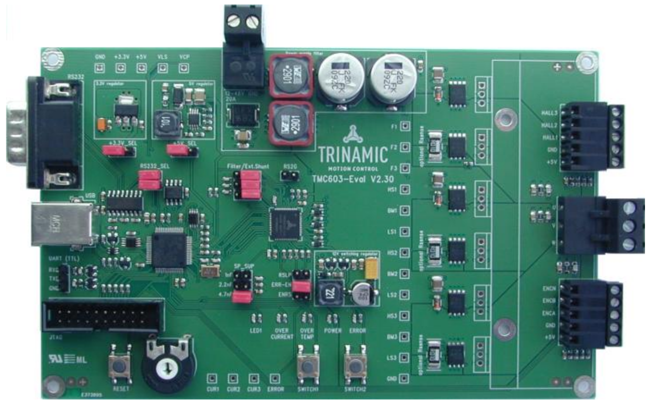
Firmware Version V1.47

EVAL BOARD MANUAL



TMC603-EVAL

Evaluation board for TMC603 three phase motor driver
18.8A RMS / 12… 48V DC
RS232 / UART, USB (mini-USB)
back EMF commutation hall*FX*™
encoder interface

TRINAMIC Motion Control GmbH & Co. KG
Hamburg, Germany

www.trinamic.com

Table of contents

2 Features

The TMC603 evaluation board makes it possible to evaluate the features of the TMC603 three phase BLDC motor driver with back EMF commutation hall*FX*™. On the evaluation board the STM32F ARM Cortex-M3 microcontroller is used to control the TMC603. The FLASH memory of the microcontroller contains a program which configures the TMC603 and controls the communication with the PC via the USB interface and the RS232 interface. The PC software is based on Windows and allows tuning of all operation parameters for every three phase BLDC motor.

Application

- Evaluation of the features of the TMC603 three phase motor driver

Electrical data

- Motor current: up to 18.8A RMS nominal motor current
- Supply voltage: 12V… 48V operating voltage

Interfaces

- RS232 (UART)
- USB (type B)
- Inputs for encoder (ABN)
- Three digital hall sensors

Motor type

- Three phase BLDC motor
- Sine or block commutation
- Rotor position feedback: sensorless, encoder or hall sensor

Safety features

- Overcurrent / short to GND and undervoltage protection with diagnostics integrated

Software

- TMCL™ (TMCL-IDE and TMCL-BLDC)
- Firmware update via USB interface and RS232 interface

Highlights

- Integrated current measurement using power MOS transistor RDSon
- hall*FX*™ sensorless back EMF commutation emulates hall sensors
- Integrated break-before-make logic. No special microcontroller PWM hardware required
- EMV optimized current controlled gate drivers
- Internal QGD protection supports latest generation of Power MOSFETs
- Integrated supply concept with step down switching regulator
- Common rail charge pump allows for 100% PWM duty cycle

3 Order codes

Order code	Description	Size of unit [mm^3]
TMC603-EVAL	Evaluation board for TMC603 three phase motor pre-driver	160 x 100 x 13.5

Table 3.1: Order codes

4 Hardware

4.1 Mechanical and electrical interfacing

4.1.1 Size of TMC603-EVAL and mounting holes

The board dimensions are 160mm x 100mm. Maximum component height (height above PCB level) without mating connectors is 13.5mm. There are six mounting holes (hole diameter: 3.2mm) suitable for M3 screws.

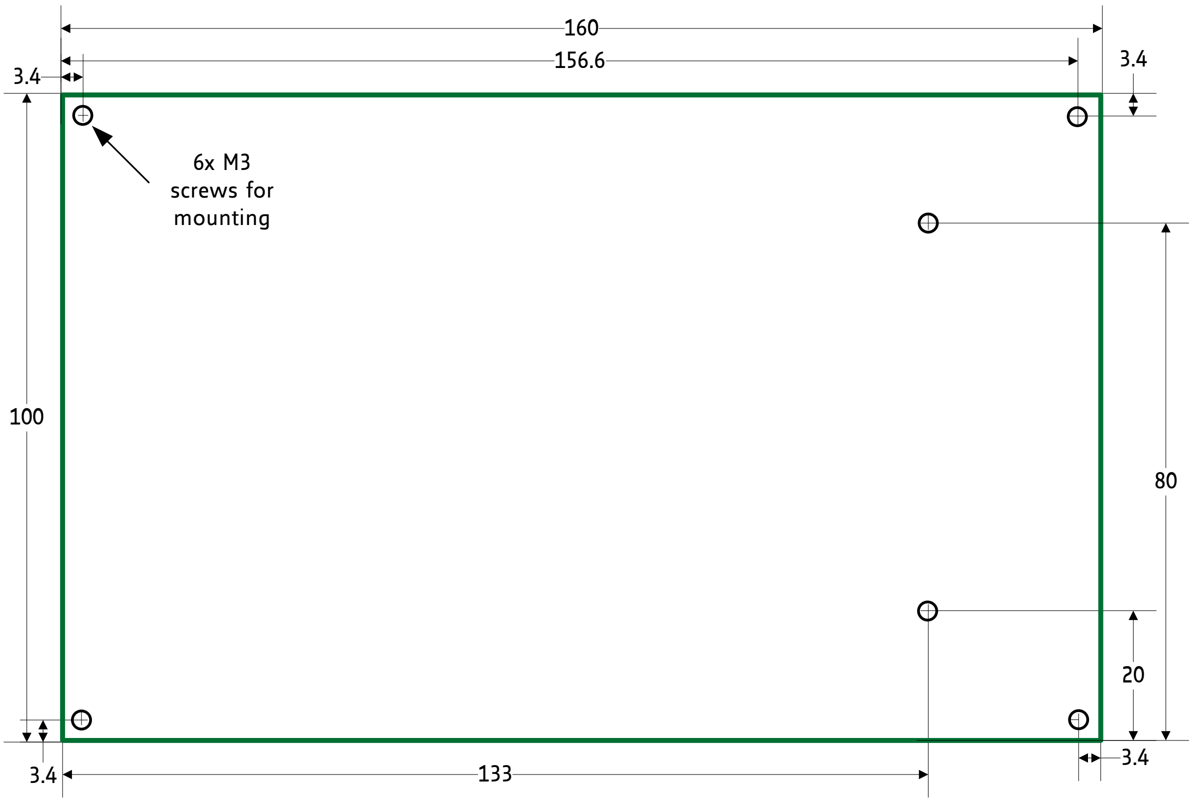


Figure 4.1: TMC603-EVAL dimensions

4.1.2 Connectors

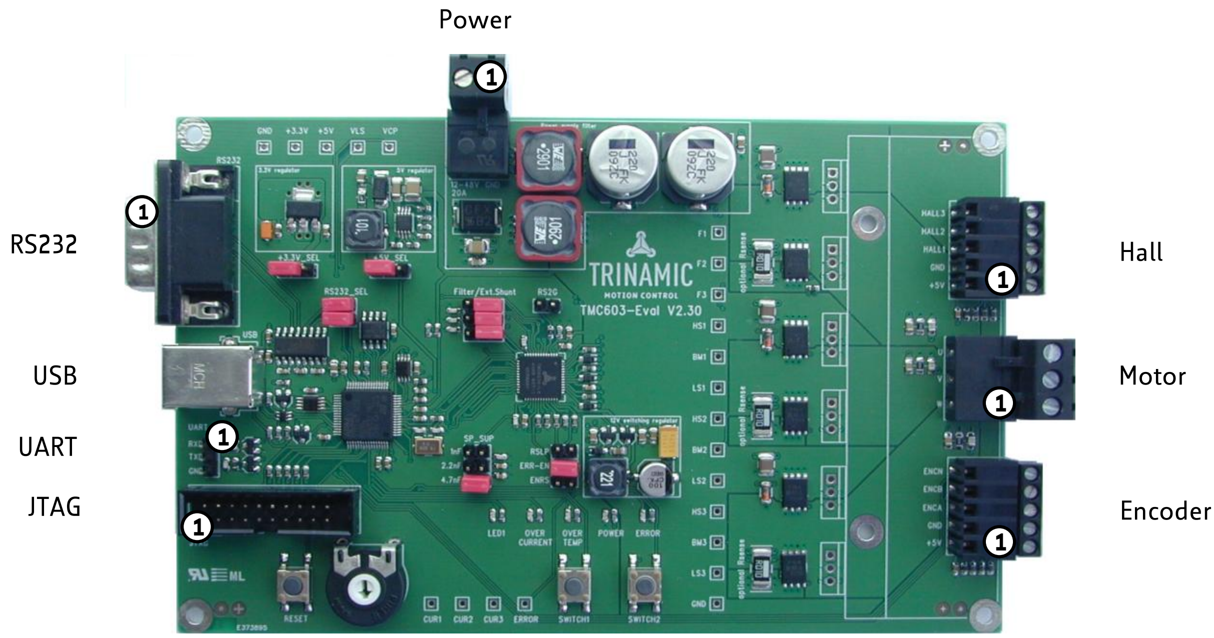


Figure 4.2: Connectors of TMC603-EVAL

CONNECTORS OF TMC603-EVAL

Label	Connector type	Mating connector type
Power	RIA 330-02, 2 pol., 5mm pitch, shrouded header	RIA 349-2, screw type terminal block, pluggable, centerline 5 mm / 0.197 inches, wire entry parallel to plug direction
Motor	RIA 330-03, 3 pol., 5mm pitch, shrouded header	RIA 349-3, screw type terminal block, pluggable, centerline 5 mm / 0.197 inches, wire entry parallel to plug direction
Encoder	RIA 182-05, 5 pol., 3.5mm pitch, header	RIA 169-05, screw type terminal block, pluggable, centerline 3.5 mm / 0.138 inches, wire entry parallel to plug direction
Hall sensor	RIA 182-05, 5 pol., 3.5mm pitch, header	RIA 169-05, screw type terminal block, pluggable, centerline 3.5 mm / 0.138 inches, wire entry parallel to plug direction
USB	USB, type B, 4 pol., female	USB, type B, 4 pol., male
UART	Multi-pin-connector, 3 pol., 2.54mm pitch	Female connector with 2.54mm pitch
RS232	DSUB, vertical, 9 pol., male, US-type	DSUB, 9 pol., female
JTAG	Low profile box header without locking bar, type 8380, 20 pol., DIN 41651, 2.54mm pitch	Low profile IDC socket connector, 20pol., DIN41651, 2.54mm pitch

Table 4.1: Connectors

4.1.2.1 Power connector

Pin	Label	Direction	Description
1	GND	Power (GND)	Power supply and signal ground
2	+UB	Power (Supply input)	Supply voltage: +12… +48V DC

Table 4.2: Power connector

4.1.2.2 Motor connector

Pin	Label	Direction	Description
1	W	Output	Motor coil connection W
2	V	Output	Motor coil connection V
3	U	Output	Motor coil connection U

Table 4.3: Motor connector

4.1.2.3 Hall sensor connector

Pin	Label	Direction	Description
1	+5V	Output	Power supply output for hall sensor, nom. +5V DC
2	GND	GND	Power supply and signal ground
3	HALL1	Input (5V TTL)	Hall sensor 1
4	HALL2	Input (5V TTL)	Hall sensor 2
5	HALL3	Input (5V TTL)	Hall sensor 3

Table 4.4: Hall sensor connector

4.1.2.4 Encoder connector

Pin	Label	Direction	Description
1	+5V	Output	Power supply output for hall sensor, nom. +5V DC
2	GND	GND	Power supply and signal ground
3	ENC_A	Input	Encoder signal A
4	ENC_B	Input	Encoder signal B
5	ENC_N	Input	Encoder signal N

Table 4.5: Encoder connector

4.1.2.5 USB connector

Pin	Label	Description
1	+5V	Board is self-powered – just use to detect availability of attached host system (e.g. PC)
2	USB-	Differential USB bus
3	USB+	Differential USB bus
4	GND	System and module ground

Table 4.6: USB connector

4.1.2.6 RS232 connector

Pin	Label	Description
2	RXD	Received data line
3	TXD	Transmitted data line
5	GND	RS232 signal and system ground

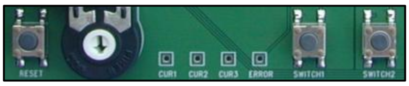
Table 4.7: RS232 connector

4.1.2.7 UART connector

Pin	Label	Description
1	RXD	Received data line
2	TXD	Transmitted data line
3	GND	Signal and system ground

Table 4.8: UART connector

4.1.3 Switches and potentiometer



Reset Potentiometer Switch1 Switch2

Figure 4.3: Switches and potentiometer

Label	Description
Reset	The reset switch is connected to the NRST pin of the µC. Press it to reset the module.
Potentiometer	The potentiometer is connected to pin PC5 of the µC (ADC_IN15/PC5). It can be used customer specific by GAP 22.
Switch1	Switch1 is connected to pin PB1 of the µC (PB1/ADC_IN9/T3_CH4). The switch can be used customer specific by GAP 21.
Switch2	Switch2 is connected to pin PB0 of the µC (PB0/ADC_IN8/T3_CH3). The switch can be used customer specific by GAP 20.

Table 4.9: Switches and potentiometer

4.1.4 Jumpers

Jumper	Label	Description
Select +3.3V supply	+3.3V_SEL	Jumper pins: Result 1, 2: +5V supply voltage of additional switching regulator (*default*). Precondition: +5V_SEL jumper on pin 1 and 2. 2, 3: +12V low side driver supply voltage of TMC603.
Select +5V supply	+5V_SEL	Jumper pins: Result 1, 2: +5V supply voltage of additional switching regulator (*default*) 2, 3: +5V Internal supply voltage of TMC603
Short to GND control resistor	RS2G	The short to GND control resister controls the delay time between switching on the high side MOSFET and the short to GND check. Jumper plugged: R_{S2G} = 470kΩ (t_{S2G} = 2µs) Jumper unplugged: R_{S2G} = 1.47MΩ (t_{S2G} = 6µs) *default*
Slope control resistor	RSLP	The slope control resistor sets output current for MOSFET drivers. Jumper plugged: R_{SLP} = 100kΩ (I_{gate} = 100mA) Jumper unplugged: R_{SLP} = 147kΩ (I_{gate} = 68mA) *default*
Error enable	ERR-EN	Jumper plugged: In case of error (e.g. short circuit on output) the TMC603 turns off itself by hardware. Jumper unplugged: Without jumper the shutdown has to be realized by the processor.

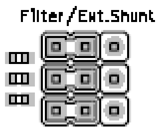
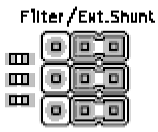
Jumper	Label	Description
Enable/disable shunt resistors for current measurement	ENRS	Set this jumper together with the three jumpers Filter/Ext_Shunt. There are two possibilities. - Set ENRS jumper together with three jumpers Filter/Ext_Shunt on pin position 2-3 (extern shunt). Now, the measurement of the motor current will be done with the shunt resistors on the board. - If the ENRS jumper is not set and the three jumpers Filter/Ext_Shunt are set on pin position 1-2 (filter position), there will be the filtered, measured motor voltages on the measuring points F1, F2, and F3.
Spike suppression time control capacitor	SP_SUP	An external capacitor on this pin controls the commutation spike suppression time for hall*FX™*. This pin charges the capacitor via an internal current source. All jumpers unplugged C_{SP_SUP} = 470pF (t_{SP_SUP} = 47µs) Jumper 1nF plugged C_{SP_SUP} = 1.47nF (t_{SP_SUP} = 147µs) Jumper 2.2nF plugged C_{SP_SUP} = 2.67nF (t_{SP_SUP} = 267µs) Jumper 1nF and 2.2nF C_{SP_SUP} = 3.67nF (t_{SP_SUP} = 367µs) Jumper 4.7nF plugged C_{SP_SUP} = 5.17nF (t_{SP_SUP} = 517µs) *Default* Jumper 1nF and 4.7nF C_{SP_SUP} = 6.17nF (t_{SP_SUP} = 617µs) Jumper 2.2nF and 4.7nF C_{SP_SUP} = 7,37nF (t_{SP_SUP} = 737µs) All jumpers plugged C_{SP_SUP} = 8,37nF (t_{SP_SUP} = 837µs)
Filter output / external sense resistor input	Filter/Ext.Shunt	Selects output signal of internal switched capacitor filters or input for external sense resistor (*default*). When the internal R_{DSon} of optional MOSFETs in TO-220 package are used to measure the actual motor current, set the jumper to pin 1 and 2. In that case, the filter outputs of TMC603 are  connected to the measuring points Fx (refer chapter 10). In addition, set the axis parameter 225 to zero (refer chapter 6), to select the current measurement by using the internal R_{DSon} of optional MOSFETs. When the external sense resistors are used to measure the actual motor current, set the jumpers to pin 2 and 3 (*default*). In that case, the external sense resistors are connected to the analog  inputs RSx of TMC603 (refer TMC603 datasheet). In addition, set the axis parameter 225 to one (refer chapter 6), to select the current measurement by using the external sense resistor.
RS232 select	RS232_SEL	Both jumpers plugged RS232 usable on D-sub connector Both jumper unplugged UART TTL usable on D-sub connector

Table 4.10: Jumpers

4.1.5 LEDs

LEDS OF THE TMC603-EVAL

Status	Label	Description
Error signal	ERROR	This red LED lights up upon an error is occurred by undervoltage of VLS or VCP as well as by short to ground of the power MOS half bridge.
Power on	POWER	This green LED lights up upon the power supply is working.
Temperature warning	OVER TEMP	This red LED flashes upon the power state has exceeded a critical temperature of 100°C (prewarning). This red LED lights up upon the power stage has exceeded a critical temperature of 120°C. The motor becomes switched off, until temperature falls below 110°C.
Current limit	OVER CURRENT	This red LED lights up upon the motor current has exceeded the current limit.
LED without defined function	LED1	This yellow LED is applicable and can be used customer specific.

Table 4.11: LEDs

4.1.6 Measuring points

MEASURING POINTS OF THE TMC603-EVAL

Measuring point	Label	Description
Charge pump supply voltage	VCP	Charge pump supply voltage. Provides high side driver supply.
Low side driver supply voltage	VLS	Low side driver supply voltage for driving low side gates.
Power supply +5V DC	+5V	Power supply, nom. +5V DC
Power supply +3.3V DC	+3.3V	Power supply, nom. +3.3V DC
Ground	GND	Power supply and signal ground. There are two ground pins for the oscilloscope.
Error output	ERROR	Error output (open drain). The TMC603 has three different sources for signaling an error: - Undervoltage of the low side supply - Undervoltage of the charge pump - Short to GND detection Upon any of these events the error output is pulled low.
Current output	CUR1/2/3	Output of current measurement amplifier (for phase 1 to 3). The output signal is centered to 1.65V.
Filter output	F1/2/3	Output of internal switched capacitor filter (for phase 1 to 3). This signal is available only if filter outputs of TMC603 are selected (refer chapter 8).
High side output	HS1/2/3	High side MOSFET driver output (for phase 1 to 3)
Bridge output	BM1/2/3	Sensing input for bridge output. Used for MOSFET control and current measurement. (for phase 1 to 3)
Low side output	LS1/2/3	Low side MOSFET driver output (for phase 1 to 3)

Table 4.12: Measuring points

5 TMCL™ overview

As with most TRINAMIC modules the software running on the STM32F ARM Cortex-M3 processor of the TMC603-EVAL consists of two parts, a boot loader and the firmware itself. Whereas the boot loader is installed during production and testing at TRINAMIC and remains – normally – untouched throughout the whole lifetime, the firmware can be updated by the user. The firmware shipped with this evaluation board is related to the standard TMCL™ firmware shipped with most of TRINAMIC modules with regard to protocol and commands. There are two software tools: the *TMCL-BLDC* and the *TMCL-IDE*. Whereas the TMCL-BLDC is used for testing different configurations in all modes of operation the TMCL-IDE is mainly designed for conceiving programs and firmware updates. New versions of the TMCL-BLDC and the TMCL-IDE can be downloaded free of charge from the TRINAMIC website (http://www.trinamic.com).

5.1 TMCL-BLDC evaluation tool

The *TMCL-BLDC* is a special tool for adjusting and testing settings of the TMC603-EVAL in all modes of operation. Use this software tool for finding initial settings. For a better understanding of the PID regulation and its backgrounds we recommend the study of chapter 8.

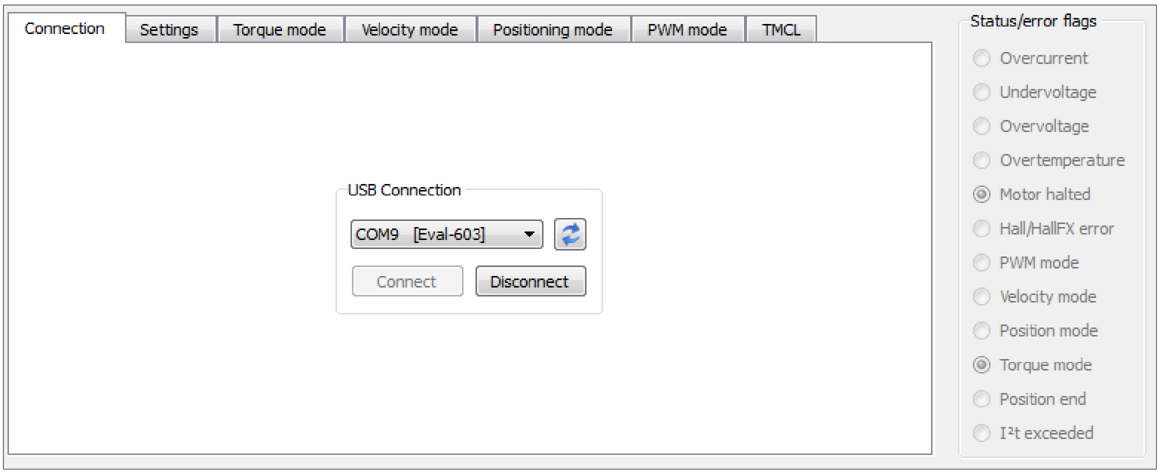


Figure 5.1: Connection tab of TMCL-BLDC

The first step is to connect the module. The structure of tabs of the TMCL-BLDC shows clearly how to proceed with the program. On the right side of the window the status and the error flags are shown. Below the input area are diagrams for velocity, position and current/PWM. These diagrams and the status/error information can be used for controlling new settings visually in order to identify best results as well as deficient settings. It is possible to scale each X-axis and Y-axis to get a comfortable report. These parts of the program are for diagnostic tasks and remain independent from the chosen tab, but the input area on top can be chosen by selecting a specific tab.

Please note that the status/error information and the curves have to be activated by starting the trace controller on the settings tab.

TMCL-BLDC tabs:

Settings
Torque mode
Velocity mode
Positioning mode
PWM mode
TMCL™

The *settings* tab is needed for general settings of the evaluation board. The next four tabs are designed for testing the four *modes of operation* (torque mode, velocity mode, positioning mode, PWM mode). On the tabs for *torque mode*, *velocity mode*, and *positioning mode* is a block for the PID control of the operation mode. Here are fields for filling in values of PID parameter set 1 and set 2. Set 1 is intended for lower velocities, set 2 for higher velocities. Below them is a value field for filling in the threshold between both PID sets. The TMC603-EVAL manages the switch-over between set 1 and set 2 smoothly avoiding disruptive factors. The last tab is used for controlling the evaluation board with *TMCL™ direct mode*. You can enter all TMCL™ commands as usually. The TMCL™ commands are explained in detail in chapter 0.

5.1.1 Settings tab

After connecting the module with the *connect* button you can choose the *settings* tab and fill in basic values: *motor settings*, *encoder settings* and *commutation mode*. All settings correspond to specific axis parameters, which are described in detail in chapter 6.

The *trace controller* has to be started for displaying the curves on the diagrams below. Clicking the *start* button of the trace controller enables the status/error flags, too.

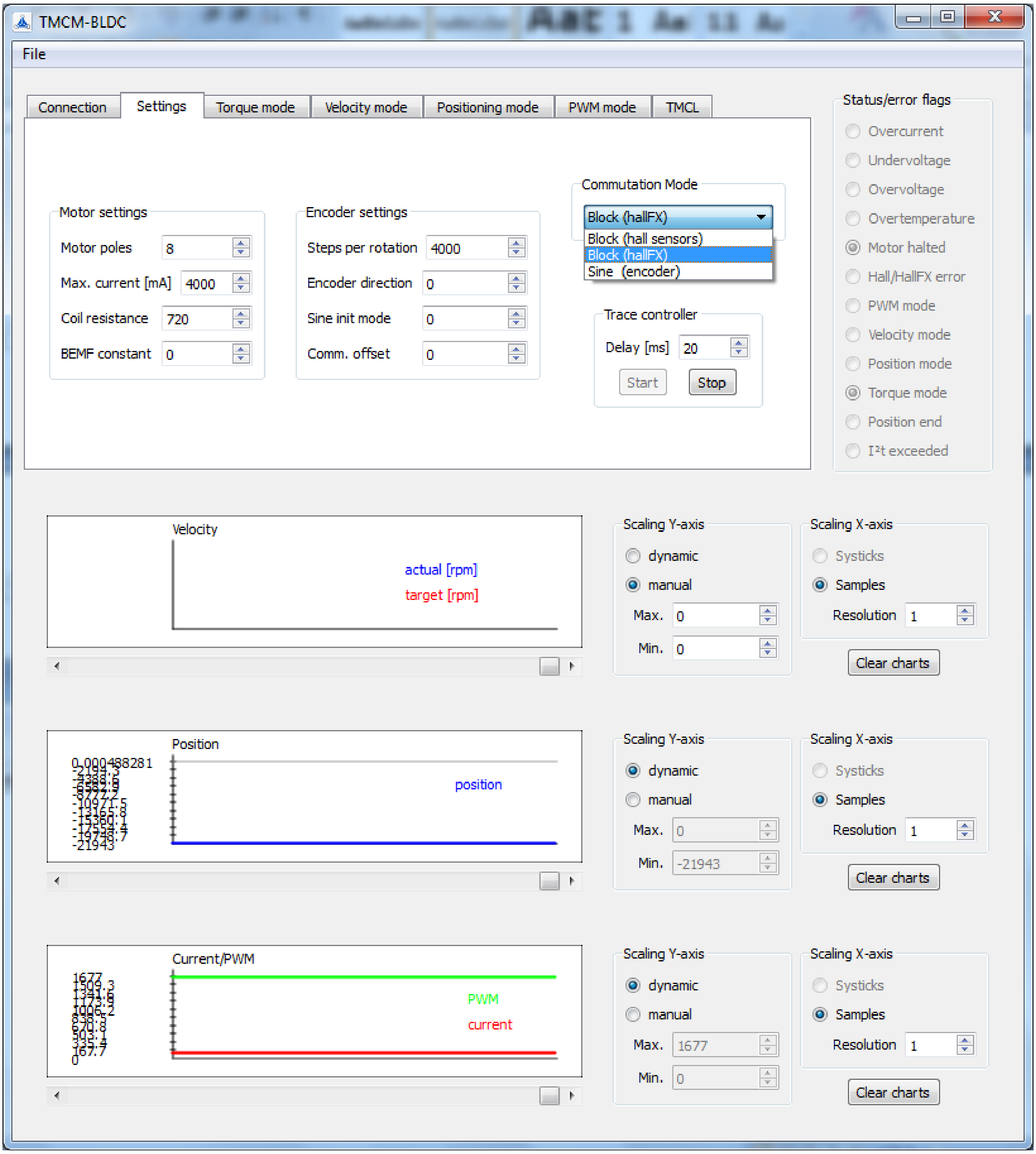


Figure 5.2: Settings tab of TMCL-BLDC

5.1.2 Torque mode tab

The *torque mode tab* offers the possibility to test different current settings and to evaluate the *current PID control* by choosing values for the P, I, and D parameters of set 1 and set 2. Further the I clipping value for each set can be chosen and the threshold for the switch-over between set 1 and set 2 can be set.

The drive can be started (in positive and negative direction) and stopped with the buttons in the *current control* field. The values can be calibrated on the fly while the drive is still active. The results will be shown immediately on the diagrams below.

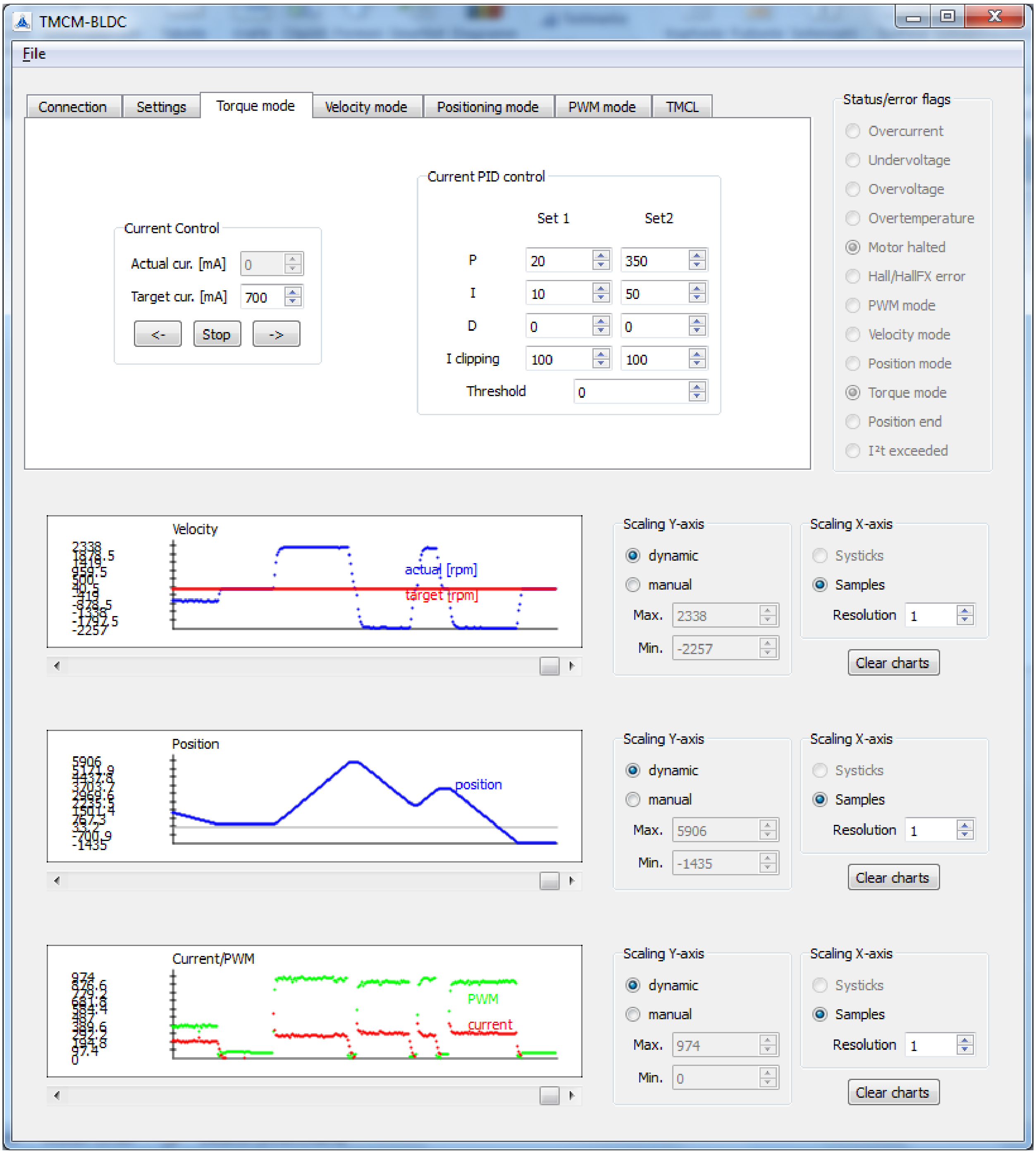


Figure 5.3: Torque mode tab of TMCL-BLDC

5.1.3 Velocity mode tab

The input area of the velocity tab has three parts: the *velocity ramp control*, the *velocity control* and the *velocity PID control*. In the middle of the input area is the *velocity control*, which is used to start the drive (positive and negative direction) in velocity mode with a chosen speed [rpm] or stop it. The *velocity ramp control* is needed for setting the maximum velocity [rpm] and the acceleration [rpm/s]. Further, the velocity ramp can be enabled by ticking the appropriate field. Disabling the velocity ramp leads to a hard stop.

On the right side is the *velocity PID control*. Here, the P, I, D, and I clipping parameter values for set 1 and set 2 as well as the threshold value for the switch-over between set 1 and set 2 can be set. The values can be calibrated on the fly while the drive is still active. The results will be shown immediately on the diagrams.

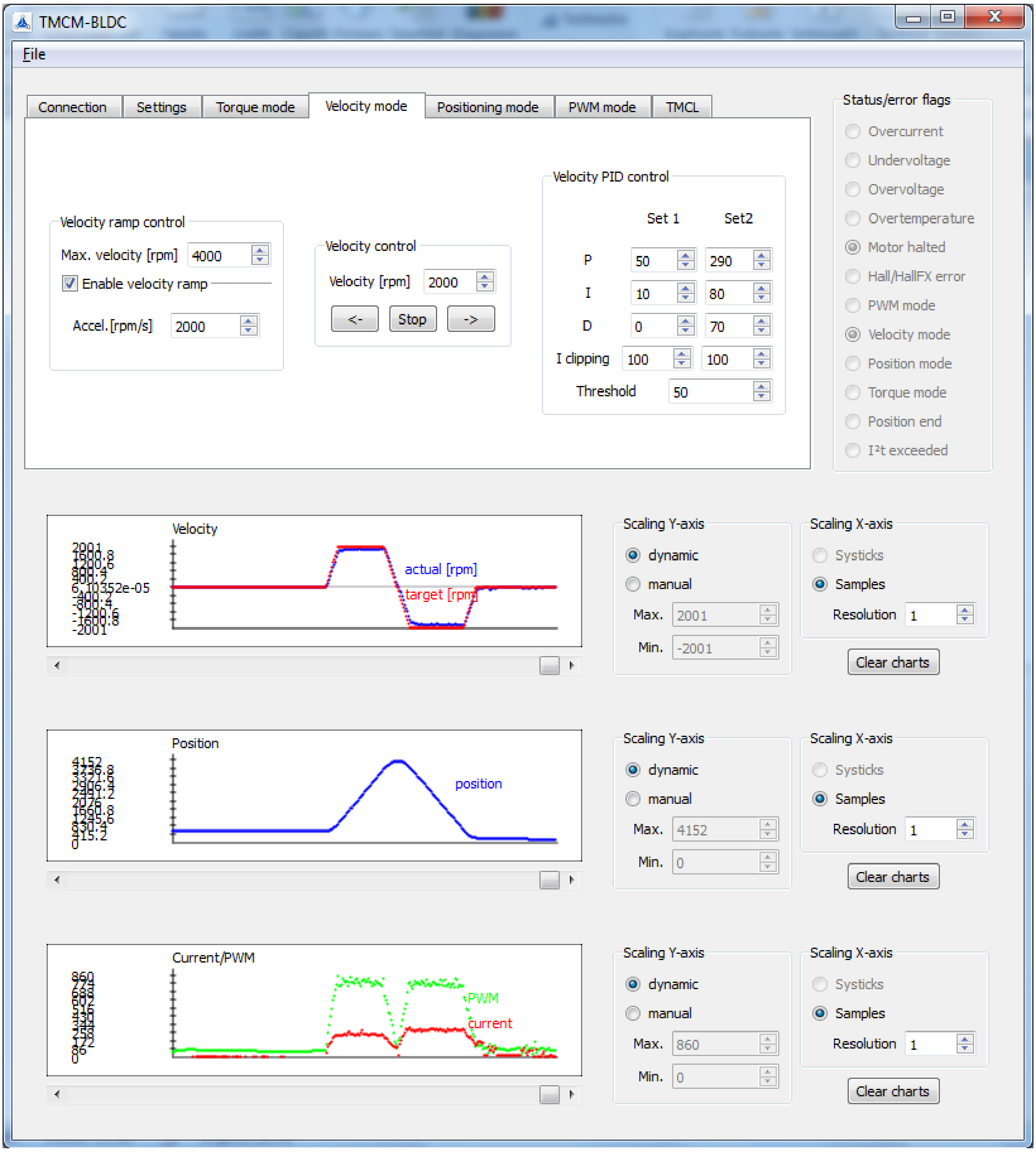


Figure 5.4: Velocity mode tab of TMCL-BLDC

5.1.4 Positioning mode tab

The input area of the positioning mode tab has three parts: the *velocity ramp control*, the *positioning control*, and the *positioning PID control*.

The velocity ramp control is the same as on the velocity mode tab. Maximum velocity and acceleration can be chosen and the velocity ramp can be enabled or disabled. In the middle of the positioning mode input area is the *positioning control* field. This is adequate designed to the TMCL™ command MVP (*move to position*). There are two possibilities to move in positive or negative direction: move absolutely or relatively to the actual position. The unit of the target position is *encoder steps per rotation* (operation with encoder, only). The unit for positioning with hall sensors is $\frac{6 \times motorpoles}{2}$ *steps per motor rotation*.

The button *clear* sets the counter for positioning to zero. *Clear on NULL* is used with encoder. The actual position is set to zero when crossing the next N channel.

On the right side of the positioning mode tab input area is the *position PID control*. Here, values of the P, I, D, and I clipping parameters (set 1 and set 2) for the position PID control can be filled in and the specific threshold between both sets can be set. The values can be calibrated on the fly while the drive is still active. The results will be shown immediately on the diagrams.

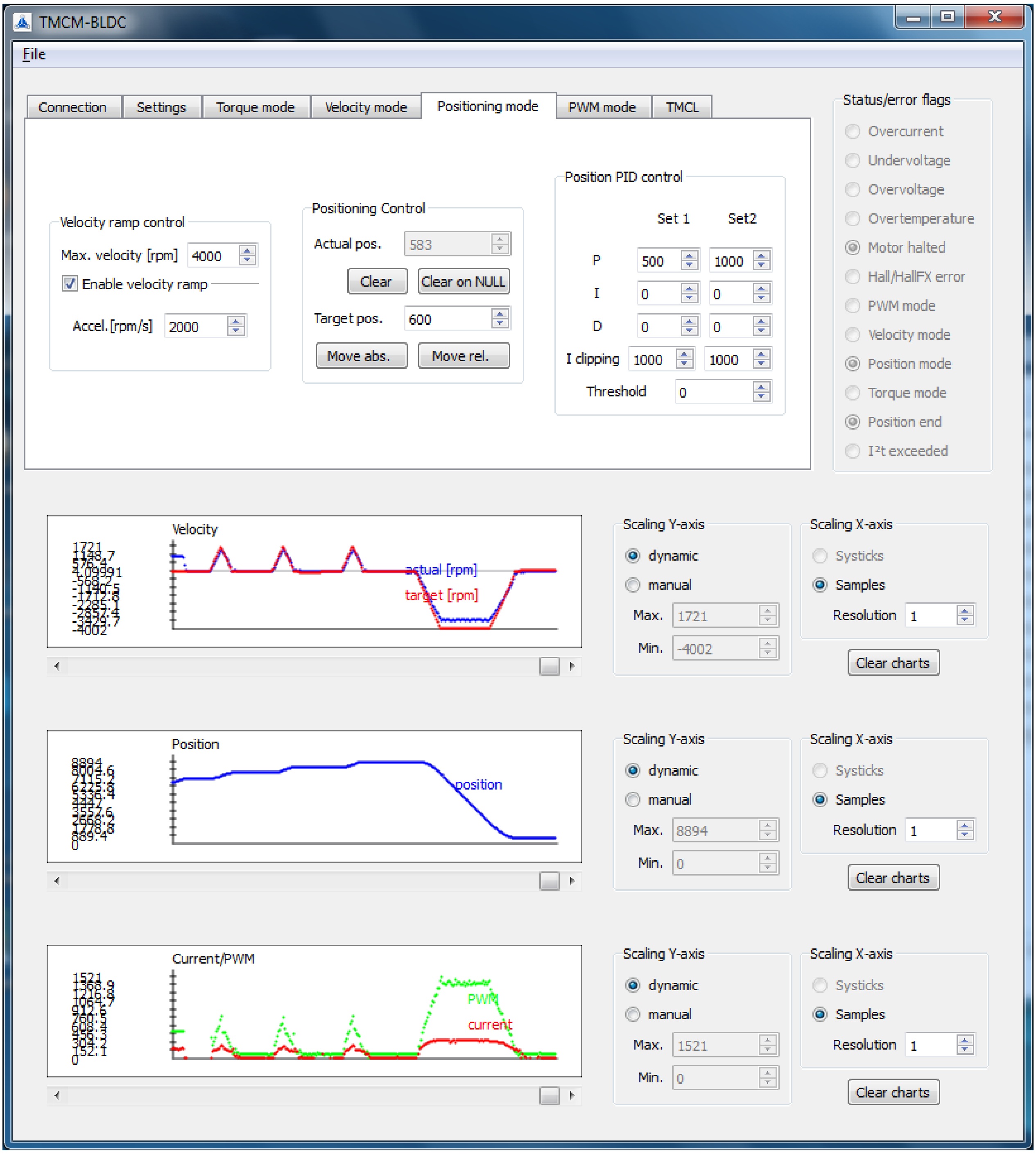


Figure 5.5: Positioning mode tab of TMCL-BLDC

5.1.5 PWM mode tab

On the PWM mode tab is the *PWM control* field, where a target PWM can be chosen. The drive can be started and stopped as on the other tabs with the appropriate buttons below the value field of the target PWM. The target PWM value can be changed on the fly while the drive is still active. The results will be shown immediately on the diagrams.

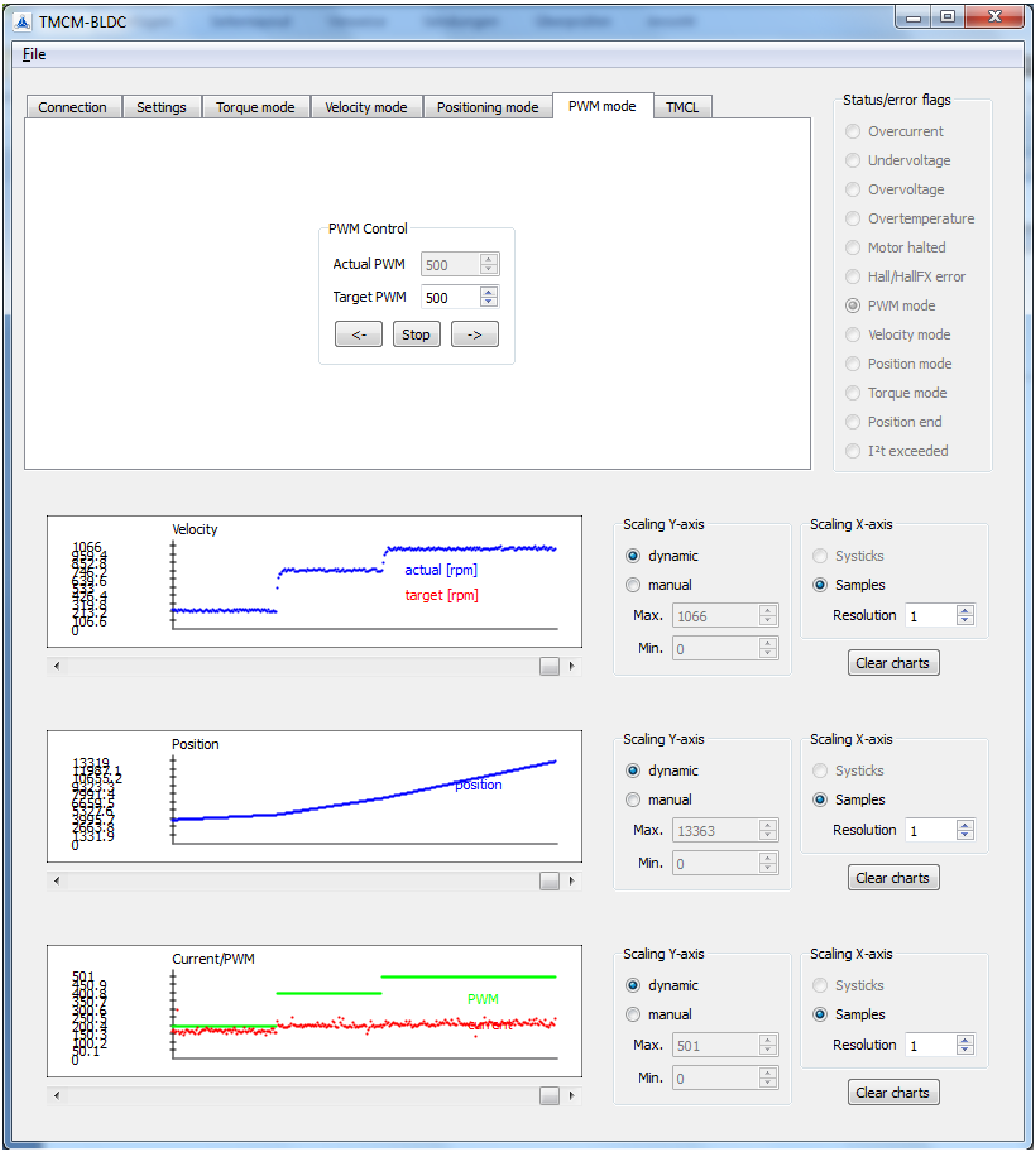


Figure 5.6: PWM mode tab of TMCL-BLDC

5.1.6 TMCL™ tab

The input area of the TMCL™ tab has the same structure as the appropriate window for *TMCL™ direct mode* of the TMCL-IDE. *Command number*, *type*, *motor/bank* and a chosen *value* can be set. By clicking the *Send* button the request will be sent to the evaluation board. Immediately the reply of the TMC603-EVAL will be displayed in the *Reply* field.

An overview of all TMCL™ commands is given in the following paragraphs. Please refer to the complete lists of axis parameters and global parameters in chapter 6, too.

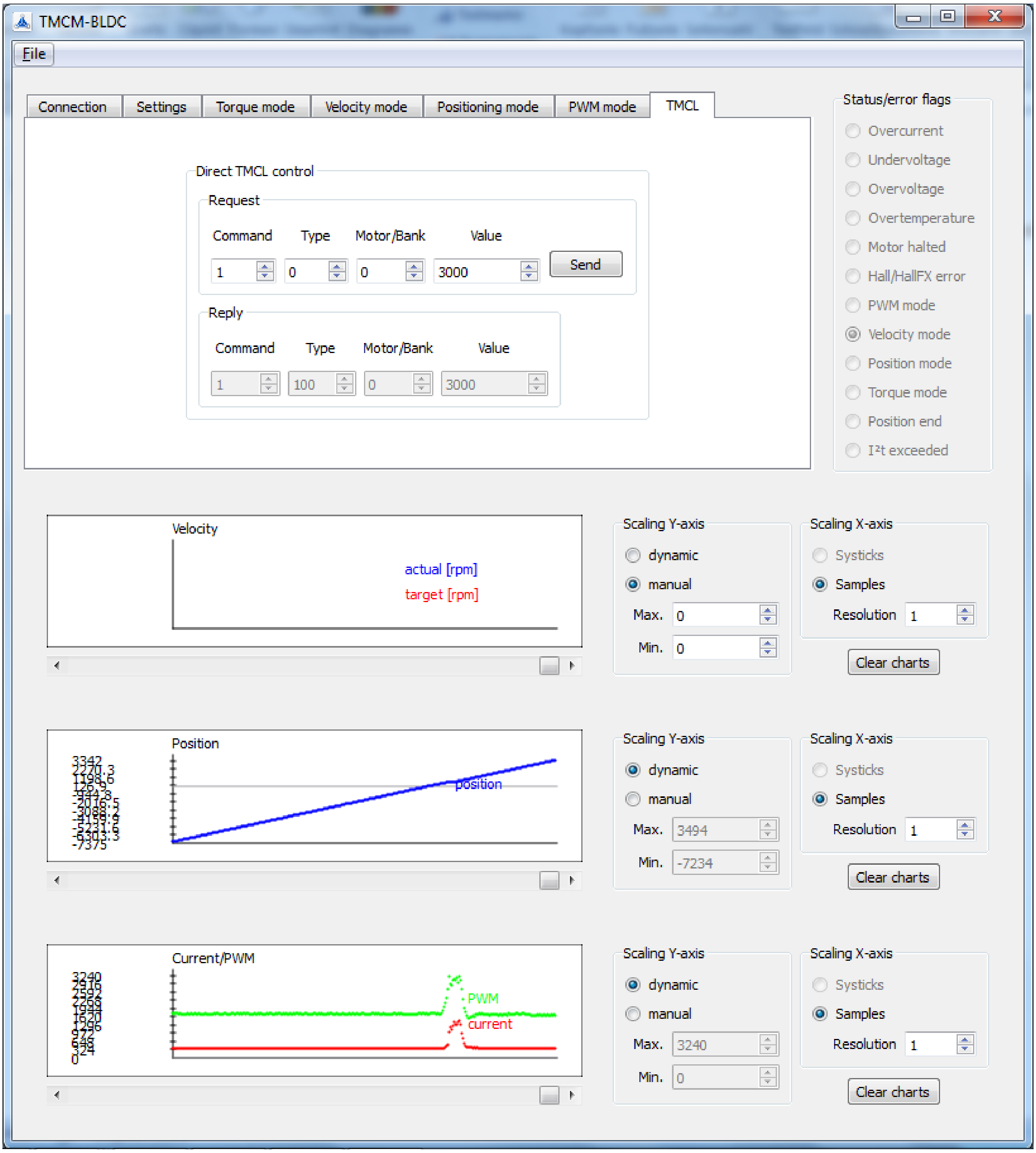


Figure 5.7: TMCL™ tab

5.1.7 File menu of TMCL-BLDC

The file menu of the TMCL-BLDC offers the possibility to import and to export settings. This is useful for transferring settings from one module to another. Settings can be exported (*Export settings to *.ini*) and afterwards imported to another module with the command *Import settings from *.ini*.

Further, it is useful to export evaluated adjustments of the TMCL-BLDC program to a TMCL script used later in the TMCL-IDE. Therefore choose *Export settings to TMCL*.

Certainly actual values can be stored and restored.

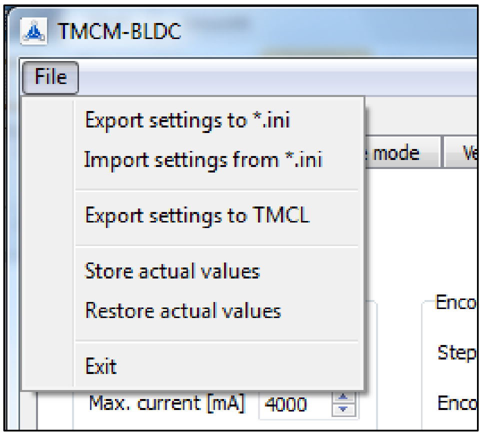


Figure 5.8: File menu of TMCL-BLDC

TRINAMIC offers different scripts (www.trinamic.com) for testing block hall, hallFX™, encoder, switches and the rotary switch:

- Block_Hall_torque_velocity_rotary_switch.tmc
- HallFXTest.tmc
- Sinus_Encoder_Limit_Switch_Test.tmc

5.2 TMCL-IDE

The TMCL-IDE is an integrated development environment mainly for developing standalone TMCL™ applications, but it also includes a function for using the TMCL™ commands in direct mode. The TMCL-IDE is a PC application running under Windows 95/98/NT/2000/XP/Vista/Windows 7.

For the TMC603-EVAL in particular the dialogue for configuring BLDC modules is important. Most other functions can be used, too. Please refer to the TMCL-IDE User Manual (www.trinamic.com) for detailed information about this.

Please be sure to always use the latest version of the TMCL-IDE as its functionality is being extended and improved constantly.

5.2.1 BLDC tool of the TMCL-IDE

The BLDC tool of the TMCL-IDE consists mainly of two parts: the left for adjusting axis parameters and testing them directly in praxis and the right for reporting generated values.

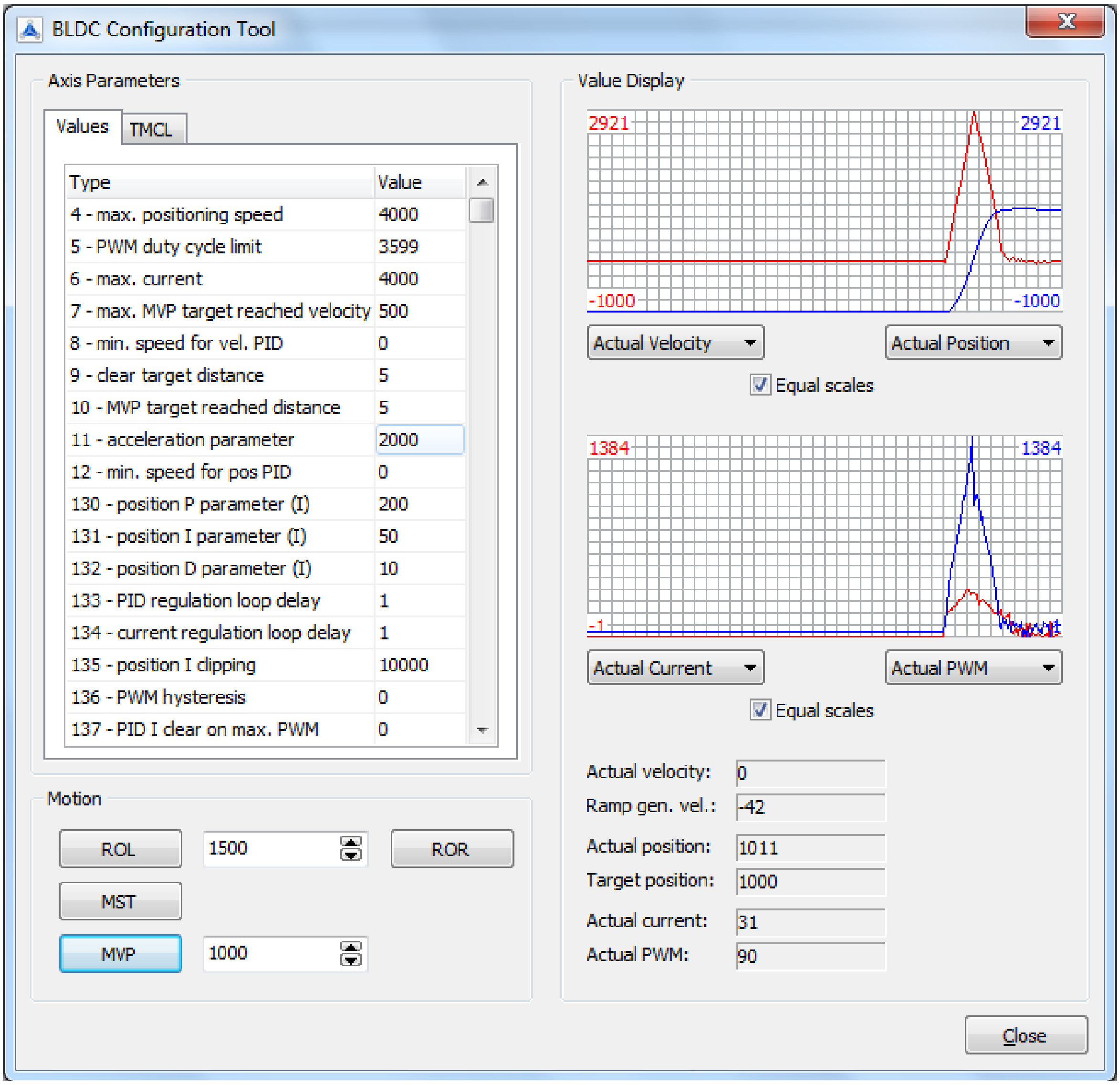


Figure 5.9: Configuration tool of the TMCL-IDE

5.2.1.1 Axis parameters and motion

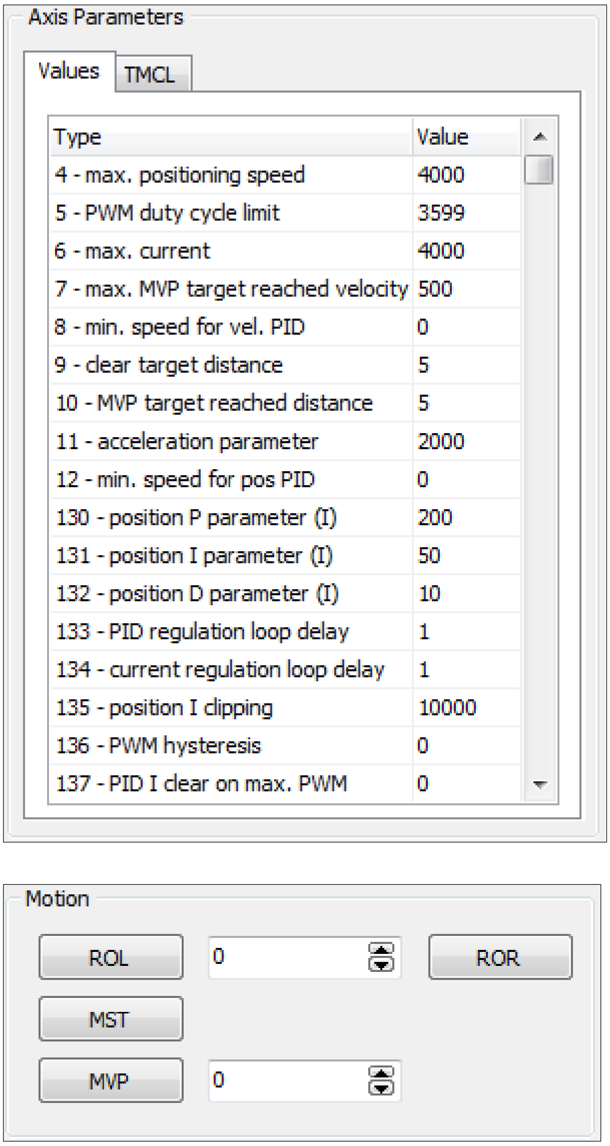


Figure 5.10: Values tab of the TMCL-IDE

On the left side of the BLDC tool are two tabs, which can be chosen:

- *Values*
- *TMCL*

The *values tab* offers a table with two columns: numbers of the TMCL™ axis parameters and appropriate descriptions on the left and given values on the right. The value of each axis parameter can be changed directly. If the overwritten value field is left the new value becomes valid. The TMCL™ command behind this action is *set axis parameter* (SAP). Immediately changes will be recognized and shown on the *TMCL™ tab*, where all SAP commands are displayed as TMCL™ mnemonics / text strings. The content of the TMCL™ tab can be marked and copied by clicking the *copy* button. Certainly it is possible to paste copied sections into the editor of the TMCL-IDE in order to create one's own application.

Below the two tabs of the *axis parameter* area is an input area for *motion* control. Using its functionality is important for testing the adjusted axis parameters directly in praxis. In the middle of this area are two value input fields: the above is for setting the speed for the commands *rotate left* (ROL) or *rotate right* (ROR). The value input field on the bottom is used for positioning. The TMCL™ command behind this action is termed *move to absolute position* (MVP ABS).

You will find further information about the TMCL™ commands in chapter 0.

5.2.1.2 Value display

While the TMCL™ BLDC tool is active the *value display* offers a smart visualization of all chosen adjustments. Changes of axis parameters and other commands will be shown immediately in graphics and table.

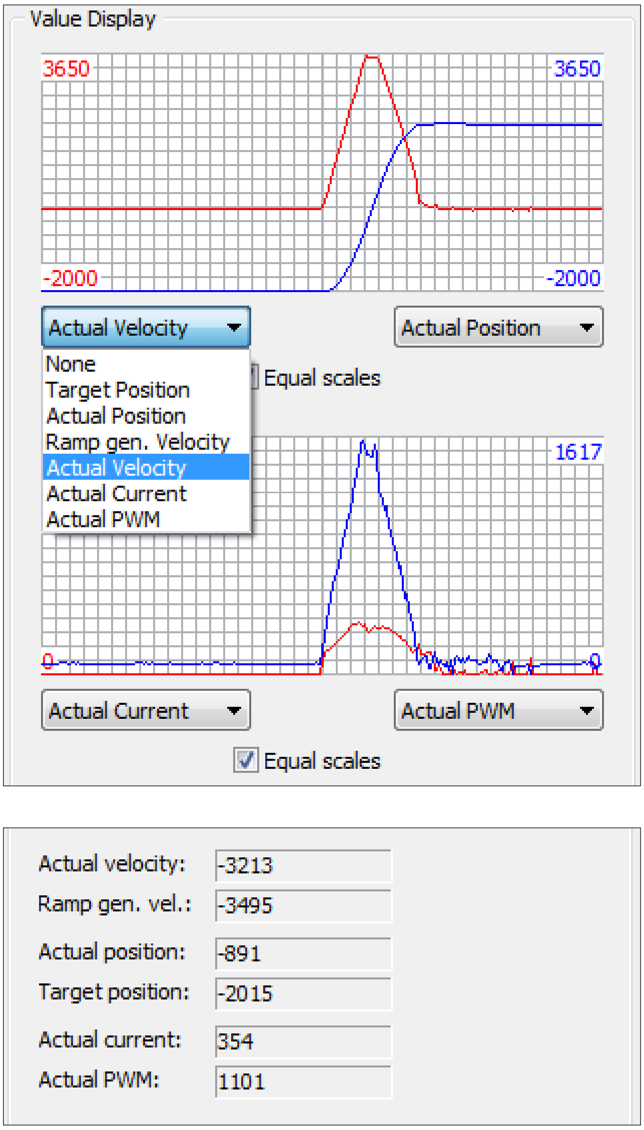


Figure 5.11: Value display of the TMCL-IDE

All parameter changes and commands which have been initiated with the help of the *values tab* and the *motion area* will be shown immediately on the screens and the accordant table of the *value display*. These tools offer read out values used for adjusting positioning operations, mainly. The TMCL™ command behind this is the *get axis parameter* command (GAP).

The *value display* area offers possibilities for changing its adjustments. Under each graphic are two buttons for choosing the aspects which have to be examined.

Select out of the following catalog:

- *None*
- *Target Position*
- *Actual Position*
- *Ramp. gen. Velocity*
- *Actual Velocity*
- *Actual Current*
- *Actual PWM*

At all, the curves of up to four aspects can be shown on the two screens at the same time.

Depending on the targets a customer is engaged with it might be helpful to equal the scales by ticking the appropriate fields (*default*).

5.3 TMCL™ command overview

In this section a short overview of the TMCL™ commands is given.

5.3.1 Motion commands

These commands control the motion of the motor. They are the most important commands and can be used in direct mode or in stand-alone mode.

Mnemonic	Command number	Description
ROR	1	Rotate right
ROL	2	Rotate left
MST	3	Motor stop
MVP	4	Move to position

Figure 5.12: Motion commands

5.3.2 Parameter commands

These commands are used to set, read and store axis parameters or global parameters. Axis parameters can be set independently for the axis, whereas global parameters control the behavior of the module itself. These commands can also be used in direct mode and in stand-alone mode.

Mnemonic	Command number	Description
SAP	5	Set axis parameter
GAP	6	Get axis parameter
STAP	7	Store axis parameter into EEPROM
RSAP	8	Restore axis parameter from EEPROM
SGP	9	Set global parameter
GGP	10	Get global parameter
STGP	11	Store global parameter into EEPROM
RSGP	12	Restore global parameter from EEPROM

Figure 5.13: Parameter commands

5.4 Commands

The module specific commands are explained in more detail on the following pages. They are listed according to their command number.

5.4.1 ROR (rotate right)

With this command the motor will be instructed to rotate with a specified velocity in *right* direction (increasing the position counter).

Internal function: First, velocity mode is selected. Then, the velocity value is transferred to axis parameter #2 (*target velocity*).

Related commands: ROL, MST, SAP, GAP

Mnemonic: ROR 0, <velocity>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
1	don't care	0	<velocity> -2147483648… +2147483647

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	1	don't care

Example:
Rotate right, velocity = 350
Mnemonic: ROR 0, 350

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$01	$00	$00	$00	$00	$01	$5e

5.4.2 ROL (rotate left)

With this command the motor will be instructed to rotate with a specified velocity (opposite direction compared to ROR, decreasing the position counter).

Internal function: First, velocity mode is selected. Then, the velocity value is transferred to axis parameter #2 (*target velocity*).

Related commands: ROR, MST, SAP, GAP

Mnemonic: ROL 0, <velocity>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
2	don't care	0	<velocity> -2147483648… +2147483647

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	2	don't care

Example:

Rotate left, velocity = 1200
Mnemonic: ROL 0, 1200

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$02	$00	$00	$00	$00	$04	$b0

5.4.3 MST (motor stop)

With this command the motor will be instructed to stop.

Internal function: The axis parameter *target velocity* is set to zero.

Related commands: ROL, ROR, SAP, GAP

Mnemonic: MST 0

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
3	don't care	0	don't care

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	3	don't care

Example:

Stop motor
Mnemonic: MST 0

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$03	$00	$00	$00	$00	$00	$00

5.4.4 MVP (move to position)

With this command the motor will be instructed to move to a specified relative or absolute position. It will use the acceleration/deceleration ramp and the positioning speed programmed into the unit. This command is non-blocking (like all commands) – that is, a reply will be sent immediately after command interpretation and initialization of the motion controller. Further commands may follow without waiting for the motor reaching its end position. The maximum velocity and acceleration are defined by axis parameters #4 and #11.

Two operation types are available:

- Moving to an absolute position in the range from -2147483648… +2147483647.
- Starting a relative movement by means of an offset to the actual position. In this case, the new resulting position value must not exceed the above mentioned limits, too.

Internal function: A new position value is transferred to the axis parameter #0 *target position*.

Related commands: SAP, GAP, and MST

Mnemonic: MVP <ABS|REL>, 0, <position|offset number>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
4	0 ABS – absolute	0	<position>
	1 REL – relative	0	<offset>

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	4	don't care

Example MVP ABS:

Move motor to (absolute) position 9000
Mnemonic: MVP ABS, 0, 9000

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$04	$00	$00	$00	$00	$23	$28

Example MVP REL:

Move motor from current position 1000 steps backward (move relative -1000)
Mnemonic: MVP REL, 0, -1000

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$00	$04	$01	$00	$ff	$ff	$fc	$18

5.4.5 SAP (set axis parameter)

With this command most of the motion control parameters of the module can be specified. The settings will be stored in SRAM and therefore are volatile. That is, information will be lost after power off. ***Please use command STAP (store axis parameter) in order to store any setting permanently.***

Related commands: GAP, STAP, and RSAP

Mnemonic: SAP <parameter number>, 0, <value>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
5	<parameter number>	0	<value>

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	5	don't care

A list of all parameters which can be used for the SAP command is shown in section 6.

Example:

Set the absolute maximum current to 200mA
Mnemonic: SAP 6, 0, 200

Binary:

Byte Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$05	$00	$00	$00	$00	$00	$c8

5.4.6 GAP (get axis parameter)

Most parameters of the TMC603-EVAL can be adjusted individually. With the GAP command they can be read out.

Related commands: SAP, STAP, and RSAP

Mnemonic: GAP <parameter number>, 0

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
6	<parameter number>	0	don't care

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	6	don't care

A list of all parameters which can be used for the GAP command is shown in section 6.

Example:

Get the actual position of motor
Mnemonic: GAP 1, 0

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$06	$01	$00	$00	$00	$00	$00

Reply:

Byte Index	0	1	2	3	4	5	6	7
Function	Host-address	Target-address	Status	Instruction	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$00	$00	$64	$06	$00	$00	$02	$c7

⇨ **status=no error, position=711**

5.4.7 STAP (store axis parameter)

The STAP command stores an axis parameter previously set with a *Set Axis Parameter command (SAP)* permanently. Most parameters are automatically restored after power up (refer to axis parameter list in chapter 6).

Internal function: An axis parameter value stored in SRAM will be transferred to EEPROM and loaded from EEPORM after next power up.

Related commands: SAP, RSAP, and GAP

Mnemonic: STAP <parameter number>, 0

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
7	<parameter number>	0	don't care*

*** The value operand of this function has no effect. Instead, the currently used value (e.g. selected by SAP) is saved.**

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	7	don't care

A list of all parameters which can be used for the STAP command is shown in section 11.9.

Example:

Store the maximum speed
Mnemonic: STAP 4, 0

Binary:

Byte Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$07	$04	$00	$00	$00	$00	$00

Note: The STAP command will not have any effect when the configuration EEPROM is locked. The error code 5 (configuration EEPROM locked) will be returned in this case.

5.4.8 RSAP (restore axis parameter)

For all configuration related axis parameters non-volatile memory locations are provided. By default, most parameters are automatically restored after power up (refer to axis parameter list in chapter 6). A single parameter that has been changed before can be reset by this instruction also.

Internal function: The specified parameter is copied from the configuration EEPROM memory to its RAM location.

Related commands: SAP, STAP, and GAP

Mnemonic: RSAP <parameter number>, 0

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
8	<parameter number>	0	don't care

Reply in direct mode:

STATUS	COMMAND	VALUE
100 – OK	8	don't care

A list of all parameters which can be used for the RSAP command is shown in section 6.

Example:

Restore the maximum current
Mnemonic: RSAP 6, 0

Binary:

Byte Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$08	$06	$00	$00	$00	$00	$00

5.4.9 SGP (set global parameter)

Global parameters are related to the host interface, peripherals or application specific variables. The different groups of these parameters are organized in *banks* to allow a larger total number for future products. Currently, only bank 0 and 1 are used for global parameters, and only bank 2 is intended to use for user variables. Lists of global parameters are available in section 0.

Related commands: GGP, STGP, RSGP

Mnemonic: SGP <parameter number>, <bank number>, <value>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
9	<parameter number>	<bank number>	<value>

Reply in direct mode:

STATUS	VALUE
100 – OK	don't care

Example: set variable 0 at bank 2 to 100
Mnemonic: SGP, 0, 2, 100

Binary:

Byte Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$09	$00	$02	$00	$00	$00	$64

5.4.10 GGP (get global parameter)

All global parameters can be read with this function. Lists of global parameters are available in section 0.

Related commands: SGP, STGP, RSGP

Mnemonic: GGP <parameter number>, <bank number>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
10	<parameter number>	<bank number>	don't care

Reply in direct mode:

STATUS	VALUE
100 – OK	<value>

Example: get variable 0 from bank 2
Mnemonic: GGP, 0, 2

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$0a	$00	$02	$00	$00	$00	$00

5.4.11 STGP (store global parameter)

Some global parameters are located in RAM memory, so modifications are lost at power down. This instruction copies a value from its RAM location to the configuration EEPROM and enables permanent storing. Most parameters are automatically restored after power up. Lists of global parameters are available in section 0.

Related commands: SGP, GGP, RSGP

Mnemonic: STGP <parameter number>, <bank number>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
11	<parameter number>	<bank number>	don't care

Reply in direct mode:

STATUS	VALUE
100 – OK	don't care

Example: copy variable 0 at bank 2 to the configuration EEPROM
Mnemonic: STGP, 0, 2

Binary:

Byte Index	0	1	2	3	4	5	6	7
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$0b	$00	$02	$00	$00	$00	$00

5.4.12 RSGP (restore global parameter)

This instruction copies a value from the configuration EEPROM to its RAM location and so recovers the permanently stored value of a RAM-located parameter. Most parameters are automatically restored after power up. Lists of global parameters are available in section 0.

Related commands: SGP, GGP, STGP

Mnemonic: RSGP <parameter number>, <bank number>

Binary representation:

COMMAND	TYPE	MOT/BANK	VALUE
12	<parameter number>	<bank number>	don't care

Reply in direct mode:

STATUS	VALUE
100 – OK	don't care

Example: copy variable 0 at bank 2 from the configuration EEPROM to the RAM location
Mnemonic: RSGP, 0, 2

Binary:

Byte Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Function	Target-address	Instruction Number	Type	Motor/ Bank	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0
Value (hex)	$01	$0c	$00	$02	$00	$00	$00	$00

5.4.13 Customer specific TMCL™ command extension (UF0… UF7/user function)

The user definable functions UF0… UF7 are predefined, functions without topic for user specific purposes. Contact TRINAMIC for customer specific programming of these functions.

Internal function: Call user specific functions implemented in *C* by TRINAMIC.

Related commands: none

Mnemonic: UF0… UF7

Binary representation:

INSTRUCTION NO.	TYPE	MOT/BANK	VALUE
64… 71	(user defined)	(user defined)	(user defined)

Reply in direct mode:

Byte Index	0	1	2	3	4	5	6	7	8
Function	Target-address	Target-address	Status	Instruction	Operand Byte3	Operand Byte2	Operand Byte1	Operand Byte0	Checksum
Value (hex)	$02	$01	(user defined)	64… 71	(user defined)	(user defined)	(user defined)	(user defined)	<checksum>

5.4.14 TMCL™ control functions

There are several TMCL™ control functions, but for the user is only command 136 interesting. Other control functions can be used with axis parameters.

Command	Type	Parameter	Description	Access
136	0 – string 1 – binary	Firmware version	Get the module type and firmware revision as a string or in binary format. (*Motor/Bank* and *Value* are ignored.)	read

Type set to 0 - reply as a string:

Byte index	Contents
1	Host Address
2… 9	Version string (8 characters, e.g. 603V1.46)

There is no checksum in this reply format!

Type set to 1 - version number in binary format:
Please use the normal reply format. The version number is output in the *value* field of the reply in the following way:

Byte index in value field	Contents
1	Version number, low byte
2	Version number, high byte
3	Type number, low byte (currently not used)
4	Type number, high byte (currently not used)

6 Axis parameter overview (SAP, GAP, STAP, RSAP)

The following section describes all axis parameters that can be used with the SAP, GAP, STAP and RSAP commands.

Meaning of the letters in column Access:

R = readable (GAP)
W = writable (SAP)
E = stored and automatically restored from EEPROM after reset or power-on
P = protected

Number	Axis Parameter	Description	Range [Unit]	Access
0	Target position	The target position of a currently executed ramp.	-2147483648… +2147483647	RW
1	Actual position	Set/get the position counter without moving the motor.	-2147483648… +2147483647	RW
2	Target speed	Set/get the desired target velocity.	-2147483648… +2147483647 [rpm]	RW
3	Actual speed	The actual velocity of the motor.	-2147483648… +2147483647 [rpm]	R
4	Max. ramp velocity	The maximum velocity used for velocity ramp in velocity mode and positioning mode. Set this value to a realistic velocity which the motor can reach!	-2147483648… +2147483647 [rpm]	RWE
5	PWM limit	Set/get PWM limit (0%… 100%).	0… 3599	RWE
6	Max current	Set/get the max allowed motor current. *This value can be temporarily exceeded marginal due to the operation of the current regulator.	0… +4294967295 [mA]	RWE
7	MVP Target reached velocity	Maximum velocity at which end position can be set. Prevents issuing of end position when the target is passed at high velocity.	-2147483648… +2147483647 [rpm]	RWE
8	Threshold speed for velocity PID	Threshold speed for velocity regulation to switch between first and second velocity PID parameter set.	-2147483648… +2147483647 [rpm]	RWE
9	Motor halted velocity	If the actual speed is below this value the motor halted flag will be set.	-2147483648… +2147483647 [rpm]	RWE
10	MVP target reached distance	Maximum distance at which the position end flag is set.	-2147483648… +2147483647	RWE
11	Acceleration	Acceleration parameter for ROL, ROR, and the velocity ramp of MVP.	-2147483648… +2147483647 [RPM/s]	RWE
12	Threshold speed for position PID	Threshold speed for position regulation to switch between first and second position PID parameter set.	-2147483648… +2147483647 [rpm]	RWE
13	Ramp generator speed	The actual speed of the velocity ramp used for positioning and velocity mode.	-2147483648… +2147483647 [rpm]	R
14	velocity threshold for hallFX™	Velocity to switch from controlled to hallFX™ mode. Set this value to a realistic velocity which the motor can reach in controlled mode!	-2147483648… +2147483647 [rpm]	RWE
20	Switch 2 active	0: inactive 1: active	0/1	R
21	Switch 1 active	0: inactive 1: active	0/1	R
22	Potentiometer	The value of the analog input.	0…4095	R

Number	Axis Parameter	Description	Range [Unit]	Access
25	Thermal winding time constant	Thermal winding time constant for the used motor. Used for I^2t monitoring.	0… +4294967295 [ms]	RWE
26	I^2t limit	An actual I^2t sum that exceeds this limit leads to increasing the I^2t exceed counter.	0… +4294967295	RWE
27	I^2t sum	Actual sum of the I^2t monitor.	0… +4294967295	R
28	I^2t exceed counter	Counts how often an I^2t sum was higher than the I^2t limit.	0… +4294967295	RWE
29	Clear I^2t exceeded flag	Clear the flag that indicates that the I^2t sum has exceeded the I^2t limit.	(ignored)	W
30	Minute counter	Counts the module operational time in minutes.	0… +4294967295 [min]	RWE
130	P parameter for position PID (I)	P parameter of position PID regulator (first parameter set)	-2147483648… +2147483647	RWE
131	I parameter for position PID (I)	I parameter of position PID regulator (first parameter set)	-2147483648… +2147483647	RWE
132	D parameter for position PID (I)	D parameter of position PID regulator (first parameter set)	-2147483648… +2147483647	RWE
133	PID regulation loop delay	PID calculation delay. Set PID operational frequency.	0… +4294967295 [ms]	RWE
134	Current regulation loop delay	Delay of current limitation algorithm / PID current regulator.	0… +4294967295 [50µs]	RWE
135	I-Clipping parameter for position PID (I)	I-Clipping parameter of position PID regulator (first parameter set) (A too high value causes overshooting at positioning mode.)	-2147483648… +2147483647	RWE
136	PWM-Hysteresis	Compensates dead time of PWM and motor friction.	0… +4294967295	RWE
140	P parameter for velocity PID (I)	P parameter of velocity PID regulator (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
141	I parameter for velocity PID (I)	I parameter of velocity PID regulator (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
142	D parameter for velocity PID (I)	D parameter of velocity PID regulator (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
143	I-Clipping parameter for velocity PID (I)	I-Clipping parameter of velocity PID (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
146	Activate ramp	1: Activate velocity ramp generator for position PID control. Allows usage of acceleration and positioning velocity for MVP command.	0/1	RWE
150	Actual motor current	Get actual motor current.	-2147483648… +2147483647 [mA]	R
151	Actual voltage	Actual supply voltage.	0… +4294967295	R
152	Actual driver temperature	Actual temperature of the motor driver.	0… +4294967295	R
153	Actual PWM duty cycle	Get actual PWM duty cycle.	-3599… +3599	R
154	Target PWM	Get desired target PWM or set target PWM to activate PWM regulation mode. (+= turn motor in right direction; -= turn motor in left direction)	-3599… +3599	RW
155	Target current	Get desired target current or set target current to activate current regulation mode. (+= turn motor in right direction; -= turn motor in left direction)	-2147483648… +2147483647 [mA]	RW

Number	Axis Parameter	Description	Range [Unit]	Access
156	Error/Status flags	Bit 0: Overcurrent flag. This flag is set if overcurrent limit is exceeded. Bit 1: Undervoltage flag. This flag is set if supply voltage to low for motor operation. Bit 2: Overvoltage flag. This flag is set if the motor becomes switched off due to overvoltage. Bit 3: Overtemperature flag. This flag is set if overtemperature limit is exceeded. Bit 4: Motor halted flag. This flag is set if motor has been switched off. Bit 5: Hall error flag. This flag is set upon a hall error. Bit 6: unused Bit 7: unused Bit 8: PWM mode active flag. Bit 9: Velocity mode active flag Bit 10: Position mode active flag. Bit 11: Torque mode active flag. Bit 12: unused Bit 13: unused Bit 14: Position end flag. This flag is set if the motor has been stopped at the target position. Bit 15: unused Bit 16: unused for TMC603-EVAL (value should be 0 or 1) Bit 17: I^2t exceeded flag. This flag is set if the I^2t sum exceeded the I^2t limit of the motor. (reset by SAP 29 after the time specified by the I^2t thermal winding time constant) *Flag 0 to 15 are automatically reset. Only flag 16 and 17 must be cleared manually.*	0…+4294967295	R
159	Commutation mode	0: Block commutation with hall sensors 1: Sensorless block commutation (hallFX™) 3: Sine commutation with encoder	0, 1, 3	RWE
160	Re-Initialization of Sine	0: sine commutation is still re-initializing 1: sine commutation is re-initialized Attention: Depending on initialization mode, stop motor before issuing this command!	0/1	RW
161	Encoder set NULL	1: set position counter to zero at next N channel event.	0/1	RWE
162	Switch set NULL	1: set position counter to zero at next switch event.	0/1	RWE
163	Encoder clear set NULL	1: set position counter to zero only once 0: always at an N channel event, respectively switch event.	0/1	RWE

Number	Axis Parameter	Description	Range [Unit]	Access
164	Activate stop switch	Bit 0 – Left stop switch enable – When this bit is set the motor will be stopped if it is moving in negative direction and the left stop switch input becomes active Bit 1 – Right stop switch enable – When this bit is set the motor will be stopped if it is moving in positive direction and the right stop switch input becomes active Please see parameter 166 for selecting the stop switch input polarity.	0… 3	RWE
165	Actual encoder commutation offset	This value represents the internal commutation offset. (0 … max. encoder steps per rotation)	-2147483648… +2147483647	RWE
166	Stop switch polarity	Bit 0 – Left stop switch polarity – Bit set: Left stop switch input is high active Bit clear: Left stop switch input is low active Bit 1 – Right stop switch polarity – Bit set: Right stop switch input is high active Bit clear: Right stop switch input is low active	0… 3	RWE
167	Block PWM scheme	0: PWM chopper on high side, HI on low side 1: PWM chopper on low side, HI on high 2: PWM chopper on low side and high side	-128… +127	RWE
168	P parameter for current PID (I)	P parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
169	I parameter for current PID (I)	I parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
170	D parameter for current PID (I)	D parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
171	I-Clipping parameter for current PID (I)	I-Clipping parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
172	P parameter for current PID (II)	P parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
173	I parameter for current PID (II)	I parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
174	D parameter for current PID (II)	D parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
175	I-Clipping parameter for current PID (II)	I-Clipping parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
176	Threshold speed for current PID	Threshold speed for current regulation to switch between first and second current PID parameter set.	-2147483648… +2147483647 [rpm]	RWE

Number	Axis Parameter	Description	Range [Unit]	Access
177	Start current	Motor current for controlled commutation. This parameter is used in commutation mode 1, 4, 5 and in initialization of sine.	0… +4294967295 [mA]	RWE
200	Current PID error	Actual error of current PID regulator	-2147483648… +2147483647	R
201	Current PID error sum	Sum of errors of current PID regulator	-2147483648… +2147483647	R
209	Actual encoder position	Actual encoder position / counter value	-2147483648… +2147483647	R
226	Position PID error	Actual error of position PID regulator	-2147483648… +2147483647	R
227	Position PID error sum	Sum of errors of position PID regulator	-2147483648… +2147483647	R
228	Velocity PID error	Actual error of velocity PID regulator	-2147483648… +2147483647	R
229	Velocity PID error sum	Sum of errors of velocity PID regulator	-2147483648… +2147483647	R
230	P parameter for position PID (II)	P parameter of position PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
231	I parameter for position PID (II)	I parameter of position PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
232	D parameter for position PID (II)	D parameter of position PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
233	I-Clipping parameter for position PID (II)	I-Clipping parameter of position PID regulator. (second parameter set) (A too high value causes overshooting at positioning mode.)	-2147483648… +2147483647	RWE
234	P parameter for velocity PID (II)	P parameter of velocity PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
235	I parameter for velocity PID (II)	I parameter of velocity PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
236	D parameter for velocity PID (II)	D parameter of velocity PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
237	I-Clipping parameter for velocity PID (II)	I-Clipping parameter of velocity PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
238	Mass inertia constant	Mass inertia constant. Compensates the rotor inertia of the motor.	-2147483648… +2147483647	RWE
239	BEMF constant	BEMF constant of the motor. Used for current, position, and velocity regulation. Feed forward control for current, position, and velocity regulation is disabled if BEMF constant is set to zero.	-2147483648… +2147483647 [rpm/(10V)]	RWE
240	Motor coil resistance	Resistance of motor coil. Used for current, position, and velocity regulation.	-2147483648… +2147483647 [mΩ]	RWE
241	Init sine speed	Velocity for sine initialization. A positive sign initializes in right direction, a negative sign in left motor direction.	-32768… +32767 [rpm]	RWE
242	Init sine block offset CW	This parameter helps to tune hall sensor based initialization in a way, that the motor has the same velocity for left and right turn. It compensates for tolerance and hysteresis of the hall sensors. It is added to the Commutation offset upon CW turn initialization.	-32768… +32767	RWE

Number	Axis Parameter	Description	Range [Unit]	Access
243	Init sine block offset CCW	This parameter helps to tune hall sensor based initialization in a way, that the motor has the same velocity for left and right turn. It compensates for tolerance and hysteresis of the hall sensors. It is added to the Commutation offset upon CCW turn initialization.	-32768… +32767	RWE
244	Init sine delay	Duration for sine initialization sequence. This parameter should be set in a way, that the motor has stopped mechanical oscillations after the specified time.	-32768… +32767 [ms]	RWE
245	Overvoltage protection	1: Enable overvoltage protection.	0/1	RWE
247	Sine Compensation Factor	Compensates the propagation delay of the MPU	0… +255	RWE
249	Init sine mode	0: Initialization in controlled sine commutation (determines the encoder offset) 1: Initialization in block commutation using hall sensors 2: Initialization in controlled sine commutation (use the previous set encoder offset)	-128… +127	RWE
250	Encoder steps	Encoder steps per rotation.	0… +4294967295	RWE
251	Encoder direction	Set the encoder direction in a way, that ROR increases position counter.	0/1	RWE
253	Number of motor poles	Number of motor poles.	+2… +254	RWE
254	Hall sensor invert	1: Hall sensor invert. Invert the hall scheme, e.g. used by some Maxon motors.	0/1	RWE

6.1 Axis parameter sorted by functionality

The following section describes all axis parameters that can be used with the SAP, GAP, STAP and RSAP commands.

Meaning of the letters in column Access:

R = readable (GAP)
W = writable (SAP)
E = stored and automatically restored from EEPROM after reset or power-on

MOTOR/MODULE SETTINGS

Number	Axis Parameter	Description	Range [Unit]	Access
253	Number of motor poles	Number of motor poles.	+2… +254	RWE
5	PWM limit	Set/get PWM limit (0%… 100%).	0… 3599	RWE
239	BEMF constant	BEMF constant of the motor. Used for current, position, and velocity regulation. Feed forward control for current, position, and velocity regulation is disabled if BEMF constant is set to zero.	-2147483648… +2147483647 [rpm/(10V)]	RWE
240	Motor coil resistance	Resistance of motor coil. Used for current, position, and velocity regulation.	-2147483648… +2147483647 [mΩ]	RWE
238	Mass inertia constant	Mass inertia constant. Compensates the rotor inertia of the motor.	-2147483648… +2147483647	RWE
136	PWM-Hysteresis	Compensates dead time of PWM and motor friction.	0… +4294967295	RWE
25	Thermal winding time constant	Thermal winding time constant for the used motor. Used for I^2t monitoring.	0… +4294967295 [ms]	RWE
26	I^2t limit	An actual I^2t sum that exceeds this limit leads to increasing the I^2t exceed counter.	0… +4294967295	RWE
27	I^2t sum	Actual sum of the I^2t monitor.	0… +4294967295	R
28	I^2t exceed counter	Counts how often an I^2t sum was higher than the I^2t limit.	0… +4294967295	RWE
29	Clear I^2t exceeded flag	Clear the flag that indicates that the I^2t sum has exceeded the I^2t limit.	(ignored)	W
30	Minute counter	Counts the module operational time in minutes.	0… +4294967295 [min]	RWE
245	Overvoltage protection	1: Enable overvoltage protection.	0/1	RWE

ENCODER/INITIALIZATION SETTINGS

Number	Axis Parameter	Description	Range [Unit]	Access
254	Hall sensor invert	1: Hall sensor invert. Invert the hall scheme, e.g. used by some Maxon motors.	0/1	RWE
250	Encoder steps	Encoder steps per rotation.	0… +4294967295	RWE
209	Actual encoder position	Actual encoder position / counter value	-2147483648… +2147483647	R
251	Encoder direction	Set the encoder direction in a way, that ROR increases position counter.	0/1	RWE
165	Actual encoder commutation offset	This value represents the internal commutation offset. (0 … max. encoder steps per rotation)	-2147483648… +2147483647	RWE
177	Start current	Motor current for controlled commutation. This parameter is used in commutation mode 1, 4, 5 and in initialization of sine.	0… +4294967295 [mA]	RWE

Number	Axis Parameter	Description	Range [Unit]	Access
249	Init sine mode	0: Initialization in controlled sine commutation (determines the encoder offset) 1: Initialization in block commutation using hall sensors 2: Initialization in controlled sine commutation (use the previous set encoder offset)	-128… +127	RWE
241	Init sine speed	Velocity for sine initialization. A positive sign initializes in right direction, a negative sign in left motor direction.	-32768… +32767 [rpm]	RWE
244	Init sine delay	Duration for sine initialization sequence. This parameter should be set in a way, that the motor has stopped mechanical oscillations after the specified time.	-32768… +32767 [ms]	RWE
14	velocity threshold for hallFX™	Velocity to switch from controlled to hallFX™ mode. Set this value to a realistic velocity which the motor can reach in controlled mode!	-2147483648… +2147483647 [rpm]	RWE
159	Commutation mode	0: Block commutation with hall sensors 1: Sensorless block commutation (hall*FX*™) 3: Sine commutation with encoder	0, 1, 3	RWE
160	Re-Initialization of Sine	0: sine commutation is still re-initializing 1: sine commutation is re-initialized Attention: Depending on initialization mode, stop motor before issuing this command!	0/1	RW
247	Sine Compensation Factor	Compensates the propagation delay of the MPU	0… +255	RWE
167	Block PWM scheme	0: PWM chopper on high side, HI on low side 1: PWM chopper on low side, HI on high 2: PWM chopper on low side and high side	-128… +127	RWE
242	Init sine block offset CW	This parameter helps to tune hall sensor based initialization in a way, that the motor has the same velocity for left and right turn. It compensates for tolerance and hysteresis of the hall sensors. It is added to the Commutation offset upon CW turn initialization.	-32768… +32767	RWE
243	Init sine block offset CCW	This parameter helps to tune hall sensor based initialization in a way, that the motor has the same velocity for left and right turn. It compensates for tolerance and hysteresis of the hall sensors. It is added to the Commutation offset upon CCW turn initialization.	-32768… +32767	RWE

PWM mode

Number	Axis Parameter	Description	Range [Unit]	Access
5	PWM limit	Set/get PWM limit (0%… 100%).	0… +3599	RWE
153	Actual PWM duty cycle	Get actual PWM duty cycle.	0… +3599	R
154	Target PWM	Get desired target PWM or set target PWM to activate PWM regulation mode. (+= turn motor in right direction; -= turn motor in left direction)	-3599…+3599	RW

Torque regulation mode

Number	Axis Parameter	Description	Range [Unit]	Access
6	Max current	Set/get the max allowed motor current. This value can be temporarily exceeded marginal due to the operation of the current regulator.	0… +4294967295 [mA]	RWE
150	Actual motor current	Get actual motor current.	-2147483648… +2147483647 [mA]	R
155	Target current	Get desired target current or set target current to activate current regulation mode. (+= turn motor in right direction; -= turn motor in left direction)	-2147483648… +2147483647 [mA]	RW
134	Current regulation loop delay	Delay of current limitation algorithm / PID current regulator.	0… +4294967295 [50µs]	RWE
176	Threshold speed for current PID	Threshold speed for current regulation to switch between first and second current PID parameter set.	-2147483648… +2147483647 [rpm]	RWE
168	P parameter for current PID (I)	P parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
169	I parameter for current PID (I)	I parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
170	D parameter for current PID (I)	D parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
171	I-Clipping parameter for current PID (I)	I-Clipping parameter of current PID regulator. (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
172	P parameter for current PID (II)	P parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
173	I parameter for current PID (II)	I parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
174	D parameter for current PID (II)	D parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
175	I-Clipping parameter for current PID (II)	I-Clipping parameter of current PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
200	Current PID error	Actual error of current PID regulator	-2147483648… +2147483647	R
201	Current PID error sum	Sum of errors of current PID regulator	-2147483648… +2147483647	R

Velocity regulation mode

Number	Axis Parameter	Description	Range [Unit]	Access
3	Actual speed	The actual velocity of the motor.	-2147483648… +2147483647 [rpm]	R
2	Target speed	Set/get the desired target velocity.	-2147483648… +2147483647 [rpm]	RW
9	Motor halted velocity	If the actual speed is below this value the motor halted flag will be set.	-2147483648… +2147483647 [rpm]	RWE
133	PID regulation loop delay	PID calculation delay. Set PID operational frequency.	0… +4294967295 [ms]	RWE
8	Threshold speed for velocity PID	Threshold speed for velocity regulation to switch between first and second velocity PID parameter set.	-2147483648… +2147483647 [rpm]	RWE
140	P parameter for velocity PID (I)	P parameter of velocity PID regulator (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
141	I parameter for velocity PID (I)	I parameter of velocity PID regulator (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
142	D parameter for velocity PID (I)	D parameter of velocity PID regulator (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
143	I-Clipping parameter for velocity PID (I)	I-Clipping parameter of velocity PID (first parameter set, used at lower speed)	-2147483648… +2147483647	RWE
234	P parameter for velocity PID (II)	P parameter of velocity PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
235	I parameter for velocity PID (II)	I parameter of velocity PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
236	D parameter for velocity PID (II)	D parameter of velocity PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
237	I-Clipping parameter for velocity PID (II)	I-Clipping parameter of velocity PID regulator. (second parameter set, used at higher speed)	-2147483648… +2147483647	RWE
228	Velocity PID error	Actual error of PID velocity regulator	-2147483648… +2147483647	R
229	Velocity PID error sum	Sum of errors of PID velocity regulator	-2147483648… +2147483647	R

Velocity ramp parameter

Number	Axis Parameter	Description	Range [Unit]	Access
4	Max. ramp velocity	The maximum velocity used for velocity ramp in velocity mode and positioning mode. Set this value to a realistic velocity which the motor can reach!	-2147483648… +2147483647 [rpm]	RWE
11	Acceleration	Acceleration parameter for ROL, ROR, and the velocity ramp of MVP.	-2147483648… +2147483647 [RPM/s]	RWE
13	Ramp generator speed	The actual speed of the velocity ramp used for positioning and velocity mode.	-2147483648… +2147483647 [rpm]	R
146	Activate ramp	1: Activate velocity ramp generator for position PID control. Allows usage of acceleration and positioning velocity for MVP command.	0/1	RWE

Position regulation mode

Number	Axis Parameter	Description	Range [Unit]	Access
1	Actual position	Set/get the position counter without moving the motor.	-2147483648… +2147483647	RW
0	Target position	The target position of a currently executed ramp.	-2147483648… +2147483647	RW
7	MVP Target reached velocity	Maximum velocity at which end position can be set. Prevents issuing of end position when the target is passed at high velocity.	-2147483648… +2147483647 [rpm]	RWE
10	MVP target reached distance	Maximum distance at which the position end flag is set.	-2147483648… +2147483647	RWE
161	Encoder set NULL	1: set position counter to zero at next N channel event.	0/1	RWE
162	Switch set NULL	1: set position counter to zero at next switch event.	0/1	RWE
163	Encoder clear set NULL	1: set position counter to zero only once 0: always at an N channel event, respectively switch event.	0/1	RWEP
12	Threshold speed for position PID	Threshold speed for position regulation to switch between first and second position PID parameter set.	-2147483648… +2147483647 [rpm]	RWE
130	P parameter for position PID (I)	P parameter of position PID regulator (first parameter set)	-2147483648… +2147483647	RWE
131	I parameter for position PID (I)	I parameter of position PID regulator (first parameter set)	-2147483648… +2147483647	RWE
132	D parameter for position PID (I)	D parameter of position PID regulator (first parameter set)	-2147483648… +2147483647	RWE
135	I-Clipping parameter for position PID (I)	I-Clipping parameter of position PID regulator (first parameter set) (A too high value causes overshooting at positioning mode.)	-2147483648… +2147483647	RWE
230	P parameter for position PID (II)	P parameter of position PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
231	I parameter for position PID (II)	I parameter of position PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
232	D parameter for position PID (II)	D parameter of position PID regulator. (second parameter set)	-2147483648… +2147483647	RWE
233	I-Clipping parameter for position PID (II)	I-Clipping parameter of position PID regulator. (second parameter set) (A too high value causes overshooting at positioning mode.)	-2147483648… +2147483647	RWE
226	Position PID error	Actual error of PID position regulator	-2147483648… +2147483647	R
227	Position PID error sum	Sum of errors of PID position regulator	-2147483648… +2147483647	R

Status information

Number	Axis Parameter	Description	Range [Unit]	Access
151	Actual voltage	Actual supply voltage.	0… +4294967295	R
152	Actual driver temperature	Actual temperature of the motor driver.	0… +4294967295	R
156	Error/Status flags	Bit 0: Overcurrent flag. This flag is set if overcurrent limit is exceeded. Bit 1: Undervoltage flag. This flag is set if supply voltage to low for motor operation. Bit 2: Overvoltage flag. This flag is set if the motor becomes switched off due to overvoltage. Bit 3: Overtemperature flag. This flag is set if overtemperature limit is exceeded. Bit 4: Motor halted flag. This flag is set if motor has been switched off. Bit 5: Hall error flag. This flag is set upon a hall error. Bit 6: unused Bit 7: unused Bit 8: PWM mode active flag. Bit 9: Velocity mode active flag Bit 10: Position mode active flag. Bit 11: Torque mode active flag. Bit 12: unused Bit 13: unused Bit 14: Position end flag. This flag is set if the motor has been stopped at the target position. Bit 15: unused Bit 16: unused for TMC603-EVAL (value should be 0 or 1) Bit 17: I^2t exceeded flag. This flag is set if the I^2t sum exceeded the I^2t limit of the motor. (reset by SAP 29 or after the time specified by the I^2t thermal winding time constant) *Flag 0 to 15 are automatically reset. Only flag 16 and 17 must be cleared manually.*	0…+4294967295	R
157	Module supply current	Get actual supply current of the module.	0… +4294967295 [mA]	R

Switches and analog inputs

Number	Axis Parameter	Description			Range [Unit]	Access
20	Switch 2 active	0: inactive 1: active			0/1	R
21	Switch 1 active	0: inactive 1: active			0/1	R
22	Potentiometer	The value of the analog input.			0…4095	R
164	Activate stop switch	Bit 0	Left stop switch enable	When this bit is set the motor will be stopped if it is moving in negative direction and the left stop switch input becomes active	0… 3	RWE
		Bit 1	Right stop switch enable	When this bit is set the motor will be stopped if it is moving in positive direction and the right stop switch input becomes active		
		Please see parameter 166 for selecting the stop switch input polarity.				
166	Stop switch polarity	Bit 0	Left stop switch polarity	Bit set: Left stop switch input is high active Bit clear: Left stop switch input is low active	0… 3	RWE
		Bit 1	Right stop switch polarity	Bit set: Right stop switch input is high active Bit clear: Right stop switch input is low active		

7 Global parameter overview (SGP, GGP, STGP, RSGP)

The following section describes all global parameters that can be used with the SGP, GGP, STGP and RSGP commands.

Meaning of the letters in column Access:
R = readable (GGP)
W = writable (SGP)
E = automatically restored from EEPROM after reset or power-on

Global parameters are grouped into 3 banks:

- bank 0 (global configuration of the module)
- bank 1 (normally not available; for customer specific extensions of the firmware)
- bank 2 (user TMCL™ variables)

Please use SGP and GGP commands to write and read global parameters. Further you can use the STGP command in order to store user variables permanently in the EEPROM of the module. With the RSGP command the contents of a user variable can be restored from the EEPROM, if this is necessary.

7.1 Bank 0

Parameters 64… 255
Parameters with numbers above 63 configure stuff like the serial address of the module RS232 baud rate or the telegram pause time. Change these parameters to meet your needs. The best and easiest way to do this is to use the appropriate functions of the TMCL-IDE. The parameters with numbers between 64 and 85 are stored in EEPROM only. A SGP command on such a parameter will always store it permanently and no extra STGP command is needed.

Take care when changing these parameters, and use the appropriate functions of the TMCL-IDE to do it in an interactive way.

Meaning of the letters in column *Access*:

- R = readable (GGP)
- W = writeable (SGP)
- E = automatically restored from EEPROM after reset or power-on.

Number	Global parameter	Description	Range	Access
64	EEPROM magic	Setting this parameter to a different value as $E4 will cause re-initialization of the axis and global parameters (to factory defaults) after the next power up. This is useful in case of miss-configuration.	0… 255	RWE
65	RS232 baud rate	0 9600 baud *Default* 1 14400 baud 2 19200 baud 3 28800 baud 4 38400 baud 5 57600 baud 6 76800 baud *Not supported by Windows!* 7 (115200 baud)	0… 7	RWE
66	serial address	The module (target) address for RS232 and virtual COM port	0… 255	RWE
73	configuration EEPROM lock flag	Write: 1234 to lock the EEPROM, 4321 to unlock it. Read: 1=EEPROM locked, 0=EEPROM unlocked.	0/1	RWE
75	telegram pause time	Pause time before the reply via RS232 is sent. For RS232 set to 0.	0… 255	RWE
76	serial host address	Host address used in the reply telegrams sent back via RS232.	0… 255	RWE

Number	Global parameter	Description	Range	Access
77	auto start mode	0: Do not start TMCL™ application after power up (*default*). 1: Start TMCL™ application automatically after power up.	0/1	RWE
81	TMCL™ code protection	Protect a TMCL™ program against disassembling or overwriting. 0 – no protection 1 – protection against disassembling 2 – protection against overwriting 3 – protection against disassembling and overwriting *If you switch off the protection against disassembling, the program will be erased first!* *Changing this value from 1 or 3 to 0 or 2, the TMCL™ program will be wiped off.*	0, 1, 2, 3	RWE
85	do not restore user variables	0 – user variables are restored (*default*) 1 – user variables are not restored	0/1	RWE
128	TMCL™ application status	0 –stop 1 – run 2 – step 3 – reset	0… 3	R
129	download mode	0 – normal mode 1 – download mode	0/1	R
130	TMCL™ program counter	The index of the currently executed TMCL™ instruction.	o… 2047	R
132	tick timer	A 32 bit counter that gets incremented by one every millisecond. It can also be reset to any start value.	0… +4294967295	RW
255	suppress reply	0 – reply (*default*) 1 – no reply	0/1	RW

7.2 Bank 1

The global parameter bank 1 is normally not available. It may be used for customer specific extensions of the firmware. Together with user definable commands (see section 7.3) these variables form the interface between extensions of the firmware (written in C) and TMCL™ applications.

7.3 Bank 2

Bank 2 contains general purpose 32 bit variables for the use in TMCL™ applications. They are located in RAM and can be stored to EEPROM. After booting, their values are automatically restored to the RAM.

Up to 256 user variables are available.

Meaning of the letters in column *Access*:

- R = readable (GGP)
- W = writeable (SGP)
- E = automatically restored from EEPROM after reset or power-on.

Number	Global parameter	Description	Range	Access
0… 55	general purpose variable #0… 55	for use in TMCL™ applications	-2^{31}…$+2^{31}$	RWE
56… 255	general purpose variables #56… #255	for use in TMCL™ applications	-2^{31}…$+2^{31}$	RW

8 PID regulation

8.1 Structure of the cascaded motor regulation modes

The TMC603-EVAL supports current, velocity, and position PID regulation modes for motor control in different application areas. These regulation modes are cascaded as shown in Figure 8.1. The specific modes are explained in the following sections.

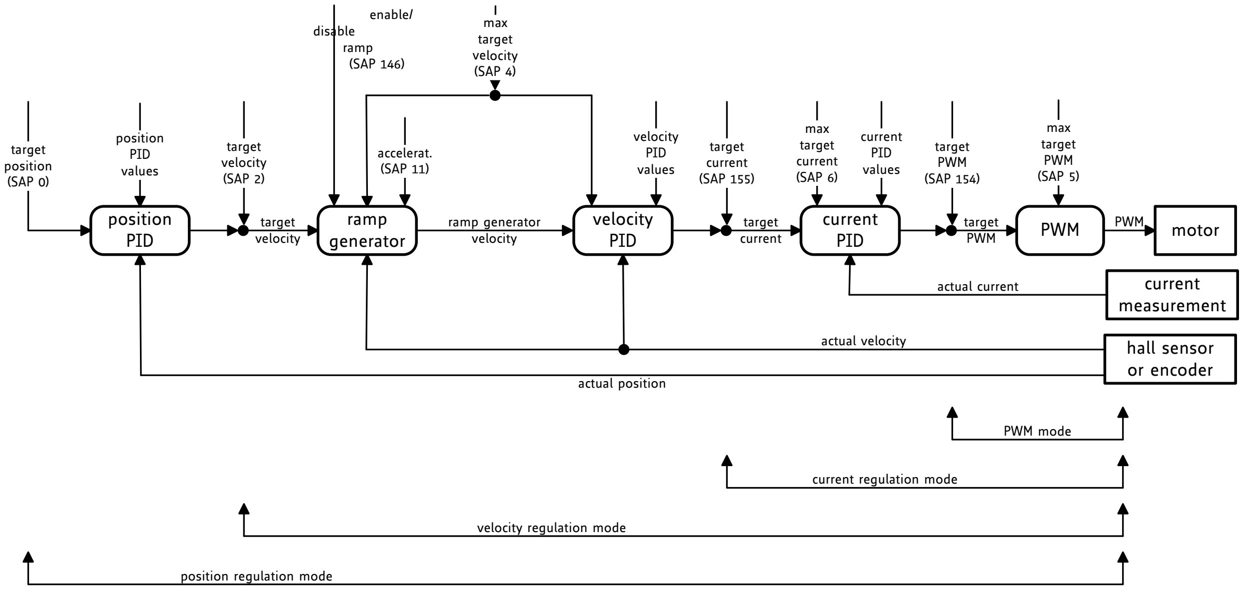


Figure 8.1: Cascaded PID regulation

8.2 PWM regulation

The PWM regulation mode is the most direct control mode for the TMCM-1632/TMCM-KR-841. Thereby, a target PWM given by axis parameter 154 is adjusted directly **without limiting the motor current**. The target PWM is only limited by axis parameter 5 (max target PWM). The sign of the target PWM controls the motor rotation direction.

8.3 Current PID regulation

Based on the PWM regulation the current regulation mode uses a PID regulator to adjust a desired motor current. This target current can be set by axis parameter 155. The maximal current is limited by axis parameter 6.

The PID regulation uses five basic parameters: The *P*, *I*, *D* and *I-Clipping value* as well as the *timing control value*. The timing control value (*current regulation loop delay*, axis parameter 134) determines how often the current regulation is invoked. It is given in multiple of 50µs:

$$t_{PIDDELAY} = x_{PIDRLD} \cdot 50\mu s$$

$t_{PIDDELAY}$ = resulting delay between two PID calculations
x_{PIDRLD} = *current regulation loop delay* parameter

For most applications it is recommended to leave this parameter unchanged at its default of 50µs. Higher values may be necessary for very slow and less dynamic drives. The structure of the current PID regulator is shown in Figure 8.2. It has to be parameterized with respect to a given motor.

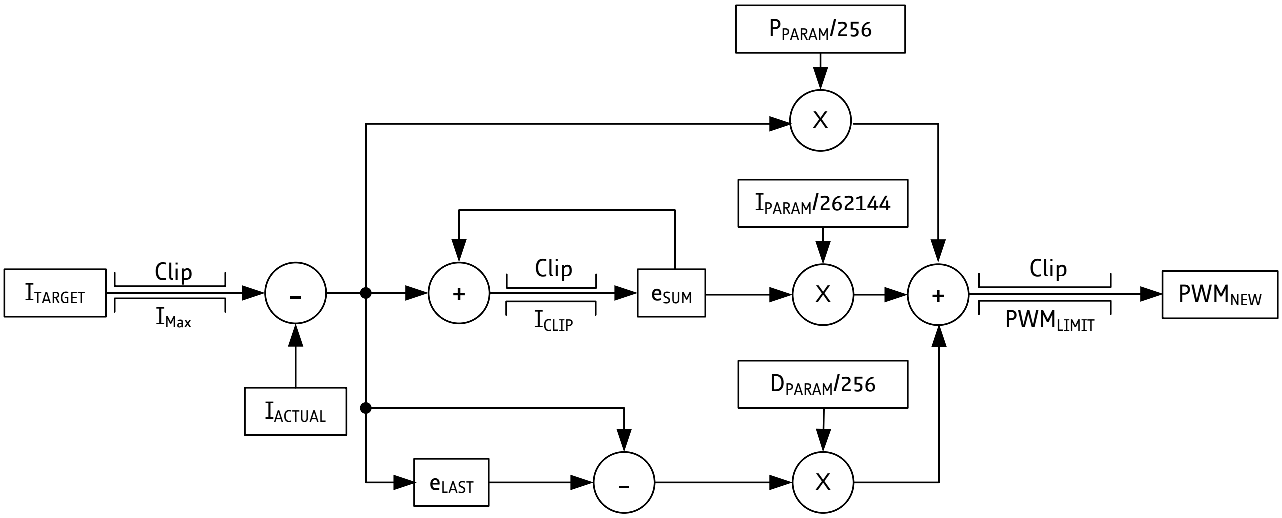


Figure 8.2: Current PID regulation

Parameter	Description
I_{ACTUAL}	Actual motor current (GAP 150)
I_{TARGET}	Target motor current (SAP 155)
I_{Max}	Max. motor current (SAP 6)
e_{LAST}	Error value of the last PID calculation (GAP 200)
e_{SUM}	Error sum for integral calculation (GAP 201)
P_{PARAM}	Current P parameter (SAP 168, SAP 172)
I_{PARAM}	Current I parameter (SAP169, SAP 173)
D_{PARAM}	Current D parameter (SAP 170, SAP 174)
I_{CLIP}	Current I-Clipping parameter (SAP 171, SAP175)
PWM_{LIMIT}	PWM Limit (SAP 5)
PWM_{NEW}	New target PWM value (GAP 153)

To parameterize the current PID regulator for a given motor, first set the P, I and D parameter of both parameter sets to zero. Then start the motor by using a low target current (e.g. 1000mA). Then modify the *current P parameter (II)*. This is the P parameter of parameter set 2. Start from a low value and go to a higher value, until the actual current nearly reaches the desired target current.
After that, do the same for the *current I parameter (II)* with the *current D parameter (II)* still set to zero. For the *current I parameter (II)*, there is also a clipping value. The *current I clipping parameter (II)* should be set to a relatively low value to avoid overshooting, but high enough to reach the target current. The *current D parameter (II)* can still be set to zero.
After having found suitable values for parameter set 2, the first parameter set (PID Parameter Set 1) should be set to lower values, to minimize overshooting during zero-time of motor start. Then stop the motor and start again to test the current regulation settings. If the motor current is overshot during zero-time, set the PID parameter set 1 once more to lower values.

For all tests set the motor current limitation to a realistic value, so that your power supply does not become overloaded during acceleration phases. If your power supply goes to current limitation, the unit may reset or undetermined regulation results may occur.

8.4 Velocity PID regulation

Based on the current regulation the motor velocity can be controlled by the velocity PID regulator. Also, the velocity PID regulator uses a timing control value (*PID regulation loop delay*, axis parameter 133) which determines how often the PID regulator is invoked. It is given in multiple of 1ms:

$$t_{PIDDELAY} = x_{PIDRLD} \cdot 1\text{ms}$$

$t_{PIDDELAY}$ = resulting delay between two PID calculations
x_{PIDRLD} = *PID regulation loop delay* parameter

For most applications it is recommended to leave this parameter unchanged at its default of 1ms. Higher values may be necessary for very slow and less dynamic drives. The structure of the velocity PID regulator is shown in Figure 8.3.

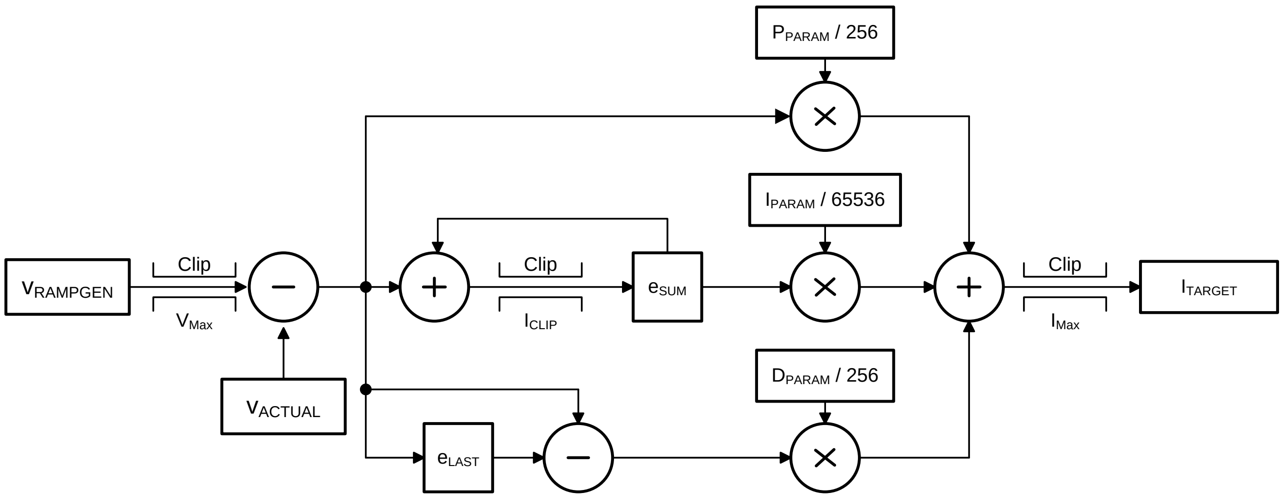


Figure 8.3: Velocity PID regulation

Parameter	Description
v_{ACTUAL}	Actual motor velocity (GAP 3)
$v_{RAMPGEN}$	Target velocity of ramp generator (SAP 2, GAP 13)
v_{Max}	Max. target velocity (SAP 4)
e_{LAST}	Error value of the last PID calculation (GAP 228)
e_{SUM}	Error sum for integral calculation (GAP 229)
P_{PARAM}	Velocity P parameter (SAP 140, SAP 234)
I_{PARAM}	Velocity I parameter (SAP 141, SAP 235)
D_{PARAM}	Velocity D parameter (SAP 142, SAP 236)
I_{CLIP}	Velocity I-Clipping parameter (SAP 143, SAP 237)
I_{Max}	Max. target current (SAP 6)
I_{Target}	Target current for current PID regulator (GAP 155)

To parameterize the PID regulator set the *velocity I parameter* and *velocity D parameter* to zero and start the motor by using a medium target velocity (e.g. 3000 rpm). Then modify the *velocity P parameter*, only. Start from a low value and go to a higher value, until the actual motor speed reaches 80 or 90% of the desired motor speed. The rest of the speed difference can be reduced by using a high I clipping value (e.g. 500000) and a slow increase of the *velocity I parameter* with the *velocity D parameter* still set to zero. For the first tests, both PID parameter sets can be set equal.

8.5 Velocity ramp generator

For a controlled start up of the motor's velocity a velocity ramp generator can be activated/deactivated by axis parameter 146. The ramp generator uses the maximal allowed motor velocity (axis parameter 4), the acceleration (axis parameter 11) und the desired target velocity (axis parameter 2) to calculate a ramp generator velocity for the following velocity PID regulator.

8.6 Position PID regulation

Based on the current and velocity PID regulators the TMC603-EVAL supports a positioning mode based on encoder or hall sensor position. During positioning the velocity ramp generator can be activated to enable motor positioning with controlled acceleration or disabled to support motor positioning with max allowed speed. The structure of the position PID regulator is shown in Figure 8.4.

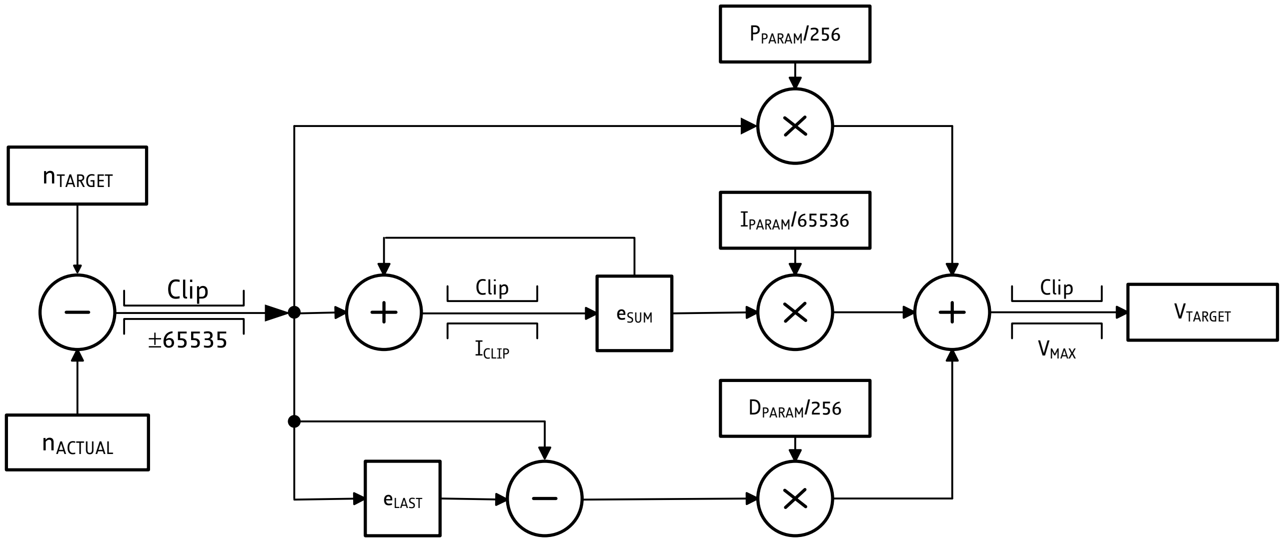


Figure 8.4: Positioning PID regulation

Parameter	Description
n_{ACTUAL}	Actual motor position (GAP 1)
n_{TARGET}	Target motor position (SAP 0)
e_{LAST}	Error value of the last PID calculation (GAP 226)
e_{SUM}	Error sum for integral calculation (GAP 227)
P_{PARAM}	Position P parameter (SAP 130, SAP 230)
I_{PARAM}	Position I parameter (SAP 131, SAP 231)
D_{PARAM}	Position D parameter (SAP 132, SAP 232)
I_{CLIP}	Position I-Clipping parameter (SAP 135, SAP 233)
V_{MAX}	Max. allowed velocity (SAP 4)
V_{TARGET}	New target velocity for ramp generator (GAP 13)

The PID regulation uses five basic parameters. The P, I, D, and I-Clipping value as well as a timing control value. The timing control value (*PID regulation loop delay* - axis parameter 133) determines how often the PID regulator is invoked. It is given in multiple of 1ms:

$$t_{PIDDELAY} = x_{PIDRLD} \cdot 1\text{ms}$$

$t_{PIDDELAY}$ = the resulting delay between two PID calculations

x_{PIDRLD} = *PID regulation loop delay* parameter

For most applications it is recommended to leave this parameter unchanged at its default of 1ms. Higher values may be necessary for very slow and less dynamic drives.

Based on the velocity PID regulator the position PID regulator can be parameterized as P regulator in the simplest case. Therefore, disable the velocity ramp generator and set *position P, I, and D parameters* to zero. Now, set a target position and increase the *position P parameter* until the motor reaches the target position approximately.

After finding a good *position P parameter* the velocity ramp generator can be switched on again. Based on the *max. positioning velocity* (axis parameter 4) as well as the *acceleration* (axis parameter 11) value the ramp generator automatically calculates the slow down point, i.e. the point at which velocity is to be reduced in order to stop at the desired target position. Reaching the target position is signaled by setting the *position end flag*.

In order to minimize the time until this flag becomes set, a positioning tolerance (*MVP target reached distance*) can be chosen by axis parameter 10. Since the motor typically is assumed not to signal target reached when the target was just passed in a short moment at a high velocity, additionally a maximum target reached velocity (*MVP target reached velocity*) can be defined by axis parameter 7. A value of zero is the most universal, since it implies that the motor stands still at the target. But when a fast rising of the *Position end flag* is desired, a higher value for *MVP target reached velocity* parameter will save a lot of time. The best value should be tried out in the actual application. The correlation of axis parameter 10, 7, the target position and the position end flag are summarized in Figure 8.5.

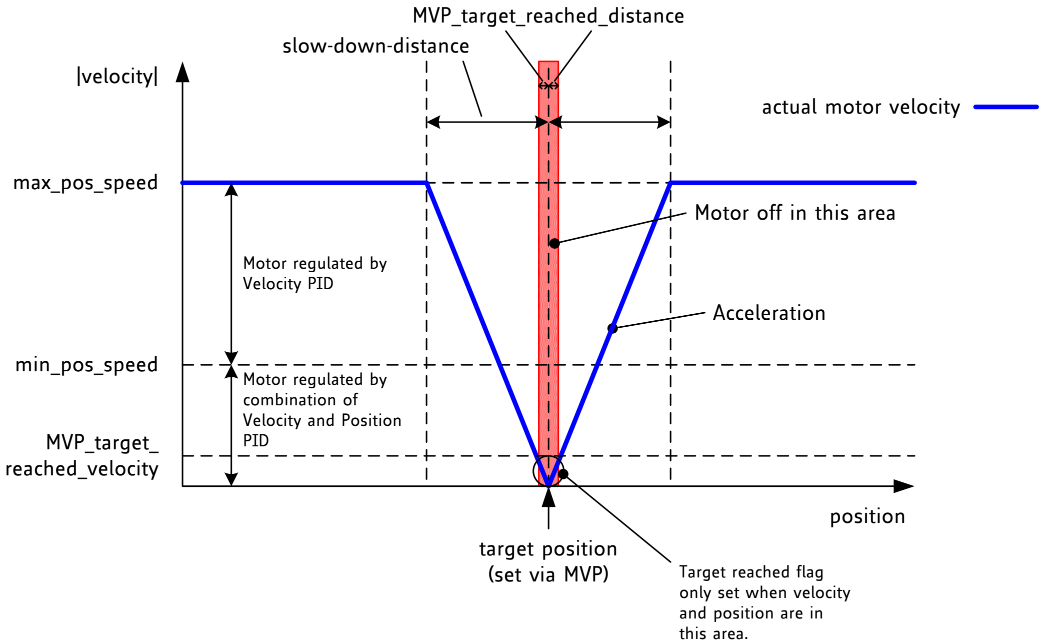


Figure 8.5: Positioning algorithm

Depending on the motor and mechanics respectively, a bit of oscillation is normal, in the best case it can be reduced to be at least +/-1 encoder step, because otherwise the regulation cannot keep the position.

8.7 Parameter sets for PID regulation

Every PID regulation provides two parameter sets, which are used as follows:

- Below a specified velocity threshold the PID regulator uses a combination of parameter set 1 and parameter set 2
- Above the velocity threshold the PID regulator uses only parameter set 2. If the velocity threshold is set to zero, parameter set 2 is used all the time. (The switch over between both parameter sets is soft.)

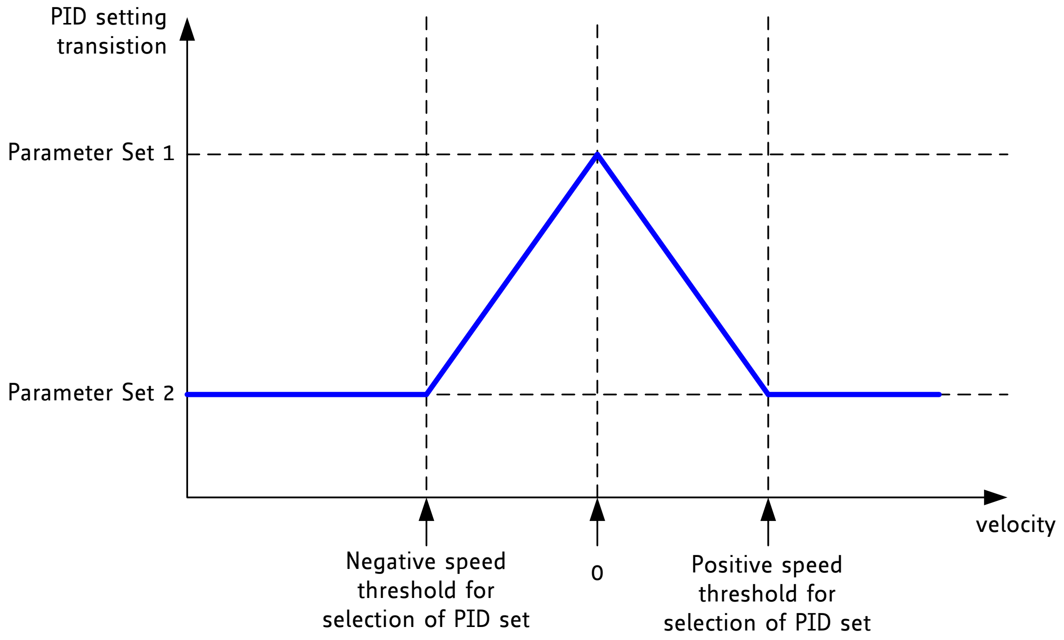


Figure 8.5: Transition between PID parameter sets

The velocity thresholds for current, velocity, and position PID regulation can be set as follows:

- axis parameter 176: velocity threshold for current PID regulator
- axis parameter 8: velocity threshold for velocity PID regulator
- axis parameter 12: velocity threshold for position PID regulator

Attention: For all tests set the motor current limitation to a realistic value, so that your power supply does not become overloaded during acceleration phases. If your power supply goes to current limitation, the unit may reset or undetermined regulation results may occur.

9 Temperature calculation

Axis parameter 152 delivers the actual ADC value of the motor driver. This ADC value can be converted to a temperature in °C as follows:

$$\text{ADC} = actual\ value\ of\ GAP\ 152$$

$$\text{B} = 3434\ (material\ constant)$$

$$\text{R}_{\text{NTC}} = \frac{9011{,}2}{ADC} - 2.2$$

$$\text{T} = \frac{B * 298{,}16}{B + (\ln\left(\frac{R_{NTC}}{10}\right) * 298.16} - 273.16\ ℃$$

Example 1:
ADC = 1000
R_{NTC} ≈ 6.81
T ≈ 35℃

Example 2:
ADC = 1200
R_{NTC} ≈ 5.31
T ≈ 42℃

10 I²t monitoring

The I²t monitor determines the sum of the square of the motor current over a given time. The integrating time is motor specific. In the datasheet of the motor this time is described as *thermal winding time constant* and can be set for each module using axis parameter 25. The number of measurement values within this time depends on how often the current regulation and thus the I²t monitoring is invoked. The value of the actual I²t sum can be read by axis parameter 27. With axis parameter 26 the default value for the I²t limit can be changed (default: 211200). If the actual I²t sum exceeds the I²t limit of the motor, flag 17 (in axis parameter 156) is set and the motor PWM is set to zero as long as the I²t exceed flag is set. The actual regulation mode will not be changed. Furthermore, the I²t exceed counter is increased once every second as long as the actual I²t sum exceeds the I²t limit. The I²t exceed flag can be cleared manually using parameter 29 but only after the cool down time given by the *thermal winding time constant* has passed. The I²t exceed flag will not be reset automatically. The I²t limit can be determined as follows:

$$I^2t = \frac{I\ [mA]}{1000} * \frac{I\ [mA]}{1000} * t_{tw}\ [ms]$$

I is the desired average current
t_{tw} is the thermal winding time constant given by the motor datasheet

Example:

I²t limits for an average current of a) 1A, b) 2A, c) 3A and d) 4A over a thermal winding time of 13,2s.

a) $I^2t\ limit = \frac{1000\ [mA]}{1000} * \frac{1000\ [mA]}{1000} * 13200\ [ms] = 13200\ [mA^2 * ms]$

b) $I^2t\ limit = \frac{2000\ [mA]}{1000} * \frac{2000\ [mA]}{1000} * 13200\ [ms] = 52800\ [mA^2 * ms]$

c) $I^2t\ limit = \frac{3000\ [mA]}{1000} * \frac{3000\ [mA]}{1000} * 13200\ [ms] = 118800\ [mA^2 * ms]$

d) $I^2t\ limit = \frac{4000\ [mA]}{1000} * \frac{4000\ [mA]}{1000} * 13200\ [ms] = 211200\ [mA^2 * ms]$

11 Life support policy

TRINAMIC Motion Control GmbH & Co. KG does not authorize or warrant any of its products for use in life support systems, without the specific written consent of TRINAMIC Motion Control GmbH & Co. KG.

Life support systems are equipment intended to support or sustain life, and whose failure to perform, when properly used in accordance with instructions provided, can be reasonably expected to result in personal injury or death.

12 Revision history

12.1 Firmware revision

Version	Date	Author	Description
1.46	2011-AUG-25	ED	Complete renewed version
1.47	2011-NOV-24	ED	PID regulation updated I^2t monitoring updated

Table 12.1: Firmware revision

12.2 Document revision

Version	Date MW – Michael Weiss SD – Sonja Dwersteg	Author	Description
0.93	2009-AUG-11	MW	Preliminary version
1.00	2011-SEP-30	SD	Initial version
1.01	2011-OCT-06	SD	Minor changes
1.02	2011-OCT-12	ED	Minor changes
1.03	2011-OCT-17	SD	Minor changes
1.04	2011-OCT-25	SD	Minor changes
1.05	2011-DEC-13	SD	- Axis parameters 5, 150, 153, 159, and 154 corrected - Axis parameter 246 deleted - Chapter 10 updated (I^2t monitoring) - Chapter 8 updated (PID regulation)
1.06	2012-AUG-29	SD	Axis parameters 164 and 166 added

Table 12.2: Document revision

13 References

[TMC603] TMC603 Datasheet (please refer to our homepage http://www.trinamic.com)