Principles of Positioning

for Stepper Motor Controllers

TRANSLATION OF THE GERMAN ORIGINAL MANUAL

Positioning

In this manual you will find the feature descriptions and programming a controller for positioning of a stepper motor.

This manual is supplementary to the „phyLOGIC™-Command References" and „ProfiNet/ProfiBus Interfaces".

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.de/) 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.*

Positioning

Observe the following safety instructions!

Safety Instructions

i

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.

i

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.

i

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.

i

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   Introduction

*phy*LOGIC[TM] is the programming language to communicate with phytron programmable logic controllers like the MCC-Series or our *phy*MOTION[TM].

*phy*LOGIC[TM] instructions can easily be sent to the controller with phytron's programming software (*phy*LOGIC[TM] Toolbox) via USB, embedded into other protocols like Ethernet or into interface protocols like ProfiBus / ProfiNet.

You can paramterise your commands (e.g. a driving command) per axis either just the first time you set up your system, or adjust the parameters temporarily before sending a driving command.

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

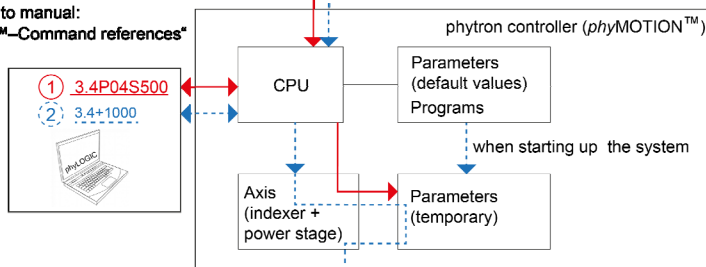Use this illustration to find the adequate manual for your programming task:
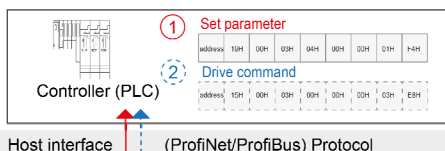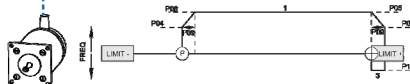
Fig. 1:   Illustration for finding the right manual for your programming task

Each of our programmable controllers comes along with preset 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   Setting Stepper Motor Parameters

For operating a phytron stepper motor controller several adjustable variables as speed, acceleration ramps or current values are required. These adjustable variables are called **parameters**. E.g. run frequency parameter (P14).

Chapter 6 in this manual contains the complete list of all parameters.

4.1   Velocity Profile of the Stepper Motor Controller

Revolution frequency of the stepper motor

The revolution frequency of a stepper motor is usually indicated in rpm. From the view of the stepper motor module a frequency is displayed at the output terminal (Run frequency P14). The relationship between the speed of the stepper motor (velocity n) and the displayed frequency (P14) is as follows:

P14 = (n x s)/ (60 s/min)

P14= Run frequency in [Hz]

n = Motor speed in [rpm]

s = Full step resolution of the stepper motor (typical: 200 steps/rev). For further information refer to the technical data for the stepper motor.

General velocity profile of the controller

Normally each incremental move is always carried out by the same velocity profile.

The stepper motor accelerates without a ramp to the start-stop frequency P04. Then the stepper motor follows over a ramp to the desired run frequency P14. The Range 2 is characterized by moving with constant speed. In range 3 the stepper motor is decelerated by a ramp. A system-specific frequency F_{max} limits the maximum speed of the drive system.
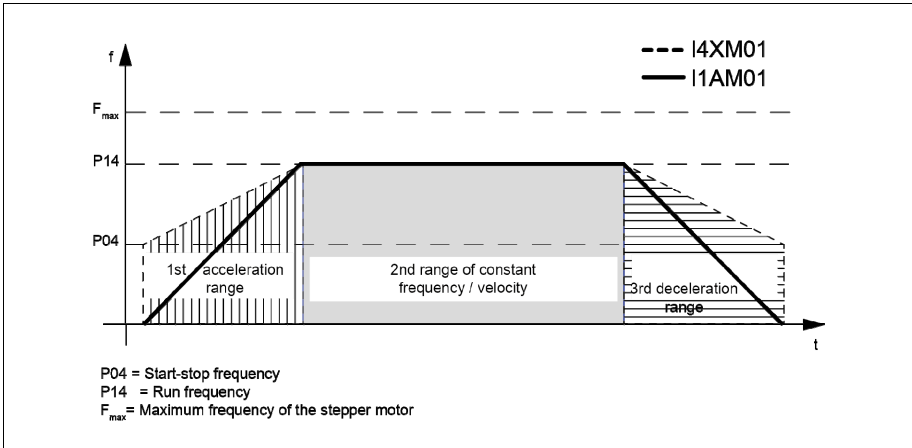
Fig. 2: Motion profile of the stepper motor controller

P04= start-stop frequency
(for P14>P04 I1AM01 does not use P04 for accelerating)

P14 = run frequency

F_{max}= *system-specific, maximum frequency of the stepper motor with a load applied*

Run frequency / velocity P14

- *The run frequency can be chosen for each drive.*

- *P14 is always lower than F_{max} and there should be a safety margin between P14 and F_{max}. Phytron recommends a safety factor of 1.4 to 2.*

Setting the run frequency / velocity P14

The run frequency can be set with parameter P14. This parameter is set in Hz. If you have used the formula above to calculate your frequency out of the needed motor velocity – you have a frequency in Hz full step. If you have selected a different step resolution than full step (P45) you have to increase your run frequency by the factor of the step resolution.

Example: If you have selected half step resolution, you have to double your run frequency to achieve the same speed as in full step mode. If you have selected 1/8 step resolution, you have to multiply your full step frequency by 8 etc.

Start-stop frequency P04

The start-stop frequency is the frequency to which the motor can instantly be accelerated under load from a standstill without losing the synchronization of the electrical field and also without losing steps.

The maximum start-stop frequency P04 mainly depends on the moment of inertia of the load, as well as on the friction of the system.

If the stepper motor must pass through a frequency range of resonances in the acceleration phase, either a ramp should be configured as steep as possible to pass through the resonance region quickly or the start-stop frequency should be set above the resonance frequency, or the mechanical system could be damped.

- *The parameter P04 has also to be set in Hz.*
- *If your selected speed (P14) is lower than the start-stop frequency (P04) the motor will be accelerated instantly (without a ramp) to the selected speed (P14).*
- *If your selected speed (P14) is greater than the start-stop frequency (P04)*
  - o *Your motor connected to the I1AM01 will accelerate on a defined ramp to the final speed (P14)*
  - o *Your motor connected to a high end indexer like I4XM01 will instantly accelerate to the start-stop frequency (P04) and from there on with a ramp to final speed (P14). This increases the possible speed of movement.*

Maximum frequency / Velocity of the Axis F_{max}

When choosing a stepper motor the maximum frequency/velocity is determined by the application. At this frequency, the motor must reach a torque high enough to move its load.

The maximum frequency F_{max} can be estimated from the corresponding characteristic curve.

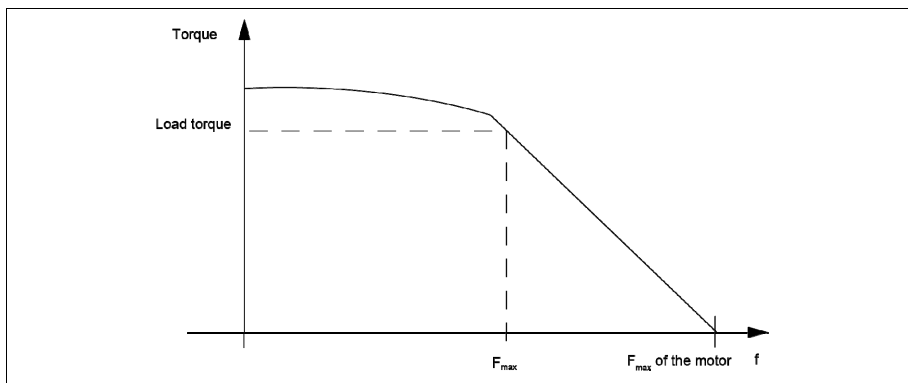Please note that a sufficient large safety margin must be applied.

Fig. 3: Torque Characteristic Curve of a Stepper motor

Acceleration / Deceleration ramp a (P15)

The maximum permitted acceleration / deceleration depends on the load to be moved.

The motor must reach a torque high enough to accelerate or delay the load without loss of step.

Depending on the application, you must also take into account additional criteria for setting the acceleration/deceleration, such as smooth starting and stopping.

- *The parameter P15 is also set in Hz.*

- *For the I1AM01 module: the ramp during acceleration can be set in 4 kHz steps.*

4.2 Phase Currents (Run, Stop, Boost Current)

Three different phase currents can be indicated for the controller: run current, stop current and boost current.

i The selected current is proportional to the created torque of the stepper motor. So before testing your system, you have to cross check that your current parameters are set properly to not destroy the attached motor or system. Always start with low current settings.

The run current is the one that is produced at a constant velocity (P14) during the run mode. After the motor is brought to a stop we recommend switching to a reduced stop current after a parameterized Run Current Delay Time (t_{DELAY} = P43). This reduces the thermal losses of the motor at standstill and saves power consumption.

While a stepper motor is accelerated or decelerated, it needs more torque and thus more power compared to a pure run with a constant velocity (P14). The torque can then be increased in the phases of acceleration and deceleration by using a boost current (P17, P42).
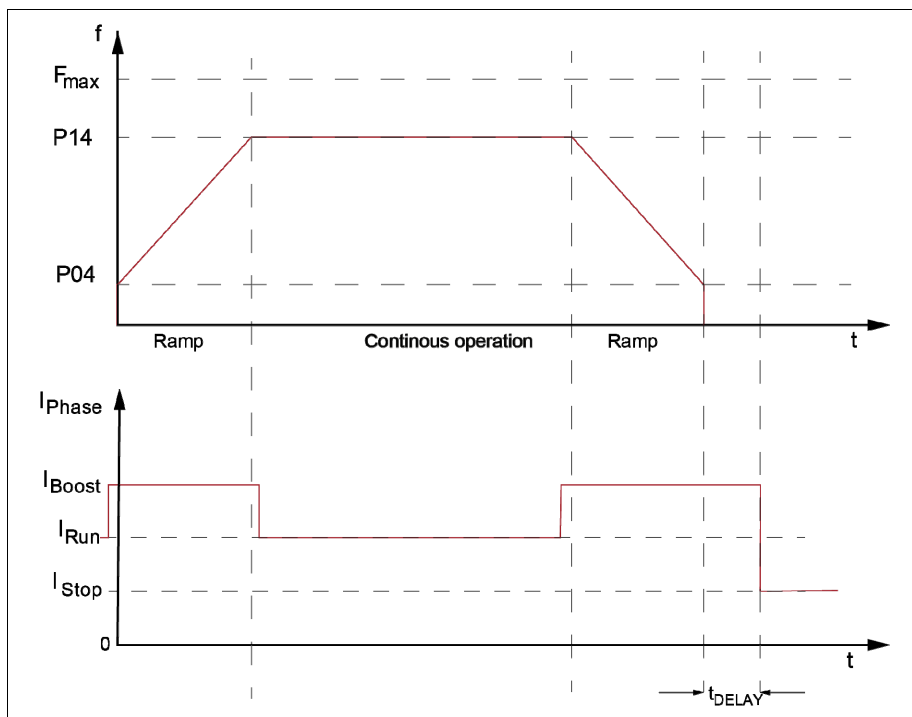
Fig. 4: Velocity profile versus current adjustment at the power stage

According to a time set in the parameter "Current Hold Time" t_{DELAY} (P43) it will be switched to stop current I_{STOPP} (P40) after the run is finished.

For more information, why a current delay time can be necessary, please continue reading in chapter 4.4.

4.3 Step Resolution

Full step

The "full step" mode is the operating mode in which a 200-step motor, for example, drives 200 steps per revolution. The physical resolution of the motor is achieved in the full step mode. Any further increase of the step resolution (e. g. half step, quarter step, etc.) is done electronically. In the full step mode, both stepper motor phases are permanently energized.

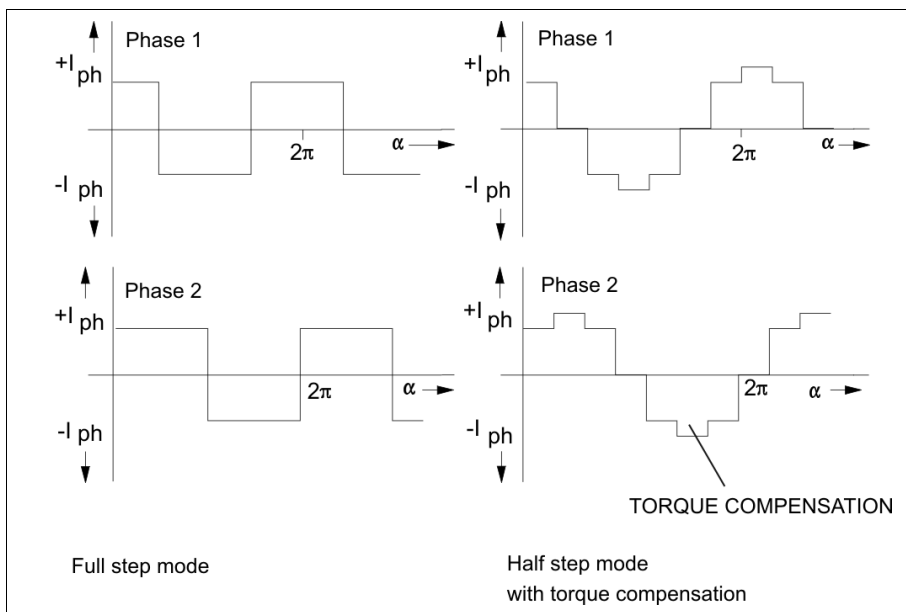The step resolution can be set with parameter P45.

Fig. 5: Phase current curves

Half Step

The motor step resolution can be electronically multiplied by 2 by alternately energizing the stepper motor's phases 1, 1+2, 2 etc. This is the "half step" mode. The torque, however, is reduced in the half step mode, compared to the full step mode.

To compensate for this lack of torque, the operating mode "half step mode with torque compensation" was developed: the current is increased by $\sqrt{2}$ in the energized phase. Compared to the full step mode, the torque delivered is almost the same and most of the resonance is suppressed.

The following diagram shows the magnitude and direction of the holding torques of a 4 step motor during one revolution without and with torque compensation. In the full step position two phases are energized, in the half step position only one phase is energized. The total torque is the result of the vector sum for any phases that are energized.

The Torque Full Step, M_{FS}, as compared to the torque in the half-step mode is: $\left| M_{FS} \right| = \left| M_{HS} \right| \times \sqrt{2}$

This means, when a single phase is energized, the current must be increased by a $\sqrt{2}$ factor to obtain an identical torque.
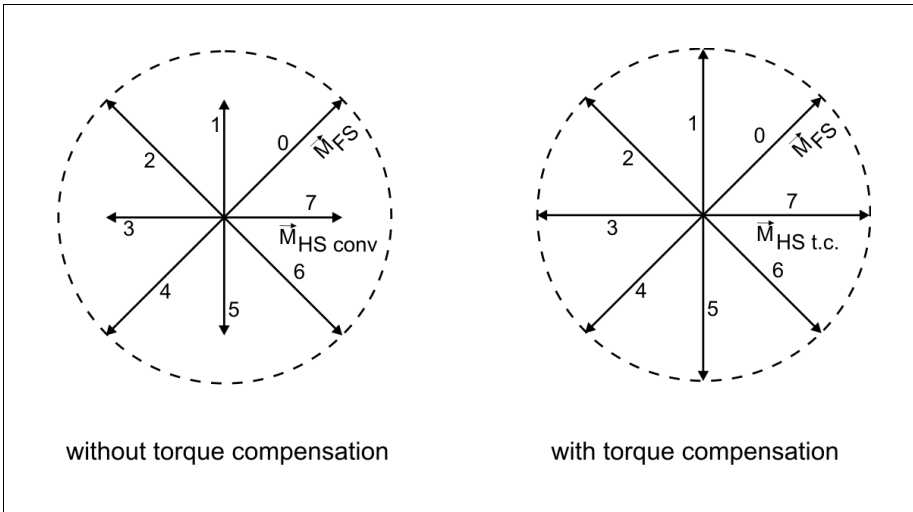

without torque compensation   with torque compensation

Fig. 6: Holding torques without/with torque compensation

- *All recent phytron power stages automatically perform the torque compensation.*

Micro Step

The step resolution of the controller can be increased electronically to up to 1/512 of a full step (depending on the selected module). A 200 step motor can, in theory, be commanded to one of 102,400 positions (equal to 0.0035° per micro step) per revolution.

Various advantages are obtained with the micro step mode:

- The torque undulation drops when the number of micro steps is increased.
- The achievable torque can increase up to 1/8 step, also a further increase of the resolution does not increase torque.
- Resonance and overshoot phenomena are greatly reduced; the motor operation is almost resonance-free.
- The motor noise also drops when the number of micro steps is increased.
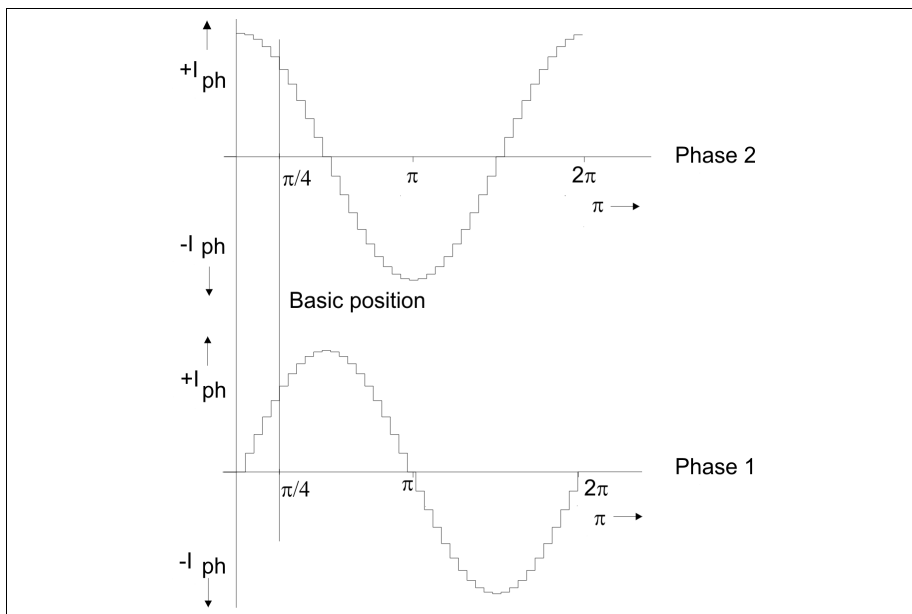
Fig. 7: Schematic profile of the phase currents with 1/10 micro step (of a full step)

i If using the highest micro step settings to perform accurate and absolute precision positioning use an attached encoder in order to achieve this. Then you can ensure the achievement of the target position or readjust if necessary. Even the slightest mechanical failure in the stepper motor can cause an incorrect micro step.

Fig. 8: Deviation from the target positioning in degree (for the APS01 power stage module)

4.4 Current Hold Time

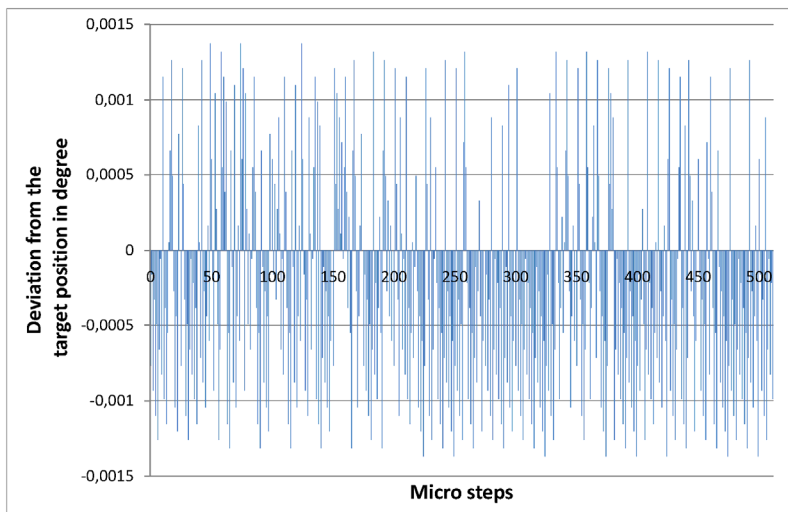After the last control pulse, the stop current is activated after a set time to minimize power consumption and motor heating. The time after the last control pulse until changing to the stop current is called Current Hold Time (t_{DELAY} / P43).

phytron recommends specifying t_{DELAY} so that the motor's oscillations are decaying after the last motor step and positioning is more accurate. The higher current in this case reduces the decay - and incorrect positioning is avoided.

Automatic change from run to stop current:

After the stop current is applied, the ratio between both phase currents remains the same in the respective current feed pattern. Changing from run to stop current is achieved synchronously. This is absolutely necessary to not loose the current position. The phytron power stages lower the current automatically synchronous.
In the following figure the next motor step follows after every **rising** control pulse edge:
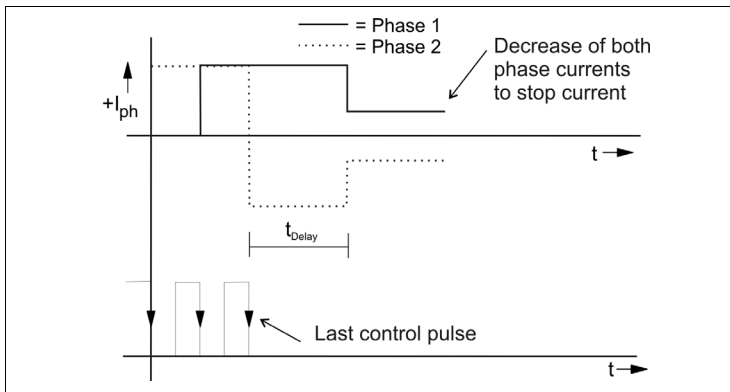
Fig. 9: Decrease to stop current after the last control pulse (full step)

Decreasing to stop current has the following advantages:

- Motor and power stage heating and power consumption is reduced.
- EMC is further improved due to smaller current values at a standstill.

The Current Hold Time t_{DELAY} after the last step of a motion job has the following advantages:

- The release time of the stepper motor at its target position will be accelerated. So the next motion job can be started quicker.
- Step loss, therefore incorrect positioning, by decaying effects on reaching a position is minimized.

5   Driving Functions of the Controller

The controller's task is to position a stepper motor (incremental modes) or to travel continuous with specifiable frequencies (velocity control mode). In addition a lot of technology parameters as described above can be adapted in a way that the performance of the stepper motor and the customer´s drive system is optimised.

5.1 Relative Positioning

The mode "relative positioning" is used to move the stepper motor a defined distance and thus approach a specified position.

Command: e.g. 1.2+1000

The 1.2 is the driving command in phyLOGIC^{TM} to set a motion command for axis 2 on module card 1. +1000 means: drive 1000 steps in the selected step resolution in + direction

5.2   Absolute Positioning

The "absolute positioning" mode is used to move the stepper motor to a defined position and thus approach a specified position.

Command: e.g. 1.2A1000

> *Again the second axis on driving card 1 is selected. This time "A" stands for absolute positioning mode. In this case 1000 stands for the absolute position 1000.*

5.3 Velocity Control Mode

This operating mode specifies the frequency with which the pulses (steps) are output. When the frequency is changed, the pulses are output with the new frequency after an acceleration or deceleration phase. The output is carried out continuous until either stopping the job or a certain limit.

Command: e.g. 1.2L+

> *Again axis 2 on driving card 1 is selected. This time L+ stands for drive in + direction with parameterized speed.*

5.4 Search for Reference

The home position marks the point of reference of the drive system for the following motion jobs. You can determine the home position by, for example, installing a proximity switch and connecting it to one of the limit switch connectors on the driving cards. Alternatively also a NOC (normally open contact) switch is possible.

Starting from the actual position $\textcircled{P}$ there are different ways to move to the reference point REF C $\oplus$ (+ optional offset $\textcircled{O}$). This search is defined by several parameters.

Parameter no.	Meaning
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.
P08	f_{max} MØP (mechanical zero point) Run frequency during initializing (referencing) Enter in Hz
P09	Ramp MØP Ramp during initializing, associated to parameter P08 For I1AM01 the ramp can be adjusted in 4000-Hz/sec-steps
P10	f_{min} MØP Run frequency for leaving the limit switch / center switch Enter in Hz This frequency must be lower or equal to the start stop frequency to guarantee an accurate positioning.
P11 / P12	If parameters P11 / P12 are not 0 after driving to REF C $\oplus$ the controller will drive the parameterised offset to a defined point. $\textcircled{O}$ Set the offset P11 (away from "LIMIT+" switch, towards "LIMIT–" switch) or P12 (away from "LIMIT–" switch towards "LIMIT+" switch) parameter.
P14	f_{max} Run frequency during program operation Enter in Hz (integer value)
P15	Ramp for run frequency (P14) Input in 4000-Hz/sec-steps (4000 to 500 000 Hz/sec)

Explanation of the reference run commands

R+   Reference run towards "LIMIT+" switch, optional offset*

R+C   Reference run towards "LIMIT+" switch, then towards center switch, then optional offset* (away from "LIMIT+" switch towards „LIMIT−" switch) P11

RC+   Reference run towards center switch (the direction depends on the center switch status') then + offset (away from „LIMIT−"switch towards „„LIMIT+" switch) P12

R+C^I   Reference run towards "LIMIT+" switch, center switch and then Encoder Index, then optional offset*

R-   Reference run towards "LIMIT−" switch, optional offset*

R-C   Reference run towards "LIMIT−" switch, then towards center switch, then optional offset (away from "LIMIT−" switch towards "LIMIT+" switch) P12

RC-   Reference run towards center switch (the direction depends on the center switch status') then − offset* (away from "LIMIT+" switch towards "LIMIT−" switch) P11

R-C^I   Reference run towards "LIMIT−" switch, center switch and then Encoder Index, then optional offset*

*) offset only if P11/P12 ≠ 0

Note:

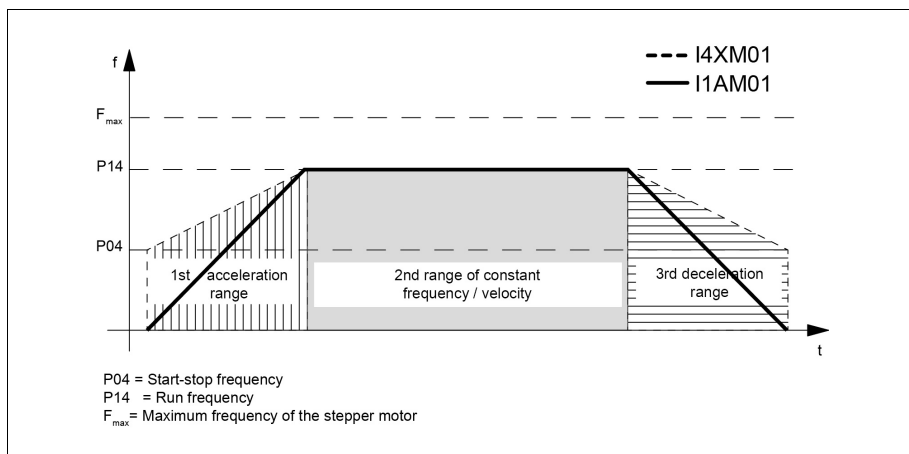The following diagrams are drawn for I4XM01. For I1AM01 the ramp starts always at 0.

P04 = Start-stop frequency
P14 = Run frequency
F_{max}= Maximum frequency of the stepper motor

Fig. 10: Motion profile of the stepper motor controller

Positioning

Driving on a *reference signal* to "center" (& offset) referring to command: "RC+"

Center switch 50% damped

Fig. 11: Driving on a reference signal **starting from "signal active"** ("+ offset": P12)

Fig. 12: Driving on a reference signal **starting from "signal inactive"** ("+offset": P12)

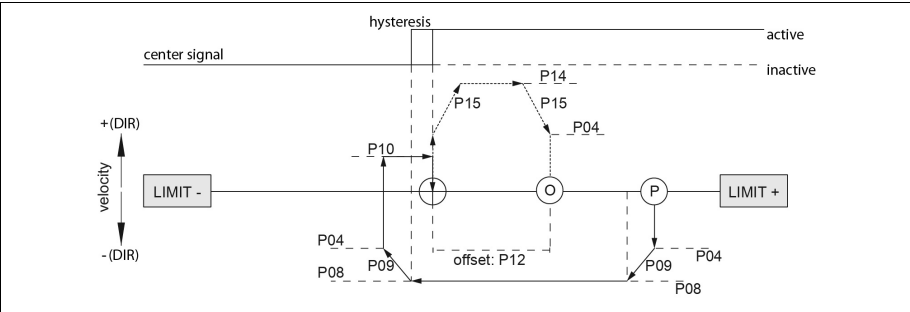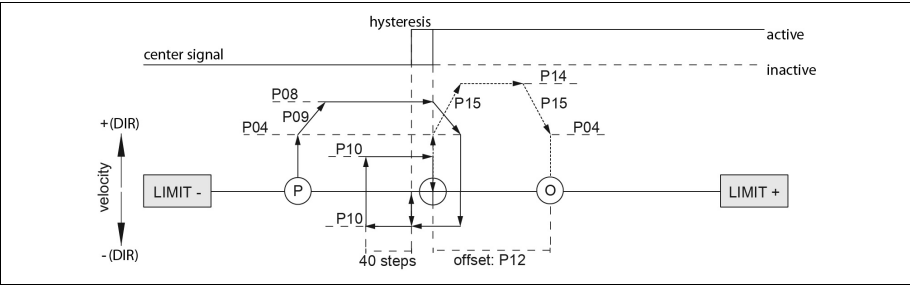**Driving on a *reference signal* to "center" (& offset) referring to command "RC-"**
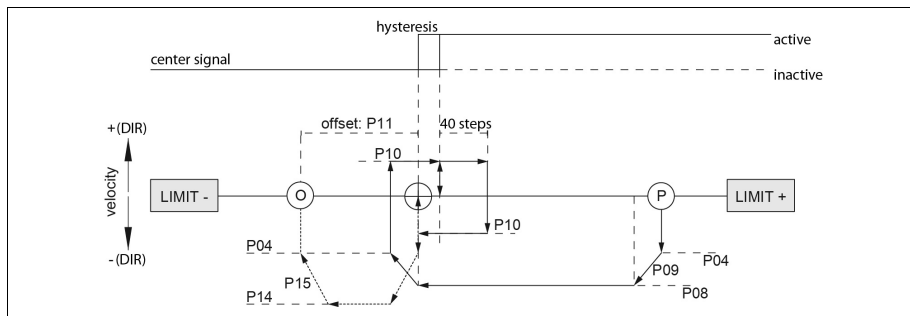

Fig. 13: Driving on a reference signal **starting from "signal active"** ("-offset": P11)

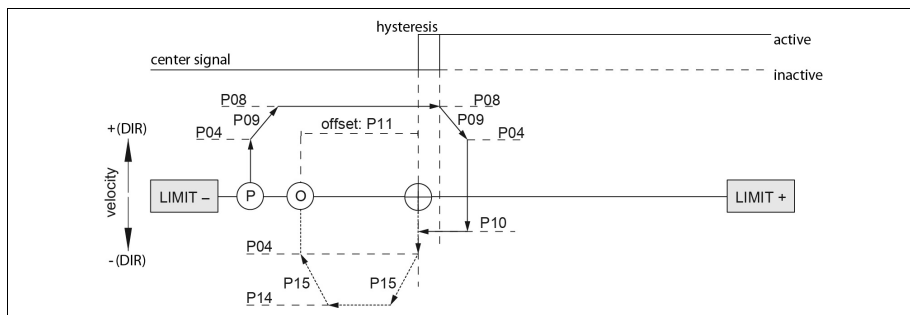Fig. 14: Driving on a reference signal **starting from "signal inactive"** ("-offset": P11)

Positioning

Driving on a *center switch* (& offset) referring to command "R+C"

Center switch selective damped

Fig. 15: Driving on a center switch **starting right to "center"** via "LIMIT+" switch ("- offset": P11)

Fig. 16: Driving on a center switch **starting left to "center"** ("- offset": P11)

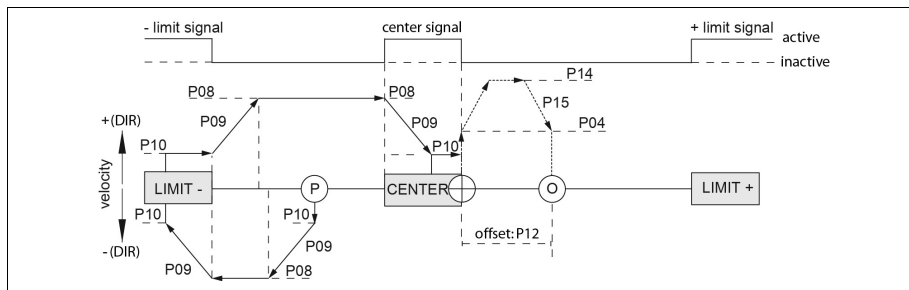**Driving on a *center switch* (& offset) referring to command "R-C"**

Fig. 17: Driving on a center switch **starting left to "center"** via "LIMIT–" switch ("+ offset": P12)
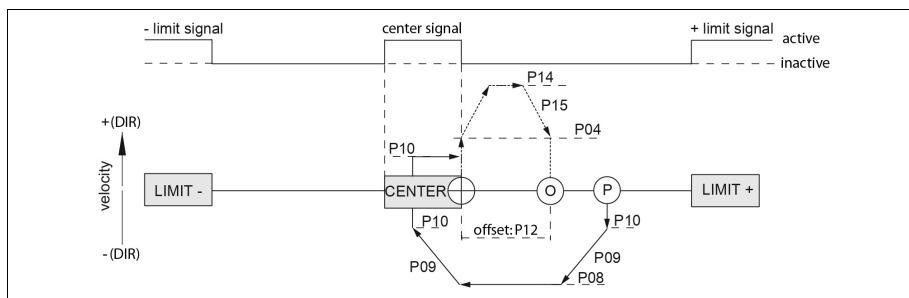

Fig. 18: Driving on a center switch **starting right to "center"** ("+ offset": P12)

5.5 Linear Interpolation

Only with the modules I4XM01 or I2XM01

An interpolation function is necessary for displaying paths with more than one axis. The new position is automatically approached by then velocity adjustment of all participated axes on a straight path. The linear interpolation can handle all 4 axes.
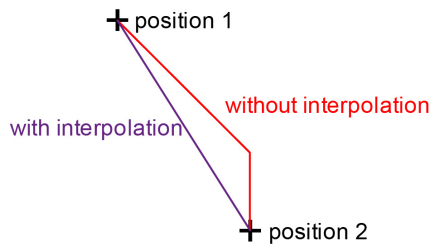
Without interpolation you must adjust the parameters for the velocity of both axes manually. So both axes start and finish the run simultaneously.

The linear interpolation ensures this automatically so that always position 2 is approached directly (linear connection).

Use the following *phy***LOGIC**^TM commands for programming the linear interpolation:

IS   Set the   here is the
         comma   one row.

<center>**In Preparation!**</center>
<center>You receive the current status</center>
<center>from the phytron Support.</center>

IR   Reset th

5.6 Circular Interpolation

Only with the modules I4XM01 or I2XM01

The circular interpolation is necessary for running multiple axes in a coordinated motion similar to the linear interpolation. Here, however, they are circular or elliptical paths. From the current position of the central point calculated with the radius and the starting angle with the radius a circular movement is performed. The circular movement stops at the indicated rotation angle.

The sign of the rotation angle defines the direction:

Rotation angle > 0   → Direction counter-clockwise

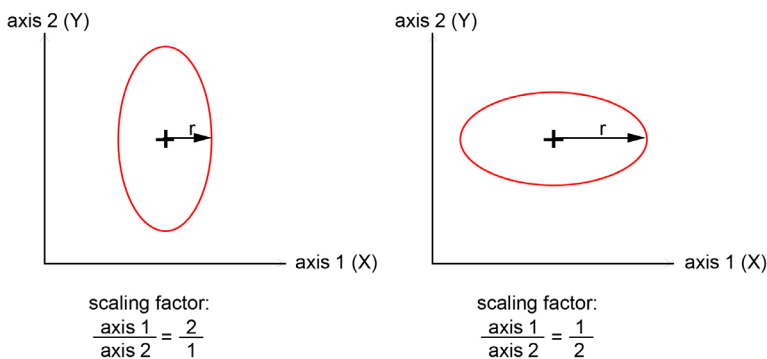Rotation angle < 0   → Direction clockwise

The circular interpolation can handle with 2 of 4 axes.
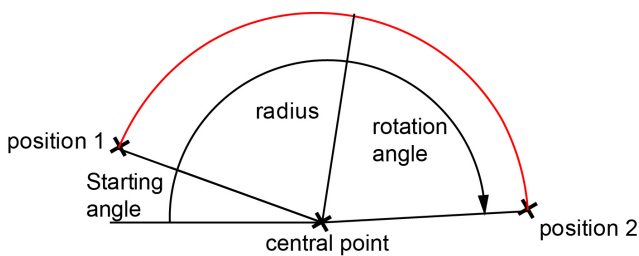
For displaying ellipses a different adjustment of the factor of both participated axes is necessary. The indicated radius always defines the dimension of axis 1 (X).

Example:

scaling factor:
$$\frac{axis\ 1}{axis\ 2} = \frac{2}{1}$$

scaling factor:
$$\frac{axis\ 1}{axis\ 2} = \frac{1}{2}$$

Positioning

Use the following *phy**LOGIC***^{TM} commands for programming the circular interpolation:

x**KR**n   Set the radius n of the circular arc for the Indexer module x,
the unit and the factor of n are defined in P2 and P3 (see chap. 6)

x**KS**n   Set the starting point n on the circular path for the Indexer module x in
degree (°)
n =0 to 360°

x**KW**n   Set the path (sector) n in degree (°) from the starting point for the Indexer
module x
n =0 to 360° (CCW)
n =0 to -360° (CW)

                      Important: Write these 3 commands in **1** program line!

                      Example: 1KR100 1KS90 1KW180

x**KG**a;b   Set the axis assignment of the Indexer card x ,

                      a= Master axis (1,2 or 3)

                      b=Slave axsis (1,2,3 or 4)

x**KT**a:b   Set the divider of axis a and axis b of the Indexer module x (for ellipsis run)

                      a: divider for axis 1
b: divider for axis 2

6 List of Parameters

For operating a stepper motor controller several presettings as speed, acceleration ramps or waiting time are required. These presettings are called **Parameters**.

Default parameters are stored which can be used in several applications at delivery. You can read and edit these parameters with *phy***LOGIC**^{TM}ToolBox.

Several counters are also contained in the list of parameters, which will be continuously actualized by the program. The counters can be read and some of them can be edited, too.

- For each axis separate parameters have to be set. Insert a modul and axis number to mark the axis in front of the parameter number.

  Example: m.aP15 is the acceleration ramp value for axis m.a.

- Parameters (e.g. speeds) may be modified several times within a program, too.

- Parameter values can be entered or read.

- P49 can only be read.

- P20 to P22 are counters. They will be actualized by the program during axis movement.

- P27 to P54 are special parameters for the *phy***MOTION**^{TM}.

No.	Meaning	Default
P01	Type of movement (free run, relative / absolute, reference run) 0 = Rotational movement (ignoring limit switches) 1 = Hardware limit switches are monitored for XY tables or other linear systems, 2 limit switches: Mechanical zero and limit direction – Limit direction + 2 = Software limit switches are monitored 3 = Hardware and software limit switches are monitored	0
P02	Measuring units of movement: only used for displaying 1 = step 2 = mm 3 = inch 4 = degree	1

Positioning

No.	Meaning	Default
P03	Conversion factor for the thread 1 step corresponds to ... If P03 = 1 (steps) the conversion factor is 1. Computing the conversion factor: $$Conversion\ factor = \frac{Thread}{Number\ of\ steps\ per\ revolution}$$ <u>Example</u>: 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
P05 **P06**	not used	
P07	Emergency stop ramp The frequency is programmed in 4000-Hz/sec-steps.	100 000
P08	f_{max} MØP (mechanical zero point) Run frequency during initializing (referencing) Enter in Hz (integer value)	4000
P09	Ramp MØP Ramp during initializing, associated to parameter P08 Enter in 4000-Hz/sec-steps	4000
P10	f_{min} MØP Run frequency for leaving the limit switch range Enter in Hz	400

No.	Meaning	Default
P11	MØP offset for limit switch direction + (away from "LIMIT+" switch, towards "LIMIT–" switch) Distance between reference point MØP and limit switch activation Unit: is defined in parameter P02	0
P12	MØP offset for limit switch direction – (away from "LIMIT–" switch, towards "LIMIT+" switch) 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 (for I1AM01: 40 000 maximum)
P15	Ramp for run frequency (P14) Input in 4000-Hz/sec-steps (4000 to 500 000 Hz/sec)	4000
P16	Recovery time position Time lapse after positioning Input in msec	20
P17	Boost 0 = off 1 = on during motor run 2 = on during acceleration and deceleration ramp <u>Remarks:</u> 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

Positioning

No.	Meaning	Default
P18	not used	
P19	not used	
P20	Mechanical zero counter This counter contains the number of steps referred to the mechanical zero (MØP). If the axis reaches the MØP, P20 will be set to zero.	0
P21	Absolute counter Encoder, multi turn and also for single turn.	0
P22	Encoder counter Indicates the true absolute encoder position.	0
P23	Software Limit Switch (Axial limitation pos. direction +) If the number of steps is reached, the run in + direction is aborted. 0 = no limitation	0
P24	Software Limit Switch (Axial limitation neg. direction –) If the number of steps is reached, the run in – direction is aborted. 0 = no limitation	0
P25	Compensation for play Indicates the step number, 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 1 to 8 (= 100 to 800 kHz) 1 = 100 kHz 2 = 200 kHz 3 = 300 kHz 4 = 400 kHz 5 = 500 kHz 6 = 600 kHz 7 = 700 kHz 8 = 800 kHz	1

No.	Meaning	Default
P27	Limit switch type NCC: normally closed contact NOC: normally open contact	0

	LIMIT–	Center/Ref	LIMIT+
0	NCC	NCC	NCC
1	NCC	NCC	NOC
2	NOC	NCC	NCC
3	NOC	NCC	NOC
4	NCC	NOC	NCC
5	NCC	NOC	NOC
6	NOC	NOC	NCC
7	NOC	NOC	NOC

No.	Meaning	Default
P 28 to **P33** not used		
P34	Encoder type 0 = no encoder 1 = incremental 5.0 V 2 = incremental 5.5 V 3 = serial interface SSI binary Code 5.0 V 4 = serial interface SSI binary Code 5.5 V 5 = serial interface SSI Gray Code 5.0 V 6 = serial interface SSI Gray Code 5.5 V 7 = Endat 5 V 8 = Endat 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 = dynamical position correction (only A/B)	0
P37	not used	

Positioning

No.	Meaning	Default
P38	Encoder preferential direction of rotation 0 = + (positive) 1 = – (negative)	0
P39	Encoder conversion factor 1 increment corresponds to ...	1
P40	Stop current in 0.01 A_{eff} steps depending on the power stage I1AM01: 0 to 250 (0 to 2.5 A_{eff}) ZMX$^+$: 0 to 630 (0 to 6.3 A_{eff}) APS: 0 to 350 (0 to 3.5 A_{eff})	2
P41	Run current in 0.01 A_{eff} steps I1AM01: 0 to 250 (0 to 2.5 A_{eff}) ZMX$^+$: 0 to 630 (0 to 6.3 A_{eff}) APS: 0 to 350 (0 to 3.5 A_{eff})	6
P42	Boost current in 0.01 A_{eff} steps I1AM01: 0 to 250 (0 to 2.5 A_{eff}) ZMX$^+$: 0 to 630 (0 to 6.3 A_{eff}) APS: 0 to 350 (0 to 3.5 A_{eff})	10
P43	Current hold time in msec	20
P44	Origin of the Control pulses for the axis 0 = 1:1 (Input=Output) 1 = from X 2 = from Y 3 = from Z 4 = from U 5 = from external	0
P45	Step resolution 1 to 256 0 = 1/1 step 7 = 1/16 step 1 = 1/2 step 8 = 1/20 step 2 = 1/2.5 step 9 = 1/32 step 3 = 1/4 step 10 = 1/64 step 4 = 1/5 step 11 = 1/128 step 5 = 1/8 step 12 = 1/256 step 6 = 1/10 step 13 = 1/512 step (e.g. APS01)	3
P46	not used	

No.	Meaning	Default
P47	not used	
P48	not used	
P49	Power stage temperature in 1/10 °C (read only)	(read only)
P50	Divider for Control pulses not for the I1AM01 Control pulses $_{Output}$=1/(n+1) * Control pulses $_{Input}$ 0 : 1/(0+1)=1 1: 1/(1+1)= 1/2 2: 1/(2+1) =1/3 3: 1/(3+1)=1/4 4: 1/(4+1)=1/5 5: 1/(5+1)=1/6	n=0
P51	Pulse width: (n+1)*100 ns not for the I1AM01 n: 0….31 e.g. n=19: (19+1)*100 ns=2000 ns= 2µs -> F_{max}=1/(2*2 µs)=250 kHz	n=19
P52	Trigger position	0
P53	Power stage monitoring 0 = off 1 = on	1
P54	Motor temperature in 1/10 °C; read only -999999: Temperature module not existent -9999: negative overflow or temperature lower -220 °C at PT100 9999: positive overflow or temperature higher +390 °C at PT100	-999999

7   Index