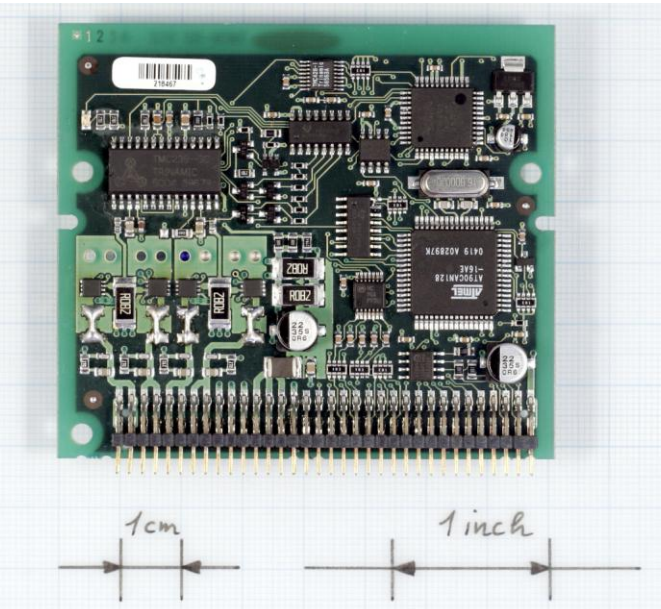
TMCM-101

One Axis Closed Loop Controller / Driver Module
3A / 28.5V



Manual

Version: 1.08
June 24th, 2009

Trinamic Motion Control GmbH & Co KG
Sternstraße 67
D - 20 357 Hamburg, Germany
http://www.trinamic.com

Contents

1 Features

The TMCM-101 is a single axis 2-phase stepper motor controller and driver module. The build in encoder feedback makes it an optimum solution for high reliability drives. It is equipped with a CAN interface and a serial interface that can be used as an RS232, RS422 or RS485 interface by adding an appropriate line driver/level shifter. The module also has two stop switch inputs (stop left and stop right), eight general purpose digital outputs and seven general purpose inputs where each one can be used either as a digital input or as an analogue input.
An interface for incremental encoders (with A, B and N channel) allows exact position control and closed loop operation. Furthermore, using the Trinamic StallGuard feature overload or obstruction of the motor can be detected.
The TMCM-101 can be programmed using the Trinamic Motion Control Language (TMCL) which allows to control the module by a host or to run stand alone, executing a TMCL program that is stored in the 16kByte EEPROM on the module (the EEPROM can store up to 2047 TMCL commands).
All inputs and outputs of the module are provided on an 68 pin connector with 2mm pitch. The size of the board is very compact (80 x 70mm).

Applications

- Controller board for control of one two-phase bipolar motor
- Versatile possibilities of applications in stand alone or pc controlled mode

Electrical Data

- Up to 3A coil current RMS (4.2A peak)
- 7V to 28.5 V motor supply voltage
- 5V DC logic power supply

Supported motors

- two-phase bipolar motors with 1A to 3A coil current
- incremental encoder (2 or 3 channel)

Interface

- RS-232, RS-485 or CAN 2.0b host interface
- Classical analog interface for drivers
- Inputs for reference and stop switches, general purpose analog and digital I/Os

Highlights

- 64 times microstepping
- Memory for 2048 TMCL commands
- Automatic shaped ramp generation in Hardware
- On the fly alteration of motion parameters (e.g. position, velocity, acceleration)
- Closed loop for highly dependable drives
- StallGuard™ option for sensorless motor stall detection

Software

- Stand-alone operation using TMCL or remote controlled operation
- PC-based application development software TMCL-IDE included
- Optional simple to use ASCII protocol

Other

- 68 pin connector carries all signals (2*34 pins, 2mm pitch)
- RoHS compliant latest from 1 July 2006
- Size: 80x70mm²

Order code	Description
TMCM-101 (-option)	1-axis controller /driver 3A / 28.5V
Option	
-H	horizontal pin connector (standard)
-V	vertical pin connector (on request)

Table 1.1: Order codes

2 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.

3 Technical Data

- Size: 80 x 70mm
- Supply voltage: +5V DC (core), +7..28V DC (motor)
- Host interface: Serial (RS232, RS422 or RS458), CAN
- Connector type: 2x34 pins, 2mm pitch
- Stepper motor type: two phase bi-polar
- Maximum coil current (RMS): 3A (can be set by software)
- Microcontroller: AT90CAN128
- Motion controller: TMC428
- Motor driver: TMC249, with SI7501 MOSFETs
- Microsteps: max. 64
- TMCL program memory: 16kByte EEPROM (2047 TMCL commands)

4 Electrical and Mechanical Interfacing

4.1 Dimensions

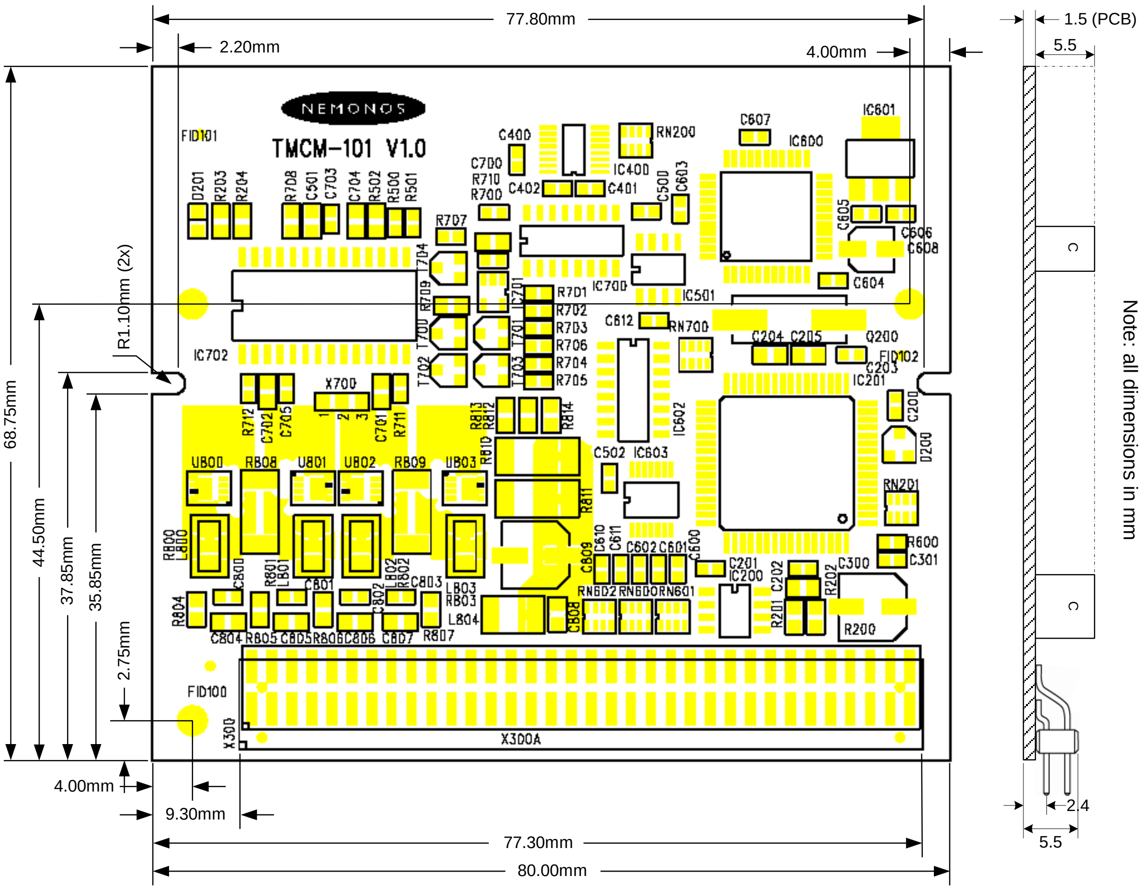


Figure 4.1: Dimensions

The size of the module 80x70mm, similar as the other Trinamic motion control modules (80x50). It also uses the same connector.

Connector type: 2x34 pins, 2mm pitch

4.2 Pin Assignments

The module is equipped with a 68 pin connector (two rows of 34 pins each) with 2mm pitch. (There is a special Trinamic document about mating connectors). Figure 4.2 shows how to locate pin 1. The following table shows all pin assignments:

Pin	Signal name	See section	Pin	Signal name	See section
1	Motor A1	6.3	2	Motor A1	6.3
3	Motor A1	•	4	Motor A1	•
5	Motor A2	•	6	Motor A2	•
7	Motor A2	•	8	Motor A2	•
9	Motor B1	•	10	Motor B1	•
11	Motor B1	•	12	Motor B1	•
13	Motor B2	•	14	Motor B2	•
15	Motor B2	•	16	Motor B2	•
17	GND	6.2	18	GND	6.2
19	GND	•	20	GND	•
21	GND	•	22	GND	•
23	7..28.5V DC	•	24	7..28.5V DC	•
25	7..28.5V DC	•	26	7..28.5V DC	•
27	7..28.5V DC	•	28	7..28.5V DC	•
29	GND	•	30	GND	•
31	Address bit 0	6.9	32	not connected	---
33	Address bit 1	•	34	Reset (low active,internally pulled up)	6.9
35	Address bit 2	6.9	36	PDI (leave open, for Trinamic only)	---
37	Right stop switch (internally pulled high)	6.7	38	PDO (leave open, for Trinamic only)	---
39	Left stop switch (internally pulled high)	6.7	40	SCK (leave open, for Trinamic only)	---
41	Encoder Channel A	6.4	42	Alarm output	6.9
43	Encoder Channel B	•	44	Shutdown input	6.9
45	Encoder Channel N	6.4	46	General purpose output 0	6.8
47	General purpose input 7	6.8	48	General purpose output 1	•
49	General purpose input 6	•	50	General purpose output 2	•
51	General purpose input 5	•	52	General purpose output 3	•
53	leave open (Temperature sensor on input 4)	6.9	54	General purpose output 4	•
55	General purpose input 3	6.8	56	General purpose output 5	•
57	General purpose input 2	•	58	General purpose output 6	•
59	General purpose input 1	•	60	General purpose output 7	•
61	General purpose input 0	•	62	RS485 direction (H=send, L=receive)	6.5.3
63	CAN high	6.5.1	64	Serial RxD (RS232 / RS422 / RS485)	6.5.2
65	CAN low	6.5.1	66	Serial TxD (RS232 / RS422 / RS485)	6.5.2
67	+5V DC (± 10%)	6.2	68	+5V DC (± 10%)	6.2

Table 4.1: Pinout 68-pin connector

Please note that these pin assignments are **not** compatible with the other TMCM modules, so this module can not be used with the TMCM Evaluation board.
Signals with higher current are assigned to more than one pin. For those signals all pins must be used. Otherwise the connector may get damaged by too high current on a pin. So, always use **all** GND, +24V, +5V and motor pins.

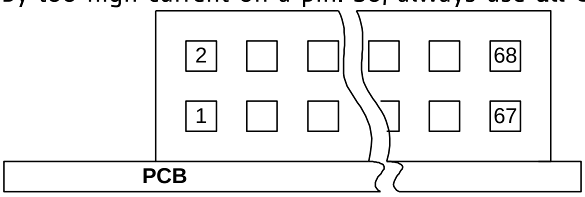


Figure 4.2: Locating the pins

5 Operational Ratings

The operational ratings show the intended / the characteristic range for the values and should be used as design values. In no case shall the maximum values be exceeded.

Symbol	Parameter	Min	Typ	Max	Unit
V_{+5V}	+5V DC input (max. 200mA required)	4.75	5.0	5.25	V
V_S	Power supply +7..28V DC	7	24	28.5	V
V_{INPROT}	Input voltage for StopL, StopR, GPIO (internal protection diodes)	-0.5	0 … 5	V_{+5V}+0.5	V
V_{ANA}	INx analog measurement range		0 … 5		V
V_{INLO}	INx, StopL, StopR low level input		0	0.9	V
V_{INHI}	INx, StopL, StopR high level input (integrated 10k pullup to +5V for Stop)	2	5		V
I_{OUTI}	OUTx max +/- output current (CMOS output) (sum for all outputs max. 50mA)			+/-20	mA
T_{ENV}	Environment temperature at rated current (no cooling)	-40		+45	°C
	Environment temperature at 80% of rated current or 50% duty cycle (no forced cooling required)	-40		+70	°C

Table 5.1: Operational Ratings

6 Functional Description

In Figure 6.1 the main parts oft the TMCM-101 module are shown. The module mainly consists of a TMC428 motion controller, the TMCL program memory (EEPROM), the host interfaces (RS-232, RS-485 and CAN), the TMC249 high power driver and an encoder interface.

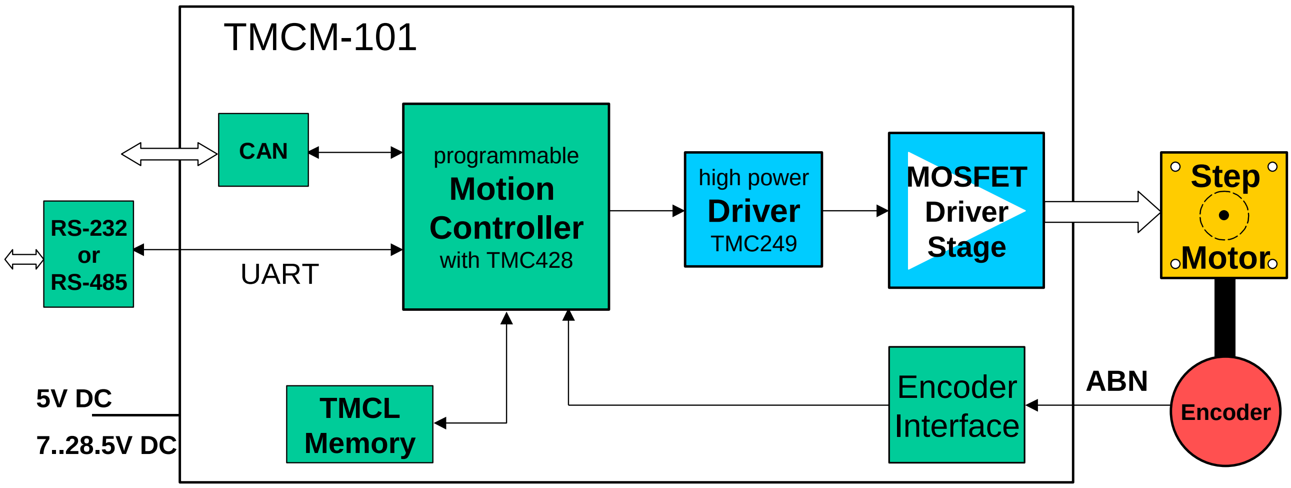


Figure 6.1: Main parts of the TMCM-101

6.1 System Architecture

The TMCM-101 integrates a microcontroller with the TMCL (Trinamic Motion Control Language) operating system. The motion control real-time tasks are realized by the TMC428.

6.1.1 Microcontroller

On this module, the Atmel AT90CAN128 is used to run the TMCL operating system and to control the TMC428. The CPU has 128Kbyte flash memory, 4 Kbyte RAM and 4Kbyte EEPROM. The microcontroller runs the TMCL (Trinamic Motion Control Language) operating system which makes it possible to execute TMCL commands that are sent to the module from the host via the interface. The microcontroller interprets the TMCL commands and controls the TMC428 which executes the motion commands.
The flash ROM of the microcontroller holds the TMCL operating system and the EEPROM memory of the microcontroller is used to permanently store configuration data.
The TMCL operating system can be updated via the serial interface (RS232 or RS422). Please use the latest version of the TMCL IDE to do this.

6.1.2 TMCL EEPROM

To store TMCL programs for stand alone operation the TMCM-101 module is equipped with a 16kByte EEPROM attached to the microcontroller. The EEPROM can store TMCL programs consisting of up to 2048 TMCL commands.

6.1.3 TMC428 Motion Controller

The TMC428 is a high-performance stepper motor control IC and can control up to three 2-phase-stepper-motors. Motion parameters like speed or acceleration are sent to the TMC428 via SPI by the microcontroller. Calculation of ramps and speed profiles are done internally by hardware based on the target motion parameters.

6.1.4 Stepper Motor Drivers

The stepper motor driver used on the TMCM-101 module is the TMC249 chip together with SI7501 MOSFETs and an extension for up to 64 microsteps. This driver circuit is very easy to use. It can control the currents for the two phases of the stepper motors. 64x microstepping and a maximum coil current of 3A RMS is possible.
The power dissipation of the TMC249 chip and the MOSFETs is very low so that the temperature of the MOSFETs normally does not exceed 125°C. Depending on the ambient temperature a cooling fan may be needed. The coils will be switched off automatically when the temperature or the current exceeds the limits and are automatically switched on again when the values are within the limits again.

6.1.5 Incremental encoder interface

The encoder interface of the TMCM-101 module allows to connect two channel incremental encoders with an optional null channel. The signal of the encoder must have TTL (+5V) level.

6.2 Power supply requirements

Two different power supplies have to be provided for the TMCM-101: +5VDC for the controller part and +7..28.5VDC for the motor supply. Please connect all listed pins for the power supply inputs and ground in parallel. Refer to 4.2 Pin Assignments.

Pin	Function
67, 68	+5V DC (+/- 5%), I_{max}=50mA power supply
29, 30	Ground
23 to 28	+7..28.5V DC motor power supply
17 to 22	Ground

Table 6.1: Pinning of Power supply

It is recommended to use capacitors of some 1000µF and a choke close to the module for the motor supply. This ensures a stable power supply and minimizes noise injected into the power supply cables. The choke especially becomes necessary with larger distributed systems using a common power supply.

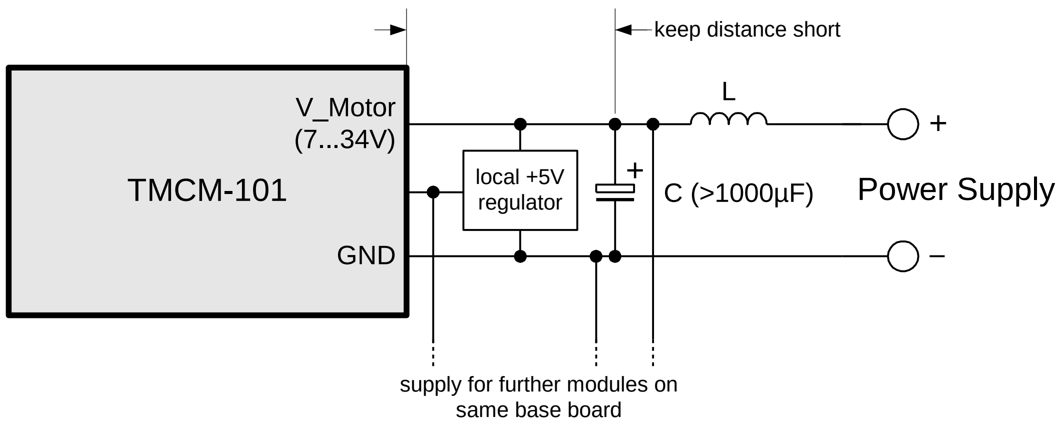


Figure 6.2: Power supply requirements for TMCM-101

Especially with bus controlled systems (e.g. CAN or RS485) it is important to ensure a stable ground potential of all modules. The stepper driver modules draw peak currents of some Ampere from the power supply. It has to be made sure, that this current does not cause a substantial voltage difference on the interface lines between the module and the master, as disturbed transmissions could result.

The following hints help avoiding transmission problems in larger systems. Not all hints have to be followed:

- Use power supply filter capacitors of some 1000µF on the base board for each module in order to take over current spikes. A choke in the positive power supply line will prevent current spikes from changing the GND potential of the base board, especially when a central power supply is used.
- Optionally use an isolated power supply for the TMCM-Modules (no earth connection on the power supply, in case the CAN master is not optically decoupled)
- Do not supply modules with the same power supply which are mounted in a distance of more than a few meters.
- For modules working on the same power supply (especially the same power supply as the master) use a straight and thick, low-resistive GND connection.
- Use a local +5V regulator on each base-board.

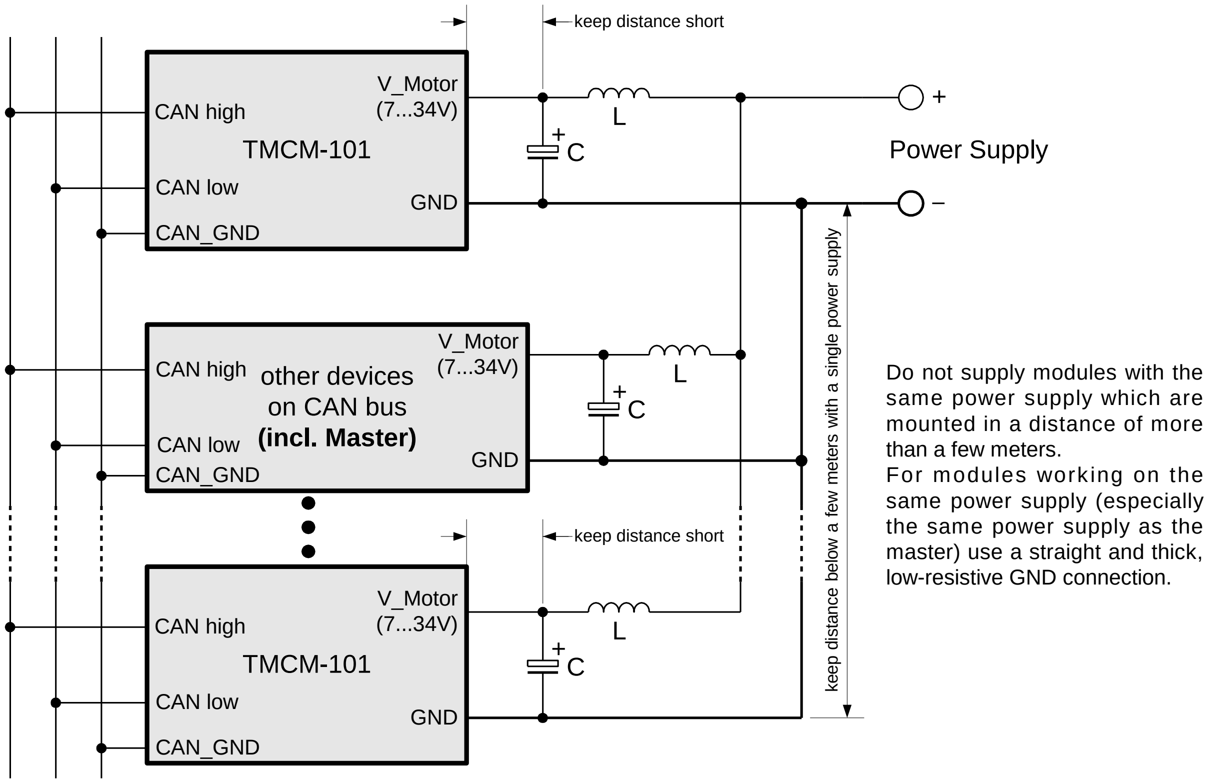


Figure 6.3: Power supply requirements for TMC-Modules in a bus system

For large systems, an optically decoupled CAN bus for each number of nodes, e.g. for each base board with a number of TMCM-101 modules may make sense, especially when a centralized power supply is to be used. Be aware that different ground potentials of the CAN sender (e.g. a PC) and the power supply may damage the modules. Please make sure that the GND lines of the CAN sender and the module(s) and power supplies are connected by a cable.

6.3 Motor Connection

<u>Warning</u>: Never connect or disconnect the motors while the TMCM-101 Module is switched on. Doing this will destroy the driver ICs!

The TMCM-101 controls a single 2-phase stepper motors. The connections between the motor and the 68-pin connector must be done as shown in Table 6.2.

Pin Number	Direction	Name	Motor Numbers and Coils
1 to 4	Out	Motor A1	Motor #0, Coil A1
5 to 8	Out	Motor A2	Motor #0, Coil A2
9 to 12	Out	Motor B1	Motor #0, Coil B1
13 to 16	Out	Motor B2	Motor #0, Coil B2

Table 6.2: Pinout for Motor Connections

6.4 Encoder interface

An incremental encoder with TTL (5V) outputs can be connected to the pins 41 (channel A), 43 (channel B) and 45 (*optional* null channel). The encoder counter can be read by software and can be used to control the exact positioning of the motor. This also makes closed loop operation possible.

The Encoder channel N is for nulling of the encoder counter. It can be selected as high or as low active, and it is automatically checked in parallel to the Encoder channel A and B inputs, to give an exact reference. It has to be determined by the user, which potential of the encoder A and B inputs correspond to the zero position. This depends on the actual encoder type used.

To read out or to change the position value of the encoder, axis parameter #209 is used. So, to read out the position of encoder 0 (the only one) use GAP 209, 0. The position values can also be changed using command SAP 209, 0, <n> , with n = ± **0**,1,2,…

To change the encoder settings of an encoder, axis parameter #210 is used:

Please note that the pre-scaler values can only be set and not read back or stored to EEPROM (it has to be initialized after every power down). To change the pre-scaler of encoder 0 use SAP 210, 0, <p>

To select a pre-scaler, the following values can be used for <p>:

Value for <p>	Resulting pre-scaler	SAP command for motor 0 SAP 210, M0, <p>	Resulting steps per rotation for a 400 line (1600 quadrate count) encoder
64	0.125	SAP 210, M0, 64	200
128	0.25	SAP 210, M0, 128	400
256	0.5	SAP 210, M0, 256	800
512	**1**	**SAP 210, M0, 512**	**1600**
1024	2	SAP 210, M0, 1024	3200
2048	4	SAP 210, M0, 2048	6400
4096	8	SAP 210, M0, 4096	12800
8192	16	SAP 210, M0, 8192	25600
16384	32	SAP 210, M0, 16384	51200
32768	64	SAP 210, M0, 32768	102400

Table 6.3: TMCL – Pre-scaler values

Formula for resulting steps per rotation:
StepsPerRotation = LinesOfEncoder * 4 * PreScaler

There are some special functions that can also be configured using these values. To select these functions just add the following values to <p>:

Adder for <p>	SAP command for motor 0 SAP 210, M0, <p>
1	Encoder Channel A polarity for Null channel event (0=negative, 1=positive)
2	Encoder Channel B polarity for Null channel event (0=negative, 1=positive)
4	Clear encoder with next null channel event
16	Encoder Channel N polarity for encoder clearing (0=negative, 1=positive)

Table 6.4: Special encoder functions

Add up both values <p> from this tables to get the required value for the SAP210 command. The resulting pre-scaler is Value/512. Other pre-scaler values are possible, also.

- Automatic motor stop on deviation error is also usable. This can be set using axis parameter 212 (maximum deviation). This function is turned off when the maximum deviation is set to 0.

6.5 Host Communication

Communication to a host takes place via one or more of the onboard interfaces. The module provides a wide range of different interfaces, like CAN, RS-232, RS-422 and RS-485. The following chapters explain how the interfaces are connected with the 68-pin connector.

The TMCL protocol is used for the communication between the module and a host. Please see the TMCL Reference and Programming manual for further information. The TMCL-IDE can be used to put the module into operation and to develop TMCL programs that can run on the TMCL-101 in stand alone mode.

To use the serial interface a level shifter / line driver must be connected to the pins RxD (pin 64) and TxD (pin 66). For RS485 drivers a direction signal is provided on pin 62.
When connecting a single TMCM-101 module to an RS232 level shifter or to an RS485 line driver an additional pull-up resistor should be connected to the TxD pin (66) of the module.

6.5.1 CAN 2.0b

The module is equipped with a CAN line driver (PCA 82C250) so that no additional components are needed to use the CAN interface. The CAN interface is compatible with the CAN 2.0b standard.

Pin Number	Direction	Name	Limits	Description
63	InOut	CAN -	-8…+18V	CAN Input / Output
65	InOut	CAN +	-8…+18V	CAN Input / Output

Table 6.5: Pinout for CAN Connection

6.5.2 RS-232 and RS-422

Pin Number	Direction	Name	Limits	Description
64	In	RxD	TTL	RS-232 Receive Data
66	Out	TxD	TTL	RS-232 Transmit Data

Table 6.6: Pinout for RS-232 Connection

Note: The RS-232 must be operated with TTL-Levels (0V, 5V), since there is no level shifter onboard!

When the RS232 interface is in receive mode the TxD pin will be tri-stated. This feature can be used to connect multiple TMCM-101 modules to only one RS422 line driver, just by connecting all TxD pins and all RxD pins of the modules to the same input and output pins of the RS422 diver.

6.5.3 RS-485

Pin Number	Direction	Name	Limits	Description
62	Out	RS485_DIR	TTL	Driver / Receiver enable for RS-485 Transceiver. 0: Receiver enable 1: Driver enable (send)
64	In	RxD	TTL	RS-485 Receive Data
66	Out	TxD	TTL	RS-485 Transmit Data

Table 6.7: Pinout for RS-485 Connection

Note: The TMCM-101 Module does not contain any RS-485 transceivers!

6.6 StallGuard™ - Sensorless Motor Stall Detection

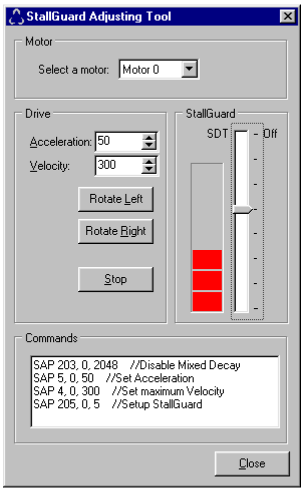
The TMCM-101 modules are equipped with StallGuard. The StallGuard option makes it possible to detect if the mechanical load on a stepper motor is too high or if the traveler has been obstructed. The load value can be read using a TMCL command or the module can be programmed so that the motor will be stopped automatically when it has been obstructed or the load has been to high.
StallGuard can also be used for finding the reference position without the need for a reference switch: Just activate StallGuard and then let the traveler run against a mechanical obstacle that is placed at the end of the way. When the motor has stopped it is definitely at the end of its way, and this point can be used as the reference position.
To use StallGuard in an actual application, some manual tests should be done first, because the StallGuard level depends upon the motor velocities and on the occurrence of resonances.
Mixed decay should be switched off when StallGuard is turned on in order to get usable results.

Value	Description
0	StallGuard function is deactivated (default)
1..7	Motor stops when StallGuard value is reached

Table 6.8: StallGuard parameter SAP 205

To activate the StallGuard feature use the TMCL-command SAP 205 and set the StallGuard threshold value according to Table 6.8. The actual load value is given by GAP 206. The TMCL IDE has some tools which let you try out and adjust the StallGuard function in an easy way. They can be found at "StallGuard" in the "Setup"-menu and are described in the following chapters.

6.6.1 StallGuard adjusting tool

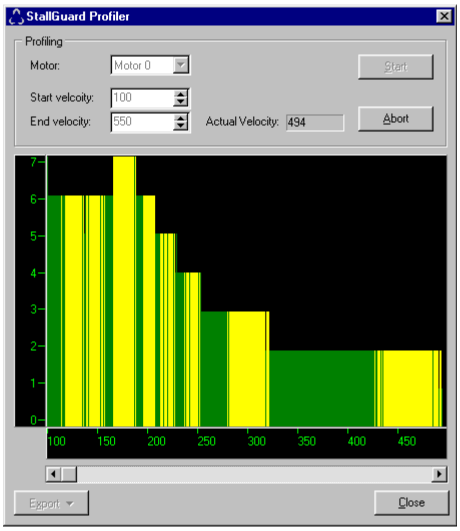


The StallGuard adjusting tool helps to find the necessary motor parameters when StallGuard is to be used. This function can only be used when a module is connected that features StallGuard. This is checked when the StallGuard adjusting tool is selected in the "Setup" menu. After this has been successfully checked the StallGuard adjusting tool is displayed.
First, select the axis that is to be used in the "Motor" area.
Now you can enter a velocity and an acceleration value in the "Drive" area and then click "Rotate Left" or "Rotate Right". Clicking one of these button will send the necessary commands to the module so that the motor starts running. The red bar in the "StallGuard" area on the right side of the windows displays the actual load value. Use the slider to set the StallGuard threshold value. If the load value reaches this value the motor stops. Clicking the "Stop" button also stops the motor.
All commands necessary to set the values entered in this dialogue are displayed in the "Commands" area at the bottom of the window. There, they can be selected, copied and pasted into the TMCL editor.

Figure 6.4: StallGuard adjusting tool

6.6.2 StallGuard profiler

The StallGuard profiler is a utility that helps you find the best parameters for using stall detection. It scans through given velocities and shows which velocities are the best ones. Similar to the StallGuard adjusting tool it can only be used together with a module that supports StallGuard. This is checked right after the StallGuard profiler has been selected in the "Setup" menu. After this has been successfully checked the StallGuard profiler window will be shown.



First, select the axis that is to be used. Then, enter the "Start velocity" and the "End velocity". The start velocity is used at the beginning of the profile recording. The recording ends when the end velocity has been reached. Start velocity and end velocity must not be equal. After you have entered these parameters, click the "Start" button to start the StallGuard profile recording. Depending on the range between start and end velocity this can take several minutes, as the load value for every velocity value is measured ten times. The "Actual velocity" value shows the velocity that is currently being tested and so tells you the progress of the profile recording. You can also abort a profile recording by clicking the "Abort" button.
The result can also be exported to Excel or to a text file by using the "Export" button.

Figure 6.5: The StallGuard Profiler

6.6.2.1 The result of the StallGuard profiler

The result is shown as a graphic in the StallGuard profiler window. After the profile recording has finished you can scroll through the profile graphic using the scroll bar below it. The scale on the vertical axis shows the load value: a higher value means a higher load. The scale on the horizontal axis is the velocity scale. The colour of each line shows the standard deviation of the ten load values that have been measured for the velocity at that point. This is an indicator for the vibration of the motor at the given velocity. There are three colours used:

- Green: The standard deviation is very low or zero. This means that there is effectively no vibration at this velocity.
- Yellow: This colour means that there might be some low vibration at this velocity.
- Red: The red colour means that there is high vibration at that velocity.

6.6.2.2 Interpreting the result

In order to make effective use of the StallGuard feature you should choose a velocity where the load value is as low as possible and where the colour is green. The very best velocity values are those where the load value is zero (areas that do not show any green, yellow or red line). Velocities shown in yellow can also be used, but with care as they might cause problems (maybe the motor stops even if it is not stalled).
Velocities shown in red should not be chosen. Because of vibration the load value is often unpredictable and so not usable to produce good results when using stall detection.
As it is very seldom that exactly the same result is produced when recording a profile with the same parameters a second time, always two or more profiles should be recorded and compared against each other.

6.7 Reference switches

With reference switches, an interval for the movement of the motor or the zero point can be defined. Also a step loss of the system can be detected, e.g. due to overloading or manual interaction, by using a travel-switch.
The polarity of the stop switch inputs is software selectable (by default, the stop switch inputs must be low to let the motor run and high to stop the motor). They an also be completely de-activated by software.

Pin Number	Direction	Name	Limits	Description
37	In	STOPR	TTL	Right reference switch input for Motor #0
39	In	STOPL	TTL	Left reference switch input for Motor #0

Table 6.9: Pinout reference switches

Note: 10k pullup resistors for reference switches are included on the module.

6.7.1 Left and right limit switches

The TMCM-101 can be configured so that a motor has a left and a right limit switch (Figure 6.6). *The motor then stops when the traveler has reached one of the limit switches.*

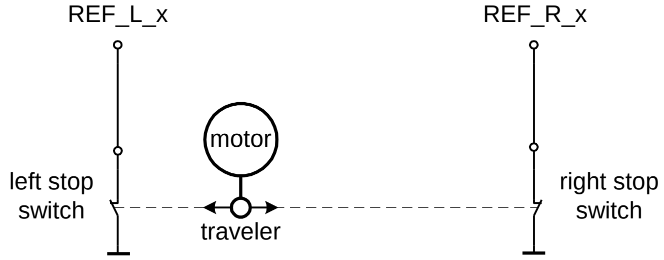


Figure 6.6: Left and right limit switches

6.7.2 Triple Switch Configuration

It is possible to program a tolerance range around the reference switch position. This is useful for a triple switch configuration, as outlined in Figure 6.7. In that configuration two switches are used as automatic stop switches, and one additional switch is used as the reference switch between the left stop switch and the right stop switch. The left stop switch and the reference switch are wired together. The center switch (travel switch) allows for a monitoring of the axis in order to detect a step loss.

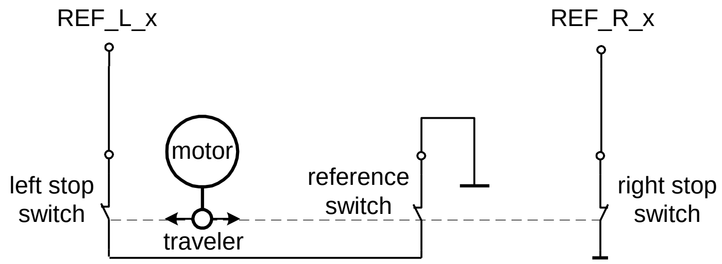


Figure 6.7: Limit switch and reference switch

6.7.3 One Limit Switch for circular systems

If a circular system is used (Figure 6.8), only one reference switch is necessary, because there are no end-points in such a system.

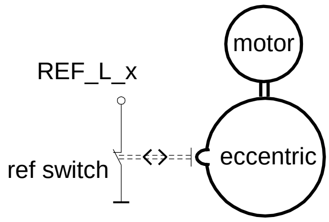


Figure 6.8: One reference switch

6.8 Port Expansion

For further expansion and adaptation to user requirements the module provides a port expansion for the microcontroller. The expansion includes eight TTL input pins and eight TTL output pins, which are accessible via the 68-pin connector. The outputs can drive up to 20mA. Each of these inputs can either be used as a digital input (TTL) or as an analogue input (0..+5V, 10 bit accuracy). Please note that all these I/Os are not protected!

Pin Number	Direction	Name	Limits	Description
47	In	INP_7	TTL	Port expansion Pin 7, input
49	In	INP_6	TTL	Port expansion Pin 6, input
51	In	INP_5	TTL	Port expansion Pin 5, input
53	In	INP_4	TTL	leave open (Temperature sensor on input 4)
55	In	INP_3	TTL	Port expansion Pin 3, input
57	In	INP_2	TTL	Port expansion Pin 2, input
59	In	INP_1	TTL	Port expansion Pin 1, input
61	In	INP_0	TTL	Port expansion Pin 0, input
46	Out	Out_0	TTL	Port expansion Pin 0, output
48	Out	Out_1	TTL	Port expansion Pin 1, output
50	Out	Out_2	TTL	Port expansion Pin 2, output
52	Out	Out_3	TTL	Port expansion Pin 3, output
54	Out	Out_4	TTL	Port expansion Pin 4, output
56	Out	Out_5	TTL	Port expansion Pin 5, output
58	Out	Out_6	TTL	Port expansion Pin 6, output
60	Out	Out_7	TTL	Port expansion Pin 7, output

Table 6.10: Pinout port expansion

6.9 Miscellaneous Connections

Pin Number	Direction	Name	Limits	Description
31, 33, 35	Out	Modul_SelX	TTL	Module address setting
34	In	Reset	TTL	Reset (low active, internally pulled high)
42	Out	Alarm	TTL	Alarm, active high
44	In	Shutdown	TTL	Shutdown, active high
53	In	INP_4	TTL	Temperature sensor on input 4

Table 6.11: Miscellaneous Connections

The shutdown input (pin 44) can be used for emergency stop functions (e.g. to connect an emergency stop switch). Its functionality can be selected by software. The alarm output (pin 42) can be used to signal errors like driver overheat. Its functionality is determined by software.
The address pins (pins 31, 33, 35) have no function when pulled to ground. Otherwise they define the lower three bit of the module address. Thus multiple TMCM-101 can be used easily in a backplane All these pins are not protected.
The module address is set by software via the TMCL-IDE in the "Setup/Configure Module" menu. Please refer to [TMCL]. The standard address is "1".

6.10 Resetting the module

All settings of the module can be reset to factory default values by hardware. To do this, turn off the power, short the pins 38 and 40 of the connector, and then turn on the power. The LED on the module then flashes very fast. Then turn off the power, remove the link between the pins 38 and 40 and turn on the power again. Wait until the LED flashes normally. All settings are then at their factory default values.

6.11 Firmware update

By using the function "Install OS" from the "Setup" menu you can update the firmware of a module. First, load the new firmware file (which must be in Intel-HEX format) by clicking the "Load" button. The file is then checked if it is a TMCL firmware file, and its device type and version number will be displayed. Then, click the "Start" button to program the new firmware into the module. Please make sure that there will be no power cut or cut of the serial connection during the programming process. The program checks if the device type in the firmware file and the device type of the module are identical. An error message will be displayed if this should not be the case. If everything is okay, the new firmware will be programmed into the module and verified afterwards. The programming progress is shown by the status bar.

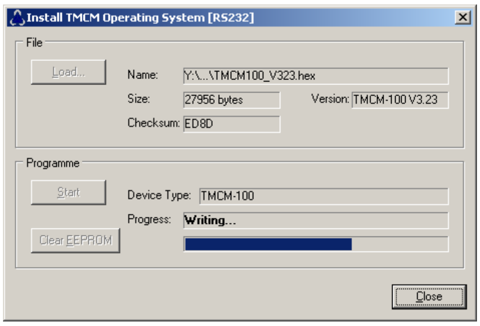


Fig. 6.1: Firmware update in progress

If you click the "Clear EEPROM" button, all settings stored in the EEPROM of the module will be erased. If you then power the module off and on again, all settings will be restored to their factory defaults.

7 Putting the TMCM-101 into operation

On the basis of a small example it is shown step by step how the TMCM-101 is set into operation. Users who are already familiar with TMCL and other Trinamic modules may skip this chapter. It is assumed that all necessary hardware connections are made properly.

Example: The following application is to be implemented on the TMCM-101 module using the TMCL-IDE Software development environment.

A formula how "speed" is converted into a physical unit like rotations per seconds can be found in chapter 8.

The simple application is:

- Move the Motor to position 150000
- Wait 2 seconds
- Move the Motor back to position 0
- Wait 1 second
- Start again with the first step

To implement this simple application on theTMCM-101 it is necessary to do the following things:

Step 1: Connect the host interface to the PC

Step 2: Connect the motor to the motor connector

Step 3: Connect the power supply voltage to the module

Step 4: Switch on the power supply. The activity LED should start to flash. This indicates the correct configuration of the microcontroller.

Step 5: Start the TMCL-IDE Software development environment. Enter the program shown in the following listing. A description of the TMCL commands can be found in the TMCL Reference and Programming Manual.

```
//A simple example for using TMCL and the TMCL-IDE

        SAP 4, 0, 100              //Set the maximum speed

Loop:   MVP ABS, 0, 150000         //Move to position 150000
        WAIT POS, 0, 0
        WAIT TICKS, 0, 200
        MVP ABS, 0, 0              //Move back to position 0
        WAIT POS, 0, 0
        WAIT TICKS, 0, 100

        JA Loop                    //Infinite Loop
```

Step 6: Click the "*Assemble*" icon to convert the TMCL program into byte code.
Then download the program to the TMCM-101 module by clicking the "Download" icon.

Step 7: Click the "Run" icon. The downloaded program will now be executed.

A detailed documentation about the TMCL operations and the TMCL IDE can be found in the TMCL Reference and Programming Manual. The next chapter shows how the velocity and acceleration values are calculated.

8 Calculating Velocity and Acceleration

The values of the parameters sent to the TMCM-101 do not have typical motor values, like rotations per second as velocity. But these values can be calculated from the TMC-101 parameters, as shown in this document. The parameters for the TMC-101 are:

Signal	Description	Range
f_{CLK}	clock-frequency	0..16 MHz
velocity	-	0..2047
a_max	maximum acceleration	0..2047
pulse_div	Velocity pre-divider. The higher the value is, the less is the maximum velocity. Default value = 3 Can be changed in TMCL using SAP 154.	0..13
ramp_div	Acceleration pre-divider. The higher the value is, the less is the maximum acceleration default value = 7 Can be change in TMCL using SAP 153.	0..13
Usrs	Microstep resolution (microsteps per fullstep = 2^{usrs}). Can be changed in TMCL using SAP 140.	0..6

Table 8.1: TMC428 velocity parameters

The **microstep-frequency** of the stepper motor is calculated with

$$usf[Hz] = \frac{f_{CLK}[Hz] \cdot velocity}{2^{pulse\_div} \cdot 2048 \cdot 32}$$ with usf: microstep-frequency

To calculate the **fullstep-frequency** from the microstep-frequency, the microstep-frequency must be divided by the number of microsteps per fullstep.

$$fsf[Hz] = \frac{usf[Hz]}{2^{usrs}}$$ with fsf: fullstep-frequency

The change in the pulse rate per time unit (pulse frequency change per second – the **acceleration a**) is given by

$$a = \frac{{f_{CLK}}^2 \cdot a_{max}}{2^{pulse\_div+ramp\_div+29}}$$

This results in an acceleration in fullsteps of:

$$af = \frac{a}{2^{usrs}}$$ with af: acceleration in fullsteps

Example:
f_CLK = 16 MHz on the TMCM-110 module
velocity = 1000
a_max = 1000
pulse_div = 1
ramp_div = 1
usrs = 6

$$msf = \frac{16\,MHz \cdot 1000}{2^1 \cdot 2048 \cdot 32} = \underline{\underline{122070.31 Hz}}$$

$$fsf[Hz] = \frac{122070.31}{2^6} = \underline{\underline{1907.34 Hz}}$$

$$a = \frac{(16 Mhz)^2 \cdot 1000}{2^{1+1+29}} = \underline{\underline{119.21 \frac{MHz}{s}}}$$

$$af = \frac{119.21 \frac{MHz}{s}}{2^6} = \underline{\underline{1.863 \frac{MHz}{s}}}$$

If the stepper motor has e.g. 72 fullsteps per rotation, the number of rotations of the motor is:

$$RPS = \frac{fsf}{\text{fullsteps per rotation}} = \frac{1907.34}{72} = 26.49$$

$$RPM = \frac{fsf \cdot 60}{\text{fullsteps per rotation}} = \frac{1907.34 \cdot 60}{72} = 1589.46$$

9 Software

TMCL, the Trinamic Motion Control Language is used to send commands from the host to the TMCM-101 module and to write programs that can be stored in the EEPROM of the module so that the module can execute the TMCL commands in stand alone mode.

TMCL is described in a separate documentation, the TMCL Reference and Programming Manual. This document also describes the TMCL Integrated Development Environment (TMCL IDE), a program running on Windows which allows easy development of TMCL applications.
All the manuals are provided on the TMC TechLib CD and on the web site of TRINAMIC Motion Control GmbH & Co. KG (http://www.trinamic.com). Also the latest versions of the firmware (TMCL operating system) and PC software (TMCL IDE) can be found there.

10 Revision History

10.1 Documentation Revision

Version	Comment	Author	Description
1.00	17-Nov-05	OK	Initial version
1.01	12-Jun-06	OK	Adapted to hardware rev. 1.03
1.02	25-Oct-06	HC	Major Revision, encoder info added
1.03	08-Nov-06	HC	Reset and Firmware update added
1.04	16-Jan-07	HC	Addition to 6.9 (address pins) and to 4.2 (references)
1.05	07-Nov-07	HC	Chapter 6.4: Encoder interface updated/now supported; Chapter 6.9: support of address setting by hardware
1.06	05-Dec-07	HC	Chapter 6.2 Power supply requirements updated
1.07	14-Jul-08	OK	Chapter 6.10 corrected
1.08	24-Jun-09	OK	Chapter 6.6 corrected

Table 10.1: Documentation Revisions

10.2 Firmware Revision

Version	Comment	Description
6.20	Initial Release	Please refer to TMCL documentation (encoder pre-scaler not implemented)
6.28	Several updates	Updates include encoder pre-scaler.

Table 10.2: Firmware Revisions

11 References

[TMCL] TMCL manual (see http://www.trinamic.com)