Parameterization of StallGuard2™ & CoolStep™

Valid for TMC260, TMC261, TMC262, TMC2041, TMC2130, TMC2209, TMC2226, TMC2300, TMC2660, TMC5031, TMC5041, TMC5062, TMC5072, TMC5130, and TMC5160

This application note is a practical guide for parameterization of the TRINAMIC ICs with StallGuard2, StallGuard4, and CoolStep as well as for evaluation boards and module products using these ICs.

StallGuard2™ High precision sensorless load measurement using the back EMF on the coils. Works with the SpreadCycle current control algorithm.

StallGuard4™ High precision sensorless load measurement using the back EMF on the coils. Works with the StealthChop voltage control algorithm.

CoolStep™ Load-adaptive automatic current scaling based on the load measurement via StallGuard2 or StallGuard4 adapting the required current to the load. Energy consumption can be reduced by as much as 75%.

Note:

In order to understand where to find the parameters mentioned, and how to set them, please refer to the specific datasheet of your driver chip.
The StallGuard of the TMC246 and TMC249 differs from both StallGuard2 and StallGuard4.

1 What is StallGuard?

StallGuard is a sensorless load measurement for stepper motors. We call it StallGuard, because the main use case is to safely detect a stall of the motor in order to replace a mechanical stop switch. However, there are many different uses of StallGuard, like load detection and load dependent motor current control (CoolStep). In order to detect the motor load, StallGuard measures electrical energy flowing into the motor and electrical energy flowing out of the motor again. The difference between both energies gives an indication of the mechanical load taken from the motor. StallGuard2 measures which part of the energy fed to the motor goes back to the power supply. This spare energy is a measure for the mechanical load applied to the motor. If it goes to zero, no spare energy will be left and the motor may stall.

While the energy-based principle is very deterministic, parasitic effects like resistive losses within the motor and motor construction dependence give a major contribution to the quality of the measurement result. These factors can be compensated for by proper settings and proper usage conditions of StallGuard2 in a certain velocity range. Therefore, it is important to understand its limitations. One basic limitation is the usable velocity range: at zero velocity, the motor efficiency is zero, because it cannot deliver mechanical energy to the application, despite that standstill torque is required in many applications. This means, that efficiency at low velocities is also low. Most energy goes to thermal power dissipation in the motor coils. At high velocity, the motor driver cannot bring the full current into the motor anymore, and the motor does not take much more energy than it can consume. The part of energy going back to the power supply decreases. Therefore, the applicable velocity range for StallGuard is limited to something in between low and high velocities. This application note shows how to find the usable range and the required settings.

1.1 How do I select an optimum motor for StallGuard?

A motor with a good efficiency is best, because StallGuard2 measures the difference between the energy going into the motor and the energy used for moving mechanics. Resistive losses within the coil ($P = 2 * R * I_{COIL}^2$) add an offset. This offset will vary with motor temperature because the coil resistance increases with the coil temperature. Also, motor resonance may disturb the measurement. In order for the load measurement to work properly, the motor must work in microstep operation. The motor coils must be able to follow the sign wave, i.e., the velocity must not exceed the value where the Back-EMF plus the resistive voltage loss of the motor exceeds the supply voltage.

From these considerations, a motor will give a more precise StallGuard signal, when it has lower resistive losses within the coil. While the motor torque is proportional to the motor current, resistive losses rise with the square of the current. When a motor has been designed to give the application torque with 80% current, its resistive losses reduce to 64% while 80% of the torque is available.

2 Parameterization of StallGuard2

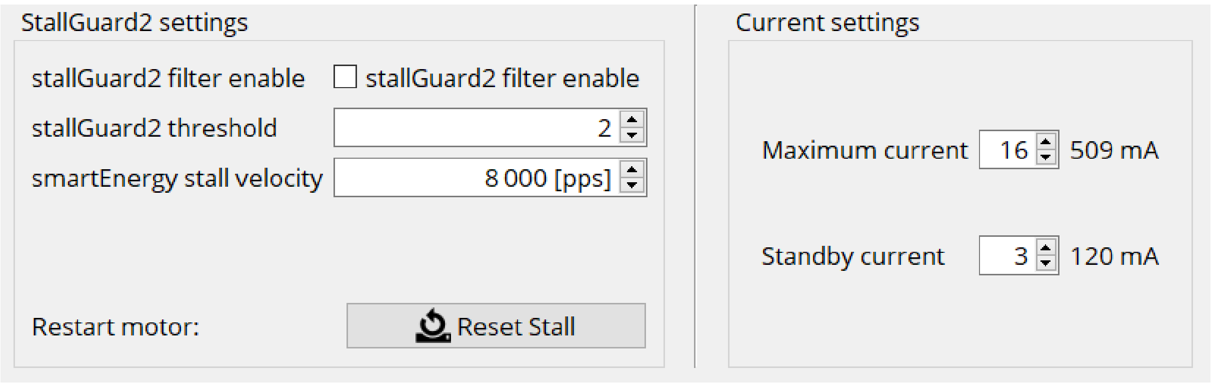
The PWM registers for the SpreadCycle chopper current regulation need to be parameterized first as a base for StallGuard2 and CoolStep. An additional application note is available concerning parameterization of the chopper control.

The graphs are taken from our TMCL-IDE 3.2.0.0 evaluation software.

The CoolStep feature must be switched off (by setting parameter SGMIN=0) while doing the parameterization of StallGuard2. When StallGuard2 is proper parameterized you can proceed with the parameterization of CoolStep.

2.1 Determination of the SGT Parameter

To set up StallGuard2, one single parameter named SGT (StallGuard Threshold) needs to be chosen. With that, one gets a characteristic StallGuard2 value named SG that represents the mechanical load angle resp. the reserve in load angle to stall. The SGT parameter needs to be adjusted in a way that SG = 0 when the motor comes close to stall. SGT ranges from -64…+63.



stallGuard2 filter enable □ stallGuard2 filter enable
stallGuard2 threshold 2
smartEnergy stall velocity 8 000 [pps]
Maximum current 16 509 mA
Standby current 3 120 mA

Figure 2.1 A single parameter named SGT (StallGuard2 threshold) configures StallGuard2

When using the TMCL-IDE another parameter, smartEnergy stall velocity also needs to be set. When the velocity of the motor exceeds the smartEnergy stall velocity, the motor will now stop if the SG value reaches 0. If the smartEnergy stall velocity is above the motor velocity, the motor will not stop if SG reaches 0.
This should be set to a velocity relatively close to the actual velocity. This is because during the acceleration phase the back EMF is changing quickly and this could cause a preemptive StallGuard event to occur.

For our driver IC’s this is not a register, but it is implemented within the TMCL-IDE software. However, for our cDriver IC’s (e.g., TMC5130A) this is a register that must be set. The register name differs from IC to IC so please review the specific driver’s datasheet to find the proper register.

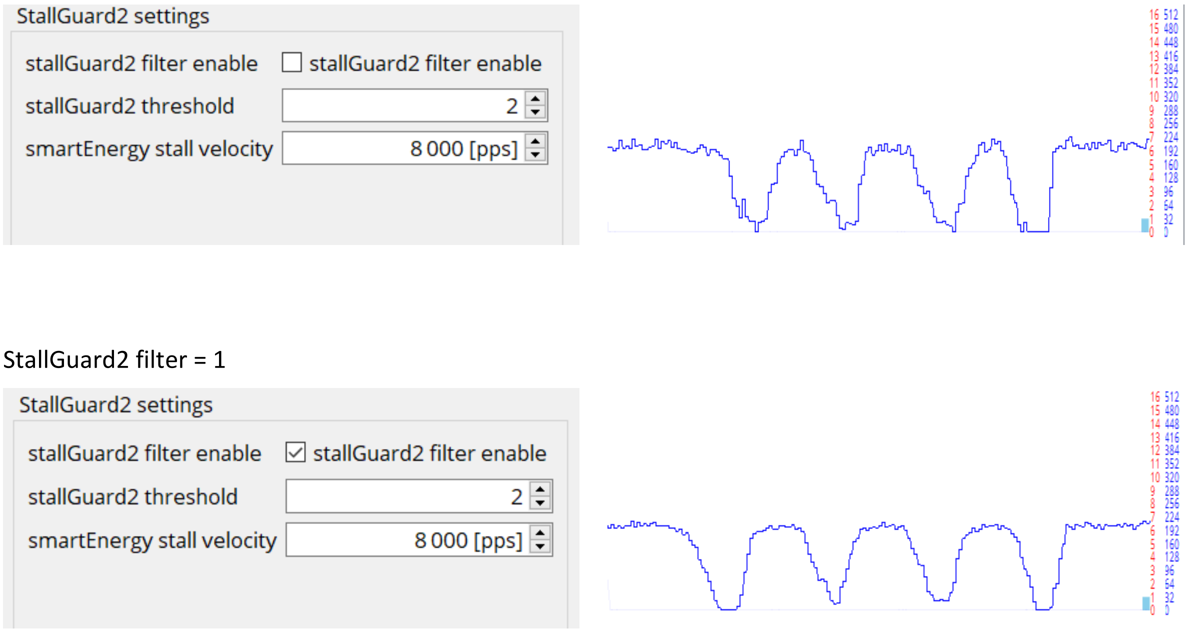
2.1.1 SFILT Switch – Optional Filtering of StallGuard2 Values

The switch named SFILT is available for filtering off the StallGuard2 values SG. Filtering is done for noise reduction over one electrical period (four full steps) of the SG signal (Figure 2.2).

SFILT=O The filtering is disabled, and a change of load is detected within a quarter electrical period that is one full step.

SFILT=1 The filtering is enabled, and a change of load is detected within an electrical period. For higher dynamic change of load the SFILT needs to be switched off.

StallGuard2 filter = 0



StallGuard2 settings

stallGuard2 filter enable □ stallGuard2 filter enable

stallGuard2 threshold 2

smartEnergy stall velocity 8 000 [pps]

StallGuard2 filter = 1

StallGuard2 settings

stallGuard2 filter enable ☑ stallGuard2 filter enable

stallGuard2 threshold 2

smartEnergy stall velocity 8 000 [pps]

Figure 2.2 The StallGuard filter SFILT reduces noise by filtering over four full steps

The SG value depends on a set of base parameters: motor parameters (EMF constant, inductance, and resistance), supply voltage, speed, current, and last but not least the amount of mechanical load, and the StallGuard2 threshold SGT parameter. For a given set of parameters one gets SG values depending on the load.

> The SGT value needs to be adjusted in a way that SG = 0 is at maximum load. Typically, there are more valid SGT values than one which satisfy the requirement SG = 0 at maximum load.

2.1.1.1 Finding the StallGuard2 Threshold Value (SGT Parameter)

Proceed as follows:

1) Choose your nominal phase current (typically as RMS current for motor, peak current for driver).

2) Choose velocity. Try first with speed in the range of a few RPS (revolutions per second).

3) Search SGT within the range -10 … +10, starting with SGT = 0
   a. Set maximum supply voltage (this will be explained later).
   b. Select SGT that you get StallGuard2 values of some 100 in case without load.
   c. Select SGT that you get StallGuard2 values of 0 with mechanical load close before stall.

4) Select one SGT value out of a set of valid SGT values.
   a. Select SGT values with maximum SG value without mechanical load.
   b. Select SGT value close to zero (if possible).
   c. Select SGT value from center of valid range.
   d. Select SGT values almost independent from current (if possible).

Typically, StallGuard2 works best with an SGT value within the range of -10… +10.
If StallGuard2 requires the SGT value to be outside the range of -10… +10, the adjustment might be challenging. In such cases it is recommended to change base parameters (speed, motor coil inductance, motor coil resistance). Even though StallGuard2 works with a StallGuard2 threshold parameter (SGT) out of the range of -10… +10 the set of motor parameters, speed, supply voltage, and the stall detection might work, but the combination of these parameters might be sub-optimal.

2.2 Detecting a Stall

While monitoring the StallGuard2 value via the TMCL-IDE, you may actually miss the last StallGuard2 value which precedes the stall, especially when working at higher velocities. This is due to the limitations of the update rate of the display. By finding the lowest possible reading preceding the stall, you will gain safety margin and avoid wrong stall detections. For this application, the StallGuard filtering should be turned off to give fastest response.

In order to detect the lowest possible reading, you should monitor the StallGuard output the driver IC has. This way, you can increase SGT (or decrease the SG threshold which your software uses to detect a stall) starting from an initial value where StallGuard2 acts too sensitively, until a stall is not safely detected anymore. For a safe and stable result, the test should be repeated with different motors and electronics and at different motor temperatures. This way, you find the possible stray and get a feeling for the required safety margin to compensate for production stray.

Some drivers, like the TMC262 or TMC2660, have a dedicated pin for the StallGuard output (i.e., SG_TST).
Other IC’s, like the TMC2130 and TMC2160, use DIAG pins for this functionality. To set up the DIAG pin to operate as a StallGuard output go through the specific IC’s datasheet and read the diagnostics section.

2.3 How to Apply Mechanical Load

From mechanical point of view, the application of mechanical load might be a challenge. For small motors one can apply mechanical load manually. For large stepper motors this is not possible. This application note proposes a solution with two stepper motors of the same type with both motors mechanically coupled.

2.3.1 Applying Mechanical Load (manually) with Wheel

For small motors (NEMA 17 resp. 42er with some 0.1 Nm) one can apply mechanical load close to stall with a wheel mounted at the motor axis. With that, one can hold the motor by hand and observe the StallGuard2 signal if it comes to zero if the motor is close to stall or stalls.

Be careful when applying mechanical load manually. Braking a motor manually can cause injury and is on your own risk!

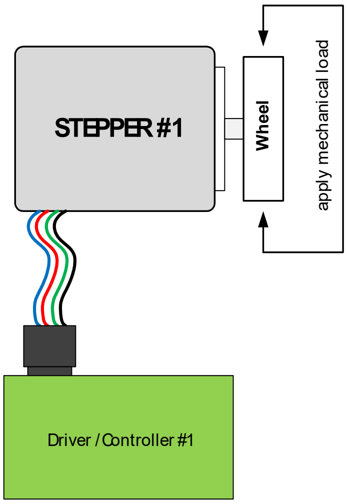


Figure 2.3 Applying mechanical load with a single motor

2.3.2 Apply Mechanical Load with a 2nd Motor

The best way applying mechanical load to a stepper motor is using two identical stepper motors coupled mechanically axis to axis. The setup is outlined in Figure 2.4 and Figure 2.5. Both stepper motors need to be fixed on a common base plane.

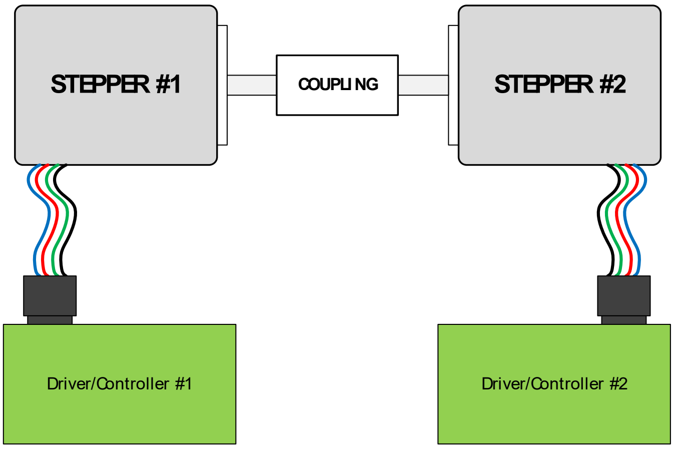


Figure 2.4 Applying mechanical motor load using a second mechanical coupled stepper motor

2.3.3 Mechanical Load via the Difference of Micro Step Positions

To apply load from zero up to stall or close to stall one needs to run both steppers with a difference with up to one full step. This can be archived by

a) running two motors at identical speed with a constant micro step difference or by
b) running two motors at slightly different speeds.

First one needs to switch on the current for both stepper motors. Going close to stall but avoiding it is possible by choosing a lower current for stepper motor #2. Stepper motor#2 operates as brake for motor #1.

First case (a): Run motor #1 a number of micro steps to apply a load. Then run both motors at identical speed.

Second case (b): One periodically drives through the whole range of mechanical load.

Depending on the microstep direction one can simulate both kinds of mechanical load: braking load and acting as a generator.
Due to frequency differences of quartz oscillators (typical within range of some 0.001%) or quartz resonators (typical within range of 0.5%) the load can change slowly over time - even if two coupled stepper motor controllers are configured to run at the same speed. To avoid this one can either use a multi-axis controller clocked with a single clock or clock for two different stepper motor controllers with a single clock.

2.3.3.1 Mounting Procedure of Mechanical Coupling of Two Stepper Motors

Two mechanical coupled stepper motors (Figure 2.5) need to be fixed on a base plate. In principle one can hold two of those coupled stepper motors on a desktop, but for higher torque stepper motors a fixing is required. Both stepper motors rotate against each other if the speed is different. The torque between two coupled stepper motors that rotate against each other is the applied torque.

Proceed as follows:

1) Both stepper motors first should be mounted on the base plate, with a freewheeling mechanical coupling.
2) Both fixed stepper motors need to be driven with the same current at the same microstep position.
3) Fixing the screws of the mechanical coupling element gives a mechanical coupling without torque.

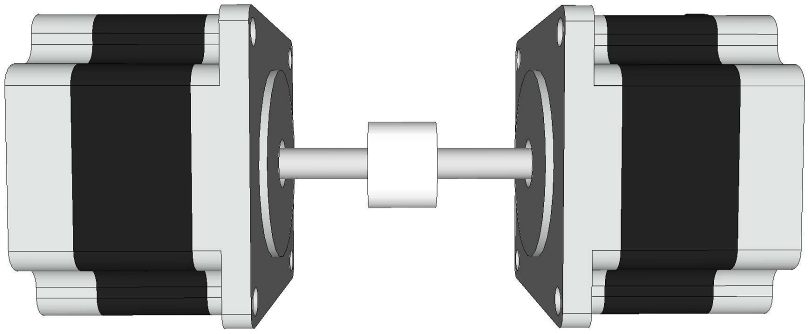


Figure 2.5 Two mechanical coupled stepper motors of same type

2.4 Load Angle

The load angle is the angle between the magnetic orientation of rotor and stator. The magnetic orientation of the rotor is assigned to the mechanical angle of the axis of a stepper motor. The magnetic orientation of the stator results from the phase currents feed through the coils of the stepper motor. Figure 2.6 outlines the rotor and stator with magnet poles outlined (red: north pole, green: south pole).

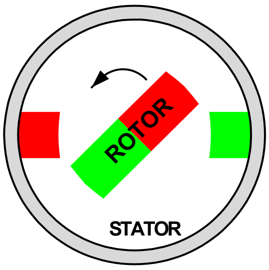


Figure 2.6 Outline of rotor and stator

In Figure 2.7 three different cases of load angles are outlined. These are *NO LOAD* when there is no mechanical load at the axis while the stepper motor is freely running; *HALF LOAD* when the load is between *NO LOAD* and the maximum load; *MAXIMUM LOAD* when the rotor is *close to stall*.

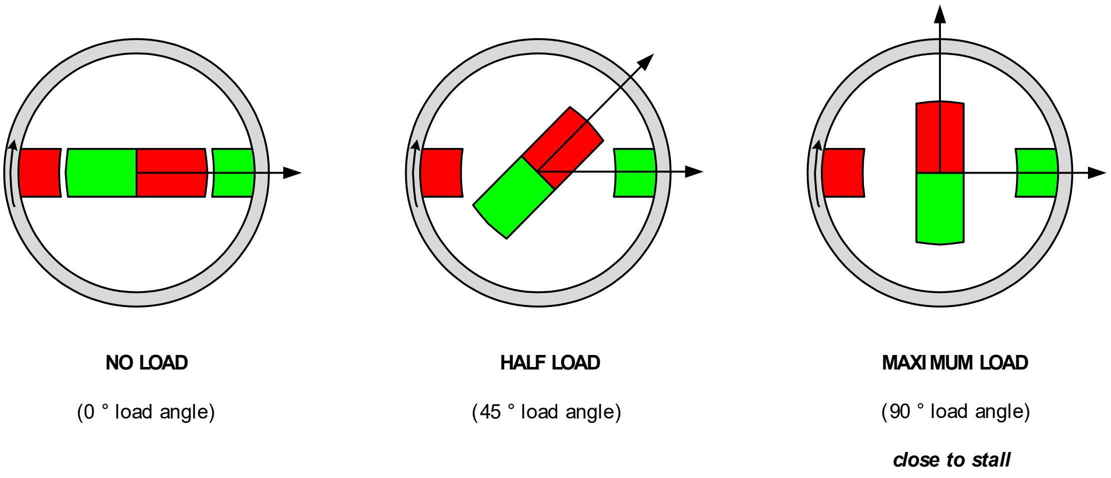
The stator is static, and the rotor is rotating. Turning the rotor is forced by the rotating magnetic field of the stator. From the rotor point of view, there is a magnetic field pulling or pushing more or less that causes a phase shift between the magnetic field direction of the rotor and the magnetic field direction of the rotating field of the stator. This phase shift is the load angel. It is the angle between the magnetic field direction of the rotor and the magnetic field direction of the rotating magnetic field of the stator.

The load angle depends on the external torque applied to the axis of the stepper motor. The StallGuard2 value SG *represents* the load angle and the torque.

The maximum possible load for a stepper without stall is one full step. This is equivalent to a load angle of 90°. With a sufficient SGT value, one gets StallGuard2 values in a range of SG_MAX to SG_MIN=0 where SG_MAX represents *NO LOAD* and SG_MIN=0 represents the *MAXIMUM LOAD* resp. *stall*.

Without mechanical load: the StallGuard2 threshold SGT needs to be chosen in a way that one gets a sufficient StallGuard2 value SG in the middle of the full range of SG.

With load close to stall: the SG value needs to be zero by tuning SGT (see 2.1.1.1).


NO LOAD

(0 ° load angle)

HALF LOAD

(45 ° load angle)

MAXI MUM LOAD

(90 ° load angle)

close to stall

Figure 2.7 Outline of different load angles

2.5 Plots of StallGuard2 Value SG for Different SGT Values

The following plots are done with a setup of two mechanical coupled stepper motors of the same type (according to Figure 2.4 and Figure 2.5) running with a slightly different speed (v#1 of motor 1 = 1.001 * v#2 of motor #2) and different currents.

The driving motor #1 that is under parameterization is driven with nominal current (100%). Motor #2 that is used to apply mechanical load is set to lower current (e.g., 50% of nominal current of motor #1). So, motor #1 is always able to drive the load from motor #2 without stall but close to stall of motor #1. The load motor #2 periodically loses four full steps because of its lower current, at that point the mechanical load vanishes for a short time (Figure 2.8, Figure 2.9).

For parameterization of StallGuard2, the driving motor #1 must run faster than the load motor #2 to simulate pulling load. Otherwise, it would operate as a generator and will lead to an increase in the StallGuard2 signal with increasing load.

With a good choice of the StallGuard2 threshold parameter SGT, one gets a plot like Figure 2.8 with StallGuard2 values SG in a range of some hundred and a clear zero for maximum load.

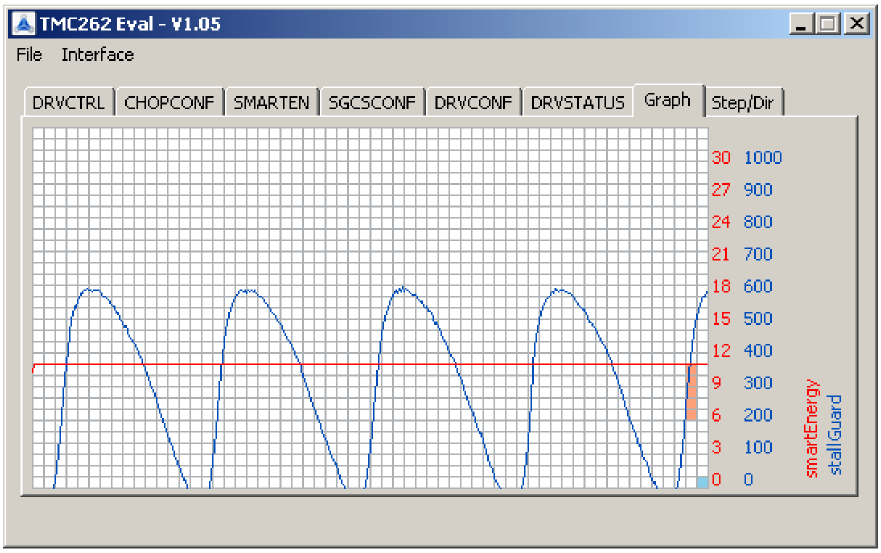


Figure 2.8 SG values for good choice of StallGuard2 threshold SGT

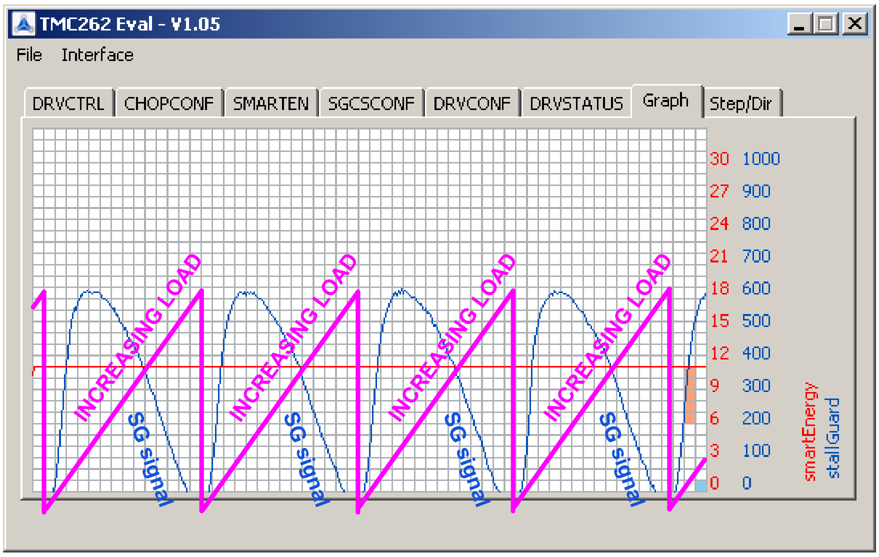


Figure 2.9 StallGuard2 values SG decrease with increasing mechanical load

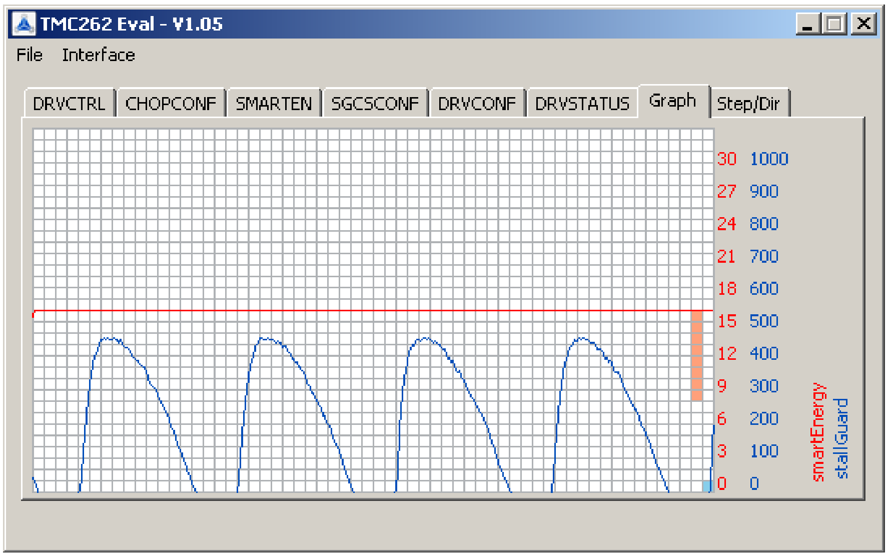


Figure 2.10 StallGuard2 values SG with too low StallGuard2 threshold value SGT

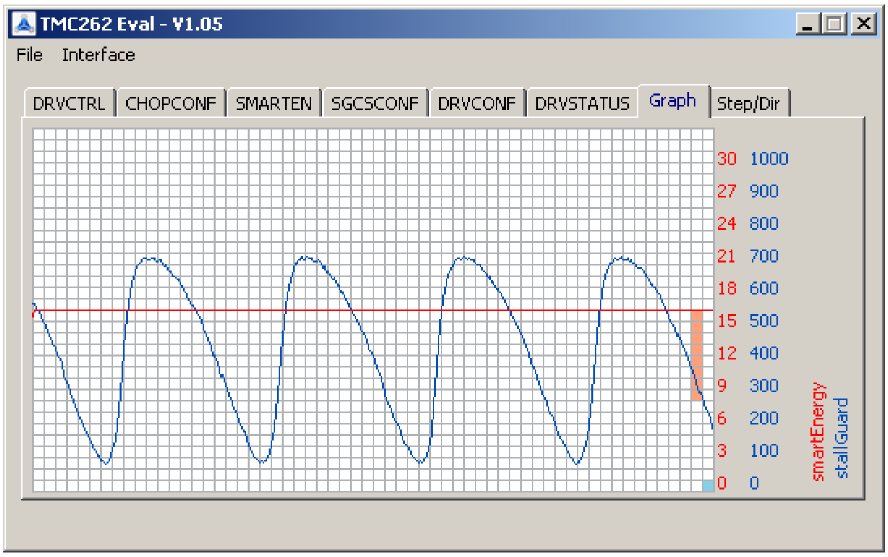


Figure 2.11 StallGuard2 values SG with high StallGuard2 threshold value SGT

2.5.1 StallGuard2 Values SG

A StallGuard2 value represents the actual mechanical load for a given fixed parameter set. Generally, there are a couple of SGT values that give sufficient StallGuard2 values SG. The challenge is to choose a range of speed for operation, to benchmark a chosen setting, and to choose the SGT value which fits best.

From mathematical point of view, the StallGuard2 value

SG = f(V_M[V], kappa_e[V/rad/s], L[mH], R_L,[Ohm] v[FS/s], phi[FS], SGT)

is a function of

kappa_e[V/rad/s]	⇔ Back EMF constant of the motor [in unit Volt per rad per second]
L[H]	⇔ coil inductance of the motor [in unit Henry]
R_L[Ohm]	⇔ coil resistance of the motor [in unit Ohm]
v[FS/s]	⇔ speed [in unit full step per second]
phi[FS]	⇔ load angle [in unit full step]
I_L[A]	⇔ phase current [in unit Ampere]
V_M[V]	⇔ motor supply voltage [in unit Volt]

2.5.1.1 Variation of StallGuard2 Values SG with Supply Voltage

For most industrial applications constant voltage power supplies are used. The supply voltage is constant within a specified range of tolerance. The StallGuard2 value SG varies with the supply voltage (Figure 2.12). The upper value of SG representing NO LOAD decreases with increasing supply voltage. The lower value of SG slightly increases with increasing supply voltage.

> The adjustment of the StallGuard2 threshold SGT parameter – that needs to be set in a way that SG=0 at maximum mechanical load – should be done at maximum supply voltage.

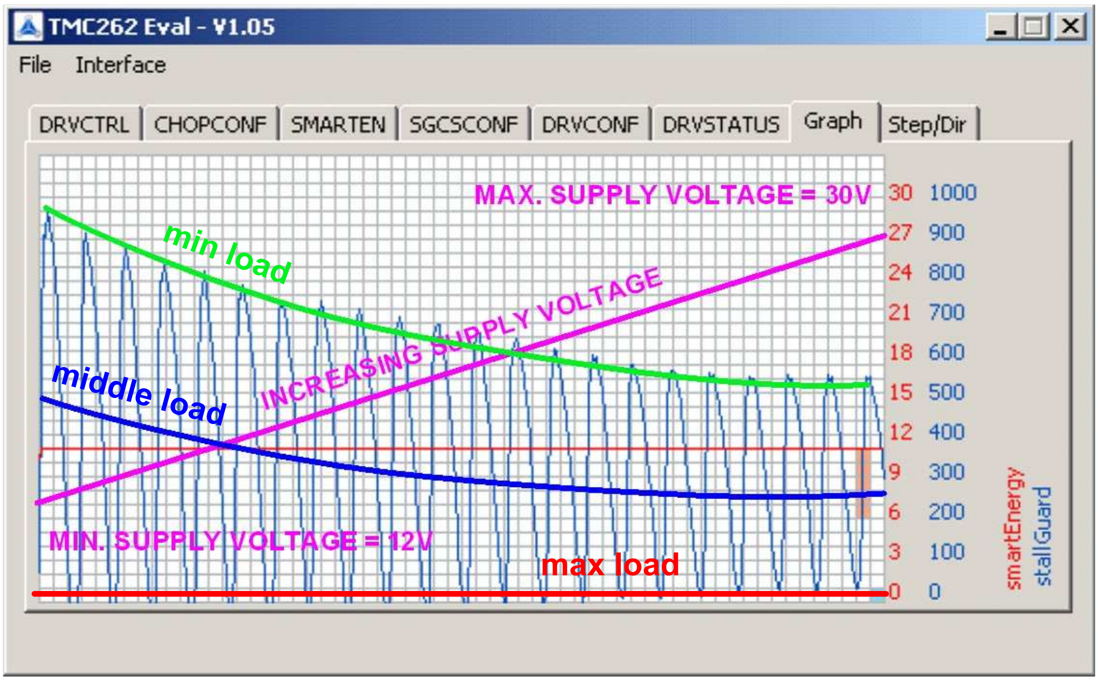


Figure 2.12 Variation of StallGuard values SG with supply voltage

2.5.1.2 Variation of StallGuard2 Values SG with Selected Current

The StallGuard2 value SG depends on the amplitude of the motor current linked to the supply voltage. For using StallGuard2 itself with a constant voltage supply this is not important. It is important for CoolStep because it changes the current amplitude. Programmable upper and lower limits of CoolStep are provided for handling the dependency of the StallGuard2 value SG from the current amplitude.

Figure 2.13 shows an example of StallGuard2 values SG for four different current settings (current scale CS = 7, 15, 23, 31) at different constant supply voltages without mechanical load. Depending on the supply voltage, the dependency varies. For that example, the dependency vanishes for a supply voltage of near 21.0V.

Figure 3.2 shows an example with load at constant supply voltage together with the CoolStep function. There, one can see that the StallGuard2 value SG changes with the change of current. In addition, the dynamic change of current on the detection of mechanical load causes a dynamic change of mechanical load and that has an effect on the StallGuard2 vale SG, also.

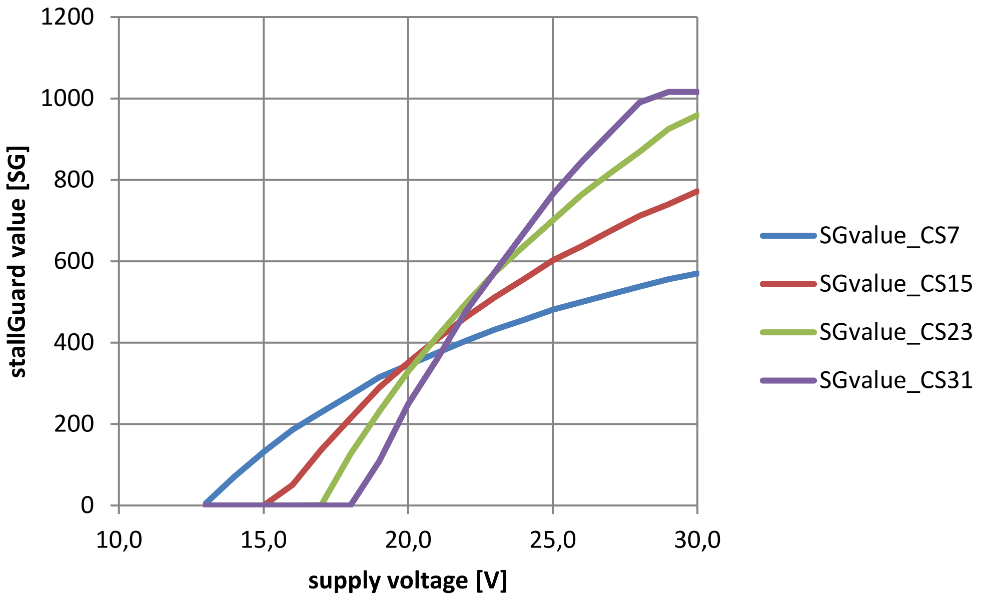


Figure 2.13 Dependency of StallGuard value [SG] from current scaling

2.5.1.3 Variation of StallGuard2 Values SG with Speed

The StallGuard2 value SG varies with the velocity because it bases on the Back-EMF of the stepper motor. At low speed, the Back-EMF is too low for measurement. Figure 2.14 shows the StallGuard2 value SG over speed without mechanical load. With mechanical load close to stall, one gets StallGuard2 values SG of zero. At Resonance, one can see the detection of dynamic load due to the resonance. Increasing the speed causes the StallGuard2 value SG to become more and more independent from speed. Nevertheless, the dependency needs to be evaluated for a given type of stepper motor and one need to define limits where StallGuard2 can be used.

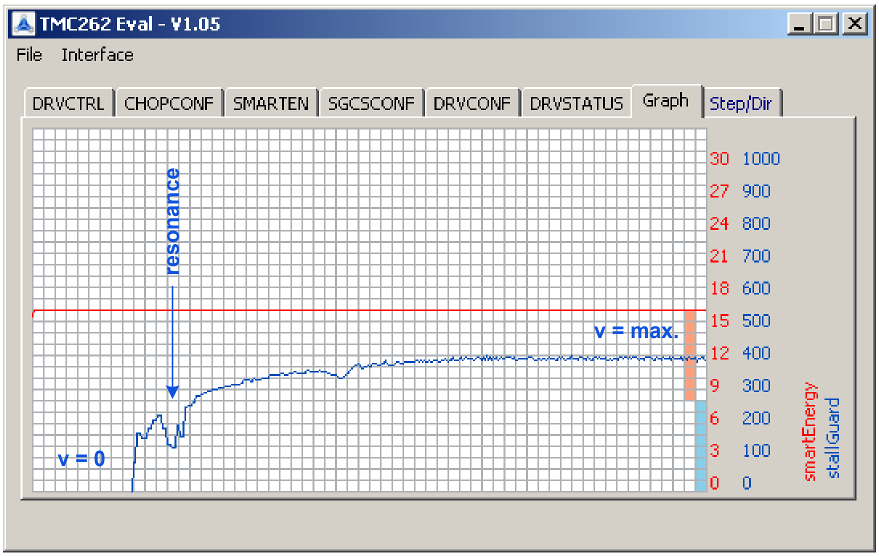


Figure 2.14 Dependence of StallGuard value SG from velocity v

Figure 2.14 only shows the behavior of the StallGuard2 value SG until it becomes stable with respect to increasing velocity. Nevertheless, when increasing velocity beyond the point where the current in the motor no longer reaches the chopping current due to the Back-EMF the StallGuard2 value SG will decrease again. In this case one has to check for an increasing StallGuard2 value SG to identify situations of increased load angle and possibility of motor stall.

In such situations, when Back-EMF becomes too high and the chopping current cannot be reached any more, one can lower the CS value (current scale value) of the Trinamic driver IC to reduce the chopping current to a peak value that can still be reached within the motor coils even with high Back-EMF. This is OK since the effective torque outputs depends on the current in the motor coils.

To summarize: there are different StallGuard2 values for different speeds / operating points. In applications with more than one operating point and when the operating points are spread over a wide velocity range it is recommended to use individual settings for these operating points if StallGuard2 is used.

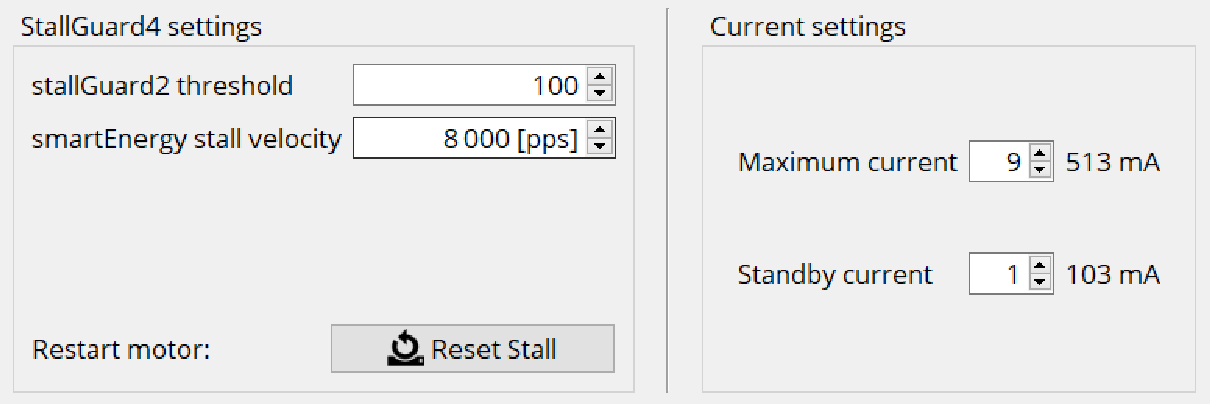
3 Parameterization of StallGuard4

StallGuard4 differs from StallGuard2 in multiple aspects. It uses the StealthChop voltage chopper and not SpreadCycle. The StealthChop tuning must be done before parameterizing StallGuard4. Please refer to the IC’s datasheet to tune StealthChop.

The graphs are taken from our TMCL-IDE 3.2.0.0 evaluation software.

3.1 Determination of the SGTHRS Parameter

The SGTHRS parameter operates much differently than the SGT parameter. This value controls the StallGuard4 threshold for stall detection. It compensates for motor specific characteristics and controls sensitivity. A higher value gives a higher sensitivity. A higher value makes StallGuard4 more sensitive and requires less torque to indicate a stall.
The SGTHRS value ranges from 0...255. The double of this value is compared to SG_RESULT. The stall output becomes active if SG_RESULT falls below this value.
SG_RESULT is the StallGuard value. It is similar to the SG value from StallGuard2. However, SG_RESULT has a range of 0...510.



StallGuard4 settings

stallGuard2 threshold: 100

smartEnergy stall velocity: 8 000 [pps]

Restart motor: Reset Stall

Current settings

Maximum current: 9 513 mA

Standby current: 1 103 mA

Figure 3.1 A single parameter named SGTHRS (StallGuard4 threshold) configures StallGuard4

When using the TMCL-IDE, another parameter, smartEnergy stall velocity also needs to be set. When the velocity of the motor exceeds the smartEnergy stall velocity, the motor will now stop if the SG value reaches 0. If the smartEnergy stall velocity is above the motor velocity, the motor will not stop if SG reaches 0.
This should be set to a velocity relatively close to the actual velocity. This is because during the acceleration phase the back EMF is changing quickly and this could cause a preemptive StallGuard event to occur.

On the TMC2209 and TMC2226 the smartEnergy stall velocity is the TCOOLTHRS register. TCOOLTHRS is compared to TSTEP, which is the measured time between two 1/256 microsteps derived from the step frequency. As this is the time in-between step pulses, a smaller value means a higher velocity. When **TSTEP < TCOOLTHRS**, StallGuard4 have the ability to activate.

3.1.1 Finding the StallGuard4 Threshold Value (SGTHRS Parameter)

Proceed as follow:

1) Choose your supply voltage and nominal phase current (typically as RMS current for motor, peak current for driver).

2) Choose velocity. Try first with a speed in the range of some rps (revolutions per second).

3) Apply a slowly increasing mechanical load to the motor. Check the lowest value of SG_RESULT before the motor stalls. Take this SG_RESULT value, divide it by 2, and put it into SGTHRS.

4) Set TCOOLTHS to match the lower velocity limit for StallGuard operation. Monitor the StallGuard output signal via the DIAG pin output. When this pin goes high, stop sending steps to the driver.

5) The optimum setting is reached when a stall is safely detected and leads to a pulse at DIAG in the moment where the stall occurs

3.2 Detecting a Stall

While monitoring the StallGuard4 value via the evaluation board tool, you may miss the last StallGuard4 value which precedes the stall, especially when working at higher velocities. This is due to the limitations of the update rate of the display. By finding the lowest possible reading preceding the stall, you will gain safety margin and avoid wrong stall detections.

In order to detect the lowest possible reading, you should monitor the DIAG output, or use special software in order to reach to a certain threshold for the StallGuard4 reading e.g., by stopping the motor. This way you can adapt SGTHRS starting from an initial value where StallGuard4 reacts too sensitively, until a stall is not safely detected anymore. For a safe and stable result, the test should be repeated with different motors and electronics and at different motor temperatures.

3.3 Variation of StallGuard4 values

StallGuard4 operates off a different MOSFET chopping scheme than StallGuard2 does. StealthChop is a voltage control scheme for the MOSFETs and is used with StallGuard4. Please see the StealthChop application note to learn more on how it works. StealthChop adapts the PWM_SCALE values to ensure the motor is reaching the target current. As the motor voltage decreases, the PWM_SCALE value increases. Because of this operation, the StallGuard4 value does not have the same level of dependence on the motor voltage as the StallGuard2 value does.

Figure 3.2 shows this non-dependency. This graph was taken at a constant CS value of 16. As the supply voltage of the motor is increasing over time, the PWM_SCALE value slowly decreases. The StallGuard value is varying but is not impacted by the change in voltage.

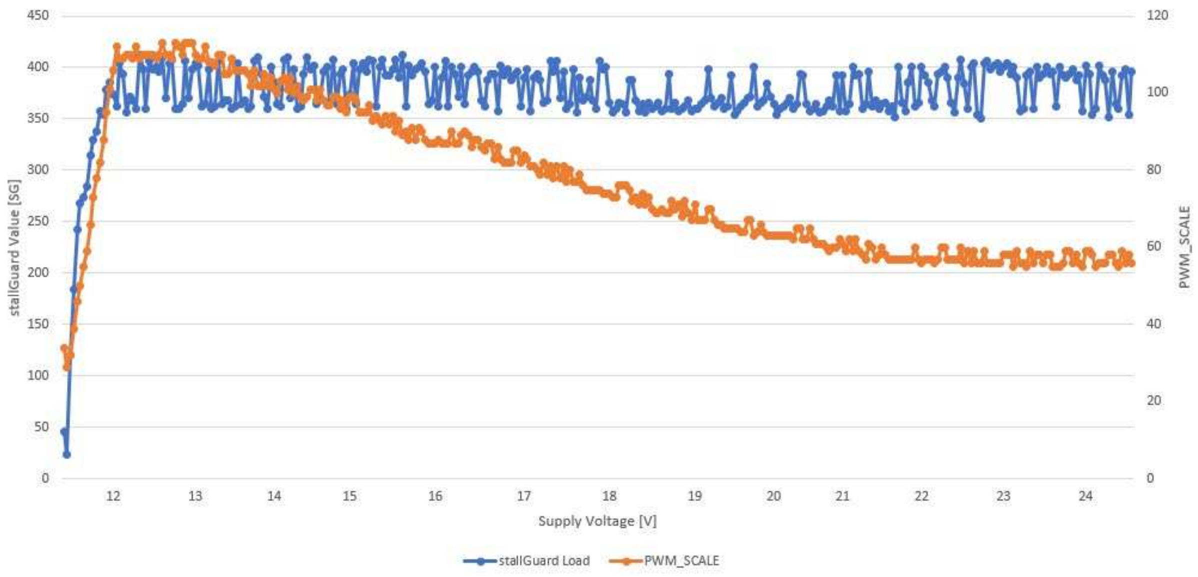


Figure 3.2 Non-dependency of StallGuard4 value [SG] from voltage

This is also true for the motor run current. In figure 3.3 you can see that as the current increases (red line) the StallGuard value (blue line) does not change in the same way as it did for StallGuard2. The StallGuard value does become more consistent as the current increases, but it does not jump up or down by noticeable margins like it did with StallGuard2.

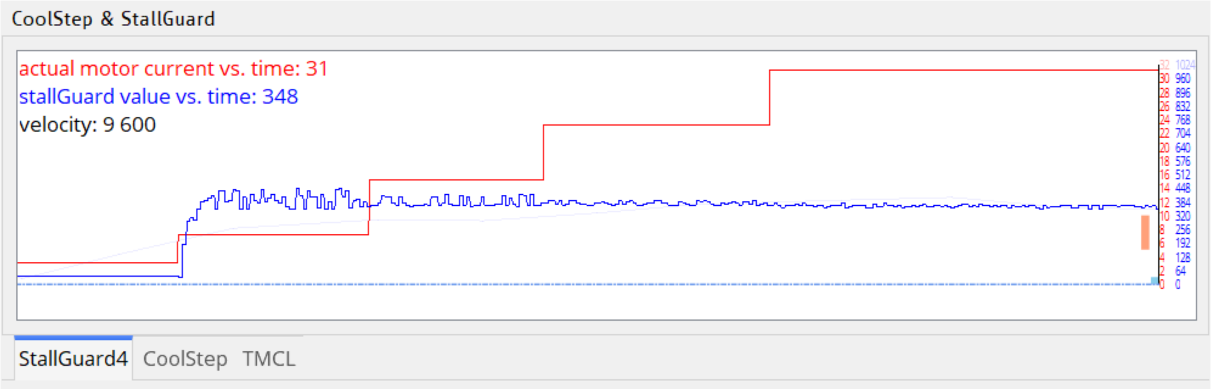


Figure 3.3 Non-dependency of StallGuard4 value [SG] from current

Like StallGuard2, the StallGuard4 value varies with the velocity of the motor. This is because StallGuard is based on the back-EMF the motor is producing.
Section 2.5.1.3 goes over this in more detail.

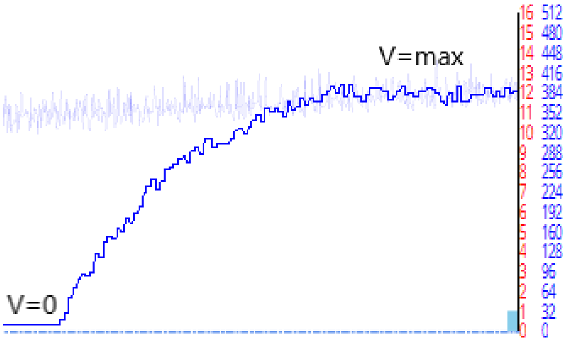
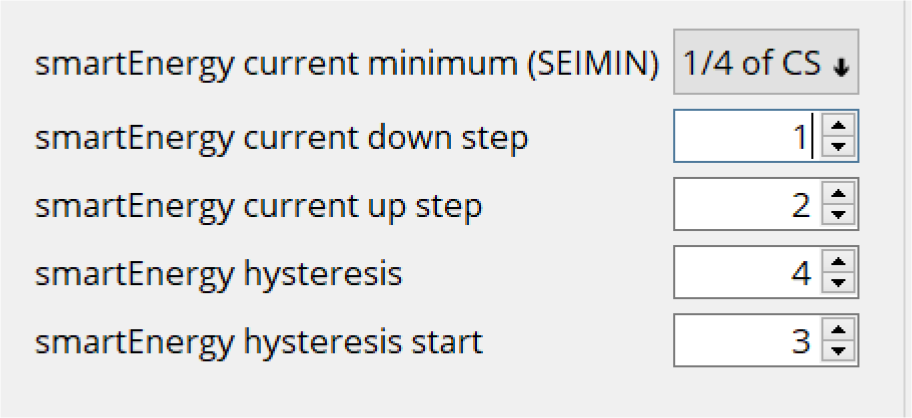


Figure 3.4 Dependency of StallGuard4 value [SG] from velocity

4 Parameterizing CoolStep

The CoolStep function is based on the sensorless stall detection. Before CoolStep can be used, StallGuard2 needs to be adequately parameterized. The CoolStep parameter register (SMARTEN or COOLCONF) is described within the specific IC datasheet.

With SEIMIN=O the CoolStep function is disabled.



smartEnergy current minimum (SEIMIN)	1/4 of CS
smartEnergy current down step	1
smartEnergy current up step	2
smartEnergy hysteresis	4
smartEnergy hysteresis start	3

Figure 4.1 Five parameters configure smart energy CoolStep

For choosing CoolStep parameters, proceed as follows:

1) Choose scaling ½ or ¼ with SEIMIN (start with ½).
2) Choose current decreasing rate SEDN (start with 32 or 8).
3) Choose SG_LOW lower limit and calculate SEMIN for chosen SG_LOW.
4) Choose SG_HIGH upper limit and calculate SEMAX for SG_HIGH.
5) Choose current increasing rate SEUP (start with fast 3 or 2).

4.1 SEIMIN ½ vs. ¼

This parameter selects the scale of current reduction. Two setting are available for ½ current and for ¼ current when the load below the programmed limit is detected. Half current is best for most applications that are compatible with CoolStep. Quarter current also fits with many applications that are compatible with CoolStep, but parameterization might be more challenging.

Half current results in 25% power dissipation because the power dissipation is proportional to the square of the current. With 25% power dissipation you get a 75% power saving. Quarter current results in 6.25% power dissipation. If an application works with other than 50% or 25% current reduction, you can do this under software control.

First, evaluate an application with 1/2 current setting.

4.2 SEMIN = 0 ⇔ Switch CoolStep = OFF Else Lower StallGuard2 Value SG Limit

With SEMIN = 0 CoolStep is switched off. The SEMIN parameter defines the minimum value of the StallGuard2 signal SG that triggers the increasing of the current if load is detected. The SEMIN parameter is internally multiplied by 32. So, the increasing of current is triggered if SG < (SEMIN * 32). To start parameterization, first choose a SEMIN in the range of SG_MAX / 4 … SG_MAX / 8.

4.3 SEDN ⇔ Current Decreasing Speed

This parameter defines the speed of decreasing the current. A low decreasing rate fits best but a high decreasing rate saves more energy. So, you should start with one of the lowest current decreasing rates (SEDN=00 or SEDN=01). A low decreasing speed avoids oscillations of current regulation. Later, you can optimize energy saving by evaluation faster decreasing rates.

4.4 SEMAX ⇔ Upper StallGuard2 Value SG Limit

This parameter defines the upper threshold for the StallGuard value SG. This represents the lower load limit, because the StallGuard2 value SG represents the reserve of load angle to stall condition. The decreasing of current is triggered when the load is below a limit. In other words, it is triggered when the SG value is above a limit. The decreasing of current is triggered if SG > (SEMIN + SEMAX + 1) * 32.

4.5 SEUP ⇔ Current Increasing Speed

This parameter defines the speed of increasing the current when a load is detected. A high increasing rate ensures fast reaction on change of load but a very fast might cause a small jerk. So, you should start with one of the fastest current increase rates (SEUP=11 or SEUP=10) and reduce the increase rate if necessary due to jerk.

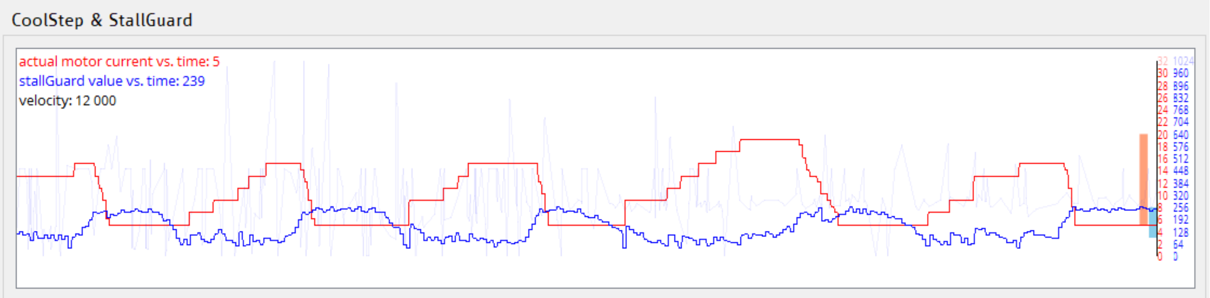


Figure 4.2 Change of SG value due to change of current done by CoolStep

In Figure 4.2 there are two bars on the right. The orange bar is representing the maximum current and SEIMIN value. The blue bar is the hysteresis created from the SEMAX and SEMIN parameters. As the motor is loaded the StallGuard value goes down until it reaches the bottom of the hysteresis. At this point the current increases based on the SEUP value. CoolStep will continue at this current until either the StallGuard value goes down further, which will cause the current to increase, or the StallGuard value increases, which will cause the current to decrease. This is repeated multiple times in figure 4.2.

5 Summary

This application note describes how to set up StallGuard by determination of a single parameter: the StallGuard threshold value. Solutions are presented to apply mechanical load that is required for optimal adjustment of the StallGuard threshold value. As the StallGuard value depends on parameters like velocity and motor characteristics (Back-EMF) more than one set of parameters should be defined in applications with different operating points. These parameter sets can be configured at runtime in the Trinamic driver ICs.

6 Revision History

6.1 Document Revision

Version	Date	Author
		LL = Dr. Lars Larsson
		SD = Sonja Dwersteg
		SK = Stephan Kubisch
		BD= Bernhard Dwersteg
			Description

1.0	2012-JAN-24	LL	Initial version, first published.
1.01	2012-FEB-16	SD	New design, minor changes.
1.02	2013-APR-09	SK	Additional information on high velocity situations and Back-EMF dependency in Section 1.4.1.3
1.03	2013-APR-19	BD	Added "what is StallGuard" and considerations on choice of motor, added "detecting a stall"
1.04	2014-NOV-03	BD	Improved wording for basic introduction
1.05	2018-MAR-01	LL	Updated Figure 2.12 Variation of StallGuard values SG with supply voltage
1.06	2021-APR-12	BK	Updated StallGuard2 graphs to TMCL-IDE 3.2.0.0 Added StallGuard 4 section

Table 1: Document revision

7 References

[TMC2130]	TMC2130 Datasheet
[TMC2209]	TMC2209 Datasheet
[TMC262]	TMC262 Datasheet
[TMC262-EVAL]	TMC262-EVAL Manual
[TMC429+TMC26x-EVAL]	TMC429+TMC26x-EVAL Manual

Please refer to our web page http://www.trinamic.com.