User's Guide

MCF8329A Tuning Guide

ABSTRACT

This tuning guide provides step-by-step guidance to setup the MCF8329EVM, connect the MCF8329EVM to Motor Studio, and tune a 3-phase brushless DC motor using the MCF8329A motor driver.

Table of Contents

1 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

DATE	REVISION	NOTES
September 2023	*	Initial Release

Trademarks

2 Introduction

The MCF8329A is a 4.5-V to 60-V, three-phase brushless-DC (BLDC) gate driver IC with code-free sensorless field oriented control (FOC) for motor drive applications. The device integrates a single shunt current sense amplifier (CSA) along with an external shunt resistance to sense the motor current. This document helps customers to set up the MCF8329A, enabling them to experience the devices' powerful performance and flexible programmability.

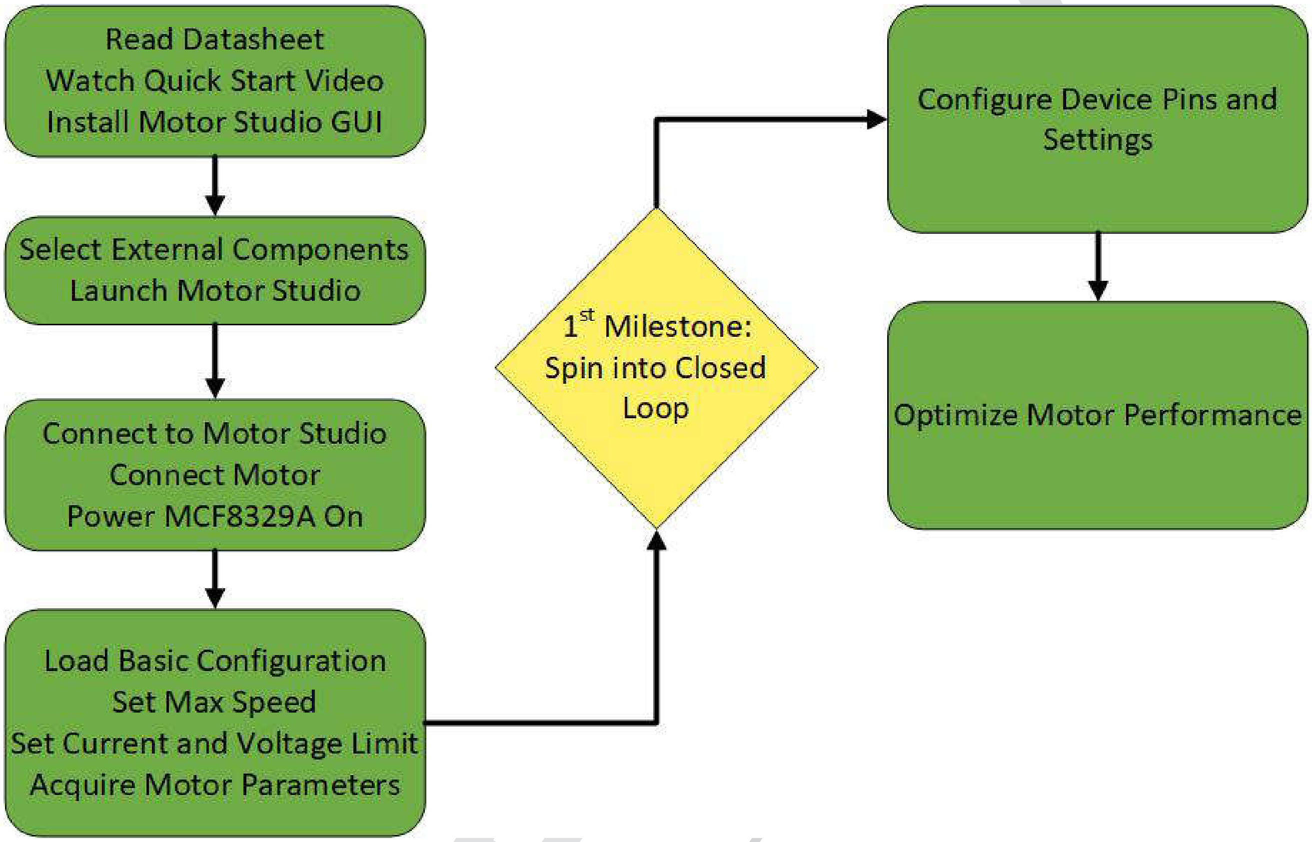

Figure 2-1. Tuning Sequence of Events

Before proceeding through this tuning guide make sure to do the following:

1. Read the MCF8329A datasheet, MCF8329EVM User's Guide, and watch the quick start video.
2. Get a MCF8329EVM board.
3. Install the Motor Studio application.

---

3 Hardware Design and Setup

The goal of this section is to help users select appropriate components for the external power stage components of the MCF8329A and set up the user configurable settings of the MCF8329EVM.

3.1 Board Design

The following sections provide equations and guidelines for selecting power stage components to achieve desired performance from the motor driver system.

3.1.1 External MOSFET Selection

The MOSFETs for the external half bridge that can be supported by the MCF8329A can be determined by inputting the MOSFET gate charge, output PWM switching frequency, and PVDD voltage into the Max Qg MOSFET Calcualtor Tool available on ti.com.

3.1.2 Gate Resistor Selection

Selection of an appropriate gate resistance to limit the gate drive current so that the drain-to-source voltage slew rate (VDS) is set to an appropriate level for the external MOSFETS is essential to achieving good system performance. For more information on the importance of and how to select an appropriate gate resistor value, please refer to the Gate Drive Current and Gate Resistor Selection sections in the MCF8329A datasheet. To simplify the gate resistance selection process, the Gate Resistor Calculator can be used to estimate the gate resistance required to achieve a desired VDS rise and fall time with an accuracy of ±30%.

3.1.3 Bootstrap and GVDD Capacitor Selection

The bootstrap and GVDD capacitors must both be sized appropriately to maintain the bootstrap voltage above the under-voltage lockout threshold during normal operation. For instructions to determine an appropriate capacitance for both the bootstrap capacitors and GVDD capacitor, please refer to the Bootstrap Capacitor and GVDD Capacitor Selection sections of the MCF8329A datasheet.

3.1.4 Current Shunt Resistor Selection

The internal FOC algorithm uses the output of the internal current sense amplifier (CSA) in its computations. It is recommended to set the max measurable current of the internal CSA to 10% above the motors stall current. To determine an appropriate value for the CSA gain and external low-side shunt resistor, please refer to section 7.3.5 of the MCF8329A datasheet.

3.1.5 VREG MOSFET Selection

the GCTRL pin can be used to drive an external MOSFET which can be used as a voltage regulator to provide power to the VREG pin to reduce power dissipation within the MCF8329A. Instructions on how to select an appropriate MOSFET are provided in section 8.2.1 of the MCF8329A datasheet.

3.1.6 Additional External Power Stage Components

For additional considerations for external components of the MCF8329A and for high power systems, please refer to the System Considerations in High Power Designs, Capacitor Voltage Ratings, and External Power Stage Components sections of the MCF8329A datasheet.

4 Connecting to the GUI

Before connecting the MCF8329EVM to the computer, start up the Motor Studio application and select MCF8329A from the drop down. Click on *Proceed* and then click on the *Setup Now* button for instructions on how to connect power, connect a motor, and configure the jumpers and switches on the EVM.

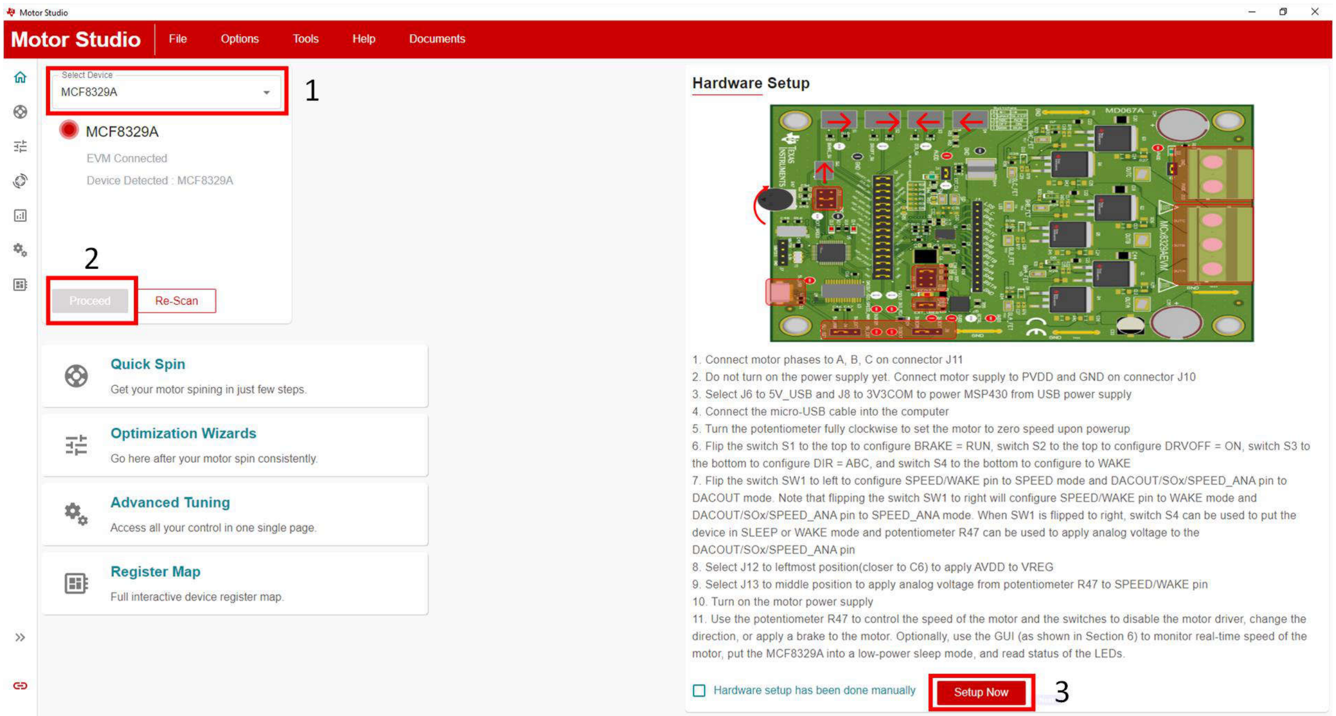

Figure 4-1. EVM Hardware Setup

Once the hardware setup is completed, turn on the power supply connected to the EVM. After the PVDD LED D3 lights up connect a micro-USB to USB cable between the EVM and PC. After a few seconds, Motor Studio should connect to the EVM and the two icons outlined in Figure 4-2 will turn green. If the EVM is not connecting click the *Re-Scan* button.

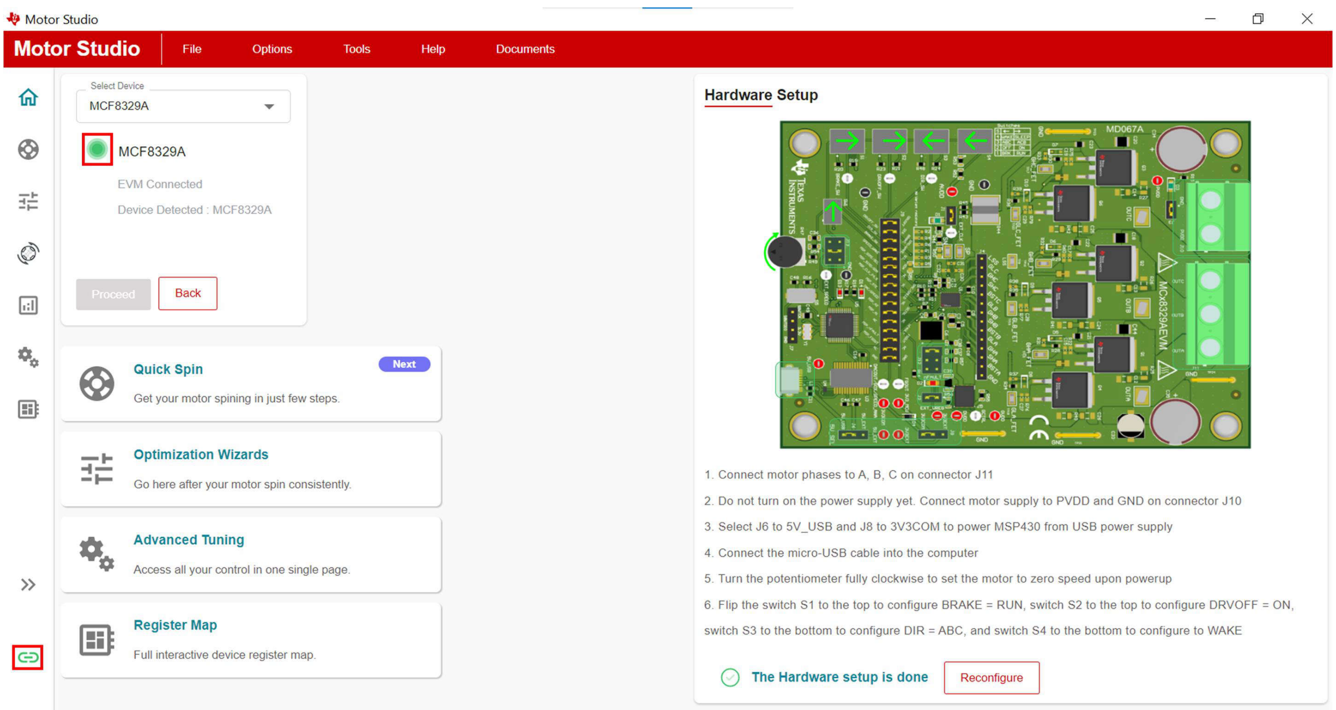

Figure 4-2. EVM connected Indicators

Note

If the GUI is not able to connect to the GUI after a minute, disconnect the EVM from the PC, restart the Motor Studio GUI. After Motor Studio has started again, reconnect the EVM to the PC.

---

5 Spinning into Closed Loop

This section provides standardized steps to tune the MCF8329A's settings so that the motor can successfully spin-up and enter closed loop control.

The general steps to tune the MCF8329A's registers so the motor can spin-up and enter closed loop control are outlined in #none##none##none#.

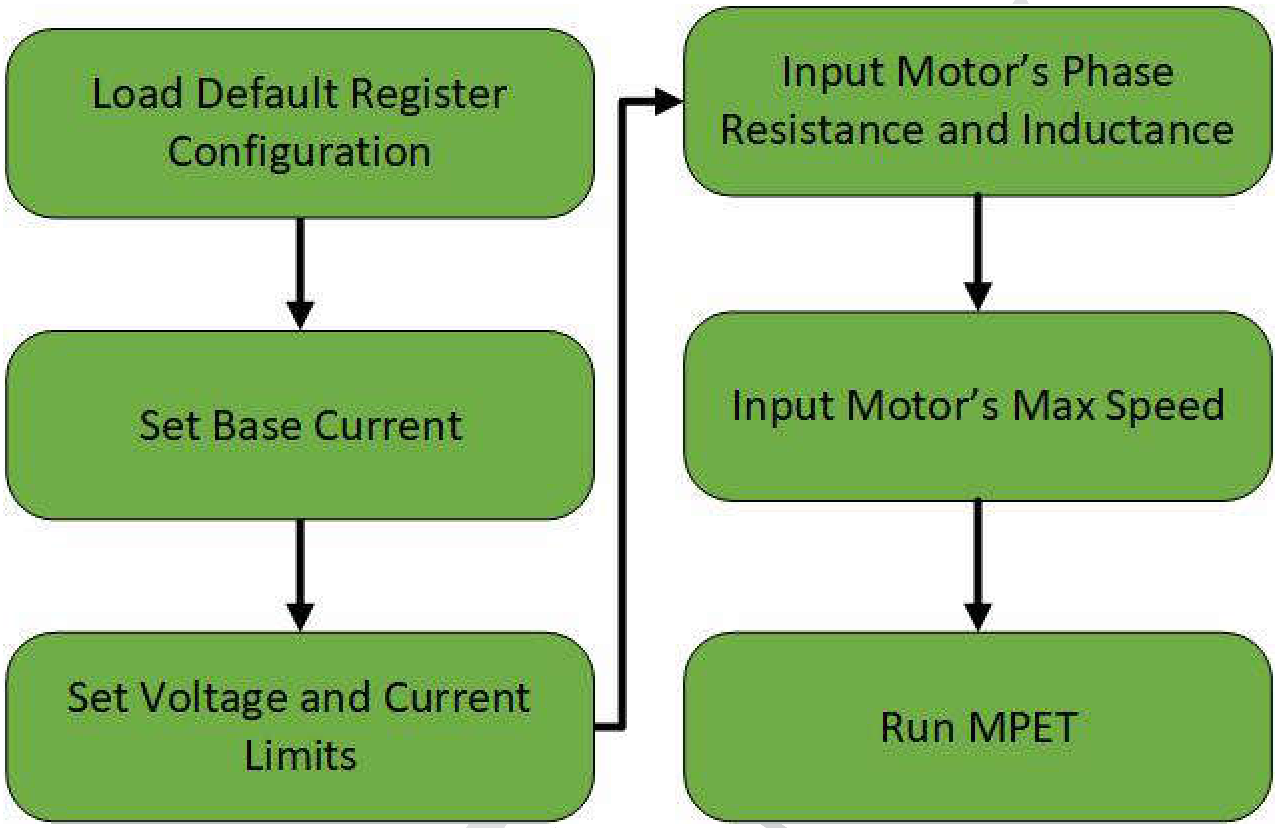

Figure 5-1. Essential Controls Flow Chart

5.1 Essential Configuration

5.1.1 Loading Recommended Default Values

Select the *Quick Spin* option on the Motor Studio Home page or on the menu on the left side of the window. Using the *Select Preset* drop down menu in the *Load Default* section select a register configuration which is most similar to the application use case or the *Default MCF8329A Registers* option. After selecting the desired register configuration, click on the *Load Preset Values* button.

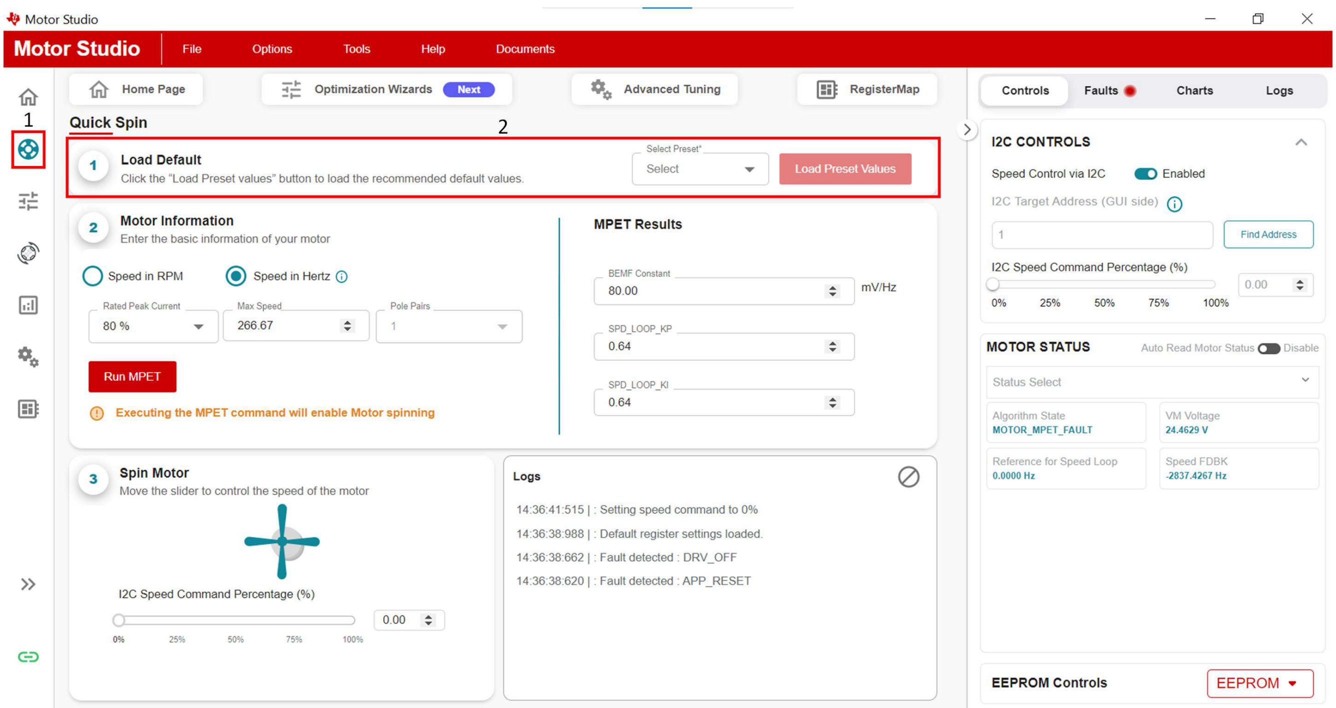

Figure 5-2. Load Default Register Configuration

5.1.2 Setting Base Current

Using the shunt resistor value and CSA gain value determined in Section 3.1.4, use equation 1 to calculate the max measurable current of the internal CSA.

$$1 \frac{1.5}{RSENSE \times CSA\_GAIN} \times \frac{32768}{1200} \quad (1)$$

After converting the result from equation 1 into a hexadecimal value, input the result into the BASE_CURRENT bit field in register GD_CONFIG2 using the *Register Map* page.

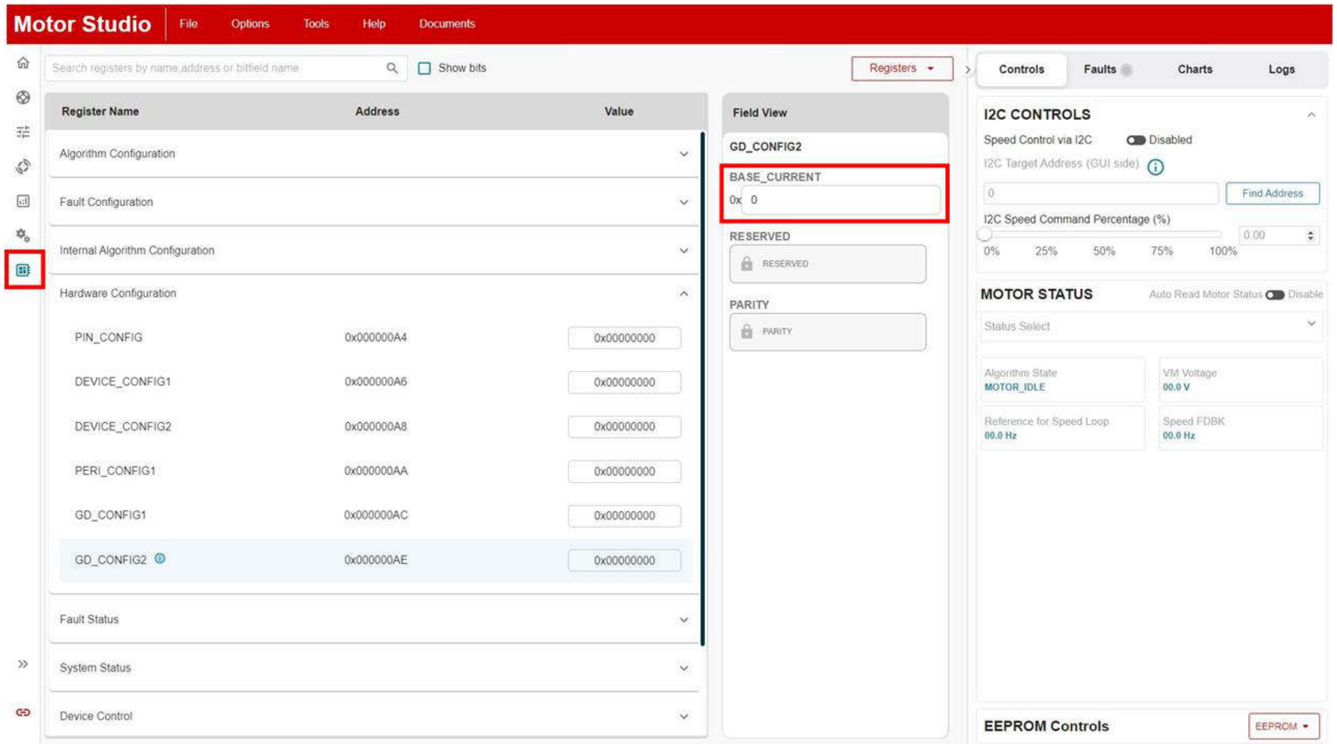

Figure 5-3. BASE_CURRENT Bit Field

5.1.3 Setting Current Limits

All the current limits within the MCF8329A are set as a percent of the value programmed into the BASE_CURRENT bit field. For example, if BASE_CURRENT is set to 37.5A and ILIMIT is set to 50%, then the current limit set by ILIMIT will be 18.75A.

HW_LOCK_ILIMIT and LOCK_ILIMIT are configurable current limits intended to protect the system from damage. It is recommended to set these limits to two times higher than the motors rated peak phase current. If the motors rated peak phase current falls between two adjacent limit settings in the configuration, choose the higher of the two settings.

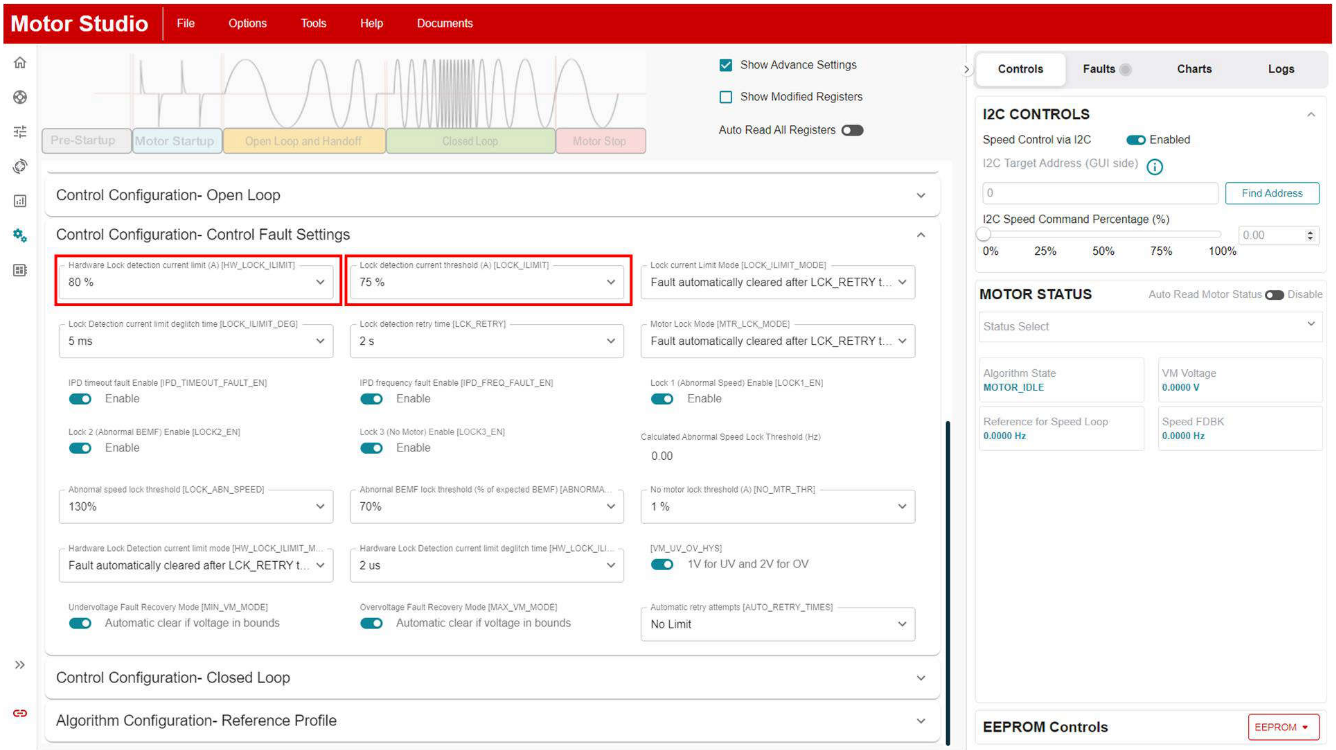

Figure 5-4. Current Protection Limits

ILIMIT, OL_ILIMIT, ALIGN_OR_SLOW_CURRENT_ILIMIT, and IPD_CURR_THR are the max current that will be used by the motor driver during the various stages of motor operation. It is recommended to set these values to less than or equal to the rated max phase current of the motor.

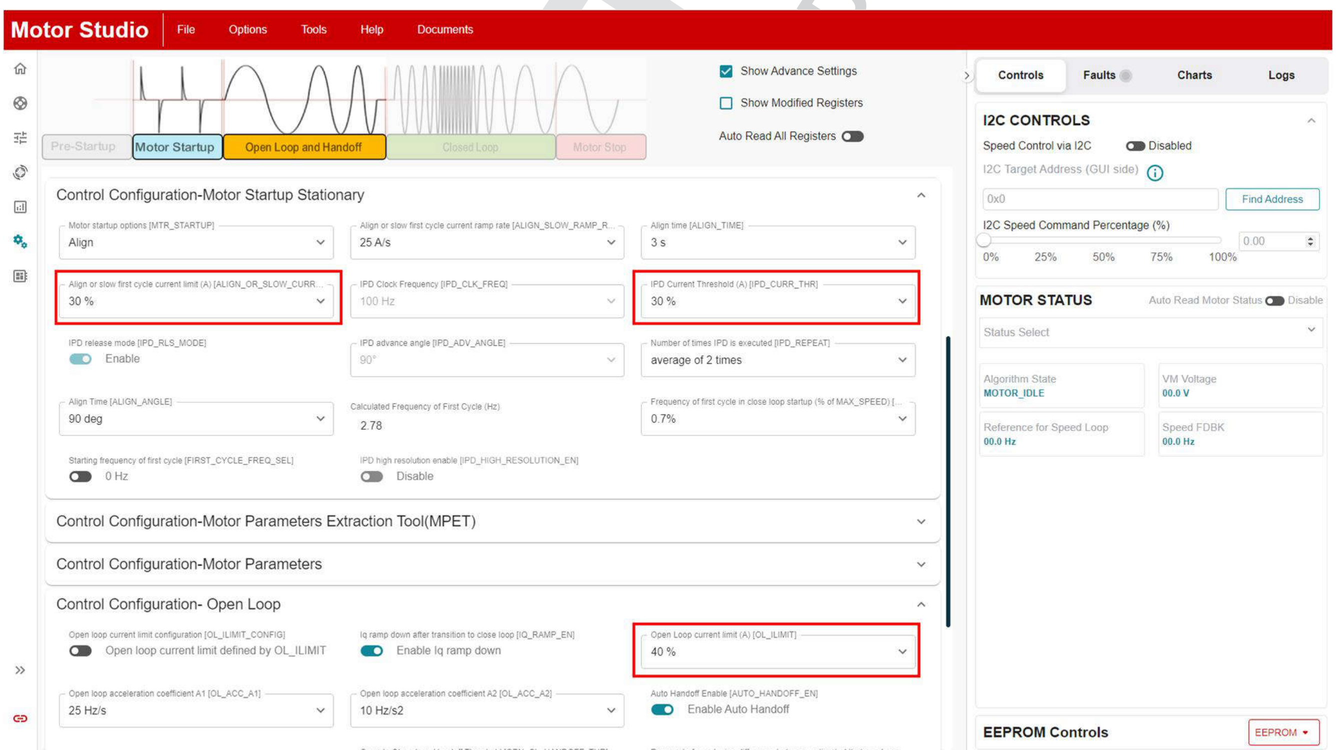

Figure 5-5. OL_ILIMIT, ALIGN_OR_SLOW_CURRENT_ILIMIT, and IPD_CURR_THR Current Limits

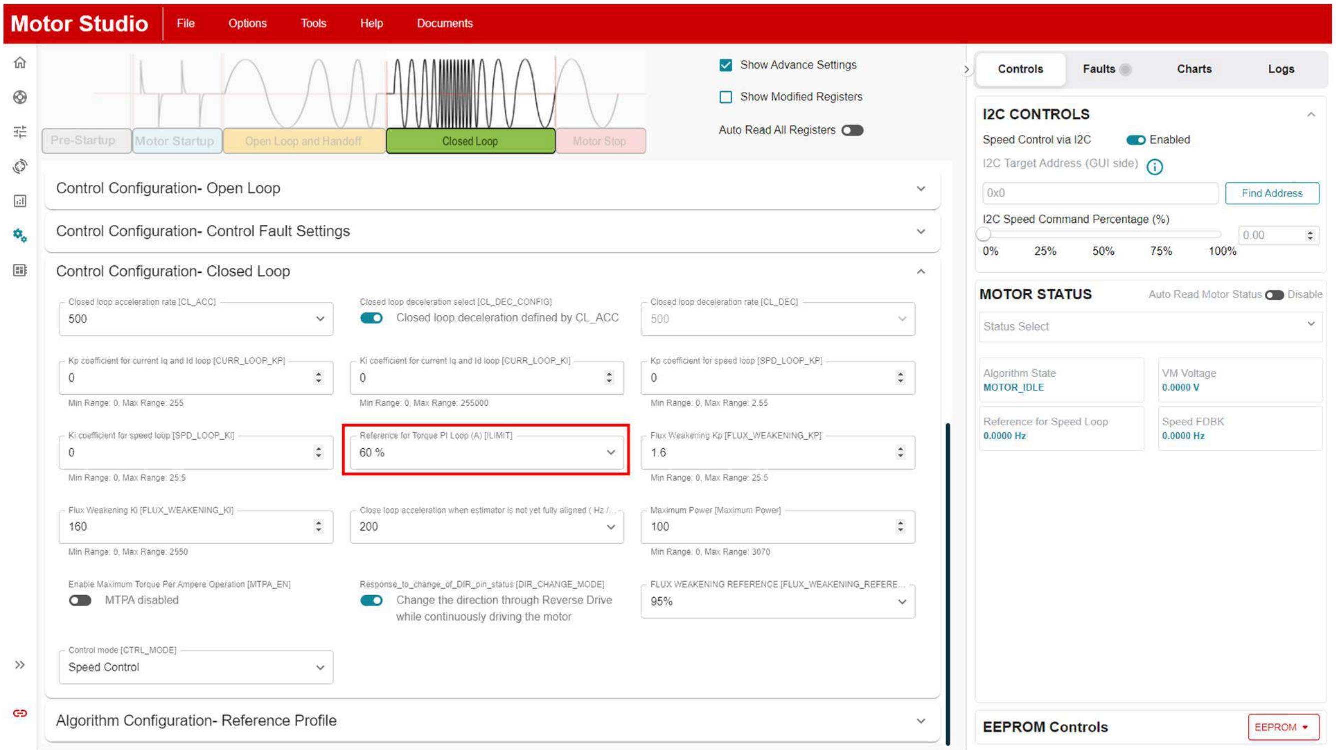

Figure 5-6. ILIMIT Current Limit

5.1.4 Setting Voltage Limits

BUS_VOLT is used to normalize the modulation algorithm that will be used by the MCF8329A. To improve the resolution of the modulation algorithm at lower motor voltages set the BUS_VOLT to the closest value that is still greater than the expected DC bus voltage or phase voltage. Make sure to account for voltage spikes on the phase node when determining the max expected voltage value.

In applications where the motor voltage cannot go above a certain level MAX_VM_MOTOR can be used to set the desired voltage limit.

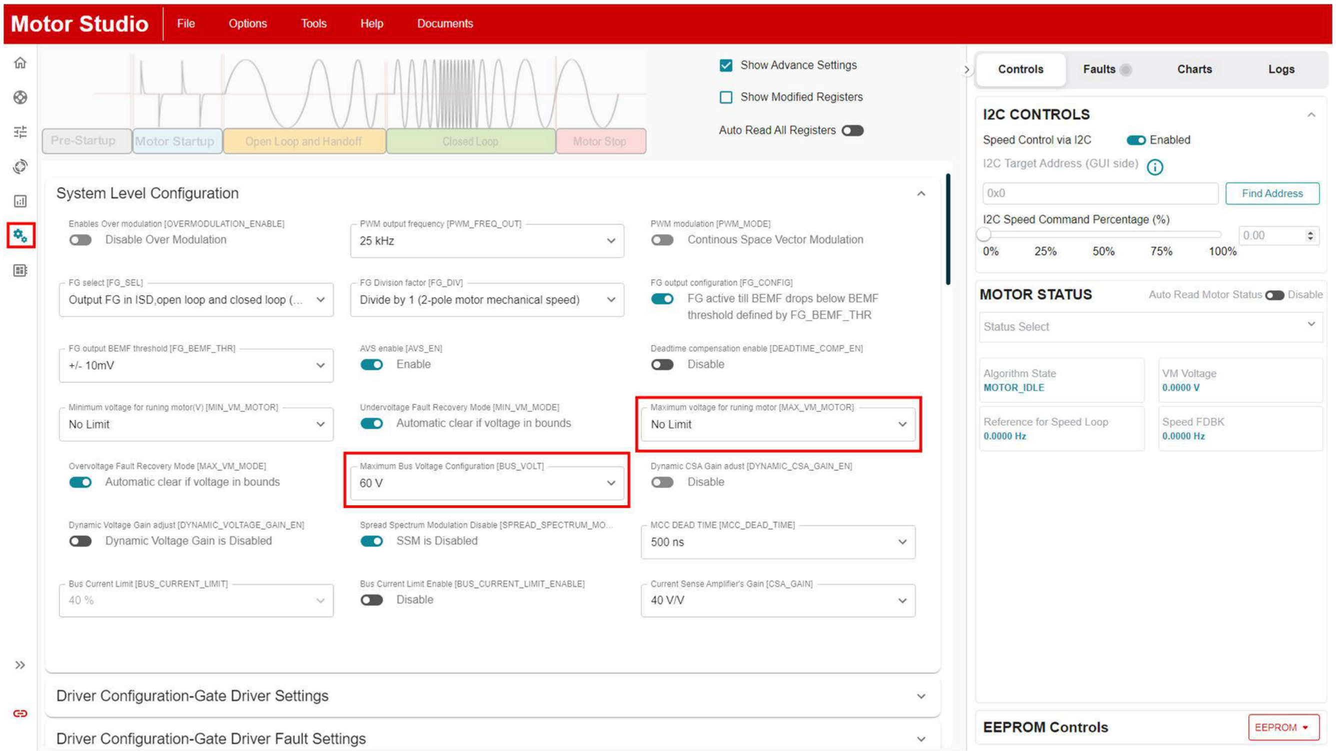

Figure 5-7. Voltage Limits

5.1.5 Input the Motor's Phase Resistance and Inductance

Using the instructions in the motor parameters FAQ, find the motors phase resistance and inductance. Once these values are found, input the phase resistance into the *Motor Phase Resistance* box and the phase inductance into both the *Lq* and *Ld* boxes in the *Motor Parameters* tab on the *Advanced Tuning* page.

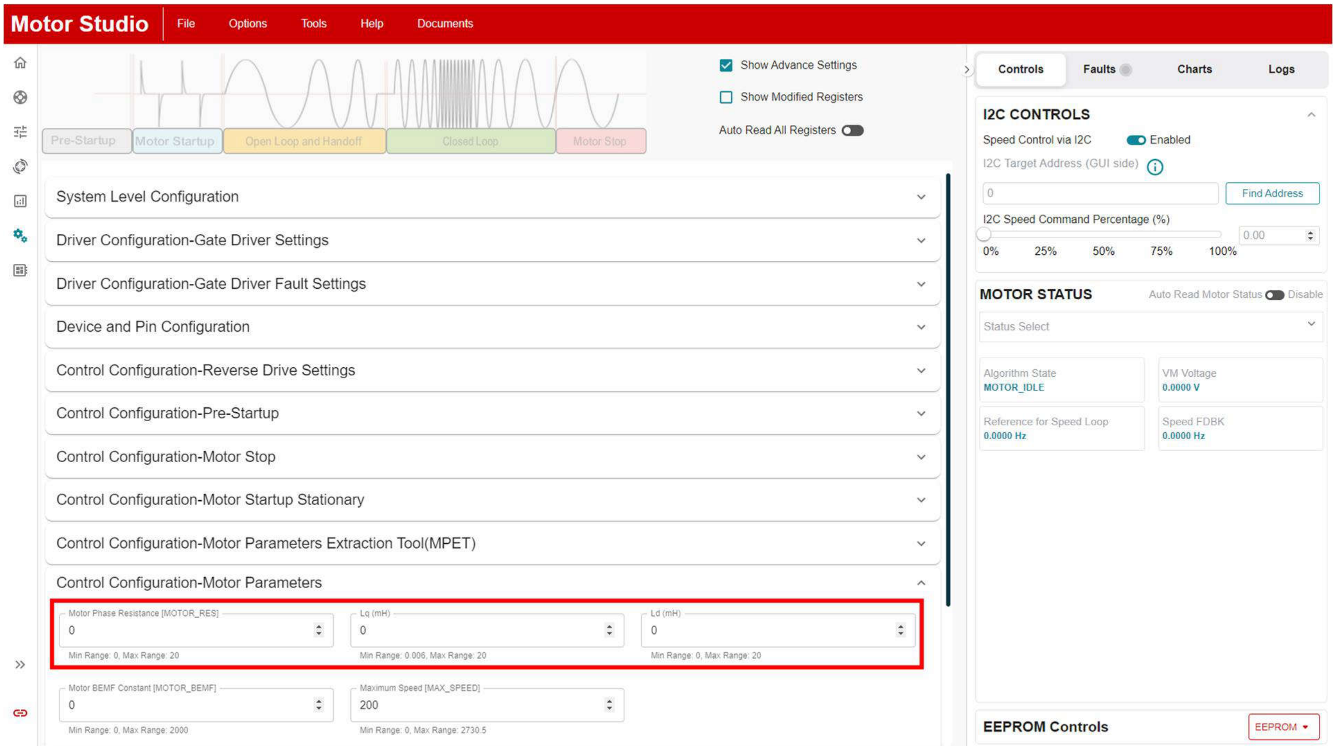

Figure 5-8. Motor Resistance and Inductance

5.1.6 Maximum Electrical Speed (Hz)

Go to the *Motor Information* Section of the GUI on the *Quick Spin* tab and follow the steps below to set the motor's max speed:

1. Select *Speed in RPM* or *Speed in Hz* depending on the unit of speed provided by the motor's datasheet.
2. Input the speed in the Max Speed box. If inputting a speed in RPM, also input the number of pole pairs the motor has using the *Pole Pairs* box.

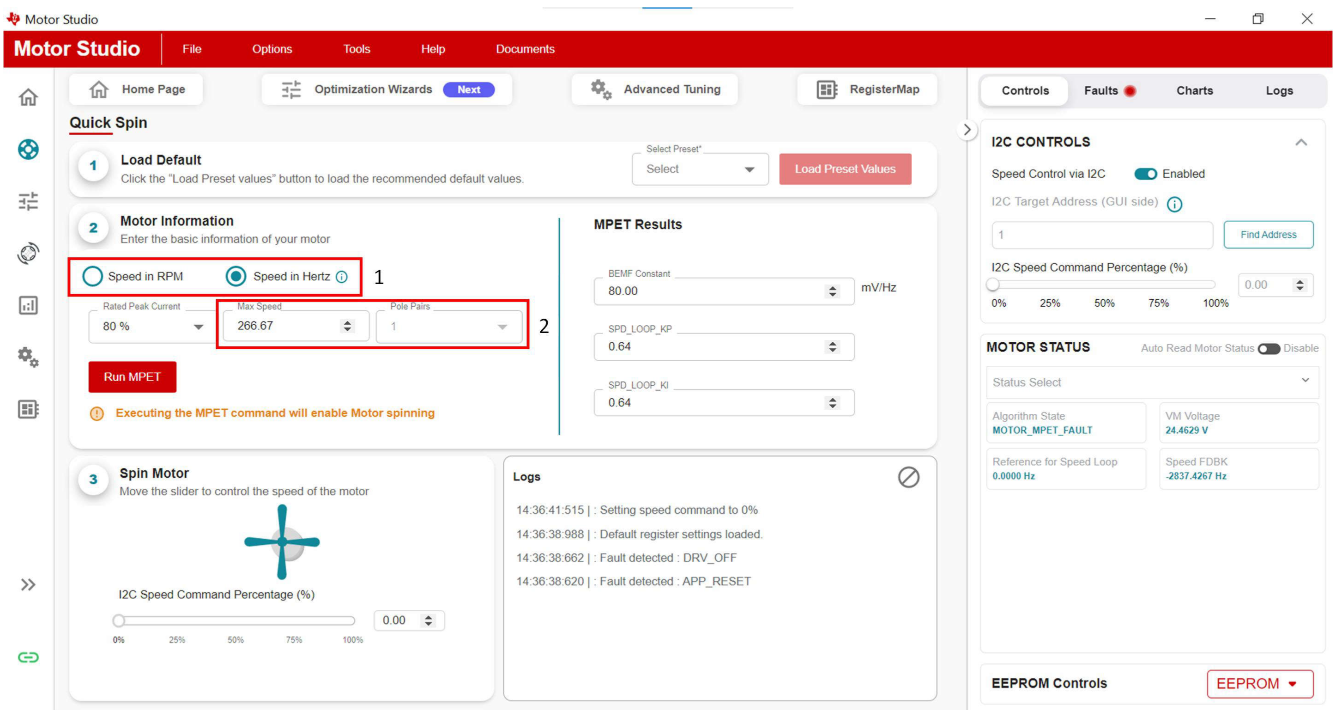

Figure 5-9. Motor Max Speed

Note

Determining number of motor poles without a motor datasheet:

1. Use a lab power supply and make sure it's current limit is set to less than the motor rated current. Do not turn on the supply
2. Connect V+ of the supply to phase A and V- of the supply to phase B of the motor. Any 2 of the 3 phases can be chosen at random if they are not labeled
3. Turn on supply. The rotor should have settled at one position with the injecting current.
4. Manually rotate the rotor until rotor snaps to another settle position. It will have several settle-down positions around one mechanical cycle
5. Count the number of settle-down positions for one fully mechanical cycle, which is the number of pole pairs. Multiplying by two calculates the number of poles.

Be careful of gearing systems within a motor. The gear ratio will determine how many rotor revolutions correlate to the shaft's mechanical revolution.

5.1.7 Run MPET to Identify Motor Parameters

Once all the other settings covered in the Section 5.1 section are set, the MPET algorithm within the MCF8329A can be used to measure the BEMF constant and speed loop gains needed to spin the motor in closed loop. Before running MPET, go to the *MPET* page and check that the IPD and open loop current limits are set according to the instructions in Section 5.1.3. Next, enable the *Measure Motor BEMF* Constant and *Measure Motor Mechanical Parameter* switches and click the *Run MPET* button. The motor should begin to spin, after the motor stos spinning MPET has completed measuring. Click the *Write MPET Results To Shadow Registers* button to use the results measured by MPET.

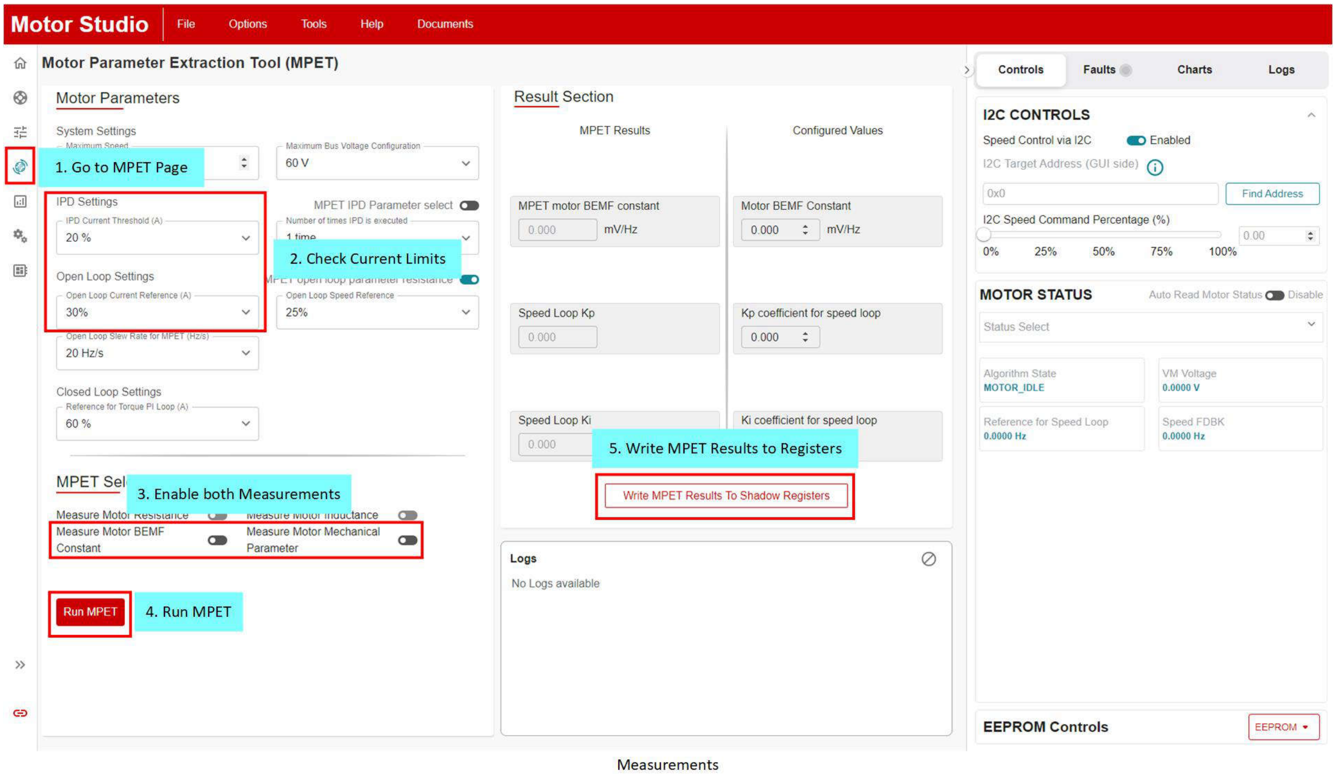

Figure 5-10. How to Run MPET

Note

If a fault is reported or MPET cannot measure the BEMF constant, go to Section 7 for assistance.

5.1.7.1 Skipping MPET Measurements

MPET measurement of the BEMF constant or speed loop gains can be skipped by making sure the register for the parameter is filled with a non-zero value and the measurement switch is disabled.

To disable BEMF constant measurement:

1. Fill in the Motor BEMF constant with a non-zero value, preferably with the value from the motor datasheet or a hand measured value.
2. Disable the Measure Motor BEMF Constant switch

To disable measurement of the speed loop gains:

1. Fill in the Kp and Ki coefficient for speed loop with non-zero values.
2. Disable the Measure Motor Mechanical Parameter

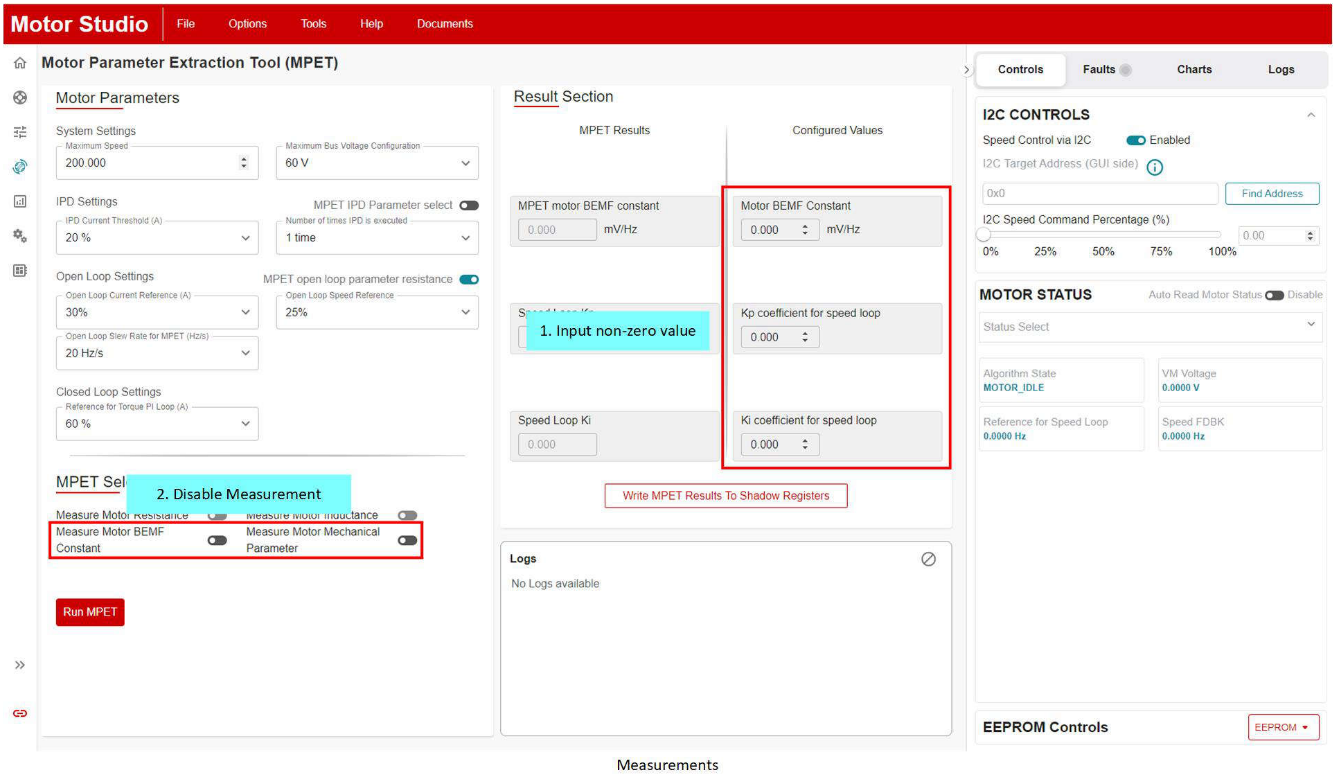

Figure 5-11. How to Skip MPET

5.2 Testing for Successful Startup into Closed Loop

1. Apply a nonzero speed command using the slider or text box within the *Spin Motor* section of the Quick Spin page. Once a speed command is provided, the motor should begin to spin and accelerate until the motor reaches the target speed.
2. After the motor stops accelerating check that the values in *Reference for Speed Loop* and *Speed FDBK* under the *Motor* Status section are close to the same value.
3. Check the color of the colored circle next to the *Faults* tab. If the circle is red in color, go to Section 7 and follow the debug steps for the fault reported
4. Once the motor is able to spin into closed loop and not trigger any faults, stop the motor and save the register configuration to a json file by clicking on *File -> Save Registers*. In the window that pops up, select *Json File* and click on the *Save* Button.
5. To have configuration for the registers covered in section 7.7 of the MCF8329A datasheet load when the device powers-up these register values can be loaded into EEPROM. To write the configured register values to EEPROM, click the *EEPROM* drop down located at the bottom right of Motor Studio and select the *Write To EEPROM* option. Click the *Yes* button in the window that pops up.

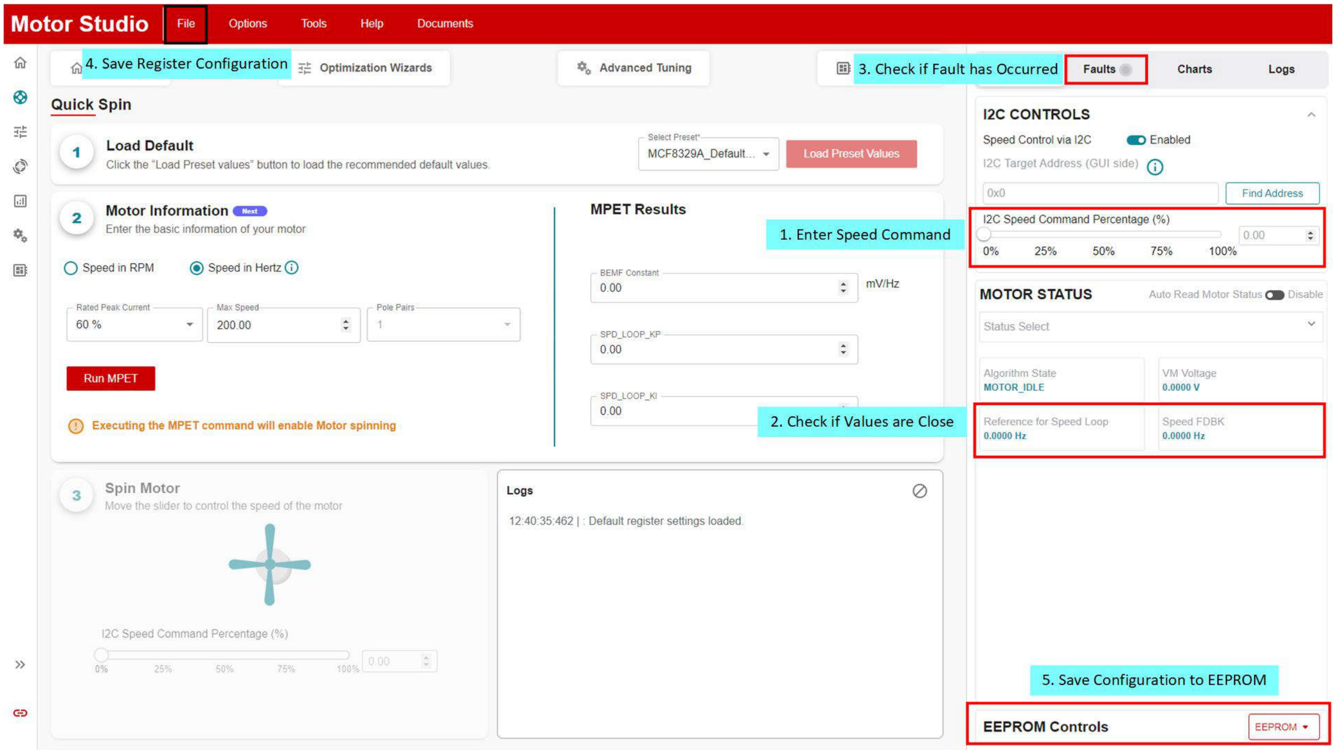

Figure 5-12. Closed Loop Spin Test Steps

6 Basic Controls

This section provides tuning guidance for optimizing the motors performance for many use-case needs.

Note

It s expected to skip the subsection use-cases and scenarios that do not apply to the system or end equipment.

6.1 Speed Input Mode

The MCF8329A offers four options to control the speed of the motor: PWM, frequency, Analog, and I2C. The desired speed mode can be set by changing the value of the SPEED_MODE register on the *Advanced Tuning* page. A description of how to configure these control methods is provided in section 7.3.7 of the MCF8329A datasheet.

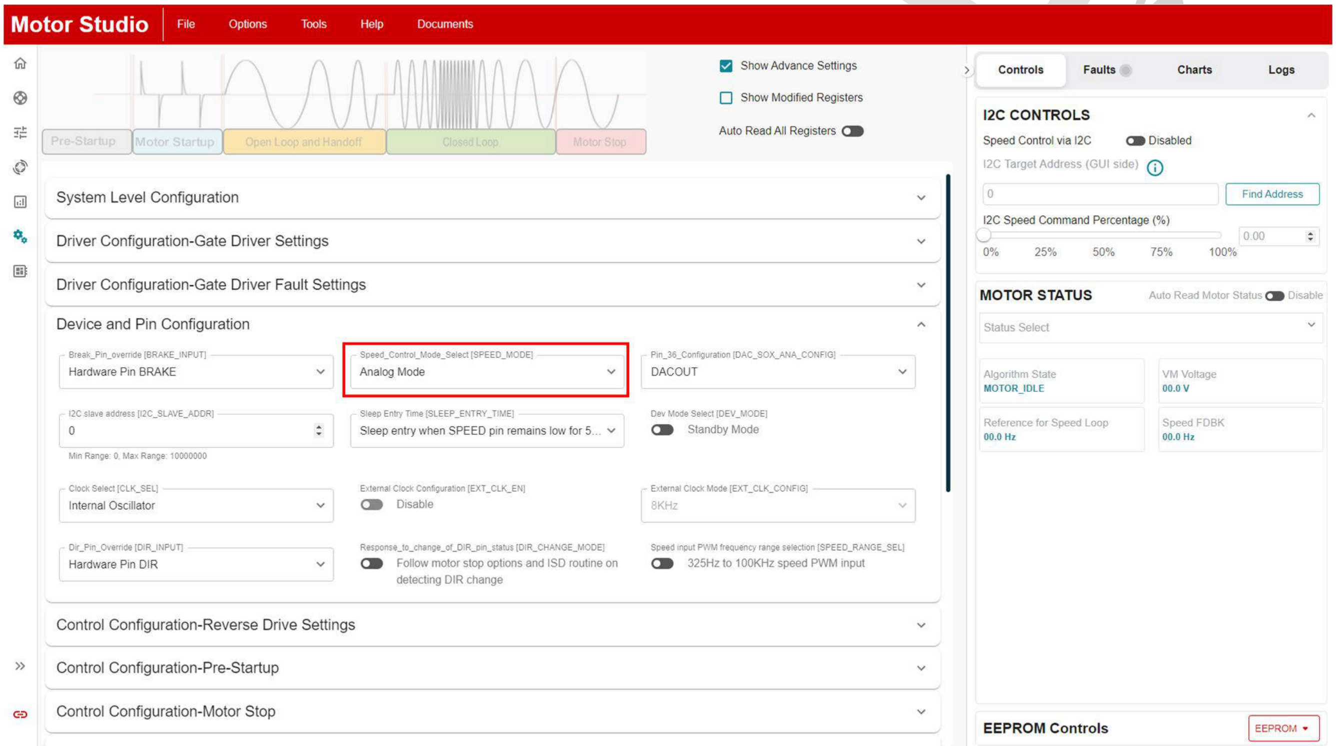

Figure 6-1. Speed Mode Selection

If I^2C speed input is chosen, flip SW1 away from the other switches, see Figure 6-2, this will provide a the wake switch signal to the SPEED/WAKE pin to keep the MCF8329A out of sleep/standby mode. If a speed mode other than I2C is being used, flip the switch to the opposite as what is shown in Figure 6-2 to connect the speed pin to J13. See Table 2-2 of the MCF8329EVM User's Guide for information on how to set J13.

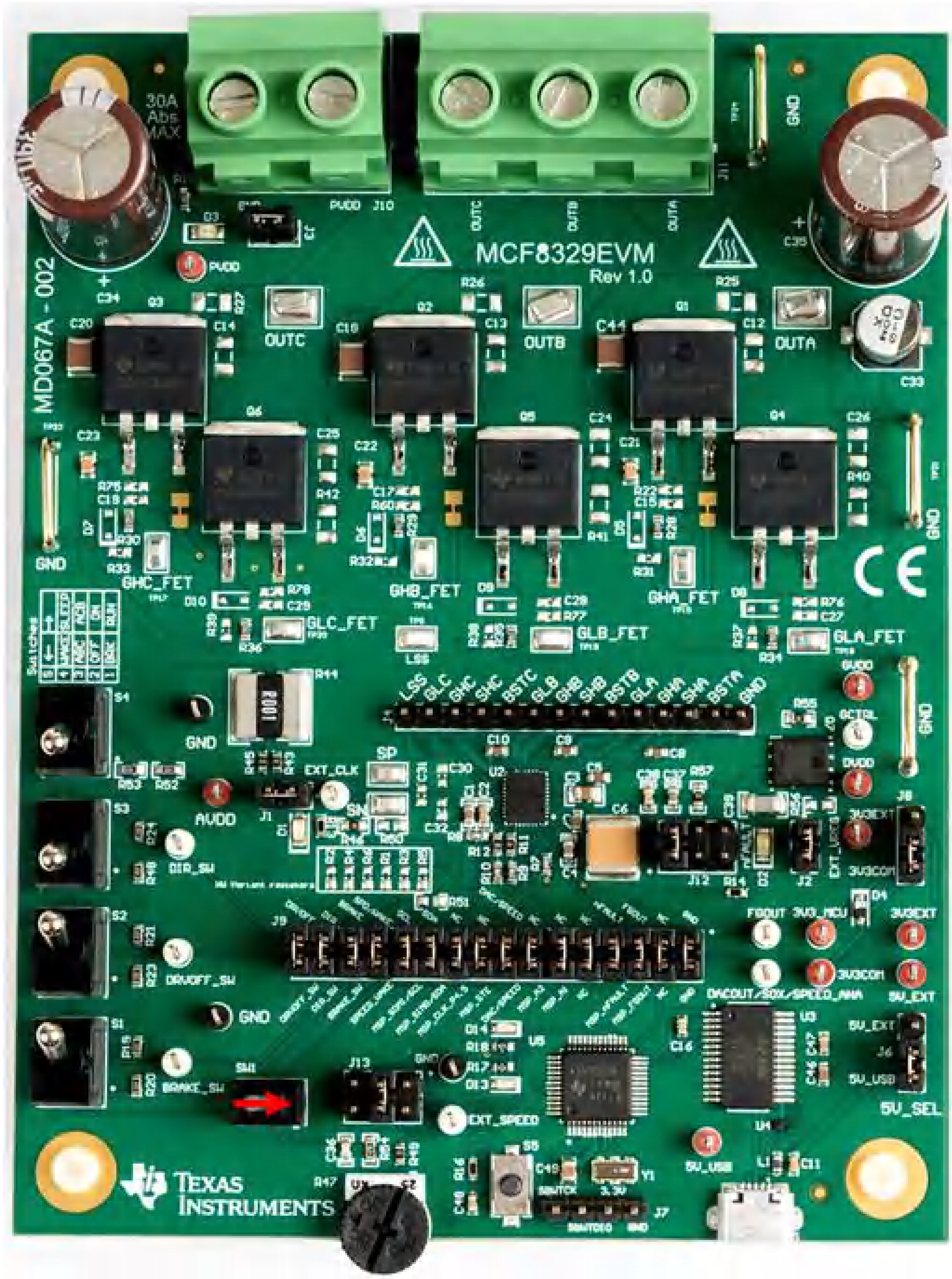

Figure 6-2. SW1 Position for I2C Speed Mode

6.2 Preventing Back Spin of Rotor During Startup

Option 1: Initial Position Detection (IPD)

Step 1: Go to *Optimal Startup* in the *Optimization Wizards* page, select IPD, and click the *Next* button for instructions to set up the IPD startup method.

Step 2: Set the IPD Advance Angle [IPD_ADV_ANGLE] to 90º to get maximum startup torque. If there is sudden jerk observed during startup, then it is recommended to reduce the angle to 60º or 30º for a smoother startup.

Note

Device triggers IPD timeout fault [IPD_T1_FAULT] for motors with very high inductance, or if the motor is not connected. If this fault gets triggered, it is recommended to check if motor is connected to the device.

Device triggers IPD Frequency fault [IPD_FREQ_FAULT] if the IPD clock frequency is set too high. If this fault gets triggered, it is recommended to decrease the IPD Clock value [IPD_CLK_FREQ].

Option 2: Slow First Cycle

Step 1: Go to *Optimal Startup* in the *Optimization Wizards* page, select slow first cycle, and click the *Next* button for instructions to set up the slow first cycle startup method.

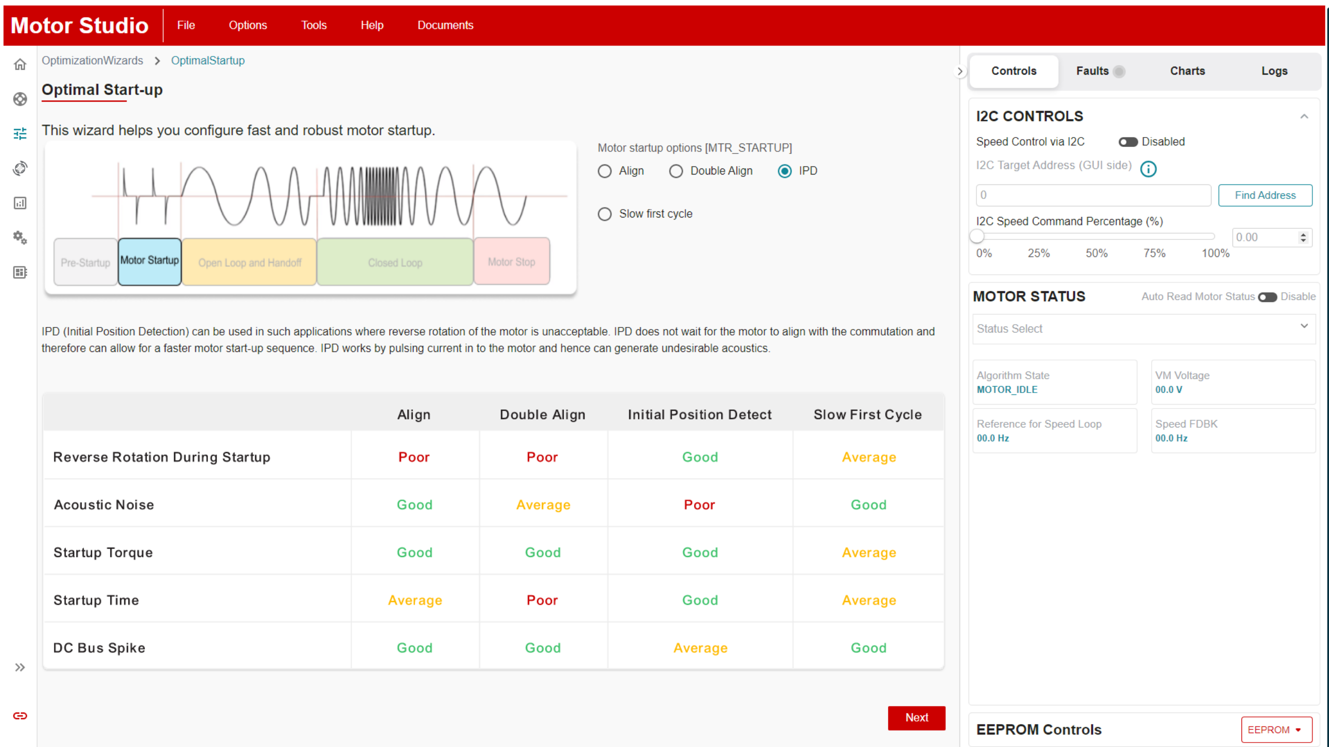

Figure 6-3. Optimal Startup Page

6.3 Faster Startup Timing

Option 1: Initial Position Detection (IPD)

Step 1: Go to *Optimal Startup* in the *Optimization Wizards* page, select IPD, and click the *Next* button.

Step 2: Increase IPD current threshold (A) [IPD_CURR_THR] to the rated current of the motor.

Step 3: Increase IPD clock value [IPD_CLK_FREQ] to a higher frequency up to a value where the device does not trigger IPD frequency fault.

Step 4: Set IPD repeating times [IPD_REPEAT] to 1 time.

Step 5: Set Open loop current limit configuration [OL_ILIMIT_CONFIG] to Open loop current limit defined by ILIMIT.

Step 6: Increase Open loop acceleration coefficient A1 [OL_ACC_A1] and Open loop acceleration coefficient A2 [OL_ACC_A2].

Note

A1 and A2 can be increased until open loop current reaches Lock detection current threshold [LOCK_ILIMIT]. Open loop current can be measured using oscilloscope.

Increasing Open loop acceleration coefficient A1 [OL_ACC_A1] and Open loop acceleration coefficient A2 [OL_ACC_A2] might trigger LOCK_LIMIT fault. If this happens, reduce A1 and A2 until LOCK_LIMIT fault no longer triggers.

Step 7: For ultra-fast startup time (less than 100 ms) it is recommended to follow below steps.

- Disable auto-handoff [AUTO_HANDOFF].
- Configure open to closed loop handoff threshold [OPN_CL_HANDOFF_THR] to a value lesser than or equal to 20 Hz.

For startup times above 100ms, it is recommended to follow below steps.

- Enable auto-handoff [AUTO_HANDOFF].

Note

If Abnormal speed fault [ABN_SPEED] gets triggered, it is recommended to decrease open loop acceleration constants [OL_ACC_A1] and [OL_ACC_A2] and also retune IPD by increasing the IPD current threshold [IPD_CURR_THR] and IPD repeat times [IPD_REPEAT].

Step 8: Increase Closed loop acceleration rate [CL_ACC]

Note

LOCK_LIMIT fault handling:

Closed loop acceleration rate [CL_ACC] can be increased until closed loop current reaches Lock detection current threshold [LOCK_ILIMIT]. Closed loop current can be measured using oscilloscope. Increasing closed loop acceleration rate [CL_ACC] might trigger LOCK_LIMIT. If this happens, reduce closed loop acceleration rate [CL_ACC] until no longer triggers.

Option 2: Align

Step 1: Go to *Optimal Startup* in the *Optimization Wizards* page, select Align, and click the *Next* button for instructions to set up the Align startup method.

Step 2: Configure align time [ALIGN_TIME] to 10 ms.

Step 3: Follow Step 6 to Step 9 in Option 1.

shows FG, phase current and motor electrical speed waveform. Motor takes 50 ms to reach target speed from zero speed.

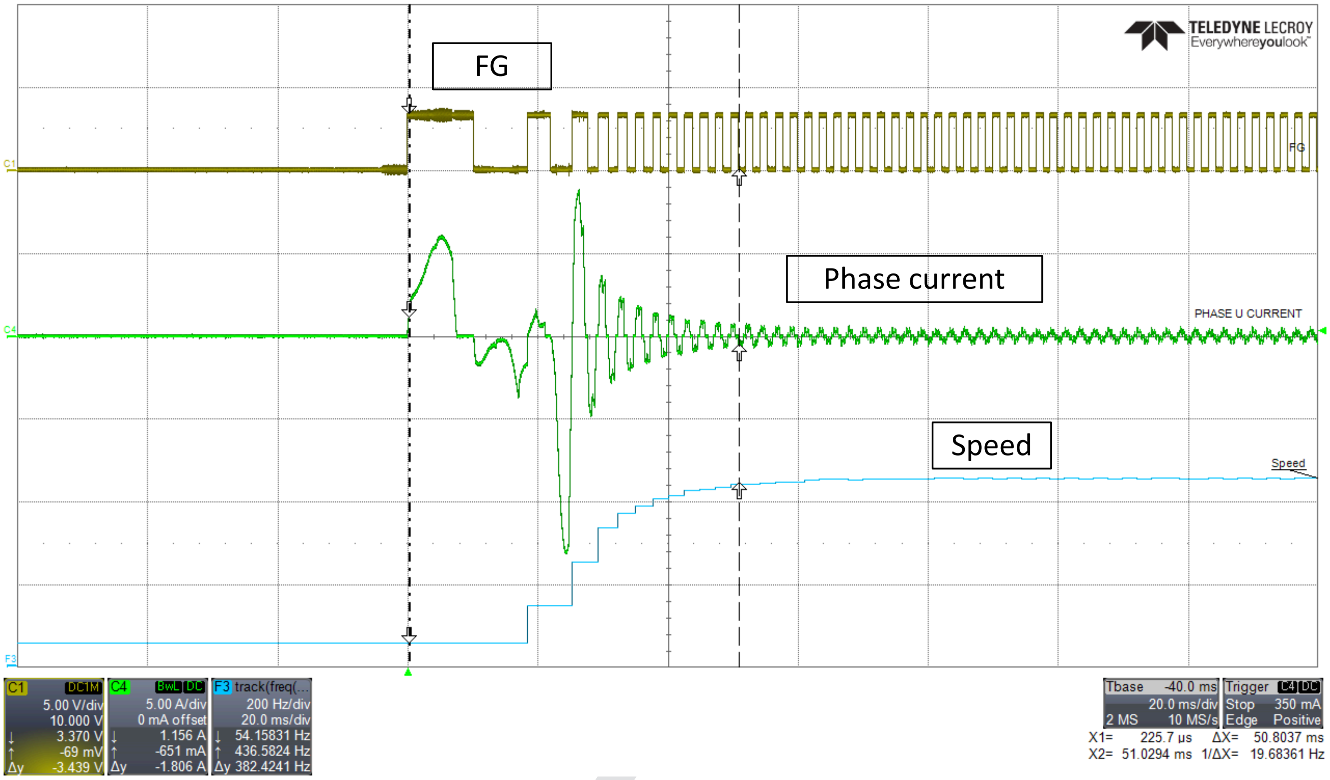

Figure 6-4. Phase current, FG and motor speed - Faster startup time

---

Note

If Abnormal speed fault [ABN_SPEED] or Loss of sync [LOSS_OF_SYNC] fault gets triggered, it is recommended to follow below debug steps.

1. Select Double align as the motor startup method in [MTR_STARTUP].
2. Increase align time [ALIGN_TIME].
3. Configure align current threshold [ALIGN_OR_SLOW_CURRENT_ILIMIT] to 50% of ILIMIT.
4. Configure First cycle frequency select [FIRST_CYCLE_FREQ_SEL] to 0.

6.4 Improving Speed Regulation

For applications that require better speed regulation, it is recommended to tune Speed loop PI controllers [SPD_LOOP_KP] and [SPD_LOOP_KI]. Kp coefficient of speed loop [SPD_LOOP_KP] controls the settling time and speed overshoots. Ki coefficient of Speed loop [SPD_LOOP_KI] controls speed overshoot and ensures regulation of speed at set value and drives the error to zero. Speed loop PI controller gains can either be auto-tuned by MCF8329A or tuned manually.

Auto Tuning: MCF8329A auto calculates the Speed loop PI controller gains when [SPD_LOOP_KP] and [SPD_LOOP_KI] are set to zero.

Manual Tuning: Follow below steps to tune Speed loop PI controller gains manually

Step 1: Set the control mode [CTRL_MODE] to modulation index control (11b)

Step 2: Issue non-zero speed command to start the motor (refer to Section 5.2, step 1 on how to issue non-zero speed command). Motor will spin in open loop.

Step 3: Allow the open loop current to settle down and then measure the peak open loop current.

Step 4: Stop the motor and set the control mode [CTRL_MODE] to current control.

Step 5: Slowly increase the speed command until the motor speed reaches the max speed. Note down the Iq_ref value being reported in the *IQ_REF_CLOSED_LOOP* register.

Step 6: Speed loop Kp [SPD_LOOP_KP] is calculated using Equation 2.

$$Speed\ loop\ Kp\ =\ \frac{Iq\ reference\ at\ maximum\ speed}{Maximum\ Electrical\ Speed\ in\ Hz} \tag{2}$$

Step 7: Speed loop Ki [SPD_LOOP_KI] is calculated using Equation 3.

$$Speed\ loop\ Ki = 0.1\ \times Speed\ loop\ Kp \tag{3}$$

Step 8: Stop the motor and set the control mode [CTRL_MODE] to speed control.

Note

Tuning speed loop Kp and Ki is experimental. If the above recommendation doesn't work, then we recommend to manually tune Speed loop Kp and Ki till the desired results are achieved.

6.5 Limiting and Regulating Supply Power

MCF8329A provides options to limit and regulate supply power. This feature can be utilized in battery powered motor driver applications such as cordless vacuum cleaners, power tools etc.

Follow below steps to limit supply power. In this mode, supply power is only limited to reference power and not actively regulated.

Step 1: Configure CTRL_MODE to power control (1b).

Step 2: Configure MAX_POWER. This sets the maximum power that MCF8329A can draw from the DC input supply at 100% duty command. For example, if MAX_POWER is configured to 25 W, MCF8329A draws 12.5 W from power supply at 50% duty command.

Step 3: The power control loop uses the same PI controller parameters as in the speed loop mode. Kp and Ki coefficients are configured through SPD_POWER_KP and SPD_POWER_KI. Tuning SPD_POWER_KP and SPD_POWER_KI is experimental. The recommendation is to manually tune both parameters until the desired results are achieved.

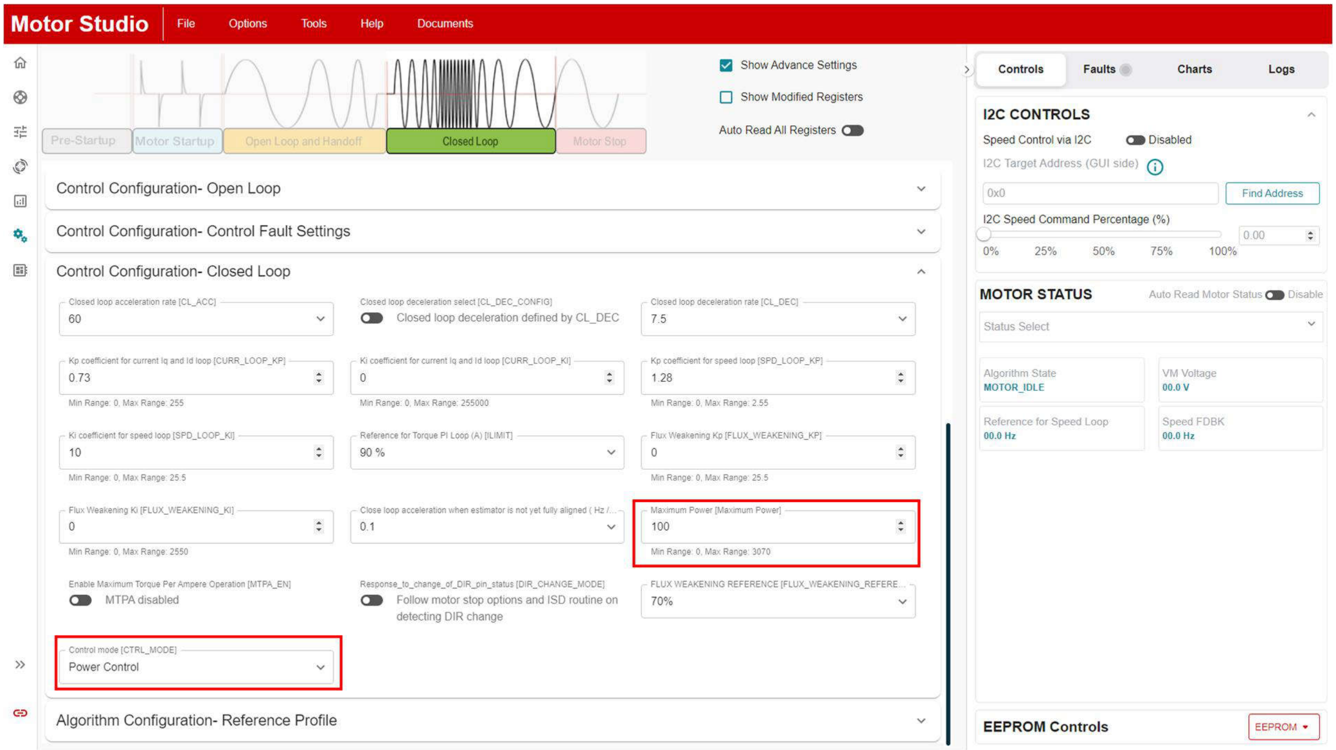

Figure 6-5. Power Control Settings

6.6 MTPA Tuning

Maximum torque per ampere (MTPA) is a feature in the MCF8329A to maximize the torque generated per ampere of current for salient motors. To enable MTPA, set MTPA_EN to 1b and set the SALIENCY_PERCENTAGE to a non-zero value by setting the Lq and Ld values if they are provided in the datasheet.

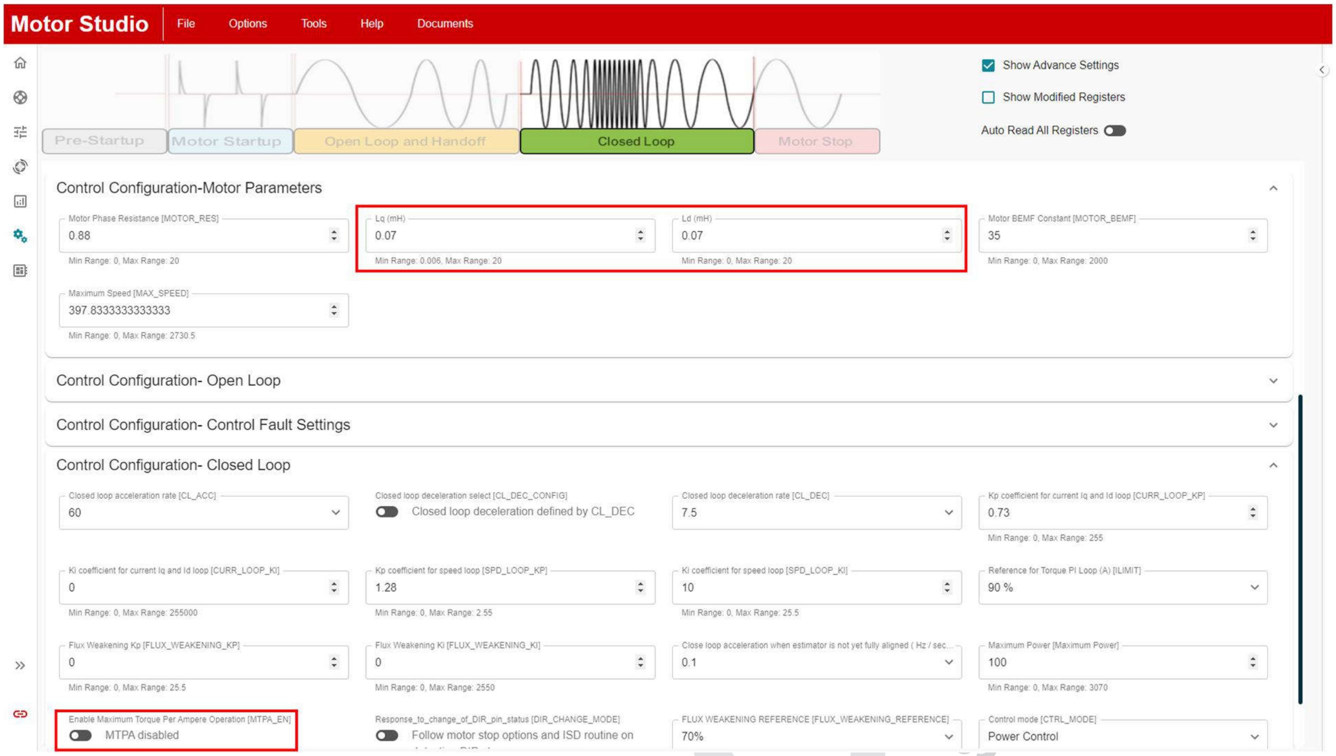

Figure 6-6. MTPA Settings

Note

If the motors Ld or saliency percentage is not known, the approximate SALIENCY_PERCENTAGE can be determined by following the steps below:

1. Set the SALIENCY_PERCENTAGE to 0x1h
2. Set the CTRL_MODE to Current Control mode
3. Provide a speed command.
4. While the motor is spinning, increment the SALIENCY_PERCENTAGE value by 1h until the motors speed begins to decrease.

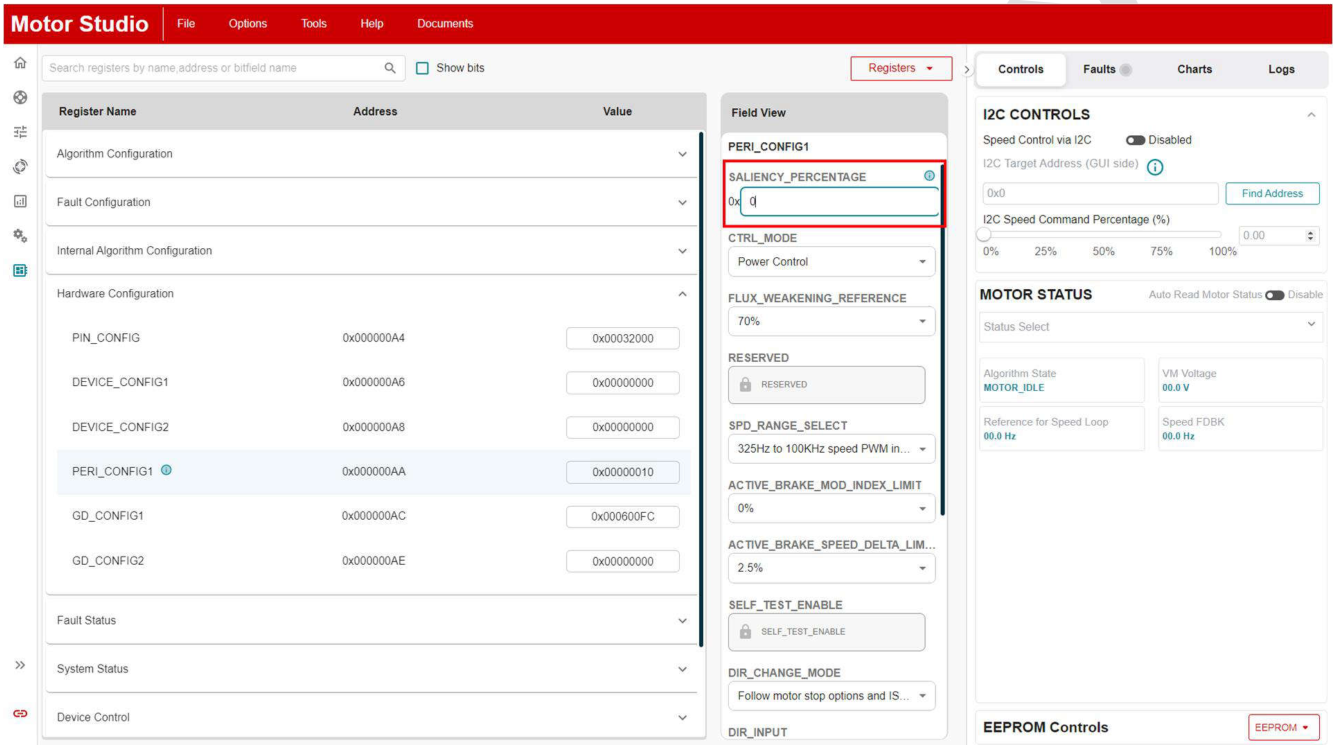

Figure 6-7. Saliency Register

6.7 Motor Studio Optimization Wizards

Refer to the *Optimization Wizards* page on Motor Studio for step-by-step guidance on configuring the MCF8329A for additional use-cases and optimization features.

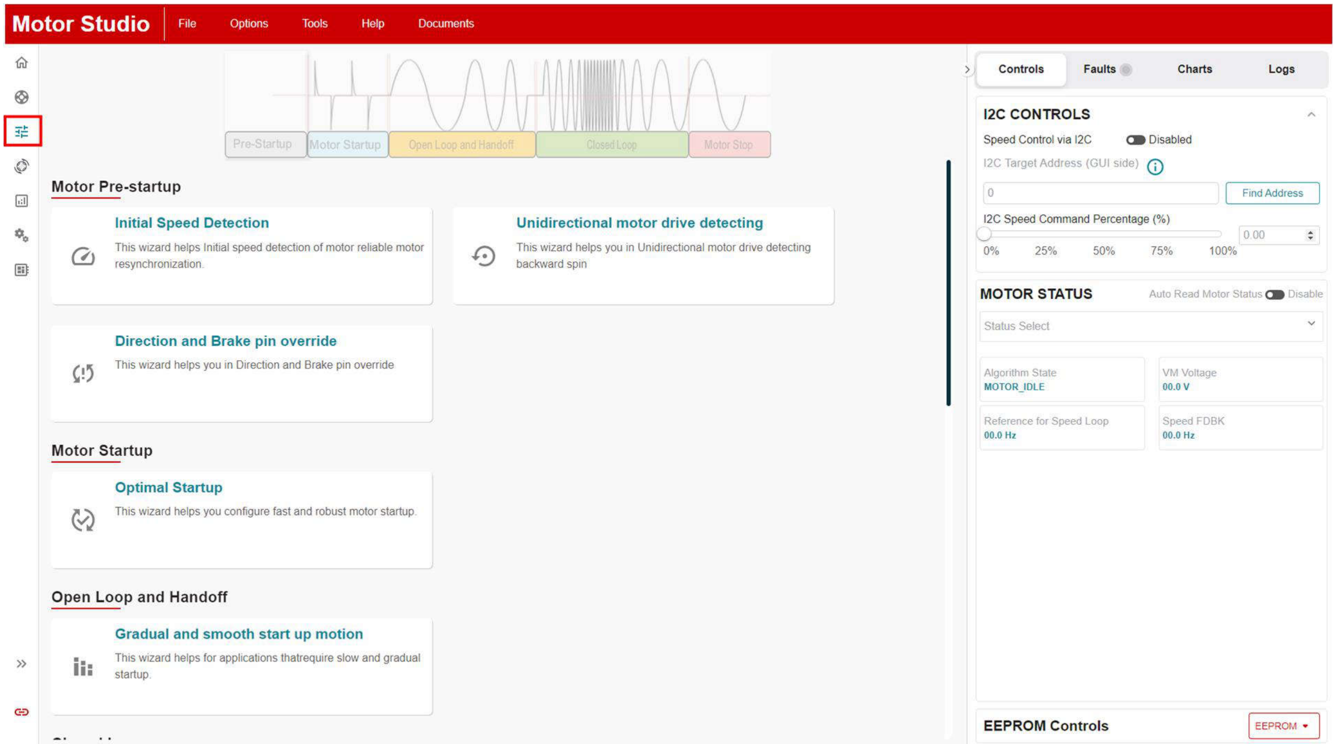

Figure 6-8. Optimization Wizards Page

7 Fault Handling

To see which fault has been reported by the MCF8329A, go to the *Faults* tab and check if any faults with red circles appear. If a fault is shown in this tab, refer to the section below which has a title similar to the reported fault.

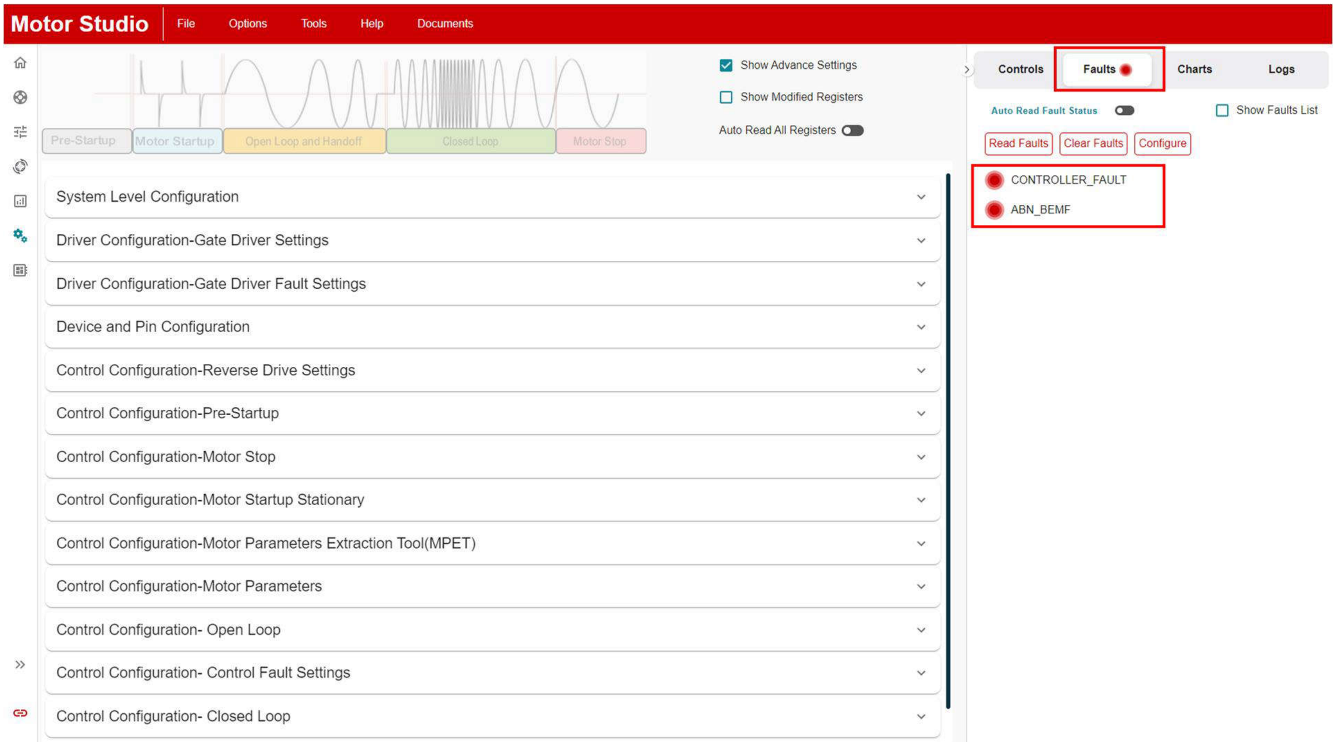

Figure 7-1. Faults Tab

7.1 MPET BEMF FAULT [MPET_BEMF_FAULT]

A MPET_BEMF_FAULT gets reported when the meased BEMF is less than the threshold set in STAT_DETECT_THR. If this fault is triggered, go the *MPET* page in Motor Studio and follow the suggestions below.

Step 1: Set the *MPET Open Loop Parameter Resistance* so the circle is on the right side

Step 2: Increase the *Open Loop Current Reference* value

Step 3: Decrease the *Open Loop Slew Rate for MPET* value

Step 4: If the fault still persists, refer to the Motor Parameters FAQ for instructions on how to obtain the motor's BEMF constant through the motor's datasheet for through hand measurement. Once the motor's BEMF constant value is found, input the BEMF constant value into the *Motor BEMF Constant* box in the *Configured Values* section on the *MPET* page.

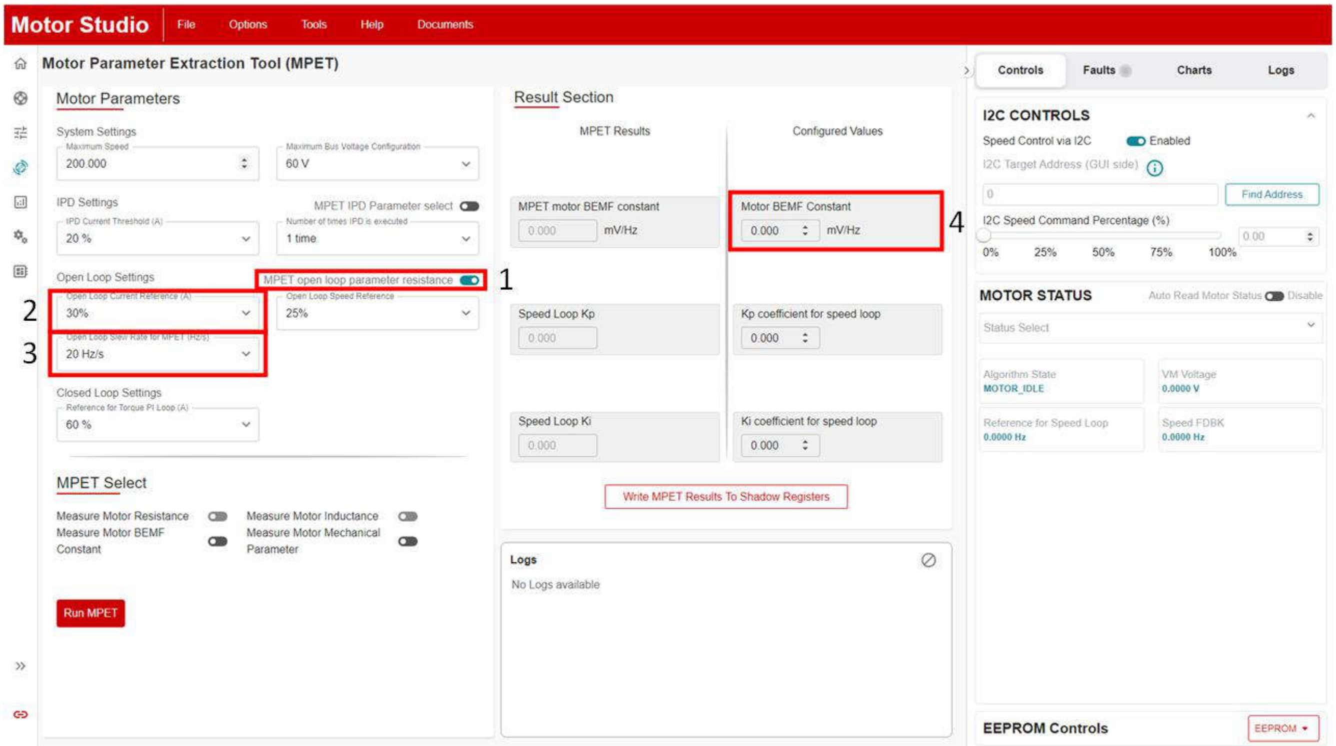

Figure 7-2. MPET_BEMF_FAULT

7.2 Abnormal BEMF Fault [ABN_BEMF]

This fault gets triggered when the difference between the estimated BEMF voltage exceeds the threshold set by ABNORMAL_BEMF_THR. If this fault is triggered, then please go the *Control Fault Settings* tab in the *Advanced Tuning* page in Motor Studio and follow the below suggestions.

Step 1: For applications with load dynamics (sudden change in load), it is recommended to set the Abnormal BEMF threshold to 70% to avoid triggering this fault.

Step 2: This fault can get triggered if the programmed BEMF constant is inaccurate. Follow steps recommended in step 4 of Section 7.1 to obtain accurate BEMF constant.

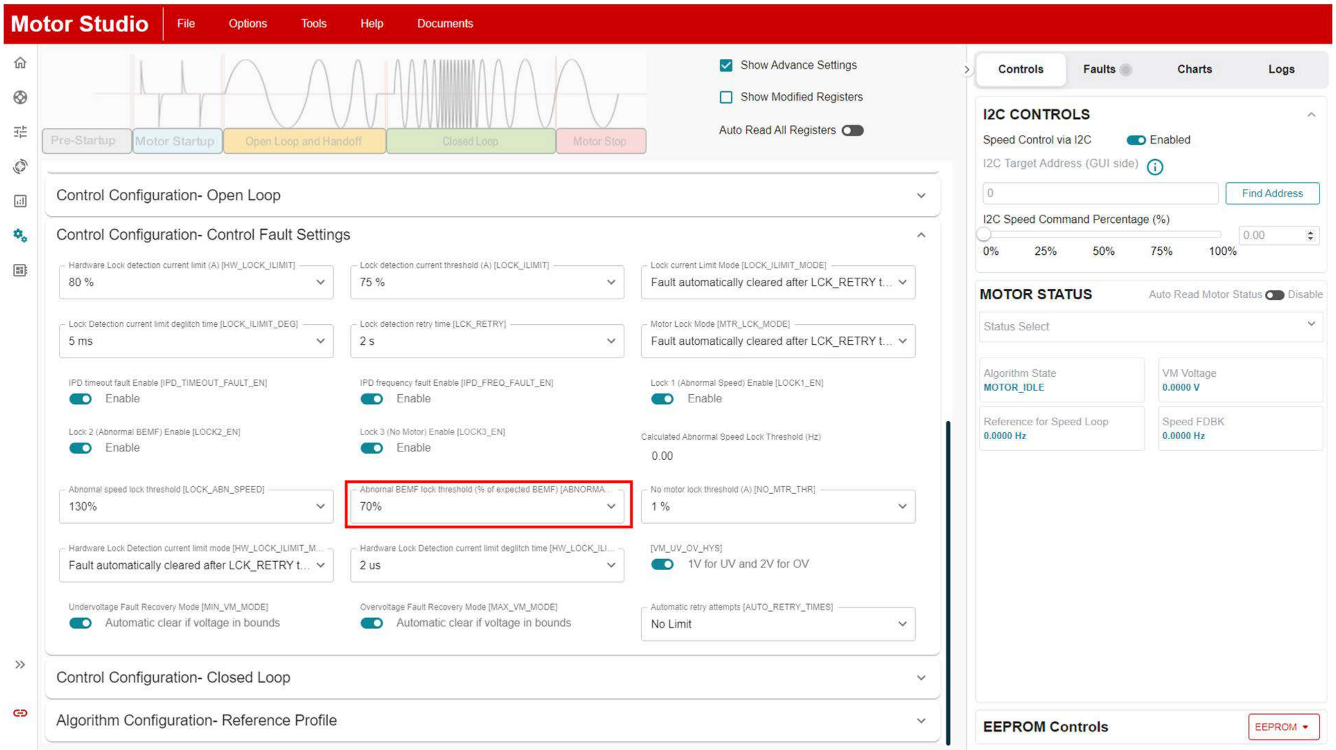

Figure 7-3. ABNORMAL_BEMF_THR

7.3 Lock Current Limit [LOCK_LIMIT]

This fault gets triggered when the phase current exceeds the LOCK_ILIMIT threshold. If this fault is triggered, check the motor datasheet for stall torque and load the motor below the stall torque specified in the datasheet. If the load torque is still within the stall torque, go to the *Control Fault Settings* tab in the *Advanced Tuning* page and increase the value of LOCK_ILIMIT.

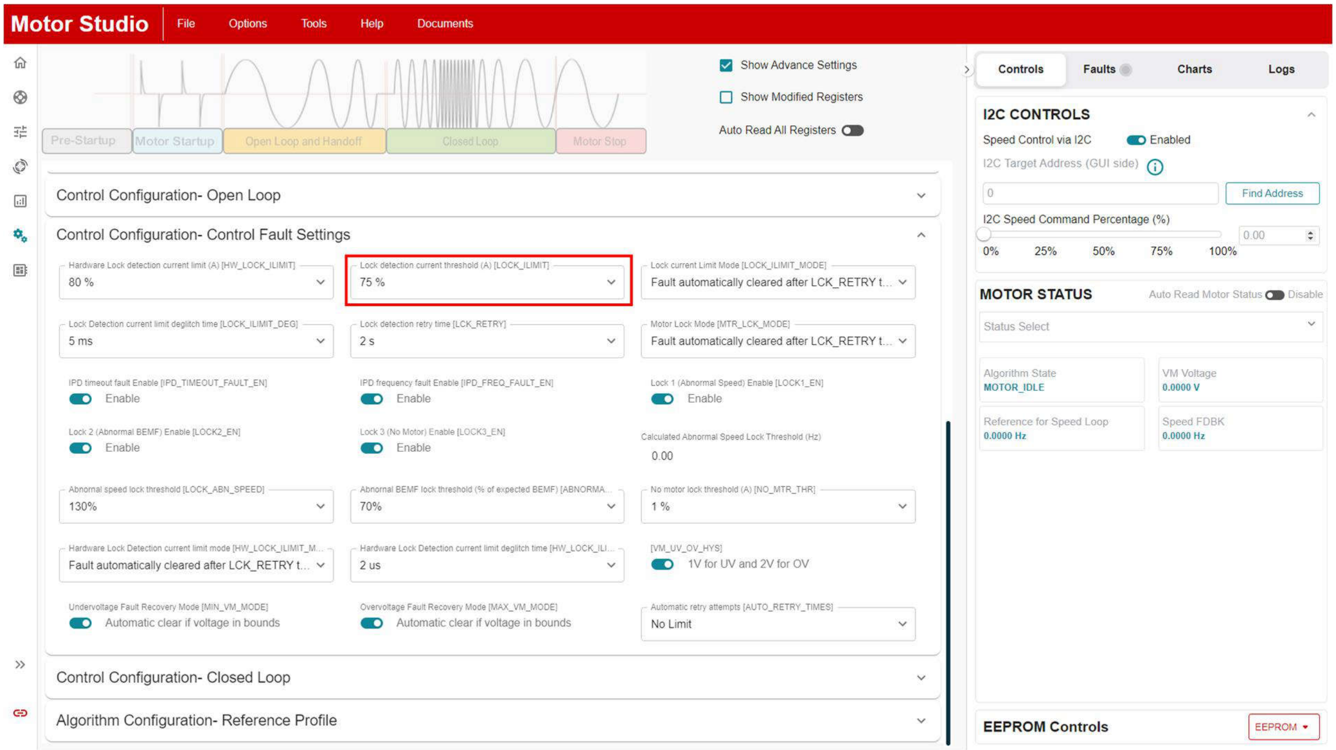

Figure 7-4. LOCK_ILIMIT

7.4 Hardware Lock Current Limit [HW_LOCK_LIMIT]

This fault gets triggered when the phase current exceeds the the HW_LOCK_ILIMIT threshold. If this fault is triggered, follow below recommendations.

Step 1: Using the fields circled in Figure 7-5, set SPD_LOOP_KP, SPD_LOOP_KI, CURR_LOOP_KP, and CURR_LOOP_KI to zero. This enables the MCF8329A to automatically calculate the speed loop and current loop PI controller gains.

Step 2: If the fault still persists, check the continuity across phase-to-phase, phase-to-GND, and PVDD-to-GND to make sure there is no short across these terminals.

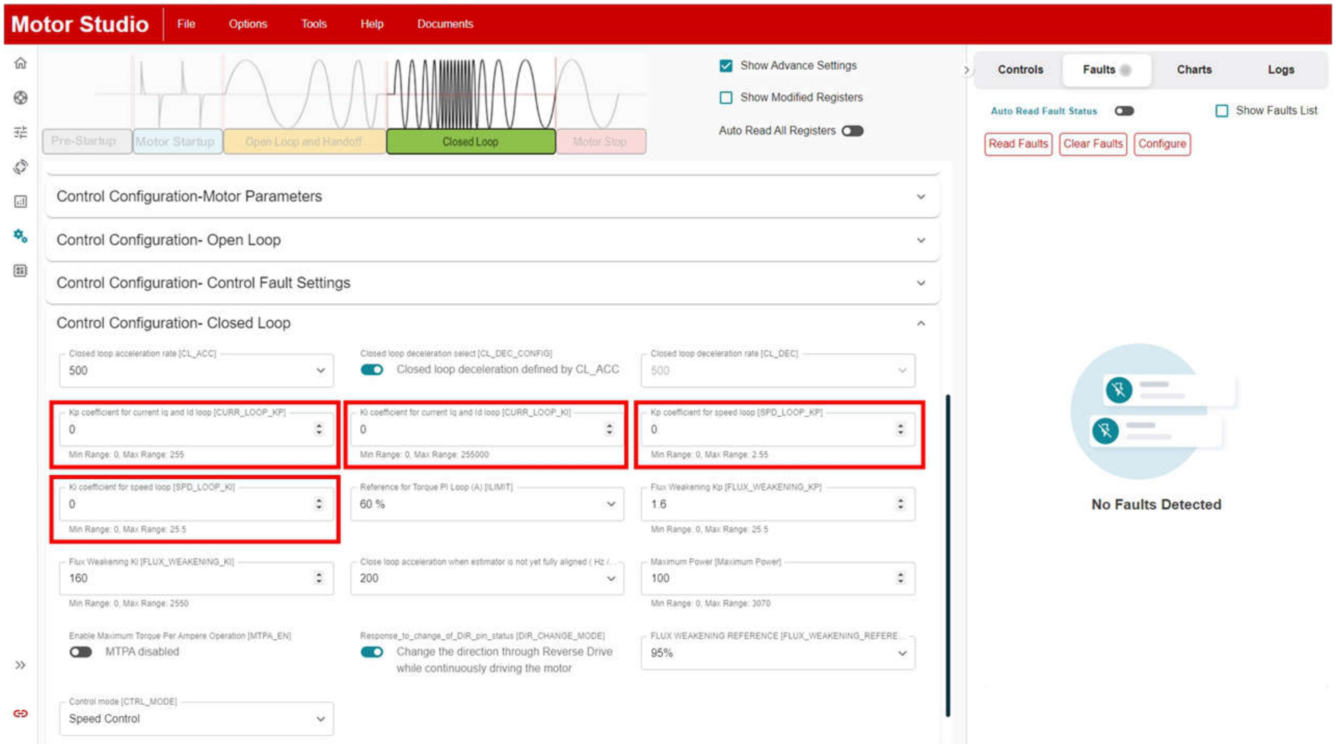

Figure 7-5. HW_LOCK_LIMIT

7.5 No Motor Fault [NO_MTR]

This fault gets triggered when the phase current is below the no motor lock threshold for 500ms during open loop. When this fault gets triggered, follow the recommendations below:

Step 1: Make sure the motor phases are securly connected to the OUTA, OUTB, and OUTC test points or the connector block J11.

Step 2: If the fault persists, set the no motor lock current threshold [NO_MTR_THR] to 5%.

Step 3: For low inductance motors, increase PWM switching frequency [PWM_FREQ_OUT].

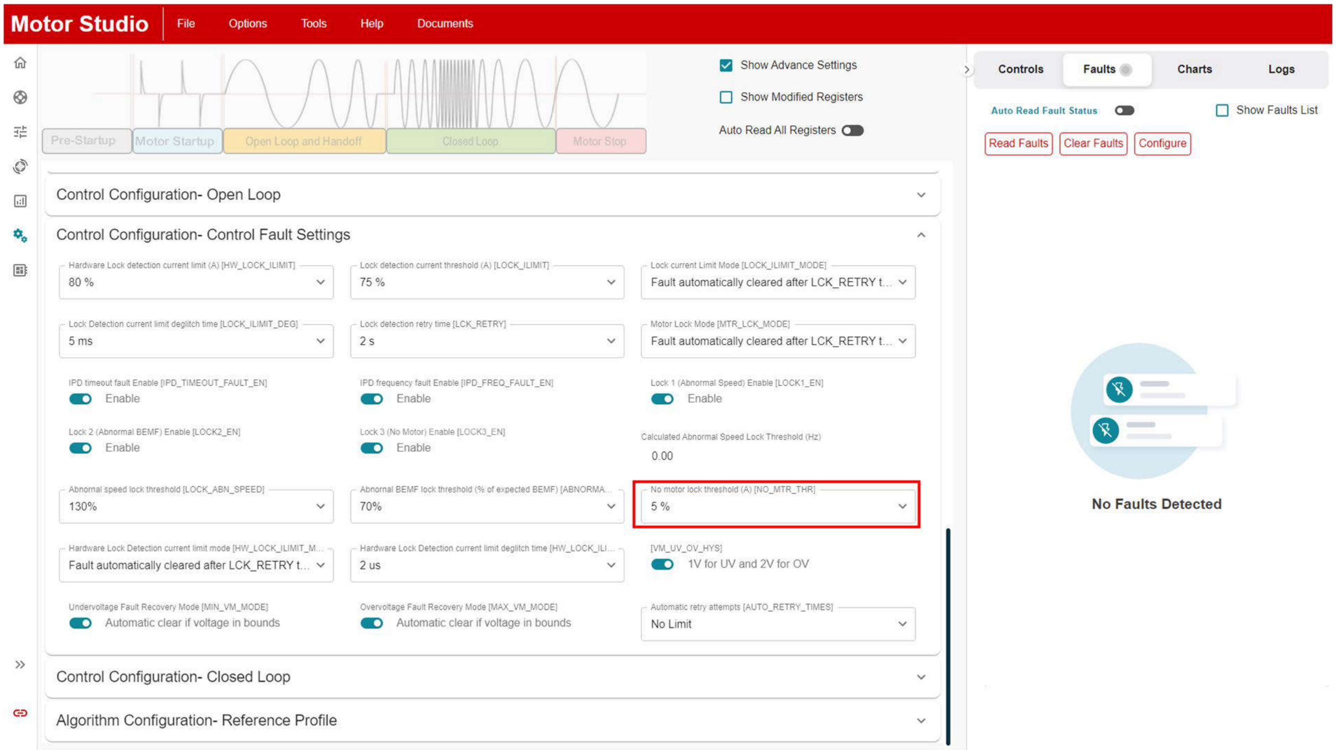

Figure 7-6. NO_MTR

Note

The MCF8329A might trigger loss of sync [LOSS_OF_SYNC] when motor phases are disconnected while the motor is spinning.

7.6 Abnormal Speed [ABN_SPEED]

This fault gets triggered when motor speed exceeds abnormal speed threshold [LOCK_ABN_SPEED]. When this fault gets triggered, follow below recommendations:

1. Increase align time [ALIGN_TIME], decrease slow first cycle frequency [SLOW_FIRST_CYC_FREQ], or increase the IPD current threshold [IPD_CURR_THR] and IPD repeat times [IPD_REPEAT] depending on the start-up mode selected.
2. Decrease open loop acceleration A1 [OL_ACC_A1] and open loop acceleration A2 [OL_ACC_A2].
3. Decrease closed loop acceleration [CL_ACC]

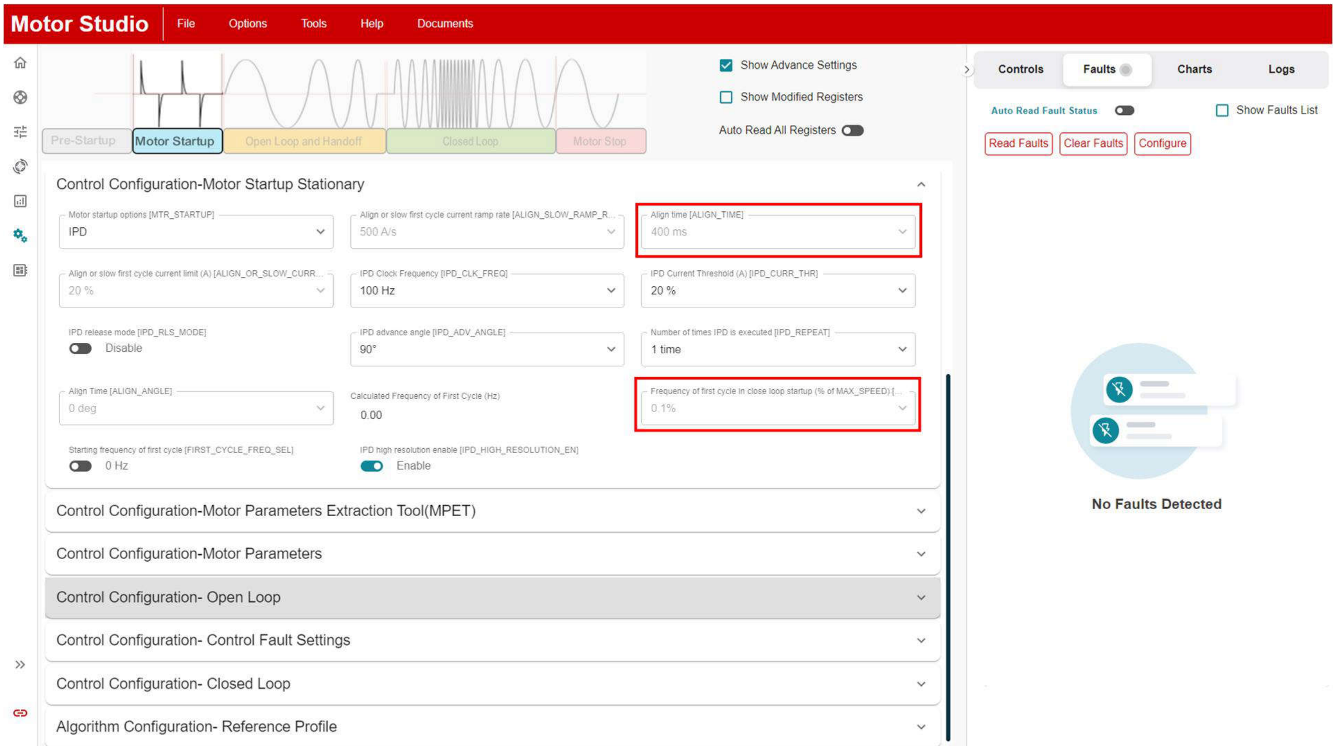

Figure 7-7. ABN_SPEED

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265