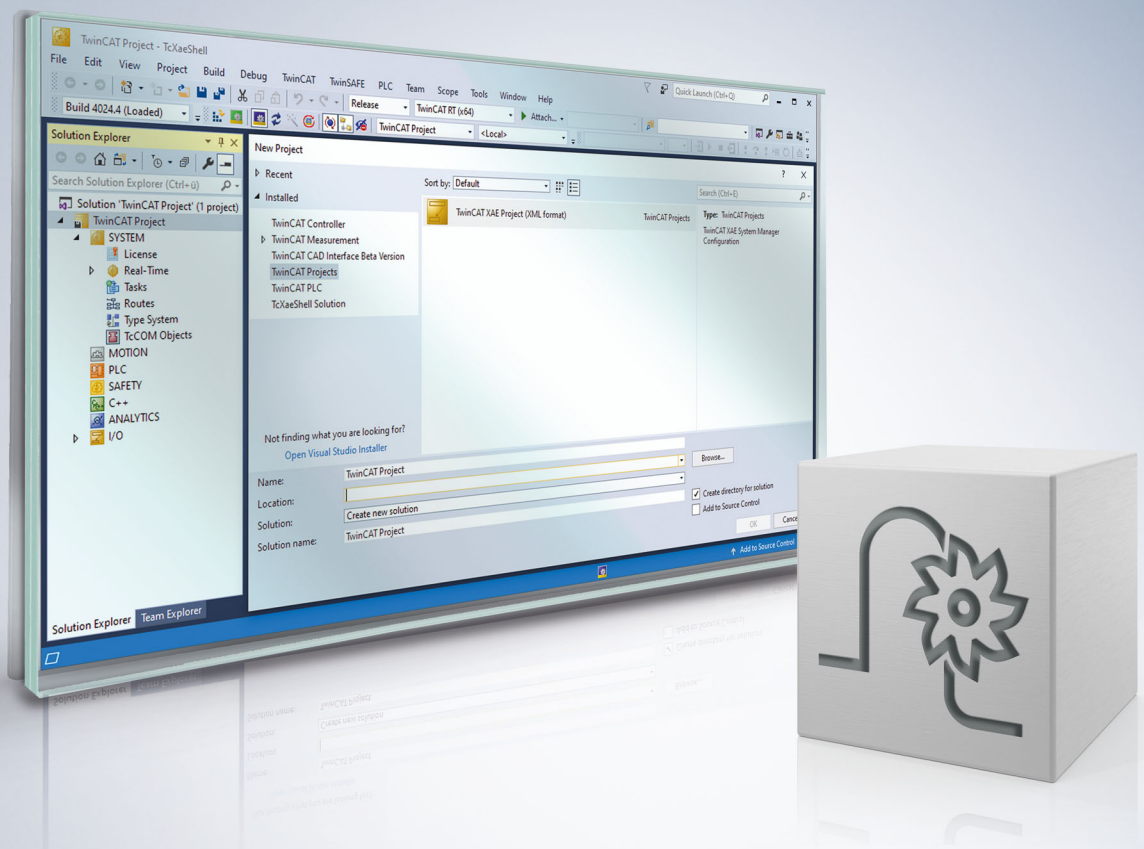
BECKHOFF New Automation Technology

Manual | EN

TF5200 | TwinCAT 3 CNC

Manual mode parameter



6/5/2025 | Version: 1.17

Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT technology is patent protected, in particular by the following applications and patents:
EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
with corresponding applications or registrations in various other countries.

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

General and safety instructions

Icons used and their meanings

This documentation uses the following icons next to the safety instruction and the associated text. Please read the (safety) instructions carefully and comply with them at all times.

Icons in explanatory text

1. Indicates an action.

⇨ Indicates an action statement.

⚠ DANGER

Acute danger to life!

If you fail to comply with the safety instruction next to this icon, there is immediate danger to human life and health.

⚠ CAUTION

Personal injury and damage to machines!

If you fail to comply with the safety instruction next to this icon, it may result in personal injury or damage to machines.

NOTICE

Restriction or error

This icon describes restrictions or warns of errors.

Tips and other notes

This icon indicates information to assist in general understanding or to provide additional information.

General example

Example that clarifies the text.

NC programming example

Programming example (complete NC program or program sequence) of the described function or NC command.

Specific version information

Optional or restricted function. The availability of this function depends on the configuration and the scope of the version.

Table of contents

Overview of manual mode parameter

The overview of real-time parameters is sorted into a 4-column table.

- Column 1 contains the unambiguous identifier of the real-time parameter called the “ID” which consists of the prefix “P-MANU” and a unique 5-digit number,
  e.g. P-MANU-00002.
- Column 2 represents the data structure which defines the parameter,
  e.g. hr_data[i].
  The structure is a categorisation aid and is described in the following section. If an entry is missing in ‘structure’, this is not an error. The parameter in column 3 is then only valid on its own.
- Column 3 contains the “parameter” with its exact name,
  e.g. log_hr_nr
  The important thing is that “structure”+”parameter” always belong together and must be configured in exactly the same way in the manual mode parameter list,
  e.g. hr_data[i].log_hr_nr
- Column 4 contains the “functionality” in a summarised term/short description,
  e.g. Logical handwheel number.

ID	Structure	Parameter	Functionality/short description
P-MANU-00001 [▶ 11]		hr_anzahl	Number of physical handwheels
P-MANU-00002 [▶ 12]	hr_data[i].	log_hr_nr	Logical handwheel number
P-MANU-00003 [▶ 12]	hr_data[i].	hr_zaehler_direct_access	Access to handwheel counter address
P-MANU-00004 [▶ 13]	hr_data[i].	adr_hr_zaehler	Handwheel counter address
P-MANU-00005 [▶ 14]	hr_data[i].	hr_aufl_z	Handwheel resolution numerator
P-MANU-00006 [▶ 14]	hr_data[i].	hr_aufl_n	Handwheel resolution denominator
P-MANU-00007 [▶ 14]	hr_data[i].	max_inkr_pro_takt	Upper limit for handwheel increments
P-MANU-00008 [▶ 16]		tasten_anzahl	Number of inching key pairs
P-MANU-00009 [▶ 16]	tasten_data[i].	log_tasten_nr	Logical key number
P-MANU-00010 [▶ 17]	feedhold	hr_inkr_verwerfen	Handwheel mode
P-MANU-00011 [▶ 17]	feedhold	.jog_inkr_verwerfen	Jogging mode
P-MANU-00012 [▶ 15]	hr_data[i].	hr_abs_pos_init	Offset on activation
P-MANU-00013 [▶ 15]	hr_data[i].	mode	Processing handwheel increments
P-MANU-00014 [▶ 17]		move_limit_warning	Output a message at offset limit

1 General description

1.1 Links to other documents

For the sake of clarity, links to other documents and parameters are abbreviated, e.g. [PROG] for the Programming Manual or P-AXIS-00001 for an axis parameter.

For technical reasons these links only function in the Online Help (HTML5, CHM) but not in pdf files since pdfs do not support cross-linking.

1.2 Structure and classification of manual mode parameters

The parameters of the **M**achine**D**ata**B**lock contain configuration data for manual mode.

The list assigns hardware information of physical handwheels and inching keys required for manual mode to logical manual mode elements. For example, the hardware addresses of physical handwheels are assigned to logical handwheels. It also defines the number of handwheels and inching key pairs and general configuration data for manual mode.

The parameters are structured as follows:

- Number of configured handwheels
- Parameterisation data of individual handwheels
- Number of configured inching keys
- Parameterisation data of individual inching keys
- Processing manual mode motion paths during a feedhold

The field indices to be specified in the list correspond to the internal manual mode indexing of axis and manual mode elements.

The value ranges of parameters are also defined if necessary by specifying a limit arising from the data format, e.g. MAX(UNS32) etc.

1.3 Syntax and interpretation of ASCII list file

An interpreter copies the entries in the ASCII list file into identical internal structures which are then checked for plausibility. To ensure reliable controller start-up every time, any defective entries found by the plausibility check are replaced by default values.

Unknown entries are not taken over. These irregularities are displayed by warning messages. We advise you to investigate the cause for these warning messages and remove defective entries from the ASCII list file.

The following agreement applies to BOOLEAN data:

Value	Meaning
0	Definition of FALSE
1	Definition of TRUE

The following agreement applies to STRING data:

If a character string containing characters with a special meaning in ASCII lists (e.g. comment characters, spaces [▶ 10]) is assigned to a STRING type list parameter, this character string must be defined in inverted commas "..." (available as of V3.1.3081.0, V3.1.3108.0).

```
example[0].name "STRING_WITH_COMMENT( # /*)_CHARACTERS"
```

Trailing spaces are discarded on import. The entry..

```
example[0].name "STRING_WITH_POST_SPACES "
```

..has the same meaning as

```
example[0].name "STRING_WITH_POST_SPACES"
```

If the character string only contains characters without any special meaning, no inverted commas are required.

```
example[0].name STRING_WITH_STANDARD_CHARACTERS!
```

1.4 Comments in the ASCII list file

Comments can be in an entire line or can be added at the end of a line.

With a comment spanning an entire line, the comment character "'#" must be placed at the start of the line and followed by a blank.

If a comment is to be inserted at the end of a line, only a blank is required before the comment. Blank lines are also possible.

Comments in the ASCII list file

```
# ************************************************************************
# Data
# ************************************************************************
#
# List comments after numerical values

dummy[1] 1 Comment
dummy[2] 1 # Comment
dummy[3] 1 ( Comment
dummy[4] 1 /* Comment
...
...
```

However, if a character string was assigned to the list parameter as a value in the line, any following comment must be opened by a bracket '('. The comment characters space, # and /* are not permitted.

If a '(' itself is part of the character string, the character string must be defined in inverted commas ″..″ (available as of V3.1.3081.0, V3.1.3108.0).

```
# List comments after strings

beispiel[0].bezeichnung STRING_1 (comment requires a '(' bracket!)

beispiel[1].bezeichnung″ STRING_(2)″ (comment requires a '(' bracket!)
```

2 Description of elements

2.1 Settings for handwheels

2.1.1 Number of physical handwheels (P-MANU-00001)

P-MANU-00001	Number of physical handwheels
Description	This parameter defines the handwheel number.
Parameter	hr_anzahl
Data type	UNS16
Data range	0 ... 6 (maximum number of handwheels, application-specific)
Dimension	----
Default value	0
Remarks	The number of handwheels must correspond to the handwheels assigned in the handwheel data list 'hr_data[i]'. Parameterisation example: Two physical handwheels are to be configured. *hr_anzahl 2*

2.1.2 Handwheel data (hr_data[i].*)

The configuration data for existing handwheels are stored in the structure 'hr_data[i]'.

Structure name	Index
hr_data[i]	i = 0 ... 5 (number of handwheels: 6, application-specific)

2.1.2.1 Logical handwheel number (P-MANU-00002)

P-MANU-00002	Logical handwheel number
Description	This parameter defines the logical handwheel number of a handwheel. The logical handwheel number is unique throughout the entire system. The logical handwheel number is used to assign activation and deactivation jobs by software to a physical handwheel. The logical handwheel number is also used to assign a physical handwheel to a logical axis.
Parameter	hr_data[i].log_hr_nr
Data type	UNS16
Data range	1 ... MAX(UNS16)
Dimension	----
Default value	0
Remarks	A logical handwheel number may not be assigned several times. If this is the case, the data plausibility check generates an error message. The logical handwheel number ‘0’ is not permitted. Activating and deactivating manual modes is described in [PROG]. Parameterisation example: Two physical handwheels are configured. They are identified by logical handwheel numbers ‘1’ and ‘2’. *hr_data[0].log_hr_nr 1 #logical handwheel number Handwheel 1* *.....* *hr_data[1].log_hr_nr 2 #logical handwheel number Handwheel 2* *.....*

2.1.2.2 Access to handwheel counter address (P-MANU-00003)

P-MANU-00003	Access to handwheel counter address
Description	This parameter defines whether the handwheel increments are read in from a counter module or via the High Level Interface (HLI).
Parameter	hr_data[i].hr_zaehler_direct_access
Data type	BOOLEAN
Data range	0: Handwheel counter address is located on High Level Interface (HLI). Always with operating systems with virtual address space (e.g. RTWin, NT). 1: Handwheel counter address is located on the hardware (HW). At present, only implemented for operating systems with absolute address space (e.g. VxWorks, OS9).
Dimension	----
Default value	0
Remarks	Parameterisation example: Handwheel increments are read directly by a counter module. *hr_data[0].hr_zaehler_direct_access 1 #Direct hardware access Handwheel1.*

2.1.2.3 Handwheel counter address (P-MANU-00004)

P-MANU-00004	Handwheel counter address
Description	Depending on the element P-MANU-00003 [▶ 12] the parameter here defines either the index in the HLI array to read in the increments or the physical handwheel counter address.
Parameter	hr_data[i].adr_hr_zaehler
Data type	UNS16
Data range	0 ... 5: Array index in the High Level Interface (HLI) where P-MANU-00003 [▶ 12] = 0 >0: Physical address of a counter module where P-MANU-00003 [▶ 12] = 1
Dimension	----
Default value	0
Remarks	Parameterisation example 1: Increments for handwheels ‘1’ and ‘2’ are read in via the HLI interface. *hr_data[0].log_hr_nr 1 #logical handwheel number Handwheel 1* *hr_data[0].hr_zaehler_direct_access 0 #HLI access Handwheel 1* *hr_data[0].adr_hr_zaehler 0 #Index in HLI array Handwheel 1* *…….* *hr_data[1].log_hr_nr 2 #logical handwheel number Handwheel 2* *hr_data[0].hr_zaehler_direct_access 0 #HLI access Handwheel 2* *hr_data[1].adr_hr_zaehler 1 #Index in HLI array Handwheel 2* Parameterisation example 2: Increments for handwheels ‘1’ and ‘2’ are read in by direct hardware access. *hr_data[0].log_hr_nr 1 #logical handwheel number Handwheel 1* *hr_data[0].hr_zaehler_direct_access 1 #Hardware access Handwheel 1* *hr_data[0].adr_hr_zaehler 0x0030 #Address Handwheel 1* …. *hr_data[1].log_hr_nr 2 #logical handwheel number Handwheel 2* *hr_data[1].hr_zaehler_direct_access 1 #Hardware access Handwheel 1* *hr_data[1].adr_hr_zaehler 0x0080 #Address Handwheel 2*

2.1.2.4 Handwheel resolution

The number of handwheel increments per handwheel revolution is entered as the handwheel resolution. This parameter must be specified by a numerator and a denominator.

NOTICE

Handwheel resolution = numerator / denominator = hr_aufl_z/ hr_aufl_n

Handwheel resolution unit: [increments/revolution]

Do not confuse this handwheel-specific resolution with the axis-specific resolution for handwheel mode [PROG].

2.1.2.4.1 Handwheel resolution numerator (P-MANU-00005)

P-MANU-00005	Handwheel resolution numerator
Description	This parameter defines the handwheel resolution numerator.
Parameter	hr_data[i].hr_aufl_z
Data type	UNS32
Data range	1 ... MAX(UNS32)
Dimension	increments
Default value	0
Remarks	

2.1.2.4.2 Handwheel resolution denominator (P-MANU-00006).

P-MANU-00006	Handwheel resolution denominator
Description	The parameter defines the handwheel resolution denominator.
Parameter	hr_data[i].hr_aufl_n
Data type	UNS32
Data range	1 ... MAX(UNS32)
Dimension	U
Default value	0
Remarks	Parameterisation example: Resolution 100 increments/resolution applies to handwheel ‘1’ and resolution 2700 increments/14 revolutions to handwheel ‘2’. *hr_data[0].log_hr_nr 1 #logical handwheel number Handwheel 1* *hr_data[0].hr_aufl_z 100 #Handwheel resolution numerator Handwheel 1* *hr_data[0].hr_aufl_n 1 # Handwheel resolution denominator Handwheel 1* *…….* *hr_data[1].log_hr_nr 2 #logical handwheel number Handwheel 2* *hr_data[1].hr_aufl_z 2700 #Handwheel resolution numerator Handwheel 2* *hr_data[1].hr_aufl_n 14 #Handwheel resolution denominator Handwheel 2*

2.1.2.5 Upper limit for handwheel increments (P-MANU-00007)

P-MANU-00007	Upper limit for handwheel increments
Description	This parameter defines the upper limit for the number of handwheel increments read in per cycle. It is a plausibility limit for the handwheel hardware.
Parameter	hr_data[i].max_inkr_pro_takt
Data type	UNS32
Data range	< MAX(UNS32)
Dimension	increments

Default value	0
Remarks	Parameterisation example: The maximum number of handwheel increments read in per cycle is 1000. *hr_data[0].max_inkr_pro_takt 1000 #Upper limit for handwheel increments*

2.1.2.6 Offset on activation (P-MANU-00012)

P-MANU-00012	**Offset on activation**
Description	This parameter defines whether adoption on the handwheel counter status is absolute or relative when handwheel mode is activated.
Parameter	hr_data[i].hr_abs_pos_init
Data type	BOOLEAN
Data range	0: Relative adoption of handwheel counter status when handwheel mode is activated. 1: Absolute adoption of handwheel counter status when handwheel mode is activated.
Dimension	----
Default value	0
Remarks	Relative mode is practical for typical handwheel mode, i.e. there is no path motion, even if the handwheel counter status is not 0 when handwheel mode is activated.

2.1.2.7 Processing handwheel increments (P-MANU-00013)

P-MANU-00013	**Processing handwheel increments**
Description	This parameter influences the behaviour of handwheel mode when increments (path) are processed.
Parameter	hr_data[i].mode
Data type	UNS32
Data range	0: True-path handwheel; all increments generated are carried out. Axis motion may run on depending on the dynamic setting 1: Non true-path handwheel; all generated handwheel increments are cancelled after the handwheel rotation stops (see inching key mode). Axis motion also stops immediately. 2: Conditional true-path handwheel, generated handwheel increments are cancelled if the command velocity of the handwheel was reached before handwheel rotation stops. Note: The maximum velocity of the handwheel (P-AXIS-00213) is weighted with feed override depending on P-CHAN-00186 . (Mode available as of V3.1.3074)
Dimension	----
Default value	0
Remarks	

2.2 Inching key settings

2.2.1 Number of inching key pairs (P-MANU-00008)

P-MANU-00008	Number of inching key pairs
Description	This parameter defines the number of inching key pairs.
Parameter	tasten_anzahl
Data type	UNS16
Data range	0 ... 9 (maximum number of inching key pairs, application-specific)
Dimension	----
Default value	0
Remarks	The number of inching keys must correspond to the elements assigned in the inching key data list 'tasten_data[i]'. Parameterisation example: Two inching key pairs are configured. *tasten_anzahl 2*

2.2.2 Inching key data (tasten_data[i].*)

The structure 'tasten_data[i]' defines the configuration data of existing inching key pairs.

Structure name	Index
tasten_data[i]	i = 0 ... 8 (Number of inching key pairs: 9, application-specific)

2.2.2.1 Logical key number (P-MANU-00009)

P-MANU-00009	Logical key number
Description	This parameter defines the logical key number of an inching key pair. The logical key number is unique throughout the entire system. The logical key number is used to assign an activation or deactivation job to the physical inching key pair. The logical key number is also used to assign a physical key pair to a logical axis.
Parameter	tasten_data[i].log_tasten_nr
Data type	UNS16
Data range	1 ... MAX(UNS16)
Dimension	----
Default value	0
Remarks	A logical key number may not be assigned several times. If this is the case, the plausibility check of the manual mode data generates an error message. The logical key number ‘0’ is not permitted. Activating and deactivating manual modes is described in [PROG]. Parameterisation example: Two physical inching key pairs are configured. These are the logical key numbers ‘1’ and ‘2’. *tasten_data[0].log_tasten_nr 1 #logical key number Key pair 1* *…….* *tasten_data[1].log_tasten_nr 2 #logical key number Key pair 2* *…….*

2.3 Processing feedhold (feedhold.*)

The ‘feedhold’ structure defines the processing of read-in handwheel increments and jogging mode motion paths during a feedhold.

2.3.1 Handwheel mode (P-MANU-00010)

P-MANU-00010	Handwheel mode
Description	If the parameter is set to TRUE, the handwheel increments read in during a feedhold are skipped. Otherwise, these handwheel increments are output after feedhold is removed.
Parameter	feedhold.hr_inkr_verwerfen
Data type	BOOLEAN
Data range	0/1
Dimension	----
Default value	0
Remarks	Parameterisation example: Handwheel increments read in during a feedhold should not be skipped. *feedhold.hr_inkr_verwerfen 0 #Skip handwheel increments*

2.3.2 Jogging mode (P-MANU-00011)

P-MANU-00011	Jogging mode
Description	If the parameter is set to TRUE, jogging mode motion paths generated during feedhold are skipped. Otherwise, these motion paths are output after feedhold is removed.
Parameter	feedhold.jog_inkr_verwerfen
Data type	BOOLEAN
Data range	0/1
Dimension	----
Default value	0
Remarks	Parameterisation example: Jogging mode path motions read in during a feedhold should be skipped. *feedhold.jog_inkr_verwerfen 1 #Skip jogging mode motion paths*

2.4 Messages at offset limits (P-MANU-00014)

P-MANU-00014	Output a message at offset limit
Description	If this parameter is set to TRUE, the CNC generates a warning if a manual movement stops at a relative offset limit (P-AXIS-00137, P-AXIS-00138) or at an absolute offset limit (P-AXIS-00492, P-AXIS-00493).
Parameter	move_limit_warning
Data type	BOOLEAN
Data range	0/1
Dimension	----
Default value	0
Remarks	This parameter is available as of CNC Build **2.11.2804.12**

3 Example of assigning manual mode parameters

```
# ****************************************************************************
# Manual mode data
# ****************************************************************************
# Important note : Behind the comment character ´#´
# a blank (space) must be added
# ****************************************************************************
# Number of handwheels
****************************************************************************
hr_anzahl 2
****************************************************************************
# Number of inching key pairs
****************************************************************************
tasten_anzahl 3
****************************************************************************
# Handwheel data
****************************************************************************
hr_data[0].log_hr_nr 1
hr_data[0].hr_aufl_z 100
hr_data[0].hr_aufl_n 1
hr_data[0].hr_zaehler_direct_access 1
hr_data[0].adr_hr_zaehler 0x8a0a0030
hr_data[0].max_inkr_pro_takt 1000
hr_data[0].hr_abs_pos_init 0
#
hr_data[1].log_hr_nr 2
hr_data[1].hr_aufl_z 180
hr_data[1].hr_aufl_n 1
hr_data[1].hr_zaehler_direct_access 1
hr_data[1].adr_hr_zaehler 0x8a0a0080
hr_data[1].max_inkr_pro_takt 1000
hr_data[1].hr_abs_pos_init 0

****************************************************************************
# Inching key data
****************************************************************************
tasten_data[0].log_tasten_nr 1
#
tasten_data[1].log_tasten_nr 2
#
tasten_data[2].log_tasten_nr 3
****************************************************************************
# Feedhold processing
****************************************************************************
feedhold.hr_inkr_verwerfen 0
feedhold.jog_inkr_verwerfen 1
```

4 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Download finder

Our download finder contains all the files that we offer you for downloading. You will find application reports, technical documentation, technical drawings, configuration files and much more.

The downloads are available in various formats.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on our internet page: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963-157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963-460
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963-0
e-mail:	info@beckhoff.com
web:	www.beckhoff.com

Index

P

More Information:
www.beckhoff.com/TF5200

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com