Servo-controller to Profibus

6.1 General

This PROFBUS DP documentation only applies to the SK1000E device range.

The PROFIBUS is connected to the basic unit by a Profibus- CAN Gateway (CAN-CBM-DP) from "esd electronic system design gmbh“. A Siemens SIMATIC from S7-300 is required for controlling.

Up to 8 inverters can be operated at a gateway. Up to 16 inverters are possible at a gateway, with communication limitations.

6.2 Data Transfer

6.2.1 Reference Data Structure

Cyclical data traffic between the Master and the inverter is described in this section.

The reference data is divided in two sections:

- PKW-Area (Parameterisation; **P**arameter-Recognition-Value)
- PZD-Area (**Pro**cess **d**ata)

Parameters can be read and written via the PKW area of the reference data. All task, which take place via the PKW interface, are predominantly configuration, monitoring and diagnostic tasks.
The PZD area is used to control the inverter. The control word and status word, as well as setpoints and current values are transferred via the process data.

Access comprises order and response telegrams. In the order telegrams, reference data is transferred from the Master to the Slave. In the response telegrams, reference data is transferred from the Slave to the Master. The construction of both telegrams is identical.

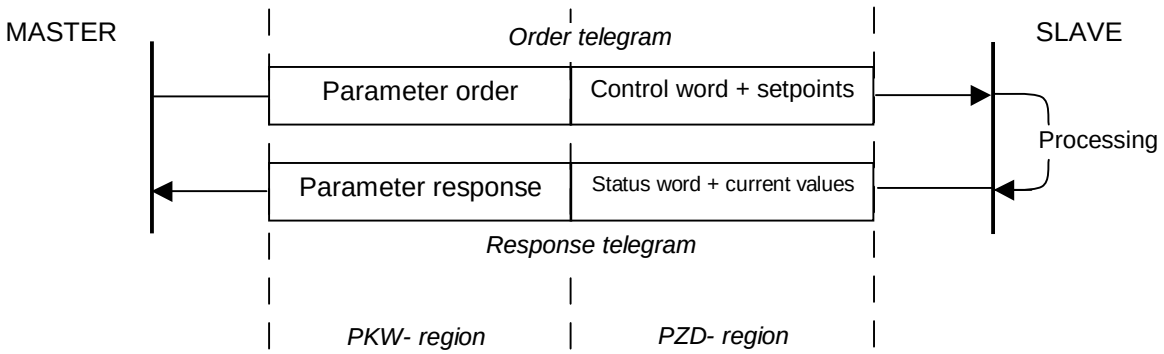


Image: Telegram traffic / Construction of Reference Data Area

Process data and parameter data are processed immediately in the inverter.

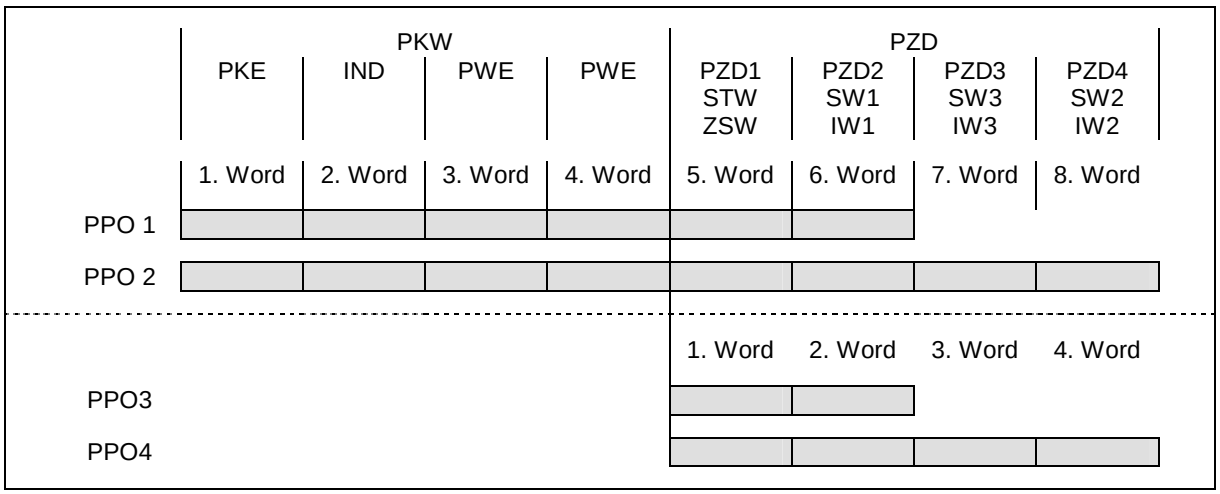
6.2.2 PPO - Types

A Parameter Process data Object (PPO) is defined for cyclical data traffic, with which both Process Data (PZD) and Parameters (PKW) can be transferred from the Master to the inverter. The inverter can process PPO types 1,2,3 or 4.

Type	Task
PPO1	Expanded parameter telegram with 32Bit parameter value and process data
PPO2	Telegram with expended process data (main and two subsidiary setpoints) and 32Bit parameter value.
PPO3	Process data telegram with main setpoint without parameter data
PPO4	Expanded process data telegram with main and subsidiary setpoints without parameter data

PPO 3 and PPO 4 are pure process data objects for applications, which exist without cyclical parameter processing.

The following diagram shows an overview of the PPO types supported.



Abbreviations used:

PPO	Parameter Process Data Object	STW	Control Word
PKW	Parameter identifier Value	ZSW	Status Word
PZD	Process Data	SW1..3	Setpoint 1-3
PKE	Parameter identifier	IW1..3	Current Value 1-3
IND	Index		
PWE	Parameter Value		

Note:

A PLC can usually only transmit double words via I/O memory access consistently. System functions (e.g. SFC 14/15) must be used for longer data formats (PKW channel always / PZD files with PPO2 or PPO4).

6.2.3 Process Data

In the Process Data area (PZD), control words and setpoints are transferred from the Master to the inverter and in return, status words and current values are sent from the inverter to the Master. The construction of the PZD area is always the same in terms of the sequence of its elements (words), however, dependent upon direction of data Master ⇒ Inverter / Inverter ⇒ Master, it is described differently.

The process data area of the reference data is constructed in the following manner:

- STW: Control Word; Length 16Bit, Order Telegram contains Control bits (e.g. Enable, Emergency Stop, Fault Acknowledgement)

- ZSW: Status word; Length 16Bit, Response Telegram contains Status bits (e.g. Inverter running, Fault)

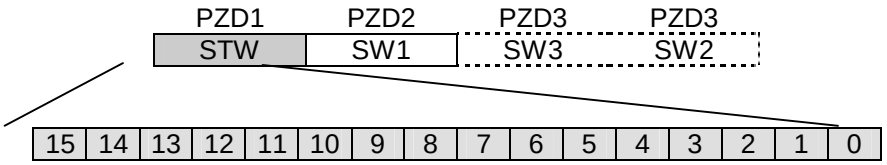
- SW1..3 **S**etpoints; maximum 3 possible, 16 or 32Bit, Order Telegram
  e.g. Positioning setpoint, speed setpoint, moment setpoint.

- IW1..3 Current values; maximum 3 possible, 16 or 32Bit, Response Telegram
  e.g. Positioning current value, speed current value, moment current value.

	1. Word	2. Word	3. Word	4. Word	
PZD area with 1x16-Bit setpoint	STW ZSW	SW1 IW1			PP0-Type 1,3
PZD- area with up to 3 16-Bit setpoints	STW ZSW	SW1 IW1	SW3 IW3	SW2 IW2	PP0-Type 2,4
PZD area with 1x 32-Bit setpoint and 1x 16-Bit	STW ZSW	SW1 IW1		SW2 IW2	PP0-Type 2,4

6.2.3.1 The Control Word (STW)

The control word (STW) is the first word transferred to the inverter in the process data area in an order telegram.

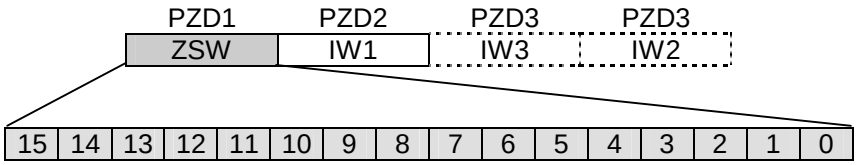


Meaning of individual Bits

Bit	Value	Meaning	Comment
0	0	Out 1	Braking with speed ramp at 0 rpm and then current activation
	1	In	**Operational**
1	0	Out 2	Block current; the servo output current is switched off, servo goes to **standby** status**.**
	1	Operating condition	Out 2 is rescinded
2	0	Out 3	Emergency stop at current limit or emergency stop ramp at 0 rpm and then current activation. Servo goes to **standby** status.
	1	Operating condition	Out 3 is rescinded
3	0	Block operation	Block current; the servo output current is switched off, servo goes to **standby** status.
	1	Release operation	Release output current, execute adjacent setpoint
4	0/1	no function	
5	0/1	no function	
6	0	Block setpoint	All servo-controller setpoints are blocked. (This also applies to setpoints generated by the internal PLC).
	1	Release setpoint	All servo-controller setpoints are released and can be written
7	0		
	1	Acknowledge	With the switch from 0 to 1, no longer active faults will be acknowledged.
8	0/1	no function	
9	0/1	no function	
10	0/1	no function	
11	0/1	no function	
12	0/1	no function	
13	0/1	no function	
14	0/1	no function	
15	0		
	1	Start Homing Run	With the switch from 0 to 1, the Homing run set in the servo-controller will be started.

6.2.3.2 The Status Word (ZSW)

The status word (ZSW) is the first word transferred to the inverter in the process data area in a response telegram.



Meaning of individual Bits

Bit	Value	Meaning	Comment
0	0	Not at standby	
	1	Standby	Initialisation complete, load relay on, output current blocked
1	0	Not operational	Reason: No On command present, fault present, OUT2 or OUT3 present, **Switch On Block** status present.
	1	Operational	On command present, no faults present Servo can be started with RELEASE OPERATION command.
2	0	Operation blocked	
	1	Operation released	Output current released, run-up to present setpoint.
3	0	No faults	
	1	Fault	Drive malfunctioning and so out of order, if acknowledgement is successful, will go to **Switch On Block** status.
4	0	OUT 2	OUT 2 command present.
	1	No OUT 2	
5	0	OUT 3	OUT 3 command present.
	1	No OUT 3	
6	0	No Switch On Block	
	1	Switch On Block	Goes through OUT1 to **Standby** status
7	0/1	no function	
8	0	Current value not OK	Set position not yet reached
	1	Current value OK	Set position reached
9	0	Local guidance	Local guidance active at device
	1	Guidance required	The Master is called upon, to apply guidance
10	0/1	no function	
11	0		
	1	Rotation right	Motor rotates clockwise
12	0		
	1	Rotation left	Motor rotates anticlockwise
13	0	No Home	The servo-controller has not yet carried out a homing run.
	1	Home OK	A homing run has been completed successfully.
14	0		
	1	Home Error	The homing run set is not supported by the servo-controller or a homing run was interrupted.
15	0	I*t out	The complete surge current is available.

	1	I*t in	The servo-controller has been operated with too high current (greater than the rated current) for too long, and has reduced the maximum possible current to the value of the rate current.

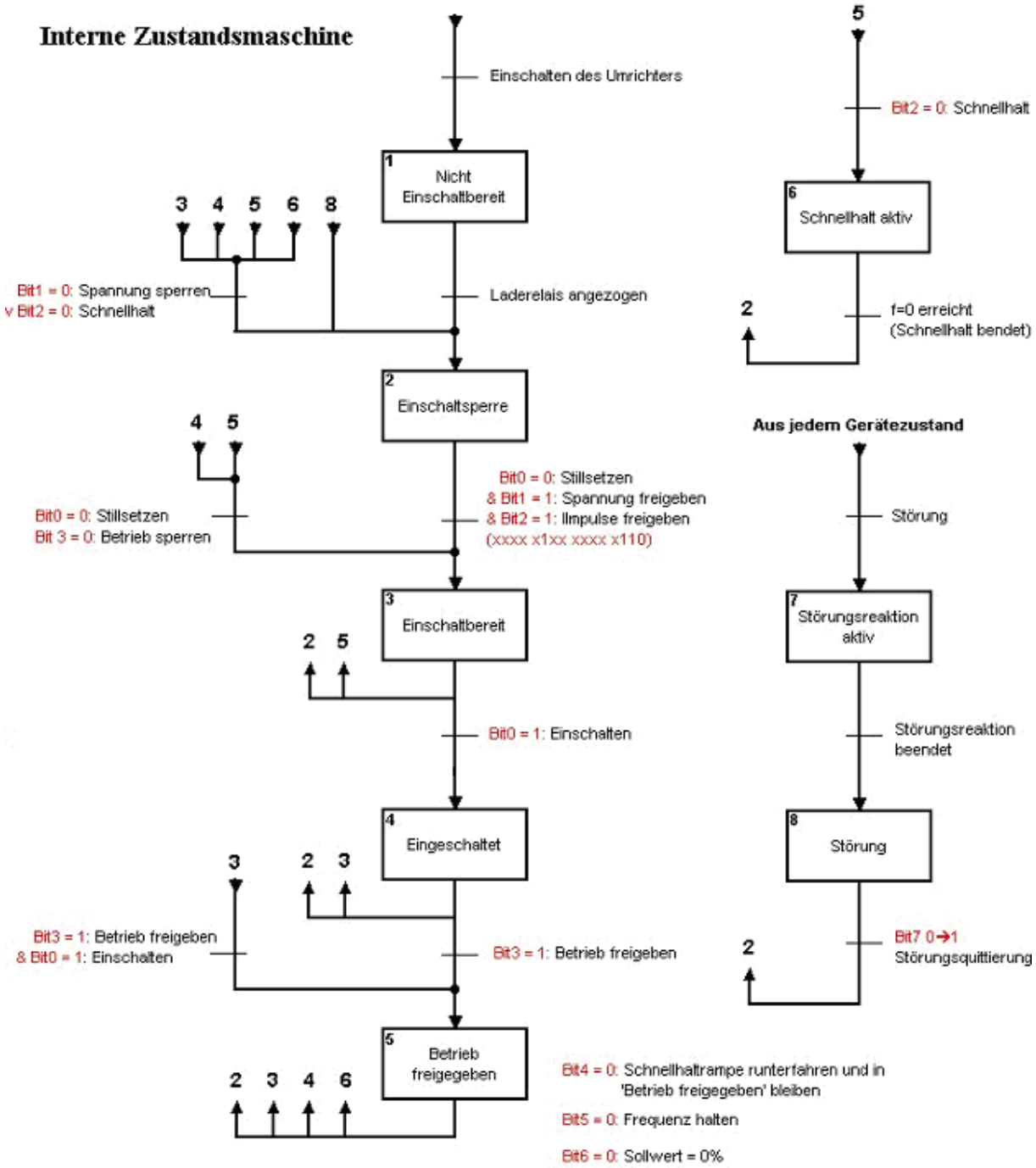
6.2.4 Status Machine

The inverter always controls itself via its status machine. The transitions between the various states occur by the appropriate control commands in the process data control word. The current status is reported back in the process data status word.

After switching on, the servo-controller enters **Switch On Block** status. After the DC-link capacitors are loaded, this status is exited automatically, the servo then remains in **Standby** status.

The following status word bits provide the servo-controller's status:

Status	Bit6 Switch On Block	Bit5 Emergency Stop	Bit4 Block Current	Bit3 Fault	Bit2 Release Operation	Bit1 Operational	Bit0 Standby
Not at standby	0	X	X	0	0	0	0
Switch On Block	1	X	X	0	0	0	0
Standby	0	1	1	0	0	0	1
Switched On	0	1	1	0	0	1	1
Operation Released	0	1	1	0	1	1	1
Fault	0	X	X	1	0	0	0
Fault Active	0	X	X	1	1	1	1
Emergency Stop Active	0	0	1	0	1	1	1



Steuerbits

0. Betriebsbereit / Stillsetzen
1. Spannung freigeben / sperren
2. Impulde freigeben / Schnellhalt
3. Betrieb freigeben / sperren
4. frei
5. frei
6. Sollwert freigeben / sperren
7. Störungsquittierung (0→1)
10. Steuerdaten gültig / ungültig
11. - 14. frei
15. Start - Home - Fahrt

Priorität der Steuerbefehle:

1. Spannung sperren
2. Schnellhalt
3. Stillsetzen
4. Betrieb freigeben
5. Einschalten
6. Betrieb sperren
7. Reset Störung

Kennzeichnung der Zustände:

1: Bit 0 = 0
2: Bit 6 = 1
3: Bit 0 = 1
4: Bit 1 = 1
5: Bit 2 = 1
6: Bit 5 = 0
7: Bit 2 & Bit 3 = 1
8: Bit 3 = 1

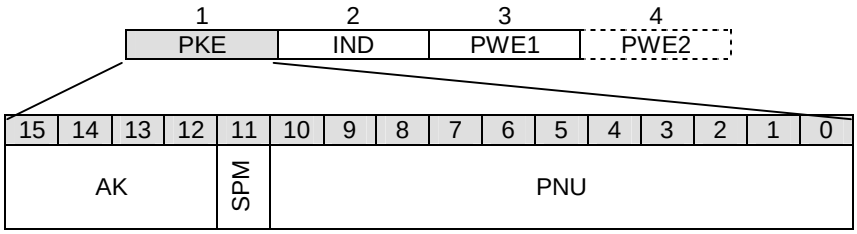
6.2.5 Parameter Area (PKW)

The PKW mechanism enables parameter processing to be carried out in cyclical data traffic. To do this, the Master formulates an order and the inverter formulates the answer to it.
The parameter area is always made up of a **Parameter identifier**, in which the order type (write, read etc.) and the relevant parameters are set. With the help of the **Index**, individual parameter sets or array elements can be addressed. The **Parameter Value** comprises the value to be written or the value to be read.

Note:
A parameter order must be repeated until the inverter replies with an appropriate response telegram.

6.2.5.1 Parameter Identifier (PKE)

Order or response and the relevant parameters are encoded in the Parameter Identifier (**PKE**).



The Parameter Identifier (**PKE**) is always a 16Bit identifier.

PNU: Bits 0 to 10 contain the number of the parameter required (**PNU**), or, in the inverter's response telegram, the number of the current parameter.

Note: Please refer to the following section for parameter numbers (**PNU**) for NORDAC SK1000 E series inverters.

SPM: Bit 11 is the Toggle Bit for spontaneous reports. This function is **not** supported!

AK: Bits 12 to 15 contain the order or response identifier.

The following table lists all of the orders, which can be transferred from Master to Inverter. The right column contains the response, which would normally be sent (response identifier positive). Only particular response identifiers are possible, dependent upon the order identifier. In the event of a fault (AK negative), the value 7 is always sent from the Inverter to the Master in the order identifier.

AK	6.2.5.2 Function	Positive Response Identifier
1	Request Parameter Value	1 / 2
2	Change Parameter Value (word)	1
3	Change Parameter Value (double word)	2
4	Reserved	-
5	Reserved	-
6	Request Parameter Value (Array)	4 / 5
7	Change Parameter Value (array word)	4
8	Change Parameter Value (array double word)	5

Meaning of response identifier values sent:

AK	Function
0	No response
1	Parameter value transferred (word)
2	Parameter value transferred (double word)
4	Parameter value transferred (array word)
5	Parameter value transferred (array double word)
7	Order not executable (with fault number in PWE2)

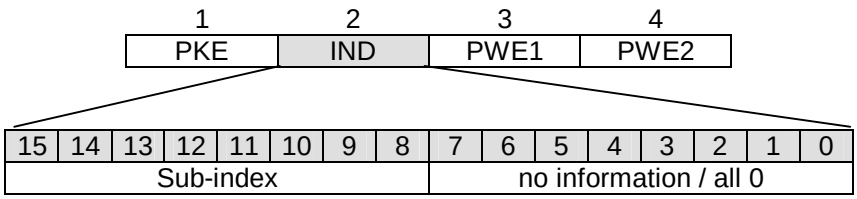
The inverter supplies the response to the last order, for as long as the order is not executed.Therefore, the Master must be check ed to ensure that the response received matches the order sent. **The value in the response identifier (AK), the parameter number received (PNU) with the relevant Index (IND) and the current parameter value (PWE) for writing parameters can be used for the plausibility check.**

Fault reports, if the order is not to be executed

If the response identifier is "Order not executable" (AK = 7), then a fault report is added to the parameter value (**PWE2**) of the inverter response. The following table displays the meanings of the values transferred,

No.	Statement
0	Parameter number not permissible
1	Parameter value may not be changed
2	upper or lower value limit exceeded
3	faulty sub-index
4	no array
5	Data type not permissible
6	Reset only (only 0 may be written)
7	Writing element may not be changed
9	Writing data not present
201	Invalid order element in last order received
202	Internal response identifier not presentable

6.2.5.3 Sub-index (IND)



The sub-index is entered in the top 8Bit of the word.

6.2.5.4 Parameter Value (PWE)

Dependent on PPO type or parameter, the transfer of the parameter value (PWE) always occurs as word (16Bit) or double word (32Bit). Only one parameter value can ever be transferred in a telegram.
A 32Bit parameter value comprises PWE1 (higher-value word) and PWE2 (lower-value word).
A 16Bit parameter value is transferred in PWE2 with PPO 1 and PPO.

6.3 Parameters

6.3.1 Base

Index	1150
Sub – Index	0
Name	Set position
Explanation	Contains the target position to be moved to.
Type	Integer 32
Access	write and read
Unit	Increments

Index	1151
Sub – Index	0
Name	Profile speed
Explanation	This is the speed to which maximum acceleration will be made in Positioning Mode.
Type	Unsigned 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	1152
Sub – Index	0
Name	Profile acceleration
Explanation	Acceleration ramp in positioning mode.
Type	Unsigned 16
Value range	2 to 7FFFh / 2 = fastest acceleration
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	1153
Sub – Index	0
Name	Profile braking
Explanation	Brake ramp in positioning mode
Type	Unsigned 16
Value range	2 to 7FFFh / 2 = fastest braking
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	1154
Sub – Index	0
Name	Ramp type
Explanation	Enables ramp type for braking and acceleration ramps to be set in positioning mode. -2 = Acceleration S-curve and braking linear -1 = Acceleration linear and braking S-curve 0 = Acceleration and braking linear 1 = Acceleration and braking S-curve
Type	Integer 16
Value range	-2 to 1
Access	write and read
Unit	none

Index	**1157**
Sub – Index	0
Name	Set speed
Explanation	Setpoint in speed mode.
Type	Integer 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

If this set-point is set irregularly to zero in positioning mode, this will act as circuit entering and affect the control behaviour of the servo-controller.

Index	**1158**
Sub – Index	0
Name	Speed ramp
Explanation	Acceleration and braking ramp in speed mode. Sending a zero switches the function off.
Type	Unsigned 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**1160**
Sub – Index	0
Name	Set moment
Explanation	Setpoint in moment mode.
Type	Integer 16
Access	write and read
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

If this set-point is set irregularly to zero in speed or positioning mode, this will act as circuit entering and affect the control behaviour of the servo-controller.

Index	**1165**
Sub – Index	0
Name	Controller mode
Explanation	Enables controller operation type to be set Positioning Regulation = 1 Speed Regulation = 2 Moment Regulation = 4
Type	Integer 8
Access	write and read
Unit	none

Index	**1170**
Sub – Index	0
Name	Outputs
Explanation	Sets the servo-controller's digital outputs and digital relay.
Type	Unsigned 8
Access	write and read
Unit	none

Bit–No.	Output	Bit–No.	Output
0	Output 1	4	Output 5
1	Output 2	5	Output 6
2	Output 3	6	Available
3	Output 4	7	Relay output

Assignment of Bits and Outputs for Index 170

If outputs are assigned by pre-defined functions (NORD SERV, "Outputs" parameter dialog box), this will lead to the outputs written in Index 170 being continuously overwritten.

Index	**1175**
Sub – Index	0
Name	Home mode
Explanation	Enables referencing method to be set. Please see the "Home" parameter dialog box in the NORD SERV Help for a description of Home methods.
Type	Integer 8
Value range	2 to -7
Access	write and read
Unit	none

6.3.2 Regulation

Index	1350
Sub – Index	0
Name	Position P-factor
Explanation	Boost factor for positioning controller
Type	Unsigned 16
Value range	0 to 7FFFh / 0 = smallest possible boost
Access	write and read
Unit	none

Index	1351
Sub – Index	0
Name	Position I-factor
Explanation	Integration factor for position controller
Type	Unsigned 16
Value range	0 to 100 / 0 = smallest possible I-share
Access	write and read
Unit	none

Index	1360
Sub – Index	0
Name	Speed P-factor
Explanation	Boost factor for speed controller
Type	Unsigned 16
Value range	0 to 7FFFh / 0 = smallest possible boost
Access	write and read
Unit	none

Index	1361
Sub – Index	0
Name	Speed I-factor
Explanation	Integration factor for speed controller
Type	Unsigned 16
Value range	0 to 7FFFh / 0 = smallest possible I-share
Access	write and read
Unit	none

Index	**1370**
Sub – Index	0
Name	Gear factor
Explanation	Contains the transmission factors for the gear function. Both factors are sent together as a string (see string construction below). The new transmission is valid immediately, even if the inverter is currently in gear mode.
Type	32Bit String
Access	write and read
Unit	none

32 Bit String			
1. Word (Unsigned 16)		2. Word (Unsigned 16)	
Gear factor Master		Gear factor Slave	
10h	00h	20h	00h

In this example, a gear transmission from 1000h/2000h or 1 to 2 is set.

Note:
Always enter the largest possible numbers in the gear factors, as this increases the precision of the function.

Index	**1371**
Sub – Index	0
Name	Flying saw synchronisation range
Explanation	Gives the acceleration ramp for the Flying Saw in increments multiplied by 8. At the end of the ramp, the Slave moves synchronous with the Master speed. Example: 10000 sent generates an acceleration range of 80000 increments. The increments are the Master encoder's increments.
Type	Unsigned 16
Value range	1 - 32767
Access	write and read
Unit	Increments * 8

Note:
This parameter may not be changed whilst a gear function is active.

6.3.3 Additional Parameter

Index	1560
Sub – Index	1
Name	save all parameters
Explanation	Save all parameters and the PLC program in the controller permanently.
Type	Unsigned 32
Value range	To save, write access of the following value must be sent: 65 76 61 73h
Access	write and read
Unit	none

Note:
This command may only be started when the inverter is switched off (motor without current / "Switch On Block" or "Standby" status), otherwise a fault report will occur.
No communication with the inverter may take place via the bus for the duration of the save process. A report will be sent via the parameter channel when the save procedure is complete (approx. 9 seconds).

6.3.4 Information

Index	1750
Sub – Index	0
Name	Position
Explanation	Displays the current position of the servo-controller.
Type	Integer 32
Access	read only
Unit	Increments

Index	1751
Sub – Index	0
Name	Tracking error
Explanation	displays the current difference between set position and current position
Type	Integer 16
Access	read only
Unit	Increments

Index	1755
Sub – Index	0
Name	Current speed
Explanation	Displays the current speed of the servo-controller at the engine shaft
Type	Integer 16
Access	read only
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	1760
Sub – Index	0
Name	Current moment
Explanation	Displays the current moment of the servo-controller at the engine shaft
Type	Integer 16
Access	read only
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**1765**
Sub – Index	0
Name	DC-Link
Explanation	Displays the current DC link (current at capacitors) of the servo-controller.
Type	Unsigned 16
Access	read only
Unit	None, for conversion, see factor in NORD SERV "Device / Constants" menu

Index	**1766**
Sub – Index	0
Name	Input
Explanation	Displays state of servo-controller's inputs
Type	Unsigned 16
Access	read only
Unit	none

Bit–No.	Meaning	Bit–No.	Meaning	Bit–No.	Meaning
Bit 0	Home	Bit 4	Motor – Temperature	Bit 8	Input 3
Bit 1	Limit 1	Bit 5	Enable	Bit 9	Input 4
Bit 2	Limit 2	Bit 6	Input 1	Bit 10	Input 5
Bit 3	Available	Bit 7	Input 2	Bit 11	Input 6

Bit12 to Bit15 are occupied by undocumented signals.

6.3.5 PLC- Parameters

Index	**1800**
Sub – Index	0
Name	PLC Control word
Explanation	Enables PLC to be started and current execution position to be read out
Type	Unsigned 16
Access	write and read
Unit	none

Bit	Function
0	PLC Address Bit 0
1	PLC Address Bit 1
2	PLC Address Bit 2
3	PLC Address Bit 3
4	PLC Address Bit 4
5	PLC Address Bit 5
6	PLC Address Bit 6
7	PLC Address Bit 7
8	PLC Address Bit 8
9	PLC Address Bit 9
10	
11	
12	
13	
14	
15	PLC on/off Bit 15 = 1 = SPS on

Bit15 is used to switch the PLC function on and off for write access to this Index. Furthermore, with a write access where Bit 15 = 1, the PLC address is interpreted as the start address for the PLC program. This enables individual program segments to be started individually. The respective address

can be seen in the PLC window in NORD SERV, they are however, only visible in Complex mode (Options/NORD SERV Functionality/Complex command). If the PLC program starts at the beginning, then the address 0 is sent.

For read access of this index, Bit 15 displays the status of the PLC and the PLC address shows the current execution position.

Warning

The address sent in Index 2C00 is not checked. An incorrect address can lead to faulty PLC program execution. For this reason, take appropriate care with addresses, which jump in the middle of the PLC program.

Index	**1801**
Sub – Index	0
Name	Marker 1
Explanation	Marker 1 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1802**
Sub – Index	0
Name	Marker 2
Explanation	Marker 2 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1803**
Sub – Index	0
Name	Marker 3
Explanation	Marker 3 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1804**
Sub – Index	0
Name	Marker 4
Explanation	Marker 4 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1805**
Sub – Index	0
Name	Marker 5
Explanation	Marker 5 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1806**
Sub – Index	0
Name	Marker 6
Explanation	Marker 6 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1807**
Sub – Index	0
Name	Marker 7
Explanation	Marker 7 of the PLC program can be written and read out directly via this Index.
Type	Integer 32
Access	write and read
Unit	none

Index	**1810**
Sub – Index	0 to 200
Name	Teach Buffer
Explanation	The memory of the PLC program can be written and read out via this Index. The memory written can be imported and evaluated in PLC markers. Memory locations are addressed by the Sub-Index. Example: Sub- Index 0 = Memory location 0 Sub- Index 1 = Memory location 1
Type	Integer 32
Access	write and read
Unit	none

This Index can only be written if the Teach Memory (command = Technology Functions/Teach In) is declared and started in the first line of the PLC program.

Connection of the gateway to the PLC

The Gateway is parameterised with the SIMATIC Manager:

Firstly, the Gateway's GSD file is loaded in the "HW Config" window. Next, select the "CAN-CBM/DP“ device ("Profibus DP / Additional field devices / Gateway / CAN / V01") from the catalogue. A window now opens, in which you must set the Gateway's PROFIBUS station address.

Note: Ensure that the address entered is the same as the address entered on the hardware side at the gateway.

Double-click on the Gateway symbol to open a dialog box containing the object properties. The properties on the Can side are set in the section "Device-specific Parameters".

Enter the following settings:

CAN- Bit rate	= as required
Communication Window	= No
RTR- Frames	= No
CANopen- Slave	= No
CANopen- Master	= Yes
Start- Frame	= Yes
Page- Mode	= No
Module ID	= 20
Wake Up Time	= 1
Sync Time	= 0

6.3.6 Slot Assignment

The servo-controller's setpoints and current values are assigned to the address area of the PLC via the DP Slave's (Gateway) slots.

To do this, highlight the required gateway slot and double-click "Universal module" in the catalogue under "CAN-CBM/DP“.

You can set the object properties of a slot in the DP-Slave window, by double-clicking on the relevant place.

For further explanation of this dialog box, see the documentation provided by the gateway manufacturer (see chapter "Slot Configuration).

6.4 Servo- controller Bus Settings

In order for the CAN addresses used by the Gateway have the required setpoints and current values, the appropriate settings must be entered in the servo-controller's datasets.

Normally, the maximum transfer in the send and receive directions is 2 x 4 words. An emulation of PPO types 3 and 4 is possible with this. You have a free selection of the content of this transfer, process and/or parameter data can be transferred.

Set the data contents in the *"CANopen"* parameter dialog box in the servo-controller user-interface program NORD SERV. The required transfer data is set in the send and receive PDO's. The respective PDO identifiers (addresses) must be entered in the converter's slots in SIMATIC Manager. This can be seen in the example, which follows.

Operation of the dialog box is explained in the NORD SERV Help. There is a list of all necessary setpoints and current values, as well as parameters in the NORD SERV handbook. It can be called up as a PDF file, or via the *"Info/Manual"* command in NORD SERV.

6.5 Examples: Control Words

6.5.1 Switch On

Starting requirements:

- The servo-controller is in **Standby** status, this occurs automatically after the current is switched on at the servo-controller.

The following control word must be sent to the servo-controller:

High – Byte (Bit 15 – Bit 8)	Low – Byte (Bit 7 – 0)
04h	7Fh

The servo-controller switches the motor current on and immediately applies the subsequent setpoint. If this is not desired, since no homing run has yet take place, then set Bit 6 or 10 to 0. If the servo-controller is being operated with a mechanical motor brake, then it must be switched on with a blocked setpoint. A setpoint may only be given, if the time frame set in NORD SERV (brake release time) has expired and the servo-controller is in **Operation Released** status. Immediate release of a setpoint might lead to a tracking error on the servo-controller, or the setpoint would be ignored.

6.5.2 Starting a Homing Run

Starting requirements:

- Servo-controller is in **Operation Released** status.
- A homing run is parameterised in the servo-controller.

The following control word must be sent to the servo-controller:

High – Byte (Bit 15 – Bit 8)	Low – Byte (Bit 7 – 0)
80h	0Fh

When the following status word is present, then the homing run has been successfully completed.

High – Byte (Bit 15 – Bit 8)	Low – Byte (Bit 7 – 0)
23h	37h

6.5.3 Switch Off

If the servo-controller is switched off via Bits 1 or 3, then the motor will trundle. This is not advisable for motors with mechanical brakes, because the brakes experience increased wear. In this event, it is always advisable to switch the servo-controller off via Bits 0 or 2. In both cases, electrical speed braking takes place at the current limit, until the axle is at a standstill.

6.5.4 Reset Fault

Starting requirements:

- Servo-controller is in **Fault** status.

The following control word must be sent to the servo-controller:

High – Byte (Bit 15 – Bit 8)	Low – Byte (Bit 7 – 0)
00h	80h

The servo-controller automatically goes to **Standby** status.

6.6 Example: Setpoint / Current Value and Parameter Interface

6.6.1 Setpoint / Current Value

IN this example, all parameter settings in SIMATIC Manager and NORD SERV for a positioning application will be explained. The aim of the settings is to send and receive process data to the servo-controller in PPO4 format. The send telegram contains a control word, a target position and a movement speed. The receive telegram contains a status word, the current position and the current speed.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Control Word (16 Bit)		Set position (32 Bit)				Profile Speed (16 Bit)	

Send Telegram construction

The construction and function of the control word are explained in the previous section. The set position is given in increments of the motor encoder. Note that the encoder line number is quadrupled in the servo-controller. Thus, a 2048 incremental encoder has 8192, or a Resolver 4096 increments per motor rotation. The profile speed is the maximum motor speed to which the set position is moved. It is sent to the servo-controller in binary format. Look up these conversion constants in NORD SERV under the "Devices/Constants" command, they are dependent upon motor encoder type.

Example conversion for a motor with resolver encoder:

Constant: 1 binary value = 9.1554 rpm

$$Speed_{binary} = \frac{1000U/\min}{9{,}1554}$$

$$Speed_{binary} = 109$$

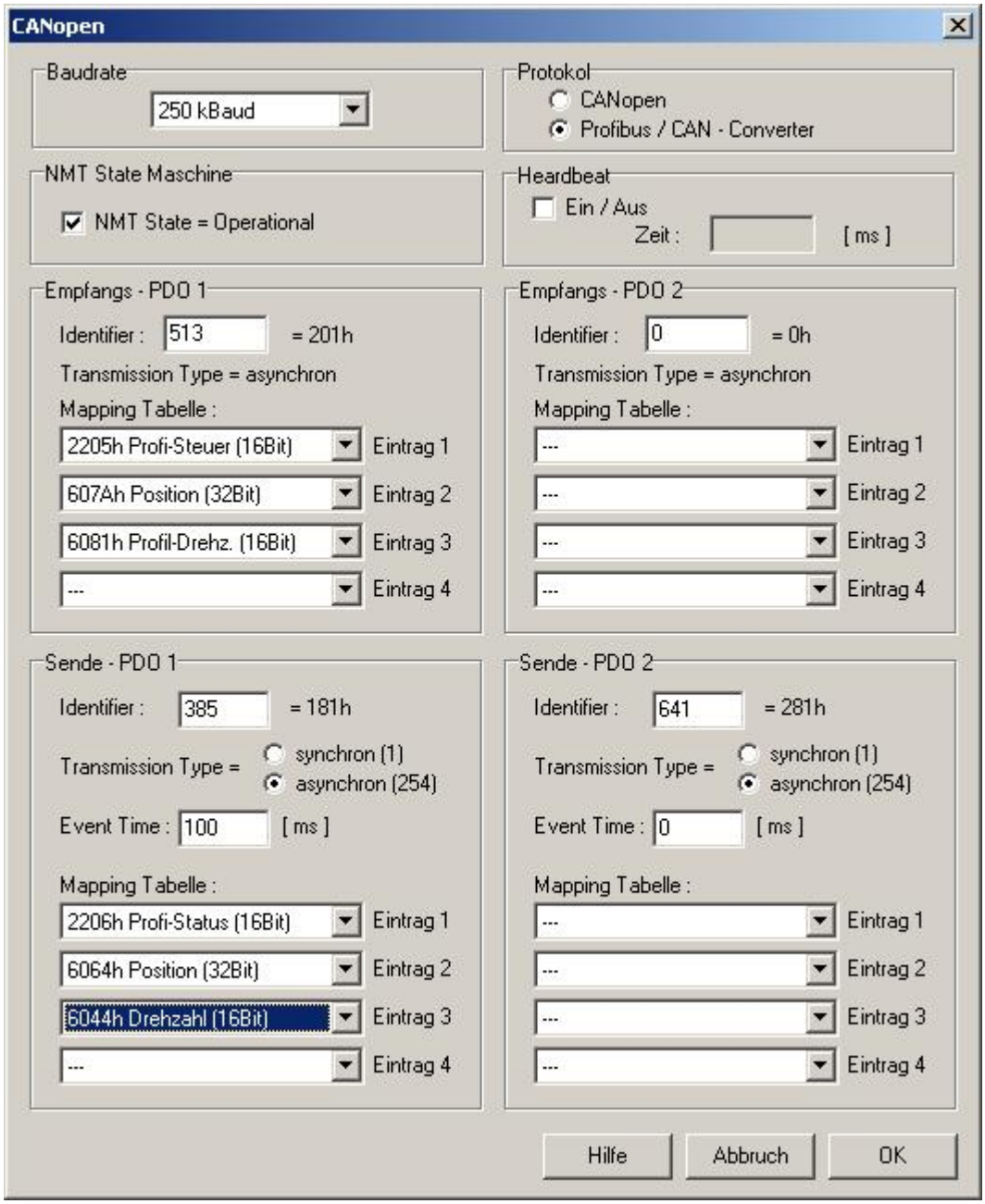
If a value of 109 is sent to the servo-controller as a profile speed, this will generate a maximum speed of 1000 rpm.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Status Word (16 Bit)		Current Position (32 Bit)				Current speed (16 Bit)	

Receive Telegram construction

The construction and function of the status word are explained in the previous section. The current position is sent by the servo-controller in increments. The current speed is sent in the same binary format as the set speed and must be reconverted by multiplication with the same constants in rpm.

6.6.2 Settings under NORD SERV



"CAN open" Parameter dialog box

Apply settings as seen here.

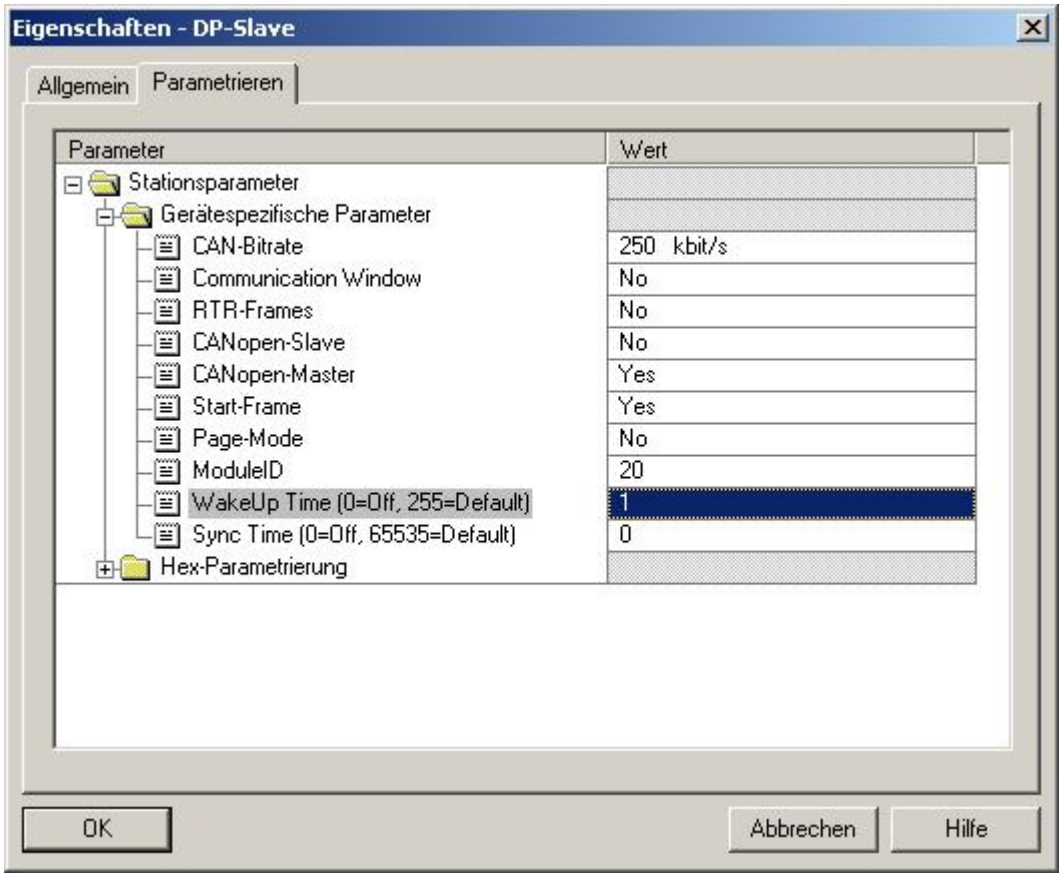
All settings for receiving a PPO4 setpoint telegram are contained in the *"Receive - PDO 1"* grouping. Generally, other Identifiers (addresses) and setpoints/current values or parameters can also be entered here.

The same applies for the *"Send – PDO 1"* grouping, except that this channel sends the current values to the controller. The time rhythm in which this message is sent by the servo is set via the "Event Time" parameter.

The unused send and receive PDO 2 is available for additional setpoints, current or parameter values.

6.6.3 SIMATIC Manager Settings

Setting the PROFIBUS – CAN Gateway object properties.



Apply the parameters as in the image above. With the exception of the baud rate, apply the values (in example 250) from the "CANopen" NORD SERV parameter dialog box.

6.6.4 PROFIBUS – CAN Converter Slot Settings

6.6.4.1 Servo Setpoints

Setpoint settings

All parameters must be applied as above, apart from the PLC address. The entry under *"Manufacturer-specific Data"* is explained as follows:

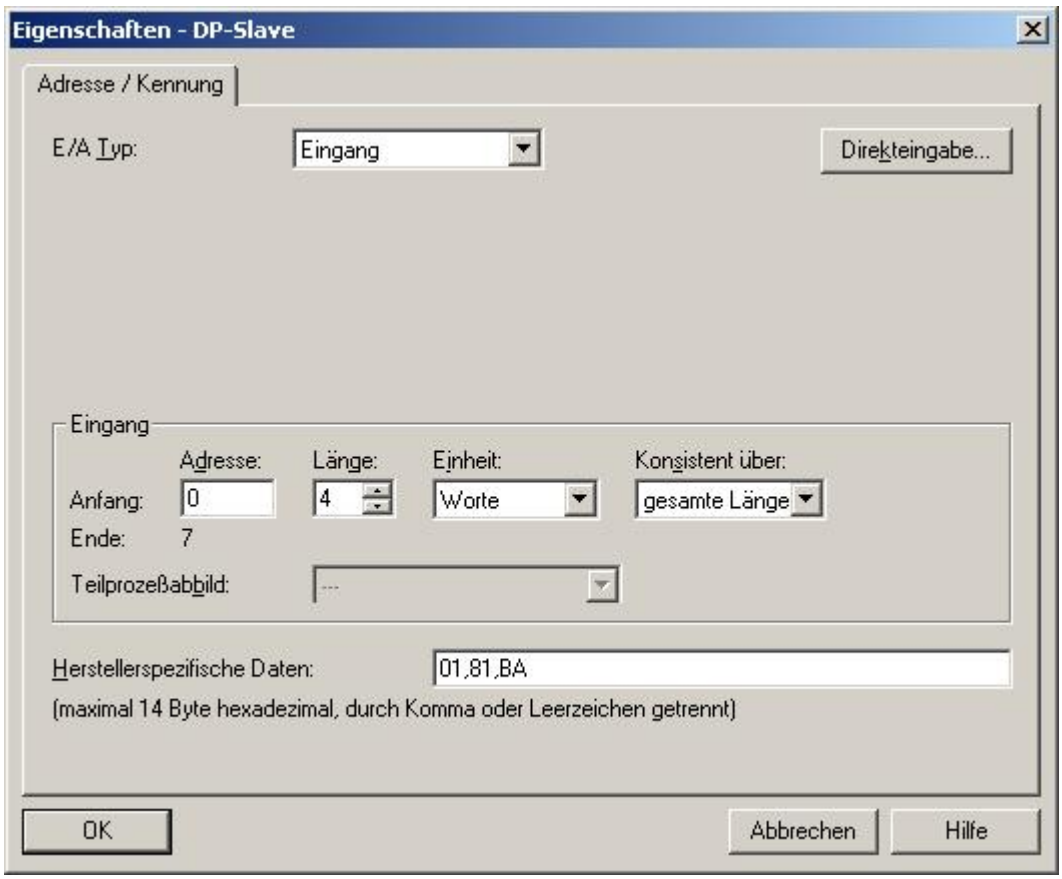
Servo-controller's receive-identifier = 201h, thus generating the first two numbers "02,01".

The "BA" number serves to convert the data sent to Intel syntax. The Gateway documentation contains further explanation.

Setpoint PLC addressing:

Control word on A - Address =	0
Set position on A - Address =	2
Profile speed on A - Address =	6

6.6.4.2 Servo Current Values



Current value settings

All parameters must be applied as above, apart from the PLC address. The entry under "Manufacturer-specific Data" is explained as follows:

Servo-controller's send-identifier = 181h, thus generating the first two numbers "01,81".

The "BA" number serves to convert the data sent to Intel syntax. The Gateway documentation contains further explanation.

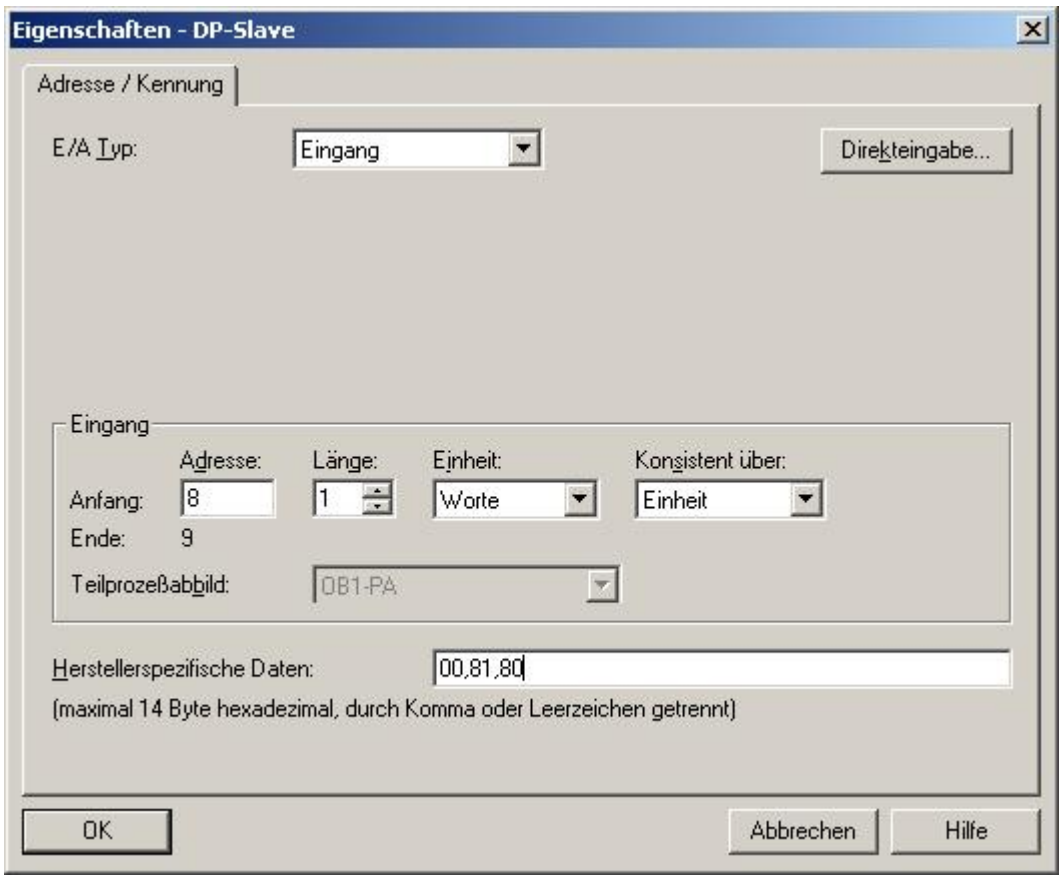
Current value PLC addressing:

Status word on E - Address = 0

Current position on E - Address = 2

Current speed on E - Address = 6

6.6.4.3 Servo Fault Numbers



Servo fault number readout settings

If Fault status is signalised in the status word, then the cause of the fault can be gathered from this parameter.

All parameters must be applied as above, apart from the PLC address. The entry under "Manufacturer-specific Data" is explained as follows:

Servo-controller's emergency-identifier = 81h, thus generating the first two numbers "00,81".

The number "80" serves to convert the data sent to Intel syntax. The converter documentation contains further explanation.

The emergency identifier is derived from the number 80h = the CAN controller address. With an emergency identifier of 81h, all switches at the servo-controller's quad DIP switch must be in the Off position (find more detailed explanation in the servo-controller document "BU1100_DE Hardwaredescription.pdf"").

There is an explanation of fault numbers in this document in the chapter on CANopen. Alternatively, select the *"Info/Manual"* command in NORD SERV.

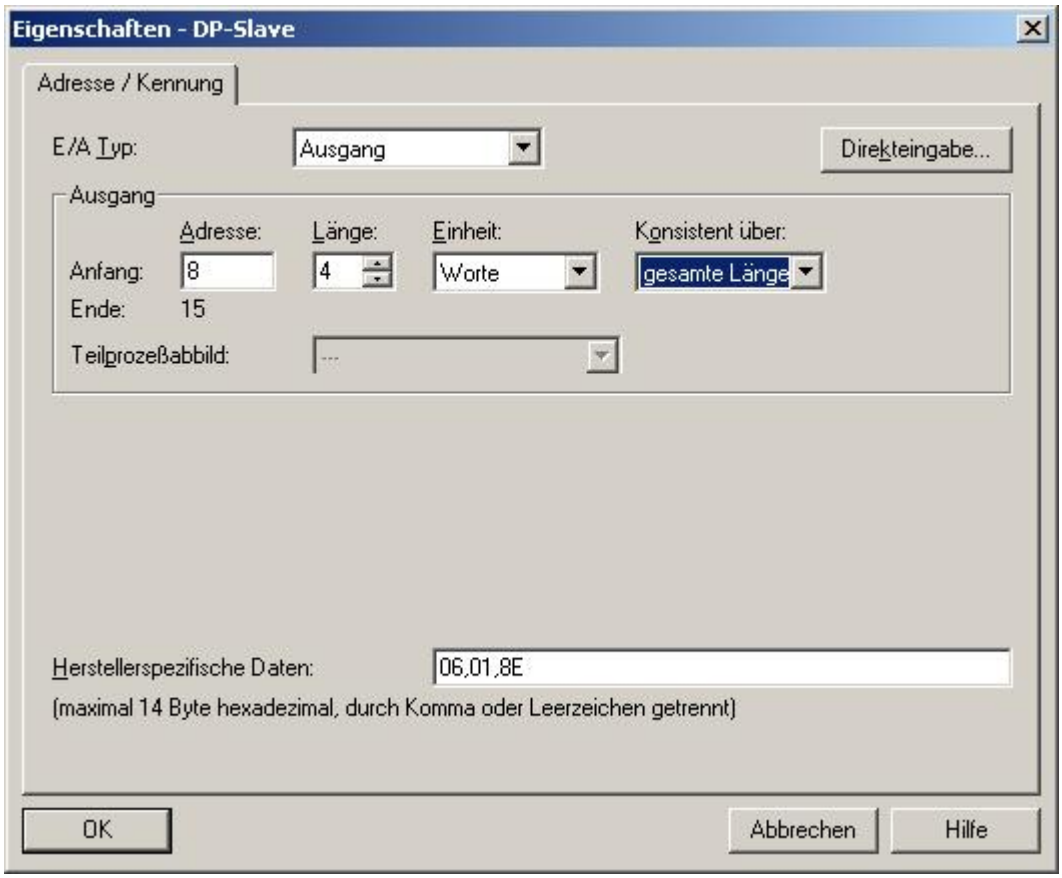
Current value PLC addressing:

Fault number on E - Address = 8

6.6.5 Parameters

Take the following steps to set up the parameter channel.
Ensure the "Profibus / CAN – Converter“ setting has been selected in the "CANopen“ parameter dialog box in the "NORD SERV" program.
Now you need only change the following settings in the Siemens Hardware Manager:

6.6.5.1 Send – Parameter Channel



All parameters must be applied as above, apart from the PLC address. The entry under "Manufacturer-specific Data" is explained as follows:

The first two numbers in the "Manufacturer-specific data" field are derived from the converter address. The number "06,01" represents the controller address.

This number is derived as follows:

601h = 600h + controller address

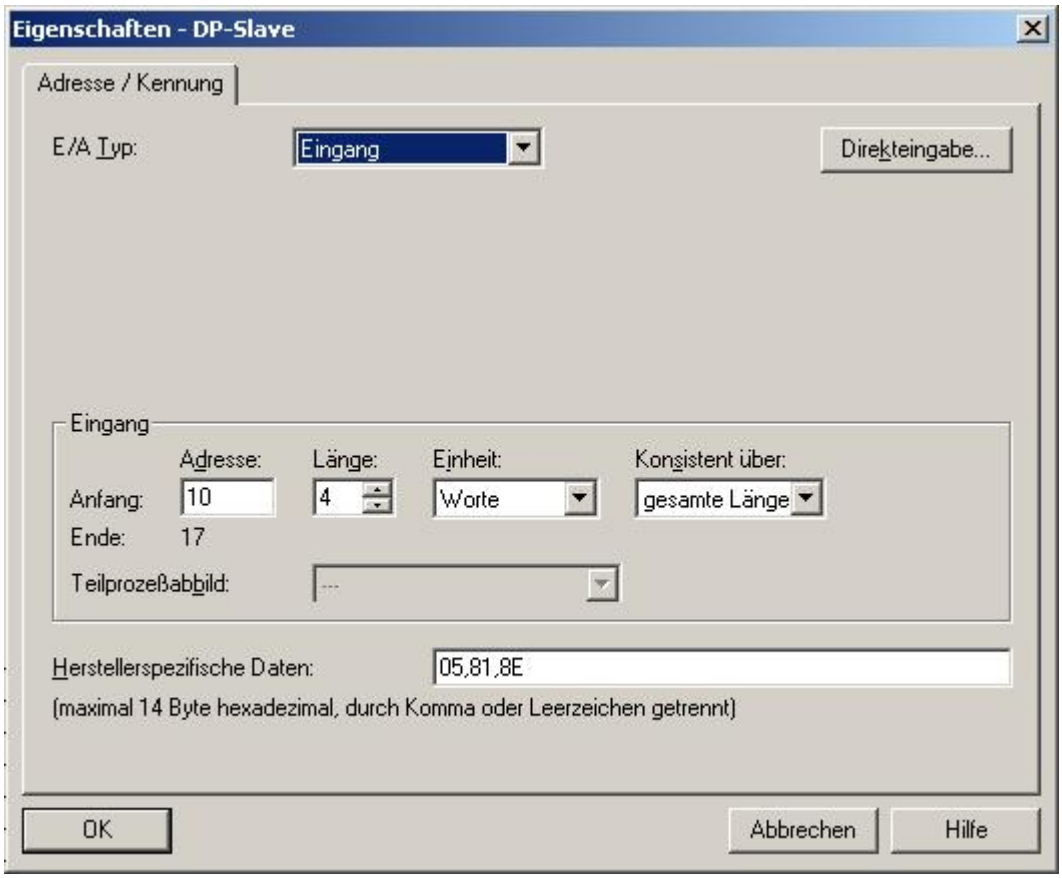
In the image above, the controller address = 1, i.e. all of the switches at the converter's quad DIP – switch were in the off position. There is a more detailed explanation in the document "BU1100_DE Hardwaredescription.pdf".

The number "8E" serves to convert the data sent to Intel syntax. The Gateway documentation contains further explanation.

Setpoint PLC addressing:

Parameter identifier (PKE)	on A - Address = 8
Sub-index (IND)	on A - Address = =10
16Bit Parameter (PWE2)	on A - Address = 14
32Bit Parameter (PWE1 & PWE2)	on A - Address = 12

6.6.5.2 Receive – Parameter Channel



All parameters must be applied as above, apart from the PLC address. The entry under "Manufacturer-specific Data" is explained as follows:

The first two numbers in the "Manufacturer-specific data" field are derived from the converter address. The number "05,81" represents the controller address.

This number is derived as follows:

581h = 580h + controller address

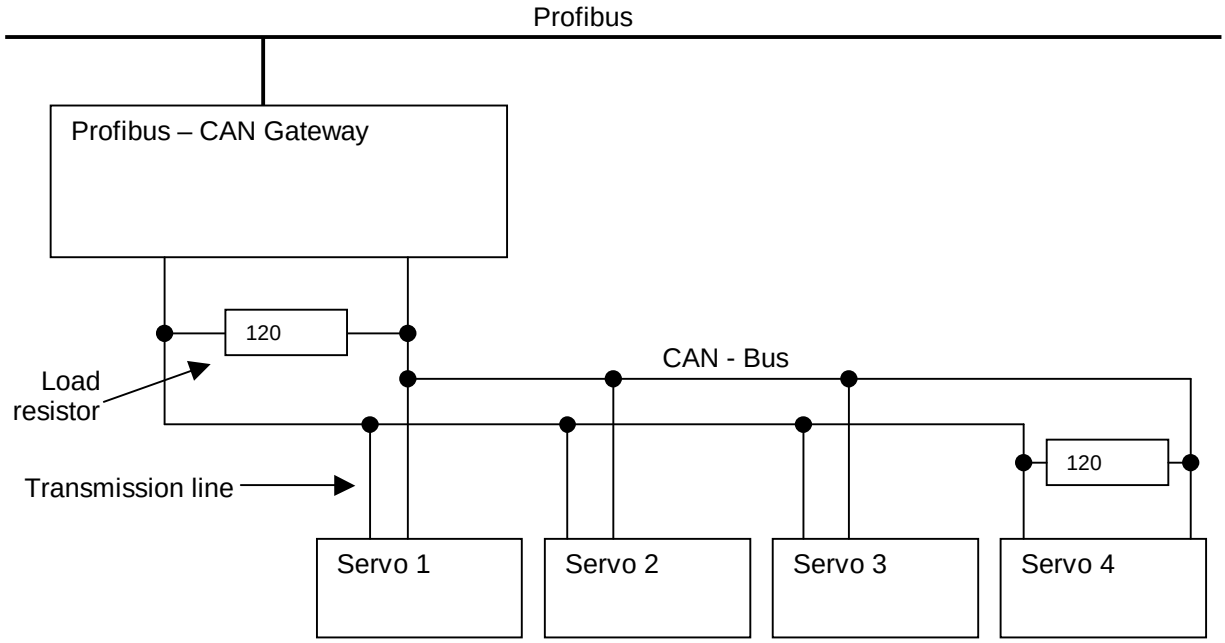
In the image above, the controller address = 1, i.e. all of the switches at the converter's quad DIP – switch were in the off position. There is a more detailed explanation in the document "BU1100_DE Hardwaredescription.pdf".

The number "8E" serves to convert the data sent to Intel syntax. The Gateway documentation contains further explanation.

Setpoint PLC addressing:

Parameter identifier (PKE)	on A - Address = 10
Sub-index (IND)	on A - Address = 12
16Bit Parameter (PWE2)	on A - Address = 16
32Bit Parameter (PWE1 & PWE2)	on A - Address = 14

6.7 General Cabling Notes



CAN Bus cabling schematic

Note the following points when cabling the CAN bus:

- a CAN network may not branch, short transmission lines are an exception.
- Transmission lines to devices may not be longer than 30 cm.
- 120 Ohm load resistors must be provided at both ends of the CAN bus.
- The max. baud rate must e tailored to the length of the bus line (see table below).
- Do not lay CAN lines in immediate proximity of fault causes, i.e. at a distance from motor cables.

If a servo-controller is located at the end of the bus, then the integrated load resistor in the servo can be hooked up. Use the double DIP – switch under the plastic cover to do this. Switch 2 must be in the ON position. There is a detailed connection description in the document "BU1100_DE Hardwaredescription.pdf".

Attention: Switch off the device's supply voltage, before opening the cover.

Bit – Rate	Bus length
1 Mbit/s	25 m
800 kbit/s	50 m
500 kbit/s	100 m
250 kbit/s	250 m
125 kbit/s	500 m
50 kbit/s	1000 m

Achievable line lengths, dependent upon transfer speeds selected.

7 The RS232 and RS485 Bus protocol

The protocols for the operation of the RS232 and RS485 interfaces are identical. For this reason, the following explanation applies to both standards. The controller can be parameterised and controlled via these interfaces. These interfaces can be accessed simultaneously by operating the CAN bus. If the servo-controller is operated via the RS485 bus, then all devices must have different bus addresses. Setting is explained in the "CAN – bus" section. The status machine and servo-controller parameters have already been explained in the CANopen– protocol" section.

Number transfer takes places in Intel number format, i.e. first the LOW and then the HIGH portion of a number are transferred.

Byte 0	Byte 1
11h	22h

The 16Bit number 2211h displayed in Intel format.

Byte 0	Byte 1	Byte 2	Byte 3
01h	23h	45h	67h

The 32Bit number 76543210h displayed in Intel format.

7.1 Communication Model

The communication model used by the servo-controller meets the client / server model. The servo-controller only communicates when requested by the Master. Each message from the Master is answered by the servo-controller. If the Master has sent a message to the servo-controller, then it must wait until the servo-controller answers. If it sends successive, multiple messages, without waiting for an answer, only the first will be processed and all the rest will be ignored.

7.2 Telegram Construction

A telegram type with a flexible length is used for data transfer.

Byte 0	Byte 1	Byte 2	Byte 3 to …..	Byte ....
STX	LGE	ADR	Data	BCC

STX :
This is the start sign, it always has a value of 02 and enables the start of a telegram to be recognised.

LGE :
The telegram length provides the length of the message in Bytes. Byte 2 to Bcc are counted.

ADR :
The 8 Bit address of the recipient is entered here. It is possible to assign addresses for servo-controllers in the range from 0 to 15, see "CAN – Bus" section.
The address FFh (255) is occupied by the Master, i.e. all servo-controllers queries are answered with this address.
The address FEh (254) is reserved for RS232 operation. If the servo-controller receives a message with this address, it reacts without checking the address set at the servo-controller. If this address is used at the RS485 Bus, with multiple servo-controllers, all devices will be replied to and a transfer fault will occur.

BCC :
The checksum is located in the last Byte of the Telegram. It is generated according to the following formula:

$$BCC_{new} = BCC_{old} \textbf{ XOR } \text{Telegram byte}_n \quad / \; n = 0 \text{ to } (LGE + 1)$$

Example ("Save current parameter set in non-volatile memory" command)

Byte 0	Byte 1	Byte 2	Byte 3	Byte4
STX	LGE	ADR	Data	BCC
02	03	00	22h	23h

7.2.1 Data Block Construction

The length of the data block is variable and dependent on the respective communication command. The first byte of the data block is always reserved for this command, see following table. The number of subsequent bytes is dependent upon the respective command.

7.2.1.1 Master Commands

Command Index	Command	Description
10h	Write Data	Write a parameter in the servo-controller. Send format: <Commandindex><Parameterindex><Parameterdata> Servo-controller response: <Returnindex>
12h	Read Data	Reads a parameter from the servo-controller. Send format: <Commandindex><Parameterindex> Servo-controller response: <12h><Parameterdata>
20h	Error	General receipt error, i.e. the recipient received the last message with an incorrect checksum. If the servo-controller receives this command, then it will repeat its last message. Send format: <Commandindex> Servo-controller response: The faulty message is resent.
22h	Store	With this command, the servo-controller will store the parameter set in non-volatile memory. This process takes approximately 3 seconds and only after this time, will the servo-controller respond. Send format: <Commandindex> Servo-controller response: <Returnindex>

7.2.1.2 Servo-controller Responses

Return Index	Response	Description
50h	OK	Used to acknowledge messages, which require no response.
52h	Not Used	Used by the controller as a response to a sent message, which contains commands or parameters, which are not supported by the controller.
54h	Save Error	An error occurred when saving the parameter set. This error is reported, when the values saved are compared with the active parameter set and a discrepancy is found.
56h	Read Only	An attempt was made to overwrite a "read only" parameter.
5Ch	Value Out of Range	The value range was exceeded by the last "Set Data" command.

7.2.2 Report Readouts

Reading out faults via the RS232/485 interfaces takes place via Index 2220h. 4 fault values are read out per 32Bits.

Byte 0	Byte 1	Byte 2	Byte 3
Fault Index		Status byte	

Fault value display

Fault number:
- Identifies fault, see "Fault Reports", under "CANopen Protocol".
- If fault number = 0, then there is no fault in this and the following fault words.

Status byte:
- Bit 0 : if = 1, then the servo-controller's I*t – function was active during the fault. This means that only the rated current was available to the servo-controller, which can be concluded as being the consequence of a speed or tracking error. In order to resolve this fault, the work cycles, which are in the overcurrent area, must be minimised.

Example of failure memory call

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
STX	LGE	ADR	Read Data	Index	Index	BCC
02	05	00	12h	20h	22h	17h

Master requirement

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 to Byte 19	Byte 20
STX	LGE	ADR	Read Data	Fault number 1 to 4	BCC
02	13h	00	12h	00 20 00 00 00 00 00 00 ...	...

Servo-controller's response (servo-controller reports only one fault, Fault index = 2000h = current fault).

7.2.3 Example

Sending the Parameter Mode 6060h to the Servo- controller

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
STX	LGE	ADR	Write Data	Index – Low	Index – High	Parameters	BCC
02	06	Feh	10h	60h	60h	01h	EBh

Message from Master to a servo-controller in RS232 Mode (address FEh)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
STX	LGE	ADR	OK	BCC
02	03	FFh	50h	AEh

Servo-controller's response

Loading status word 6041h from servo-controller

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
STX	LGE	ADR	Read Data	Index – Low	Index – High	BCC
02	05	00h	12h	41h	60h	34h

Message from Master to a servo-controller with address 00h.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
STX	LGE	ADR	Write Data	Data – Low	Data – High	BCC
02	05	FFh	12h	21h	00h	CBh

Servo-controller's response (Status word = 21h = Status "Ready to Switch On“)

8 Subsidiaries and Branches

N O R D G e a r branches, in Germany:

Northern Branch

NORD Gear GmbH & Co. KG
Rudolf-Diesel-Str. 1
22941 Bargteheide

Tel. 04532 / 401 – 0
Fax 04532 / 401 - 253

Western Branch

NORD Gear GmbH & Co. KG
Großenbaumerweg 10
40472 Düsseldorf

Tel. 0211 / 99 555 – 0
Fax 0211 / 99 555 - 45

Southern Branch

NORD Gear GmbH & Co. KG
Katharinenstr. 2-6
70794 Filderstadt- Sielmingen

Tel. 07158 95608 – 0
Fax 0715895608 - 20

Eastern Branch

NORD Gear GmbH & Co. KG
Leipzigerstr. 58
09113 Chemnitz

Tel. 0371 / 33 407 – 0
Fax 0371 / 33 407 - 20

Sales Office **Bremen**

NORD Gear GmbH & Co. KG
Wohlers Feld 16
27211 Bassum

Tel. 04249 / 9309 – 0
Fax 04249 / 9309 - 30

Sales Office **Butzbach**

NORD Gear GmbH & Co. KG
Marie-Curie-Str. 2
35510 Butzbach

Tel. 06033 / 9623 – 0
Fax 06033 / 9623 - 30

Sales Office **Nürnberg**

NORD Gear GmbH & Co. KG
Schillerstr. 3
90547 Stein

Tel. 0911 / 67 23 11
Fax 0911 / 67 24 71

Sales Office **Berlin**

NORD Gear GmbH & Co. KG
Roedernstr. 8
12549 Berlin

Tel. 030 / 639 79 413
Fax 030 / 639 79 414

Sales Office **Munich**

NORD Gear GmbH & Co. KG
Untere Bahnhofstr. 38a
82110 Germering

Tel. 089 / 840 794 – 0
Fax 089 / 840 794 - 20

Representative:
Hans-Hermann Wohlers
Handelsgesellschaft mbH
Ellerbuscher Str. 179

32584 Löhne

Tel. 05732 / 4072
Fax 05732 / 123 18

Head Office Germany:

NORD Gear GmbH & Co. KG
Rudolf- Diesel- Straße 1
D – 22941 Bargteheide

Tel. +49 / (0) 4532 / 401 – 0
Fax +49 / (0) 4532 / 401 – 253

Info@nord-de.com

http://www.nord.com

NORD Gear subsidiaries, worldwide:

Austria

NORD Gear GmbH
Deggendorfstr. 8
A – 4030 Linz

Tel.: +43-732-318 920
Fax: +43-732-318 920 85

Info@nord-at.com

Canada

NORD Gear Limited
41, West Drive
CDN - Brampton, Ontario, L6T 4A1

Tel.: +1-905-796-3606
Fax: +1-905-796-8130

Info@nord-ca.com

Denmark

NORD Gear Danmark A/S
Kliplev Erhvervspark 28 – Kliplev
DK – 6200 Aabenraa

Tel.: +45 73 68 78 00
Fax: +45 73 68 78 10

info@nord-dk.com

Great Britain

NORD Gear Limited
11, Barton Lane
Abingdon Science Park
GB – Abingdon, Oxfordshire OX 14 3NB

Tel.: +44-1235-534404
Fax: +44-1235-534414

info@nord-uk.com

Italy

NORD Motoriduttori s.r.l.
Via Modena 14
I – 40019 Sant' Agata Bolognese (BO)

Tel.: +39-051-6829711
Fax: +39-051-957990

info@nord-it.com

P.R. China

NORD (Beijing) Power Transmission Co.Ltd.
No.5 Tangjiacun,

Guangqudonglu, Chaoyangqu
Beijing 100022

Tel.: +86-10-67704-069(-787)
Fax: +86-10-67704-330

Fpan@nord-cn.com

Slovakia

NORD Pohony, s.r.o
Stromová 13
SK – 83101 Bratislava

Tel.: +42-7-54791317
Fax: +42-7-54791402

info@nord-sl.com

Switzerland

NORD Gear AG
Bächigenstr. 18
CH - 9212 Arnegg

Tel.: +41-71-388 99 11
Fax: +41-71-388 99 15

info@nord-ch.com

Belgium

NORD Aandrijvingen Belgie N.V.
Boutersem Dreef 24
B - 2240 Zandhoven

Tel.: +32-3-4845 921

Fax: +32-3-4845 924

Info@nord-be.com

Croatia

NORD Pogoni d.o.o.
Obrtnicka 9
HR - 48260 Krizevci

Tel.: +385-48 711 900
Fax: +385-48 711 900

Finland

NORD Gear Oy
Aunankorvenkatu 7
FIN - 33840 Tampere

Tel.: +358-3-254 1800
Fax: +358-3-254 1820

info@nord-fi.com

Hungary

NORD Hajtastechnika Kft.
Törökkö u. 5-7
H - 1037 Budapest

Tel.: +36-1-437-0127
Fax: +36-1-250-5549

info@nord-hg.com

Netherlands

NORD Aandrijvingen Nederland B.V.
Voltstraat 12
NL - 2181 HA Hillegom

Tel.: +31-2525-29544
Fax: +31-2525-22222

info@nord-nl.com

Poland

NORD Napedy Sp. z.o.o.
Ul. Grottgera 30
PL – 32-020 Wieliczka

Tel.: +48-12-288 22 55
Fax: +48-12-288 22 56

biuro@nord.pl

Spain

NORD Motorreductores
Ctra. de Sabadell a Prats de Llucanès
Aptdo. de Correos 166
E - 08200 Sabadell

Tel.: +34-93-7235322
Fax: +34-93-7233147

info@nord-es.com

Turkey

NORD-Remas Redüktör San. ve Tic. Ltd. Sti.
Tepeören Köyü
TR - 81700 Tuzla – Istandbul

Tel.: +90-216-304 13 60
Fax: +90-216-304 13 69

info@nord-tr.com

Brazil

NORD Motoredutores do Brasil Ltda.
Rua Elias Gannam, 83
CEP:02552 - 040 São Paulo SP

Tel.: +55-11-3951 5855
Fax: +55-11-3955 2144

info@nord-br.com

Czech. Republic

NORD Poháněci Technika s.r.o
Ulrichovo námesti 854
CZ - 50002 Hradec Králové

Tel.: +420-49 521 02 95
Fax: +420-49 521 06 91

info@nord-cz.com

France

NORD Réducteurs sarl.
17-19 Avenue Georges Clémenceau
F - 93421 Villepinte Cedex

Tel.: +33-1-49 63 01 89
Fax: +33-1-49 63 08 11

info@nord-fr.com

Indonesia

PT NORD Indonesia
Jln. Raya Serpong KM. 7
Kompleks Rumah Multi Guna Blok D No. 1
Pakulonan (Serpong) - Tangerang
West Java - Indonesia

Tel.: +62-21-5312 2222
Fax: +62-21-5312 2288
info@nord-hg.com

Norway

NORD Gear Norge A/S
Vestre Haugen 21
N - 1054 Furuset / Oslo

Tel.: +47-23 33 90 10
Fax: +47-23 33 90 15

info@nord-no.com

Singapore

NORD Gear Pte. Ltd.
33 Kian Teck Drive, Jurong
Singapore 628850

Tel.: +65-265 9118
Fax: +65-265 6841

info@nord-sg.com

Sweden

NORD Drivsystem AB
Ryttargatan 277 / Box 2097
S - 19402 Upplands Väsby

Tel.: +46-8-594 114 00
Fax: +46-8-594 114 14

info@nord-se.com

United States / USA

NORD Gear Corporation
800 Nord Drive / P.O. Box 367
USA - Waunakee, WI 53597-0367

Tel.: +1-608-849 7300
Fax: +1-608-849 7367

info@nord-us.com