Three Phase Stepper with TMC5062

Valid for TMC5062

The TMC5062 supports driving up to two three phase stepper motors. This application note describes the additional register settings available in the TMC5062 for three phase motor operation.
Additionally it covers the option bits to change the microstep resolution for TMC5062 and TMC5031 (see Table 1.2). This allows increasing the effective acceleration rate, e.g. half microstep gives double velocity and acceleration rate.

Table of Contents

1 Three Phase Motor Operation

The TMC5062 is prepared to drive three phase motors. However, three phase motors are not often found in stepper drives, but they provide an additional smoothness at low velocity operation due to a lower intrinsic detent torque. The following chapters show additional information adding to the TMC5062 datasheet in order to provide the required settings. For three phase motors, the TMC5062 also supports stallGuard as well as coolStep, but dcStep is only available with two phase steppers. The TMC5062 three phase section is identical to the TMC389.

1.1 Three Phase Motor Application

In a three phase motor application, only three half bridges and a single sense resistor are required to drive a stepper motor. One or both motor drivers can be configured for driving a three phase motor. *It is important to match the external wiring before switching on this option of the power driver stage. Do not enable a two phase setting with a three phase motor.*

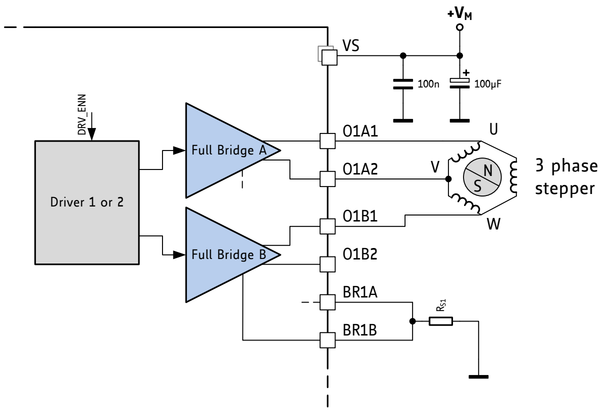

Figure 1.1 Connecting a three phase motor

Pin	Number	Type	Function
O2A1	14	O (VS)	Motor 2 A1 output (stepper motor coil A), or three phase stepper motor U output.
BR2A	15		Motor 2 bridge A negative power supply and current sense input. Provide external sense resistor to GND.
O2A2	16	O (VS)	Motor 2 A2 output (stepper motor coil A), or three phase stepper motor V output.
VS	17, 19		Driver 2 positive power supply. Connect to VS and provide sufficient filtering capacity for chopper current ripple.
GNDP	18	GND	Power GND for driver 2. Connect to GND.
O2B1	20	O (VS)	Motor 2 B1 output (stepper motor coil B), or three phase stepper motor W output.
BR2B	21		Motor 2 bridge B negative power supply and current sense input. Provide external sense resistor to GND. Connect to BR2A for three phase motor operation.
O2B2	22	O (VS)	Motor 2 B2 output (stepper motor coil B), unused for three phase stepper motor.
O1B2	39	O (VS)	Motor 1 B2 output (stepper motor coil B), unused for three phase stepper motor.
BR1B	40		Motor 1 bridge B negative power supply and current sense input. Provide external sense resistor to GND. Connect to BR1A for three phase motor operation.
O1B1	41	O (VS)	Motor 1 B1 output (stepper motor coil B), or three phase stepper motor W output.
VS	42, 44		Driver 1 positive power supply. Connect to VS and provide sufficient filtering capacity for chopper current ripple.
GNDP	43	GND	Power GND for driver 1. Connect to GND.
O1A2	45	O (VS)	Motor 1 A2 output (stepper motor coil A), or three phase stepper motor V output.
BR1A	46		Motor 1 bridge A negative power supply and current sense input. Provide external sense resistor to GND.
O1A1	47	O (VS)	Motor 1 A1 output (stepper motor coil A), or three phase stepper motor U output.

Table 1 Driver stage pins showing three phase motor option

1.2 Register Set to support Three Phase Motors

Programmable microstep shapes allow optimizing the motor performance. Both drivers can drive a two- or a three phase stepper motor. In order to drive higher current motors, a parallel mode allows double current for a single driver, by paralleling both output power stages. In order to operate a three phase motor, the MSLUTSTART entry has to be re-initialized to fit the 120° phase shift required for a three phase motor, rather than 180° phase shift for a two phase motor. The settings most important for a three phase motor are highlighted.

MOTOR DRIVER REGISTER SET (MOTOR 1: 0X60…0X6F, MOTOR 2: 0X70…0X7F)					
R/W	**Addr**	**n**	***Register***	**Description / *bit names***	**Range [Unit]**
W	0x69 0x79	8 + 8	*MSLUTSTART*	bit 7… 0: *START_SIN* bit 23… 16: *START_SIN90_120* *START_SIN* gives the absolute current at microstep table entry 0. *START_SIN90_120* gives the absolute current for microstep table entry at positions 256 resp. 171 for a three phase motor. Start values are transferred to the microstep registers *CUR_A* and *CUR_B*, whenever the reference position *MSCNT*=0 is passed. *Attention: Default values are for two phase motor, initialize before first motion with three phase motor!*	*START_SIN* *reset default* *=0* *(2ph/3ph)* *START_SIN90_120* *reset default* *=247* *(2ph)* *Set to 214 for 3 phase motor when using default table.*
R	0x6A 0x7A	10	*MSCNT*	Microstep counter. Indicates actual position in the microstep table for *CUR_A*. *CUR_B* uses an offset of 256 (2 phase motor) resp. 171 (3 phase motor). *Hint:* Move to a position where *MSCNT* is zero before re-initializing *MSLUTSTART* or *MSLUT* and *MSLUTSEL*.	

CHOPCONF – Chopper Configuration

0x6C, 0x7C: *CHOPCONF* – Chopper Configuration 31…16				
Bit	**Name**	**Function**	**Comment**	
31	*ph3sel*	*1: 3 phase motor*	*ph3sel*=0: 2 phase motor *ph3sel*=1: 3 phase motor (see Table 1.3)	
30	*diss2g*	short to GND protection disable	0: Short to GND protection is on 1: Short to GND protection is disabled	
29	*0*	Do not use	Set 0	*Leave 0 for normal operation*
28	*intpol16*	16 microsteps with interpolation	1: In 16 microstep mode, the microstep resolution becomes extrapolated to 256 microsteps for smoothest motor operation	
27	*mres3*	*MRES* micro step resolution	%0000: Native 256 microstep setting. Normally use this setting when the IC is operated with the internal ramp generator.	
26	*mres2*			
25	*mres1*			
24	*mres0*		%0001 … %1000: 128, 64, 32, 16, 8, 4, 2, FULLSTEP Reduced microstep resolution. The resolution gives the number of microstep entries per sine quarter wave. Please take care to switch at certain microstep positions, when switching to a low resolution. The switching position determines the sequence of patterns. step width=2^*MRES* [microsteps] *Attention: Optional use to increase effective acceleration in high acceleration applications, only. A 128 microstep setting will double the acceleration range, as the motor operates with double velocity at the same setting as with 256 microsteps. Use carefully. Does not integrate with dcStep.*	
23	*sync3*	*SYNC* PWM synchronization clock	*ph3sel*=0, only	
22	*sync2*			
21	*sync1*			
20	*sync0*			
19	*vhighchm*	high velocity chopper mode	*ph3sel*=0, only	
18	*vhighfs*	high velocity fullstep selection	*ph3sel*=0, only	
17	*vsense*	sense resistor voltage based current scaling	0: Low sensitivity, high sense resistor voltage 1: High sensitivity, low sense resistor voltage	
16	*tbl1*	*TBL* blank time select	%00 … %11: Set comparator blank time to 16, 24, 36 or 54 clocks *Hint*: %10 is recommended for most applications	

Table 1.2 Chopper settings bits 31… 16. Refer separate table for bits 15… 0 for 3 phase motor

CHOPPER CONFIGURATION 15… 0 FOR 3 PHASE MOTOR

0X6C, 0X7C: *CHOPCONF* – CHOPPER CONFIGURATION 15…0 WITH *PH3SEL*=1 (*3 PHASE MOTOR*)			
Bit	**Name**	**Function**	**Comment**
15	*tbl0*		(see *tbl1* in bit 16)
14	-	reserved	set to 0
13	*rndtf*	random *TOFF* time	0: Chopper off time is fixed as set by *TOFF* 1: Random mode, *TOFF* is random modulated by dN_{CLK}= -12 … +3 clocks.
12	*csync*	chopper synchronization	0: Chopper freely running 1: Synchronize chopper to full steps
11	-	reserved	set to 0
10	*nosd3ph*		0: Each chopper on-cycle is followed by a slow decay phase as set by *TOFF* 1: Slow decay phases are skipped between the chopper phases, except directly following a short to GND or chopper synchronization. Set blank time *TBL* ≥ 2.
9	*hyst5*	*HYST* hysteresis value for three phase chopper	DAC hysteresis setting: %00000 … %11111: 0…63 (1/512 of this setting adds to current setting) *Attention:* *Effective HYST/2 must be ≤ 255-sinewave peak (248 at max. current setting) – Reduce current setting to 28 for maximum hysteresis.* *Do not work with too small setting (poor performance).* *I_RUN*=31 requires *HYST* < 16 *I_RUN*=30 requires *HYST* < 32 *I_RUN*=29 requires *HYST* < 48 *Hint:* Hysteresis decrement is done each 16 clocks
8	*hyst4*		
7	*hyst3*		
6	*hyst2*		
5	*hyst1*		
4	*hyst0*		
3	*toff3*	*TOFF* off time and driver enable	Off time setting controls duration of slow decay phase N_{CLK}= 12 + 32**TOFF* (Minimum is 64 clocks) %0000: Driver disable, all bridges off %0001: 1 – use only with *TBL* ≥ 2 %0010 … %1111: 2 … 15
2	*toff2*		
1	*toff1*		
0	*toff0*		

Table 1.3: Chopper settings for three phase motor (bits 15… 0)

COOLCONF – Smart Energy Control coolStep and stallGuard2

0x6D, 0x7D: *COOLCONF* – Smart Energy control coolStep and stallGuard2			
Bit	**Name**	**Function**	**Comment**
…	-	reserved	set to 0
25	*sspd*	stallGuard2 speed (3 phase motor only)	0: Standard mode, high time resolution for stallGuard2 1: stallGuard2 uses more filtering, use for low motor velocity only. Set this bit, if stallGuard value ripple is too high.
24	*sfilt*	stallGuard2 filter enable	0: Standard mode, high time resolution for stallGuard2 1: Filtered mode, stallGuard2 signal updated for each four fullsteps (resp. six fullsteps for 3 phase motor) only to compensate for motor pole tolerances
23	-	reserved	set to 0
22	*sgt6*	stallGuard2 threshold value	This signed value controls stallGuard2 level for stall output and sets the optimum measurement range for readout. A lower value gives a higher sensitivity. Zero is the starting value working with most motors. -64 to +63: A higher value makes stallGuard2 less sensitive and requires more torque to indicate a stall.
21	*sgt5*		
20	*sgt4*		
19	*sgt3*		
18	*sgt2*		
17	*sgt1*		
16	*sgt0*		
15	*seimin*	minimum current for smart current control	0: 1/2 of current setting (*IRUN*) 1: 1/4 of current setting (*IRUN*)
14	*sedn1*	current down step speed	%00: For each 32 stallGuard2 values decrease by one %01: For each 8 stallGuard2 values decrease by one %10: For each 2 stallGuard2 values decrease by one %11: For each stallGuard2 value decrease by one
13	*sedn0*		
12	-	reserved	set to 0
11	*semax3*	stallGuard2 hysteresis value for smart current control	If the stallGuard2 result is equal to or above (*SEMIN*+*SEMAX*+1)*32, the motor current becomes decreased to save energy. %0000 … %1111: 0 … 15
10	*semax2*		
9	*semax1*		
8	*semax0*		
7	-	reserved	set to 0
6	*seup1*	current up step width	Current increment steps per measured stallGuard2 value %00 … %11: 1, 2, 4, 8
5	*seup0*		
4	-	reserved	set to 0
3	*semin3*	minimum stallGuard2 value for smart current control and smart current enable	If the stallGuard2 result falls below *SEMIN**32, the motor current becomes increased to reduce motor load angle. %0000: smart current control coolStep off %0001 … %1111: 1 … 15
2	*semin2*		
1	*semin1*		
0	*semin0*		

DRV_STATUS – stallGuard2 Value and Driver Error Flags

0x6F, 0x7F: DRV_STATUS – stallGuard2 value and driver error flags			
Bit	**Name**	**Function**	**Comment**
31	*stst*	standstill indicator	This flag indicates motor stand still in each operation mode. It is especially useful for step & dir mode.
30	*olb*	open load indicator phase B	1: Open load detected on phase A or B or on any phase for a three phase motor. *Hint:* This is just an informative flag. The driver takes no action upon it. False detection may occur in fast motion and standstill. Check during slow motion, only.
29	*ola*	open load indicator phase A	
28	*s2gb*	short to ground indicator phase B	1: Short to GND detected on phase A or B or on any phase for a three phase motor. The driver becomes disabled. The flags stay active, until the driver is disabled by software or by the ENN input.
27	*s2ga*	short to ground indicator phase A	
26	*otpw*	overtemperature pre-warning flag	1: Overtemperature pre-warning threshold is exceeded. The overtemperature pre-warning flag is common for both drivers.
25	*ot*	overtemperature flag	1: Overtemperature limit has been reached. Drivers become disabled until *otpw* is also cleared due to cooling down of the IC. The overtemperature flag is common for both drivers.
24	*stallGuard*	stallGuard2 status	1: Motor stall detected (*SG_RESULT*=0)
23 22 21	-	reserved	Ignore these bits
20 19 18 17 16	*CS ACTUAL*	actual motor current / smart energy current	Actual current control scaling, for monitoring smart energy current scaling controlled via settings in register *COOLCONF*.
15	*fsactive*	full step active indicator	1: Indicates that the driver has switched to fullstep as defined by chopper mode settings and velocity thresholds.
14 13 12 11 10	-	reserved	Ignore these bits
9 8 7 6 5 4 3 2 1 0	*SG_ RESULT*	stallGuard2 result respectively PWM on time for coil A in stand still for motor temperature detection	Mechanical load measurement: The stallGuard2 result gives a means to measure mechanical motor load. A higher value means lower mechanical load. A value of 0 signals highest load. With optimum *SGT* setting, this is an indicator for a motor stall. The stall detection compares *SG_RESULT* to 0 in order to detect a stall. *SG_RESULT* is used as a base for coolStep operation, by comparing it to a programmable upper and a lower limit. *SG_RESULT* is not applicable when dcStep is active – stallGuard2 works best with microstep operation. Temperature measurement: In standstill, no stallGuard2 result can be obtained. *SG_RESULT* shows the chopper on-time for motor coil A instead. If the motor is moved to a determined microstep position at a certain current setting, a comparison of the chopper on-time can help to get a rough estimation of motor temperature. As the motor heats up, its coil resistance rises and the chopper on-time increases.

1.3 spreadCycle 3-Phase Motor Chopper

The spreadCycle chopper scheme for three phase motors (pat.fil.) is a precise and simple to use chopper principle, which automatically determines the optimum fast decay portion for the motor. Anyhow, a number of settings can be made in order to optimally fit the driver to the motor.

Three parameters control the chopper scheme for a three phase motor:

Parameter	Description	Setting	Comment
HYST	The hysteresis setting is the main control for the chopper and determines the chopper frequency. A higher setting introduces more current ripple and thus reduces frequency. A too low setting will result in the coil current only loosely following the target current and thus reduced microstep performance, especially in the current zero crossing. A too high setting can cause audible chopper noise. In order to use the full hysteresis range, be careful to set motor current to max. 28.	0 … 63	Hysteresis for the chopper
nosd	Selection of the *TOFF* insertion	0	Use *TOFF* setting for additional slow decay phases
		1	No slow decay phase
csync	This bit switches on *chopper synchronization*. If enabled, the chopper engine becomes reset with each motor fullstep, in order to avoid a beat occurring between full step sequence and chopper clock.	0	Chopper runs freely
		1	Synchronization enable
TOFF	Sets the slow decay time (*off time*). This setting also limits the maximum chopper frequency. For most 3-phase motor applications an off time will not be required. In this case, a dummy value needs to be programmed to this register to enable the driver and the *nosd* flag shall be set. Setting this parameter to zero completely disables all driver transistors and the motor can free-wheel.	0	chopper off

Attention!
Initialize START_SIN90_120 in MSLUTSTART correctly before operating a three phase motor.

Each chopper cycle is comprised of an *on phase*, a *fast decay phase* and a *slow decay phase* (see Figure 1.2). Optional additional slow decay phases can be added (switch off using *nosd* bit). The hysteresis determines the chopper frequency by forcing the driver to introduce some amount of current ripple into the motor coils. The motor inductivity determines the ability to follow a changing motor current. The duration of the on- and fast decay phase needs to cover at least the blank time, because the current comparator is disabled during this time.

A suitable positive value for the hysteresis can be calculated as follows:

$$f_{CHOP} = \frac{V_M}{2 * \frac{2}{3} L_{COIL} * I_{HYST}}$$

Where:
f_{CHOP} is the resulting chopper frequency.
I_{COIL} is the peak motor coil current at the maximum motor current setting CS,
R_{COIL} and L_{COIL} are motor coil inductivity and motor coil resistance.

The current hysteresis I_{HYST} results from the HYST setting as follows:

$$I_{HYST} = HYST * \frac{I_{COIL}}{2 * 248} * \frac{32}{CS + 1}$$

The calculated chopper frequency should preferably lie between 18 kHz and 60 kHz. If a too high chopper frequency results, you can try adding a slow decay phase.

Example:

For a three phase stepper motor with 3 mH, 3.2 Ω phase and 1.6 A RMS current at current setting CS=28 and HYST=40 operating from a 24 V supply:

$$I_{HYST} = 40 * \frac{1.6\,A}{496} * \frac{32}{29} = 142\,mA$$

$$f_{CHOP} = \frac{24\,V}{\frac{4}{3} * 3\,mH * 0.142\,A} = 42\,kHz$$

Note:
The choice of a hysteresis setting of 40 results in a good chopper frequency, but a higher hysteresis also will not harm.

Experimenting with the Motor

The setting can also be determined by experimenting with the motor: a too low setting will result in reduced microstep accuracy, while a too high setting will lead to more chopper noise and motor power dissipation. The correct setting can be determined best by rotating the motor slowly, and increasing hysteresis setting, until the motion of the motor is very smooth (feel with fingers or add a long pointer to the axis, e.g. laser pointer).

Measuring Motor Currents at the Sense Resistors

Alternatively you can measure the motor currents at the sense resistor with a current probe or with an oscilloscope. Check the waves for a pure sine wave. A further increment of the hysteresis setting will lower the chopper frequency and might generate audible chopper noise at some point.

For high inductivity motors, audible noise might occur at optimum setting. Increase supply voltage, or choose a motor with a different, higher current winding.

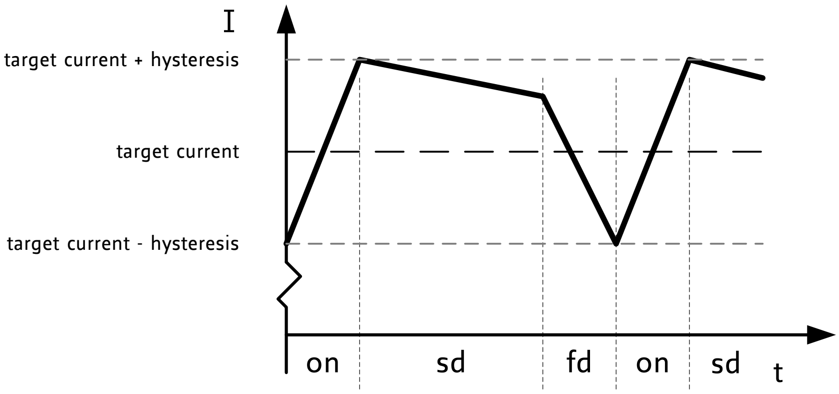

Figure 1.2 spreadCycle 3 phase chopper scheme showing the coil current within a chopper cycle

For additional information on the parameterization of the chopper, please refer to the spreadCycle application note AN001, chapter 4.

2 Disclaimer

3 Revision History

Document Revision

Version	Date	Author BD – Bernhard Dwersteg	Description
1.00	2014-NOV-24	BD	Initial version copied from datasheet version 1.08

4 References

TMC5062 datasheet, www.trinamic.com
Appnote spreadCycle, www.trinamic.com