Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

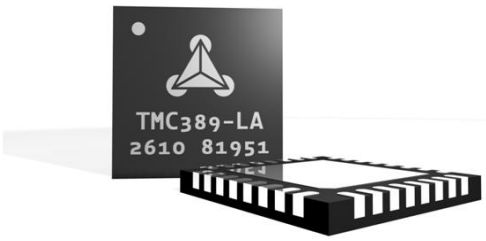
We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

TMC389–DATASHEET

Energy saving high resolution microstepping three phase stepper driver with step and direction interface and external power stage



coolStep™

3 PHASE STEPPER

TRINAMIC® Motion Control GmbH & Co. KG
Hamburg, GERMANY

www.trinamic.com

1 Features

The TMC389 is an energy efficient three phase stepper motor driver for high resolution microstepping applications. It integrates a low resonance three phase chopper for quiet motor operation. Its step and direction interface allows simple use. An SPI™ management interface allows for parameterization and diagnostics. The TMC389 directly drives 3 external N/P channel dual MOSFETs for motor currents up to 8A and up to 60V. Protection and diagnostic features further reduce system cost and increase reliability.

Highlights

- Up to 171 microsteps (256 sine wave steps) using step/direction interface or 20 Bit SPI™ interface
- High precision sensorless motor load measurement stallGuard2
- Energy efficiency and coolness by automatic load dependant motor current regulation coolStep™: Save up to 75% of energy!
- Internal microstep extrapolation allows 256 wave step smoothness with low frequency step input
- Dual edge step option allows half step frequency requirement, e.g. for opto-couplers
- Up to 8A Motor current using external N&P channel MOSFET pairs
- Synchronous rectification reduces transistor heating
- 9V to 60V operating voltage (peak)
- 3.3V or 5V interface
- QFN32 package for extremely small solution with superior thermal performance
- EMV optimized current controlled gate drivers – up to 45mA gate current
- Overcurrent, short to GND and overtemperature protection and diagnostics integrated

Applications

- Precision three phase stepper motor drives
- Stage lighting
- Medical applications
- Optical applications
- Robotics

Motor type

- 3 phase Stepper

2 Table of contents

2.1 Disclaimer

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.

Information given in this data sheet is believed to be accurate and reliable. However no responsibility is assumed for the consequences of its use nor for any infringement of patents or other rights of third parties which may result from its use.

Specifications are subject to change without notice.

™: All trademarks used are property of their respective owners.

3 Principle of operation

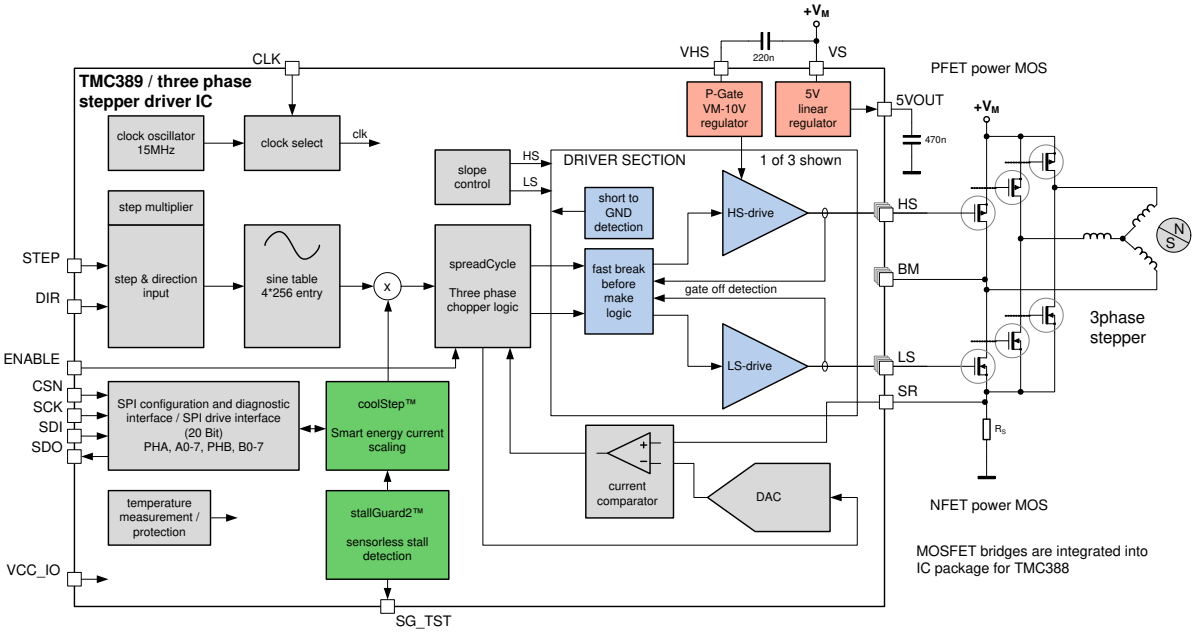


figure 1: Basic application block diagram

3.1 Moving the motor

3.1.1 Step and direction control

The TMC389 is a chopped stepper motor driver with integrated sequencer and SPI interface. It provides two possibilities to control the motor: The motor can be controlled by applying pulses on the step and direction interface, following an initialization phase which uses the SPI interface to parameterize the driver for the application. Control and diagnostic registers give the flexibility to react to changing operation conditions and to modify the behavior of the chip when it receives a step impulse. An internal microstep table supplies sine and cosine values which control the motor current for each step. Each step impulse advances the step pointer in the tables and hence leads to the IC executing the next microstep.

3.1.2 SPI control

A second mode of operation uses the SPI interface, only. The motor coil currents can be controlled via the SPI interface, while taking advantage of all other control and diagnostic functions. This mode is more flexible, as the microstep waves can be specially adapted to the motor to give the best fit for smoothest operation. It requires slightly more CPU overhead to look up the driver tables and to send out new current values for both coils. The SPI update rate corresponds to the step rate at low velocities. At highest velocities the update rate can be limited to a few 10kHz or some 100kHz, depending on the processor power, or alternatively to an update rate corresponding to a fullstep.

3.2 Chopped motor coil driver

The driver use a cycle by cycle chopper mode: The motor current becomes regulated by comparing the motor current to a set value for each chopper cycle. This constant off time chopper scheme allows highest dynamic. The spreadCycle chopper scheme automatically integrates a fast decay cycle and guarantees smooth zero crossing performance. In an optional operation mode, fast decay length per cycle can be selected by the user. In this classic constant off time mode, zero crossing can be optimized by setting a programmable current offset.

3.3 Energy efficient driver with load feedback

The TMC389 integrates a high resolution load measurement stallGuard2™, which allows sensing the mechanical load on the motor. This gives more information on the drive allowing functions like sensorless homing. Its coolStep™ feature uses load measurement information to reduce the motor current to the minimum motor current required in the actual load situation. This saves lots of energy and keeps components cool, making the drive an efficient and precise solution.

4 Pinning

4.1 TMC389-LA

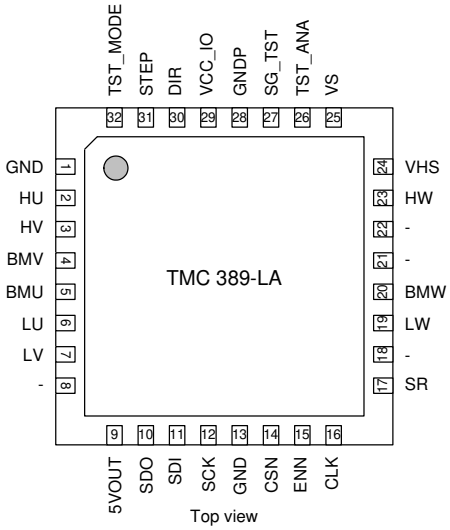


figure 2: TMC389 pinning

4.2 Package codes

Type	Package	Temperature range	Code/marking
TMC389	QFN32 (ROHS)	-40°C ... +125°C	TMC389-LA
TMC389 eng. sample	QFN32 (ROHS)	-40°C ... +125°C	TMC389-ES

4.3 Dimensional drawings

For drawings, see next page.
Attention: Drawings not to scale.

4.3.1 QFN32 dimensions

Parameter	Ref	Min	Nom	Max
total thickness	A	0.80	0.85	0.90
stand off	A1	0.00	0.035	0.05
mold thickness	A2	-	0.65	0.67
lead frame thickness	A3		0.203	
lead width	b	0.2	0.25	0.3
body size X	D		5.0	
body size Y	E		5.0	
lead pitch	e		0.5	
exposed die pad size X	J	3.2	3.3	3.4
exposed die pad size Y	K	3.2	3.3	3.4
lead length	L	0.35	0.4	0.45
package edge tolerance	aaa			0.1
mold flatness	bbb			0.1
coplanarity	ccc			0.08
lead offset	ddd			0.1
exposed pad offset	eee			0.1

All dimensions are in mm.

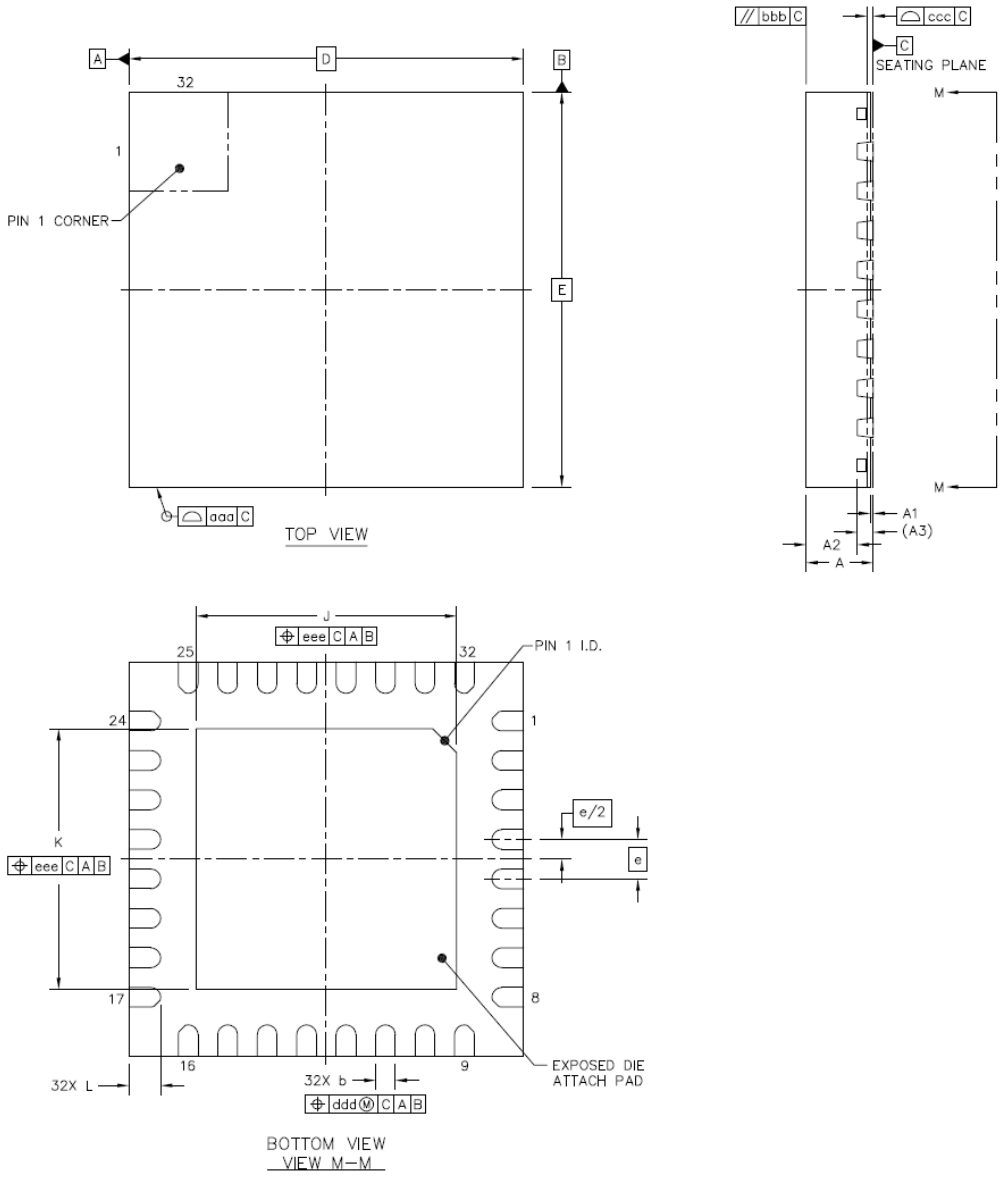


figure 3: QFN32 5x5 dimensions

5 Block diagram

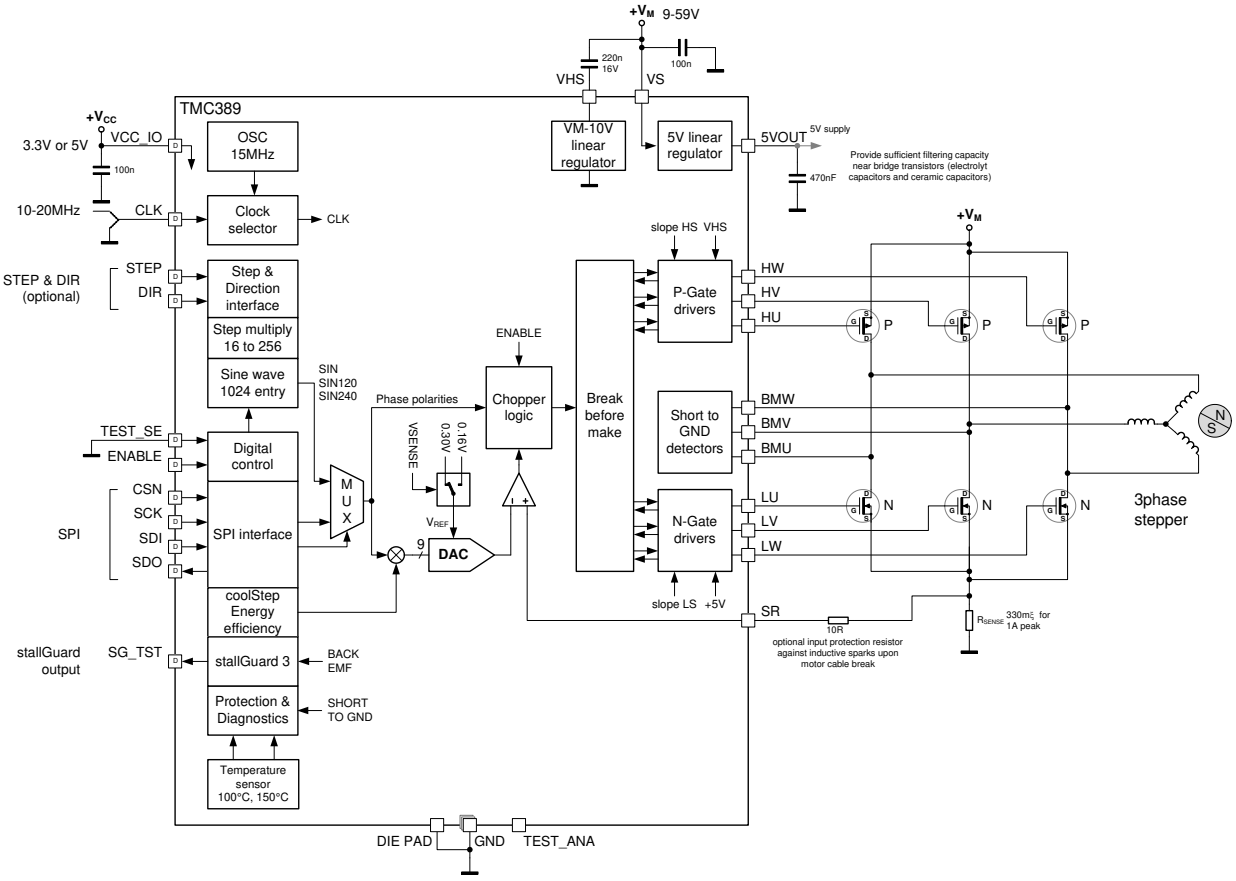


figure 4: TMC389 block and application schematic

The application schematic shows the basic building blocks of the IC and the connections to the power bridge transistors, as well as the power supply. The connection of the digital interface lines to the microcontroller and / or a motion controller is specific to the system architecture and the micro-controller type. Do not leave any input floating over extended periods of time, as there are no pull up or pull down resistors integrated. The choice of power MOSFETs for the TMC389 depends on the desired motor current and supply voltage. Please refer chapter 17.4. For even higher motor current capability, external MOSFET drivers can be added using full N channel bridges.

5.1 Pin description of TMC389-LA

Pin	Number	Type	Function
GND	1, 13		Digital and analog low power GND
HU HV HW	2, 3, 23	O (VS)	High side P-channel driver output. Becomes driven to VHS to switch on MOSFET.
BMV BMU BMW	4 5 20	I (VS)	Sensing input for bridge outputs. Used for short to GND protection. May be tied to VS if unused.
LU LV LW	6 7 19	O 5V	Low side MOSFET driver output. Becomes driven to 5VOUT to switch on MOSFET.
SR	17	AI	Sense resistor input of chopper driver.
5VOUT	9		Output of internal 5V linear regulator. This voltage is used to supply the low side drivers and internal analog circuitry. An external capacitor to GND close to the pin is required. Place the capacitor near to pin 9 and pin 13. 470nF ceramic are sufficient for most applications, an additional tantalum capacitor (10µF or more) improves performance with high gate charge MOSFETs.
SDO	10	DO VIO	Data output of SPI interface (Tristate)
SDI	11	DI VIO	Data input of SPI interface (Scan test input in test mode)
SCK	12	DI VIO	Serial clock input of SPI interface (Scan test shift enable input in test mode)
CSN	14	DI VIO	Chip select input of SPI interface
ENN	15	DI VIO	Enable not input for drivers. Switches off all MOSFETs.
CLK	16	DI VIO	Clock input for all internal operations. Tie low to use internal oscillator. A high signal disables the internal oscillator until power down.
VHS	24		High side supply voltage (motor supply voltage - 10V)
VS	25		Motor supply voltage
TST_ANA	26	AO VIO	Analog mode test output. Leave open or tie to GND for normal operation.
SG_TST	27	DO VIO	stallGuard2™ output. Signals motor stall (high active).
GNDP	28		Power GND for MOSFET drivers. Connect directly to GND
VCC_IO	29		Input / output supply voltage VIO for all digital pins. Tie to digital logic supply voltage. Allows operation in 3.3V and 5V systems.
DIR	30	DI VIO	Direction input. Is sampled upon detection of a step to determine stepping direction. An internal glitch filter for 60ns is provided.
STEP	31	DI VIO	Step input. An internal glitch filter for 60ns is provided.
TST_MODE	32	DI VIO	Test mode input. Puts IC into test mode. Tie to GND for normal operation.
Exposed die pad	-	GND	Connect the exposed die pad to a GND plane. It is used for cooling of the IC and may either be left open or be connected to GND.

6 SPI™ mode shift register

The TMC389 requires a configuration via SPI prior to operation. Its SPI interface also allows for reading back status flags. The SPI interface can operate up to the half clock frequency. The MSB (bit 19) is transmitted first. See chapter 6.6 and 19.2 for more details.

6.1 Overview (write)

Register/ Bit	DRVCTRL (*SDOFF*=1)	DRVCTRL (*SDOFF*=0)	CHOPCONF	SMARTEN	SGCSCONF	DRVCONF
19	0	0	1	1	1	1
18	0	0	0	0	1	1
17	PHU	-	0	1	0	1
16	CU7	-	TBL1	0	SFILT	TST
15	CU6	-	TBL0	SEIMIN	SSPD	SLPH1
14	CU5	-	-	SEDN1	SGT6	SLPH0
13	CU4	-	RNDTF	SEDN0	SGT5	SLPL1
12	CU3	-	CSYNC	-	SGT4	SLPL0
11	CU2	-	CDIR	SEMAX3	SGT3	-
10	CU1	-	NOSD	SEMAX2	SGT2	DISS2G
9	CU0	INTPOL	HYST5	SEMAX1	SGT1	TS2G1
8	PHV	DEDGE	HYST4	SEMAX0	SGT0	TS2G0
7	CV7	-	HYST3	-	-	*SDOFF*
6	CV6	-	HYST2	SEUP1	-	VSENSE
5	CV5	-	HYST1	SEUP0	-	*RDSEL1*
4	CV4	-	HYST0	-	CS4	*RDSEL0*
3	CV3	MRES3	TOFF3	SEMIN3	CS3	-
2	CV2	MRES2	TOFF2	SEMIN2	CS2	-
1	CV1	MRES1	TOFF1	SEMIN1	CS1	-
0	CV0	MRES0	TOFF0	SEMIN0	CS0	-

6.2 Overview (read)

Bit	*RDSEL*=00	*RDSEL*=01	*RDSEL*=10
19	MSTEP9	SG9	SG9
18	MSTEP8	SG8	SG8
17	MSTEP7	SG7	SG7
16	MSTEP6	SG6	SG6
15	MSTEP5	SG5	SG5
14	MSTEP4	SG4	SE4
13	MSTEP3	SG3	SE3
12	MSTEP2	SG2	SE2
11	MSTEP1	SG1	SE1
10	MSTEP0	SG0	SE0
9	-	-	-
8	-	-	-
7	STST		
6	-		
5	OL		
4	-		
3	S2G		
2	OTPW		
1	OT		
0	SG		

6.3 Driver control register bit assignment

The driver control register is used to operate the device in SPI mode by setting phase currents for Phase U and Phase V. Phase W is automatically calculated from the formula CW=-(CU+CV). In StepDir mode, it selects Step and Direction interface specific parameters. They need to be initialized once upon power up, and whenever basic parameters are required to be changed. Only write access is possible.

Notation of hexadecimal and binary numbers:
0x precedes a hexadecimal number, % precedes a multi-bit binary number

The meaning of register 0 depends on the mode selection between SPI mode and StepDir mode as selected by SDOFF (configuration register 11, bit 7).

6.3.1 Driver control register bit assignment in SPI mode

DRVCTRL		**write 0xxx, SDOFF=1**	
Bit	**Name**	**Function**	**Comment**
19	CFR	select configuration register	**0**: Operation mode dependent settings (see SDOFF)
18	-	reserved	set to 0
17	PHU	Polarity U	
16	CU7	Current U MSB	0 to max. 248 due to hysteresis setting. Depending on the hysteresis setting, the maximum value becomes even lower. The resulting value is not allowed to overflow 255.
15	CU6		
14	CU5		
13	CU4		
12	CU3		
11	CU2		
10	CU1		
9	CU0	Current U LSB	
8	PHV	Polarity V	
7	CV7	Current V MSB	0 to max. 248 due to hysteresis setting. Depending on the hysteresis setting, the maximum value becomes even lower. The resulting value is not allowed to overflow 255.
6	CV6		
5	CV5		
4	CV4		
3	CV3		
2	CV2		
1	CV1		
0	CV0	Current V LSB	

6.3.2 Driver control register bit assignment in StepDir mode

<table>
<tr><th colspan="2">DRVCTRL</th><th colspan="2">write 0xxx, SDOFF=0</th></tr>
<tr><th>Bit</th><th>Name</th><th>Function</th><th>Comment</th></tr>
<tr><td>19</td><td>CFR</td><td>select configuration register</td><td>0: Operation mode dependent settings (see SDOFF)</td></tr>
<tr><td>18</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>17</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>16</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>15</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>14</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>13</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>12</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>11</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>10</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>9</td><td>INTPOL</td><td>enable step interpolation</td><td>1: Enable step impulse multiplication by 16. Only in resolution 16x microsteps, the microstepping becomes extrapolated to 256 microsteps. Interpolation is possible starting below step distance of max. 2^20 CLK periods.</td></tr>
<tr><td>8</td><td>DEDGE</td><td>enable double edge step pulses</td><td>1: Enable step impulse at each step edge to reduce step frequency requirement</td></tr>
<tr><td>7</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>6</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>5</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>4</td><td>-</td><td>reserved</td><td>set to 0</td></tr>
<tr><td>3</td><td>MRES3</td><td rowspan="4">micro step resolution for step/direction mode</td><td rowspan="4">%0000 … %1000

<table>
<tr><th>MRES</th><th>electrical</th><th>mechanical</th></tr>
<tr><td>%0000</td><td>256</td><td>170.66</td></tr>
<tr><td>%0001</td><td>128</td><td>85.33</td></tr>
<tr><td>%0010</td><td>64</td><td>42.66</td></tr>
<tr><td>%0011</td><td>32</td><td>21.33</td></tr>
<tr><td>%0100</td><td>16</td><td>10.66</td></tr>
<tr><td>%0101</td><td>8</td><td>5.33</td></tr>
<tr><td>%0110</td><td>4</td><td>2.66</td></tr>
<tr><td>%0111</td><td>2</td><td>1.33</td></tr>
<tr><td>%1000</td><td>1</td><td>0.66</td></tr>
</table>

The electrical values given describe the number of steps taken per electrical quarter sine wave. The resulting mechanical microstep resolution which describes the number of microsteps between two motor fullsteps is 2/3 of the corresponding value. Please take into account, that the microstep position when switching to a lower resolution determines the sequence of patterns.
step width=2^MRES [electrical microsteps]
step width=2/3*2^MRES [motor microsteps]</td></tr>
<tr><td>2</td><td>MRES2</td></tr>
<tr><td>1</td><td>MRES1</td></tr>
<tr><td>0</td><td>MRES0</td></tr>
</table>

6.4 Configuration register bit assignment

The configuration registers select the mode of operation and set all motor and application dependent parameters. They need to be initialized once upon power up, and whenever basic parameters are required to be changed. Only write access is possible.

CHOPCONF		**write 100x: Chopper Configuration**		
Bit	**Name**	**Function**	**Comment**	
19	CFR	select configuration register	**1**: Configuration register	
18	CFRSEL1	select configuration register	**%00**: Chopper configuration register	
17	CFRSEL0			
16	TBL1	blank time select	%00 … %11:	
15	TBL0		Set comparator blank time to 16, 24, 36 or 54 clocks	
14	CHM	chopper mode	0	Standard mode
			1	unused
13	RNDTF	random TOFF time	0	Chopper off time is fixed as set by bits t_{OFF}
			1	Random mode, t_{OFF} is random modulated by dN_{CLK}= -12 … +3 clocks.
12	CSYNC	chopper synchronization	0	Chopper runs freely
			1	Chopper becomes synchronized to step frequency
11	CDIR	chopper direction	0	Chopper direction is WVU with DIR input=0 Choose for turn left in SPI operation
			1	Chopper direction is UVW with DIR input=0 Choose for turn right (U→V→W) in SPI operation and for StepDir operation
10	NOSD	skip slow decay phase	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. Minimum blank time then is 36 clocks.
9	HYST5	hysteresis value	DAC hysteresis setting: %000000 … %111111: 0 … 63 (1/512 of this setting adds to coil 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).*	
8	HYST4			
7	HYST3			
6	HYST2			
5	HYST1			
4	HYST0			
3	TOFF3	off time and driver enable	Off time setting for constant t_{OFF} chopper N_{CLK}= 12 + 32*TOFF (Minimum is 64 clocks) %0000: Driver disable, all bridges off %0001: not allowed %0010 … %1111: 2 … 15	
2	TOFF2			
1	TOFF1			
0	TOFF0			

SMARTEN		write 1010: Smart energy control coolStep™	
Bit	**Name**	**Function**	**Comment**
19	CFR	select configuration register	**1**: configuration register
18	CFRSEL1	select configuration register	**%01**: coolStep configuration register
17	CFRSEL0		
16	-	reserved	set to 0
15	SEIMIN	minimum current for smart current control	0: 1/2 of current setting (CS) 1: 1/4 of current setting (CS)
14	SEDN1	current down step speed	%00: for each 32 stallGuard values decrease by one %01: for each 8 stallGuard values decrease by one %10: for each 2 stallGuard values decrease by one %11: for each stallGuard value decrease by one
13	SEDN0		
12	-	reserved	set to 0
11	SEMAX3	stallGuard hysteresis value for smart current control	If the stallGuard 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 steps per measured stallGuard value %00 … %11: 1, 2, 4, 8
5	SEUP0		
4	-	reserved	set to 0
3	SEMIN3	minimum stallGuard value for smart current control and smart current enable	If the stallGuard result falls below SEMIN*32, the motor current becomes increased to reduce motor load angle. %0000: coolStep current control off %0001 … %1111: 1 … 15
2	SEMIN2		
1	SEMIN1		
0	SEMIN0		

SGCSCONF		write 110x: Load measurement stallGuard2 and Current Setting		
Bit	**Name**	**Function**	**Comment**	
19	CFR	select configuration register	**1**: Configuration register	
18	CFRSEL1	select configuration register	**%10**: stallGuard and current configuration register	
17	CFRSEL0			
16	SFILT	stallGuard filter enable	0	Standard mode, high time resolution for stallGuard
			1	Filtered mode, stallGuard signal updated for each six fullsteps only to compensate for motor tolerances
15	SSPD	stallGuard speed	0	Standard mode, high time resolution for stallGuard
			1	StallGuard uses more filtering, use for low motor velocity only
14	SGT6	stallGuard threshold value	This signed value controls stallGuard 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 stallGuard less sensitive and requires more torque to indicate a stall.	
13	SGT5			
12	SGT4			
11	SGT3			
10	SGT2			
9	SGT1			
8	SGT0			
7	-	reserved	set to 0	
6	-	reserved	set to 0	
5	-	reserved	set to 0	
4	CS4	current scale (scales digital currents A and B)	Current scaling for SPI and step/direction operation %00000 … %11111: 1/32, 2/32, 3/32, … 32/32 *Attention: Maximum possible current scale setting might be below 31, depending on hysteresis setting.*	
3	CS3			
2	CS2			
1	CS1			
0	CS0			

DRVCONF		write 111x: Driver Configuration	
Bit	**Name**	**Function**	**Comment**
19	CFR	select configuration register	**1**: Configuration register
18	CFRSEL1	select configuration register	**%11**: Driver configuration register
17	CFRSEL0		
16	TST	reserved TEST mode	Set to 0. When 1, SG_TST outputs digital test values, and TEST_ANA outputs analog test values. Selection is done by SGT0 and SGT1 (%00 … %10): For TEST_ANA: anatest_2vth, anatest_dac_out, anatest_vdd_half. For SG_TST: comp_A, comp_B, CLK
15	SLPH1	Slope control high side	%00: min, %01: min + tc, %10: med + tc, %11: max In temperature compensated mode (tc), the driver strength is increased if the overtemperature prewarning temperature is reached. This compensates for temperature dependence of high side slope control.
14	SLPH0		
13	SLPL1	Slope control low side	00, 01: min, 10: med, 11: max
12	SLPL0		
11	-	reserved	set to 0
10	DISS2G	short to GND protection disable	0: Short to GND protection is on 1: Short to GND protection is disabled
9	TS2G1	short to GND detection timer	%00: 3.2µs %01: 1.6µs %10: 1.2µs %11: 0.8µs
8	TS2G0		
7	SDOFF	Step Direction input off	0: Enable step/direction mode (StepDir) 1: Enable SPI mode
6	VSENSE	sense resistor voltage based current scaling	0: Full scale sense resistor voltage is 305mV 1: Full scale sense resistor voltage is 165mV (refers to a current setting of 31 and DAC value 255)
5	RDSEL1	Select value for read out (RD bits)	%00 Microstep position read back
4	RDSEL0		%01 stallGuard level read back %10 stallGuard and smart current level read back %11 Reserved, do not use
3	-	reserved	set to 0
2	-	reserved	set to 0
1	-	reserved	set to 0
0	-	reserved	set to 0

6.5 Bit assignment for read

Information can be read back from the driver on each access. Different information may be required, depending on the application. This is selected by the bits RDSEL in the register DRVCONF.

DRVSTATUS		**read status information – Partially selected by RDSEL in DRVCONF**		
Bit	**Name**	**Function**	**Comment**	
19	RD9	microstep position in internal sine table for phase U or stallGuard bits 7 to 0 or stallGuard bits 7 to 5 and current control scale	RDSEL=%00	Actual microstep position in sine table for phase U in step/direction operation (MSTEP) (MSTEP9=PHA)
18	RD8			
17	RD7			
16	RD6		RDSEL=%01	Bits 9 … 0 of stallGuard result (SG)
15	RD5		RDSEL=%10	Bits 9 … 5 of stallGuard result (SG) and actual current control scaling Bits 4 … 0 for monitoring smart energy current setting (SE)
14	RD4			
13	RD3			
12	RD2			
11	RD1			
10	RD0			
9	0	reserved	-	
8	0	reserved	-	
7	STST	stand still step indicator	1: Indicates, that no step impulse occurred on the step input during the last 2^20 clock cycles.	
6	0	reserved	-	
5	OL	open load indicator	Flag becomes set, if no chopper event has happened during the last period with constant coil polarity. Only a current above 1/16 of maximum setting can reset this flag!	
4	0	reserved	-	
3	S2G	short to GND detection bits on high side transistors	1: Short condition is detected, driver is currently shut down (clear short condition by disabling driver) In a short circuit condition, the chopper cycle becomes terminated. The short counter is increased by each short circuit. It becomes decreased by one for each phase polarity change. The driver becomes shut down when the counter reaches 3, until the short condition becomes reset by disabling and re-enabling the driver.	
2	OTPW	Overtemperature pre-warning	1: Warning threshold is exceeded	
1	OT	Overtemperature	1: Driver is shut down due to overtemperature	
0	SG	stallGuard status	1: stallGuard threshold is reached, SG output high	

6.6 SPI™ timing

The SPI interface uses the system clock to synchronize all input and output signals. This limits the SPI clock frequency to at maximum half of the system clock frequency. For an asynchronous system using the internal clock, some 10 percent of safety margin should be used, assuming the minimum internal and maximum SPI master clock frequency, in order to ensure a reliable data transmission.

All SPI inputs as well as the ENN input are internally filtered to avoid triggering on short time glitches.

The minimum number of SCK clock pulses to be sent is 20. Additional clocks are possible – the additional bits shifted in on SDI become shifted through to the SDO pin delayed by 20 clocks via the internal shift register. The active CSN time (low) must span the whole data transmission. Upon CSN going inactive (high), the shift register content becomes latched into the internal control register.

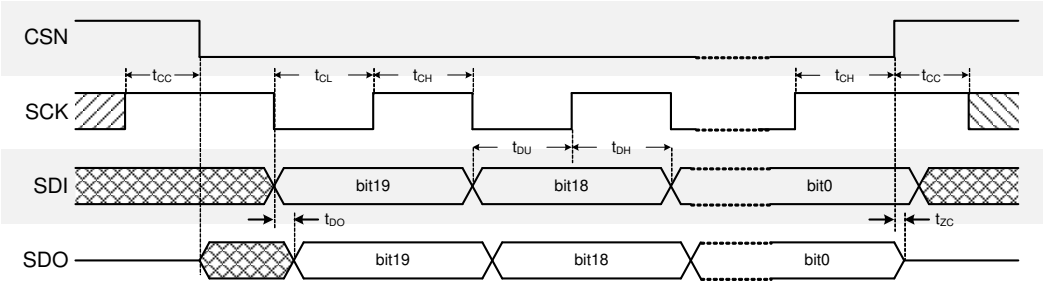


figure 5: SPI timing

Hint
Usually this SPI timing is referred to as SPI MODE 3

SPI interface timing	**AC-Characteristics** clock period is t_{CLK}					
Parameter	**Symbol**	**Conditions**	**Min**	**Typ**	**Max**	**Unit**
SCK valid before or after change of CSN	t_{CC}		10			ns
CSN high time	t_{CSH}	*) Min time is for synchronous CLK with SCK high one t_{CH} before CSN high only	t_{CLK}*)	>2t_{CLK} +10		ns
SCK low time	t_{CL}	*) Min time is for synchronous CLK only	t_{CLK}*)	>t_{CLK}+10		ns
SCK high time	t_{CH}	*) Min time is for synchronous CLK only	t_{CLK}*)	>t_{CLK}+10		ns
SCK frequency using internal clock	f_{SCK}	assumes minimum OSC frequency			4	MHz
SCK frequency using external 16MHz clock	f_{SCK}	assumes synchronous CLK			8	MHz
SDI setup time before rising edge of SCK	t_{DU}		10			ns
SDI hold time after rising edge of SCK	t_{DH}		10			ns
Data out valid time after falling SCK clock edge	t_{DO}	no capacitive load on SDO			t_{FILT}+5	ns
SDI, SCK and CSN filter delay time	t_{FILT}	rising and falling edge	12	20	30	ns

7 Step and direction interface

The step and direction interface allows easy movement of the motor and is a simple real time interface for a motion controller. Its pulse rate multiplier allows smooth motor operation even with reduced pulse bandwidth.

7.1 Timing

The step and direction interface pins are sampled synchronously with the clock signal. An internal analog filter removes disturbances caused by glitches on the signals, e.g. caused by long PCB traces. Despite this, the signals should be filtered and / or differentially transmitted, if the step source is far from the TMC389 and especially if the step signals are interconnected via cables.

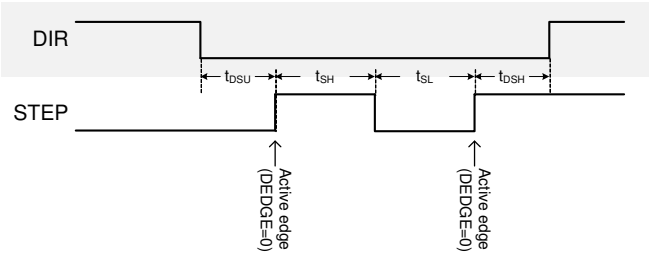


figure 6: STEP and DIR timing

STEP and DIR interface timing	**AC-Characteristics** clock period is t_{CLK}					
Parameter	**Symbol**	**Conditions**	**Min**	**Typ**	**Max**	**Unit**
step frequency (at maximum microstep resolution)	f_{STEP}	DEGDE=0			½ f_{CLK}	
		DEDGE=1			¼ f_{CLK}	
(electrical) fullstep frequency	f_{FS}				f_{CLK}/512	
STEP input low time	t_{SL}		max(t_{FILTSD}, t_{CLK}+20)			ns
STEP input high time	t_{SH}		max(t_{FILTSD}, t_{CLK}+20)			ns
DIR to STEP setup time	t_{DSU}		20			ns
DIR after STEP hold time	t_{DSH}		20			ns
STEP and DIR spike filtering time	t_{FILTSD}	rising and falling edge	36	60	85	ns
STEP and DIR sampling relative to rising CLK input	$t_{SDCLKHI}$	before rising edge of CLK input		t_{FILTSD}		ns

7.2 Internal microstep table

The internal microstep table uses 1024 sine wave entries to generate the wave. Its amplitude is +/-248 rather than +/-255, leaving some headroom for hysteresis setting within an 8 bit amplitude range. The step width depends on the microstep resolution setting. Depending on the DIR input, the microstep counter is increased (DIR=0) or decreased (DIR=1) with each STEP pulse by the step width. Due to the symmetry of the sine wave, only a quarter of the table needs to be stored. The phase V wave uses a phase shift of 120°. The W wave is calculated as CW=-(CU+CV). Despite many entries in the last quarter of the table being equal, the electrical angle continuously changes, because either sine wave or cosine wave is in an area, where the current vector changes monotonously from position to position.

Entry	0-31	32-63	64-95	96-127	128-159	160-191	192-223	224-255
0	1	49	96	138	176	207	229	243
1	2	51	97	140	177	207	230	244
2	4	52	98	141	178	208	231	244
3	5	54	100	142	179	209	231	244
4	7	55	101	143	180	210	232	244
5	8	57	103	145	181	211	232	245
6	10	58	104	146	182	212	233	245
7	11	60	105	147	183	212	233	245
8	13	61	107	148	184	213	234	245
9	14	62	108	150	185	214	234	246
10	16	64	109	151	186	215	235	246
11	17	65	111	152	187	215	235	246
12	19	67	112	153	188	216	236	246
13	21	68	114	154	189	217	236	246
14	22	70	115	156	190	218	237	247
15	24	71	116	157	191	218	237	247
16	25	73	118	158	192	219	238	247
17	27	74	119	159	193	220	238	247
18	28	76	120	160	194	220	238	247
19	30	77	122	161	195	221	239	247
20	31	79	123	163	196	222	239	247
21	33	80	124	164	197	223	240	247
22	34	81	126	165	198	223	240	248
23	36	83	127	166	199	224	240	248
24	37	84	128	167	200	225	241	248
25	39	86	129	168	201	225	241	248
26	40	87	131	169	201	226	241	248
27	42	89	132	170	202	226	242	248
28	43	90	133	172	203	227	242	248
29	45	91	135	173	204	228	242	248
30	46	93	136	174	205	228	243	248
31	48	94	137	175	206	229	243	248

figure 7: internal microstep table showing the first quarter of the sine wave

7.3 Switching between different microstep resolutions

In principle, the microstep resolution can be changed at any time. The microstep resolution determines the increment respectively the decrement, the TMC389 uses for advancing in the microstep table. At maximum resolution, it advances one step for each step pulse. At half resolution, it advances two steps and so on. This way, a change of resolution is possible transparently at each time. However, you may experience the motor behavior becoming direction dependant, when switching microstep resolutions. This behavior results from table sampling points not evenly shifted inside the microstep table with respect to the step width. To avoid this, always switch to a lower resolution, when the actual microstep position is a multiple of the desired table step width. This is always satisfied at position zero in the microstep table.

7.4 Step rate multiplier and stand still detection

The step rate multiplier can be enabled by setting the INTPOL bit. It supports a 16 microstep setting and Step/Dir mode, only. In this setting, each step impulse at the input causes the execution of 16 times 1/256 microsteps. The step rate for the 16 microsteps is determined by measuring the time interval of the previous step pulses and dividing it into 16 equal parts. This way, a smooth motor movement like in 256 microstep resolution is achieved. The maximum time between two microsteps corresponds to 2^20 i.e. roughly one million clock cycles, in order to reach evenly distributed 1/256 sine wave steps. At 16MHz clock frequency, this results in a minimum step input frequency of 16Hz for step rate multiplier operation, i.e. one and a half motor fullsteps per second. A lower step rate causes the stand still flag to become set as soon as the time is expired. Execution of microsteps will happen with a frequency of 1/(2^16) clock frequency.

Attention: The step rate multiplier will only give good results with a stable microstep frequency. Do not use the DEDGE option, if the step input does not have a 50% duty cycle.

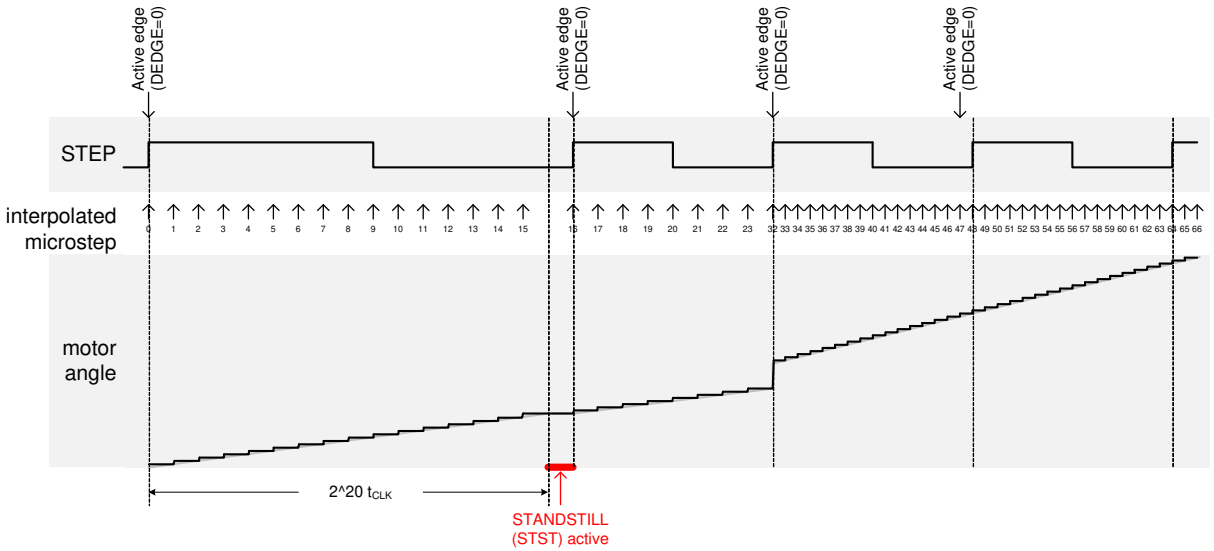


figure 8: Operation of the step multiplier in different situations

8 Current setting

The internal 5V supply voltage is used as a reference. To adapt the motor current, and to allow for different values of sense resistors, the voltage divider for full scale can be chosen as $V_{FS(HI)}$ =1/16 VDD or $V_{FS(LO)}$=1/30 of VDD. With this, the peak sense resistor voltage at a digital DAC control level of 248 is roughly 0.16V or 0.31V.

Using the internal sine wave table, which has the amplitude of 248, the RMS motor coil current thus can be calculated by:

$$I_{RMS} = \frac{CS+1}{32} * \frac{V_{FS}}{R_{SENSE}} * \frac{1}{\sqrt{2}}$$

The momentary motor current is calculated by:

$$I_{MOT} = \frac{CURRENT_{A/B}}{248} * \frac{CS+1}{32} * \frac{V_{FS}}{R_{SENSE}}$$

CS is the current scale setting as set by the CS bits and smart current scaler.

V_{FS} is the full scale voltage as determined by VSENSE control bit (please refer electrical characteristics).

$CURRENT_{A/B}$ is the value set by the current setting in SPI mode, or, the actual value from the internal sine wave table in Step/Dir mode.

Parameter	Description	Range	Comment
CS	*Current scale*. Scales both coil current values as taken from the internal sine wave table or from the SPI interface. For high precision motor operation, work with a current scaling factor in the range 16 to 28 (31), because scaling down the current values reduces the effective microstep resolution by making microsteps coarser. This setting also controls the maximum current value set by coolStep™. Keep in mind, that a value above 28 is only possible with reduced HYST setting: CS=31 requires HYST < 16 CS=30 requires HYST < 32 CS=29 requires HYST < 48	0 … 28 (… 31)	scaling factor: 1/32, 2/32, … 32/32
VSENSE	Allows control of the sense resistor *voltage range* or adaptation of one electronic module to different maximum motor currents.	0	310mV
		1	165mV

8.1 Considerations on the current sense resistors and layout

Sense resistors should be carefully selected. The full motor current flows through each sense resistor. They also see the switching spikes from the MOSFET bridges. A low inductance type resistor is required to prevent spikes causing ringing on the current measurement leading to instable measurement results. A low inductivity, low resistance layout is essential. Also, any common GND path of the sense resistors of different driver ICs needs to be prevented, because this would lead to coupling between both current sense signals. A massive GND plane is best. Especially for high current drivers or long motor cables, a spike damping with parallel capacitors can make sense (see figure 9). As the TMC389 is susceptible to negative overvoltages on the sense resistor inputs, an additional input protection resistor helps preventing damage in case of motor cable break or increased ringing on the motor lines in case of long motor cables.

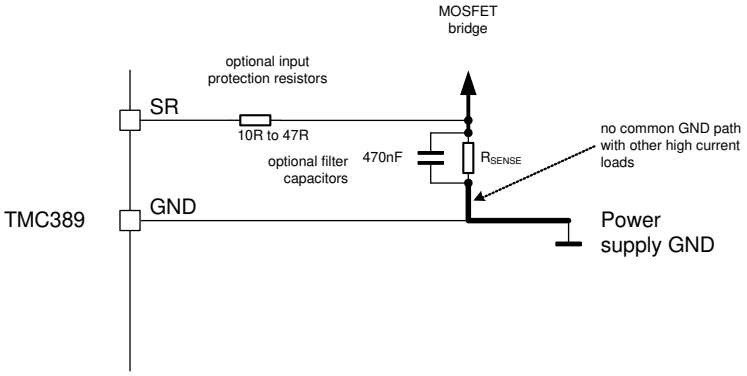


figure 9: Sense resistor grounding and optional parts

The sense resistor needs to be able to conduct the peak motor coil current in motor stand still situations, unless standby power is reduced. Under normal conditions, the sense resistor sees the coil RMS current.

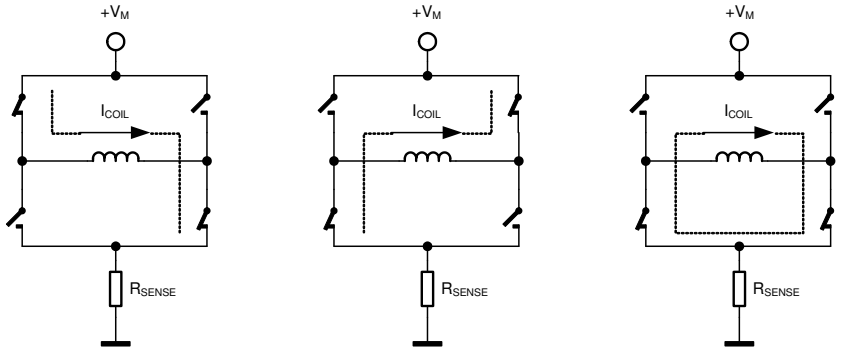
Peak sense resistor power dissipation:

$$P_{RSMAX} = \frac{\left(VSENSE * \frac{CS+1}{32}\right)^2}{R_{SENSE}}$$

For high current applications, power dissipation is halved by using the low VSENSE setting and using an adapted resistance value. Please be aware, that in this case any voltage drop in PCB traces has a larger influence on the result. A compact power stage layout with massive ground plane is best to avoid parasitic effects.

9 Chopper operation of the motor coils

The motor coils are operated using a chopper principle. The chopper regulates the current in the three coils by switching each coil in one of three different states. In figure 10 the different phases of a chopper cycle are shown for one coil, which is seen by each two half bridges. The figure assumes a triangle connection of the coils, but, a star connection of the coils virtually shows the same behavior. In the on-phase, the current is actively driven into the coils by connecting them to the power supply in the direction of the target current. A fast decay phase reverses the polarity of the coil voltage to actively reduce the current. The slow decay phase shorts the coil in order to let the current re-circulate. The current can be regulated using only on phases and fast decay phases. An optional slow decay phase can be inserted and might bring benefit for some low inductivity motors, by limiting the chopper frequency to an upper value. The current comparator can measure coil current, when the current flows through the sense resistor. Whenever the coil becomes switched, spikes at the sense resistors occur due to charging and discharging parasitic capacities. During this time (typically one or two microseconds), the current cannot be measured. It needs to be covered by the blank time setting.



on phase:	fast decay phase:	slow decay phase:
Current flows in direction of target current	Current flows in opposite direction of target current	Current re-circulation

figure 10: Chopper phases in motor operation

Parameter	Description	Range	Comment
TOFF	The *off time* setting controls the minimum chopper frequency. For most 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
		2…15	off time setting N_{CLK}=12+32*TOFF
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.	0 … 63	Hysteresis for the chopper
TBL	Selects the comparator *blank time*. This time needs to safely cover the switching event and the duration of the ringing on the sense resistor. For most low current drivers, a setting of 1 or 2 is good. For high current applications with large MOSFETs, a setting of 2 or 3 will be required.	0	16 t_{CLK}
		1	24 t_{CLK}
		2	36 t_{CLK}
		3	54 t_{CLK}
NOSD	Selection of the *TOFF* insertion	0	use TOFF setting for additional SD phases
		1	no slow decay phase
RNDTF	This bit switches on a *random off time* generator, which slightly modulates the off time t_{OFF} using a random polynomial giving a spread spectrum effect.	0	disable
		1	random modulation enable
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	disable
		1	synchronization enable
CDIR	The *chopper direction* should match the motor direction, to allow highest motor velocities. *In Step/Dir mode, this is done automatically, when CDIR is set to 1.* In SPI mode, either the DIR input or CDIR should be used, to adapt the chopper direction. Both, DIR input and CDIR are XORed.	0	DIR=0: WVU DIR=1: UVW
		1	DIR=0: UVW DIR=1: WVU

9.1 spreadCycle chopper

The spreadCycle chopper scheme (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.

Each chopper cycle is comprised of an on phase, a fast decay phase and a slow decay phase (see figure 11). 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. This is satisfied by choosing a positive value for the hysteresis as can be estimated by the following calculation:

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

where f_{CHOP} is the resulting chopper frequency. I_{COIL} is the peak motor coil current at the maximum motor current setting CS, and 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 18kHz and 60kHz. If a too high chopper frequency results, you can try adding a slow decay phase.

Example:

For a 60mm stepper motor with 0.76mH, 0.32Ω phase and 5.8A RMS current at CS=28 and HYST=40 operating from a 24V supply:

$$I_{HYST} = 40 * \frac{5.8A}{496} * \frac{32}{29} = 516mA$$

$$f_{CHOP} = \frac{24V}{\frac{4}{3} * 0.76mH * 0.516A} = 46kHz$$

With this, the choice of a hysteresis setting of 40 results in a good chopper frequency, but a higher hysteresis also will not harm.

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). Or, you can measure the motor currents with a current probe or with an oscilloscope at the sense resistor, and check the waves for a pure sine wave. A further increment of the hysteresis setting will lower chopper frequency and might at some point generate audible chopper noise. For high inductivity motors, audible noise might occur at optimum setting. Increase supply voltage, or choose a motor with a different, higher current winding.

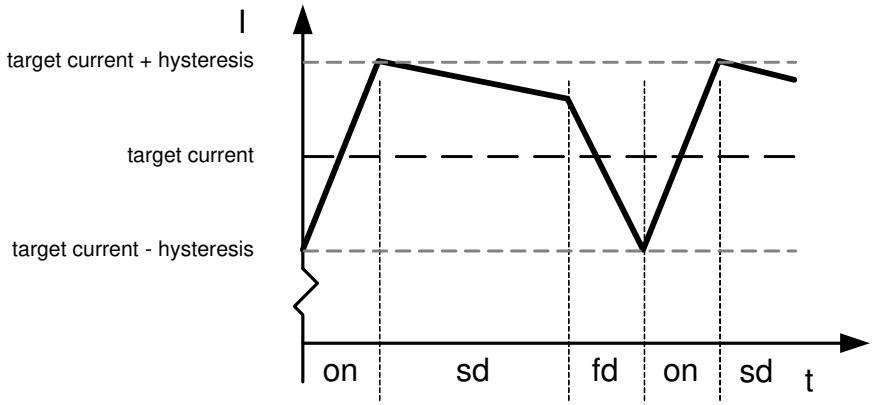


figure 11: spreadCycle (pat.fil.) chopper scheme showing the coil current within a chopper cycle