Texas Instruments

User's Guide

SLOU395D–July 2014–Revised June 2017

DRV10983 and DRV10975 Tuning Guide

The DRV10983 and DRV10975 are three-phase sensorless motor drivers, featuring an I^2C interface that allows the user to reprogram specific motor parameters in registers and burn them into the EEPROM to help optimize the performance for a given application. This document helps customers quickly set up the DRV10983 and DRV10975, enabling them to experience the devices' powerful performance and flexible programmability.

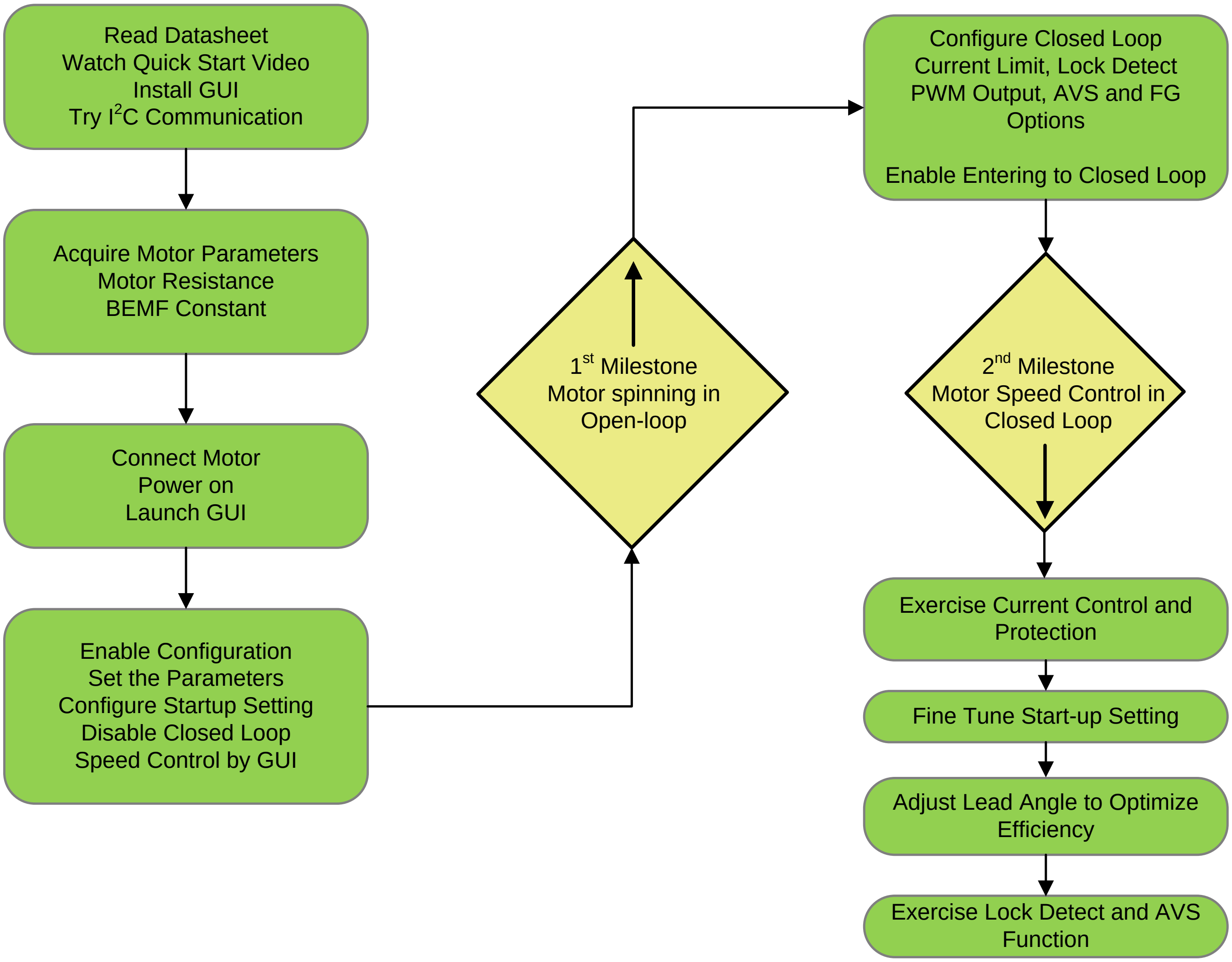


Figure 1. Sequence of Events

Copyright © 2014–2017, Texas Instruments Incorporated

1 Bench Set Up

Before connecting your own motor, read the DRV10983 (SLVSCP6) or DRV10975 datasheet (SLVSCP2), DRV10983 and DRV10975 EVM user's guide (SLOU393), and watch the quick start video. Also, install the GUI software on your computer, and make sure the I^2C communication is working. If you have not done so, please refer to the Quick Start Guide (SLYU022).

2 Acquire Motor Parameter

When a new project is started, the first step is to acquire the motor parameters (see Table 1). The motor parameters help determine whether it is suitable for the DRV10983 and DRV10975 and what the proper settings are for this motor. The following sections describe how to measure each parameter. Having a picture of the motor is also suggested to have an overview of the motor design.

Table 1. Motor Parameter Table

ITEM NUMBER	OPERATION VOLTAGE	NUMBER OF POLES	MAXIMUM SPEED (RPM)	MAXIMUM CURRENT	R (PHASE-CT)	Kt (PHASE-PHASE)	LR CONSTANT	INERTIA	PICTURE
Motor1									

2.1 Operation Voltage, Number of Poles, Maximum Speed, and Maximum Current

These four parameters should be provided by the motor specification or the application specifications.

2.1.1 Operation Voltage

Most of the time, operation voltage is fixed with less than 10% error. Some applications adjust motor speed by adjusting the operation voltage. In this condition, record the maximum operation voltage.

Copyright © 2014–2017, Texas Instruments Incorporated

2.1.2 Number of Poles

The number of poles means the number of poles of the permanent magnet. The number of pole pairs is the number of poles divided by two.

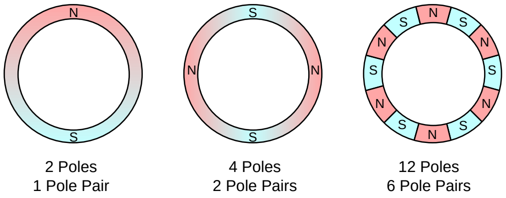


Figure 2. Pole Pairs in Permanent Magnets

If the number of poles is not listed in the motor specification, measure it with the following method:

1. Use a lab power supply and inject current from phase U to phase V. Make sure the current amplitude is less than the motor rated current.

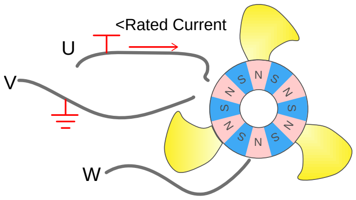


Figure 3. Injecting Current from Phase U to Phase V

2. The rotor should have settled at one position with the injecting current. Manually rotate the rotor; it will have several *settle-down* positions around one mechanical cycle.
3. Count the number of settle-down positions, which is the number of pole pairs. Multiplying by two calculates the number of poles (In Figure 4, there are six pole pairs in the motor).

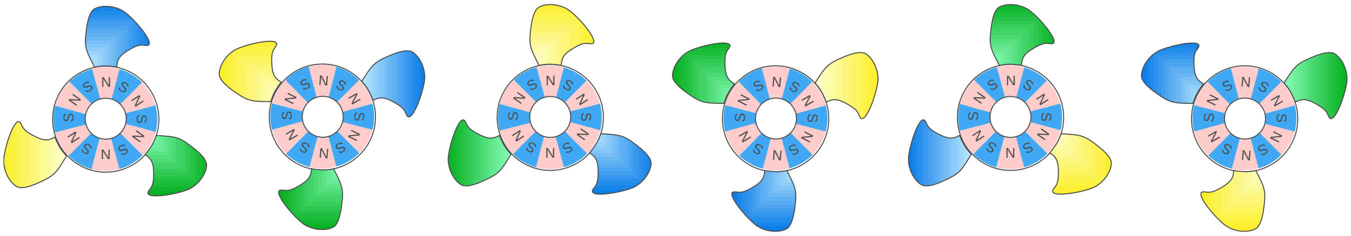


Figure 4. Multiple Motors

2.1.3 Maximum Speed and Maximum Current

If unable to determine the pole pairs, you can leave this empty and spin the motor.

Maximum current means the current consumption at steady state (not during accelerating or startup), which normally happens at maximum speed. Maximum speed and maximum current depend on the load condition. Heavier loads cause higher current and slower speed. For a fan application, know the speed and current with the blades assembled. If the full-load speed and no-load speed are both shown in the motor specification, record both.

Copyright © 2014–2017, Texas Instruments Incorporated

2.2 Motor Resistance

For a wye-connected motor, the motor phase resistance refers to the resistance from the phase output to the center tap (R_{PH_CT}). This is denoted as R_{PH_CT} in Figure 5.

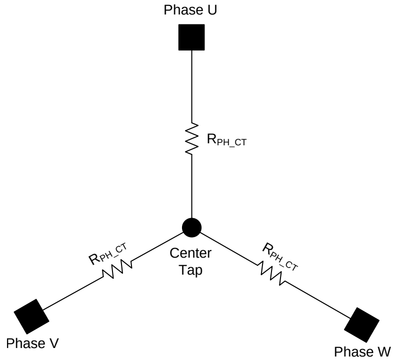


Figure 5. Wye-Connected Motor Resistance

For a delta-connected motor, the motor phase resistance refers to the equivalent phase to center tap in the wye configuration. In Figure 6, it is denoted as R_Y × R_{PH_CT} = R_Y.

For both the delta-connected motor and the wye-connected motor, the easy way to get the equivalent R_{PH_CT} is to measure the resistance between two phase terminals (R_{PH_PH}) and then divide this value by two: $R_{PH\_CT} = ½ R_{PH\_PH}$.

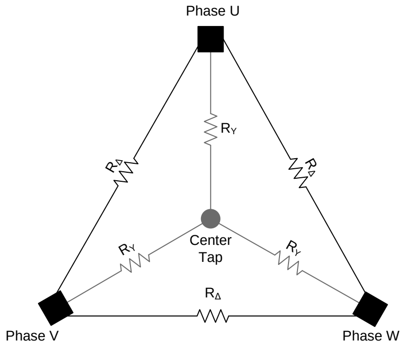


Figure 6. Delta-Connected Motor and the Equivalent Wye Connections

The maximum resistor value (R_{PH_CT}) that can be programmed for the DRV10983 is 18 Ω and the minimum resistor value is 0.029 Ω (R_{PH_CT}). For the DRV10975, the maximum resistor value (R_{PH_CT}) that can be programmed is 14 Ω and the minimum resistance is 0.0294 Ω (R_{PH_CT}).

Copyright © 2014–2017, Texas Instruments Incorporated

2.3 Motor Velocity Constant

The motor velocity constant describes the motor's phase-to-phase back electromotive force (BEMF) voltage as a function of the motor velocity. The measurement technique for this constant as used in the DRV10983 and DRV10975 is illustrated in Figure 7.

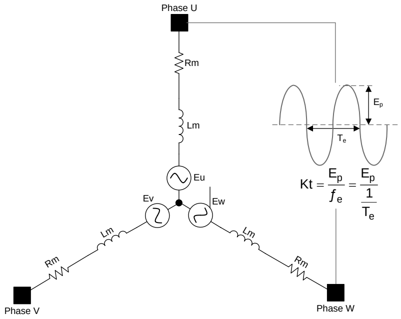


Figure 7. Kt_{PH} Definition

Manually spin the motor quickly or coast it, and use an oscilloscope to capture the differential voltage waveform between any two phases. The motor velocity constant used by the DRV10983 can be derived as shown in Equation 1:

$$Kt_{PH} = Ep \times Te$$

where

- Ep is ½ the peak-to-peak amplitude of the measured voltage
- Te is the electrical period (1)

The maximum Kt_{PH} that can be programmed in the DRV10983 is 1760 mV/Hz and the minimum that can be programmed is 0.92 mV/Hz. . For DRV10975, values are between 1330 mV/Hz and 0.70 mV/Hz.

If you are not able to figure out the motor velocity constant from the previous steps, it can be estimated by the maximum motor speed:

Motor Velocity Constant ≈ VCC / maximum motor speed (no load) (2)

- For example: motor spins with 1500 rpm (4 pole) at 24-V power supply:
  - 1500 rpm / (60 / 2 pole pairs) = 50 Hz
  - 24 V / 50 Hz = 480 mv/Hz

or:

Motor Velocity Constant ≈ (VCC – maximum current × resistance × √ 3) / maximum motor speed (full load) (3)

- For example: motor spins with 1000 rpm (4 pole) at 24-V power supply, the current is 1 A and phase resistance is 2 Ω.
  - 1000 rpm / (60 / 2 pole pairs) = 33.3 Hz
  - (24 V – 1 A × 2 Ω × √ 3) / 33.3 Hz = 620 mv/Hz

Copyright © 2014–2017, Texas Instruments Incorporated

2.4 Motor Electrical Time Constant (LR Time Constant)

The electrical time constant (T_{LR}) in a BLDC motor is defined as motor phase-to-phase inductance divided by phase-to-phase resistance.

Our suggestion is start the motor first and then measure this parameter. You can skip this section; Section 4 describes how to measure it.

Please note that phase-to-phase inductance and phase-to-phase resistance are used to calculate the T_{LR}. Phase-to-center tap (CT) inductance divided by phase-to-CT resistance does not have the same result:

$$\frac{L_{PH-PH}}{R_{PH-PH}} \neq \frac{L_{PH-CT}}{R_{PH-CT}} \tag{4}$$

- For example, winding A→B→C is put on the iron core:
  - RAB = RBC. So, RAC = 2 × RAB
  - LAB = LBC, but LAC ≠ 2 × LAB, because of the mutual inductance

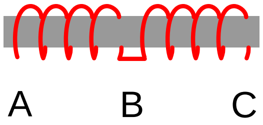


Figure 8. Mutually Coupled Windings ABC

Some multi-meters and LRC meters have an inductance measuring function. However, the result is normally the equivalent impedance of the LR circuit, not the pure inductance value. When it is converted to the inductance value, it is not accurate.

To measure the LR time constant, do not record the L value or the R value; instead, measure the current ripple value when the motor is started by the DRV10983 or DRV10975. A larger LR time constant results in a smaller current ripple.

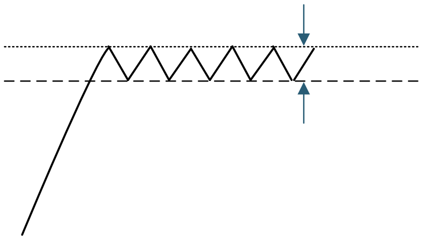


Figure 9. Current Ripple Value

When running the motor, set the align time to the maximum value and measure the V phase with a current probe. During the initial start, V phase current ripple represents the LR time constant.

Copyright © 2014–2017, Texas Instruments Incorporated

2.5 Motor Inertia

Motor inertia is the tendency of the motor to keep moving at the existing constant velocity, or to keep still. Motor inertia depends on the rotor structure, including the blades. Normally, motors with heavier blades or larger blades have greater inertia.

The unit of inertia is kg × m × m, but motor manufacturers usually don't provide this information. It is very difficult and not always necessary to find out the accurate inertia value. Instead, use the oscillation period value (T) to characterize the motor inertia (Large inertia motors have longer oscillation periods). The motor inertia affects the align time. Greater inertia means a longer align time.

The following steps can be used to measure the oscillation of the motor:

1. Connect the oscilloscope to capture the voltage between V and W.
2. Make sure the blades are assembled. Use a lab power supply with current limit function and inject 1 A of current into phase U and return from phase V of the motor. If the rated current of the motor is less than 1 A, inject the rated current and record a note in the table.
3. The motor should have settled at one position.
4. Manually rotate the fan blades off the settled position, then release. The motor will oscillate back and forth around the settle point.
5. Measure the period of the swing using an oscilloscope.

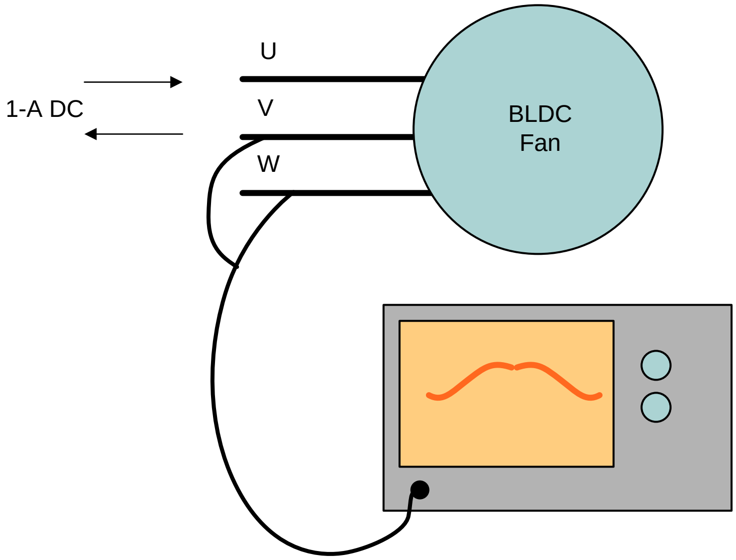


Figure 10. Oscilloscope Setup

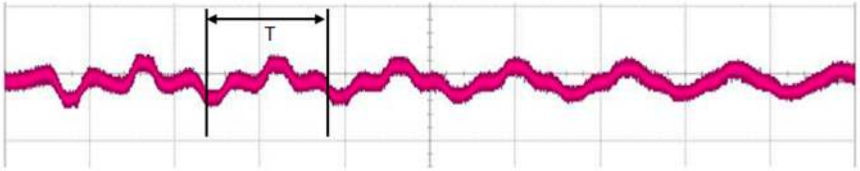


Figure 11. Oscilloscope Reading

Copyright © 2014–2017, Texas Instruments Incorporated

If unable to figure out the motor inertial, keep this cell empty and go ahead and spin the motor. After completing Section 2.1 to Section 2.5, it is possible to fill in all or most of the motor parameter table.

Here is one example:

Table 2. Motor Parameters

MOTOR NAME	OPERATION VOLTAGE	NUMBER OF POLES	MAXIMUM SPEED(RPM)	MAXIMUM CURRENT	R (PHASE-CT)	Kt (PHASE-PHASE)	LR CONSTANT	INERTIAL	PICTURE
TI_M1	12 V	4	3000	450 mA	3 Ω	100 mV/Hz	-	200 ms	

Copyright © 2014–2017, Texas Instruments Incorporated

3 Motor Connection and Power On

For a sensorless controller, you can connect phases U, V, and W of the driver to the three terminals of the motor in any sequence. If the direction of the motor is not what you want, simply swap two of the three wires and the motor direction changes.

Before powering the device on, refer to Section 5 of the DRV10983 and DRV10975 EVM user's guide (SLOU393). Connect a current probe to phase V to observe the phase current during tuning. Power the device on. The power consumption should resemble Table 3:

Table 3. Average Power Consumption

CONDITION	24-V VCC SWITCHING REGULATOR	12-V VCC SWITCHING REGULATOR	24-V VCC LINEAR REGULATOR	12-V VCC LINEAR REGULATOR
VCC current	3 mA	6 mA	11 mA	11 mA

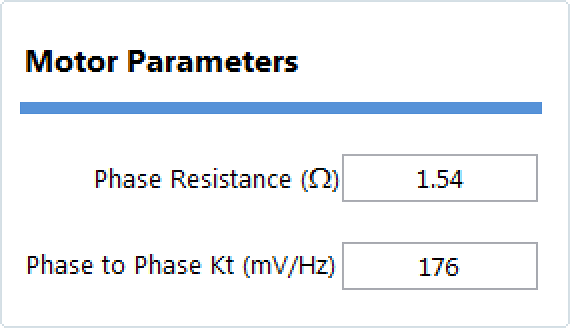
Launch the GUI software.

4 Spin the Motor

After following the steps from Section 4.1 to Section 4.8, the motor can be spun in open loop control.

4.1 Motor Parameters

Enter the motor parameters you previously measured and recorded in your table.



Motor Parameters

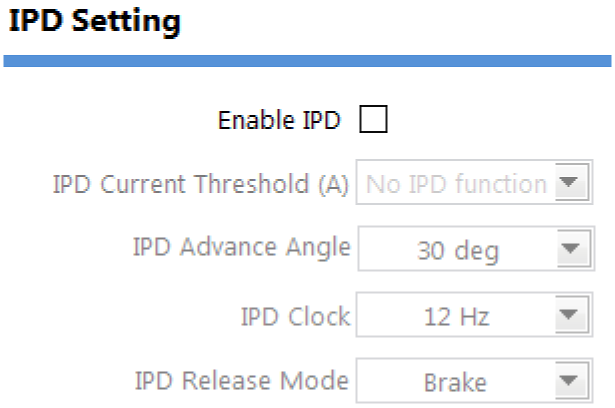
Phase Resistance (Ω) 1.54

Phase to Phase Kt (mV/Hz) 176

Figure 12. Motor Parameters

4.2 Disable IPD

Uncheck the *Enable IPD* box, disabling the initial position detection (IPD) function.



IPD Setting

Enable IPD ☐

IPD Current Threshold (A) No IPD function

IPD Advance Angle 30 deg

IPD Clock 12 Hz

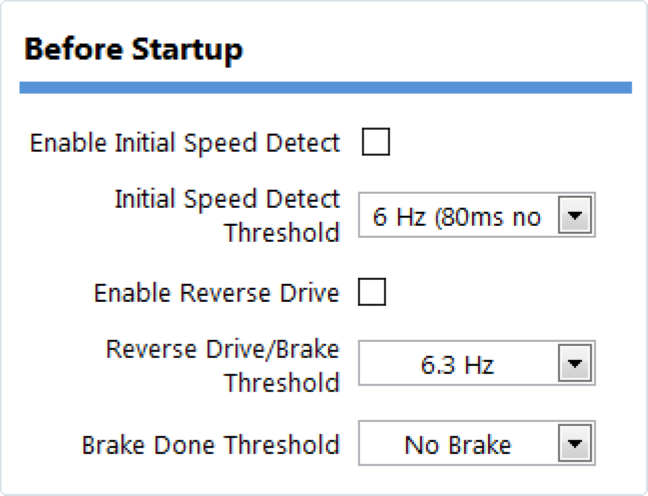
IPD Release Mode Brake

Figure 13. IPD Setting

Copyright © 2014–2017, Texas Instruments Incorporated

4.3 Roughly Configure the Before Startup

Uncheck the *Enable Initial Speed Detect* and *Enable Reverse Drive* boxes. Set the *Brake Done Threshold* to *No Brake*.



Before Startup

Enable Initial Speed Detect ☐

Initial Speed Detect Threshold: 6 Hz (80ms no

Enable Reverse Drive ☐

Reverse Drive/Brake Threshold: 6.3 Hz

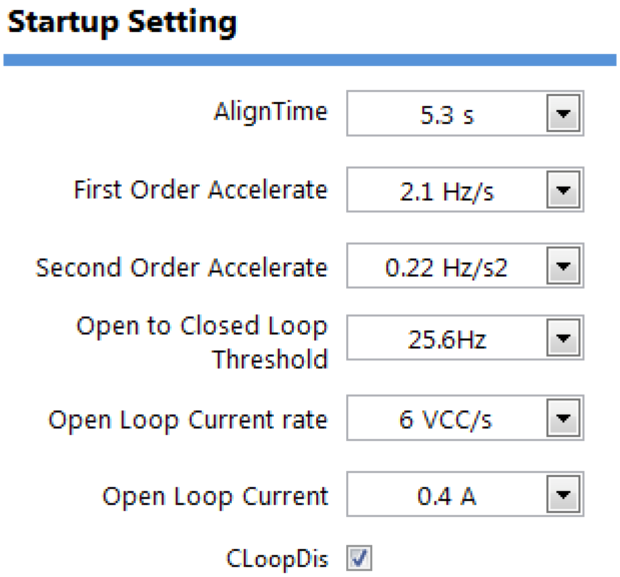
Brake Done Threshold: No Brake

Figure 14. Before Startup

Copyright © 2014–2017, Texas Instruments Incorporated

4.4 Configure the Startup Setting

1. Set the *Align Time* to the maximum value (5.3 s) to allow time to measure the phase current.
2. Set the *Open to Closed Loop Threshold* at around one-third to one-fifth of the maximum motor speed. For example, if the motor maximum speed is 80 Hz, set the *Open to Closed Loop Threshold* to 25.6 Hz.
3. Set *First Order Accelerate* and *Second Order Accelerate* based on motor inertia and *Open Loop Current rate* settings. Heavier motors need to accelerate slower. If unsure, try a slower setting: 2.1 Hz/s and 0.22 Hz/s^2, respectively. The *Open Loop Current rate* should remain the same for now.
4. Based on the SoftStart requirement of the application, select the *Open Loop Current*. Put in 0.4 A if there is no requirement, or if unsure.
5. Make sure the *CLoopDis* is checked, disabling the close loop control. This way the open loop control can be tested and verified first.



Startup Setting

AlignTime: 5.3 s

First Order Accelerate: 2.1 Hz/s

Second Order Accelerate: 0.22 Hz/s2

Open to Closed Loop Threshold: 25.6Hz

Open Loop Current rate: 6 VCC/s

Open Loop Current: 0.4 A

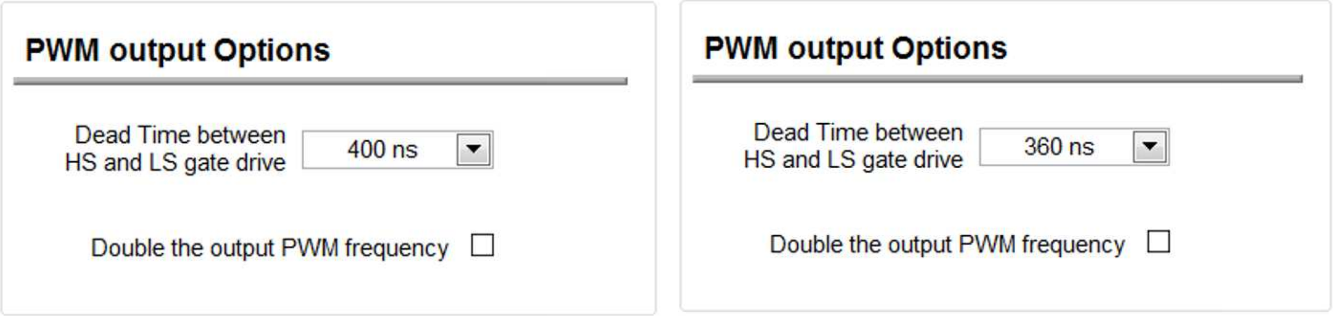
CLoopDis ☑

Figure 15. Startup Settings

Copyright © 2014–2017, Texas Instruments Incorporated

4.5 PWM Output Options

Make sure the *Dead Time between HS and LS gate drive* is sufficient to avoid shooting through. The recommended minimum dead time is 400 ns for a 24-V VCC and 360 ns for a 12-V VCC.



PWM output Options

Dead Time between HS and LS gate drive: 400 ns

Double the output PWM frequency ☐

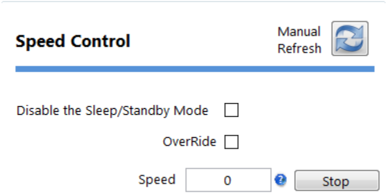
PWM output Options

Dead Time between HS and LS gate drive: 360 ns

Double the output PWM frequency ☐

Figure 16. PWM Output Options

4.6 Spin the Motor in the Open Loop



Disable the Sleep/Standby Mode ☐

OverRide ☐

Speed: 0

Figure 17. Speed Control Settings

1. Check both the *Disable the Sleep/Standby Mode* box and the *OverRide* box, enabling speed command by I^{2}C.
2. Input a non-zero speed. The motor should start to spin in open loop. The rotating speed should be the *Open to Closed Loop Threshold*. Note that the open loop operation speed is not determined by the value of *Speed* command; it is always at *Open to Closed Loop Threshold*.
3. TI suggests using "300" as *Speed* command in order to smoothly transfer to closed loop (Section 5).

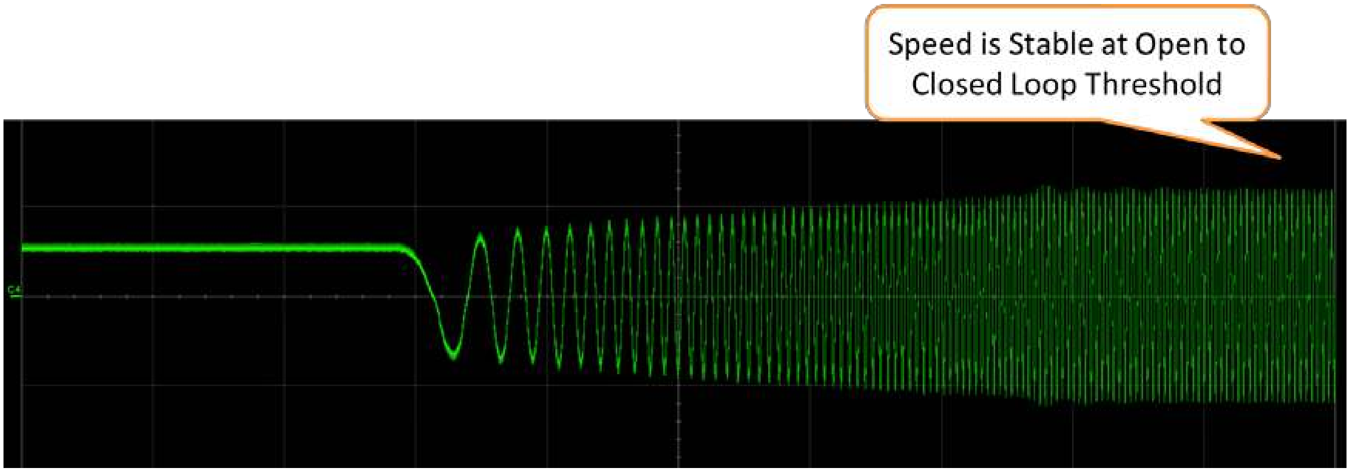


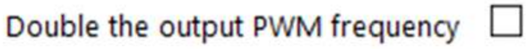
Figure 18. Speed Waveform

Copyright © 2014–2017, Texas Instruments Incorporated

4.7 Measure the LR Time Constant

The LR time constant is important to the system because it affects how long the control advance time is. A higher LR constant means a higher control advance time. This control advance time will be set in Section 5.4.

The *Align Time* is set to 5.3 seconds, this large of a value is not necessary for most applications. The reason we set it to 5.3 seconds is so the phase current can be captured during the align process in order to measure the LR time constant by finding the current ripple at this condition. The current ripple should be 25 kHz if the DRV10983 and DRV10975 output PWM frequency is set to the default 25 kHz (but verify under advanced settings: *PWM Output Options* [Section 4.5]).



Double the output PWM frequency ☐

The unchecked box is 25 kHz PWM.

The amplitude of the ripple represents the LR time constant:

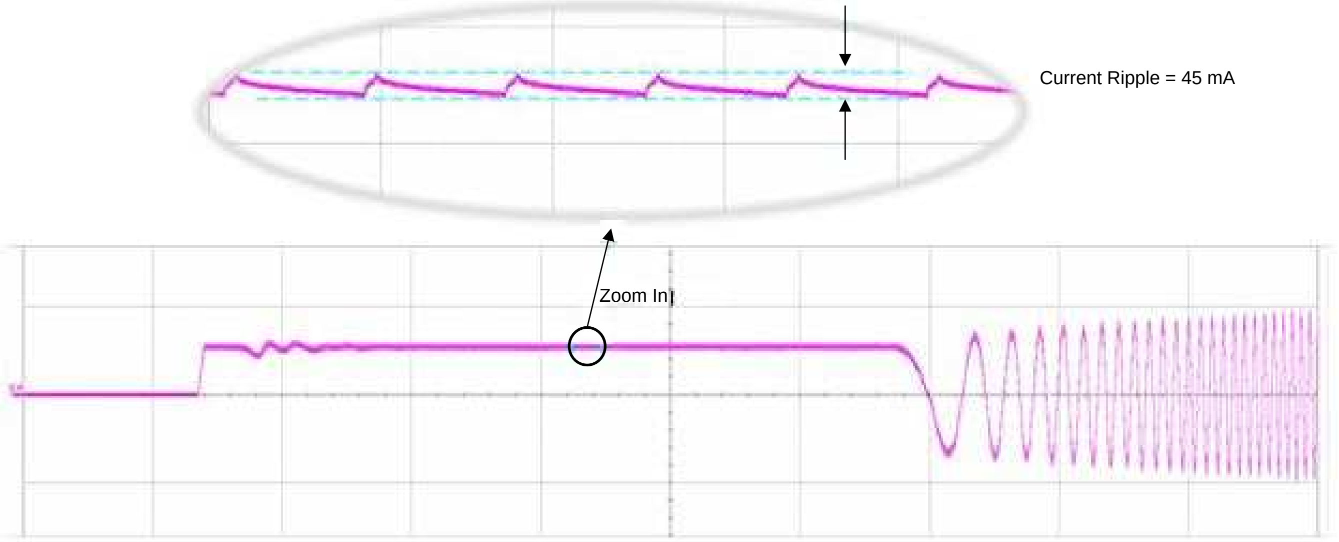


Figure 19. Ripple Amplitude

Put this current ripple value into the parameter table:

Table 4. Motor Parameters

MOTOR NAME	OPERATION VOLTAGE	NUMBER OF POLES	MAXIMUM SPEED (RPM)	MAXIMUM CURRENT	R (PHASE-CT)	Kt (PHASE-PHASE)	LR CONSTANT	INERTIAL	PICTURE
TI_M1	12 V	4	3000	450 mA	3 Ω	100 mV/Hz	45 mA	200 ms	

After measuring the LR time constant, the align time can be reduced to make the startups faster. For example, set it to 0.67 s.

AlignTime 0.67 s

Figure 20. AlignTime Setting

Copyright © 2014–2017, Texas Instruments Incorporated

4.8 Evaluate the Calculated Kt Value

Observe the waveform for the phase V current and compare it to the following pictures in Figure 21:

- Picture 1: the parameter entered for Kt is correct.
- Picture 2: the Kt entered is too big.
- Picture 3: the Kt entered is too small.

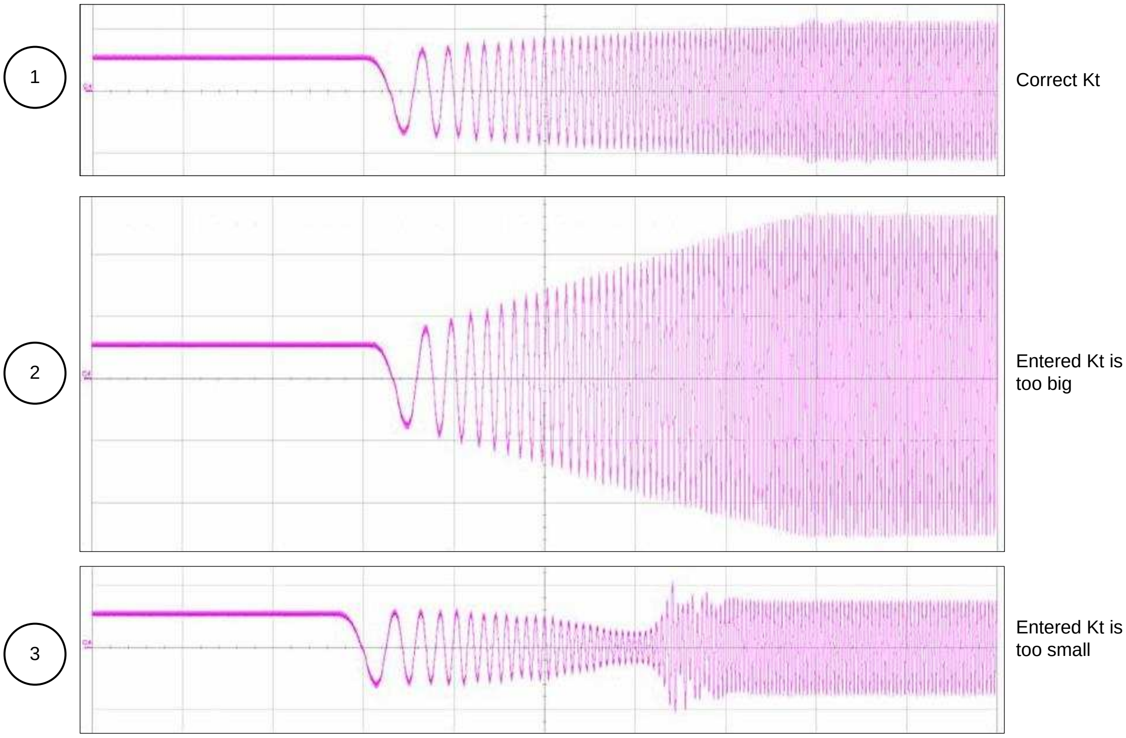


Figure 21. Waveform Comparison

The motor should have stopped at the time denoted in following image:

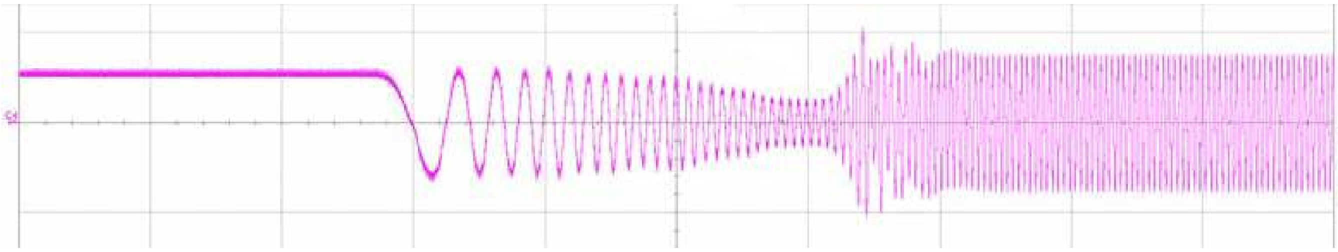


Figure 22. Motor Stopped Waveform

An alternative way to measure the Kt is by trying different Kt values until there is a fairly constant envelope current (Picture 1) during open loop. It is normal that the current envelope slightly increases while the speed is increasing.

During this measurement, if you want to look at the startup again, set the speed to 0 (use the *Stop* button), after the motor stops, set it back to a non-zero value again.

Speed 0

Speed 300

Figure 23. Speed Settings

Copyright © 2014–2017, Texas Instruments Incorporated

If the motor is not moving and the current waveform is like Figure 24 (current keeps going higher), it means the startup acceleration is too aggressive for this motor.

Figure 24. Aggressive Current Waveform

Reduce the *First Order Acceleration* until a reliable startup is achieved.

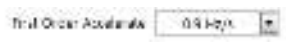


Figure 25. First Order Accelerate Setting

Copyright © 2014–2017, Texas Instruments Incorporated

5 Enter the Closed Loop

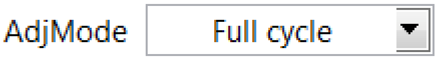
If the motors' BEMF is sinusoidal, a smooth sinusoidal current waveform should be seen in open loop operation. But the efficiency in open loop operation is not good. Also, open loop operation is very unstable, putting some external load on the motor makes it stop.

To spin the motor in closed loop, we need to configure the following parameters. You can configure these settings while the motor is spinning in open loop because it will not affect the open loop operation.

5.1 Configure the Closed Loop Setting

5.1.1 AdjMode

Set the *AdjMode* to *Full Cycle* adjustment.



AdjMode Full cycle

Figure 26. AdjMode Setting

Choose *Half Cycle* adjustment only when the motor Kt is very large and the maximum speed is low. It can make the motor dynamic response faster but the potential risk is the phase sinusoidal current gets distorted.

Also, *Half Cycle* adjustment can solve the *BEMF Abnormal* lock detect issue described in Section 8.

5.1.2 Speed Input Mode

Speed Input Mode Analog Input

Figure 27. Speed Input Mode Setting

If the system provides an analog speed command, choose *Analog Input*. If the system provides PWM speed commands, choose *PWM Input*.

If the PWM frequency is beyond the DRV10983 and DRV10975 application range (1 kHz to 100 kHz), convert the PWM input into an analog voltage using an RC filter and select the *Analog* mode instead.

The DRV10983 and DRV10975 EVMs provide the analog speed command by adjusting the potentiometer.

If we use I^2C speed commands (checking the *OverRide* box enables I^2C speed commands), both the analog input and digital input will be ignored. In the following description, we use I^2C speed commands.

Copyright © 2014–2017, Texas Instruments Incorporated

5.1.3 Closed Loop Accelerate

Closed Loop Accelerate 0.37 VCC/s

Figure 28. Closed Loop Accelerate Setting

To prevent sudden changes in the torque applied to the motor which could cause high currents (accelerate) or VCC voltage surges (decelerate), buffer the speed command with *Closed Loop Accelerate*.

Closed Loop Accelerate is a supplementary method to the acceleration current limit and mechanical anti-voltage surge (AVS). If the acceleration current limit and mechanical AVS are both working properly, *Closed Loop Accelerate* can be set at *Inf fast*. At some particular motor parameter conditions (refer to the Section 6.2 and Section 9); acceleration current limit and mechanical AVS cannot work as expected. *Closed Loop Accelerate* is the direct way to adjust the buffered speed command.

TI suggests having some level of *Closed Loop Accelerate* setting in all applications. If the speed response is very strict in your application, increase the rate, otherwise, decrease the rate.

Set *Closed Loop Accelerate* to 0.37 VCC/s now; then continue to the next steps.

5.1.4 Control Coefficient

Set the control coefficient to 1.

5.1.5 Commutate Control Advance Mode and Setting

- In *Commutate Control Advance Mode*, choose *Constant Time*.
- For *Control Advance Setting*, start with this estimated advance time setting:
  20 µs × (Imax / Current Ripple) × (*Open Loop Current* × R) / VCC
  where
  - Imax is the current at the maximum speed with full load
  - R is motor phase resistance
  - Current Ripple is current ripple measured during align state (in this example: 45 mA)
  - *Open Loop Current* is the value we programmed

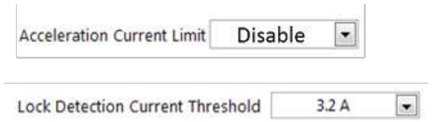
    Open Loop Current 0.4 A
  - If you do not want to do the calculation, put in 200 µs. Section 5.4 describes how to fine tune this parameter.

Copyright © 2014–2017, Texas Instruments Incorporated

5.2 Roughly Configure for Other Settings

5.2.1 Configure the Current Limit

Disable the *Acceleration Current Limit Function* and set the *Lock Detection Current Limit Threshold* to maximum.



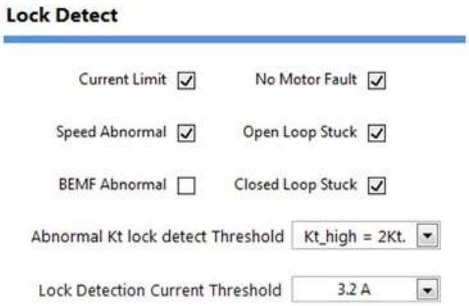
Acceleration Current Limit: Disable

Lock Detection Current Threshold: 3.2 A

Figure 29. Current Limit Settings

5.2.2 Configure Lock Detect

Enable all the *Lock Detect* options except the *BEMF Abnormal*, this feature is discussed in the *Lock Detect* section. Set the *Abnormal Kt lock detect Threshold* to Kt_high = 2 Kt, and Kt_low = 1/2 Kt.



Lock Detect

Current Limit ☑ No Motor Fault ☑

Speed Abnormal ☑ Open Loop Stuck ☑

BEMF Abnormal ☐ Closed Loop Stuck ☑

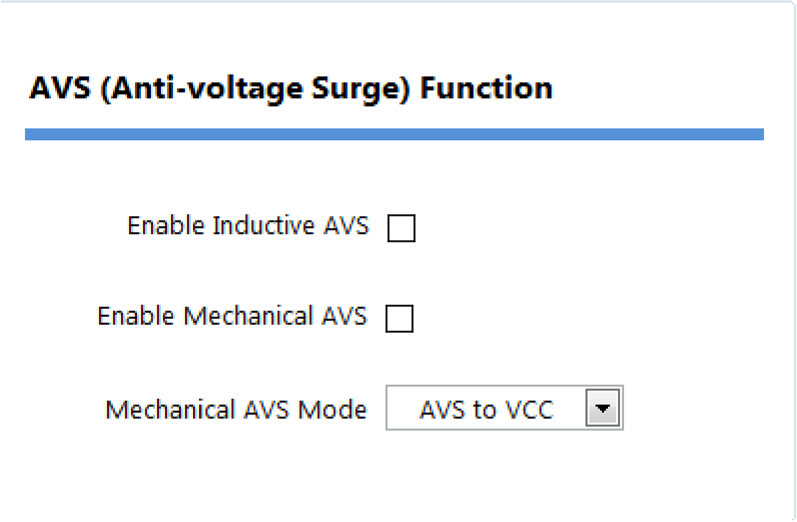
Abnormal Kt lock detect Threshold: Kt_high = 2Kt.

Lock Detection Current Threshold: 3.2 A

Figure 30. Lock Detect Settings

5.2.3 Configure the Anti-Voltage Surge (AVS) Function

Start with *AVS Function* disabled:



AVS (Anti-voltage Surge) Function

Enable Inductive AVS ☐

Enable Mechanical AVS ☐

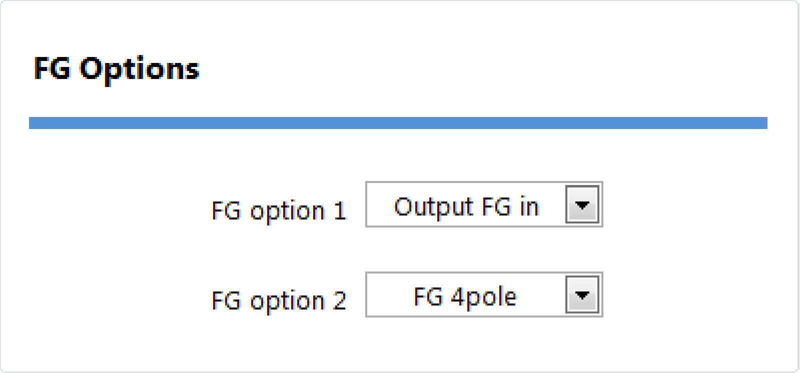
Mechanical AVS Mode: AVS to VCC

Figure 31. AVS Function Settings

Copyright © 2014–2017, Texas Instruments Incorporated

5.2.4 Configure FG Options

Keep the *FG Options* at the default settings.



FG Options

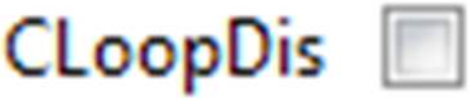
FG option 1: Output FG in

FG option 2: FG 4pole

Figure 32. FG Options

5.3 Enter Closedloop

While the motor is spinning in the open loop, enable the closed loop control.



CLoopDis ☐

Figure 33. CLoopDis

The motor speed should increase (if speed command = 300) and the phase current during open-to-closed loop transition should resemble Figure 34:

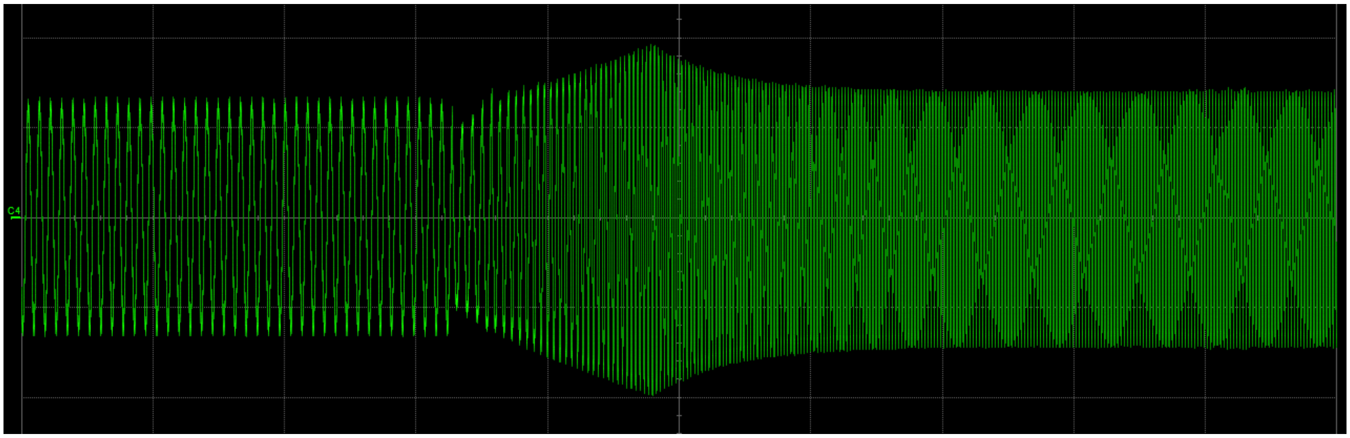


Figure 34. Waveform at Speed Command = 300

Copyright © 2014–2017, Texas Instruments Incorporated

Now, adjust the motor speed by entering the *Speed* command in the box (from 00 to 511):

Speed Control

Manual Refresh

Disable the Sleep/Standby Mode ☑

OverRide ☑

Speed 511

Figure 35. Speed Control Settings at 511

Figure 36 illustrates phase current with speed control:

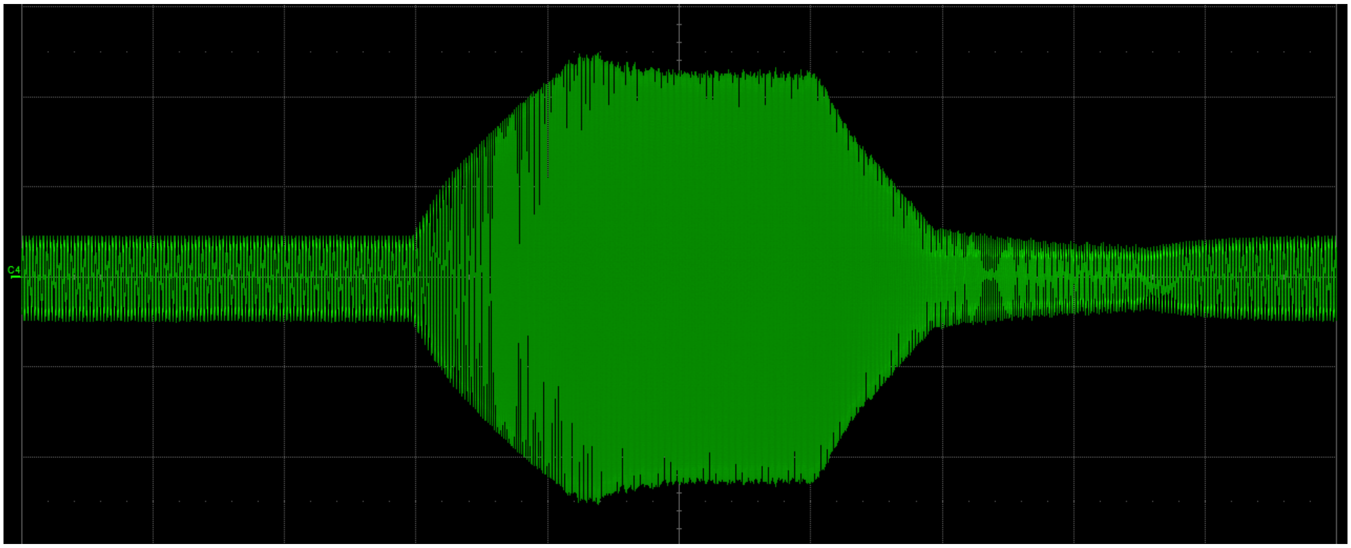


Figure 36. Phase Current With Speed Control Waveform

Copyright © 2014–2017, Texas Instruments Incorporated

5.4 Control Advance Time Fine Tune

The DRV10983 and DRV10975 have a control advance adjustment function to optimize the motor's operational efficiency. While the motor is spinning, the increases or decreases in the control advance angle will result in a phase voltage and phase current shift. At the same time, motor speed and supply current are also affected.

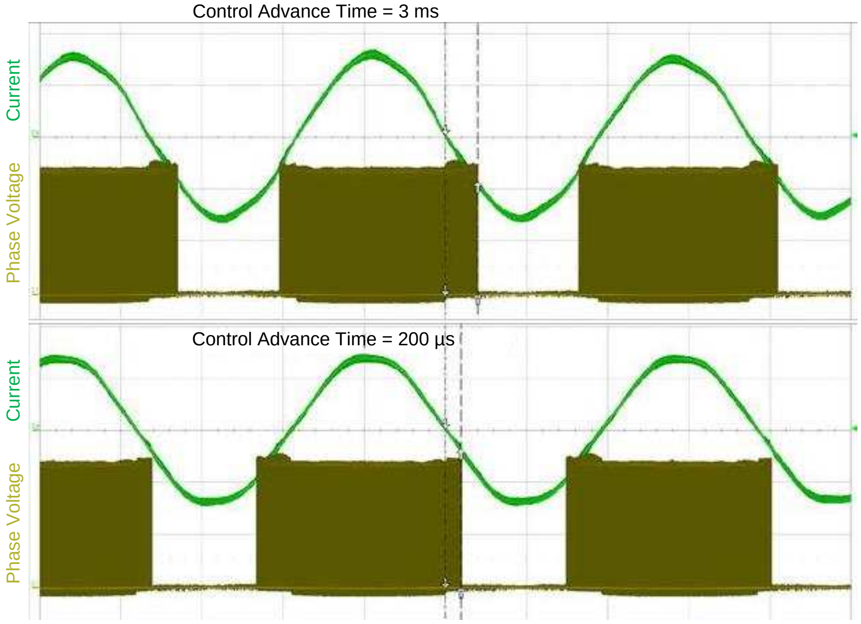


Figure 37. Control Advance Time Comparison

The criteria to find out the optimized control advance angle is

1. the motor should spin smoothly and the phase current waveform is stable, and
2. find out the smallest supply current at the same speed.

Copyright © 2014–2017, Texas Instruments Incorporated

6 Current Control

The DRV10983 and DRV10975 are able to control the motor phase current in startup and closed loop. It can also provide protection when current exceeds a set limit.

6.1 Open Loop Current Setting

The *Open Loop Current rate* setting starts the motor at the lowest current that will get the motor running. The higher the current setting, the more likely the motor will move; however, it is very inefficient to use such a large current for smaller devices.

The *Open Loop Current rate* setting has four options: 0.2 A, 0.4 A, 0.8 A, and 1.6 A. The selection depends on requirements for startup time, power supply capacity, and customer preference.

To have a SoftStart, set the *Open Loop Current* to 0.2 A; for a fast start, set it to a higher value such as 1.6 A. Normally, we suggest it be less than the full-speed operation current.

Current Setting 0.2 A

Figure 38. Current Setting

Figure 39 shows the different startup current settings:

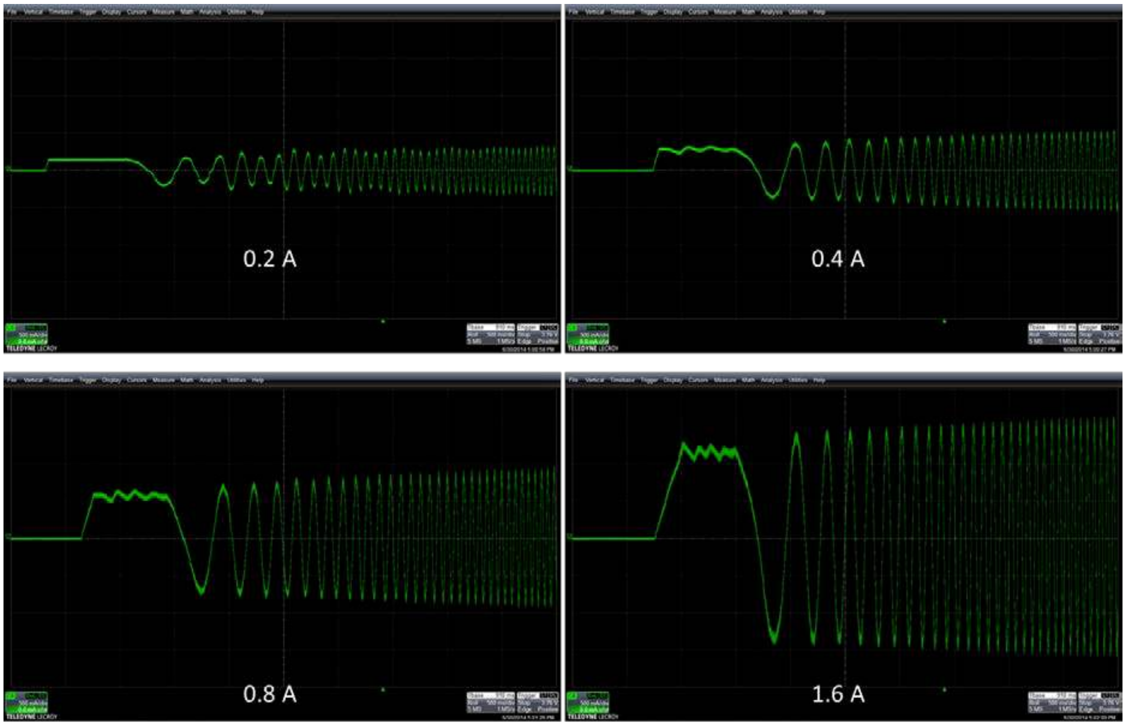
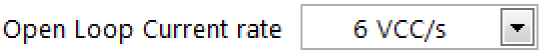


Figure 39. Waveforms of 0.2, 0.4, 0.8, and 1.6-A Currents

To avoid the acoustic noise caused by the fast slew rate of the driving phase, the DRV10983 and DRV10975 provide current ramp rate options ranging from 0.023 VCC/s to 6 VCC/s. Please refer to the *Start Up Current Ramp Up* section in the datasheet DRV10983 (SLVSCP6) or DRV10975 (SLVSCP2).



Open Loop Current rate 6 VCC/s

Figure 40. Open Loop Current Rate Setting

Copyright © 2014–2017, Texas Instruments Incorporated

6.2 Acceleration Current Limit

The acceleration current limit is useful at transitions from open loop to closed loop and also when the motor is accelerating. During these two conditions, the current amplitude may increase without the acceleration current limit. Please note that implementing the acceleration current limit will slow down the motor's acceleration. Consider the tradeoff between the acceptable current and the acceleration based on their system requirements.

Because the *Closed Loop Accelerate* function works as a buffer to slow down the acceleration and limit the phase current, it actually provides a similar function as acceleration current limit. We may not be able to see the effect of the acceleration current limit if we have set *Closed Loop Accelerate*. Figure 41 is captured with closed loop accelerate as *Inf fast*.

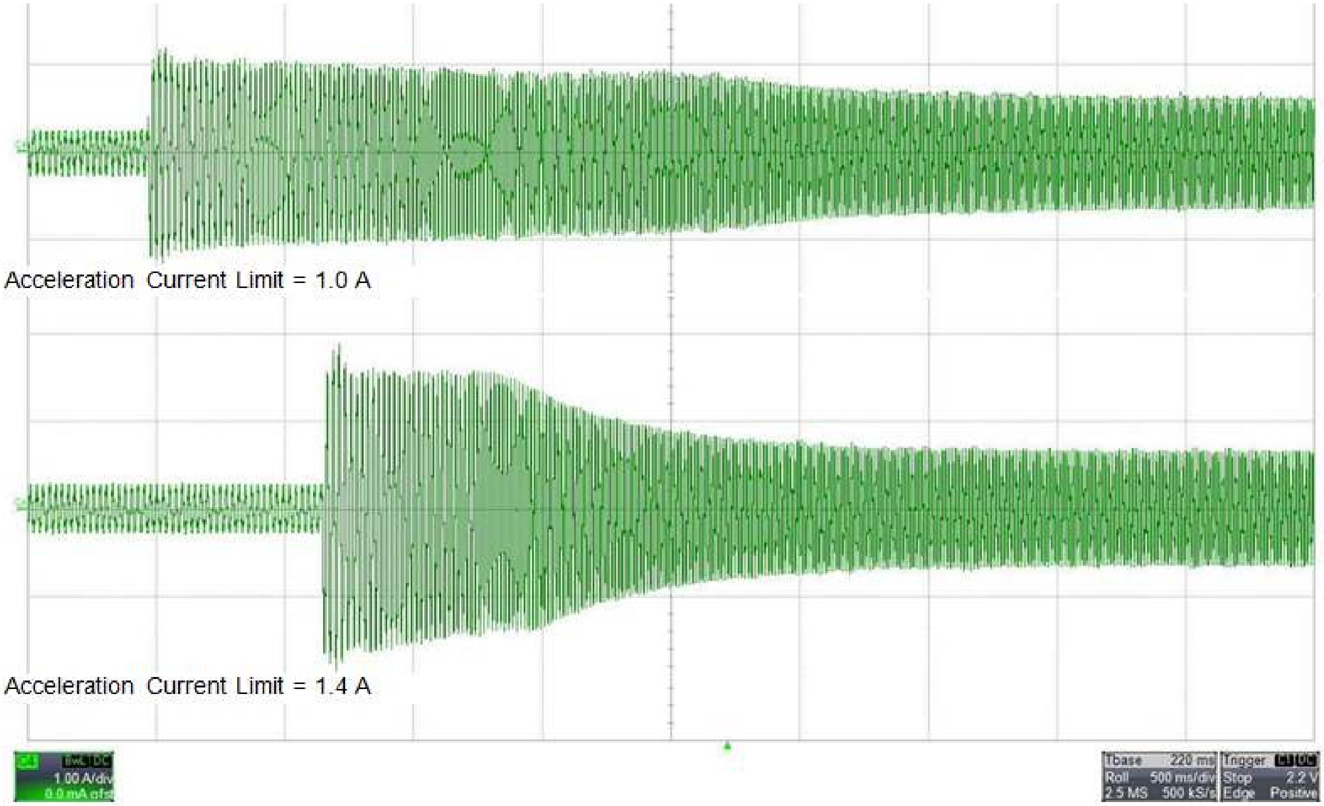
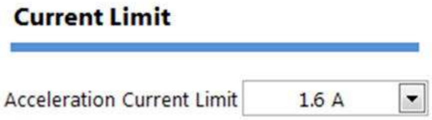


Figure 41. Acceleration Current Limit Comparison

The criteria for setting the *Acceleration Current Limit Threshold* are:

- The current setting should be higher than the current at the maximum speed with full load.
- The setting should be lower than the capability of the power supply and lower than the lock detection current limit threshold.

For example, if the motor requires 1.5 A with full load at full speed, and the power supply provides no more than 2-A current, set the *Acceleration Current Limit Threshold* to 1.8 A.



Current Limit

Acceleration Current Limit 1.6 A

Figure 42. Acceleration Current Limit Setting

Copyright © 2014–2017, Texas Instruments Incorporated

Sometimes, because of the implementation of the acceleration current limit, the motor is not able to accelerate to the target speed. In this case the *Speed Cmd Buffer* is lower than the *Speed Command*.

Speed Command (%) 100

Speed Cmd Buffer (%) 60

Figure 43. Speed Command Setting

This is because:

- The programmed Kt value is wrong and set too low. The solution is increasing the Kt value and checking whether the motor startup improves. If the difference between *Speed Command* and *Speed Cmd Buffer* gets closer, we need to continue increasing the Kt until they are the same.
- The motor has very low phase resistance; the Kt programmed is slightly lower than the correct value. However, if the value is increased by even one step, the motor cannot be controlled at low speed. In this case, choose the higher Kt and disable the mechanical AVS, or choose the lower Kt and increase the *Acceleration Current Limit Threshold* (or disable the acceleration current limit).

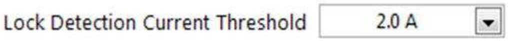
If the acceleration current limit is disabled, control the *Speed Command* slow enough (or select the slow *Closed Loop Accelerate* setting) to prevent a big inrush current.

Note that when the motor resistance and current are low, the acceleration current limit is not always accurate because of the resolution of Kt programming. *Closed Loop Accelerate* is a complementary method in this condition.

Copyright © 2014–2017, Texas Instruments Incorporated

6.3 Lock Detection Current Limit

The lock detection current limit is designed to protect the motor when it is blocked by an external force. It operates in both open loop and closed loop. The threshold is programmable and should be set to a value greater than 1.2 times the acceleration current limit. It should be set less than the short time current the system can handle, which is determined by the motor, the device, the power supply and the system specification.



Lock Detection Current Threshold 2.0 A

Figure 44. Lock Detection Current Threshold Setting

Figure 45 shows that when the lock detection current limit feature is triggered, the device stops driving the motor and waits for five seconds to retry (In this figure, the startup failed again at retry because the current at this threshold is not sufficient to make the resynchronization).

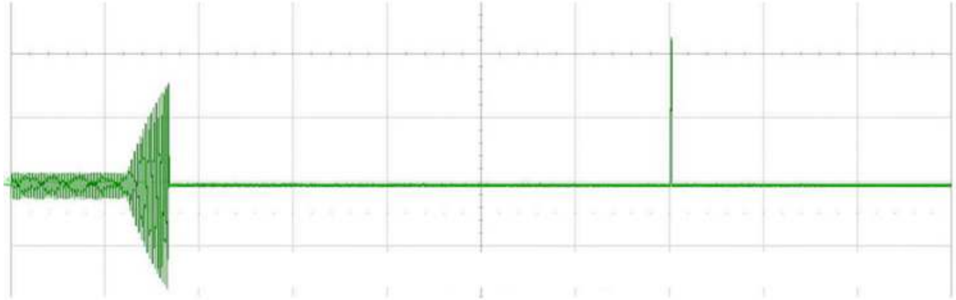


Figure 45. Lock Detection Current Limit (1.6 A) and Retry After Five Seconds

Also, the register bit will be set and the GUI indicator will turn on when the lock detection current limit is triggered.

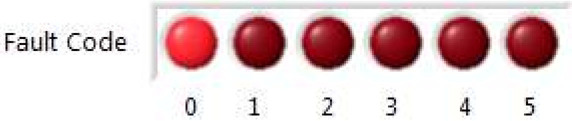


Figure 46. Fault Code for Lock Detection

Copyright © 2014–2017, Texas Instruments Incorporated

7 Startup Fine Tune

7.1 IPD

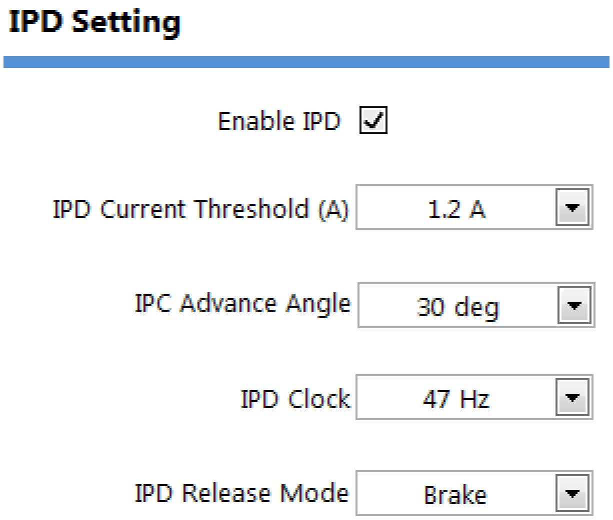
For applications where a reverse spin is not acceptable, the IPD function is an alternative way to initialize the motor. With the proper IPD setting, the motor startup is also faster.

While this function is suitable for motors with high inertia, such as heavy blades (for example: a ceiling or appliance fan), it is not suitable for motors with low inertia, such as small blades (for example: a computer fan), because the current injection will cause the motor to shake, resulting in the IPD not being accurate.

If IPD is chosen as initialization method, we need to enable IPD and configure the IPD setting section. At the same time, because the align method is not used, the AlignTime can be set to the minimum value.


AlignTime: 0.04 s

Figure 47. AlignTime Setting


IPD Setting

Enable IPD ☑

IPD Current Threshold (A): 1.2 A

IPC Advance Angle: 30 deg

IPD Clock: 47 Hz

IPD Release Mode: Brake

Figure 48. IPD Settings

7.1.1 IPD Enable and Current Threshold

IPD current threshold is selected based on the inductance saturation point of the motor. However, normally we are not able to find out this exact number by specification or calculation. We need to choose this number by experience.

A higher current has better chance to accurately detect the initial position. On the other hand, higher current will result in vibration and noise; also the current should not be higher than the maximum current the motor can handle.

Note that it is possible to not be able to find the proper settings for a particular motor for the IPD function. This is either because the current causes too much noise and vibration, or the current is not sufficient to accurately detect the initial position of the motor. In this case the align and go method should be used.

Modifying the motor can make the IPD work properly, please contact your TI representative for further advice.

Copyright © 2014–2017, Texas Instruments Incorporated

7.1.2 IPD Advance Angle

IPD advance angle is the driving angle after the initial position of the motor is identified. 30 degrees is suggested.

7.1.3 IPD Clock

IPD clock defines how fast the IPD pulses are applied. Higher inductance motors and higher current thresholds need a longer time to settle the current down, so we need set the clock at a slower time. However a slower clock makes the IPD noise louder and it lasts longer, so we suggest setting the clock as fast as possible as long as IPD current is able to settle down completely.

Looking at Figure 49, the current does not settle completely, which means the clock is too fast for this motor. This will result in IPD not being able to reliably identify the initial position of the motor.

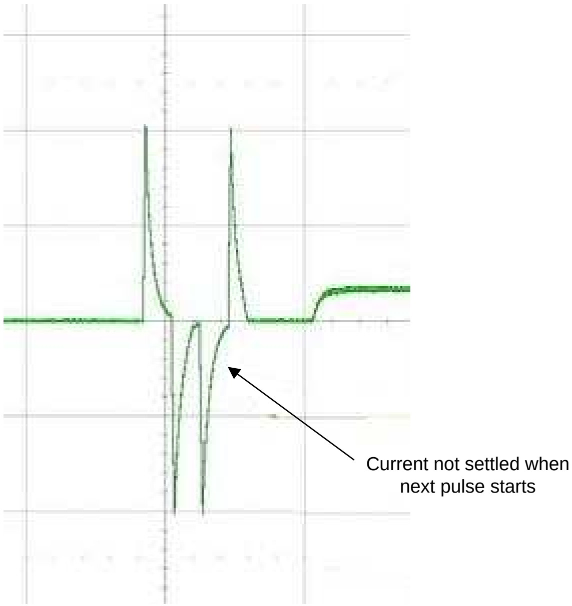


Figure 49. Current Graph

Copyright © 2014–2017, Texas Instruments Incorporated

7.1.4 IPD Release Mode

IPD release mode can be selected either as *Brake* or *Tri-state*.

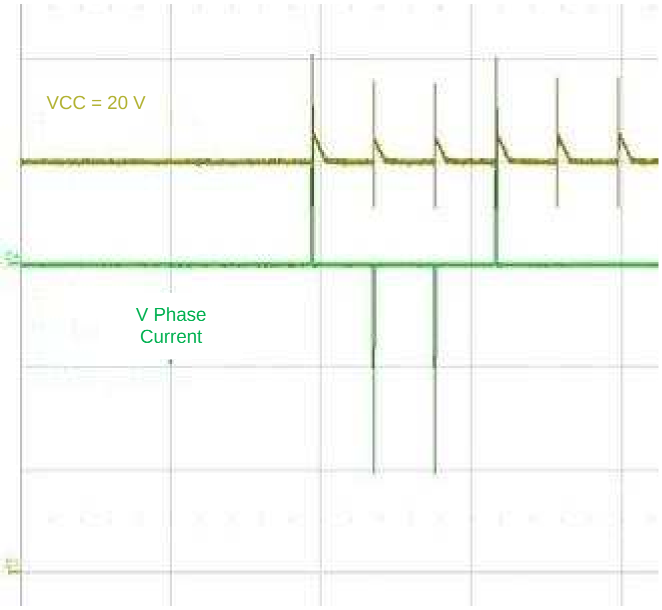


Figure 50. VCC Overshoot in Tri-State Mode

If the system input capacitor is not big enough, the *Tri-state* option will cause the VCC voltage to pump up, which should be avoided.

The advantage of *Tri-state* mode is that the current settle down time is significantly reduced.

Figure 51. Brake versus Tri-State after IPD

7.1.5 Whether to Use IPD

To decide whether to choose IPD or align and go, compare the startup difference of the two techniques. Normally, if the part of the motor that rotates is not visible and the system can tolerate reverse rotation of the motor, for example a cooling fan in a laptop or a water pump, we suggest using align and go. If the rotating part of the motor is visible to the user or if the system cannot tolerate reverse rotation of the motor, use the IPD technique.

Copyright © 2014–2017, Texas Instruments Incorporated

7.2 Improve the Align Time and Accelerate Time

The *Startup Settings* have been set to a safe value to make sure the startup was successful. However, it might not be sufficiently fast enough to meet the startup time requirement. Use the following sections to minimize the startup time.

7.2.1 Handoff Threshold Optimization

To start the motor reliably, there is a minimum speed that is required to achieve a certain BEMF voltage. A low handoff threshold causes the handoff to occur before the BEMF is at the adequate level, resulting in startup failure.

However, a high handoff threshold will result in an over shoot when the closed loop speed command is low. For example, the threshold is 100 Hz and speed command is 50 Hz, overshoot appears in the startup.

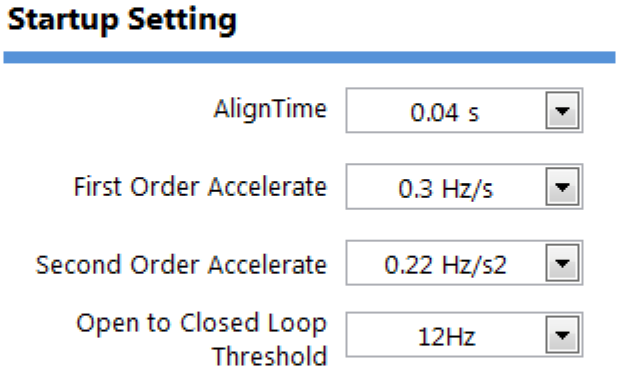
We suggest the handoff speed to be set between one-fifth and one-third of the maximum speed. The customer may also want to try a lower threshold so that the motor ramp from zero speed to full speed is faster. This is because the motor accelerates faster in closed loop. Please ensure the startup reliability at a different initial position when we set the handoff threshold lower than one-fifth of the maximum speed.

Copyright © 2014–2017, Texas Instruments Incorporated

7.2.2 AlignTime and Accelerate

Disable the transition to closed loop to fine tune the *AlignTime* and acceleration of the system so that any failures related to closed loop operation will not interfere with correctly tuning.

If IPD is implemented in the application, set the *AlignTime* to 0.04 s and set *Second Order Accelerate* to 0.22 Hz/s2. Keep increasing the *First Order Accelerate* step by step until there is startup failure; select the last value before the failure occurred.



Startup Setting

AlignTime	0.04 s
First Order Accelerate	0.3 Hz/s
Second Order Accelerate	0.22 Hz/s2
Open to Closed Loop Threshold	12Hz

Figure 52. Startup Settings

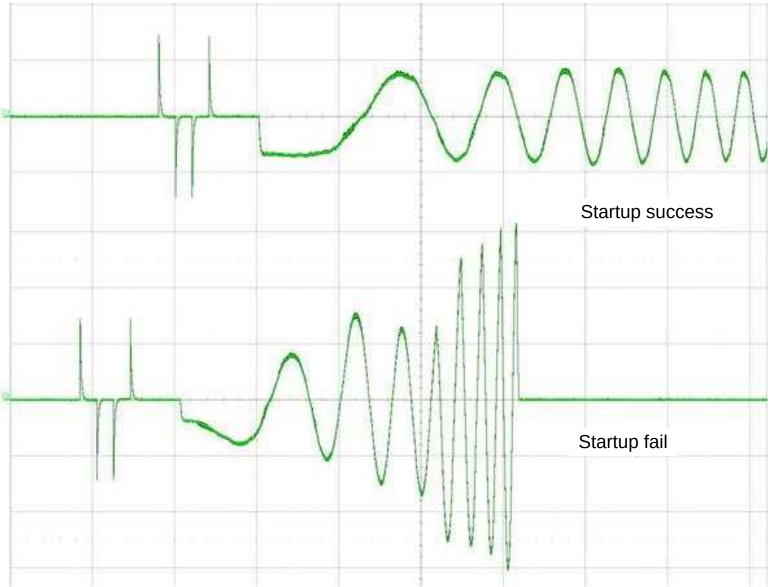


Figure 53. Startup Success versus Fail

Copyright © 2014–2017, Texas Instruments Incorporated

If the IPD is not enabled, there are two options to align and accelerate the motor:

1. The traditional align and go is to set the second order accelerate to 0.22 Hz/s2, then set the align time to be long enough that the motor can basically stop during align. Figure 54 show a sufficient align time and an insufficient align time.

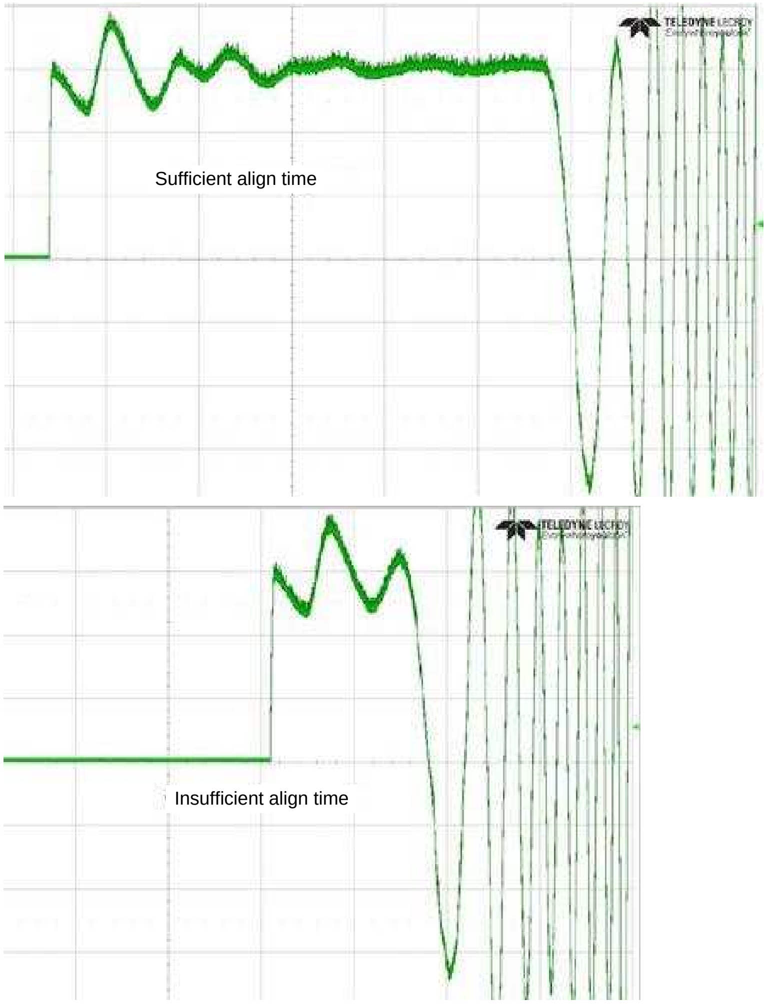


Figure 54. Sufficient versus Insufficient Align Time

After the align time is selected, increase the *First Order Accelerate* step by step until the startup fails.

Because of the complexity of the motor and the load (including friction, cogging, and initial position), the open loop setting is mainly based on experience. Many tests need to be taken to ensure the startup reliability. The simple method to verify your settings follows:

(a) First find the position where it is the most difficult to spin up the motor (the *Dead Point*).

(b) Label an initial position A on the fan, power on, look at the direction it starts to move. If the direction is counterclockwise, find another position B by tracking toward the clockwise direction. If it is clockwise, find the position B by tracking toward the counter-clockwise direction. Line up B in the prePower on, look at the direction it starts to move. If it is the same as position A, go further to find point B. If it is opposite, the *Dead Point* is between A and B.

(c) Find a point in between A and B, power on, look at the direction it starts to move. If the direction is towards B, the *Dead Point* is in between A and new point, relabel the new point B. If the direction the motor moves is towards A, the *Dead Point* is in between B and the new point, relabel the new point A.

(d) Repeat Step b. A and B become closer and closer until A and B almost overlap each other. This is the *Dead Point*.

Copyright © 2014–2017, Texas Instruments Incorporated

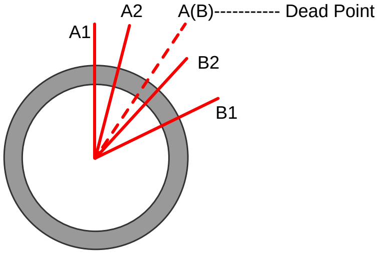


Figure 55. Dead Point

(e) Try start up at this *Dead Point* for several times, if it is safe at this point, it will be safe at other points.

In some applications, 100% successful startup at all the initial positions is not necessary. Especially when we want to have the *Dead Point* successfully startup, we will have to make the startup very slow. The DRV10983 and DRV10975 are able to retry if the first start up fails.

Make the tradeoff between startup success rate and startup time.

2. The other option is dynamic align and go, set the align time to 40 ms, set the *First Order Accelerate* and *Second Order Accelerate* to meet the startup time requirement. Here are the two steps for adjustment:
   (a) Set the second order acceleration to ‘0.22 Hz/s^2’. Make sure the startup at all positions is successful. If the startup fails at any positions, reduce the *First Order Accelerate*.
   (b) Increase the *Second Order Accelerate*, until the current has vibrations at the end of open loop. This is the indicator that the second order acceleration has reach the maximum value. If we continue to increase it, the startup will fail.

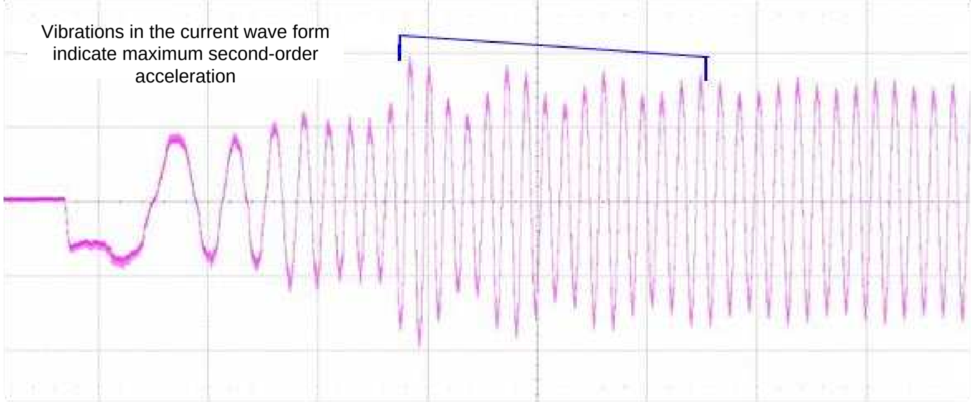


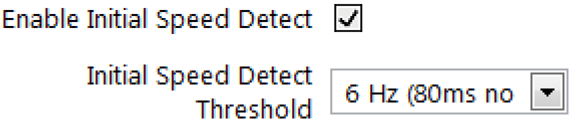
Figure 56. Current Waveform Vibrations

Copyright © 2014–2017, Texas Instruments Incorporated

7.3 Fine Tune the Before Startup Configuration

7.3.1 Initial Speed Detect (ISD)

Keep *Enable Initial Speed Detect* checked all the time. This avoids waiting for the motor to stop in the condition when motor has initial speed.



Enable Initial Speed Detect ☑

Initial Speed Detect Threshold 6 Hz (80ms no

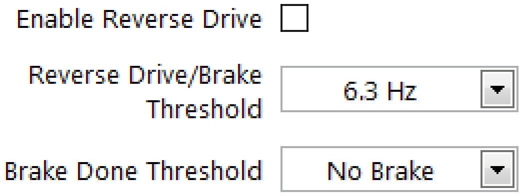
Figure 57. Initial Speed Detect Settings

The *Initial Speed Detect Threshold* is selected based on motor velocity constant and inertia. Theoretically, we should choose the threshold to be as small as possible, because we can resynchronize the motor with lower speed. However, a motor with a low velocity constant at a lower speed has a very small BEMF; it may not be able to correctly trigger the BEMF comparator, causing the resynchronization to fail. So, using a motor with Kt > 1 V/Hz, a 0.8 Hz threshold is chosen and using a motor with Kt < 100 mV/Hz, choose 6 Hz.

Copyright © 2014–2017, Texas Instruments Incorporated

7.3.2 Reverse Drive

Unless the application requires two-direction spinning or it is possible for the motor to have reserve speed before startup, the *Enable Reverse Drive* function and *Brake* function should be disabled. Refer to the datasheet DRV10983 (SLVSCP6) or DRV10975 (SLVSCP2) for detailed information.



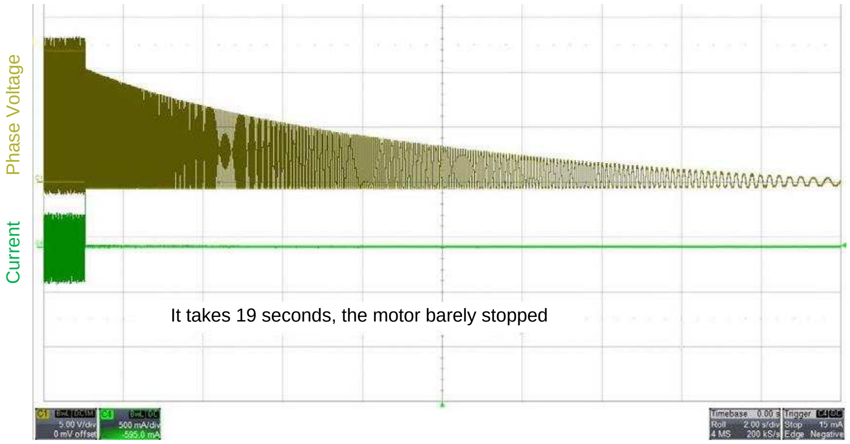
Enable Reverse Drive ☐

Reverse Drive/Brake Threshold: 6.3 Hz

Brake Done Threshold: No Brake

Figure 58. Reverse Drive and Brake Setting

The *Reverse Drive* function is very useful when an application contains a motor with big inertia and small friction. In this condition, it takes a very long time for the motor to coast down to zero speed without the *Reverse Drive* function (for example, a ceiling fan).



It takes 19 seconds, the motor barely stopped

Figure 59. Reverse Drive Waveform

Copyright © 2014–2017, Texas Instruments Incorporated

7.3.3 Brake

The *Brake* function and *Reverse Drive* function are mutually exclusive. The *Reverse Drive/Brake Threshold* needs to be reasonably low. If it is set too high, the current caused by the BEMF may damage the motor or the device.

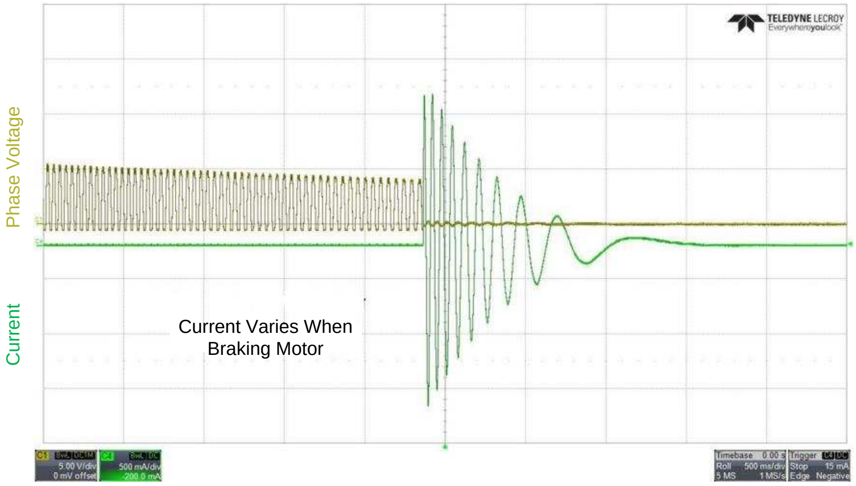


Figure 60. Brake Waveform

Brake Done Threshold determines how to finish the brake state. Bigger Kt and bigger inertia motors need a longer time while smaller Kt and smaller inertia motors need shorter time. If *Brake Done Threshold* is set too long, the startup time is unnecessarily long. If it is set too short, it is possible that the motor will not completely stop before attempting to start up again. This is especially important if we choose to use the IPD to start the motor. If the motor is not completely stopped, the IPD result will not be correct.

If the *Align&Go* is selected to start the motor, we don't need to brake the motor to a complete stop, we can even remove the brake state as long as the align state is sufficient to stop and position the motor.

7.3.4 Verify the Before Startup Settings

In order to verify these *Before Startup* functions, generate initial speed to the motor. For example, make the motor spin and control the speed command to zero, before the motor stops send the startup speed command (non-zero) again. Before the motor stops, if we change the DIR pin and send the startup speed command, we can verify the *Reverse Drive* function and the *Brake* function.

Copyright © 2014–2017, Texas Instruments Incorporated

8 Lock Detect

Put external torque on the motor and stop it. The device should be able to detect the lock condition and report the issue.

When the motor is blocked, the appropriate register bit is set and the corresponding GUI indicator is turned on.

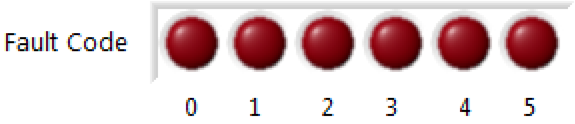
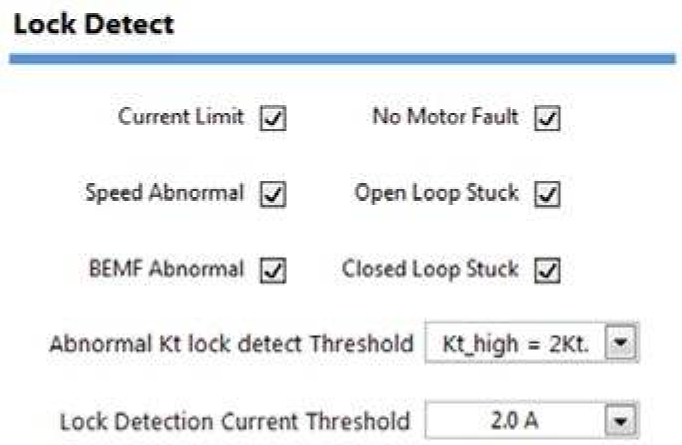


Figure 61. Fault Code Indicator for Blocked Motor

If the device is not able to effectively detect the lock condition and continues driving output current after the motor is completely blocked, enable the *BEMF Abnormal* lock detect function.



Lock Detect

Current Limit ☑
No Motor Fault ☑
Speed Abnormal ☑
Open Loop Stuck ☑
BEMF Abnormal ☑
Closed Loop Stuck ☑
Abnormal Kt lock detect Threshold: Kt_high = 2Kt.
Lock Detection Current Threshold: 2.0 A

Figure 62. Lock Detect Settings

Copyright © 2014–2017, Texas Instruments Incorporated

Sometimes enabling *BEMF Abnormal* could cause startup failure during transition from open loop to closed loop.

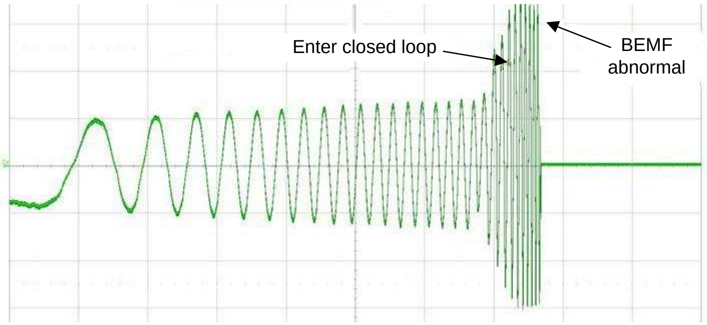


Figure 63. Startup Failure Using BEMF Abnormal

This is because during the transition, the calculated Kt is not accurate. Once the non-accurate Kt value lasts for longer than 300 ms, the device will treat it as a lock condition when actually the motor is accelerating normally. In this condition, two solutions are suggested:

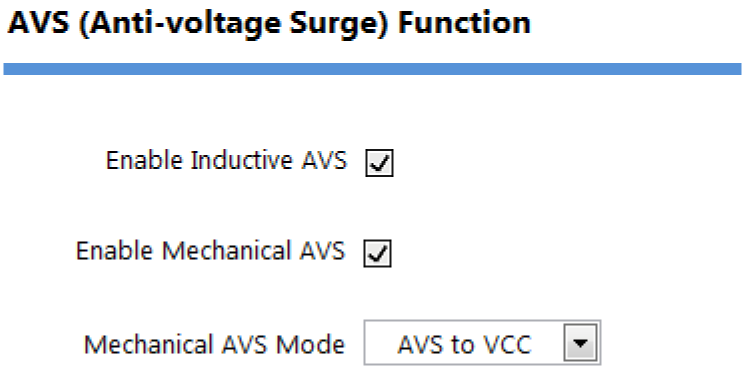
1. Increase the open to closed loop threshold, so that the transition happens at a higher speed. The device will pass the non-accurate Kt measurement period within less than 300 ms.
2. Changing the closed loop AdjMode to *Half cycle adjustment* can also help solve this issue.

After exercising the preceding configuration, if there is not a combination which can start up the motor successfully under normal conditions and can reliably detect the condition when the motor is locked, an extra MCU may be needed to detect the lock, consult a TI representative for further advice.

Copyright © 2014–2017, Texas Instruments Incorporated

9 AVS

The last step is to enable the AVS function. The AVS function prevents the voltage from surging when motor is spinning from a high speed to low speed (Mechanical AVS) and when the motor transfers from a driving state to coasting state (Inductive AVS).



AVS (Anti-voltage Surge) Function

Enable Inductive AVS ☑

Enable Mechanical AVS ☑

Mechanical AVS Mode: AVS to VCC

Figure 64. AVS Function Settings

There is a possibility that when the mechanical AVS is enabled, the motor is not able to decelerate. In this case the *Speed Cmd Buffer* is always higher than the *Speed Command*.

Speed Command (%): 60

Speed Cmd Buffer (%): 100

Figure 65. Speed Command Input

This is because:

- The programmed Kt value is wrong, it is too high. The solution is reducing the Kt value and checking whether it can be improved. If we see the difference between *Speed Command* and *Speed Cmd Buffer* getting closer, continue to reduce the Kt, until they are the same.
- The motor is operating with very low current, the Kt we have programmed is slightly higher than the correct value, but if we reduce the value by even one step, we can’t spin the motor at high speed. In this case, increase the acceleration current limit threshold (or even disable the acceleration current limit) or disable the Mechanical AVS.

If Mechanical AVS is disabled, the *Speed Command* should be controlled slow enough to prevent the VCC from pumping up. Please set the *Mechanical AVS Mode* to "AVS to VCC" if the voltage surging higher than VCC is not allowed in the application.

If VCC is less than 24 V, and the system allows VCC to go up, select "AVS to 24V", which means we protect the VCC only when VCC goes up to 24 V. Do not allow VCC to go even higher because it may damage the device.

Note that motor decelerating is slower by enabling the mechanical AVS.

Copyright © 2014–2017, Texas Instruments Incorporated

Revision D History

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

Changes from C Revision (June 2015) to D Revision	**Page**
• Changed "320 ns or 280 ns" requirement for a 24-V VCC application to 400 ns	12
• Changed 240 ns requirement for a 12-V VCC application to 360 ns	12
• Changed Figure 16	12

Revision C History

Changes from B Revision (January 2015) to C Revision	**Page**
• Changed motor coasting description	5
• Added information on how motor inertia affects the align time	7
• Added the status of the *Open Loop Current rate* when configuring the startup settings	11
• Added LR time constant introduction	13
• Added *Open Loop Current Setting* introduction	22

Revision B History

Changes from A Revision (October 2014) to B Revision	**Page**
• Added DRV10975 content and references	1
• Deleted *Question and Answer*	38

Revision A History

Changes from Original (July 2014) to A Revision	**Page**
• Deleted all DRV10975 content and references	1
• Changed 160 ns to 240 ns in PWM Output Options	12
• Changed *Software Current Limit* to *Acceleration Current Limit* globally, including images	17
• Changed *Hardware Current Limit* to *Lock Detection Current Limit* globally, including images	18

Copyright © 2014–2017, Texas Instruments Incorporated

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated (‘TI”) technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, “TI Resources”) are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI’s provision of TI Resources does not expand or otherwise alter TI’s applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED “AS IS” AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI’s standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2017, Texas Instruments Incorporated