ProfiBus

Interface

for the OMC/TMC Controller

TRANSLATION OF THE GERMAN ORIGINAL MANUAL

Phytron GmbH

Industriestraße 12

82194 Gröbenzell, Germany

Tel.: +49(0)8142/503-0

Fax: +49(0)8142/503-190

Every possible care has been taken to ensure the accuracy of this technical manual. All information contained in this manual is correct to the best of our knowledge and belief but cannot be guaranteed. Furthermore we reserve the right to make improvements and enhancements to the manual and / or the devices described herein without prior notification.

We appreciate suggestions and criticisms for further improvement.

Email address: doku@phytron.de

Questions about the use of the product described in the manual that you cannot find answered here, please contact your representative of phytron (http://www.phytron.eu/) in your local agencies.

1 Legal information

This manual:

Read this manual very carefully before mounting, installing and operating the device and if necessary further manuals related to this manual.

- Please pay special attention to instructions that are marked as follows:

	**DANGER – Serious injury!**	*Indicates a high risk of serious injury or death!*
	**DANGER – Serious injury from electric shock!**	*Indicates a high risk of serious injury or death from electric shock!*
	**WARNING – Serious injury possible!**	*Indicates a possible risk of serious injury or death!*
	**WARNING – Serious injury from electric shock!**	*Indicates a possible risk of serious injury or death from electric shock!*
	**CAUTION – Possible injury!**	*Indicates a possible risk of personal injury.*
	**CAUTION – Possible damage!**	*Indicates a possible risk of damage to equipment.*
	**CAUTION – Possible damage due to ESD!**	*Refers to a possible risk of equipment damage from electrostatic discharge.*
	**”Any heading“**	*Refers to an important paragraph in the manual.*

Safety Instructions

CAUTION – Possible damage!

Malfunctions are possible while programming the instruction codes – e.g. sudden running of a connected motor, braking etc.

- Please test the program flow step by step.

CAUTION – Possible damage!

For each application, the functional reliability of software products by external factors such as voltage differences or hardware failure, etc. is affected.

- To prevent damage due to system error, the user should take appropriate safety measures. These include back-up and shut-down mechanisms.

CAUTION – Possible damage!

Each end user system is customised and differs from the testing platform. Therefore the user or application designer is responsible for verifying and validating the suitability of the application.

- The suitability of the device's use must be tested and validated.

CAUTION – Possible damage!

Some modules are set to a default value on delivery. So, e.g., the motor current must be set to the corresponding value (see the motor data from the motor manufacturer). Connected components like motors can be damaged by incorrectly set values.

- Please check before starting, if the parameters are correct.

2 Contents

3 Basic Information

MINILOG is a programming language to communicate with phytron's programmable controllers OMC/TMC and MCC series. *phy***LOGIC**™ is the programming language for the *phy***MOTION**™ controller. The MINILOG / *phy***LOGIC**™ commands can be embedded in other communication protocols (ProfiBus/ProfiNet, Ethernet,…).

Phytron provides software for the control.
The "MiniLog-Comm" software masters only the MINILOG instruction set, but the *phy***LOGIC**™ ToolBox can operate both in MINILOG and in the *phy***LOGIC**™ instruction set.

You can parameterise your commands (e.g. a run command) per axis either only the first time, when you install your system or adjust the parameters temporarily before you send a run command.

Example: For "relative running" you can set: step resolution (P45), run current (P41), run frequency (P14), Start/stop frequency (P04), ramp (P15), recovery time position (P16), Boost (P17), Boost current (P42), current delay time (P43), etc.

Each of our programmable controllers comes along with pre set parameters (default values), which are automatically loaded into the temporary memory of each axis while starting the device. These parameters can be changed during your program is executed to optimise your motion tasks at any time.

If you want your controller to wake up with a new set of parameters, you have to explicitly store them in the non volatile storage of the main CPU unit by using a certain command.

.

4 The UNIGATE CL-ProfiBusDP Module

4.1 Overview

The UNIGATE CL-ProfiBusDP component is used to adjust the serial interface of phytron‘s OMC/TMC stepper motor controller to the ProfiBusDP acc. to EN 50 170.

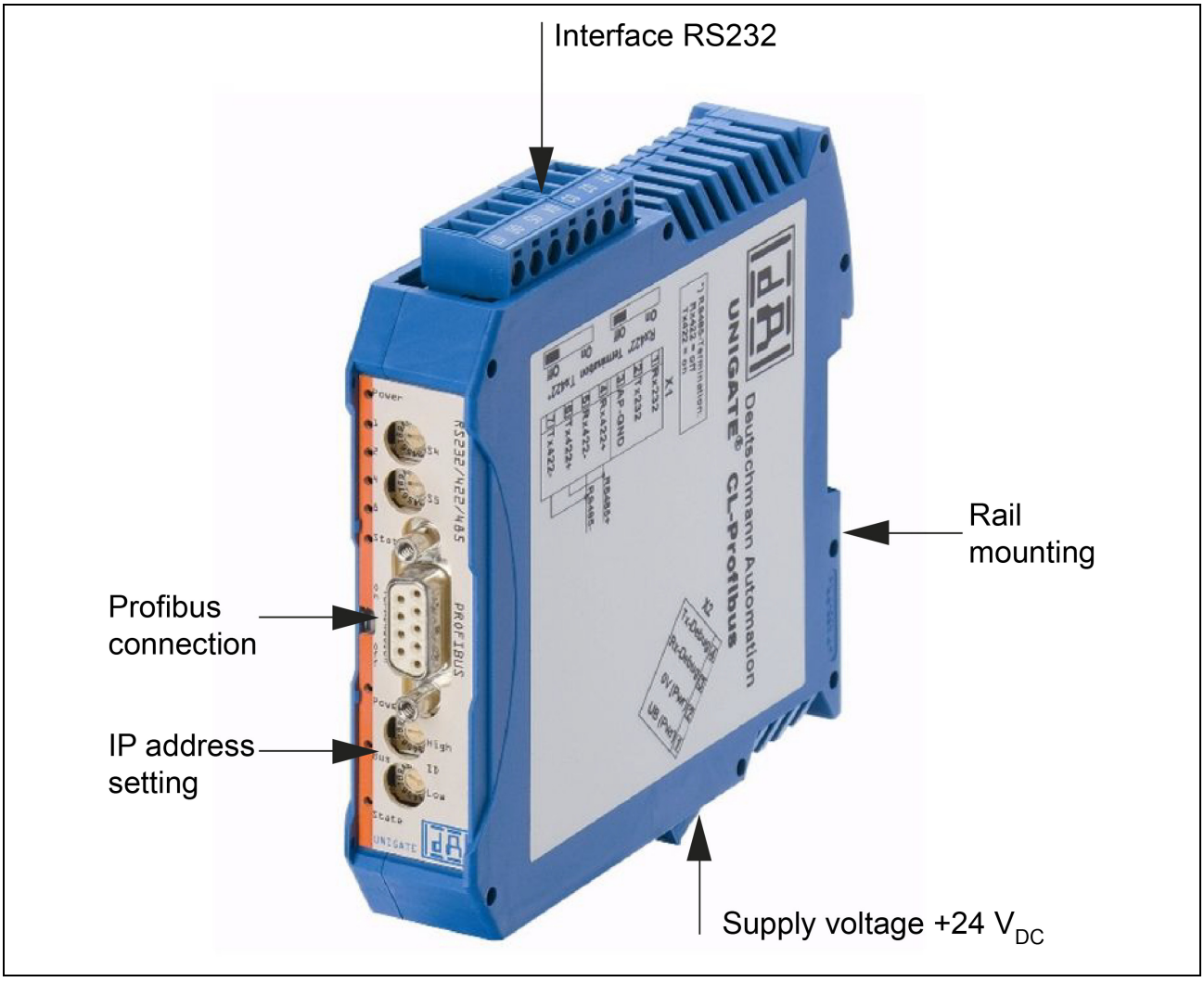


Fig.1 Overview of the UNIGATE CL-ProfiBusDP

You’ll need to connect the stepper motor controller to the ProfiBus via UNIGATE module:

- Cable from the OMC/TMC stepper motor controller (9-pole DSUB female at X31) to the UNIGATE module (free wires at X1). (ID no. 10013499)
- ProfiBus cable
- 24 V_{DC} power supply (DIN 19240)
- GSD file ‚UGIC3218.GSD‘

4.2 Commissioning of the UNIGATE Module

Perform the following steps during commissioning:

4.2.1 Connection of the OMC/TMC to the UNIGATE Module

Use the supplied cable (ID 10013499).

Plug the 9-pin DSUB connector into the X31 of the OMC/TMC and screw the wire ends into the terminals below the X1 connector of UNIGATE module:

Terminal	Wire	
01	brown	Rx232
02	white	Tx232
03	green	GND

Fig.2 X1 (UNIGATE module)

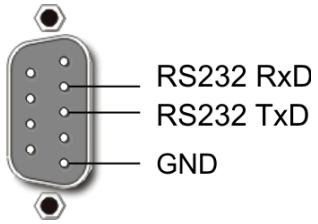


Fig.3 X31 (OMC/TMC)

4.2.2 Set the ProfiBus Address

Set the ProfiBus address in hexadecimal on the two rotary switches "ID High“ and “ID Low“.

Example:

The ProfiBus ID is 27 decimal = 1B hexadecimal. The “ID High“ switch must be set to 1 and the “ID Low“ switch to B.

The rotary switches can be set between 1 and 125.

4.2.3 Connecting the ProfiBus

Connect the PLC master unit to the ProfiBus by the interface named "PROFIBUS".

5 Conditions

- You have an S7 station, consisting of a power supply module and a CPU and configured.
- STEP 7 or TIA Portal is completely installed at your programmer (PG).
- You have knowledge of STEP 7 or TIA Portal.
- The PG is connected to the PROFINET I/O.
- The OMC/TMC controller is connected to the superior main station via ProfiBus module.

6 Configuration of the OMC/TMC via SIMATIC Manager (Example)

- Install the GSD data file ‚IC3218OM.GSD'on your PC.
- Start the SIMATIC manager and open the project that you created.
- Add the OMC/TMC device from the hardware catalogue (HW Config) with drag & drop.
- Drag the individual functional modules according to the addressing assembly (see chapter 8)
- Save and compile the hardware configuration with s*tation > save and compile*

Addressing

The addressing of the functional modules must be followed exactly to configure the controller!

- The order of the addressing is from low to high! (see chap.8)

7 Interface Assignment

In the ProfiBus system following sequence of the addressing must be maintained for the OMC/TMC controller:

Function	**Address**
Master	base address +0
Outputs	base address +8
Inputs	base address +16
Axis 1	base address +24
Axis 2*)	base address +32
Analogue Input	base address +40

*) not applicable for OMC; analogue input is addressed with base address +32.

Commands can be transmitted to any device function if they are addressed once only. In the following chapter the instruction set is defined for each function by referring to the MINILOG command reference manual.

Definition of the Data length		
Type	**Length**	**Sign**
Error code	byte	–
Program number	byte	–
Output	double word MSB--LSB	unsigned long
Input	double word MSB--LSB	unsigned long
Status	word MSB/LSB	unsigned long
Analogue value	word MSB/LSB	unsigned long
Register number	word MSB/LSB	unsigned long
Distance	double word MSB--LSB	data type
Position	double word MSB--LSB	data type
Parameter value	double word MSB--LSB	data type
Current position	double word MSB--LSB	only signed long
Register value	double word MSB--LSB	data type

Data type	unsigned long	0x00 send byte 2
	signed long	0x10 send byte 2
	float	0x40 send byte 2

7.1 Master

7.1.1 Data set (Commands) (8 bytes)

	Function of the commands (8 bytes)	Byte								According to MINILOG command (refer to the command reference manual for OMC/TMC)	
		1	2	3	4	5	6	7	8	command	chap.
Send	Read master status	00_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	ST	2.15
	Save controller's parameters (start)	02_H	01_H	00_H	00_H	00_H	00_H	00_H	00_H	SA	2.16
	Terminates the save of the controller's parameters[1)]	02_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	–	
	Start program script	03_H	01_H	00_H	00_H	00_H	00_H	00_H	XX_H[2)]	QPname A	2.13
	Stop program script	03_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	QPE	2.13
	Emergency stop of all axes and outputs are set to zero	04_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	–	
	Read register										
	signed long	40_H	10_H	register no.	00_H	00_H	00_H	00_H		RnnR	2.14
	float	40_H	40_H	register no.	00_H	00_H	00_H	00_H		RnnR	2.14
	Write register										
	signed long	41_H	10_H	register no.	register value					RnnSvalue	2.14
	float	41_H	40_H	register no.	register value					RnnSvalue	2.14

[1)] The command 0201_H sets the flag ‚Save parameters'. This flag must be reset before resaving the parameters. The command 0200_H resets this flag.

[2)] XX_H has to be replaced by the script number which should be started. It must be registered in hexadecimal code. E.g. code $0A_H$ starts script no. 10.apr .

Response	Controller Reset status	00_H	Error code	Status	00_H	00_H	00_H	00_H
	Read master status	01_H	Error code	Status	00_H	00_H	00_H	00_H
	Save controller's parameters	02_H	Error code	Status	00_H	00_H	00_H	00_H
	Program script	03_H	Error code	Status	00_H	00_H	00_H	00_H
	Read register							
	signed long	40_H	10_H	register no.	register value			
	float	40_H	40_H	register no.	register value			
	Write register							
	signed long	41_H	10_H	register no.	register value			
	float	41_H	40_H	register no.	register value			

7.1.2 Status Code (Master)

Status (2 bytes)	Meaning
0001_H	Parameters were changed
0002_H	Parameters were saved
0010_H	Axis 1 stopped
0020_H	Axis 1 reference point OK
0040_H	Axis 2 stopped
0080_H	Axis 2 reference point OK
0100_H	Internal program is running
0200_H	Forced switch over to remote via PC
0400_H	Limit switch of an axis is active
0800_H	Power stage error of an axis
1000_H	Programming error: internal program
4000_H	Input inquiry active (wait for input status)
8000_H	Remote/Local switch on Remote

7.1.3 Error Code (Master)

Error (1 byte)	Meaning
00_H	OK
01_H	Axis 1 power stage error
02_H	Axis 2 power stage error

7.2 Axes

	Function	Byte 1	2	3	4	5	6	7	8	According to MINILOG command (refer to the command reference manual for OMC/TMC) command	chap.
Send	Axis status Read	10_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	SE and XPmmR	2.15 2.20
	Axis instruction reset	11_H	00_H	Axis no.	00_H	00_H	00_H	00_H	00_H	–	–
	Axis stop										
	normal	12_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	XS	2.20
	with emergency stop ramp	12_H	01_H	00_H	00_H	00_H	00_H	00_H	00_H	XSN	2.20
	Reference run										
	– direction	13_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	X0–	2.20
	+ direction	13_H	01_H	00_H	00_H	00_H	00_H	00_H	00_H	X0+	2.20
	Free running										
	– direction	14_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	XL–	2.20
	+ direction	14_H	01_H	00_H	00_H	00_H	00_H	00_H	00_H	XL–	2.20
	Relative positioning with parameter										
	P14	15_H	10_H	00_H	00_H	distance, signed long				Xrzvalue	2.20
	P14	15_H	40_H	00_H	00_H	distance, float				Xrzvalue	2.20
	Absolute positioning with parameter										
	P14	16_H	10_H	00_H	00_H	Position, signed long				XArzvalue	2.20
	P14	16_H	40_H	00_H	00_H	Position, float				XArzvalue	2.20
	Deactivate axis	17_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	XMD	2.20

	Function	Byte 1	2	3	4	5	6	7	8	According to MINILOG command (refer to the command reference manual for OMC/TMC) command	chap.
Send	Activate axis	17_H	01_H	00_H	00_H	00_H	00_H	00_H	00_H	XMA	2.20
	Reset power stage	18_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	XC	2.20
	Write parameters										
	signed long	19_H	10_H	00_H	Parameter no.	Parameter value				XPmm Svalue	2.20
	float	19_H	40_H	Axis No.	Parameter no.	Parameter value				XPmm Svalue	2.20
	Read parameters										
	signed long	$1A_H$	10_H	00_H	Parameter no.	00_H	00_H	00_H	00_H	XPmmR	2.20
	float	$1A_H$	40_H	00_H	Parameter no.	00_H	00_H	00_H	00_H	XPmmR	2.20

	Function	Byte 1	2	3	4	5	6	7	8
Response	Read axis status	10_H	Error code	Status		Current position			
	Axis instruction Reset	11_H	Error code	Status		Current position			
	Axis stop	12_H	Error code	Status		Current position			
	Reference running	13_H	Error code	Status		Current position			
	Free running	14_H	Error code	Status		Current position			
	Relative Positioning	15_H	Error code	Status		Current position			
	Absolute Positioning	16_H	Error code	Status		Current position			
	Deactivate axis/ Activate axis	17_H	Error code	Status		Current position			
	Reset power stage	18_H	Error code	Status		Current position			
	Write parameters	19_H	Data type	00_H	Parameter no.	Parameter value			
	Read parameters	$1A_H$	Data type	00_H	Parameter no.	Parameter value			

7.2.1 Error Code (Axes)

Status (1 byte)	Meaning
00_H	Module OK
01_H	Power stage overcurrent
02_H	Power stage overvoltage
04_H	Power stage overtemperature
40_H	SFI Error
80_H	Encoder Error

7.2.2 Status Code (Axes)

Status (2 bytes)	Meaning
0001_H	Power stage overcurrent
0002_H	Power stage overvoltage
0004_H	Power stage overtemperature
0008_H	Power stage activated
0010_H	Initiator – active
0020_H	Initiator + active
0040_H	SFI Error
0080_H	Encoder Error
0100_H	Motor stops
0200_H	Reference point OK

7.3 Outputs

Function		Byte								According to MINILOG command (refer to the command reference manual for OMC/TMC)	
		1	2	3	4	5	6	7	8	command	chap.
Send	Read output status	20_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	**AG**1R	2.1
	Read and set output status	21_H	00_H	00_H	00_H	output				-	-
Response	Read output status	20_H	00_H	00_H	00_H	output					
	Read and set output status	21_H	00_H	00_H	00_H	output					

7.4 Inputs

Function		Byte								According to MINILOG command (refer to the command reference manual for OMC/TMC)	
		1	2	3	4	5	6	7	8	command	chap.
Send	Read input value	30_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	**EG**1R	2.5

Response	Read input status	30_H	00_H	00_H	00_H	input

7.5 Analogue Inputs

	Function	Byte								According to MINILOG command (refer to the command reference manual for OMC/TMC)	
		1	2	3	4	5	6	7	8	command	chap.
Send	Read analogue										
	channel 1	60_H	00_H	00_H	00_H	00_H	00_H	00_H	00_H	**ADnCxSy**	2.2
	channel 2	60_H	00_H	00_H	01_H	00_H	00_H	00_H	00_H	**ADnCxSy**	2.2
	channel 3	60_H	00_H	00_H	02_H	00_H	00_H	00_H	00_H	**ADnCxSy**	2.2
	channel 4	60_H	00_H	00_H	03_H	00_H	00_H	00_H	00_H	**ADnCxSy**	2.2

Response	Read analogue channel					
	Channel 1	60_H	00_H	00_H	00_H	Analogue value
	Channel 2	60_H	00_H	00_H	01_H	Analogue value
	Channel 3	60_H	00_H	00_H	02_H	Analogue value
	Channel 4	50_H	00_H	00_H	03_H	Analogue value

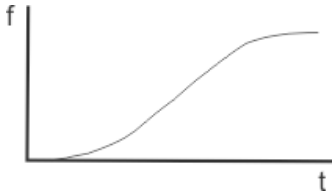
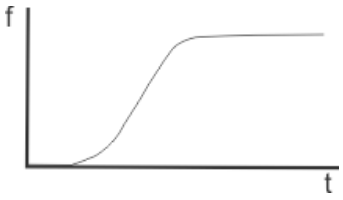
8 List of Parameters

No.	Meaning	Default
P01	Type of movement 0 = Rotational movement Rotating table, 1 limit switch for mechanical zero 1= linear, for XY tables or other linear systems, 2 limit switches: Mechanical zero and limit direction – Limit direction +	0
P02	Measuring units of movement 1 = step 2 = mm 3 = inch 4 = degree	1
P03	Conversion factor for the thread 1 step corresponds to ... If P03 = 1 (steps) the conversion factor is 1. Computing the conversion factor: $Conversion\,factor = \frac{Thread}{Number\,of\,steps\,per\,revolution}$ Example: 4 mm thread pitch 200-step motor = 400 steps/rev. in the half step mode $Conversion\,factor = \frac{4}{400} = 0.01$	1
P04	Start/stop frequency The start/stop frequency is the maximum frequency to start or stop the motor without ramp. At higher frequencies, step losses or motor stop would be the result of a start or stop without ramp. The start/stop frequency depends on various factors: type of motor, load, mechanical system, power stage. The frequency is programmed in Hz.	400
P07	Emergency stop ramp Enter in Hz/s	50 000
P08	f_{max} MØP, Run frequency during initializing Enter in Hz	4000

P09	Ramp MØP, Ramp during initializing, associated to parameter P08 Enter in Hz/s	25 000
P10	f_{min} MØP, Run frequency for leaving the limit switch range Enter in Hz	400
P11	MØP Offset for limit switch direction + Distance between reference point MØP and limit switch activation Unit: is defined in parameter P02	0
P12	MØP Offset for limit switch direction – Distance between reference point MØP and limit switch activation Unit: is defined in parameter P02	0
P13	Recovery time MØP Time lapse during initialization Enter in msec	20
P14	f_{max} Run frequency during program operation Enter in Hz (integer value)	4000
P15	Ramp for run frequency (P14) Input in Hz/s (integer value) OMC: 100 Hz/s to 4 MHz/s TMC: 1000 Hz/s to 4 MHz/s	25 000
P16	Recovery time position Time lapse after positioning Input in msec	20

P17	Boost (defined in P42) 0 = off 1 = on during motor run 2 = on during acceleration and deceleration ramp Remarks: The boost current can be set in parameter P42. You can select with parameter P17 in which situation the controller switches to boost current. P17 = 1 means, the boost current always is switched on during motor run. During motor standstill the controller switches to stop current.	0
The following four parameters are counters normally set by the program. **These parameters are not displayed in the communication software!**		
P19	Electronical zero counter Used for setting operating points. At standstill of the axis, P19 can be read or programmed during program execution.	0
P20	Mechanical zero counter This counter contains the number of steps referred to the mechanical zero (MØP). Can be read at axis standstill. If the axis reaches the MØP, P20 will be set to zero.	0
P21	Absolute counter Encoder, multi turn and also for single turn. The value of P22 is extended to P21 by software. The encoder counters have a fixed resolution, e.g. 10 bit (for single-turn encoders: the resolution is bits per turn), then the read value repeats. A saw tooth profile of the numerical values is produced during a continuous motor running. This course is "straightened" by software. P20 and P21 will be scaled to the same value per revolution by P3 and P39 and are therefore directly comparable, see P36.	0
P22	Encoder counter Indicates the true absolute encoder position.	0
P23	Software Limit Switch (Axial limitation pos. direction +) If the distance is reached, the run in + direction is aborted. 0 = no limitation	0

P24	Software Limit Switch (Axial limitation neg. direction –) If the distance is reached, the run in – direction is aborted. 0 = no limitation	0
P25	Compensation for play Indicates the distance, the target position in the selected direction is passed over and afterwards is started in reverse direction. 0 = no compensation for play	0
P26	Divider for SSI encoder Data transfer rate from 10 to 80 (=100 to 800 kHz)	0
P27	Limit switch type 0 = + and – are PNP normally closed contacts (NCC) 1 = + is a normally open contact (NOC), - is a NCC 2 = + is a NCC, - is a NOC 3 = + and – are PNP NOC	0
P28	Output motor brake Define the output number for the motor brake Input for OMC: 1 - 8 for TMC: 1 - 16 Example: output number = 4 The output 4 is set in case of an error in the power stage.	0
P29	Delay time for enabling motor brake The delay time after switch-on for releasing the brake Input in sec	0
P30	Frequency band setting 0 = manual 1 = automatic Remark: It is recommended to work with the automatic setting mode. For each run frequency (P14) and ramp (P15) the controller automatically selects suitable settings.	1

P31	Frequency and ramp predivider (only if P30 = 0, manual) Predivider values: 3 or 5 (OMC: 5, TMC: 3) This parameter can be used for individual settings for special applications.	3
P32	Positioning ramp shape 0 = s-shape 1 = linear ramp Remark: The s-shape ramp can be modified with P33 parameter.	1
P33	Arc value setting for s-shape ramp Values: OMC: 1 to 8191 TMC: 1 to 32767	1
	 P33: low value  P33: high value	
P34	Encoder type -1 = no encoder 0 = incremental 5.0 V 1 = incremental 5.5 V 2 = serial interface SSI binary Code 5.0 V 3 = serial interface SSI binary Code 5.5 V 4 = serial interface SSI Gray Code 5.0 V 5 = serial interface SSI Gray Code 5.5 V	0
P35	Encoder resolution for SSI encoder Enter max. encoder resolution in Bit (max. 32 Bit)	10
P36	Encoder function 0 = counter 1 = SFI dynamical step failure indication	0
P37	Encoder tolerance for SFI Enter tolerance value for SFI evaluation	0
P38	Encoder preferential direction of rotation 0 = + (positive) 1 = – (negative)	0

P39	Encoder conversion factor 1 increment corresponds to ...	1
P40	Stop current in 0.1 A steps Values: 0 to 6.3 A Input: 0 to 63	2
P41	Run current in 0.1 A steps Values: 0 to 6.3 A Input: 0 to 63	4
P42	Boost current as absolute value in 0.1 A steps which is not added to the run current. Values: 0 to 6.3 A Input: 0 to 63	4
P43	Current delay time in ms	20
P44	Control pulses change over 0 = power stage X (Y) on controller X (Y) 1 = power stage X (Y) on controller Y (X) 2 = external control pulses on power stage X (Y)	0
P45	Step resolution 0 = Full step 3 = 1/5 step 1 = Half step 4 = 1/10 step 2 = 1/4 step 5 = 1/20 step 6 = 1/8 step	4

9 Warranty, Disclaimer and Registered Trademarks

9.1 Disclaimer

Phytron GmbH has verified the contents of the manual to match with the hardware and software. However, errors and omissions are exempt and Phytron GmbH assumes no responsibility for complete compliance. The information contained in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

9.2 Warranty

The OMC/TMC modules are subject to **legal warranty**. Phytron will repair or exchange devices which show a failure due to defects in material or caused by the production process. This warranty does not include damage caused by the customer, for example, not intended use, unauthorised modifications, incorrect handling or wiring.

9.3 Registered Trademarks

In this manual several trademarks are used which are no longer explicitly marked as trademarks within the text. The lack of these signs may not be used to draw the conclusion that these products are free of rights of third parties. Some product names used herein are for instance.

- *phy***MOTION**™ is a trademark of the Phytron GmbH.
- *phy***LOGIC**™ is a trademark of the Phytron GmbH.
- Microsoft is a registered trade mark and WINDOWS™ is a trade mark of the Microsoft Corporation in the USA and other countries.
- *phy***LOGIC™** ToolBox is trademark of the Phytron GmbH.
- SIMATIC ET 200®S, STEP 7, TIA Portal is trademark of the SIEMENS AG.
- Microsoft is a registered trade mark and WINDOWS® is a trade mark of the Microsoft Corporation in the USA and other countries

10 Index