DRV8809/8810 User's Guide

This document is provided as a supplement to the DRV8809/8810 data sheet. It details the steps necessary to properly interface the device in most motion control and power management applications in need of Bipolar Stepper, Brushed DC motion control and Buck DC-DC regulators.

Contents

1 General Description

The DRV8809/8810 motion control and power management device provides four fully configurable H bridges plus three DC to DC converters, allowing most small to medium scale printing application's motion and power management segments, to be embedded into a single unit.

The DRV8809/8810 is fully configurable and operates by a combination of carefully selected control signals and two serial ports. The serial ports offer a window to four internal registers which contain all the configuration bits needed during setup and normal operation.

1.1 DC-DC Converters

Three totally independent DC-DC converters, working as buck regulators, provide power to other segments on the application. Two of these three converters can be nested together for increased current handling.

This document details the steps necessary to configure and control the three converters. It also contains information on how to select the external components necessary to form the Buck regulator.

1.2 H-Bridge Motor Drivers

Four totally independent H bridges allow for DC and stepper motor interfacing. These H bridges can be nested together to form larger bridges with higher current handling capacity.

Throughout this document, nested bridges are referred to as larger DC motor bridges, following the next notation:

Small DC Motor: utilizes one H bridge and can handle up to 800 mA of current

Large DC Motor: two H bridges nested together, with a current capacity of 1.6A

Ultra Large DC Motor: The four H bridges nested together, with a current capacity of 3.2A

The following figure shows a representation of the three different DC motor styles the DRV8809/8810 can drive when configured in three out of the eight different modes of operation. The representation applies to the other modes of operation containing DC motors.

4 Small DC Motors
(800mA/phase)

2 Large DC Motors
(1.6A/phase)

1 Very Large DC
Motor (3.2A/phase)
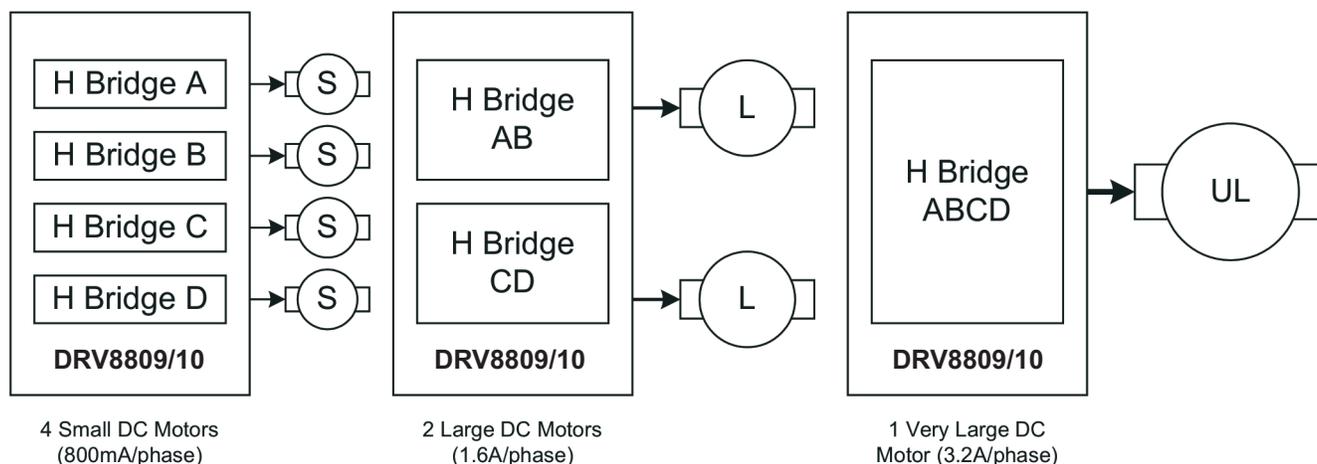
This document details the steps necessary to configure and control the four H Brdiges when configured to drive DC or stepper motors. It also contains information on how to select the external Sense Resistor an configure the maximum current parameter.

1.3 Serial Ports and Control Signals

The two serial ports on the DRV8809/8810 serve a dual purpose. First, when the device is being configured (signal nSLEEP is low), the Serial Port SPI_AB works as an access point to the SETUP and EXTENDED register. SPI_CD port has no use during nSLEEP low.

Second, if configured to drive stepper motors, after nSLEEP becomes high, the SPI_AB and SPI_CD port pins function as synchronous communication ports that write to the STEPPER_AB and STEPPER_CD registers, respectively.

If on the other hand the device was configured to operate DC motors during nSLEEP low, when nSLEEP switches to high, the SPI_AB and SPI_CD ports are disconnected, and the respective pins become enables and phases for the DC motors being controlled.

A hybrid of the previously discussed also become true if driving stepper and DC motors at the same time. If such is the case, one SPI port remains, while the other switches to its enable and phase function.

This document details how to utilize the Serial Ports and the Control Signals. It also explains how each pin behaves during the setup process and normal operation

2   Important Signals

2.1   *SPI Port AB (CLK_AB, DATA_AB, STROBE_AB)*

The Synchronous Peripheral Interface AB, SPI_AB, allows for communication with three of the four DRV8809/8810 registers. The SETUP register contains bits to configure Motion Control parameters such as Operation Mode, H Bridge Amplifier Gain, Oscillator Frequency, and Time Blanking. This SETUP register also contains bits to configure the DC to DC converter oscillator frequency and individual enable bits for each one of the three converters.

The EXTENDED Register contains bit to configure different troubleshooting modes. Per example, fail modes can be disabled, and different status signals can be made available through the LOGIC_OUT and TH_OUT pins.

The STEPPER_AB register contains bits to configure stepper motor control. Such bits specify torque settings, current settings, and phase direction for each of H bridges A and B. The STEPPER_AB register can only be accessed if the SETUP register was configured for the DRV8809/8810 to work as a stepper controller, and while nSLEEP is high.

The SPI_AB port becomes available whenever the nSLEEP signal is low. It is also available whenever phases A and B are configured to work as a stepper controller and the nSLEEP pin is high. If nSLEEP is high and windings A and B are controlling DC motors, SPI_AB port is not available as these pins are used to control Enable and Phase functionality on H bridge drivers.

2.2   *SPI Port CD (CLK_CD, DATA_CD, STROBE_CD)*

The Synchronous Peripheral Interface CD, SPI_CD, allows for communication with the last of the four DRV8809/8810 registers. The STEPPER_CD register contains bits to configure stepper motor control. Such bits specify Torque settings, Current settings, and Phase direction for each of windings C and D. The STEPPER_CD register can only be accessed if the SETUP register was configured for the DRV8809/8810 to work as a dual stepper controller and nSLEEP is high.

2.3   *DCDC_MODE*

DCDC_MODE is a pin signal used to control whether DC to DC converters B and C work as independent converters or work in a parallel configuration extremely useful when in need of larger currents than 1.5A per converter. DCDC_MODE is sampled when InReset becomes high and at power up.

If DCDC_MODE is high after coming out of reset, the DC to DC converter B controls DC to DC converter C. In other words, DC/DC Converter B and C are working in parallel mode. In this case, the FBB (Feedback for DC to DC converter channel B) is used to determine output regulated voltage. FBC is ignored. It is also important to understand that the SW B bit at the SETUP register disables both DC to DC converter channels B and C, when written with a 1. Enabling both converters can then be achieved by writing a 0 at the same SW B bit.

If DCDC_MODE is low after coming out of reset, each DC to DC converter channel, B and C, behaves as totally independent converters. Their respective FBx pin must be used to set the desired regulated output voltage.

2.4   *CSELECT*

CSELECT allows for DC to DC converters to be enabled either automatically after reset or manually by writing the respective bit at the SETUP register. CSELECT is sampled after coming out of reset (i.e. InReset is pulled high and nORT goes high).

If CSELECT is high after reset, all converters are enabled and operational. Disabling and enabling converters has then to be arranged by writing 1's for disable and 0's for enable at the respective bit on the SETUP register.

If CSELECT is low after reset, all converters are disabled. Enabling is achieved by writing a 1 to each enable bit on the SETUP register.

2.5   InReset and nORT

The DRV8809/8810's InReset input signal offers a way of resetting the entire device, clearing all internal registers. InReset's counterpart signal is the output nORT. nORT goes low whenever InReset goes low, and as stated next.

nORT is digitally filtered so that pulses shorter than 5 us on InReset, are ignored. After the 5 us, the nORT pin goes low. It remains low until InReset goes high again and after a 300 ms delay. Every DC-DC converter and H-Bridge driver is disabled while nORT is low.

nORT can be asserted as low, while InReset maintains a high level, if there is an internal fault such as over current protection.

2.6   nSLEEP

nSLEEP induces low power mode. It also serves the dual purpose of switching from setup mode and operational mode. These two modes differentiate on how the SPI ports and control signals behave.

When nSLEEP is low, the SPI_AB port is enabled and allows the user to write to the SETUP and EXTENDED registers. SPI_CD port has no function while nSLEEP is low. All H bridges are disabled, hence there is no motion control.

When nSLEEP becomes high, the DRV8809/8810 enters the operational mode with all H bridges being functional. However, their operation depends on how the SETUP register was configured. If the DRV8809/8810 was configured to work as dual steppers, SPI_AB and SPI_CD are available. There are no enable or phase pins available during this scenario.

If the DRV8809/8810 was configured to work as a stepper and DC motor controller, the SPI_AB port exists as a communication port, while SPI_CD port switches to phase and enable control signals.

Finally, if the DRV8809/8810 is to control DC motors alone, when nSLEEP is high, SPI_AB and SPI_CD ports do not exist, while their respective pins behave as enable and phase control signals.

A very important consideration the user must follow, is that the nSLEEP and STROBE_AB signal must not go high at the same time. A delay of at least 50 us, should follow the nSLEEP rising edge, before the STROBE_AB is made high.

Details on how each pin behaves depending on configuration mode and nSLEEP is provided later on this document.

2.7   Enables and Phases

Each H bridge driver has its own ENABLE_Sx and PHASE_Sx pin. However, said pins only become functional when the respective DC motor control scheme is enabled. ENABLE_Sx and PHASE_Sx pins do not exist for stepper control modes. Enabling and Phase control for steppers is achieved by writing to the STEPPER_xx register.

When working as Enables and Phases, these pins are used to control winding on/off state and current direction. Applying a PWM signal to the respective Enable pin controls the DC motor speed in a directly proportional fashion. That is, higher duty cycle percentages yield faster speed. Enable is asserted high.

Phase pins control the current direction. If the PHASE_Sx pin is high, current flows from positive terminal to negative terminal. The opposite is true if the PHASE_Sx pin is pulled low. The following table shows the way ENABLE_Sx and PHASE_Sx pins are arranged according to operational mode.

CONFIGURATION	PIN NUMBER							
	50	51	54	55	59	60	61	62
Two Steppers	-	STROBE_CD	-	STROBE_AB	DATA_CD	CLK_CD	DATA_AB	CLK_AB
Stepper and Large DC	-	ENABLE_LCD	-	STROBE_AB	-	PHASE_LCD	DATA_AB	CLK_AB
Stepper and two Small DC	ENABLE_SD	ENABLE_SC	-	STROBE_AB	PHASE_SD	PHASE_SC	DATA_AB	CLK_AB

CONFIGURATION	PIN NUMBER							
	50	51	54	55	59	60	61	62
Large DC and two Small DC	ENABLE_SD	ENABLE_SC	-	ENABLE_LAB	PHASE _SD	PHASE_SC	-	PHASE_LAB
Two Large DC	-	ENABLE_LCD	-	ENABLE_LAB	-	PHASE_LCD	-	PHASE_LAB
Four Small DC	ENABLE_SD	ENABLE_SC	ENABLE_SB	ENABLE_SA	PHASE_SD	PHASE_SC	PHASE_SB	PHASE_SA
Large Stepper	-	STROBE_CD	-	STROBE_AB	DATA_CD	CLK_CD	DATA_AB	CLK_AB
Ultra Large DC	-	-	-	ENABLE_UL	-	-	-	PHASE_UL

> **NOTE:** Through this document and the DRV8809/8810's data sheet, ENABLE_Sx may be used for the enable function when dealing with a small DC motor, ENABLE_Lxx may be used for the enable function when dealing with a large DC motor and ENABLE_ULABCD may be used for the enable function when dealing with the Ultra Large DC motor. The same applies to PHASE_Sx

2.8 RSENSE Pins

To monitor current and disable drivers if current exceeds a predetermined value, a Sense Resistor must be placed on the respective RSENSE pin (RSA, RSB, RSC and RSD). However, when two H bridges are connected in parallel, as when driving a Large DC, there is need for only one Sense Resistor, not two. In this case, however, both RSENSE pins must be connected together and then into the Sense Resistor. The following table shows which pin has to be connected to a Sense Resistor with regards to a particular operational mode.

CONFIGURATION	PIN NUMBER							
	4	5	10	11	38	39	44	45
Dual Stepper	RSA	RSA	RSB	RSB	RSC	RSC	RSD	RSD
Stepper + Large DC	RSA	RSA	RSB	RSB	RSC	RSC	RSD	RSD
Stepper + 2 Small DC	RSA	RSA	RSB	RSB	RSC	RSC	RSD	RSD
Large DC + 2 Small DC	RSA	RSA	RSLAB	RSLAB	RSCD	RSCD	RSD	RSD
Dual Large DC	RSLAB	RSLAB	RSLAB	RSLAB	RSCD	RSCD	RSCD	RSCD
Quad Small DC	RSA	RSA	RSB	RSB	RSC	RSC	RSD	RSD
Large Stepper	RSLAB	RSLAB	RSLAB	RSLAB	RSCD	RSCD	RSD	RSD
Ultra Large DC	RSLABCD	RSLABCD	RSLABCD	RSLABCD	RSLABCD	RSLABCD	RSLABCD	RSLABCD

For diagrams on how to connect the Sense Resistors at the chip level, please refer to the DRV8809/8810's data sheet.

2.9 Logic Out

As a very flexible troubleshooting tool, the LOGIC_OUT pin offers a window into many variables such as Over Current Protection, Over Voltage Protection, and Under Voltage protection status, Motor Over Current Protection, Thermal Shutdown, Revision Number, Vendor Information, Internal Oscillator, etc. For more information on how to extract the information provided at LOGIC_OUT, please refer to the "Monitoring with the EXTENDED Register" section.

2.10 Switching Regulator Outputs (OD_A, OD_B, OD_C)

Each DC to DC converter channel has an associated output, OD_A, OD_B and OD_C. Each pin connects directly to the Buck converter circuitry consisting of a Schotky diode, high current inductor, feedback resistive network and a large filter capacitor. Selecting inductors and capacitors is treated later in this document.

2.11 VREFxx

To control current through the H bridge drivers, a current control mechanism has been included for each H bridge in the DRV8809/8810. The VREFxx, for Reference Voltage, is a crucial element on this mechanism.

Current flowing through the Sense Resistor in each H bridge, results in a voltage drop across the said component. This voltage drop is compared with the Reference Voltage VREFxx. When the Sense Resistor voltage is larger than VREFxx, the H bridge is disabled for a determined period of time. After this time, the bridge is enabled once again. This process continues, generating a chopped current waveform.

VREFAB is the Reference Voltage for H bridges A and B, while VREFCD is the Reference Voltage for H bridges C and D

3 Configuring the Motion Control Engine

The DRV8809/8810 can control steppers and DC motors in a variety of sizes as described by the device's data sheet. To configure the four H bridges for one operation mode in particular, configure the Blanking Time, Gain and Oscillator Frequency, the SETUP register must be accessed. Next page shows a diagram of the important bits in the SETUP Register, pertinent to motion control. Following the next set of steps, allows for the SETUP Register to be updated accordingly.

3.1 Steps to write to the DRV8809/8810's SETUP Register

1. InReset must pulled high
2. nSLEEP must be pulled low, although it can be made low TSS_MIN prior to the strobing signal's rising edge (refer to the data sheet for more information on TSS_MIN)
3. Setup register is only accessible through the SPI_AB port, so lines STROBE_AB, CLK_AB and DATA_AB are used.
4. STROBE_AB should be low at the start of the message.
5. CLK_AB should transition at least 16 times before STROBE_AB goes high.
6. Data at DATA_AB is latched during the CLK_AB's falling edge.
7. Data at DATA_AB must read low during the STR_AB's rising edge.

> **NOTE:** Failure to comply with item 7, results in data latched on the EXTENDED Setup Register, instead of the SETUP Register.

3.1.1 SETUP Register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSCM Fx				CD Gain	AB Gain				TBLNK CDx		TBLNK ABx		MSx		
w-(0)	w-(0)			w-(0)	w-(0)				w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)

OSCM Fx   Bits 15-14   Motor Oscillator Frequency. Select oscillator frequency for the four H Bridges.

00	800 KHz
01	400 KHz
10	1.06 MHz
11	1.6 MHz

CD Gain   Bit 11   Motor C and D H Bridge Gain.

0	1/10
1	1/20

AB Gain   Bit 10   Motor A and B H Bridge Gain.

0	1/10
1	1/20

TBLNK CDx   Bits 6-5   Tblank for DC Motor H Bridge drivers C and D

00	6.25 us
01	7.50 us
10	3.75 us
11	5.00 us

TBLNK ABx   Bits 6-5   Tblank for DC Motor H Bridge drivers A and B

00	6.25 us
01	7.50 us
10	3.75 us
11	5.00 us

MSx   Bits 2-0   Motor Select. Select one of the possible eight ways to configure the four H Bridges.

000	Two Steppers
001	Stepper and Large DC Motor
010	Stepper and two Small DC Motors
011	Large DC Motor and two Small DC Motors
100	Two Large DC Motors
101	Four Small DC Motors
110	Large Stepper (Function Test Only)
111	Ultra Large DC Motor

NOTE: Shaded bits pertain to the DC-DC converter setup. They are discussed later in this document.

NOTE: W stands for Write. (0) stands for a default state of LOW after power up or reset.

3.2 *Controlling Four Small DC Motors*

Each one of the four H bridges can work independently from each other when the DRV8809/8810 is configured as a four small DC motor controller, by sending the propper serial command through the SPI_AB port. Each H bridge then posseses a phase and an enable line. Since these lines are obviously shared with the SPI_AB and SPI_CD ports , it is important to understand the chronology of events which must take place for the DRV8809/8810 to be configured as a 4 small DC motor controller.

3.2.1 Configuring as a Four Small DC Motor Controller:

1. InReset pin must be high for the device to be placed on an operational mode. If the InReset pin is low, the following steps are ignored.
2. Pull the nSLEEP pin low. This disables all H bridges and clears internal registers. SPI_AB port pins now behave with their respective synchronous communication port's functionality.
3. Send a message through the SPI_AB Port with the Motor Select three bits as 0x05 (0bxxxxxxxxxxxxx101). This is a good time to configure Time Blanking, Motor Gain and Oscillator Frequency if need be.
4. Pull the nSLEEP pin high. This enables H bridges and disables the SPI_AB and SPI_CD ports while switching the respective pins to behave with their Enable/Phase functionality.
5. Modify the VREFAB and VREFCD to the desired level. This controls the amount of current that is supplied by the DRV8809/8810 into the DC motors.
6. Modify EnableX and PhaseX pins according to the desired motion function.

| NSLEEP = Low | NSLEEP = High |
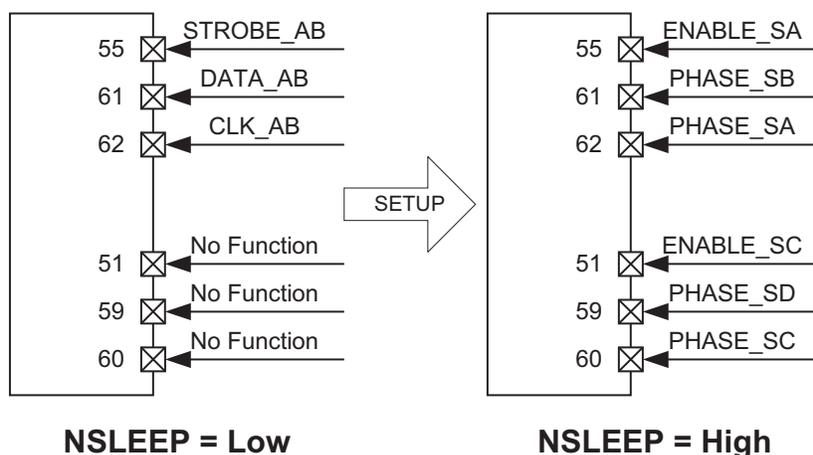
NOTE: It is very important that nSLEEP and STROBE_AB are not made high at the same time. If this were to happen, the STROBE_AB would still work as the SPI Latch signal. Latched contents are unknown. Wait at least 50 us before rising STROBE_AB (ENABLE_SA by then) after nSLEEP goes high.

---

3.3 *Controlling Two Large DC Motors*

Another feature found on the DRV8809/8810, is its ability to merge H bridges into a single, but larger, bridge. Such capability offers enhanced current handling, which translates to larger DC motor driving. In said mode, H bridges A and B are bridged together to form H bridge AB, while C and D form H bridge CD. Important things to have under consideration:

1. ENABLE_SA pin takes precedence over ENABLE_SB pin. ENABLE_SB pin is then ignored. ENABLE_SA pin becomes ENABLE_LAB pin.
2. PHASE_SA pin takes precedence over PHASE_SB pin. PHASE_SB pin is then ignored. PHASE_SA pin becomes PHASE_LAB pin.
3. Only one Sense Resistor is needed, RSB is to be a non-populate component, but the RSB pin must be tied together with the RSA pin.
4. VREFAB still works as the Reference voltage.

5. Other bits on the SETUP register still behave as before.

3.3.1 Configuring as a Two Large DC Motor Controller

1. InReset pin must be high for the device to be placed on an operational mode. If the InReset pin is low, the following steps are ignored.
2. Pull the nSLEEP pin low. This disables all H bridges and clears internal registers. SPI_AB port pins now behave with their respective synchronous communication port's functionality and write to the SETUP or EXTENDED register.
3. Send a message through the SPI_AB with the Motor Select three bits as 0x04 (0bxxxxxxxxxxxxx100). This is a good time to configure Time Blanking, Motor Gain and Oscillator Frequency if need be.
4. Pull the nSLEEP pin high. This enables H bridges and disables the SPI_AB and SPI_CD ports for the respective pins to behave with their Enable/Phase functionality.
5. Modify the VREFAB and VREFCD to the desired level. This controls the amount of current that is supplied by the DRV8809/8810 into the DC motors.
6. Modify ENABLE_Lxx and PHASE_Lxx pins according to the desired motion function.

NSLEEP = Low   **NSLEEP = High**
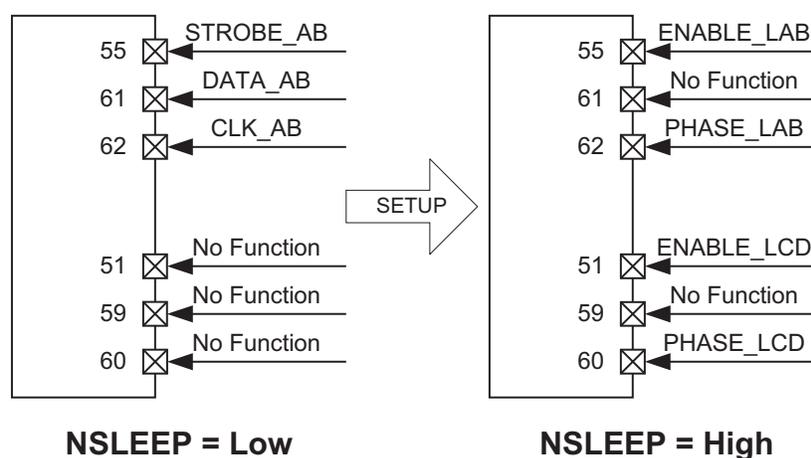
3.4 Controlling One Very Large DC Motor

The four H bridges can be nested together in parallel to form the strongest DC motor control configuration. Although only one motor can be controlled, the DRV8809/8810 can easily supply up to 3.2A continuous. In this mode, H bridges A, B, C and D are bridged together to form Ultra Large H bridge ABCD. Important things to consider are:

1. ENABLE_SA pin takes precedence over ENABLE_SB, ENABLE_SC and ENABLE_SD pins. These last three pins are then ignored. ENABLE_SA pin becomes ENABLE_ULABCD pin.
2. PHASE_SA pin takes precedence over PHASE_SB, PHASE_SC and PHASE_SD pins. These last three pins are then ignored. PHASE_SA pin becomes PHASE_ULABCD pin.
3. Only one Sense Resistor is needed, RSB, RSC and RSD are to be non-populate components, but the RSB, RSC and RSD pins must be tied together with the RSA pin.
4. VREFAB and VREFCD should be tied together.
5. Other bits on the SETUP register still behave as before.

3.4.1 Configuring as a Very Large DC Motor Controller:

1. InReset pin must be high for the device to be placed on an operational mode. If the InReset pin is low, the following steps are ignored.
2. Pull the nSLEEP pin low. This disables all H bridges and clears internal registers. SPI_AB port pins now behave with their respective synchronous communication port's functionality.
3. Send a message through the SPI_AB port with the Motor Select three bits as 0x07 (0bxxxxxxxxxxxxx111). This is a good time to configure Time Blanking, Motor Gain and Oscillator Frequency if need be.

4. Pull the nSLEEP pin high. This enables H bridges and disables the SPI_AB and SPI_CD ports for the respective pins to behave with their Enable/Phase functionality.
5. Modify the VREFAB and VREFCD to the desired level. This controls the amount of current that is supplied by the DRV8809/8810 into the DC motor.
6. Modify ENABLE_ULABCD and PHASE_ULABCD pins according to the desired motion function.
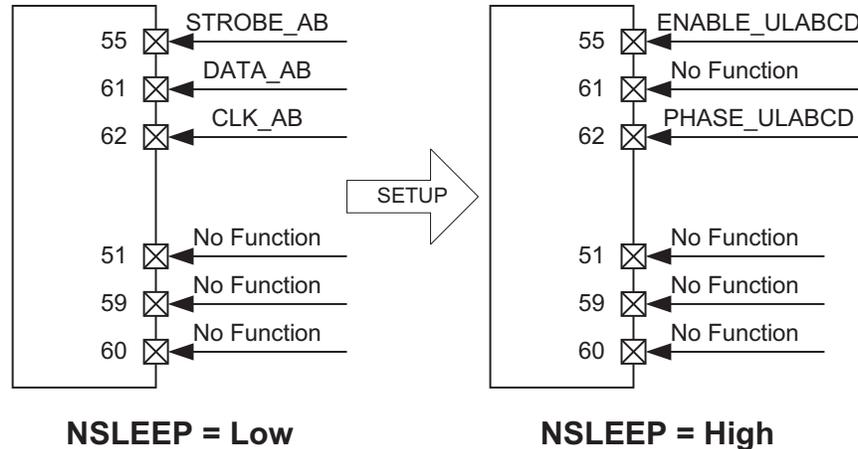
3.5 STEPPER Registers

Controlling DC motors is a matter of setting and clearing the respective ENABLEx and PHASEx pins. Although this technique works when dealing with steppers, the DRV8809/8810 makes it easier to control this motor style by integrating a powerful serial communication indexer. By sending serial command packets, each step is carefully controlled. Torque, Phase polarity and Decay Mode for both phases are integrated on a single packet.

The DRV8809/8810 can control up to two bipolar stepper motors when configured to do so (000 on the SETUP register Motor Select bits). Each stepper motor has its own synchronous serial port, SPIA and SPIB. Internal to the DRV8809/8810, each SPI port shift the incoming data into a respective STEPPER_xx register, STEPPER_AB and STEPPER_CD. The following picture shows the bit structure found on the STEPPER_xx registers.

STEPPER_AB

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PHASE A	CURRENTx A				DECAYx A		PHASE B	CURRENTx B				DECAYx B		TORQUEx	
w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)

STEPPER_CD

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PHASE C	CURRENTx C				DECAYx C		PHASE D	CURRENTx D				DECAYx D		TORQUEx	
w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)

PHASE A(C)   Bit 15   Specifies current flow for phase A (STEPPER_AB) or phase C
                                 (STEPPER_CD).
                                 0   Current flows from OUTA(OUTC)- to OUTA(OUTC)+
                                 1   Current flows from OUTA(OUTC)+ to OUTA(OUTC)-

CURRENTx A(C)   Bits   Used to control current level at the winding.
                        14-11   Divides the selected current setting in one of 16 different levels

DECAYx A(C)   Bits   Controls Decay Mode timing for H Bridge A (STEPPER_AB) or H Bridge C
                        10-9   (STEPPER_CD)
                                 00   12.5% (Do not use)
                                 01   37.5% (Do not use)
                                 10   75%
                                 11   Fast Decay

PHASE B(D)   Bit 8   Specifies current flow for phase B (STEPPER_AB) or phase D
                                 (STEPPER_CD).
                                 0   Current flows from OUTB(OUTD)- to OUTB(OUTD)+
                                 1   Current flows from OUTB(OUTD)+ to OUTB(OUTD)-

CURRENTx B(D)   Bits   Used to control current level at the winding.
                        7-4   This device has a known issue in stepper motor current setting accuracy
                                 RECOMMENDED (Set to 1111)

DECAYx B(D)   Bits   Controls Decay Mode timing for H Bridge B (STEPPER_AB) or H Bridge D
                        3-2   (STEPPER_CD)
                                 00   12.5% (Do not use)
                                 01   37.5% (Do not use)
                                 10   75%
                                 11   Fast Decay

TORQUEx   Bits   Controls Current Level for H Bridge B (STEPPER_AB) or H Bridge D
                        1-0   (STEPPER_CD)
                                 00   50%
                                 01   70%
                                 10   85%
                                 11   100%

3.6 *Controlling Two Stepper Motors*

Inside the DRV8809/8810, the necessary logic to control two steppers is embedded so that said control can be attained by sending consecutive serial commands through SPI_AB and SPI_CD ports. Combinations of stepper and DC motors are also attainable, but in this section we discuss the configuration mode 0x00, where the two steppers are enabled.

In this configuration mode, the SPI port lines behave like CLK_AB/CLK_CD as the clocking signal, STROBE_AB/STROBE_CD as the strobing signal and DATA_AB/DATA_CD as the data signal. Enable and phase pins are not available. VREFAB and VREFCD pins are used to control current.

3.6.1 Configuring as a Two Stepper Motor Controller:

1. InReset pin must be high for the device to be placed on an operational mode. If the InReset pin is low, the following steps are ignored.
2. Pull the nSLEEP pin low. This disables all H bridges and clears internal registers. SPI_AB port pins now behave with their respective synchronous communication port's functionality.
3. Send a message through the SPI_AB port with the Motor Select three bits as 0x00 (0bxxxxxxxxxxxxx000). This is a good time to configure Time Blanking and Oscillator Frequency if need be. Motor Gain does not play a role while controlling stepper motors.
4. Pull the nSLEEP pin high. This enables H bridges and enables the SPI_AB and SPI_CD ports for the respective pins to behave with their clock, strobe and data functionality.
5. Modify the VREFAB and VREFCD to the desired level. This controls the amount of current that is supplied by the DRV8809/8810 into the stepper windings.
6. Start sending serial commands to the respective STEPPER_xx register. Depending on how the PHASEx, TORQUEx, DECAYx and CURRENTx bits are programmed, the stepper moves accordingly.

NSLEEP = Low   **NSLEEP = High**
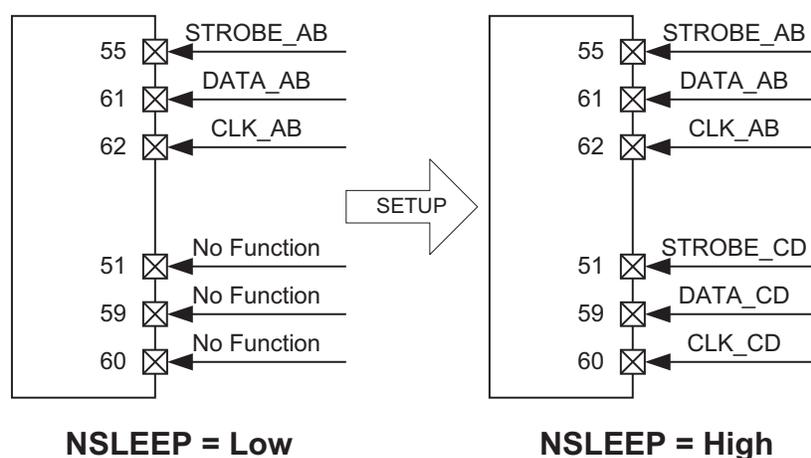
3.7 Controlling One Stepper and Two small DC Motors

In case both an stepper and DC motors are to be controlled, a combination mode allows for one stepper and two small DC motors to be wired into the device. Another mode offers the possibility of controlling one stepper and one large DC motor. In either case, setup and operation is very similar to the instructions that follow.

In this configuration mode, the SPI_AB port lines behave like Clock, Strobe and Data, while the SPI_CD port is not available, giving their functionality to ENABLE_Sx and PHASE_Sx.

3.7.1 Configuring as a One Stepper and Two DC Motor Controller:

1. InReset pin must be high for the device to be placed on an operational mode. If the InReset pin is low, the following steps are ignored.
2. Pull the nSLEEP pin low. This disables all H bridges and clears internal registers. SPI_AB port pins now behave with their respective synchronous communication port's functionality.
3. Send a message through the SPI_AB port with the Motor Select three bits as 0x20 (0bxxxxxxxxxxxxx010). This is a good time to configure Time Blanking and Oscillator Frequency if need be. Motor Gain applies only for H bridges C and D.
4. Pull the nSLEEP pin high. This enables H bridges and enables the SPI_AB port for the respective pins to behave with their clock, strobe and data functionality. SPI_CD port is unavailable, with its respective pins behaving with the enable and phase functionality.
5. Modify the VREFAB and VREFCD to the desired level. This controls the amount of current that is supplied by the DRV8809/8810 into the motor windings.
6. Start sending serial commands to the STEPPER_AB register. Depending on how the PHASEx, TORQUEx, DECAYx and CURRENTx bits are programmed, the stepper moves accordingly.

7. Modify ENABLE_Sx and PHASE_Sx pins according to the desired motion function.
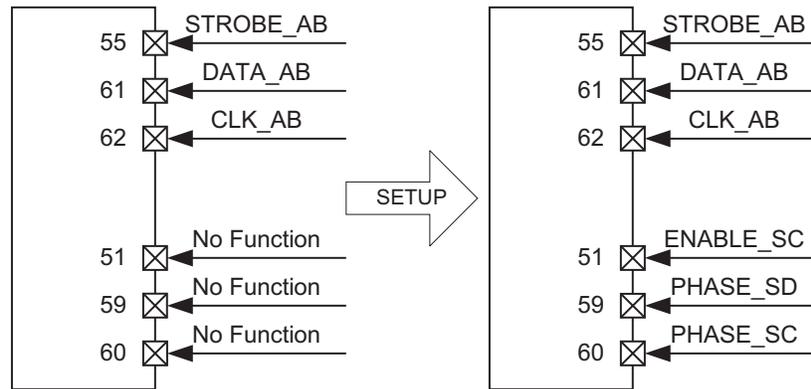
3.8 Other Motion Control Configurations

Although not all motion control modes have been discussed in this document, their respective operations are very similar to the ones previously discussed. Always remember to:

1. Pull nSLEEP low at first to configure the SETUP register. Failing to configure the SETUP register at startup, results in the DRV8809/8810 working as a dual stepper controller, which is the default.
2. Motors can only be controlled after the nSLEEP line has returned to a high state. Its low state disables all H bridges.
3. SPI ports are available when the nSLEEP pin is held low. Registers being accessed are SETUP and EXTENDED. When nSLEEP goes high, SPI ports are only enabled if the respective stepper was configured to be operational. In this case, SPI ports access the STEPPER_xx registers.
4. InReset low disables the entire device.

4 Configuring and Controlling the DC to DC switching regulators

The DRV8809/8810 contains three DC-DC converters which can be controlled either automatically at reset, or during run time by writting to the SETUP register. After coming out reset, the DCDC_MODE and CSELECT pins are sampled. The following table shows how the regulators start up depending on the previously mentioned signals.

CSELECT	DC_MODE	DC-DC CONVERTER A	DC-DC CONVERTER B	DC-DC CONVERTER C
0	0	TURNED OFF	TURNED OFF	TURNED OFF
0	1	TURNED OFF	TURNED OFF	TURNED OFF
1	0	TURNED ON	TURNED ON (parallel mode)	
1	1	TURNED ON	TURNED ON	TURNED ON

Once the device comes up, regulators can be enabled or disabled independently of each other by writting to the SETUP register.

> **NOTE:** Each DC-DC converter channel behaves independently with the exception of channels B and C whild configured with parallel drive. In this scenario, C reacts with B. All control signals to C are then ignored.

4.1 Steps to Enable/Disable DC-DC converter channels by writing to the SETUP Register

1. InReset must pulled high
2. nSLEEP must be pulled low, although it can be made low TSS_MIN prior to the strobing signal's rising edge (refer to the data sheet for more information on TSS_MIN)
3. Setup register is only accessible through the SPI Port A, so lines STROBE_AB, CLK_AB and DATA_AB are used.
4. STROBE_AB should be low at the start of the message.
5. CLK_AB should transition at least 16 times before STROBE_AB goes high.
6. Data at DATA_AB is latched during the CLK_AB's falling edge.
7. Data at DATA_AB must read low during the STROBE_AB's rising edge.
8. Write a 1 to the bits pertaining to the DC-DC channel to be disabled.
9. Write a 0 to the bits pertaining to the DC-DC channel to be enabled.

> **NOTE:** Failure to comply with item 7 results in data latched onto the EXTENDED Register, instead of the SETUP Register.

The next diagram depicts the SETUP register with the bits pertaining to DC-DC converter control.

4.2 SETUP Register – DC-DC Converter Segment

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		OSCD Fx				DCC SW	DCB SW	DCA SW							
		w-(0)	w-(0)			w-(0)	w-(0)	w-(0)							

OSCD Fx   Bits 13-12   DC-DC Converter Oscillator Frequency. Select oscillator frequency for the three converters

   00   100 KHz
   01   50 KHz
   10   200 KHz
   11   132.5 KHz

DCC SW   Bit 9   DC-DC Converter Channel C Switch Enable.
   0   Enable
   1   Disable

DCB SW   Bit 8   DC-DC Converter Channel B Switch Enable.
   0   Enable
   1   Disable

DCA SW   Bit 7   DC-DC Converter Channel A Switch Enable.
   0   Enable
   1   Disable

5 Selecting Inductors and Capacitors for the Switching Regulators
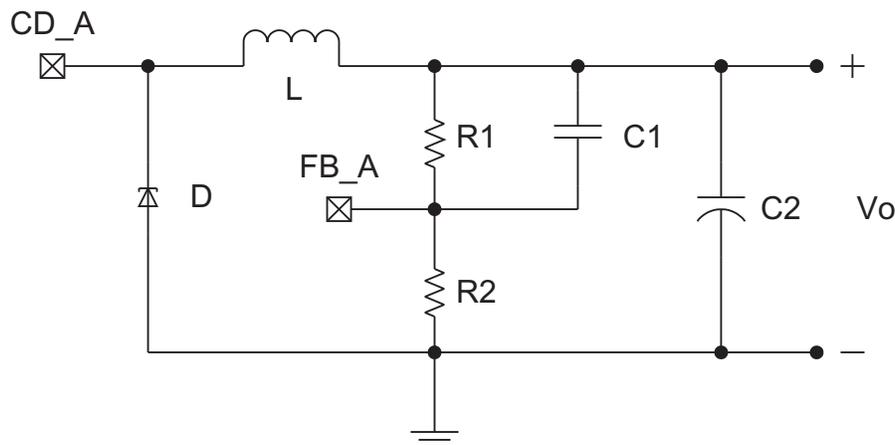
The three DC-DC converters in the DRV8809/8810 contain an integrated switch. The other components forming the Buck converter must be provided externally. Here is a guide detailing some equations which may result useful in the search for the appropiate external components.

5.1 Diode D

The external catch diode must have a forward voltage Vf lower than 1.2 V at peak current. We recommend Schotky diode MBRS360 or similar part.

5.2 Inductor L

Choosing the regulator inductance may be the most important part of the design. The inductor must be strong enough to withstand the current spikes that transpire through the winding as the regulator operates, without reaching the saturation current. An useful equation to get an idea on how much inductance the coil is to be valued at is shown below:

$$L = \frac{(Vin - Vout)}{Fs * Vin * Iripple} * Vout$$

(1)

Where:

L is the inductance the coil could be closely valued at,

Vin is the DC-DC Converter Input Voltage (VDIN),

Vout is the desired regulated output voltage (i.e. Vdd= 5V),

Fs is the Switching Frequency and

Iripple is the maximum current voltage desired at the inductor.

As a thumb rule, the designer may choose a 15% of the load current as the Iripple parameter. The table below shows the results of the equation when using 15% of the load current as Iripple and some other typical values.

Vin (V)	Vout (V)	Fswitching (KHz)	Io(Load) (A)	Iripple(approx) (A)	L(Calculated) (uH)
30	5	100	1.2	0.18	231.4814815

The calculated inductance is most likely a value that is not commonly available. For the purpose of this example, we chose 220 uH. If so, Iripple increases a tiny bit. The designer has to make sure such new ripple current does not induce saturation at the inductor or large current peaks that may go higher than the overcurrent protection limit. The table below shows the new findings.

Vin (V)	Vout (V)	Fswitching (KHz)	Io(Load) (A)	L(Calculated) (uH)	Iripple(Calculated) (A)
30	5	100	1.2	220	0.189393939

5.3 Filter Capacitor

Once the L is known, the filter capacitor C2 can be determined by the following equation:

$$C = \frac{Iripple}{8 * Fs * Vripple}$$

Where:

C is the capacitance of the Filter Capacitor,

Iripple is the ripple current for the chosen L inductance,

Fs is the same Switching Frequency and

Vripple is the ripple voltage at the capacitor.

The following table shows an example set of values for choosing the filter capacitor value.

Fswitching (KHz)	Io(Load) (A)	Iripple(Calculated) (A)	L(Calculated) (uH)	Vripple (mV)	C uF
100	1.2	0.18	220	1	236.74

The previous example determines the capacitance needed to obtain a 1-mV ripple voltage at the load. A good capacitor approximation would be to choose a 270 uf cap. Going higher decreases the ripple voltage further.

However, since capacitors are not ideal and there is a resistive component associated to them, it is vital to study how to select the estimated series resistance (ESR) before the filter capacitor is fully chosen.

The following equation relates ESR to the already-chosen inductance and capacitance values. A table with our acquired results follows.

$$ESR = 0.2 * \sqrt{\frac{L}{C}}$$

L (uH)	C (uF)	Resr (Ω)
220	270	0.18053419

Notice that we chose an inductor a little bit smaller and a capacitor larger than the ones the equation had provided. As a result, the ESR is now a smaller quantity. As long as the chosen filter capacitor is smaller than the result, the design should behave nicely.

6   Choosing RSx Sense Resistors and VREFxx Reference Voltages

The DRV8809/8810 H-bridge drivers supply current to the motor windings and control demand by means of an internal current regulator circuitry. This circuitry is very important as DC motors demand current as loading torque increases, being it at its worst during stalling periods, and stepper motors basically draw as much current as available until reaching its saturating current. Since these current levels are usually larger than what H-bridge drivers are designed to handle, current regulation ensures the motor does not use too much current.

The mechanism is quite simple. Current is measured as it passes through a sense resistor. This sense resistor is usually very small in resistivity to limit the amount of power dropped across it. However, it can not be too small as to have the voltage across it to hard to measure. The voltage across this resistor is directly proportional to the current flowing through it. This voltage is then compared against a reference Voltage. When the sense resistor current yields a voltage whose value is larger than the reference voltage, the comparator switches output polarity and disables the H bridge.
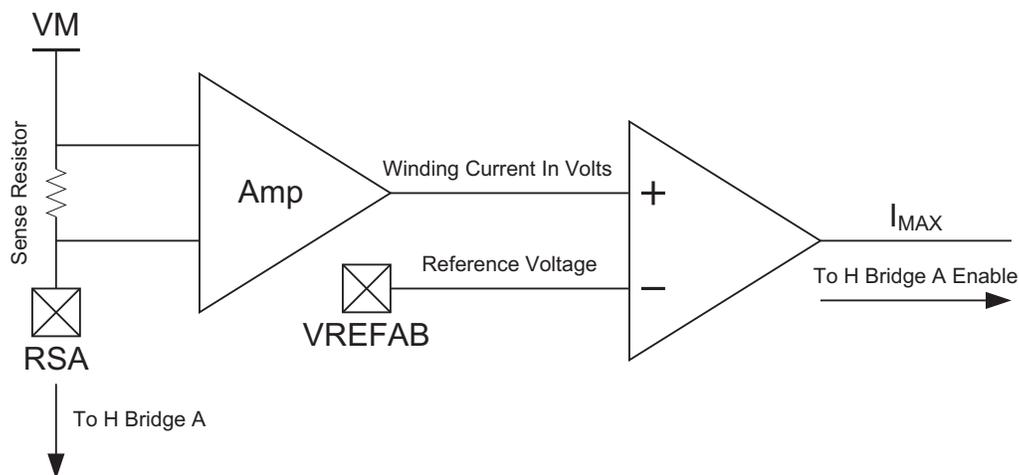
With the driver disabled, the winding current flows either through the internal NMOS body diodes or through some other path as specified by the different decay modes programmed at the Stepper Register.
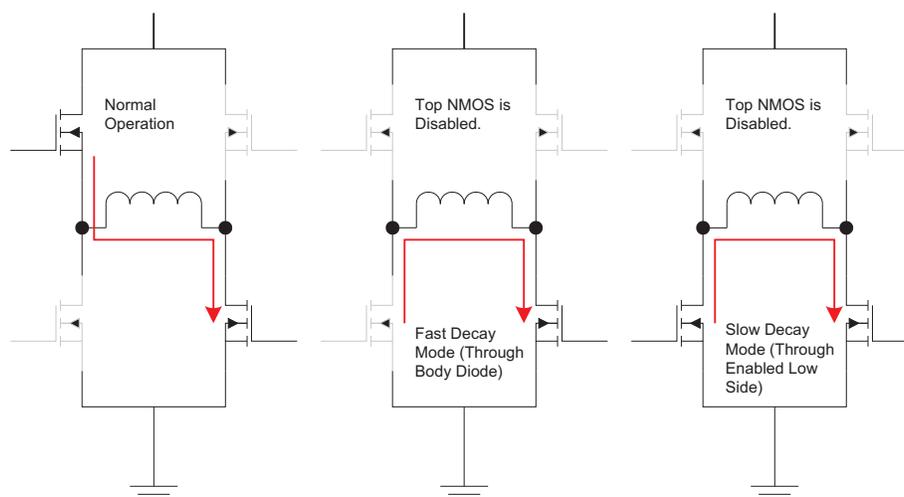
It is then vital for proper functioning, for the Reference Voltage (Vref) and Sense Resistor (Rsense) to be properly calculated. Both parameters play a role in setting the maximum current.

6.1 Imax for DC Motors:

When dealing with DC motors, the maximum current Imax is still a function of the Sense Resistor and VREFxx. However, the amplifier gain is either 1/10 or 1/20 and it can be selected by writing to the SETUP register when the nSLEEP signal is low. The equation then becomes:

$$I_{MAX} = \frac{VREFxx}{RSx} * Gain$$

Where,

VREFxx is the Reference Voltage at the VREFAB or VREFCD pin,

RSx is the Sense Resistor at the RSA, RSB, RSC or RSD pin and

Gain is a bit programmed at the SETUP register and defaults to 1/10

6.2 Imax for Stepper Motors:

When dealing with Stepper motors, the maximum current Imax is also a function of the Sense Resistor at RSx and VREFAB. However, the amplifier works on a totally different way than when dealing with DC motors. For a stepper, the amplifier gain can be one out of four values, and it can be programmed on a per step basis while writing to the respective STEPPER_xx register. The equation then becomes:

$$I_{MAX} = \frac{VREFxx}{RSx} * Torque$$

Where,

VREFxx is the Reference Voltage at the VREFAB or VREFCD pin,

RSx is the Sense Resistor at the RSA, RSB, RSC or RSD pin and

Torque consists of two bits (per phase) programmed at the respective STEPPER_xx register.

6.3   VREFxx and Sense Resistor considerations:

The way that Imax is set up may allow the user to think that by properly choosing a respective set of VREFxx and RSx parameters, any current per phase may be achieved. In a way this is true, but care must be taken to ensure that the maximum levels are not compromised. The following rules may help into choosing the proper RSx and VREFxx combination.

1. VREFxx can be any voltage from .8 V to 3.6 V. Lower than 0.8 V configures the motor to work at current chopping mode regardless of the load. Larger than 3.6 V damages the device as this is a pin voltage limit. A VREFxx equal to 2.0 V DC is recommended for best performance.

2. RSx can be any value that achieves the desired winding current, with a voltage drop accross it larger than 0.8 V. Large resistance values do decrease the eficiency as energy is wasted as heat. User must make certain the power handling capacity on the RSx resistor is large enough to withstand this disipated heat.

7 Monitoring with the Extended Setup Register

The EXTENDED Setup register offers a window to many troubleshooting options. Internal signals and events can be made available through the LOGIC_OUT pin. Different faults and reset scenarios can be disabled.

Writting to the EXTENDED Setup register can be accomplished by sending messages through the SPI_AB port when the nSLEEP line is low and DATA_AB remains high as the STROBE_AB line goes high. If DATA_AB is low as STROBE_AB goes high, data written is latched unto the SETUP register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEST MUXx			PRE TSD 1	PRE TSD 0	nORT DC_A	nORT DC_B	nORT DC_C	ISD3	ISD2	ISD1	ISD0	SIGNAL SELECTx			
w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)	w-(0)

TEST MUXx Bits 15-13 converters

- 000   Normal Operation
- 001   TSD Control - 1
- 010   TSD Control - 2
- 011   OSC Monitor Enable

PRE TSD 1 Bit 12 Pre Thermal Shutdown Output Select

- 0   Pre TSD on Logic Out (logic)
- 1   TSENS on TH Out (analog)

PRE TSD 0 Bit 11 Pre Thermal Shutdown Threshold Select

- 0   Pre TSD = TSD – 20deg.C
- 1   Pre TSD = TSD – 30deg.C

nORT DC_A Bit 10 Disable nORT Assertion for DC-DC Channel A

- 0   Normal Operation
- 1   Disable nORT Selection on DC-DC A fault. Still shutdown DC-DC A.

nORT DC_B Bit 9   Disable nORT Assertion for DC-DC Channel B

- 0   Normal Operation
- 1   Disable nORT Selection on DC-DC B fault. Still shutdown DC-DC B.

nORT DC_C Bit 8   Disable nORT Assertion for DC-DC Channel C

- 0   Normal Operation
- 1   Disable nORT Selection on DC-DC C fault. Still shutdown DC-DC C.

ISD3	Bit 7	Ignore Shutdown 3 – Thermal Shutdown
	0	Normal Operation
	1	Ignore Thermal Shutdown
ISD2	Bit 7	Ignore Shutdown 2 – Motor Over Current
	0	Normal Operation
	1	Ignore Motor Over Current Protection
ISD1	Bit 7	Ignore Shutdown 1 – DC/DC Voltage Supervisor
	0	Normal Operation
	1	Ignore DC/DC Voltage Supervisor
ISD0	Bit 7	Ignore Shutdown 0 – DC/DC Over Current
	0	Normal Operation
	1	Ignore DC/DC Over Current Protection
SIG SLCTx	Bits 3-0	LOGIC_OUT Signal Selector
	0000	DC/DC OCP_A/OVP_A/UVP_A
	0001	DC/DC OCP_B/OVP_B/UVP_B
	0010	DC/DC OCP_C/OVP_C/UVP_C
	0011	DC/DC OCP_A/OCP_B/OCP_C
	0100	DC/DC OVP_A/OVP_B/OVP_C
	0101	DC/DC UVP_A/UVP_B/UVP_C
	0110	Motor OCP
	0111	TSD
	1000	Revision[0] (1 for TPIC70600)
	1001	Revision[1] (0 for TPIC70600)
	1010	Revision[2] (1 for TPIC70600)
	1011	Vendor[0] (0 for TI)
	1100	Vendor[1] (0 for TI)
	1101	Internal Oscillator Clock
	1110	Fixed Value as 1
	1111	Fixed Value as 1

7.1 Signal Select Bits

First, LOGIC_OUT reflects the state of nine different variables internal to the device, which have been encoded into the first six addresses, and multiplexed together, by means of an OR function. Writing six times to the EXTENDED register, and with consecutive addresses (0x00 to 0x05) on the Signal Select bits, outputs the state of the variables which can then be sampled at the LOGIC_OUT pin.

Per example, when sending a 0bxxxxxxxxxxxx0000 to the EXTENDED Register, the three status bits DC/DC OCP_A, DC/DC OVP_A, and DC/DC UVP_A are OR'd together, and the function's output relayed into LOGIC_OUT. If a one is sampled at the pin, "addresses" 0x03, 0x04, and 0x05 has to be read to determine which one of the three signals raised the flag. The next table shows the possible, and meaningful, combinations.

Signal at LOGIC_OUT	0000	0001	0010	0011	0100	0101
DC/DC OCP_A	1	0	0	1	0	0
DC/DC OCP_B	0	1	0	1	0	0
DC/DC OCP_C	0	0	1	1	0	0
DC/DC OVP_A	1	0	0	0	1	0
DC/DC OVP_B	0	1	0	0	1	0
DC/DC OVP_C	0	0	1	0	1	0
DC/DC UVP_A	1	0	0	0	0	1
DC/DC UVP_B	0	1	0	0	0	1
DC/DC UVP_C	0	0	1	0	0	1

The following picture shows a suggested algorithm to acquire the fault information by sampling the LOGIC_OUT pin as the different writes occur to the EXTENDED register.

Start → Signal Select 0000 → Logic_out = 1? — Yes → 0 ; No → Signal Select 0001 → Logic_out = 1? — Yes → 1 ; No → Signal Select 0002 → Logic_out = 1? — Yes → 2 ; No → END

0 → Signal Select 0003 → Logic_out = 1? — Yes → DC/DC OCP_A Fault ; No → Signal Select 0004 → Logic_out = 1? — Yes → DC/DC OVP_A Fault ; No → Signal Select 0005 → Logic_out = 1? — Yes → DC/DC UVP_A Fault

1 → Signal Select 0003 → Logic_out = 1? — Yes → DC/DC OCP_B Fault ; No → Signal Select 0004 → Logic_out = 1? — Yes → DC/DC OVP_B Fault ; No → Signal Select 0005 → Logic_out = 1? — Yes → DC/DC UVP_B Fault

2 → Signal Select 0003 → Logic_out = 1? — Yes → DC/DC OCP_C Fault ; No → Signal Select 0004 → Logic_out = 1? — Yes → DC/DC OVP_C Fault ; No → Signal Select 0005 → Logic_out = 1? — Yes → DC/DC UVP_C Fault ; No → END

The other 10 addresses (Signal Select 0x0006 to 0x000F) offer different information as seen on the Register description. There is no need to decode data when sampling the LOGIC_OUT pin for any of

these other 10 combinations.

7.2 *ISDx Bits*

The Ignore Shutdown bits offer a powerful troubleshooting tool by disabling protection on an individual fault generator basis. The four faults that could generate a device shutdown are DC/Dc converter Over Current, DC/DC Voltage Supervisor, Motor Overcurrent, and Thermal Shutdown. If none of these bits are written with a 1, shutdown occurs when the respective fault is registered at the protection circuitry. Otherwise, any bit written with a zero disables its respective protection. It is recommended for the device not to be subjected to each particular scenario for too long as damage may occur.

7.3 *Disable nORT bits*

There are three bits utilized to disable DC/DC converter nORT assertion when fault occurs, on a per converter channel basis. When these bits are written with 0's, the respective DC/DC converter asserts the nORT signal, disabling the entire device. If on the other hand, a 1 is written to any one of these bits, the fault on that DC/DC converter still disables the converter, but does not assert the nORT signal. Possible fault sources on the DC/DC converters are overcurrent (more than 1.5 amps per channel) and Voltage Supervisor.

7.4 *Pre TSD bits*

Pre TSD stands for Pre Thermal Shutdown. Pre Thermal threshold is equivalent to a warm level, whereas a hot level would be the Shutdown point. Although the Thermal Shutdown point is fixed at 155°C, the Pre Thermal Shutdown can be configured.

Pre TSD defaults at Thermal Shutdown - 20°C. However, by writting a 1 to the Pre TSD 0 bit, the Pre TSD temperature can belowered to Thermal Shutdown - 30°C.

The Pre TSD 1 Bit selects between the warm temperature state being output at the TH_OUT pin as a logic level, or as an analog value. When Pre TSD 1 is low, a low level at TH_OUT means the temperature is less then Pre TSD, and a high level means the internal temperature has gone over the Pre TSD level. When the Pre TSD 1 bit is high, the TH_OUT pin outputs an analog value directly proportional with internal temperature.