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

TMC5160 DATASHEET

Universal high voltage controller/driver for two-phase bipolar stepper motor. stealthChop™ for quiet movement. External MOSFETs for up to 20A motor current per coil. With Step/Dir Interface and SPI.

dcStep™

coolStep™

stallGuard2™

APPLICATIONS

Industrial Drives
Robotics
Textile, Sewing Machines
Factory Automation
Lab Automation
Liquid Handling
Medical
Office Automation
CCTV
ATM, Cash recycler
Pumps and Valves

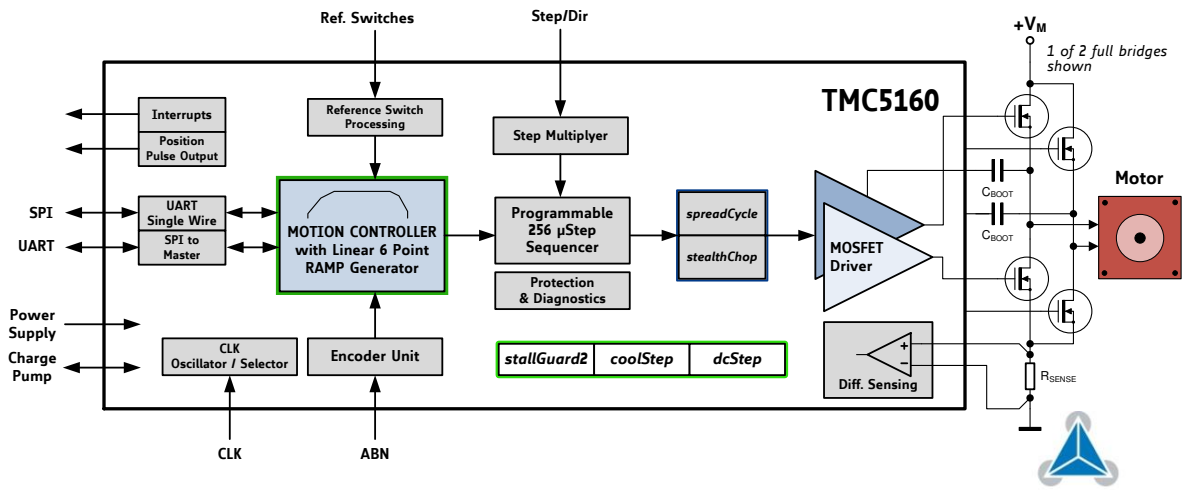
FEATURES AND BENEFITS

2-phase stepper motors up to 20A coil current (external MOSFETs)
Motion Controller with **sixPoint™** ramp
Step/Dir Interface with microstep interpolation **microPlyer™**
Voltage Range 8 … 60V DC
SPI & Single Wire UART
Encoder Interface and **2x Ref.-Switch Input**
Highest Resolution 256 microsteps per full step
stealthChop2™ for quiet operation and smooth motion
Resonance Dampening for mid range resonances
spreadCycle™ highly dynamic motor control chopper
dcStep™ load dependent speed control
stallGuard2™ high precision sensorless motor load detection
coolStep™ current control for energy savings up to 75%
Passive Braking and freewheeling mode
Full Protection & Diagnostics
Compact Size 9x9mm² TQFP48 package / 8x8mm² QFN

DESCRIPTION

The TMC5160 is a high power stepper motor controller and driver IC with serial communication interfaces. It combines a flexible ramp generator for automatic target positioning with industries' most advanced stepper motor driver. Using external transistors, highly dynamic, high torque drives can be realized. Based on TRINAMICs sophisticated spreadCycle and stealthChop choppers, the driver ensures absolutely noiseless operation combined with maximum efficiency and best motor torque. High integration, high energy efficiency and a small form factor enable miniaturized and scalable systems for cost effective solutions. The complete solution reduces learning curve to a minimum while giving best performance in class.

BLOCK DIAGRAM



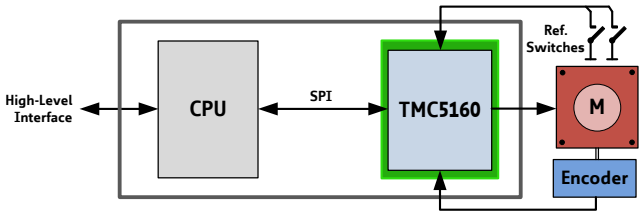
TRINAMIC Motion Control GmbH & Co. KG
Hamburg, Germany

TRINAMIC
MOTION CONTROL

APPLICATION EXAMPLES: HIGH VOLTAGE – MULTIPURPOSE USE

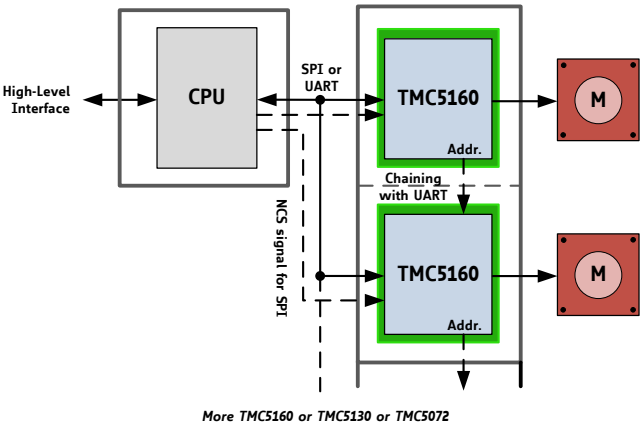
The TMC5160 scores with complete motion controlling features, powerful external MOSFET driver stages, and high quality current regulation. It offers a versatility that covers a wide spectrum of applications from battery powered, high efficiency systems up to embedded applications with 20A motor current per coil. The TMC5160 contains the complete intelligence which is required to drive a motor. Receiving target positions the TMC5160 manages motor movement. Based on TRINAMICs unique features stallGuard2, coolStep, dcStep, spreadCycle, and stealthChop, the TMC5160 optimizes drive performance. It trades off velocity vs. motor torque, optimizes energy efficiency, smoothness of the drive, and noiselessness. The small form factor of the TMC5160 keeps costs down and allows for miniaturized layouts. Extensive support at the chip, board, and software levels enables rapid design cycles and fast time-to-market with competitive products. High energy efficiency and reliability deliver cost savings in related systems such as power supplies and cooling. For smaller designs, the compatible, integrated TMC5130 driver provides 1.4A of motor current.

MINIATURIZED DESIGN FOR ONE STEPPER MOTOR

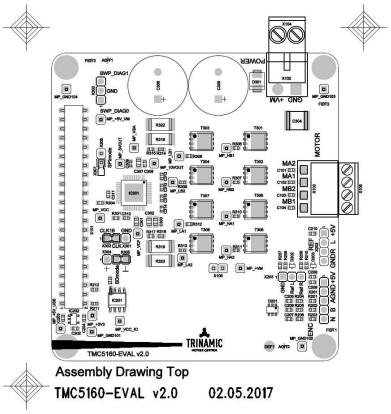


An ABN encoder interface with scaler unit and two reference switch inputs are used to ensure correct motor movement. Automatic interrupt upon deviation is available.

COMPACT DESIGN FOR MULTIPLE STEPPER MOTORS



An application with 2 stepper motors is shown. Additionally, the ABN Encoder interface and two reference switches can be used for each motor. A single CPU controls the whole system, as there are no real time tasks required to move a motor. The CPU-board and the controller / driver boards are highly economical and space saving.



The TMC5160-EVAL is part of TRINAMICs universal evaluation board system which provides a convenient handling of the hardware as well as a user-friendly software tool for evaluation. The TMC5160 evaluation board system consists of three parts: LANDUNGSBRÜCKE (base board), ESELSBRÜCKE (connector board including several test points), and TMC5160-EVAL.

ORDER CODES

Order code	Description	Size [mm²]
TMC5160-TA	stepper controller/driver for external MOSFETs; TQFP48	9 x 9
TMC5160-WA	stepper controller/driver for external MOSFETs; wett. QFN8x8	8 x 8
TMC5160-EVAL	Evaluation board for TMC5160 two phase stepper motor controller/driver	85 x 55
LANDUNGSBRÜCKE	Baseboard for TMC5160-EVAL and further evaluation boards.	85 x 55
ESELSBRÜCKE	Connector board for plug-in evaluation board system.	61 x 38

Table of Contents

1 Principles of Operation

The TMC5160 motion controller and driver chip is an intelligent power component interfacing between CPU and a high power stepper motor. All stepper motor logic is completely within the TMC5160. No software is required to control the motor – just provide target positions. The TMC5160 offers a number of unique enhancements which are enabled by the system-on-chip integration of driver and controller. The sixPoint ramp generator of the TMC5160 uses stealthChop, dcStep, coolStep, and stallGuard2 automatically to optimize every motor movement. The TMC5160 ideally extends the TMC2100, TMC2130 and TMC5130 family to higher voltages and higher motor currents.

THE TMC5160 OFFERS THREE BASIC MODES OF OPERATION:

MODE 1: Full Featured Motion Controller & Driver
All stepper motor logic is completely within the TMC5160. No software is required to control the motor – just provide target positions. Enable this mode by tying low pin SD_MODE.

MODE 2: Step & Direction Driver
An external high-performance S-ramp motion controller like the TMC4361 or a central CPU generates step & direction signals synchronized to other components like additional motors within the system. The TMC5160 takes care of intelligent current and mode control and delivers feedback on the state of the motor. The microPlyer automatically smoothens motion. Tie SD_MODE high.

MODE 3: Simple Step & Direction Driver
The TMC5160 positions the motor based on step & direction signals. The microPlyer automatically smoothens motion. No CPU interaction is required; configuration is done by hardware pins. Basic standby current control can be done by the TMC5160. Optional feedback signals allow error detection and synchronization. Enable this mode by tying pin SPI_MODE low and SD_MODE high.

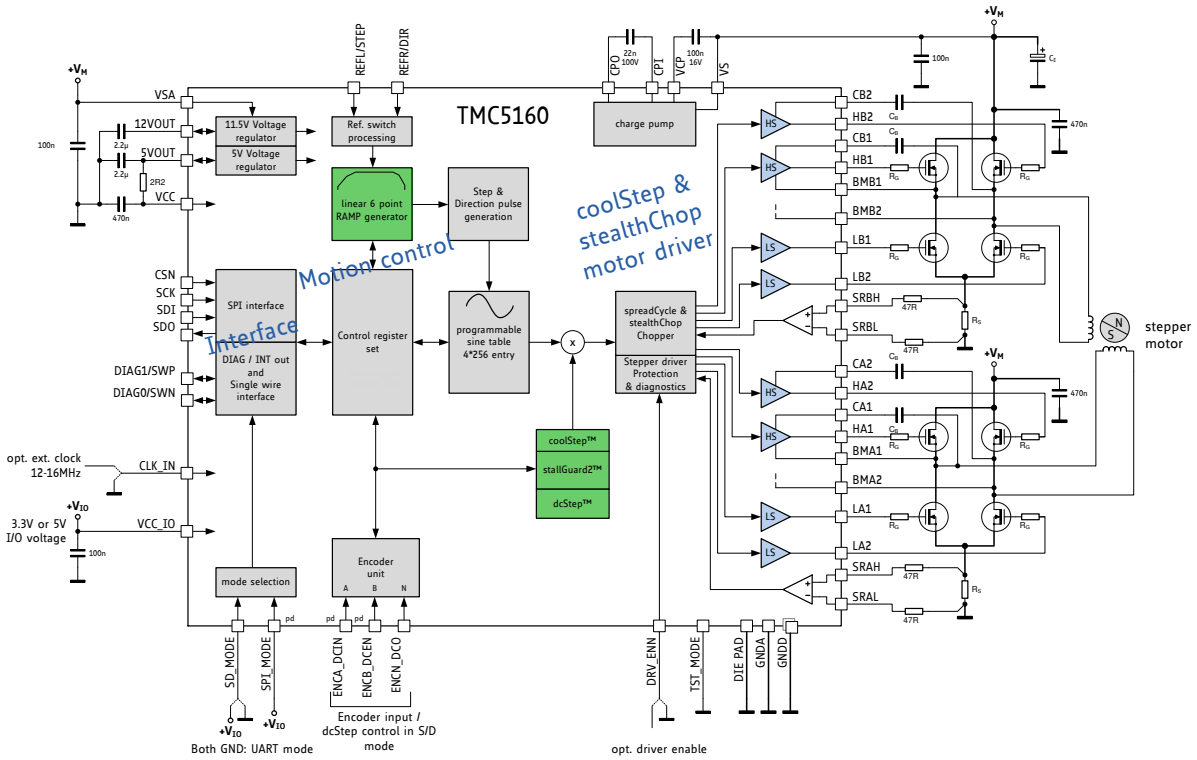


Figure 1.1 TMC5160 basic application block diagram (motion controller)

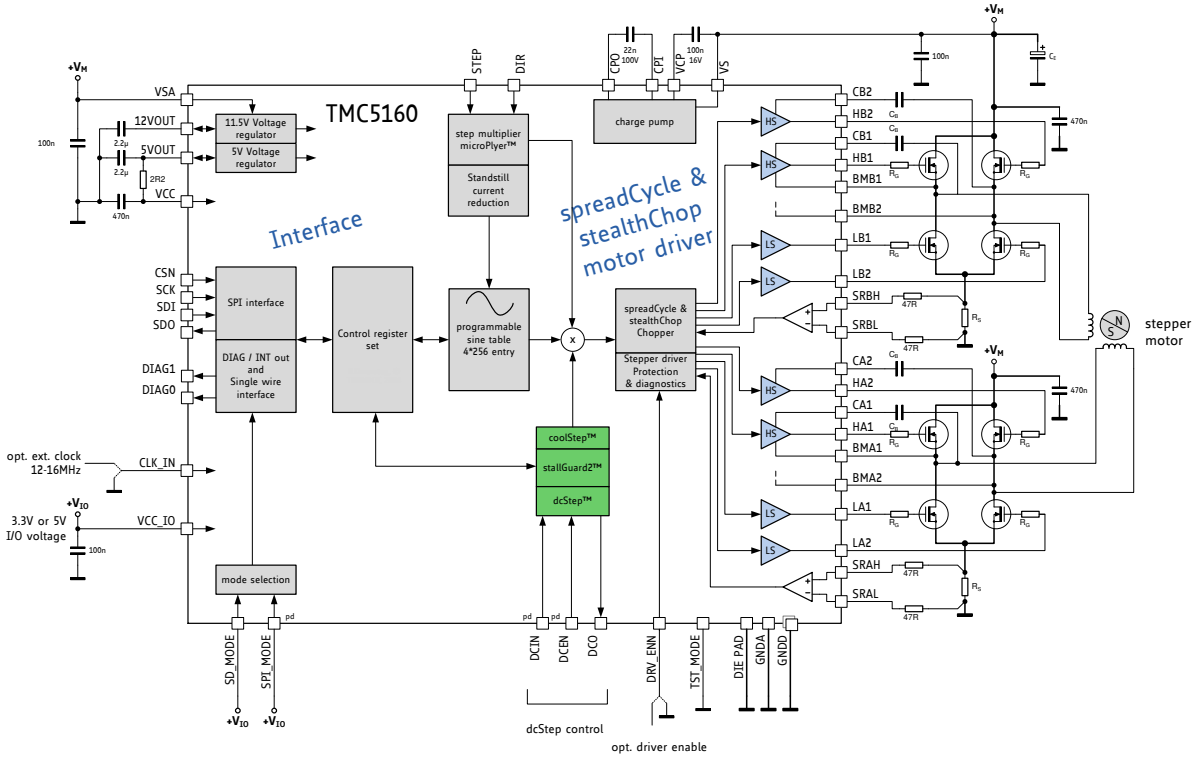


Figure 1.2 TMC5160 STEP/DIR application diagram

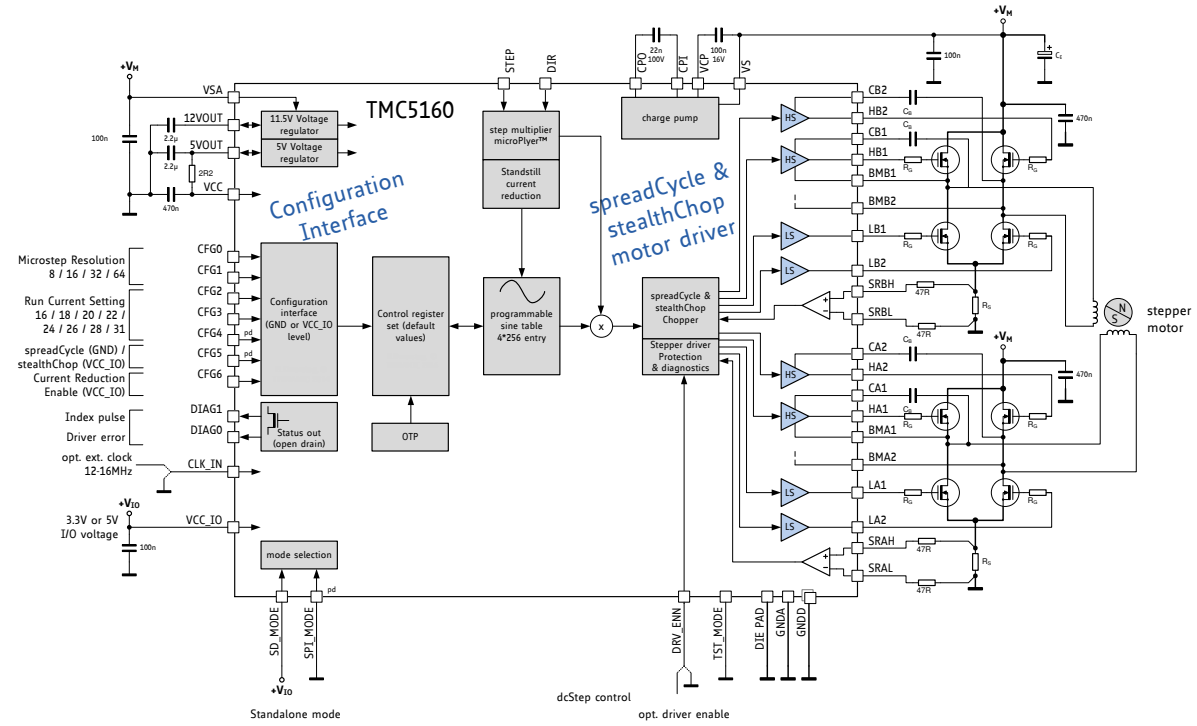


Figure 1.3 TMC5160 standalone driver application diagram

1.1 Key Concepts

The TMC5160 implements advanced features which are exclusive to TRINAMIC products. These features contribute toward greater precision, greater energy efficiency, higher reliability, smoother motion, and cooler operation in many stepper motor applications.

stealthChop2™ No-noise, high-precision chopper algorithm for inaudible motion and inaudible standstill of the motor. Allows faster motor acceleration and deceleration than stealthChop™ and extends stealthChop to low stand still motor currents.

spreadCycle™ High-precision chopper algorithm for highly dynamic motion and absolutely clean current wave. Low noise, low resonance and low vibration chopper.

dcStep™ Load dependent speed control. The motor moves as fast as possible and never loses a step.

stallGuard2™ Sensorless stall detection and mechanical load measurement.

coolStep™ Load-adaptive current control reducing energy consumption by as much as 75%.

microPlyer™ Microstep interpolator for obtaining full 256 microstep smoothness with lower resolution step inputs starting from fullstep

In addition to these performance enhancements, TRINAMIC motor drivers offer safeguards to detect and protect against shorted outputs, output open-circuit, overtemperature, and undervoltage conditions for enhancing safety and recovery from equipment malfunctions.

1.2 Control Interfaces

The TMC5160 supports both, an SPI interface and a UART based single wire interface with CRC checking. Selection of the actual interface is done via the configuration pin SW_SEL, which can be hardwired to GND or VCC_IO depending on the desired interface.

1.2.1 SPI Interface

The SPI interface is a bit-serial interface synchronous to a bus clock. For every bit sent from the bus master to the bus slave another bit is sent simultaneously from the slave to the master. Communication between an SPI master and the TMC5160 slave always consists of sending one 40-bit command word and receiving one 40-bit status word.

The SPI command rate typically is a few commands per complete motor motion.

1.2.2 UART Interface

The single wire interface allows differential operation similar to RS485 (using SWIOP and SWION) or single wire interfacing (leaving open SWION). It can be driven by any standard UART. No baud rate configuration is required.

1.3 Software

From a software point of view the TMC5160 is a peripheral with a number of control and status registers. Most of them can either be written only or read only. Some of the registers allow both read and write access. In case read-modify-write access is desired for a write only register, a shadow register can be realized in master software.

1.4 Moving and Controlling the Motor

1.4.1 Integrated Motion Controller

The integrated 32 bit motion controller automatically drives the motor to target positions, or accelerates to target velocities. All motion parameters can be changed on the fly. The motion controller recalculates immediately. A minimum set of configuration data consists of acceleration and deceleration values and the maximum motion velocity. A start and stop velocity is supported as well as a second acceleration and deceleration setting. The integrated motion controller supports immediate reaction to mechanical reference switches and to the sensorless stall detection stallGuard2.

Benefits are:

- Flexible ramp programming
- Efficient use of motor torque for acceleration and deceleration allows higher machine throughput
- Immediate reaction to stop and stall conditions

1.4.2 STEP/DIR Interface

The motor can optionally be controlled by a step and direction input. In this case, the motion controller remains unused. Active edges on the STEP input can be rising edges or both rising and falling edges as controlled by another mode bit (*dedge*). Using both edges cuts the toggle rate of the STEP signal in half, which is useful for communication over slow interfaces such as optically isolated interfaces. On each active edge, the state sampled from the DIR input determines whether to step forward or back. Each step can be a fullstep or a microstep, in which there are 2, 4, 8, 16, 32, 64, 128, or 256 microsteps per fullstep. A step impulse with a low state on DIR increases the microstep counter and a high decreases the counter by an amount controlled by the microstep resolution. An internal table translates the counter value into the sine and cosine values which control the motor current for microstepping.

1.5 Automatic Standstill Power Down

An automatic current reduction drastically reduces application power dissipation and cooling requirements. Modify stand still current, delay time and decay via register settings. Automatic freewheeling and passive motor braking are provided as an option for stand still. Passive braking reduces motor standstill power consumption to zero, while still providing effective dampening and braking! An option for faster detection of standstill is provided for both, ramp generator and STEP/DIR operation.

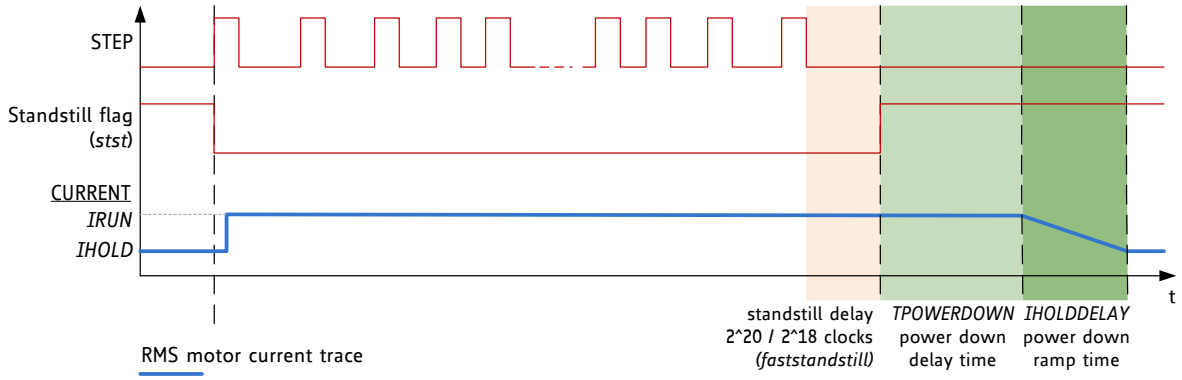


Figure 1.4 Automatic Motor Current Power Down

1.6 stealthChop2 & spreadCycle Driver

stealthChop is a voltage chopper based principle. It especially guarantees that the motor is absolutely quiet in standstill and in slow motion, except for noise generated by ball bearings. Unlike other voltage mode choppers, stealthChop2 does not require any configuration. It automatically learns the best settings during the first motion after power up and further optimizes the settings in subsequent motions. An initial homing sequence is sufficient for learning. Optionally, initial learning parameters

can be pre-configured via the interface. stealthChop2 allows high motor dynamics, by reacting at once to a change of motor velocity.

For highest dynamic applications, spreadCycle is an option to stealthChop2. It can be enabled via input pin (standalone mode) or via SPI or UART interface. stealthChop2 and spreadCycle may even be used in a combined configuration for the best of both worlds: stealthChop2 for no-noise stand still, silent and smooth performance, spreadCycle at higher velocity for high dynamics and highest peak velocity at low vibration.

spreadCycle is an advanced cycle-by-cycle chopper mode. It offers smooth operation and good resonance dampening over a wide range of speed and load. The spreadCycle chopper scheme automatically integrates and tunes fast decay cycles to guarantee smooth zero crossing performance.

Benefits of using stealthChop2:

- Significantly improved microstepping with low cost motors
- Motor runs smooth and quiet
- Absolutely no standby noise
- Reduced mechanical resonance yields improved torque

1.7 stallGuard2 – Mechanical Load Sensing

stallGuard2 provides an accurate measurement of the load on the motor. It can be used for stall detection as well as other uses at loads below those which stall the motor, such as coolStep load-adaptive current reduction. This gives more information on the drive allowing functions like sensorless homing and diagnostics of the drive mechanics.

1.8 coolStep – Load Adaptive Current Control

coolStep drives the motor at the optimum current. It uses the stallGuard2 load measurement information to adjust the motor current to the minimum amount required in the actual load situation. This saves energy and keeps the components cool.

Benefits are:

- *Energy efficiency* power consumption decreased up to 75%
- *Motor generates less heat* improved mechanical precision
- *Less or no cooling* improved reliability
- *Use of smaller motor* less torque reserve required → cheaper motor does the job

Figure 1.5 shows the efficiency gain of a 42mm stepper motor when using coolStep compared to standard operation with 50% of torque reserve. coolStep is enabled above 60RPM in the example.

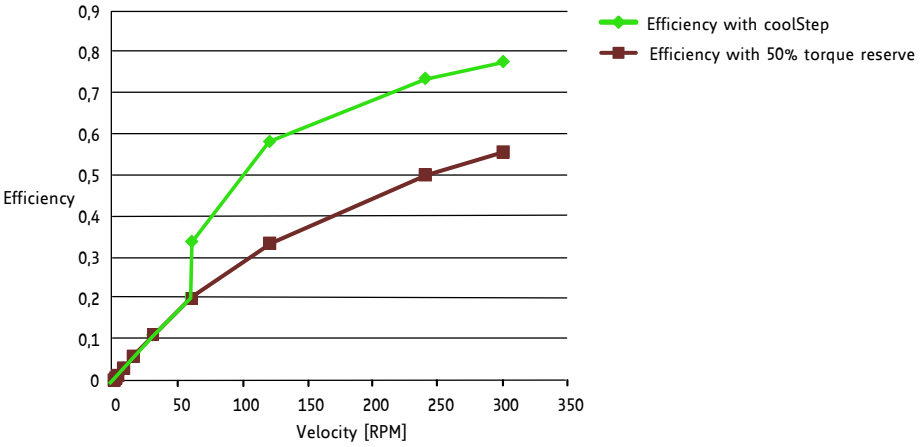


Figure 1.5 Energy efficiency with coolStep (example)

1.9 dcStep – Load Dependent Speed Control

dcStep allows the motor to run near its load limit and at its velocity limit without losing a step. If the mechanical load on the motor increases to the stalling load, the motor automatically decreases velocity so that it can still drive the load. With this feature, the motor will never stall. In addition to the increased torque at a lower velocity, dynamic inertia will allow the motor to overcome mechanical overloads by decelerating. dcStep directly integrates with the ramp generator, so that the target position will be reached, even if the motor velocity needs to be decreased due to increased mechanical load. A dynamic range of up to factor 10 or more can be covered by dcStep without any step loss. By optimizing the motion velocity in high load situations, this feature further enhances overall system efficiency.

Benefits are:

- Motor does not loose steps in overload conditions
- Application works as fast as possible
- Highest possible acceleration automatically
- Highest energy efficiency at speed limit
- Highest possible motor torque using fullstep drive
- Cheaper motor does the job

1.10 Encoder Interface

The TMC5160 provides an encoder interface for external incremental encoders. The encoder provides automatic checking for step loss and can be used for homing of the motion controller (alternatively to reference switches). A programmable prescaler allows the adaptation of the encoder resolution to the motor resolution. A 32 bit encoder counter is provided.

2 Pin Assignments

2.1 Package Outline

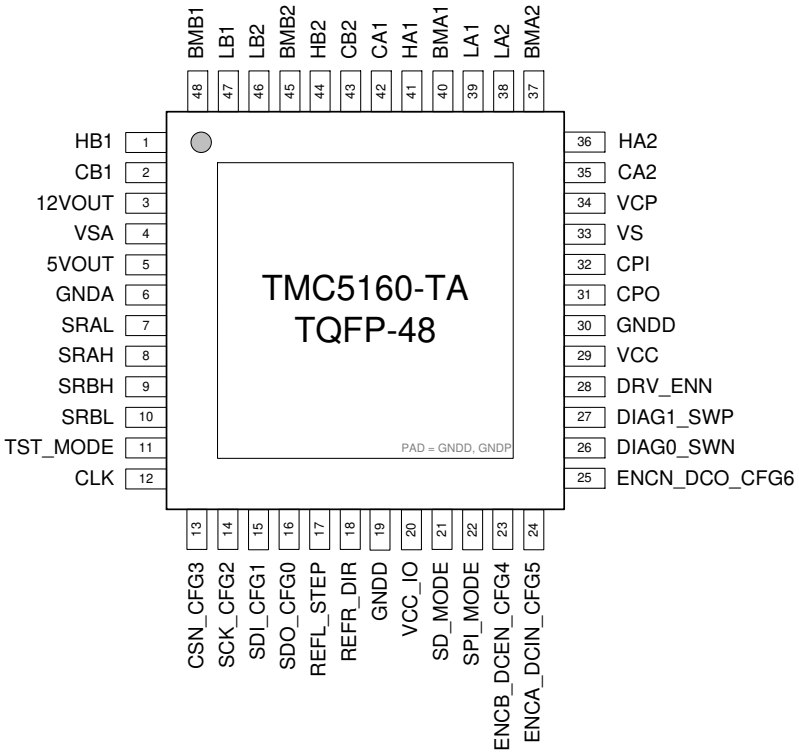


Figure 2.1 TMC5160-TA package and pinning TQFP-EP 48 (7x7mm² body, 9x9mm² with leads)

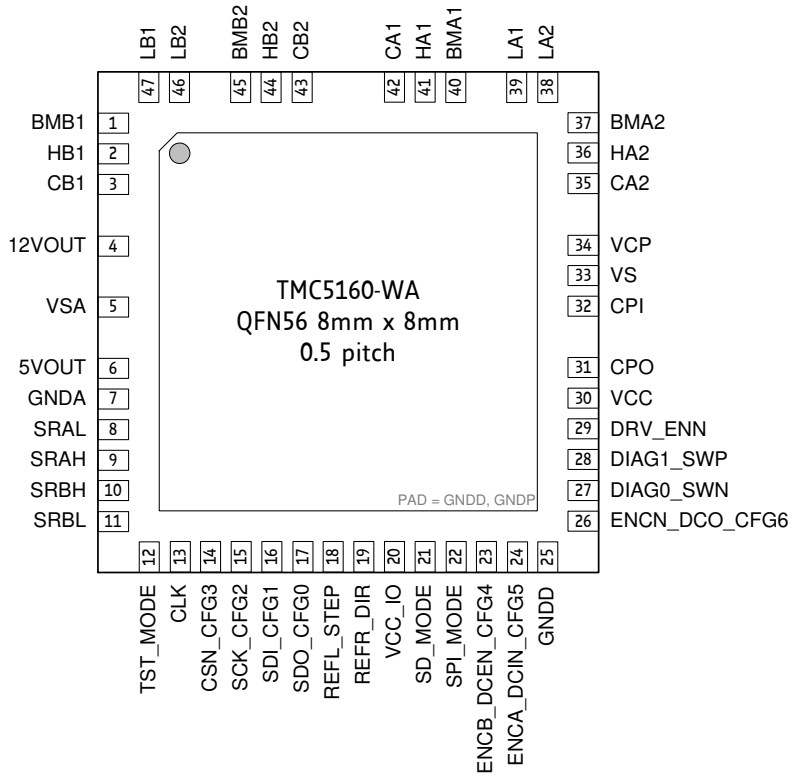


Figure 2.2 TMC5160-WA package and pinning QFN-WA (8x8mm²)

2.2 Signal Descriptions

Pin	TQFP	QFN	Type	Function
HB1	1	2		High side gate driver output.
CB1	2	3		Bootstrap capacitor positive connection.
12VOUT	3	4		Output of internal 11.5V gate voltage regulator and supply pin of low side gate drivers. Attach 2.2µF to 10µF ceramic capacitor to GND plane near to pin for best performance. Use at least 10 times more capacity than for bootstrap capacitors. In case an external gate voltage supply is available, tie VSA and 12VOUT to the external supply.
VSA	4	5		Analog supply voltage for 11.5V and 5V regulator. Normally tied to VS. Provide a 100nF filtering capacitor.
5VOUT	5	6		Output of internal 5V regulator. Attach 2.2µF to 10µF ceramic capacitor to GNDA near to pin for best performance. Output for VCC supply of the chip.
GNDA	6	7		Analog GND. Connect to GND plane near pin.
SRAL	7	8	AI	Sense resistor GND connection for phase A. Connect to the GND side of the sense resistor in order to compensate for voltage drop on the GND interconnection.
SRAH	8	9	AI	Sense resistor for phase A. Connect to the upper side of the sense resistor. A Kelvin connection is preferred with high motor currents. Symmetrical RC-Filtering may be added for SRAL and SRAH to eliminate high frequency switching spikes from other drives or switching of coil B.
SRBH	9	10	AI	Sense resistor for phase B. Connect to the upper side of the sense resistor. A Kelvin connection is preferred with high motor currents. Symmetrical RC-Filtering may be added for SRBL and SRBH to eliminate high frequency switching spikes from other drives or switching of coil A.
SRBL	10	11	AI	Sense resistor GND connection for phase B. Connect to the GND side of the sense resistor in order to compensate for voltage drop on the GND interconnection.
TST_MODE	11	12	DI	Test mode input. Tie to GND using short wire.
CLK	12	13	DI	CLK input. Tie to GND using short wire for internal clock or supply external clock. Internal clock-fail over circuit protects against loss of external clock signal.
CSN_CFG3	13	14	DI	SPI chip select input (negative active) (SPI_MODE=1) or Configuration input (SPI_MODE=0)
SCK_CFG2	14	15	DI	SPI serial clock input (SPI_MODE=1) or Configuration input (SPI_MODE=0)
SDI_CFG1	15	16	DI	SPI data input (SPI_MODE=1) or Configuration input (SPI_MODE=0) or Next address input for single wire interface.
SDO_CFG0	16	17	DIO	SPI data output (tristate) (SPI_MODE=1) or Configuration input (SPI_MODE=0) or Next address output for single wire interface.
REFL_STEP	17	18	DI	Left reference input (for internal ramp generator) or STEP input when (SD_MODE=1).
REFR_DIR	18	19	DI	Right reference input (for internal ramp generator) or DIR input (SD_MODE=1).
GNDD	19, 30	25, Pad		Digital GND. Connect to GND plane near pin.
VCC_IO	20	20		3.3V to 5V IO supply voltage for all digital pins.

Pin	TQFP	QFN	Type	Function
SD_MODE	21	21	DI	Mode selection input. When tied low, the internal ramp generator generates step pulses. When tied high, the STEP/DIR inputs control the driver. SD_MODE=0 and SPI_MODE=0 enable UART operation.
SPI_MODE	22	22	DI (pd)	Mode selection input. When tied low with SD_MODE=1, the chip is in standalone mode and pins have their CFG functions. When tied high, the SPI interface is enabled. Integrated pull down resistor.
ENCB_DCEN_ CFG4	23	23	DI (pd)	Encoder B-channel input (when using internal ramp generator) or dcStep enable input (SD_MODE=1, SPI_MODE=1) – leave open or tie to GND for normal operation in this mode (no dcStep). Configuration input (SPI_MODE=0)
ENCA_DCIN_ CFG5	24	24	DI (pd)	Encoder A-channel input (when using internal ramp generator) or dcStep gating input for axis synchronization (SD_MODE=1, SPI_MODE=1) or Configuration input (SPI_MODE=0)
ENCN_DCO_ CFG6	25	26	DIO	Encoder N-channel input (SD_MODE=0) or dcStep ready output (SD_MODE=1). With SD_MODE=0, pull to GND or VCC_IO, if the pin is not used.
DIAG0_SWN	26	27	DIO (pu+ pd)	Diagnostics output DIAG0. Interrupt or STEP output for motion controller (SD_MODE=0, SPI_MODE=1). Use external pullup resistor with 47k or less in open drain mode. Single wire I/O (negative) (only with SD_MODE=0 and SPI_MODE=0)
DIAG1_SWP	27	28	DIO (pd)	Diagnostics output DIAG1. Position compare or DIR output for motion controller (SD_MODE=0, SPI_MODE=1). Use external pullup resistor with 47k or less in open drain mode. Single wire I/O (positive) (only with SD_MODE=0 and SPI_MODE=0)
DRV_ENN	28	29	DI	Enable input. The power stage becomes switched off (all motor outputs floating) when this pin becomes driven to a high level.
VCC	29	30		5V supply input for digital circuitry within chip and charge pump. Provide 100nF or bigger capacitor to GND (GND plane) near pin. Shall be supplied by 5VOUT. A 2.2 or 3.3 Ohm resistor is recommended for decoupling noise from 5VOUT. When using an external supply, make sure, that VCC comes up before or in parallel to 5VOUT or VCC_IO, whichever comes up later!
CPO	31	31		Charge pump capacitor output.
CPI	32	32		Charge pump capacitor input. Tie to CPO using 22nF 100V capacitor.
VS	33	33		Motor supply voltage. Provide filtering capacity near pin with short loop to GND plane. Must be tied to the positive bridge supply voltage.
VCP	34	34		Charge pump voltage. Tie to VS using 100nF capacitor.
CA2	35	35		Bootstrap capacitor positive connection.

Pin	TQFP	QFN	Type	Function
HA2	36	36		High side gate driver output.
BMA2	37	37		Bridge Center and bootstrap capacitor negative connection.
LA2	38	38		Low side gate driver output.
LA1	39	39		Low side gate driver output.
BMA1	40	40		Bridge Center and bootstrap capacitor negative connection.
HA1	41	41		High side gate driver output.
CA1	42	42		Bootstrap capacitor positive connection.
CB2	43	43		Bootstrap capacitor positive connection.
HB2	44	44		High side gate driver output.
BMB2	45	45		Bridge Center and bootstrap capacitor negative connection.
LB2	46	46		Low side gate driver output.
LB1	47	47		Low side gate driver output.
BMB1	48	1		Bridge Center and bootstrap capacitor negative connection.
Exposed die pad	-	-		Connect the exposed die pad to a GND plane. Provide as many as possible vias for heat transfer to GND plane. Serves as GND pin for the low side gate drivers. Ensure low loop inductivity to sense resistor GND.

*(pd) denominates a pin with pulldown resistor
* All digital pins DI, DIO and DO use VCC_IO level and contain protection diodes to GND and VCC_IO
* All digital inputs DI and DIO have internal Schmitt-Triggers

3 Sample Circuits

The following sample circuits show the required external components in different operation and supply modes. The connection of the bus interface and further digital signals are left out for clarity.

3.1 Standard Application Circuit

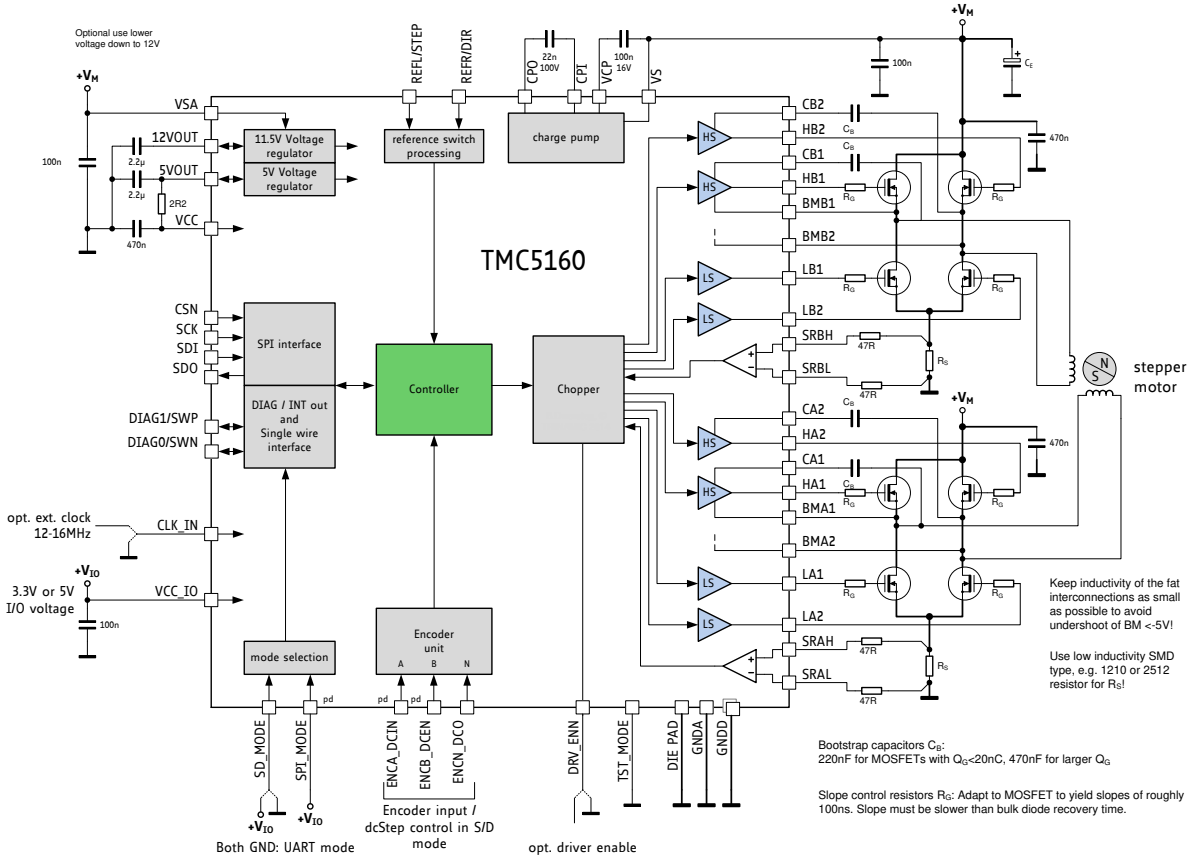


Figure 3.1 Standard application circuit

The standard application circuit uses a minimum set of additional components. Eight MOSFETs are selected for the desired current, voltage and package type. Two sense resistors set the motor coil current. See chapter 9 to choose the right value for sense resistors. Use low ESR capacitors for filtering the power supply. A minimum capacity of 100µF per ampere of coil current near to the power bridge is recommended for best performance. The capacitors need to cope with the current ripple caused by chopper operation. Current ripple in the supply capacitors also depends on the power supply internal resistance and cable length. VCC_IO can be supplied from 5VOUT, or from an external source, e.g. a 3.3V regulator. In order to minimize linear voltage regulator power dissipation of the internal 5V and 11.5V voltage regulators in applications where VM is high, a different (lower) supply voltage should be used for VSA (see chapter 3.2).

Basic layout hints
Place sense resistors and all filter capacitors as close as possible to the power MOSFETs. Place the TMC5160 near to the MOSFETs and use short interconnection lines in order to minimize parasitic trace inductance. Use a solid common GND for all GND, GNDA and GNDD connections, also for sense resistor GND. Connect 5VOUT filtering capacitor directly to 5VOUT and GNDA pin. See layout hints for more details. Low ESR electrolytic capacitors are recommended for VS filtering.

Attention
In case VSA is supplied by a different voltage source, make sure that VSA does not drop out during motor operation.

3.2 External Gate Voltage Regulator

At high supply voltages like 48V, the internal gate voltage regulator and the internal 5V regulator have considerable power dissipation, especially with high MOSFET gate charges, high chopper frequency or high system clock frequency >12MHz. A good thermal coupling of the heat slug to the system PCB GND plane is required to dissipate heat. Still, the thermal thresholds will be lowered significantly by self-heating. To reduce power dissipation, supply an external gate driver voltage to the TMC5160. Figure 3.2 shows the required connection. The internal gate voltage regulator becomes disabled in this constellation. 12V +/-1V are recommended for best results.

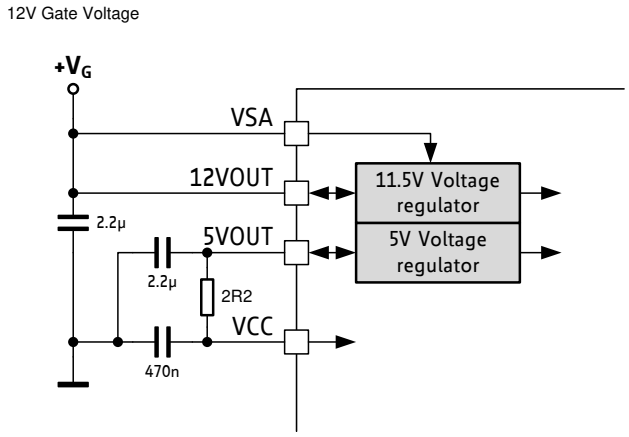


Figure 3.2 External gate voltage supply

Hint
With MOSFETs above 50nC of total gate charge, chopper frequency >40kHz, or at clock frequency >12MHz, it is recommended to use a VSA supply not higher than 40V.

3.3 MOSFETs and Slope Control

The selection of power MOSFETs depends on a number of factors, like package size, on-resistance, voltage rating and supplier. It is not true, that larger, lower RDSon MOSFETs will always be better, as a larger device also has higher capacitances and may add more ringing in trace inductance and power dissipation in the gate drive circuitry. Adapt the MOSFETs to the required motor voltage (adding 5-10V of reserve to the peak supply voltage) and to the desired maximum current, in a way that resistive power dissipation still is low for the chosen MOSFET package. The TMC5160 drives the MOSFET gates with roughly 10V, so normal, 10V specified types are sufficient. Logic level FETs (4.5V specified RDSon) will also work, but may be more critical with regard to bridge cross-conduction due to lower $V_{GS(th)}$.

The gate drive current and MOSFET gate resistors R_G (optional) should basically be adapted to the MOSFET gate-drain charge (Miller charge). Figure 3.3 shows the influence of the Miller charge on the switching event. Figure 3.4 additionally shows the switching events in different load situations (load pulling the output up or down), and the required bridge brake-before-make time.

The following table shall serve as a thumb rule for programming the MOSFET driver current (*DRVSTRENGTH* setting) and the selection of gate resistors:

MOSFET MILLER CHARGE VS. *DRVSTRENGTH* AND R_G		
Miller Charge [nC] (typ.)	*DRVSTRENGTH* setting	**Value of R_G [Ω]**
<10	0	≤ 15
10…20	0 or 1	≤ 10
20…40	1 or 2	≤ 7.5
40…60	2 or 3	≤ 5
>60	3	≤ 2.7

The TMC5160 provides increased gate-off drive current to avoid bridge cross-conduction induced by high dV/dt. This protection will be less efficient with gate resistors exceeding the values given in the table. Therefore, for larger values of R_G, a parallel diode may be required to ensure keeping the MOSFET safely off during switching events.

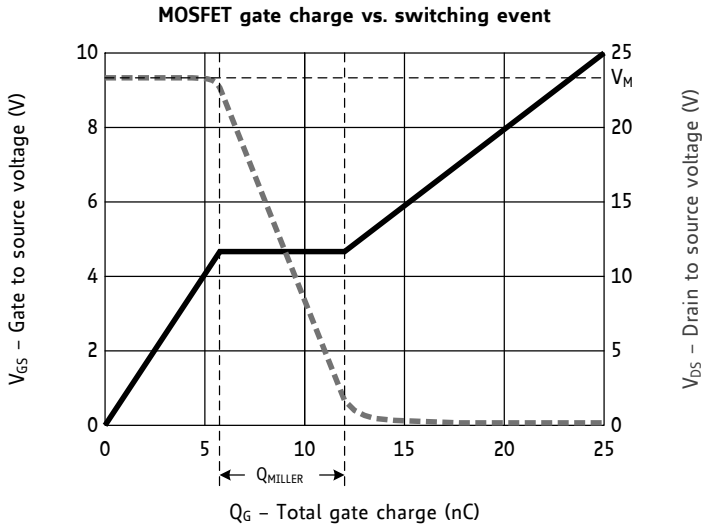


Figure 3.3 Miller charge determines switching slope

Hints
- Choose modern MOSFETs with fast and soft recovery bulk diode and low reverse recovery charge.
- A small, SMD MOSFET package allows compacter routing and reduces parasitic inductance effects.

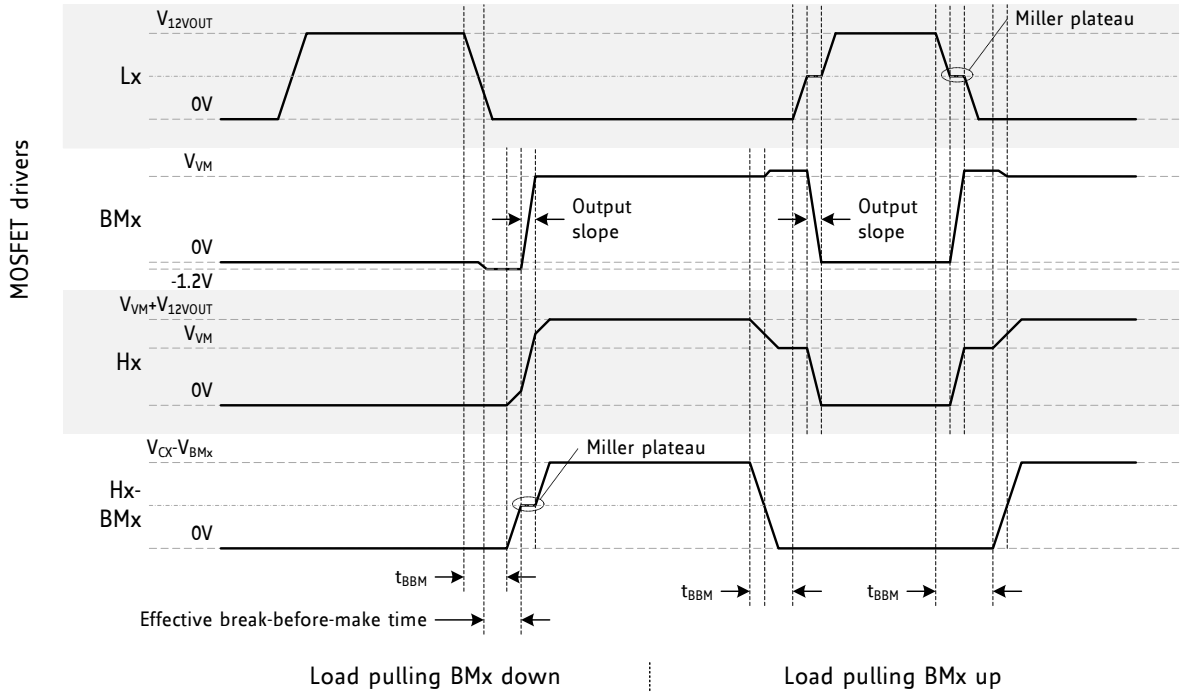


Figure 3.4 Slopes, Miller plateau and blank time

The following *DRV_CONF* parameters allow adapting the driver to the MOSFET bridge:

Parameter	Description	Setting	Comment
BBMTIME	Break-before-make time setting to ensure non-overlapping switching of high-side and low-side MOSFETs. *BBMTIME* allows fine tuning of times in increments shorter than a clock period. For higher times, use BBMCLKS.	0…24	time[ns]≈ 100ns*32/(32-*BBMTIME*) *Ensure ~30% headroom* *Reset Default: 0*
BBMCLKS	Like *BBMTIME*, but in multiple of a clock cycle. The longer setting rules (*BBMTIME* vs. *BBMCLKS*).	0…15	0: off *Reset Default: OTP 4 or 2*
DRV_ STRENGTH	Selection of gate driver current. Adapts the gate driver current to the gate charge of the external MOSFETs.	0…3	*Reset Default = 2*
FILT_ISENSE	Filter time constant of sense amplifier to suppress ringing and coupling from second coil operation *Hint:* Increase setting if motor chopper noise occurs due to cross-coupling of both coils. *(Reset Default = %00)*	0…3	00: ~100ns (*reset default*) 01: ~200ns 10: ~300ns 11: ~400ns

DRV_CONF Parameters
Use the lowest gate driver strength setting *DRVSTRENGTH* giving favorable switching slopes, before increasing the value of the gate series resistors. A slope time of nominal 40ns to 80ns is absolutely sufficient and will normally be covered by the shortest possible Break-Before-Make time setting (*BBMTIME*=0, *BBMCLKS*=0).
In case slower slopes have to be used, e.g. with large MOSFETs, ensure that the break-before-make time (*BBMTIME*, optionally use *BBMCLKS* for times >200ns) sufficiently covers the switching event, in order to avoid bridge cross conduction. The shortest break-before-make time, safely covering the switching event, gives best results. Add roughly 30% of reserve, to cover production stray of MOSFETs and driver.

3.4 Tuning the MOSFET Bridge

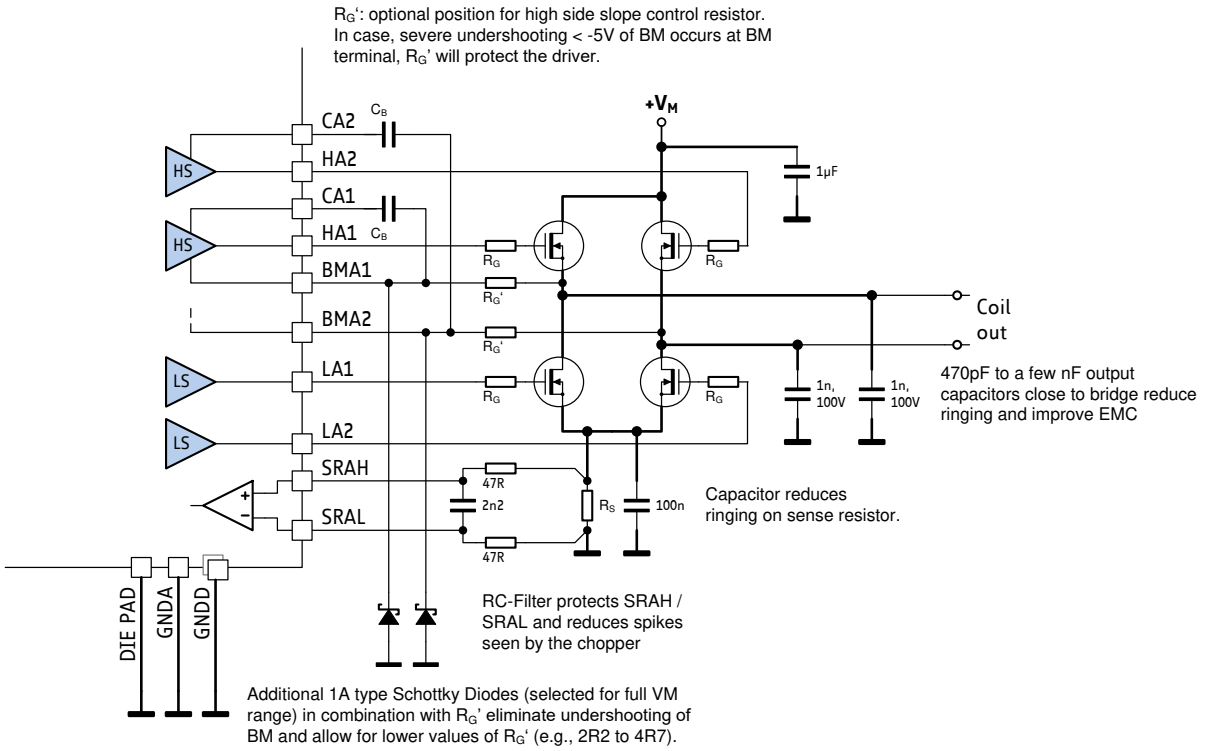
A clean switching event is favorable to ensure low power dissipation and good EMC behavior. Unsuitable layout or components endanger stable operation of the circuit. Therefore, it is important to understand the effect of parasitic trace inductivity and MOSFET reverse recovery.

Stray inductance in power routing will cause ringing whenever the opposite MOSFET is in diode conduction prior to switching on a low-side MOSFET or high-side MOSFET. Diode conduction occurs during break-before make time when the load current is inverse to the prior bridge polarity, i.e. following a fast decay cycle. The MOSFET bulk diode has a certain, type specific reverse recovery time and charge. This time typically is in the range of a few 10ns. During reverse recovery time, the bulk diode will cause high current flow across the bridge. This current is taken from the power supply filter capacitors (see thick lines Figure 3.5). Once the diode opens parasitic inductance tries to keep the current flowing. A high, fast slope results and leads to ringing in all parasitic inductivities (see Figure 3.6). This may lead to bridge voltage undershooting the GND level. It must be ensured, that the driver IC does not see spikes on its BM pins to GND going below -5V. Measure the voltage directly at the driver pins to driver GND. The amount of undershooting depends on energy stored in parasitic inductivities from low side drain to low side source and via the sense resistor RS to GND.

To improve behavior

- Tune MOSFET switching slopes (measure without inductive load) to be slower than the MOSFET bulk diode reverse recovery time. This will reduce cross conduction.
- Add optional resistors and capacitors to ensure clean switching by minimizing ringing and reliable operation. Figure 3.5 shows different options.
- Some MOSFETs eliminate this problem by integrating a Schottky diode from source to drain.

Figure 3.7 shows performance of the basic circuit after adapting switching slope and adding 1nF bridge output capacitors.



Decide use and value of the additional components based on measurements of the actual circuit using the final layout!

Figure 3.5 Bridge protection options for power routing inductivity

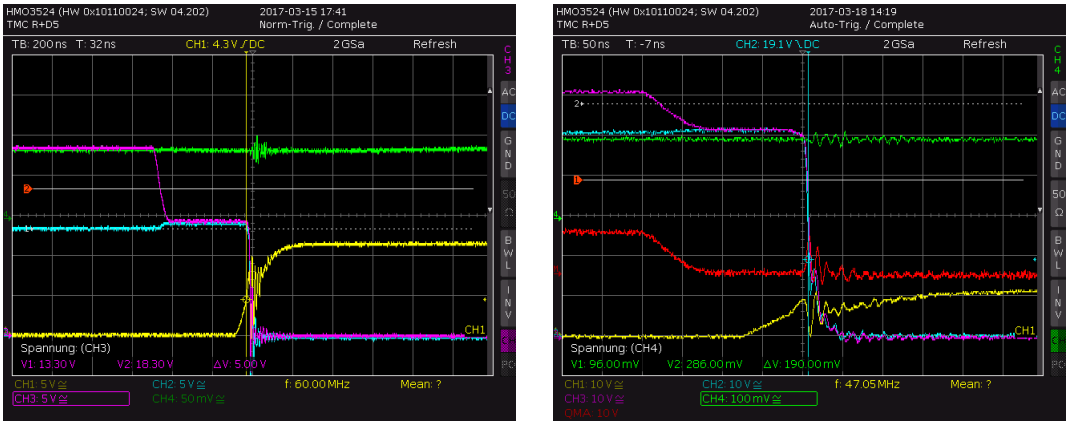


Figure 3.6 Ringing of output (blue) and Gate voltages (Yellow, Cyan) with untuned brige

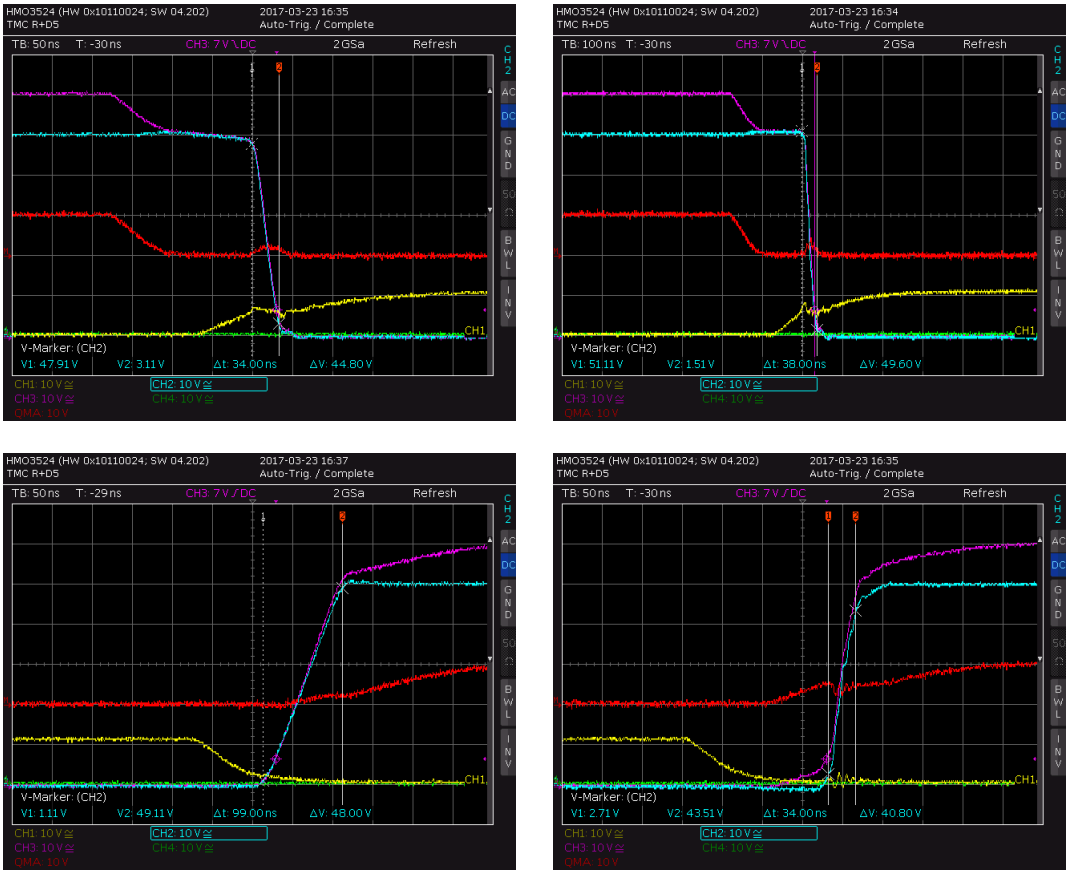


Figure 3.7 Switching event with optimized components (without / after bulk diode conduction)

BRIDGE OPTIMIZATION EXAMPLE

A stepper driver for 6A of motor current has been designed using the MOSFET AOD4126 in the standard schematic.
The MOSFETs have a low gate capacitance and offer roughly 50ns slope time at the lowest driver strength setting. At lowest driver strength setting, switching quality is best (Figure 3.6), but still shows a lot of ringing. Low side gate resistors have been added to slightly increase switching slope time following high-side bulk diode conduction by increasing the effect of Gate-Drain (Miller) charge. High side gate resistors have been added for symmetry. Tests showed, that 1nF output capacitors dramatically reduce ringing of the power bridge following bulk diode conduction (Figure 3.7). Figure 3.8 shows the actual components and values after optimization.

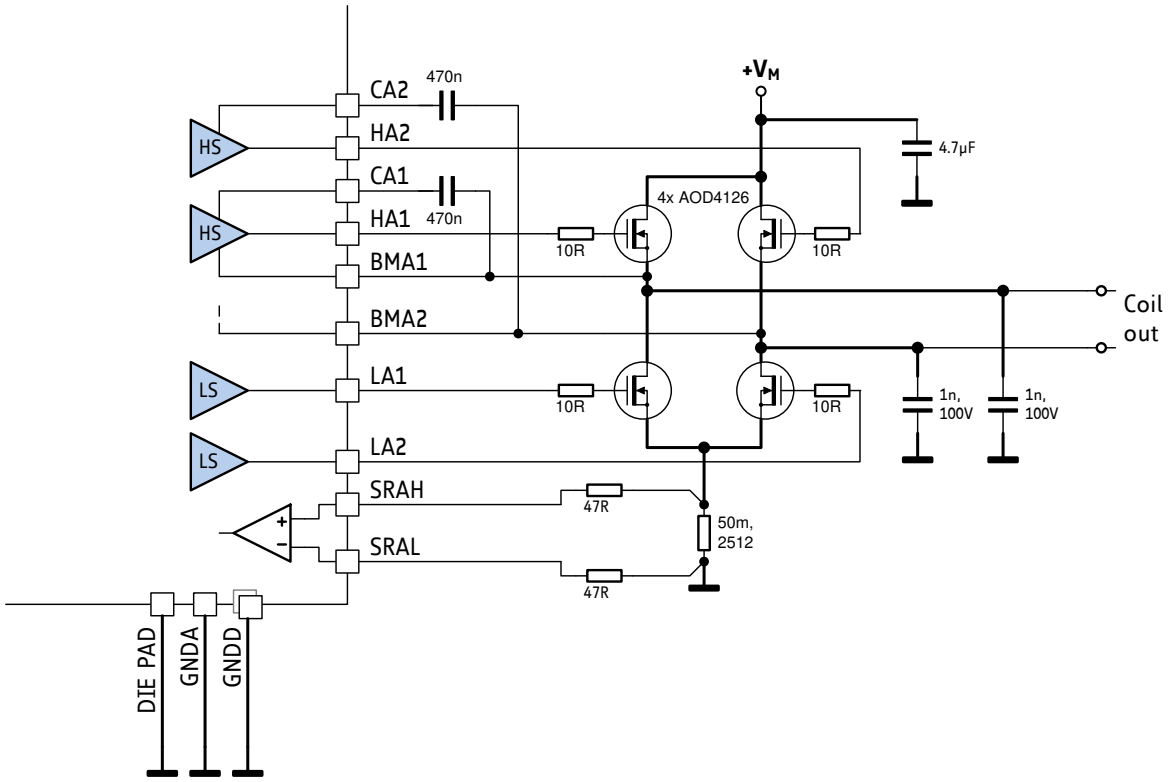


Figure 3.8 Example for bridge with tuned components (see scope shots)

Hints

- Tune the bridge layout for minimum loop inductivity. A compact layout is best.
- Keep MOSFET gate connections short and straight and avoid loop inductivity between BM and corresponding HS driver pin. Loop inductance is minimized with parallel traces, or adjacent traces on adjacent layers. A wider trace reduces inductivity (don't use minimum trace width).
- Minimize the length of the sense resistor to low side MOSFET source, and place the TMC5160 near the sense resistor's GND connection, with its GND connections directly connected to the same GND plane.
- Optimize switching behavior by tuning gate current setting and gate resistors. Add MOSFET bridge output capacitors (470pF to a few nF) to reduce ringing.
- Measure the performance of the bridge by probing BM pins directly at the bridge or at the TMC5160 using a short GND tip on the scope probe rather than a GND cable, if available.

4 SPI Interface

4.1 SPI Datagram Structure

The TMC5160 uses 40 bit SPI™ (Serial Peripheral Interface, SPI is Trademark of Motorola) datagrams for communication with a microcontroller. Microcontrollers which are equipped with hardware SPI are typically able to communicate using integer multiples of 8 bit. The NCS line of the device must be handled in a way, that it stays active (low) for the complete duration of the datagram transmission.

Each datagram sent to the device is composed of an address byte followed by four data bytes. This allows direct 32 bit data word communication with the register set. Each register is accessed via 32 data bits even if it uses less than 32 data bits.

For simplification, each register is specified by a one byte address:

- For a read access the most significant bit of the address byte is 0.
- For a write access the most significant bit of the address byte is 1.

Most registers are write only registers, some can be read additionally, and there are also some read only registers.

SPI DATAGRAM STRUCTURE

MSB (transmitted first)	40 bit	LSB (transmitted last)
39 …		… 0
→ 8 bit address **← 8 bit SPI status**	**← → 32 bit data**	
39 … 32	31 … 0	
→ to TMC5160 RW + 7 bit address ← from TMC5160 8 bit SPI status	8 bit data	8 bit data	8 bit data	8 bit data
39 / 38 … 32	31 … 24	23 … 16	15 … 8	7 … 0

w	38…32							31…28				27…24				23…20				19…16				15…12				11…8				7…4				3…0			
39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

4.1.1 Selection of Write / Read (WRITE_notREAD)

The read and write selection is controlled by the MSB of the address byte (bit 39 of the SPI datagram). This bit is 0 for read access and 1 for write access. So, the bit named W is a WRITE_notREAD control bit. The active high write bit is the MSB of the address byte. So, 0x80 has to be added to the address for a write access. The SPI interface always delivers data back to the master, independent of the W bit. The data transferred back is the data read from the address which was transmitted with the *previous* datagram, if the previous access was a read access. If the previous access was a write access, then the data read back mirrors the previously received write data. So, the difference between a read and a write access is that the read access does not transfer data to the addressed register but it transfers the address only and its 32 data bits are dummies, and, further the following read or write access delivers back the data read from the address transmitted in the preceding read cycle.

A read access request datagram uses dummy write data. Read data is transferred back to the master with the subsequent read or write access. Hence, reading multiple registers can be done in a pipelined fashion.

Whenever data is read from or written to the TMC5160, the MSBs delivered back contain the SPI status, *SPI_STATUS*, a number of eight selected status bits.

Example:

For a read access to the register (*XACTUAL*) with the address 0x21, the address byte has to be set to 0x21 in the access preceding the read access. For a write access to the register (*VACTUAL*), the address byte has to be set to 0x80 + 0x22 = 0xA2. For read access, the data bit might have any value (-). So, one can set them to 0.

action	data sent to TMC5160	data received from TMC5160
read *XACTUAL*	→ 0x2100000000	← 0xSS & unused data
read *XACTUAL*	→ 0x2100000000	← 0xSS & XACTUAL
write *VMAX*:= 0x00ABCDEF	→ 0xA700ABCDEF	← 0xSS & XACTUAL
write *VMAX*:= 0x00123456	→ 0xA700123456	← 0xSS00ABCDEF

*)S: is a placeholder for the status bits *SPI_STATUS*

4.1.2 SPI Status Bits Transferred with Each Datagram Read Back

New status information becomes latched at the end of each access and is available with the next SPI transfer.

***SPI_STATUS* – status flags transmitted with each SPI access in bits 39 to 32**		
Bit	**Name**	**Comment**
7	*status_stop_r*	*RAMP_STAT*[1] – 1: Signals stop right switch status (motion controller only)
6	*status_stop_l*	*RAMP_STAT*[0] – 1: Signals stop left switch status (motion controller only)
5	*position_reached*	*RAMP_STAT*[9] – 1: Signals target position reached (motion controller only)
4	*velocity_reached*	*RAMP_STAT*[8] – 1: Signals target velocity reached (motion controller only)
3	*standstill*	*DRV_STATUS*[31] – 1: Signals motor stand still
2	*sg2*	*DRV_STATUS*[24] – 1: Signals stallGuard flag active
1	*driver_error*	*GSTAT*[1] – 1: Signals driver 1 driver error (clear by reading *GSTAT*)
0	*reset_flag*	*GSTAT*[0] – 1: Signals, that a reset has occurred (clear by reading *GSTAT*)

4.1.3 Data Alignment

All data are right aligned. Some registers represent unsigned (positive) values, some represent integer values (signed) as two's complement numbers, single bits or groups of bits are represented as single bits respectively as integer groups.

4.2 SPI Signals

The SPI bus on the TMC5160 has four signals:

- SCK – bus clock input
- SDI – serial data input
- SDO – serial data output
- CSN – chip select input (active low)

The slave is enabled for an SPI transaction by a low on the chip select input CSN. Bit transfer is synchronous to the bus clock SCK, with the slave latching the data from SDI on the rising edge of SCK and driving data to SDO following the falling edge. The most significant bit is sent first. A minimum of 40 SCK clock cycles is required for a bus transaction with the TMC5160.

If more than 40 clocks are driven, the additional bits shifted into SDI are shifted out on SDO after a 40-clock delay through an internal shift register. This can be used for daisy chaining multiple chips.

CSN must be low during the whole bus transaction. When CSN goes high, the contents of the internal shift register are latched into the internal control register and recognized as a command from the master to the slave. If more than 40 bits are sent, only the last 40 bits received before the rising edge of CSN are recognized as the command.

4.3 Timing

The SPI interface is synchronized to the internal system clock, which limits the SPI bus clock SCK to half of the system clock frequency. If the system clock is based on the on-chip oscillator, an additional 10% safety margin must be used to ensure reliable data transmission. All SPI inputs as well as the ENN input are internally filtered to avoid triggering on pulses shorter than 20ns. Figure 4.1 shows the timing parameters of an SPI bus transaction, and the table below specifies their values.

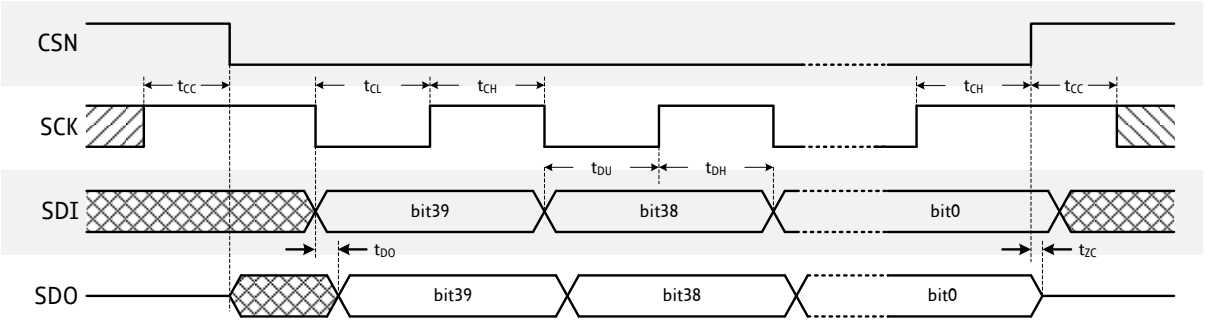


Figure 4.1 SPI timing

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

SPI interface timing	**AC-Characteristics** clock period: 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

5 UART Single Wire Interface

The UART single wire interface allows the control of the TMC5160 with any microcontroller UART. It shares transmit and receive line like an RS485 based interface. Data transmission is secured using a cyclic redundancy check, so that increased interface distances (e.g. over cables between two PCBs) can be bridged without the danger of wrong or missed commands even in the event of electro-magnetic disturbance. The automatic baud rate detection and an advanced addressing scheme make this interface easy and flexible to use.

5.1 Datagram Structure

5.1.1 Write Access

UART WRITE ACCESS DATAGRAM STRUCTURE																			
each byte is LSB...MSB, highest byte transmitted first																			
0 … 63																			
sync + reserved								**8 bit slave address**			**RW + 7 bit register addr.**			**32 bit data**			**CRC**		
0…7								8…15			16…23			24…55			56…63		
1	0	1	0	Reserved (don't cares but included in CRC)				*SLAVEADDR*			register address		1	data bytes 3, 2, 1, 0 (high to low byte)			CRC		
0	1	2	3	4	5	6	7	8	…	15	16	…	23	24	…	55	56	…	63

A sync nibble precedes each transmission to and from the TMC5160 and is embedded into the first transmitted byte, followed by an addressing byte. Each transmission allows a synchronization of the internal baud rate divider to the master clock. The actual baud rate is adapted and variations of the internal clock frequency are compensated. Thus, the baud rate can be freely chosen within the valid range. Each transmitted byte starts with a start bit (logic 0, low level on SWIOP) and ends with a stop bit (logic 1, high level on SWIOP). The bit time is calculated by measuring the time from the beginning of start bit (1 to 0 transition) to the end of the sync frame (1 to 0 transition from bit 2 to bit 3). All data is transmitted byte wise. The 32 bit data words are transmitted with the highest byte first.

A minimum baud rate of 9000 baud is permissible, assuming 20 MHz clock (worst case for low baud rate). Maximum baud rate is f_{CLK}/16 due to the required stability of the baud clock.

The slave address is determined by the register *SLAVEADDR*. If the external address pin NEXTADDR is set, the slave address becomes incremented by one.

The communication becomes reset if a pause time of longer than 63 bit times between the start bits of two successive bytes occurs. This timing is based on the last correctly received datagram. In this case, the transmission needs to be restarted after a failure recovery time of minimum 12 bit times of bus idle time. This scheme allows the master to reset communication in case of transmission errors. Any pulse on an idle data line below 16 clock cycles will be treated as a glitch and leads to a timeout of 12 bit times, for which the data line must be idle. Other errors like wrong CRC are also treated the same way. This allows a safe re-synchronization of the transmission after any error conditions. Remark, that due to this mechanism an abrupt reduction of the baud rate to less than 15 percent of the previous value is not possible.

Each accepted write datagram becomes acknowledged by the receiver by incrementing an internal cyclic datagram counter (8 bit). Reading out the datagram counter allows the master to check the success of an initialization sequence or single write accesses. Read accesses do not modify the counter.