



# **OZMTR3000 Motor Controller**

User's Manual

UM-0045

## **About Oztek**

Oztek Corp. is proven innovator of power, control, and instrumentation solutions for the most demanding industrial applications. Oztek products include variable motor drives, grid tie inverters, frequency converters, stand alone inverters, DC/DC converters, and DSP based control boards for power control applications.

## **Trademarks**

OZDSP3000 is a trademark of Oztek Corp. Other trademarks, registered trademarks, and product names are the property of their respective owners and are used herein for identification purposes only.

## **Notice of Copyright**

Oztek OZMTR3000 Motor Controller User's Manual © March 2012 Oztek Corp. All rights reserved.

## **Exclusion for Documentation**

UNLESS SPECIFICALLY AGREED TO IN WRITING, Oztek Corp. ("Oztek")

- (A) MAKES NO WARRANTY AS TO THE ACCURACY, SUFFICIENCY OR SUITABILITY OF ANY TECHNICAL OR OTHER INFORMATION PROVIDED IN ITS MANUALS OR OTHER DOCUMENTATION.
  
- (B) ASSUMES NO RESPONSIBILITY OR LIABILITY FOR LOSSES, DAMAGES, COSTS OR EXPENSES, WHETHER SPECIAL, DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL, WHICH MIGHT ARISE OUT OF THE USE OF SUCH INFORMATION. THE USE OF ANY SUCH INFORMATION WILL BE ENTIRELY AT THE USER'S RISK.
  
- (C) IF THIS MANUAL IS IN ANY LANGUAGE OTHER THAN ENGLISH, ALTHOUGH STEPS HAVE BEEN TAKEN TO MAINTAIN THE ACCURACY OF THE TRANSLATION, THE ACCURACY CANNOT BE GUARANTEED. APPROVED OZTEK CONTENT IS CONTAINED WITH THE ENGLISH LANGUAGE VERSION WHICH IS POSTED AT [WWW.OZTEKCORP.COM](http://WWW.OZTEKCORP.COM).

## **Date and Revision**

April 2013 Rev B

## **Part Number**

UM-0045

## **Contact Information**

USA

Telephone: 603-546-0090

Fax: 603-386-6366

Email [techsupport@oztekc corp.com](mailto:techsupport@oztekc corp.com)

# Table of Contents

<b>1. Introduction .....</b>	<b>1</b>
1.1 Referenced Documents .....	1
1.2 Definitions .....	1
<b>2. Functional Description .....</b>	<b>2</b>
2.1 Typical Hardware Implementation .....	2
2.1.1 Power Module .....	2
2.1.2 Brake Controller .....	3
2.1.3 Control Board .....	3
2.2 Description of Operation .....	3
2.2.1 Volts/Hertz Control .....	3
2.2.2 Field Oriented Control .....	5
2.2.3 Dual Interleaved Inverter Operation .....	6
2.2.4 Temperature Based Torque Derating .....	8
2.2.5 Maximum Torque Profile .....	9
2.2.6 Current Fold Back .....	10
2.3 Brake Controller .....	12
2.4 State Sequencing .....	12
2.4.1 Initialize .....	13
2.4.2 Calibrate .....	13
2.4.3 Idle .....	14
2.4.4 Magnetizing .....	14
2.4.5 Align .....	14
2.4.6 Running .....	14
2.4.7 Fault .....	14
2.5 Fault and Warning Conditions .....	15
2.5.1 Warnings .....	15
2.5.1.1 High Inverter Temperature .....	15
2.5.1.2 High Motor Current .....	15
2.5.1.3 High Motor Temperature .....	15
2.5.1.4 High Motor Speed .....	16
2.5.1.5 Clamped Torque Command .....	16
2.5.1.6 Braking Torque Disabled .....	16
2.5.1.7 Motor Torque Reduced Due to Thermal Derating .....	16
2.5.1.8 $I_q/I_d$ Current Foldback .....	16
2.5.1.9 High DC Link Voltage .....	17
2.5.1.10 High Brake Switch Temperature .....	17
2.5.1.11 Local Bias Supply Tolerance Warnings .....	17
2.5.2 Faults .....	17
2.5.2.1 Inverter IGBT Errors .....	17
2.5.2.2 Inverter Hardware Over Temperature .....	18
2.5.2.3 Software Inverter Over Temperature .....	18
2.5.2.4 Software Inverter Over Current .....	18
2.5.2.5 Motor Over Temperature .....	18
2.5.2.6 Motor Over Speed .....	18
2.5.2.7 Motor Align Error .....	18
2.5.2.8 Brake Switch Hardware Error .....	18
2.5.2.9 Brake Switch Hardware Over Temperature .....	19

2.5.2.10	Brake Switch Software Over Temperature.....	19
2.5.2.11	DC Link Over-Voltage.....	19
2.5.2.12	Configuration Memory Error .....	19
2.5.2.13	Calibration Error .....	19
2.5.2.14	Communications Timeout .....	20
<b>3.</b>	<b>OZDSP3000 Hardware Interfacing .....</b>	<b>20</b>
3.1	Application Interfaces.....	20
3.1.1	J11: SKiiP Power Module Interface – Inverter A .....	21
3.1.2	Custom Driver Interface Considerations.....	22
3.1.2.1	Power.....	22
3.1.2.2	Switch Commands .....	22
3.1.2.3	Error Inputs.....	22
3.1.2.4	DC Link Voltage Sensing .....	22
3.1.2.5	DC Link Hardware Over-Voltage Protection.....	22
3.1.2.6	Hardware Over-Current and Desaturation Protection.....	22
3.1.2.7	Current Sense Signals .....	23
3.1.2.8	Temperature Sense Signals .....	23
3.1.3	J13: SKiiP Power Module Interface – Inverter B (optional).....	23
3.1.4	J8: Auxiliary Contactor/Relay Drive Interface (optional) .....	24
3.1.5	J25: Bias Power Input & SPI Boot Enable .....	24
3.1.6	J14: Brake Chopper Power Module Interface (optional) .....	24
3.1.7	J12: Quadrature Encoder Interface (optional).....	26
3.1.8	J16: Expansion Board Interface – Resolver Daughter Card (optional).....	26
3.1.9	J4: Motor Temperature Input (optional).....	27
3.1.10	P2: Isolated CAN Bus Interface.....	27
3.1.10.1	J18/19 CAN Termination Jumpers.....	27
3.2	Electrical Interfaces .....	29
3.3	Mechanical Interface .....	30
<b>4.</b>	<b>Parameter Register Interface .....</b>	<b>31</b>
4.1	Register Properties .....	31
4.1.1	Parameter ID .....	31
4.1.2	Data Types.....	31
4.1.2.1	Specifying Fixed-Point Parameters.....	31
4.1.3	Access Level.....	31
4.2	Volatile Registers.....	32
4.2.1	Command Registers .....	32
4.2.1.1	On/Off Control.....	32
4.2.1.2	Mode Control .....	32
4.2.1.3	Torque Setpoint.....	32
4.2.1.4	Speed Setpoint .....	33
4.2.1.5	Fault Reset .....	33
4.2.1.6	Configuration Password .....	33
4.2.1.7	Configuration Reset.....	33
4.2.1.8	Configuration Reload.....	34
4.2.1.9	Contact Control.....	34
4.2.2	Instrumentation Registers.....	34
4.2.2.1	Operating State .....	35
4.2.2.2	Operating Mode .....	35
4.2.2.3	Motor Temperature .....	35

4.2.2.4	Motor Speed .....	35
4.2.2.5	Motor Power .....	35
4.2.2.6	Motor Current .....	36
4.2.2.7	Motor Current – Inverter A, B .....	36
4.2.2.8	Brake State .....	36
4.2.2.9	Brake Temperature.....	36
4.2.2.10	Brake Temperature.....	36
4.2.2.11	Inverter A Temperature.....	36
4.2.2.12	Inverter B Temperature.....	36
4.2.2.13	DC Link Voltage.....	36
4.2.2.14	Warning Status .....	37
4.2.2.15	Fault Status .....	37
4.2.2.16	Register Operation Status .....	38
4.2.2.17	DSP Software Revision – Major/Minor .....	38
4.2.2.18	FPGA Firmware Revision – Major/Minor .....	38
4.2.2.19	PCB Variant.....	38
4.2.2.20	Contactors State.....	38
4.3	Non-Volatile Configuration Registers .....	39
4.3.1	Configuration Control Parameters .....	40
4.3.1.1	EEPROM Header .....	40
4.3.1.2	Factory Configuration Revision – Major .....	40
4.3.1.3	Factory Configuration Revision – Minor .....	40
4.3.1.4	Application Configuration Data Revision.....	41
4.3.1.5	Hardware Configuration .....	41
4.3.1.6	User Configuration Revision .....	41
4.3.1.7	Configuration Password .....	41
4.3.2	CAN Interface Parameters.....	41
4.3.2.1	CAN Group ID.....	42
4.3.2.2	CAN Module ID .....	42
4.3.2.3	CAN Baud Rate.....	42
4.3.2.4	CAN Status Destination Group ID .....	42
4.3.2.5	CAN Status Destination Module ID.....	42
4.3.2.6	CAN Timeout.....	43
4.3.2.7	CAN Automatic Alarm Transmit Enable.....	43
4.3.2.8	CAN Update Rate – Motor Status Message.....	43
4.3.2.9	CAN Update Rate – Brake Status Message.....	43
4.3.2.10	CAN Update Rate – System Status Message .....	43
4.3.2.11	CAN Update Rate - Alarm Status Message .....	43
4.3.3	Instrumentation and Measurement Scaling Parameters.....	44
4.3.3.1	Instrumentation Interrupt Rate.....	44
4.3.3.2	DC Link Voltage Measurement – Full Scale .....	44
4.3.3.3	Motor Current Measurement – Full Scale.....	44
4.3.3.4	Motor Current Measurement Polarity Select.....	44
4.3.3.5	Low Pass Filter Cutoff – Motor Current.....	44
4.3.3.6	Low Pass Filter Cutoff – DC Link .....	44
4.3.3.7	Low Pass Filter Cutoff – Temperatures.....	44
4.3.4	Fault and Warning Parameters .....	45
4.3.4.1	DC Link Over Voltage Fault Threshold .....	45

4.3.4.2	DC Link Over Voltage Warning Threshold .....	45
4.3.4.3	DC Link Over Voltage Recover Threshold .....	45
4.3.4.4	Inverter Temperature Fault Threshold .....	45
4.3.4.5	Inverter Temperature Warning Threshold .....	46
4.3.4.6	Inverter Temperature Recover Threshold .....	46
4.3.4.7	Brake Temperature Fault Threshold .....	46
4.3.4.8	Brake Temperature Warning Threshold .....	46
4.3.4.9	Brake Temperature Recover Threshold .....	46
4.3.4.10	Motor Temperature Fault Threshold .....	46
4.3.4.11	Motor Temperature Warning Threshold .....	46
4.3.4.12	Motor Temperature Recover Threshold .....	46
4.3.4.13	Motor Over Current Fault Threshold .....	46
4.3.4.14	Motor Over Current Warning Threshold .....	47
4.3.4.15	Motor Over Current Recover Threshold .....	47
4.3.4.16	Motor Over Speed Fault Threshold .....	47
4.3.4.17	Motor Over Speed Warning Threshold .....	47
4.3.4.18	Motor Over Speed Recover Threshold .....	47
4.3.4.19	Inverter Error Pin Active High .....	47
4.3.4.20	Inverter Over Temp Pin Active High .....	47
4.3.4.21	Brake Hardware Error Pin Active High .....	48
4.3.4.22	Brake Hardware Over Temp Pin Active High .....	48
4.3.5	Inverter Control Parameters .....	48
4.3.5.1	Pulse Width Modulation Frequency .....	48
4.3.5.2	Pulse Width Modulation Deadband Time .....	48
4.3.5.3	Pulse Width Modulation Max/Min Duty Cycle .....	48
4.3.5.4	Inverter Power Stage Dead Time .....	49
4.3.6	Drive Configuration Parameters .....	49
4.3.6.1	Motor Drive Type .....	49
4.3.6.2	Motor Speed Feedback Device Select .....	49
4.3.6.3	Torque Command Slew Rate .....	50
4.3.6.4	Torque Command Maximum .....	50
4.3.6.5	Torque Command Default at Turn On .....	50
4.3.6.6	Speed Command Slew Rate .....	50
4.3.6.7	Speed Command Maximum .....	50
4.3.6.8	Speed Command Default at Turn On .....	50
4.3.6.9	Torque/Speed Mode Default at Power On .....	50
4.3.7	General Motor Parameters .....	51
4.3.7.1	Motor Pole Pair Count .....	51
4.3.7.2	Motor Rated Voltage .....	51
4.3.7.3	Motor Rated Frequency .....	51
4.3.7.4	Motor Maximum Current .....	51
4.3.7.5	Motor Maximum Frequency .....	51
4.3.8	AC Induction Motor Parameters .....	51
4.3.8.1	AC Motor Mutual Inductance .....	52
4.3.8.2	AC Motor Stator Leakage Inductance .....	52
4.3.8.3	AC Motor Rotor Leakage Inductance .....	52
4.3.8.4	AC Motor Rotor Resistance .....	52
4.3.8.5	AC Motor Magnetizing Current .....	52

4.3.8.6	AC Motor Initial Magnetizing Current .....	52
4.3.8.7	AC Motor Minimum Magnetizing Current .....	52
4.3.8.8	AC Motor Magnetizing Current Slew Rate .....	53
4.3.8.9	AC Motor Field Weakening Enable.....	53
4.3.8.10	AC Motor Field Weakening Minimum Speed .....	53
4.3.8.11	AC Motor Temperature Coefficient.....	53
4.3.9	Permanent Magnet Motor Parameters .....	53
4.3.9.1	PM Torque Constant.....	54
4.3.9.2	PM Align At Startup Enable .....	54
4.3.9.3	PM Align Current .....	54
4.3.9.4	PM Align Time.....	54
4.3.9.5	PM Align Maximum Speed Threshold .....	55
4.3.9.6	PM Align Angle Offset.....	55
4.3.10	Volts/Hertz Drive Parameters .....	55
4.3.10.1	V/Hz Minimum Frequency.....	55
4.3.10.2	V/Hz Maximum Frequency .....	56
4.3.10.3	V/Hz Inflection Frequency .....	56
4.3.10.4	V/Hz Zero Hertz Voltage .....	56
4.3.10.5	V/Hz Synchronous Startup Enable.....	56
4.3.10.6	V/Hz Startup Frequency .....	56
4.3.10.7	V/Hz Startup Pause Time.....	56
4.3.11	Maximum Torque Profile Parameters.....	56
4.3.11.1	Torque Profile Enable .....	57
4.3.11.2	Torque Profile Maximum Torque .....	57
4.3.11.3	Torque Profile Maximum Power .....	57
4.3.11.4	Torque Profile Power*Speed Region Start.....	57
4.3.12	Temperature Measurement and Derating Parameters .....	57
4.3.12.1	Inverter Temp Enable .....	58
4.3.12.2	Inverter Temp Coefficients (C0 through C3).....	58
4.3.12.3	Inverter Temp Derating Enable .....	59
4.3.12.4	Inverter Temp Derating Low Threshold .....	59
4.3.12.5	Inverter Temp Derating High Threshold.....	59
4.3.12.6	Inverter Temp Min Derating Factor.....	59
4.3.12.7	Motor Temp Enable.....	59
4.3.12.8	Motor Temp Coefficients (C0 through C3) .....	59
4.3.12.9	Motor Temp Derating Enable .....	60
4.3.12.10	Motor Temp Derating Low Threshold.....	60
4.3.12.11	Motor Temp Derating High Threshold .....	60
4.3.12.12	Motor Temp Min Derating Factor .....	60
4.3.12.13	Brake Temp Enable .....	60
4.3.12.14	Brake Temp Coefficients (C0 through C3).....	60
4.3.13	Encoder Parameters.....	61
4.3.13.1	Encoder Line Count .....	61
4.3.13.2	Encoder Positive Direction .....	61
4.3.13.3	Encoder Clock Timer Prescaler .....	61
4.3.13.4	Encoder Event Counter Prescaler.....	61
4.3.13.5	Encoder Debounce Prescaler.....	62
4.3.13.6	Encoder Debounce Select.....	62

4.3.14	Motor Speed Calculation Parameters .....	62
4.3.14.1	Motor Speed Low-to-High Crossover Mode.....	63
4.3.14.2	Motor Speed Low-to-High RPM .....	63
4.3.14.3	Motor Speed Low-to-High Hysteresis.....	63
4.3.14.4	Motor Speed Low Pass Filter Cutoff Freq.....	63
4.3.14.5	Motor Speed Deadband Period (Low Speed) .....	63
4.3.14.6	Motor Speed Update Prescale (High Speed).....	64
4.3.15	Current Regulator Parameters .....	64
4.3.15.1	Current Controller Gain Constants (Kp, Ki).....	64
4.3.15.2	Id/Iq Current Foldback Enable .....	65
4.3.15.3	Id/Iq Current Foldback Modulation Index Threshold.....	65
4.3.15.4	Id/Iq Current Foldback Controller Gain Constants (Kp, Ki).....	66
4.3.16	Speed Regulator Parameters .....	66
4.3.16.1	Speed Control Enable .....	66
4.3.16.2	Speed Controller Gain Constants (Kp, Ki) .....	67
4.3.16.3	Speed Controller Maximum/Minimum Current.....	67
4.3.17	Dual Interleaved Inverter Parameters .....	68
4.3.17.1	Dual Interleaved Inverter Enable.....	68
4.3.17.2	Zero Sequence Current Control Enable.....	68
4.3.17.3	Zero Sequence Current Controller Gain Constants (Kp, Ki).....	68
4.3.18	Brake Controller Parameters.....	69
4.3.18.1	Brake Controller Enable.....	69
4.3.18.2	Brake Controller Voltage Threshold .....	69
4.3.18.3	Brake Controller Recover Voltage Threshold .....	69
4.3.18.4	Brake Controller Fault Motor Response.....	69
4.3.18.5	Brake Gate Driver Signal Active High.....	70
<b>5.</b>	<b>Oztek PCC Tool.....</b>	<b>70</b>
<b>6.</b>	<b>Maintenance and Upgrade.....</b>	<b>70</b>
	<b>Warranty and Product Information.....</b>	<b>71</b>
	<b>Return Material Authorization Policy.....</b>	<b>73</b>



## Table of Figures

Figure 1 – Typical Drive Electrical System Schematic .....	2
Figure 2 – Volts/Hertz Profile.....	4
Figure 3 – FOC Controller Functional Block Diagram.....	5
Figure 4 – Typical Interleaved Drive Electrical System Schematic.....	7
Figure 5 – Interleaved Controller Functional Block Diagram.....	8
Figure 6 – Torque and Power in the Three Speed Regions.....	10
Figure 7 – Controller State Machine.....	13
Figure 8 – OZDSP3000 Motor Drive Application Electrical Connections .....	20
Figure 9 – Multi-Node CAN Network Configuration .....	28
Figure 10 – CAN Interface Circuit.....	28
Figure 11 – Approximate Connector, Jumper, LED, and Test Hook Locations.....	29
Figure 12 – OZDSP3000 Mechanical Dimensions .....	30
Figure 13 – Current Control PI Regulators .....	65
Figure 14 – Current Fold Back PI Regulator .....	66
Figure 15 – Speed Control PI Regulator .....	67
Figure 16 – Zero Sequence PI Regulator .....	68

## Table of Tables

Table 1 – J11 SKiiP Power Module Pin Assignment .....	21
Table 2 – J11 Current Sense Pin Assignment .....	23
Table 3 – J8 Relay Drive Pin Assignment.....	24
Table 4 – J12 RS-485 Pin Assignment.....	24
Table 5 – J14 Half Bridge Power Module Pin Assignment .....	25
Table 6 – J12 Encoder Interface Pin Assignment .....	26
Table 7 – J4 Pin Assignment.....	27
Table 8 – P2 CAN Bus Pin Assignment .....	27
Table 9 – Command Register Set .....	32
Table 10 – Instrumentation Register Set .....	34
Table 11 – Configuration Control Parameter Summary .....	40
Table 12 – CAN Interface Parameter Summary .....	42
Table 13 – Instrumentation and Measurement Scaling Parameter Summary .....	44
Table 14 – Fault and Warning Parameter Summary.....	45
Table 15 – Inverter Control Parameter Summary.....	48
Table 16 – Drive Configuration Parameter Summary.....	49
Table 17 – General Motor Parameter Summary.....	51
Table 18 – AC Induction Motor Parameter Summary.....	52
Table 19 – Permanent Magnet Motor Parameter Summary.....	54
Table 20 – Volts/Hertz Parameter Summary .....	55
Table 21 – Maximum Torque Profile Parameter Summary .....	57
Table 22 – Temperature Measurement and Derating Parameter Summary.....	57
Table 23 – Encoder Parameter Summary .....	61
Table 24 – Motor Speed Calculation Parameter Summary .....	62
Table 25 – Current Regulator Parameter Summary .....	64
Table 26 – Speed Regulator Parameter Summary.....	66
Table 27 – Dual Interleaved Inverter Parameter Summary.....	68
Table 28 – Brake Controller Parameter Summary .....	69

## 1. Introduction

This document is intended to provide instruction on how to employ the Oztek OZMTR3000 firmware application on a standard Oztek OZDSP3000 controller in an actual hardware system. It describes the electrical connections as well as the scaling of the various signals required by the control firmware.

### 1.1 Referenced Documents

Ref.	Document	Description
[1]	UM-0018	OZDSP3000 Users Manual
[2]	FS-0046	OzCan Protocol Function Specification
[3]	FS-0066	OzCan Motor Controller Device Profile
[4]	UM-0015	Oztek TMS28x CAN Bootloader Users Manual
[5]	UM-0036	Power Control Center GUI User's Manual

### 1.2 Definitions

ACI	A/C Induction Motor
CAN	Controller Area Network
DSP	Digital Signal Processor
EEPROM	Electrically Erasable Programmable Read Only Memory
EMC	Electro-Magnetic Compatibility
EMI	Electro-Magnetic Interference
FOC	Field Oriented Control
GND	Ground, low side of input power supply
GUI	Graphical User Interface
HMI	Human Machine Interface
IPM	Intelligent Power Module
NC	Not Connected
PCB	Printed Circuit Board
PCC	Power Control Center
PI	Proportional and Integral Compensator
PLC	Programmable Logic Controller
POR	Power On Reset
PMSM	Permanent Magnet Synchronous Machine
PWM	Pulse Width Modulation
SVM	Space Vector Modulator

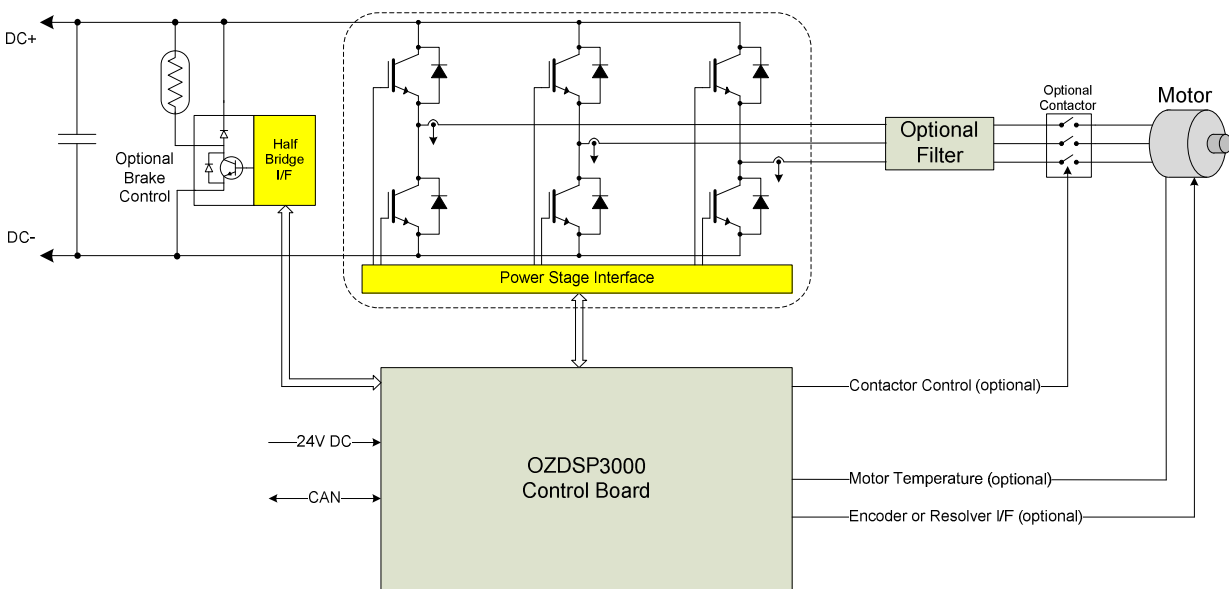
## 2. Functional Description

The OZMTR3000 controller is used to provide speed and/or torque control for AC induction or permanent magnet motors by converting DC power to AC power as required by the motor application. The DC/AC inverter controls are bidirectional, being able to not only provide power to the motor, but also able to absorb power during regeneration. Typical OZMTR3000 applications include

- Renewable energy systems such as wind turbines
- Flywheel energy storage systems
- Traction motor drives for vehicle applications
- Servo application for industrial controls

### 2.1 Typical Hardware Implementation

Figure 1 illustrates a typical OZDSP3000-based motor drive system. The diagram shows several optional interfaces such as a brake chopper, a contactor to disconnect the motor from the drive, and various motor sensors for temperature and speed feedback. In the simplest form, the system could consist of an inverter-rated motor that is directly connected to the integrated power stage module with a DC input and bypass capacitors.



**Figure 1 – Typical Drive Electrical System Schematic**

#### 2.1.1 Power Module

The power module is a three phase semiconductor bridge which controls the power transfer from the DC link to the motor and vice versa. Oztek control boards are designed to interface directly with Semikron SKiiP power modules. In addition to the power devices, these modules

provide current sensing, DC link voltage sensing, temperature sensing, and protection features including over voltage, over current, and desaturation protection.

### 2.1.2 Brake Controller

In systems that employ single-quadrant DC link supplies that can only source power to the motor, a brake controller can be used to prevent regenerative braking energy from the motor from increasing the DC link voltage to dangerously high levels. The OZMTR3000 controller provides features that support a hysteretically controlled brake switch. The switch can be used to connect a brake resistor across the DC link when the voltage rises above a configurable threshold. The resistor would then be disconnected when the DC link voltage falls below a configurable lower “recover” threshold.

### 2.1.3 Control Board

The control board is used to generate the gating signals to the power switches in order to control the motor current and/or speed. The control board also provides instrumentation and user control functions.

## 2.2 Description of Operation

The motor controller can be programmed to operating in the following drive configurations:

- Open-loop Volts/Hertz control for AC induction motors
- Closed-loop field oriented current control, supporting the following motor types:
  - AC induction motors
  - Permanent magnet motors

These drive configurations are discussed in more detail in the following sections.

### 2.2.1 Volts/Hertz Control

The OZMTR3000 controller provides a Volts/Hertz control mode for those AC induction motor applications that do not require the advanced control and performance of the closed-loop field oriented current controller. In addition to using the general motor parameters specified in PIDs 0x8070 – 0x8074, this controller is enabled and configured using V/Hz-specific parameters in PIDs 0x80A0 – 0x800A6. These parameters are used to determine the desired relationship between motor speed (i.e. electrical frequency, in Hertz) and the applied line-to-line RMS voltage.

In the simplest form, the Volts/Hertz curve can be a straight line starting at the origin (0 Hz, 0V) and ending at the specified rated voltage (PID 0x8071) and frequency (PID 0x8072) for the motor. When the user commands a particular motor speed, the controller will determine the corresponding electrical frequency (Hertz) based on the specified motor pole pairs (PID 0x8070) and will apply a voltage to the motor at the specified frequency based on the rated

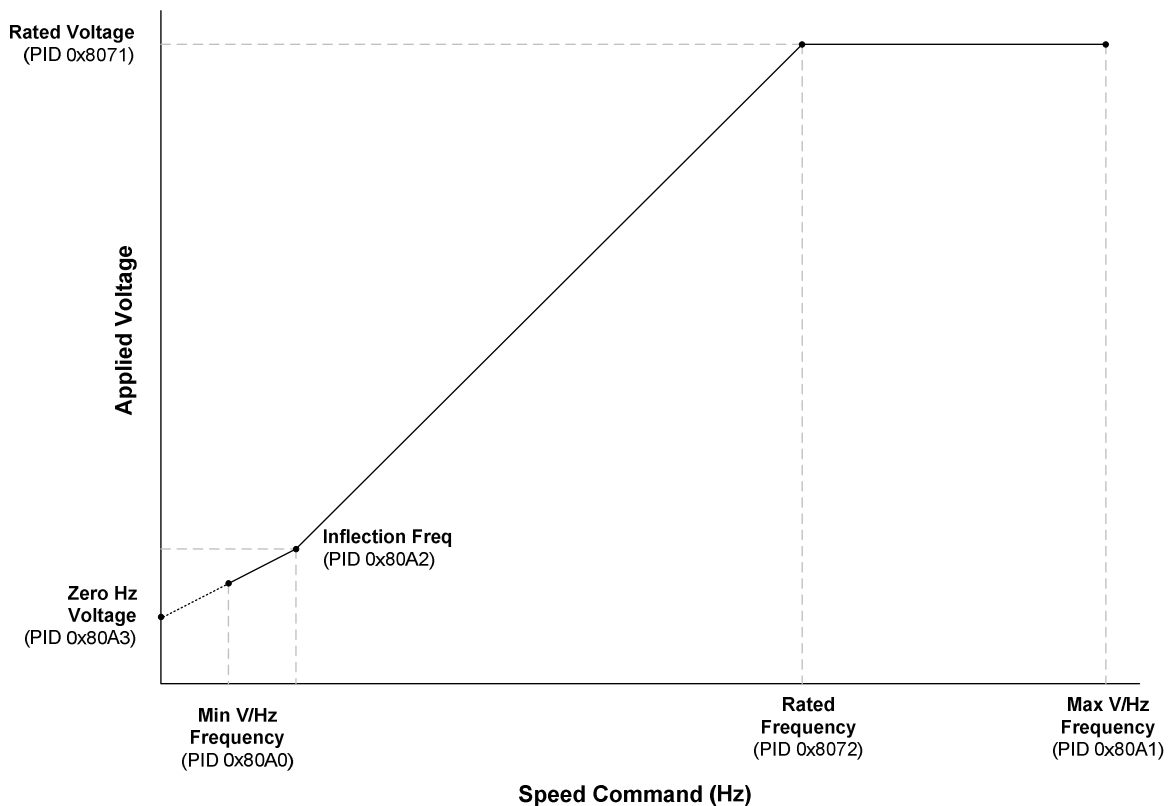
voltage/frequency ratio. Note that the controller does not immediately apply this new frequency, but rather slowly slews the applied frequency from the present operating value to the newly commanded value based on the programmed **Speed Command Slew Rate** (PID 0x8065).

When the user's speed command results in a frequency above the rated frequency, the drive will allow for a higher frequency while keeping the applied voltage to the specified rated voltage, up until a maximum defined frequency (PID 0x80A1).

For low speed operation, the drive provides a "voltage boost" feature to avoid stalling the motor. This feature allows the user to modify the low speed portion of the Volts/Hertz curve based on two additional parameters. First, the voltage at zero hertz can be specified (PID 0x80A3) to be a value greater than zero volts. Second, an "inflection point" is specified (PID 0x80A2) that defines the frequency from which the "voltage boost" curve deviates from the rated Volts/Hertz curve.

For applications that require a minimum speed/frequency, this can be programmed into the controller using PID 0x80A0. Speed commands below this value will be overridden with the minimum value specified by this parameter.

The resulting programmable Volts/Hertz profile is depicted in the figure below:



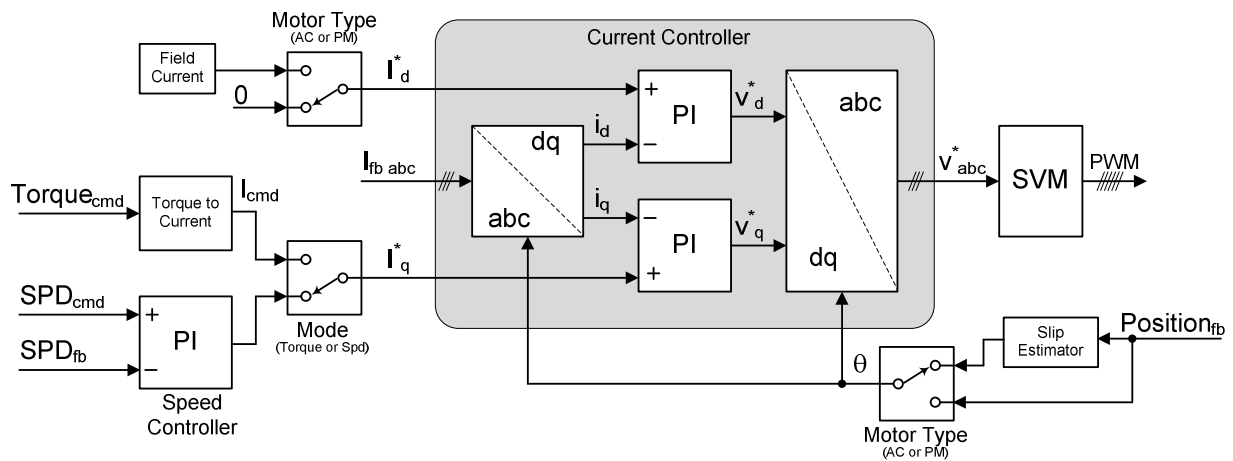
**Figure 2 – Volts/Hertz Profile**

In addition to the configurable profile discussed above, the user can configure the starting frequency (PID 0x80A5) that the drive will use when first turning on and a “pause time” (PID 0x80A6) in which to hold the drive at this startup frequency before slewing to the commanded speed.

Because the Volts/Hertz controller is inherently an open loop controller, it does not require any feedback from the motor (such as current, speed, temperature, etc) to operate. Not including these additional sensors in the application will reduce system cost and complexity, but at the expense of losing the ability to monitor those values and provide potential fault protection. For this reason, it is recommended that current sensors be used at a minimum for over current protection.

## 2.2.2 Field Oriented Control

Figure 3 presents a simplified block diagram of the field oriented control scheme employed in the motor drive firmware. PI regulators are used to control both the torque ( $I_q$ ) and field ( $I_d$ ) current components using current feedback calculated from the three phase current measurements ( $I_{fb\ a,b,c}$ ) from the inverter. The current regulators are implemented in the synchronous DQ reference frame. The outputs of the PI regulators are then used to control a space vector modulator (SVM) which generates the gating commands to the power switches.



**Figure 3 – FOC Controller Functional Block Diagram**

When operating in direct torque control mode, the torque current ( $I_q$ ) is calculated from the torque command provided by the user. In the case of AC induction motor applications, the torque-to-current calculation is performed based on the specified induction motor parameters (PIDs 0x8080 to 0x8084). In the case of permanent magnet motor applications, the firmware simply uses the specified **PM Torque Constant** parameter (PID 0x8090) parameter to make this conversion.

When operating in closed-loop speed control mode, a PI type regulator is used to control the motor speed. The speed reference (commanded by the user) is compared to the measured motor speed providing the error input to the PI regulator. The output of the speed controller serves as the torque current command for the inner current controller.

The field current ( $I_d$ ) reference depends on which type of motor has been selected (see **Motor Drive Type**, PID 0x8060). For AC induction motors, the field current is set to the specified magnetizing current (PID 0x8084). An optional automatic field weakening controller may also be configured and enabled (PIDs 0x8088 – 0x8089) to allow for reducing the magnetizing current based on the available DC link voltage and motor speed. When initially turning on the AC motor drive, the magnetizing current may optionally be increased to reduce the time it takes to develop the corresponding magnetic field in the rotor. This may be useful in applications that want to be able to turn on quickly but are using motors with large rotor time constants. For permanent magnet motor applications, no field weakening is performed by the drive - the field current is simply commanded to zero.

The generation of the electrical angle “theta” ( $\theta$ ) also depends on the specified motor type and the specified feedback device (quadrature encoder or resolver, specified in PID 0x8061). For AC induction motor applications, the relative changes in position from the feedback device are used to calculate the mechanical speed of the motor. This speed is used in conjunction with a “slip estimator” block to determine the desired electrical speed from which the desired electrical angle is derived. The “slip estimator” requires accurate motor parameters to be specified for the induction motor (PIDs 0x8080 – 0x8084).

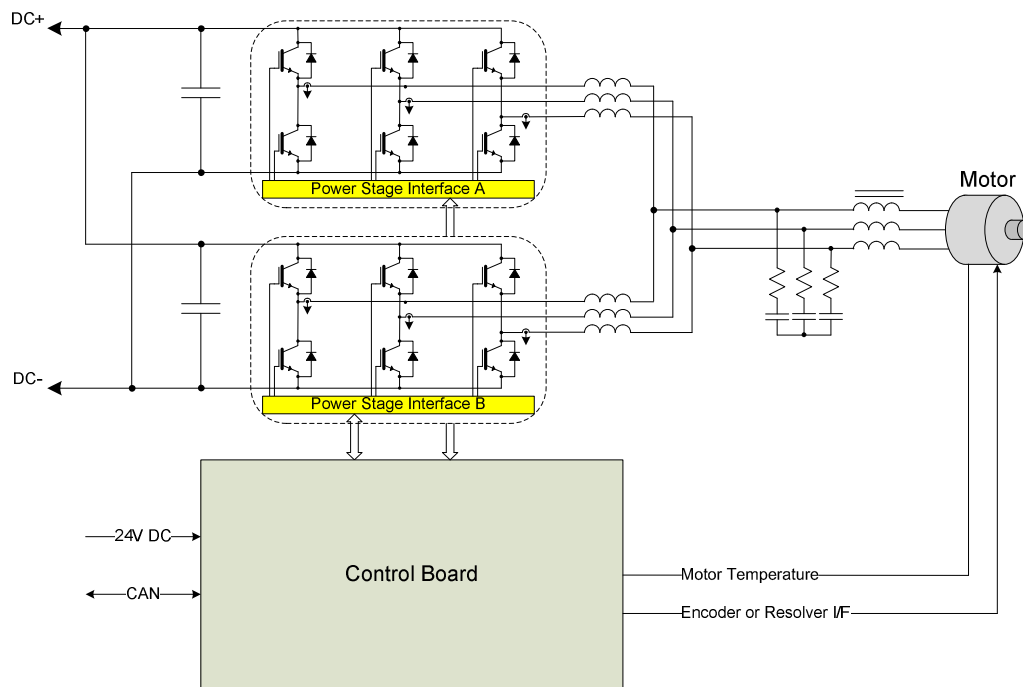
For permanent magnet motor applications, the electrical angle must be synchronous with the rotating field from the rotor, so in this case knowledge of the rotor position and field is necessary. When configured to operate with a quadrature encoder, the motor controller supports an “automatic alignment” procedure at startup. Using this procedure, the controller specifies a DC alignment current (PID 0x8092) and duration (0x8093) for which it is applied to the motor at startup. Once the specified time has passed, it is assumed that the rotor has rotated into alignment with the applied current, and the drive stores the necessary position offset required to then operate the drive. For proper alignment operation, it is required that the rotor be able to move freely. If enabled, this process is only executed the first time the drive is enabled following a power on reset of the controller. Thereafter, the controller will use the stored position offset for subsequent “turn on” events. When resolver feedback has been selected, it is assumed that the position reported by the resolver is already properly aligned with the rotor field and no additional alignment features are provided by the drive.

### 2.2.3 Dual Interleaved Inverter Operation

For specialty motor applications that require very low current ripple and harmonics, the OZMTR3000 controller provides support for an optional second inverter interface to be used when driving a dual interleaved inverter power stage. This system would look very similar to



the single inverter system shown in Figure 1 with the addition of a second IGBT bridge and a filter between the inverters and the motor as shown in the figure below.



**Figure 4 – Typical Interleaved Drive Electrical System Schematic**

From a control perspective, when operating in dual interleaved inverter mode the controller uses the same control scheme as with a single inverter with the addition of a second current controller and PWM modulator. The commanded current is generated the same as the single inverter mode and is then divided by two and sent to each of the two internal current controllers. With this scheme each inverter is handling 50% of the overall load. The PWM pulses on the second inverter use the exact same frequency as the main inverter but are phase shifted by 180 degrees to achieve the desired current ripple cancellation benefit provided by the interleaved system. Additionally, to prevent a circulating DC current from developing between the two inverters, the second current controller also uses a PI compensator to maintain a regulated DC current of zero. This interleaved control scheme is illustrated in the following figure.

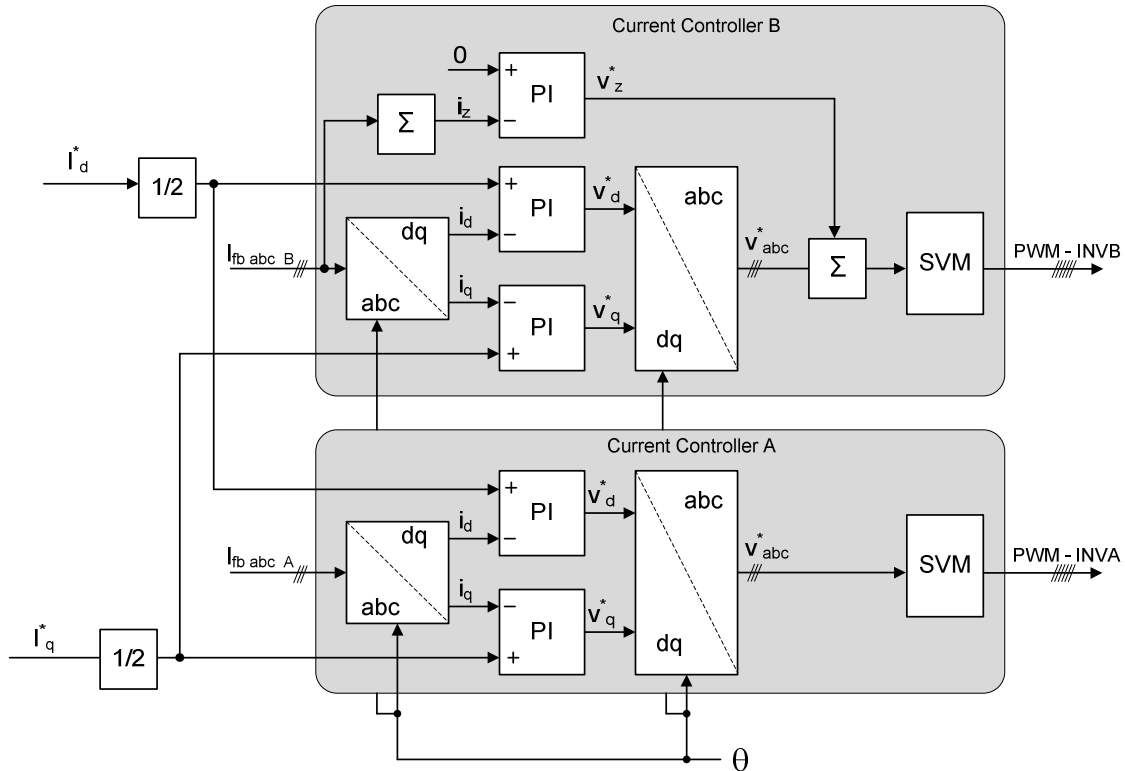


Figure 5 – Interleaved Controller Functional Block Diagram

### 2.2.4 Temperature Based Torque Derating

To prevent the inverter and/or motor from overheating and causing an over-temperature fault shutdown, the controller provides a mechanism to limit the available torque at high temperatures. Two separately enabled and configurable thermal derating blocks exist: one based on inverter temperature and one based on motor temperature.

A low and a high temperature threshold define the temperature range in which to limit the user's torque command. When the temperature is below the low threshold, the user torque command is not affected by thermal derating. When the temperature is between the low and high thresholds, the user torque command is multiplied by a derating factor that is calculated as follows:

$$DerateFactor = 1 - \left( \frac{(T_{meas} - T_{low})}{(T_{high} - T_{low})} \cdot (1 - Derate_{min}) \right)$$

The minimum derating value ( $Derate_{min}$ ) is configurable and it determines how much the drive is allowed to reduce the torque command. Above the high threshold, the derating factor will be set to this minimum value. If derating is enabled based on both inverter temperature *and* motor temperature, the derating factor is configured and calculated separately for each. The lower of the two derating factors is then chosen to reduce the user torque command. For example:

Inverter Motor Derate <sub>min</sub> = 10%	Motor Derate <sub>min</sub> = 10%
Inverter low threshold = 90 °C	Motor low threshold = 85 °C
Inverter high threshold = 100 °C	Motor high threshold = 105 °C
Inverter measured temp = 94 °C	Motor measured temp = 91 °C
Inverter Derating factor = 0.64	Motor Derating factor = 0.73

User command = 50% of Torque<sub>max</sub>    Derated user command = 50% \* 0.64 = 32%

A warning flag is reported to indicate when the user torque command is being reduced due to thermal derating.

This feature is only available when operating the drive in direct torque control mode. When operating in Volts/Hertz mode or closed-loop speed control mode, the thermal derating controller is not used.

### 2.2.5 Maximum Torque Profile

The maximum available motor torque and power is affected by the present value of the DC Link and the operating motor speed. For systems with fixed/regulated DC links, the maximum torque and power curves will be fairly deterministic and predictable. However, for systems in which the DC Link is allowed to vary (such as with battery based systems), the maximum torque and power will vary as the DC link varies. Depending on the motor application, it may be undesirable to make the user be aware of this fluctuation in torque or power. For example, in a vehicle application it may be desirable to limit the allowable torque/power based on the minimum expected DC link voltage. In doing so, the operator (i.e. driver) will not notice these fluctuations in available torque due to DC link voltage variations even when they maintain a constant position with the accelerator pedal.

To facilitate the ability to restrict the maximum torque as a function of speed, the controller provides a configurable maximum torque profile feature. This feature is enabled using the **Torque Profile Enable** configuration parameter (PID 0x80B0). The profile consists of three speed regions as follows:

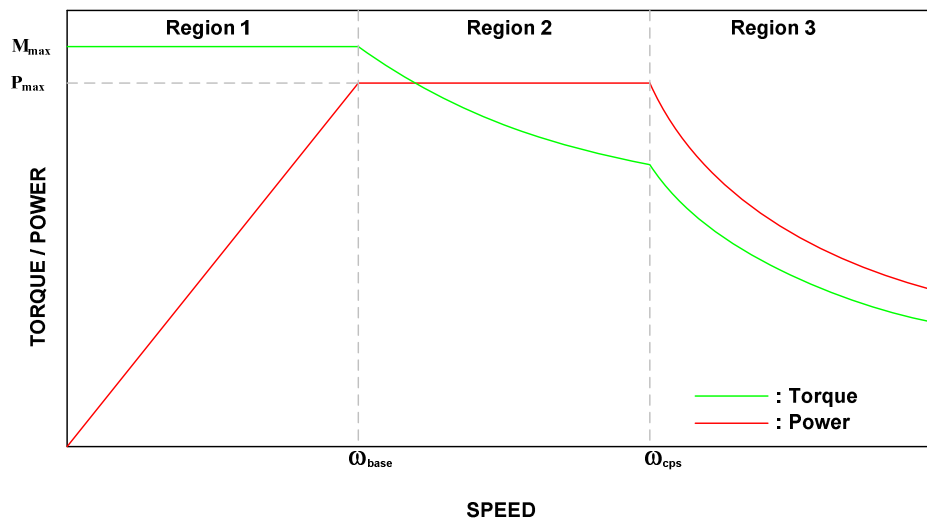
- **Region 0** (Torque Limit) – This is the low speed region of the profile, and it starts at 0 RPM. In this region, the torque command will be limited to the value specified in the **Torque Profile Maximum Torque** parameter (PID 0x80B1). Any commands from the user that exceed this value will be ignored and this value will be used instead.
- **Region 1** (Power Limit) – This is the mid speed region. In this region, the motor power is limited to the value specified in the **Torque Profile Maximum Power** parameter (PID 0x80B2). The speed at which this region begins can be calculated based on the fact that motor power equals torque times speed. Therefore, the speed at which the motor will reach this maximum power threshold can be calculated as the specified maximum

power divided by the specified maximum torque. This speed ( $\omega_{base}$ ) will be in units of radians/second when using units of Watts for power and Nm for torque.

When the user commands a torque that would result in a motor power that exceeds the specified maximum power, the controller will automatically limit the torque so that the resulting power is clamped to the maximum value.

- Region 2 (Power\*Speed Limit)** – This is the high speed region. This region begins at the motor speed specified by the **Torque Profile Constant Power\*Speed Region Start** parameter (PID 0x80B3). This speed value is used along with the specified maximum power to calculate a maximum “Power\*Speed” value. As the motor speed increases, the controller will calculate the power \* speed value and compare it to the maximum value specified. If the maximum is exceeded, the torque will then be reduced such that the power is reduced to maintain the maximum “Power\*Speed” value.

The figure below shows a pictorial summary view of the three regions created by the torque profile. Note that the  $\omega_{cps}$  represents the speed at which Region 3 starts, as specified in parameter PID 0x80B3.



**Figure 6 – Torque and Power in the Three Speed Regions**

This torque profile feature is only available when operating the drive in direct torque control mode. When operating in Volts/Hertz mode or closed-loop speed control mode, the torque profile is not used.

### 2.2.6 Current Fold Back

For protection purposes, the drive supports a current fold back feature for both the torque ( $I_q$ ) and field ( $I_d$ ) current commands in the event that the applied stator voltage gets too close to the maximum achievable level dictated by the DC link voltage. The fold back threshold is

independently programmable for both  $I_q$  and  $I_d$  (PIDs 0x80F3 and 0x80F7). The threshold defines the maximum modulation index (percent of applied voltage relative to the DC link voltage), above which the corresponding current will be reduced.

For most applications the system should be designed such that the DC link voltage provides ample voltage headroom to achieve the desired speed and torque from the motor. In this case, the fold back controllers are not required and remain disabled (PIDs 0x80F2 and 0x80F6). Otherwise, for systems with minimal voltage headroom these fold back controllers may be employed to prevent losing current regulation and potentially faulting off the drive.

To limit the torque producing current ( $I_q$ ), simple PI regulator is used to decrease the current in order to maintain an applied voltage below the specified modulation index. The output of the PI regulator is a “limit factor”, a value between 0 and 1.0 that is multiplied with the desired current command. When the DC link voltage high enough to produce the desired torque without exceeding the specified modulation index, the current will not be reduced (the compensator output will be 1.0). If the DC link were to drop, causing the modulation index to increase to the specified limit, the commanded current will be linearly reduced based on the compensator output. When this condition occurs, a warning flag is asserted that indicates to the user that the current is presently being limited by this controller.

In the event that the torque current is reduced to zero and the modulation index is still too high, a second controller may also be used to fold back the magnetizing current ( $I_d$ ). This second controller is only useful when the drive is configured to control an AC induction motor (i.e. where a magnetizing current is being regulated). For permanent magnet motors, the  $I_d$  current is already being controlled to zero and cannot be reduced any further. The fold back threshold for the  $I_d$  controller should be set higher than the  $I_q$  fold back threshold so that it will not be engaged unless the  $I_q$  fold back is not adequate. In general, the  $I_d$  current should not need to be clamped, particularly if field weakening has been enabled (PID 0x8088). The  $I_d$  fold back has been added as a potential backup protection mechanism. This may be useful for systems with fast dynamic changes in either motor speed or DC Link voltage, both of which affect field weakening. The rate of change of the actual field within the motor is based on the rotor time constant, so there could be scenarios where the  $I_d$  current may need to fold back at high speeds and at high rates of change of either the DC Link voltage or motor speed.

This feature is only available when operating the drive in closed-loop field oriented control (i.e. not Volts/Hertz mode). This feature is mainly targeted at applications that are running in direct torque control mode, where the torque command is simply reduced. When this feature may also be enabled and used when the drive is operated in closed-loop speed control mode. However, if the fold back feature is engaged when running in speed mode, the speed loop will no longer be in regulation and the resulting motor speed will change based on the load conditions.

## 2.3 Brake Controller

For scenarios where the motors are creating regenerative energy that is flowing back to the DC link (as with commanding a braking torque) and causing the DC link voltage to rise, this controller provides an optional brake controller. This controller can be used to engage a brake resistor across the DC link, thus preventing the voltage from rising to high values that could result in over-voltage fault shut downs or even component damage.

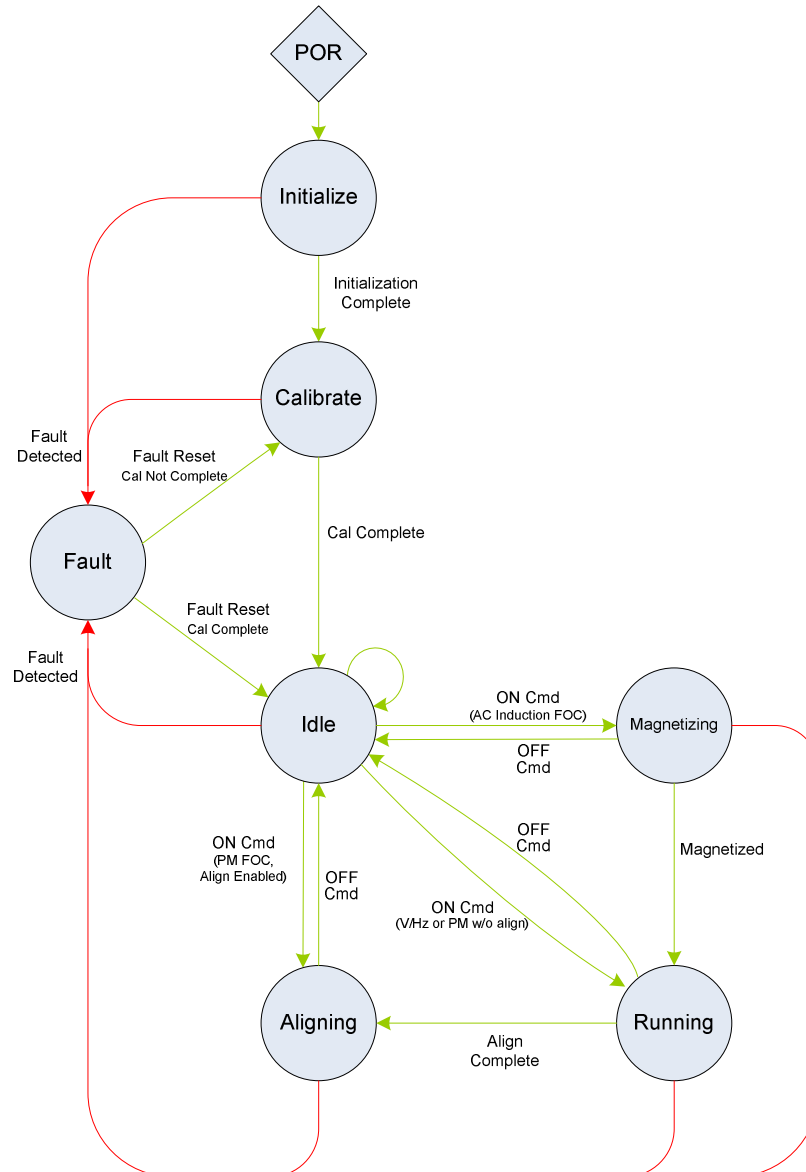
When enabled (**Brake Controller Enable**, PID 0x8110), the controller uses two voltage thresholds (PIDs 0x8111 and 0x8112) to create a simple hysteretic control for engaging the brake resistor. If the voltage rises above the upper trip threshold, the brake resistor will be engaged. Once the voltage then falls below the lower recover threshold, the brake resistor will be disconnected.

The brake contains optional hardware fault and temperature monitoring capabilities in the event that these are provided by the brake chopper power device and its corresponding gate drive solution. In the event that the fault is detected, the brake controller state machine will go to a FAULT state and braking function will be disabled. If this occurs while the motor drive is in operation, the **Brake Controller Fault Motor Response** parameter (PID 0x8113) can be used to configure the motor drive response as follows:

- Continue operating, ignoring the brake fault
- Treat this as a drive fault and shut down, regardless of the configured drive type and torque/speed mode being used
- Continue operating, but not allow a braking torque command – this is only pertinent when operating the drive as a closed-loop FOC controller running in direct torque control mode; otherwise, for speed control or Volts/Hertz operation this setting is ignored

## 2.4 State Sequencing

A state machine is used to provide deterministic control and sequencing of the motor drive states. If a fault is detected in any of the operating states, the inverter is immediately disabled and the state machine is latched into the FAULT state. Figure 7 illustrates the operating states as well as the transition logic employed in the controller's state machine.



**Figure 7 – Controller State Machine**

### 2.4.1 Initialize

The state machine resets to the *Initialize* state following a power-on-reset (POR) event. While in this state the power hardware is not operable; the firmware is initializing hardware peripherals, configuring variables, and performing self health tests. Upon successful initialization the state machine will auto-transition to the *Calibrate* state.

### 2.4.2 Calibrate

The *Calibrate* state is used to calibrate system hardware as applicable. Power hardware is not operable while in the *Calibrate* state. Following successful calibration, the state machine will auto-transition to the *Idle* state.

### 2.4.3 Idle

Once in the *Idle* state the controller is ready for use. The state machine will remain in the *Idle* state indefinitely, transitioning either due to a fault condition or a turn-on command. When turning on, the next state depends on the configured **Motor Drive Type** (PID 0x8060). For closed-loop field oriented control of an AC induction motor, the state machine will sequence to the *Magnetizing* state. For permanent magnet operation, if the **PM Align At Startup Enable** parameter is set to TRUE (PID 0x8090), and this is the first time turning on after a POR, the state machine will transition to the *Align* state. For all other cases, the state machine will transition to the *Running* state.

### 2.4.4 Magnetizing

When in this state, the controller is ramping up the magnetizing current in the AC induction motor to the desired level. Once at this initial magnetizing current, the controller waits for the induced magnetic field to finish charging in the rotor (based on the estimated rotor time constant). While magnetizing, the torque current is forced to zero amps and all attempts send a torque or speed command will be ignored. Once magnetization is complete, the state machine will transition to the *Running* state, where the user torque or speed command will then be accepted. If an OFF command is received during the magnetization process, the controller will immediately transition back to the *Idle* state and will disable the inverter.

### 2.4.5 Align

When in this state, the controller is performing the automatic alignment process for the permanent magnet motor. The controller will remain in this state for the period of time specified in the **PM Align Time** parameter (PID 0x8093), during which it will be regulating the align current specified in the **PM Align Current** parameter (PID 0x8092). While aligning, all attempts send a torque or speed command will be ignored. Once alignment is complete, the state machine will transition to the *Running* state, where the user torque or speed command will then be accepted. If an OFF command is received during the alignment process, the controller will immediately transition back to the *Idle* state and will disable the inverter.

### 2.4.6 Running

While in the *Running* state the motor drive is enabled and processing either speed or torque commands from the user. The state machine will remain in the *Running* state indefinitely, transitioning either on a fault or a turn-off command. When a turn-off command is received, the state machine will immediately disable the inverter and go to the *Idle* state.

### 2.4.7 Fault

The *Fault* state indicates that a latched fault condition has occurred and that the drive is inoperable. The various fault conditions are reported in the **Fault Status** instrumentation register (PID 0x400E) and also reported on the CAN bus in the Alarm Status CAN message.



The controller remains in the *Fault* state and the latched fault flags remain set until explicitly reset with a **Fault Reset** command (PID 0x0004). This is true even if the source(s) of the fault(s) are no longer active. Upon receiving the **Fault Reset** command, the firmware will attempt to clear all latched fault bits. It then examines the sources of all fault conditions and if any sources of faults are still active, their respective fault flags are latched again and the controller will remain in the *Fault* state. Otherwise, if no faults remain active, the controller will transition either back to the *Calibrate* state if the calibration process was not completed, or back to the *Idle* state if calibration has already been run.

## 2.5 Fault and Warning Conditions

The motor controller provides warning indicators and fault protection in the event of conditions that may cause damage to the equipment or injure personnel. The various conditions that are monitored by the controller are listed and described in the following sections.

### 2.5.1 Warnings

The motor controller provides the warning indicators listed below. These warning conditions do not prohibit operation of the drive; they are merely reported for informational purposes only. Each warning condition described below is reported in the **Warning Status** instrumentation register (PID 0x400D) and also reported on the CAN bus in the Alarm Status CAN message.

#### 2.5.1.1 High Inverter Temperature

If the **Inverter Temp Enable** parameter (PID 0x80B8) is set to TRUE, the firmware will monitor the main Inverter A temperature and will set a warning flag if it exceeds the **Inverter Temperature Warning Threshold** configuration parameter (PID 0x8034). If the **Interleaved Inverters Enable** configuration parameter (PID 0x8108) is also set to TRUE, the firmware will monitor the secondary Inverter B temperature and compare this to the same warning threshold value. These flags will remain set until the temperature(s) falls below the **Inverter Temperature Recover Threshold** configuration parameter (PID 0x8035).

#### 2.5.1.2 High Motor Current

The firmware monitors the RMS inverter current going to the motor and will set a flag if it exceeds the **Motor Over Current Warning Threshold** configuration parameter (PID 0x803D). It does this for both the main Inverter A and the secondary Inverter B if the **Interleaved Inverters Enable** configuration parameter (PID 0x80108) is set to TRUE. These flags remain set until the respective current falls below the **Motor Over-Current Recover Threshold** configuration parameter (PID 0x803E). In the case of dual interleaved inverter operation, the warning and recover thresholds are first divided by two and then compared against each inverter current (A and B).

#### 2.5.1.3 High Motor Temperature

If the **Motor Temp Enable** parameter (PID 0x80C1) is set to TRUE, the firmware will monitor the motor's temperature sensor and will set a warning flag if the temperature exceeds the **Motor**

**Temperature Warning Threshold** configuration parameter (PID 0x803A). This flag will remain set until the temperature falls below the **Motor Temperature Recover Threshold** configuration parameter (PID 0x803B).

#### 2.5.1.4 High Motor Speed

If a speed feedback device has been enabled by the **Motor Speed Feedback Device Select** parameter (PID 0x8061), the firmware will monitor the motor's speed and will set a warning flag if the speed exceeds the **Motor Over Speed Warning Threshold** configuration parameter (PID 0x8040). This flag will remain set until the speed falls below the **Motor Over Speed Recover Threshold** configuration parameter (PID 0x8041).

#### 2.5.1.5 Clamped Torque Command

If the **Torque Profile Enable** parameter (PID 0x80B0) is set to TRUE and the user attempts to operate the drive in torque control mode with a torque command above the maximum allowed by the profile, the firmware will limit the applied torque to the maximum value from the profile. When this is the case, the motor drive will set this flag to indicate that the requested torque command is not being used and that a lower torque is actually being commanded. This warning flag is automatically cleared when the motor torque returns to being controlled to the requested torque value from the user.

#### 2.5.1.6 Braking Torque Disabled

If the brake controller is enabled (PID 0x8110), the **Brake Controller Fault Motor Response** parameter (PID 0x88113) is set to "1" and a brake fault is detected, the firmware will not allow braking torque commands from the user. Instead, the torque command will be clamped to zero to override the braking command from the user. If this occurs, the controller will set this warning flag to indicate to the upper-level controller that the requested braking torque command is not being used and that the torque has been set to zero. The warning flag will be cleared when no clamping is being performed (either the brake fault is cleared or the user's torque command changes back to a 'motoring' command).

#### 2.5.1.7 Motor Torque Reduced Due to Thermal Derating

If the controller has been configured to enable thermal derating from either a high motor temperature (PID 0x80C6) or a high inverter temperature (PID 0x80BD), the firmware will reduce the user torque command applied to the motor when the corresponding temperature is too high. When this occurs, the controller will set this warning flag to indicate to the upper-level controller that the requested torque command is not being used and that a lower torque is actually being commanded due to thermal derating. The warning flag will be cleared when no thermal derating is being performed (i.e. the temperature(s) are below their minimum derating values).

#### 2.5.1.8 $I_q/I_d$ Current Foldback

If the current fold-back feature is enabled (PID 0x80F2 or 0x80F6), the firmware will reduce the  $I_q$  and/or  $I_d$  currents if the voltage applied to the motor gets too close to the full utilization of the DC Link voltage (i.e. a high modulation index). When this is the case, the motor drive will set these bits (one for  $I_d$ , one for  $I_q$ ) to indicate that corresponding current is actively being

reduced. These warning flags are automatically cleared when the corresponding current is no longer being clamped.

#### 2.5.1.9 High DC Link Voltage

The firmware monitors the DC link voltage and will set a warning flag if it exceeds the **DC Link Over Voltage Warning Threshold** configuration parameter (PID 0x8031). This flag will remain set until the voltage falls below the **DC Link Over Voltage Recover Threshold** configuration parameter (PID 0x8032).

#### 2.5.1.10 High Brake Switch Temperature

If the **Brake Temp Enable** parameter (PID 0x80CA) is set to TRUE, the firmware will monitor the brake switch temperature sensor and will set a warning flag if the temperature exceeds the **Brake Switch Temperature Warning Threshold** configuration parameter (PID 0x8037). This flag will remain set until the temperature falls below the **Brake Switch Temperature Recover Threshold** configuration parameter (PID 0x8038).

#### 2.5.1.11 Local Bias Supply Tolerance Warnings

The firmware monitors the local bias supplies (24V, 15V, 5V, 3.3V, and -15V) on the control board and will set a warning flag if the corresponding supply voltage is not within the range required by the on-board hardware. The various warning flags will remain set while the supply voltages are out of tolerance and will be cleared when the supply is within the required limits.

### 2.5.2 Faults

The motor controller provides the fault protection listed below. Whenever a fault occurs the firmware will automatically turn the inverter OFF and transition to the FAULT state. Each fault condition is latched and reported in the **Fault Status** instrumentation register (PID 0x400E) and also reported on the CAN bus in the Alarm Status CAN message.

The controller remains in the FAULT state and the latched fault flags remain set until explicitly reset with a **Fault Reset** command (PID 0x0004). This is true even if the source(s) of the fault(s) are no longer active. Upon receiving the **Fault Reset** command, the firmware will attempt to clear all latched fault bits. It then examines the sources of all fault conditions and if none are active the controller will transition back to the IDLE state. If upon re-examination any sources of faults are still active, their respective fault flags are latched again and the controller will remain in the FAULT state.

#### 2.5.2.1 Inverter IGBT Errors

The firmware provides a means to recognize hardware-based IGBT error signals from the main Inverter A interface (as is present on a typical Semikron SKiiP interface). If the **Interleaved Inverters Enable** configuration parameter (PID 0x80108) is set to TRUE, the firmware will also monitor the error signals from the secondary Inverter B interface.

### 2.5.2.2 Inverter Hardware Over Temperature

The firmware provides a means to recognize a hardware-based over-temperature error signal from the main Inverter A interface (as is present on a typical Semikron SKiiP interface). If the ***Interleaved Inverters Enable*** configuration parameter (PID 0x8108) is set to TRUE, the firmware will also monitor the over-temperature signal from the secondary Inverter B interface.

### 2.5.2.3 Software Inverter Over Temperature

The firmware monitors the main Inverter A temperature and will assert a fault if it exceeds the ***Inverter Temperature Fault Threshold*** configuration parameter (PID 0x8033). If the ***Interleaved Inverters Enable*** configuration parameter (PID 0x8108) is set to TRUE, the firmware will also monitor the temperature from the secondary Inverter B interface and check this against the same fault threshold value.

### 2.5.2.4 Software Inverter Over Current

If the ***Inverter Temp Enable*** configuration parameter (PID 0x80B8) is set to TRUE, the firmware will monitor the RMS current to the motor from the main Inverter A and asserts a fault if it exceeds the ***Motor Over Current Fault Threshold*** configuration parameter (PID 0x803C). If the ***Interleaved Inverters Enable*** configuration parameter (PID 0x8108) is also set to TRUE, the firmware will monitor the current from the secondary Inverter B interface as well. In the case of dual interleaved inverter operation, the fault threshold is first divided by two and then compared against the current from each inverter.

### 2.5.2.5 Motor Over Temperature

If the ***Motor Temp Enable*** configuration parameter (PID 0x80C1) is set to TRUE, the firmware will monitor the motor's temperature sensor and assert a fault if the temperature exceeds the ***Motor Temperature Fault Threshold*** configuration parameter (PID 0x8039).

### 2.5.2.6 Motor Over Speed

If a speed feedback device has been enabled by the ***Motor Speed Feedback Device Select*** parameter (PID 0x8061), the firmware will monitor the motor's speed and will set this flag if the speed exceeds the ***Motor Over Speed Fault Threshold*** parameter (PID 0x803F).

### 2.5.2.7 Motor Align Error

This fault is used to indicate when the drive is configured to operate as a permanent magnet motor drive with alignment at startup enabled (PID 0x8091) and the motor was spinning faster than the allowable maximum align speed (PID 0x8094).

### 2.5.2.8 Brake Switch Hardware Error

The firmware provides a means to recognize a hardware-based error signal from the brake chopper interface (as is present on a typical Semikron SKiiP Half Bridge or Brake Chopper module).

### 2.5.2.9 Brake Switch Hardware Over Temperature

The firmware provides a means to recognize a hardware-based over-temperature error signal from the brake chopper interface (as is present on a typical Semikron SKiiP Half Bridge or Brake Chopper module).

### 2.5.2.10 Brake Switch Software Over Temperature

If the **Brake Temp Enable** configuration parameter (PID 0x80CA) is set to TRUE, the firmware will monitor the brake switch temperature and will assert a fault if it exceeds the **Brake Switch Temperature Fault Threshold** configuration parameter (PID 0x8036).

### 2.5.2.11 DC Link Over-Voltage

The firmware monitors the DC link voltage and will assert a fault if it exceeds the **DC Link Over-Voltage Fault Threshold** parameter (PID 0x8030).

### 2.5.2.12 Configuration Memory Error

This fault occurs any time a read from the configuration memory is performed and the CRC for the block being read does not match the CRC stored in the memory. This may occur if the data was corrupted in transmission during the read or this may occur if the data stored in memory was corrupted. Unlike all other fault sources, this fault condition is not cleared with the **Fault Reset** command (PID 0x0004) as the fault condition indicates the possibility that the control parameters are not as intended. Instead, this fault is considered a major system fault and needs to be addressed as follows:

1. Attempt to reload the system configuration using the **Configuration Reload** command (PID 0x0007) or cycle power to the control board. Either of these actions will reset the CPU on the control board which then forces a re-initialization of the application, including reading the configuration parameters from the external configuration memory. If this completes without error, this implies the previous error occurred while the data was being read.
2. If after executing step 1 above a configuration error is still present, this may indicate bad data in the configuration memory. In this case, the memory will need to be reset to the factory defaults using the **Configuration Reset** command (PID 0x0006). Once the memory contents have been reset and any values changed to their customized settings, step 1 above should be executed to force a reload of the system variables.
3. If neither of the above result in clearing the configuration error, then there is likely an issue with the control board hardware. At this point the board should be sent back to the factory for diagnosing and repairing of any defects (see the RMA process described at the end of this document).

### 2.5.2.13 Calibration Error

When first powering up the control board the firmware attempts to calibrate the controller's internal ADC. This error is asserted if the controller is unable to perform the required calibration. There is likely an issue with the control board hardware if this error occurs, in which case the board should be sent back to the factory for diagnosing and repairing of any defects (see the RMA process described at the end of this document).

### 2.5.2.14 Communications Timeout

If the **CAN Timeout** configuration parameter (PID 0x8015) is set to a non-zero value, the firmware will monitor the amount of time elapsed between received CAN messages. A fault will be asserted if the specified timeout threshold is exceeded.

## 3. OZDSP3000 Hardware Interfacing

The OZDSP3000 is a highly integrated DSP control solution for power control applications. Typical applications include voltage output inverters, grid-tie inverters, AC induction motor controllers, brushless DC motor controllers, and Active Front-End regulators. This section describes the hardware interfaces on the OZDSP3000 that are supported by the motor controller firmware in a typical motor drive application.

### 3.1 Application Interfaces

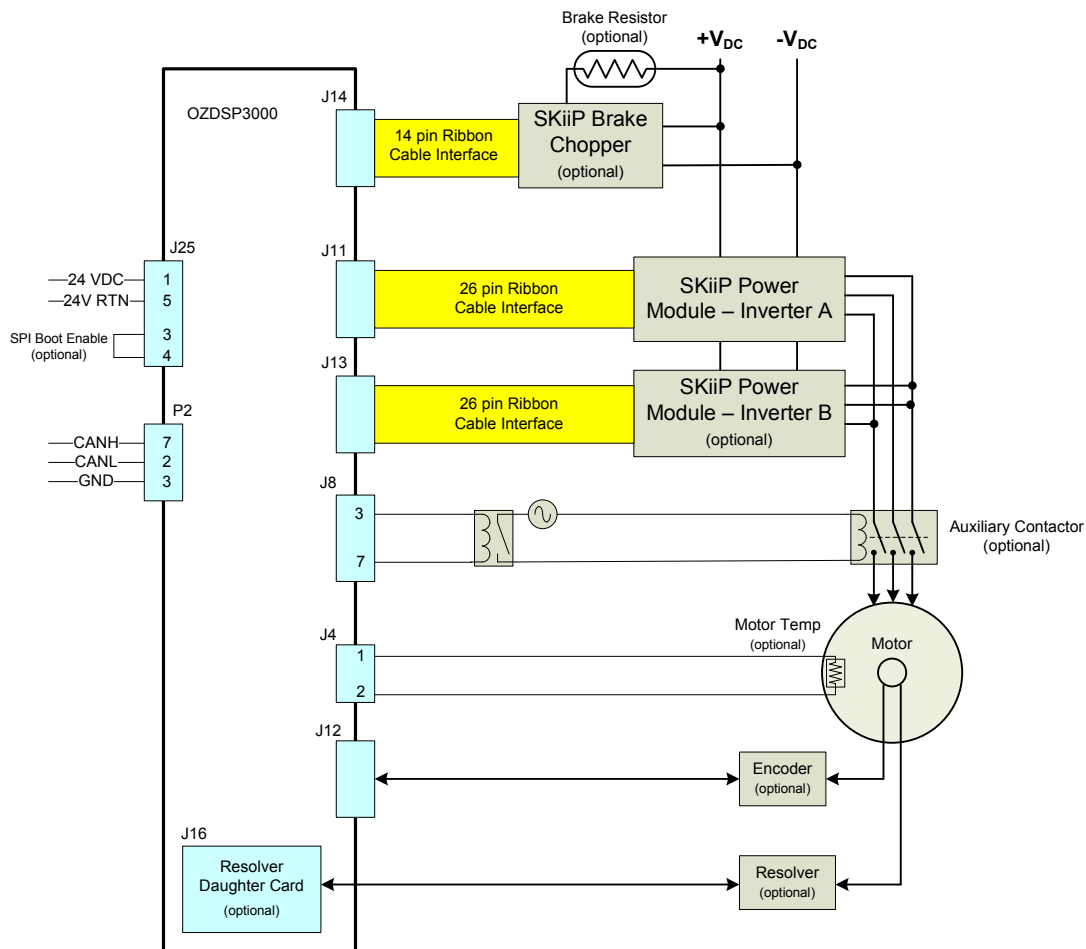


Figure 8 – OZDSP3000 Motor Drive Application Electrical Connections

**3.1.1 J11: SKiiP Power Module Interface – Inverter A**

Connector J11 provides an interface to SKiiP style Semikron power modules. This interface complies with Semikron’s required specifications. The OZDSP3000 supplies 24V power to the power module via pins 14 and 15. PWM commands (15V logic level) are supplied to the top and bottom switches of each of the three half bridges via signals INV\_TOP\_U/V/W and INV\_BOT\_U/V/W respectively. Feedback of phase currents, temperature, and DC link voltage are provided on INV\_IOUT\_U/V/W, INV\_TEMP, INV\_UDC signals respectively. Error signals from the module are similarly provided via INV\_ERR\_U/V/W and INV\_OVT. Please refer to the Semikron datasheet for the particular module being used for more information.

**Table 1 – J11 SKiiP Power Module Pin Assignment**

Pin #	Description
1	Ground
2	INV_BOT_U (Phase A)
3	INV_ERR_U (Phase A)
4	INV_TOP_U (Phase A)
5	INV_BOT_V (Phase B)
6	INV_ERR_V (Phase B)
7	INV_TOP_V (Phase B)
8	INV_BOT_W (Phase C)
9	INV_ERR_W (Phase C)
10	INV_TOP_W (Phase C)
11	INV_OVR_TEMP
12	n/c
13	INV_UDC
14	24V
15	24V
16	n/c
17	n/c
18	Ground
19	Ground
20	INV_TEMP
21	INV_IOUT_U_RTN (Phase A)
22	INV_IOUT_U (Phase A)
23	INV_IOUT_V_RTN (Phase B)
24	INV_IOUT_V (Phase B)
25	INV_IOUT_W_RTN (Phase C)
26	INV_IOUT_W (Phase C)

- **OZDSP3000 Connector Part Number:** (AMP) 499922-6
- **Mating Connector Part Number:** (AMP) 1658621-6
- **Power:** 24V @ 1.5A

### 3.1.2 Custom Driver Interface Considerations

When attempting to use the OZDSP3000 motor controller with a custom designed power stage the hardware must be designed to provide the appropriate signals expected at the J11 interface. Generally some sort of custom printed circuit board will be required to interface the J11 signals to the gate drivers, current sensors, etc.

#### 3.1.2.1 Power

The OZDSP3000 supplies 24V on J11, pins 14 & 15. This 24V may be used to power the electronics on the interface board. The supply is capable of providing 1.5A.

#### 3.1.2.2 Switch Commands

The six switching commands are provided on pins 2, 4, 5, 7, 8, and 10. These switch command signals are driven off of the OZDSP3000 at 15V logic level using MC14504B level shifting devices.

#### 3.1.2.3 Error Inputs

The OZDSP3000 expects three logic level, error inputs; one associated with each phase, on pins 3, 6, and 9. When active, these inputs cause the firmware to latch the system off into the FAULT state. Pull-ups to 3.3V are provided on board and the signals are active high. Depending on the features provided in the custom design, these signals can be used to interface single error sources, multiple protection circuits, or none at all.

The custom interface board should drive the pin with an open-collector style circuit. In the case where no protection is provided, the pins should be grounded to disable the faults.

#### 3.1.2.4 DC Link Voltage Sensing

The OZDSP3000 expects a signal proportional to DC link voltage to be provided on J11, pin 13 with respect to pin 21. This signal should be scaled such that 0-10 V represents the measurable DC link voltage range.

#### 3.1.2.5 DC Link Hardware Over-Voltage Protection

When designing a custom interface it is highly recommended that hardware over voltage protection be implemented. This can be implemented with a comparator using the DC link voltage sense output. The output of this comparator can be used to gate off the switch commands as well as assert the error inputs on each phase.

#### 3.1.2.6 Hardware Over-Current and Desaturation Protection

Semikron SKiiP power modules provide fast hardware over current and desaturation protection. When designing a custom power solution, these additional protection features



should also be considered. When including over current, desaturation, and over voltage protection into the design, the fault flags must be logically OR'd together and reported using the open collector Error signal inputs to the OZDSP3000.

### 3.1.2.7 Current Sense Signals

The OZDSP3000 expects to receive a current sense signal for each half bridge phase output. This should be a bipolar signal where +/-10V corresponds to the full scale current range. The current sense signals should be provided on the following pins:

**Table 2 – J11 Current Sense Pin Assignment**

J11 Pin #	Description
22	Current Phase A (U)
21	Gnd Reference for Current Phase A (U)
24	Current Phase B (V)
23	Gnd Reference for Current Phase B (V)
26	Current Phase C (W)
25	Gnd Reference for Current Phase C (W)

### 3.1.2.8 Temperature Sense Signals

The OZDSP3000 expects to receive a 0-10V temperature signal on pin 20 that corresponds to the hot spot temperature of the power devices.

### 3.1.3 J13: SKiiP Power Module Interface – Inverter B (optional)

Connector J13 provides a second interface to SKiiP style Semikron power modules. This interface is identical to the J11 interface described in the previous section. This is an optional interface that is only used if the OZDSP3000 motor controller is configured to drive a dual interleaved inverter power stage (if the **Dual Interleaved Inverters Enable** configuration parameter PID 0x8108 is set to TRUE).

When this second inverter interface is enabled, all of the J13 signals are used with the exception of the DC link voltage sense input (INV\_UDC, pin 13). For dual interleaved inverter operation, both inverters must be tied to the same DC link and therefore only the voltage sense input from Inverter A is used.

The J13 interface is not used when the second inverter interface is not enabled. In this case it is safe to leave connector J13 unconnected.

### 3.1.4 J8: Auxiliary Contactor/Relay Drive Interface (optional)

The controller provides the user with basic ON/OFF control for an auxiliary relay or contactor. If this optional interface is used, the device must be connected according to the table below. If this feature is not used, the J8 interface can be left unconnected.

**Table 3 – J8 Relay Drive Pin Assignment**

Pin #	Description
3	Relay 24V drive
7	Relay drive return

- **OZDSP3000 Connector Part Number:** (Molex) Micro-Fit 2x4 Header: 43045-0824
- **Mating Connector Part Number:** (Molex) Micro-Fit 2x4 Receptacle: 43025-0800
- **Output Range:** 24V, 2A continuous, 5A inrush

### 3.1.5 J25: Bias Power Input & SPI Boot Enable

The OZDSP3000 requires 24VDC power input on terminal block J25, pins 1 and 5. Pins 2 and 4 are connected to the 24VDC/GND supply for optional external use of the supply connections.

Pin 3 is used for SPI boot enable. It should be left floating or connected to ground for normal Flash Boot operation (SPI bootloader is disabled). It should be connected to 24VDC (pin 2) to enable the SPI bootloader functionality.

**Table 4 – J12 RS-485 Pin Assignment**

Pin #	Description
1	24 VDC
2	24 VDC (spare)
3	SPI Boot Enable
4	24V Return (spare)
5	24V Return

- **OZDSP3000 Connector Part Number:** (Phoenix Contact) 1733606
- **Mating Connector Part Number:** n/a (terminal block style)
- **Voltage:** 24V nominal, 18V min, 28V max
- **Current:** 4.5A maximum (Inrush current while powering three SKiiP modules)

### 3.1.6 J14: Brake Chopper Power Module Interface (optional)

Connector J14 provides an optional interface to a SKiiP style Semikron half bridge and brake module. This interface is only used if the **Brake Controller Enable** parameter (PID 0x8110) is set

to TRUE. Otherwise, if the brake controller is not enabled, this interface is not used and may be left unconnected.

**Table 5 – J14 Half Bridge Power Module Pin Assignment**

Pin #	Description
1	Ground
2	HB_BOT
3	HB_ERR
4	HB_TOP ( <i>not used</i> )
5	HB_OVR_TEMP
6	24V
7	24V
8	n/c
9	n/c
10	Ground
11	Ground
12	HB_TEMP
13	HB_IOUT_RTN ( <i>not used</i> )
14	HB_IOUT ( <i>not used</i> )

- **OZDSP3000 Connector Part Number:** (AMP) 499922-2
- **Mating Connector Part Number:** (AMP) 1658621-2
- **Power:** 24V @ 1.5A

The OZDSP3000 supplies 24V power to the brake module via pins 6 and 7. A brake enable/disable command (15V logic level) is provided on the HB\_BOT pin (pin 2). The polarity of this pin is programmable using the **Brake Gate Driver Signal Active High** parameter (PID 0x80114).

The OZDSP3000 monitors the hardware error (HB\_ERR, pin 3) and hardware over temperature (HB\_OVR\_TEMP, pin 5) signals from the power module. These are logic level open-collector inputs to the OZDSP3000. When active, these inputs cause the firmware to disable the brake control function. Pull-ups to 3.3V are provided on board. In the case where one or both of these signals is not provided by the power module, the pins should be grounded to disable the respective fault condition.

The brake controller may be configured to monitor the temperature of the brake switch (**Brake Temp Enable**, PID 0x80CA) if this feature is provided by the power module. If enabled, the controller expects to receive a 0-10V temperature signal on pin 12 that corresponds to the hot spot temperature of the power device.

The top half-bridge PWM output (HB\_TOP) and the current feedback pins (HB\_IOUT and HB\_IOUT\_RTN) are not used by the brake controller and can be left unconnected.

### 3.1.7 J12: Quadrature Encoder Interface (optional)

Connector J12 provides an optional interface to an incremental, quadrature encoder. The interface provides for the A and B quadrature inputs. If enabled to use the encoder interface (see parameter **Motor Position/Speed Feedback Device Select**, PID 0x8061), the controller will use this interface for determining the motor speed using the A and B inputs. The controller does not use this interface for absolute position feedback, and as such, the index signal (pins 5 and 10) is not used; these pins may be left unconnected. 5V power and ground are also provided on the connector to power the encoder device.

**Table 6 – J12 Encoder Interface Pin Assignment**

Pin #	Description
1	5V
2	No connect
3	A Signal – Positive
4	B Signal – Positive
5	Index – Positive ( <i>not used</i> )
6	No Connect
7	Ground
8	A Signal – Negative
9	B Signal – Negative
10	Index – Negative ( <i>not used</i> )

- **OZDSP3000 Connector Part Number:** (Molex) Micro-Fit 2x5 Header: 43045-1012
- **Mating Connector Part Number:** (Molex) Micro-Fit 2x5 Receptacle: 43024-1000
- **Power:** 5V, 300mA
- **Differential termination:** 100Ω
- **Electrical Interface:** Differential inputs. Logic high:  $V_{ID} \geq 0.2V$ . Logic low:  $V_{ID} \leq -0.2V$
- **Max Rating:** Input voltage:  $\pm 14V$ . Differential voltage:  $\pm 14V$

### 3.1.8 J16: Expansion Board Interface – Resolver Daughter Card (optional)

The OZDSP3000 control board provides an expansion board interface at the J16 connector. An optional Resolver Interface daughter board (Oztek part 10780-xx) has been developed for the motor controller application. This daughter board is designed to mount on this expansion board interface. Contact Oztek sales for additional information the resolver daughter board.

### 3.1.9 J4: Motor Temperature Input (optional)

Connector J4 may optionally be used by the user to connect an external temperature sensor for the purpose of monitoring the motor temperature. This interface provides connections for a typical 10k $\Omega$  thermistor.

**Table 7 – J4 Pin Assignment**

Pin #	Description
1	Temperature input (pulled up to +3.0V through 2k $\Omega$ ).
2	Ground (24V Return)

- **OZDSP3000 Connector Part Number:** (Molex) Micro-Fit 2 Position Header: 43650-0215
- **Mating Connector Part Number:** (Molex) Micro-Fit 2 Position Receptacle: 43645-0200

### 3.1.10 P2: Isolated CAN Bus Interface

Connector P2 provides an isolated CAN Bus communications interface. The interface is a shielded, female, DB9 style connector.

**Table 8 – P2 CAN Bus Pin Assignment**

Pin #	Description
2	CAN Low
3	CAN Ground (Isolated)
7	CAN High

- **OZDSP3000 Connector Part Number:** (AMP) 747844-5
- **Mating Connector Part Number:** Industry Standard DB9 Male

#### 3.1.10.1 J18/19 CAN Termination Jumpers

Jumper blocks J18 and J19 provide a means to terminate the CAN bus lines CAN high (CANH) and CAN low (CANL). Note that termination should only be placed at the end terminals of the CAN communication network, reference Figure 9.

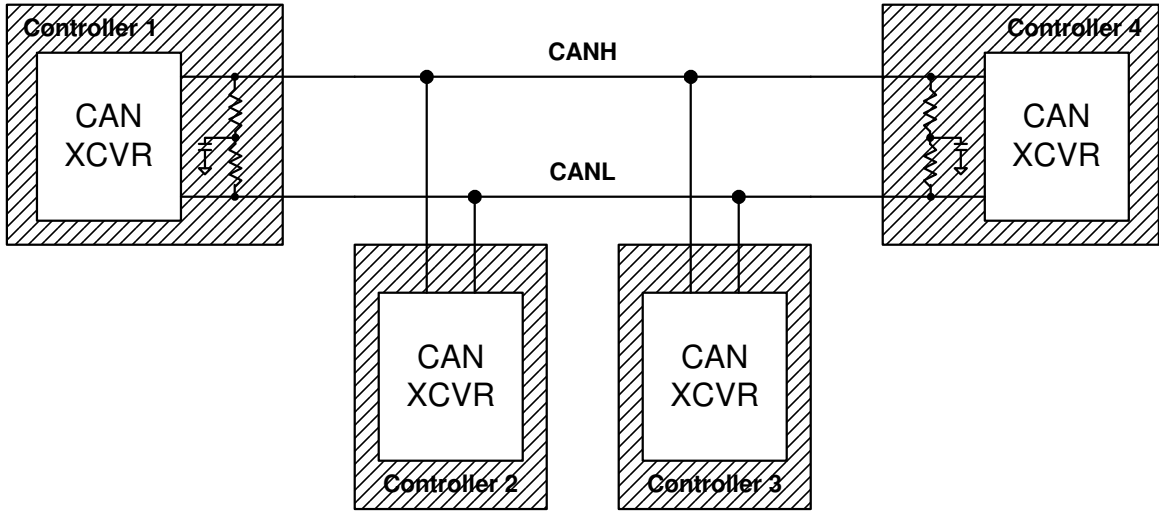


Figure 9 – Multi-Node CAN Network Configuration

Standard 0.1" jumpers should be installed on both J18 and J19 to enable the termination. With no jumpers installed, the lines remain un-terminated. Refer to Figure 10 for the applicable interface circuit.

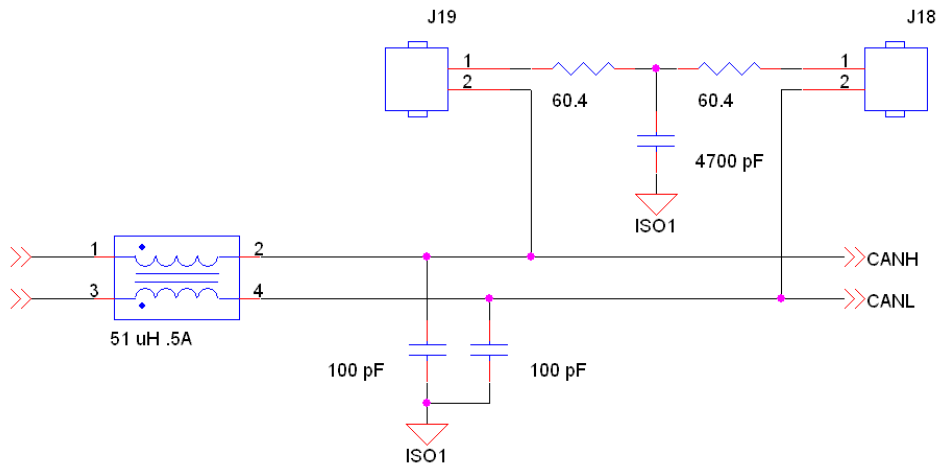


Figure 10 – CAN Interface Circuit

### 3.2 Electrical Interfaces

The approximate location of the connectors, jumper blocks, LEDs, and test hooks are illustrated in Figure 11.

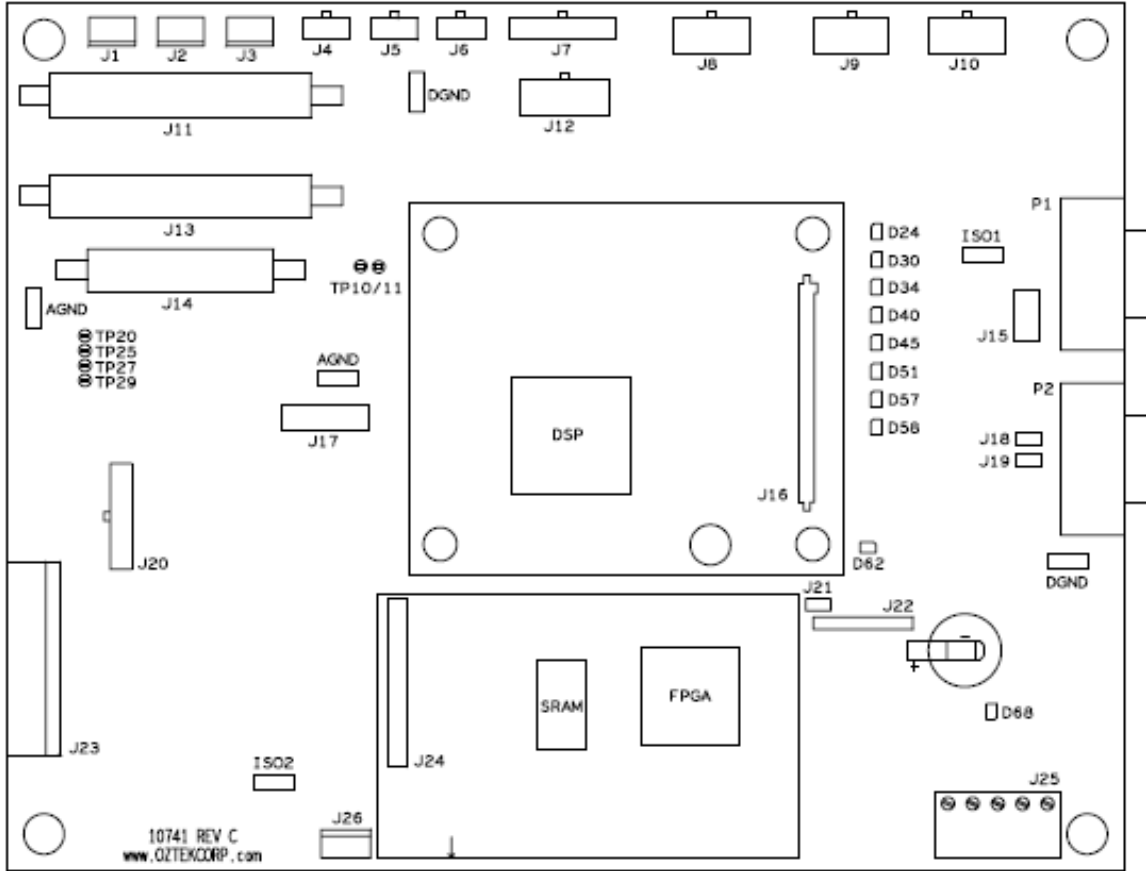


Figure 11 – Approximate Connector, Jumper, LED, and Test Hook Locations

### 3.3 Mechanical Interface

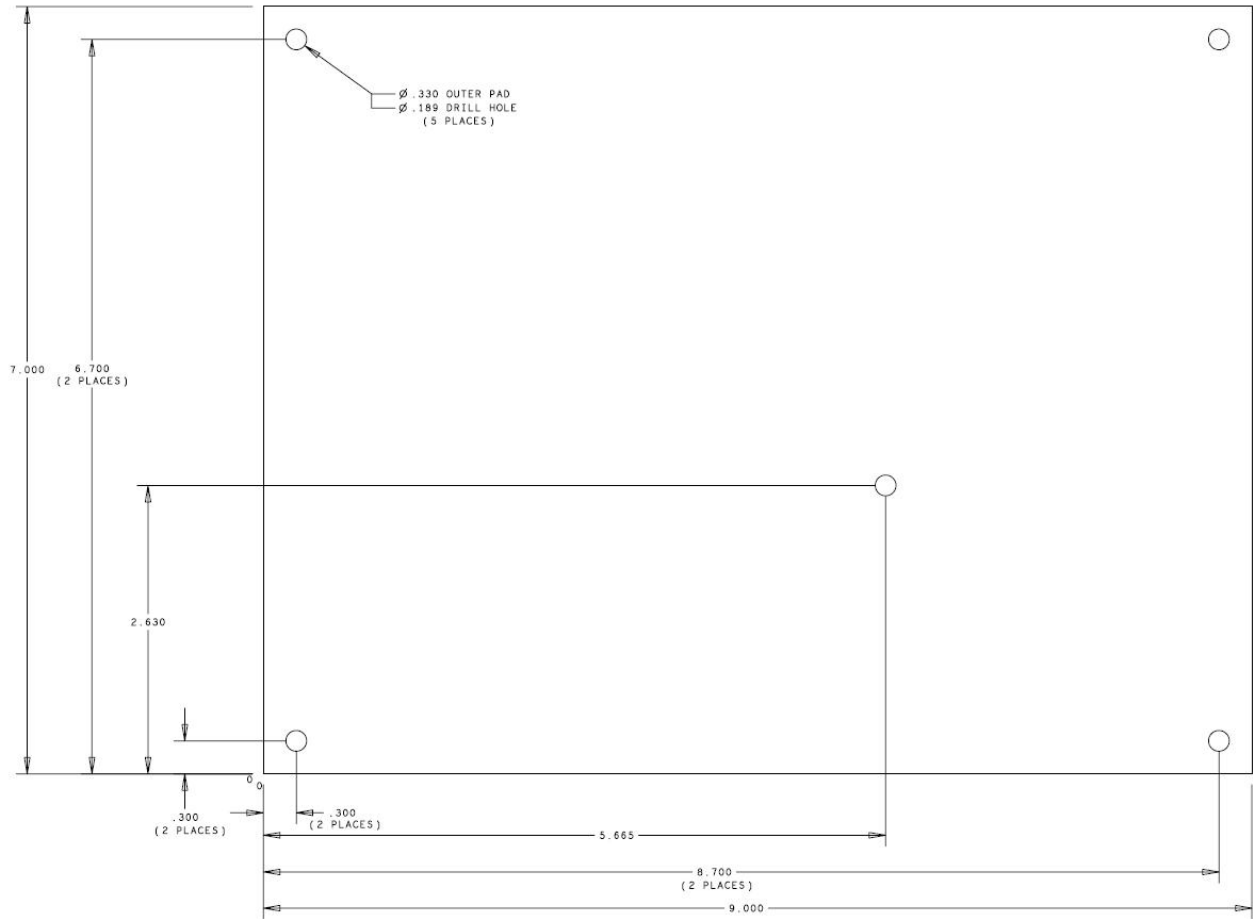


Figure 12 – OZDSP3000 Mechanical Dimensions



## 4. Parameter Register Interface

The motor drive is controlled, monitored, and configured via a parameter register set. This register set can be accessed through the CAN bus serial communication link.

### 4.1 Register Properties

#### 4.1.1 Parameter ID

The Parameter ID (PID) listed in the tables below represents a numerical identifier for each parameter.

#### 4.1.2 Data Types

The actual parameters are stored internally as either 16-bit or 32-bit quantities and are treated as either signed or unsigned entities. The tables below indicate this information using the following abbreviations for the *Data Type*:

- U16 – Parameter is an unsigned 16-bit entity
- U32 – Parameter is an unsigned 32-bit entity
- S16 – Parameter is a signed 16-bit entity
- S32 – Parameter is a signed 32-bit entity

Parameters that are specified as Boolean are stored as 16-bit entities – a value of all zeros indicates FALSE and any non-zero value indicates TRUE. Unless otherwise specified in the parameter description, the parameters are stored and treated as 16-bit unsigned values.

##### 4.1.2.1 Specifying Fixed-Point Parameters

Some parameters listed in the following sections are specified as 32-bit signed numbers with the units specified as Q16 fixed point numbers. Using this data format, the lower 16-bits (LSW) represents the fractional portion of the parameter and the upper 16-bits (MSW) represent the integer portion of the parameter. For example, the number 10.25 would be entered as 0x000A4000, where the MSW = 0x000A (hex) = 10 (decimal) and the LSW = 0x4000 (hex) = 0.25 (0x4000/0xFFFF).

#### 4.1.3 Access Level

The access level for each register is defined as follows:

- **W** (writeable) – the parameter is writable by the user
- **R** (readable) – the parameter is readable by the user
- **P** (password-protected) – the parameter may only be accessed by supplying a password

## 4.2 Volatile Registers

### 4.2.1 Command Registers

Table 9 – Command Register Set

PID	Data Type	Description	Units	Min	Max	Access Level
0x0000	U16	On/Off Control	ENUM	0	1	RW
0x0001	U16	Mode Control	ENUM	0	1	RW
0x0002	S16	Torque Setpoint	0.1%	-32768	32767	RW
0x0003	S16	Speed Setpoint	0.1%	-32768	32767	RW
0x0004	U16	Fault Reset	ENUM	0	1	RW
0x0005	U16	Configuration Password	n/a	0	65535	RW
0x0006	U16	Configuration Reset	ENUM	0	1	RWP
0x0007	U16	Configuration Reload	ENUM	0	1	RW
0x0008	U16	Contactorm Control	ENUM	0	1	RW

#### 4.2.1.1 On/Off Control

This register is used to turn the motor drive ON or OFF as follows:

- 0 – OFF: This turns the drive OFF
- 1 – ON: This turns the drive ON

#### 4.2.1.2 Mode Control

This register is used to indicate the desired operating mode. The default mode following a power-on reset is specified in the configuration memory (see section 4.3.6.9 for details). The mode may not be changed while the drive is ON.

- 0 – Speed Control: Motor speed is controlled in this mode
- 1 – Torque Control: Motor torque is controlled in this mode

#### 4.2.1.3 Torque Setpoint

This register is used to adjust the torque command when the drive is running in direct torque control mode. The default torque command used when turning the drive on in torque control mode is specified in the configuration memory (*Torque Command Default at Turn On*, PID 0x8064). The value written to this register does not change the default torque stored in the configuration memory, rather it merely provides a dynamic and temporary override from the default value.

The torque command is specified as a percentage of the *Torque Command Maximum* (PID 0x8063). This value is specified as a signed 2's complement number in units of 0.1 %. For example, 10 % would be entered as 100 (decimal), or 0x0064 (hexadecimal). Negative 10 % would be represented as 0xFF9C (hexadecimal). The legal range for this command is -1000 to +1000. Values outside of this range will be ignored.

#### 4.2.1.4 Speed Setpoint

This register is used to adjust the speed command when the drive is running in speed control mode. The default speed command used when turning the drive on in speed control mode is specified in the configuration memory (*Speed Command Default at Turn On*, PID 0x8067). The value written to this register does not change the default value in the configuration memory, rather it merely provides a dynamic and temporary override from the default value.

The speed command is specified as a percentage of the *Speed Command Maximum* (PID 0x8066). This value is specified as a signed 2's complement number in units of 0.1 %. For example, 10 % would be entered as 100 (decimal), or 0x0064 (hexadecimal). Negative 10 % would be represented as 0xFF9C (hexadecimal). The legal range for this command is -1000 to +1000. Values outside of this range will be ignored.

#### 4.2.1.5 Fault Reset

This register is used to reset any latched fault conditions and to return the controller to the IDLE state if no further faults exist. Legal values are:

- 0 – NOP: No reset action requested
- 1 – RESET: Attempt to reset all fault conditions

#### 4.2.1.6 Configuration Password

This register is used to supply a password for those configuration operations that are password protected. The password is cleared to zero at the end of the next parameter read or write operation.

#### 4.2.1.7 Configuration Reset

This register causes the system to restore its non-volatile configuration memory to the factory default configuration. Legal values are:

- 0 – NOP: No reset action requested
- 1 – RESET: Request to reset the configuration (register is auto cleared to 0)

Successful execution of this command requires the following conditions be met:

- *The Configuration Password register (PID 0x0005) must have been previously loaded with the correct password*
- *The controller must be in the IDLE state, i.e. the drive must be 'OFF'*



**CAUTION:** Upon execution, all currently stored configuration data will be permanently destroyed and over written with the factory default configuration data.

#### 4.2.1.8 Configuration Reload

This register causes any modifications to the configuration register space to be loaded from the non-volatile configuration memory. Legal values are:

- 0 – NOP: No reload action requested
- 1 – RELOAD: Reload the operational parameters from the configuration space (register is auto cleared to 0)

Successful execution of this command requires the following conditions be met:

- *The controller must be in the IDLE state, i.e. the drive must be 'OFF'*



**CAUTION:** Either execution of this command or a Power-On Reset (POR) is required before changes to the configuration space are used for operation.

#### 4.2.1.9 Contactor Control

This register will attempt to open or close the optional auxiliary contactor as follows:

- 0 – OPEN contactor
- 1 – CLOSE contactor

### 4.2.2 Instrumentation Registers

**Table 10 – Instrumentation Register Set**

PID	Data Type	Description	Units	Access Level
0x4000	U16	Operating State	ENUM	R
0x4001	U16	Operating Mode	ENUM	R
0x4002	U16	Motor Temperature	°C	R
0x4003	S32	Motor Speed	0.1 RPM	R
0x4004	S16	Motor Power	10 W	R
0x4005	S16	Motor Current	0.1 A <sub>rms</sub>	R
0x4006	S16	Motor Current – Inverter A	0.1 A <sub>rms</sub>	R
0x4007	S16	Motor Current – Inverter B	0.1 A <sub>rms</sub>	R
0x4008	U16	Brake State	ENUM	R
0x4009	U16	Brake Temperature	°C	R
0x400A	U16	Inverter A Temperature	°C	R
0x400B	U16	Inverter B Temperature	°C	R
0x400C	U16	DC Link Voltage	0.1 V	R
0x400D	U32	Warning Status	n/a	R
0x400E	U32	Fault Status	n/a	R
0x400F	U16	Register Operation Status	ENUM	R

PID	Data Type	Description	Units	Access Level
0x4010	U16	DSP Software Revision – Major	Integer	R
0x4011	U16	DSP Software Revision – Minor	Integer	R
0x4012	U16	FPGA Revision – Major	Integer	R
0x4013	U16	FPGA Revision – Minor	Integer	R
0x4014	U16	PCB Variant	Integer	R
0x4015	U16	Contactors State	ENUM	R

#### 4.2.2.1 Operating State

The present operating state is reported as shown in the table below. See section 2.4 for details on each of these operating states.

Value	State
0	Initializing
1	Calibrating
2	Idle
3	Magnetizing
4	Aligning
5	Running
6	Fault
<b>7-255</b>	<b>Unknown: Reserved for future use</b>

#### 4.2.2.2 Operating Mode

The present operating mode is reported as follows:

Value	State
0	Speed Control
1	Torque Control
<b>2-255</b>	<b>Reserved for future use</b>

#### 4.2.2.3 Motor Temperature

This register reports the measured motor temperature if a sensor has been enabled in the **Motor Temp Enable** configuration parameter (PID 0x80C1). This register will return zero when a temperature sensor is not enabled.

#### 4.2.2.4 Motor Speed

This register reports the present motor speed.

#### 4.2.2.5 Motor Power

This register reports the estimated motor power using the relationship  $\text{Power} = \text{Torque} * \text{Speed}$ . The torque is estimated based on the specified permanent magnet torque constant or the AC induction motor parameters.

#### 4.2.2.6 Motor Current

This register reports the RMS motor currents. If single inverter operation is enabled this register will report the current from the main inverter A. If dual interleaved inverter operation is enabled, this register will report the sum of the currents from both inverter A and inverter B.

#### 4.2.2.7 Motor Current – Inverter A, B

These registers report the measured RMS motor current as reported by each inverter interface. For single inverter operation, the Inverter A current will match the reported **Motor Current** (PID 0x4005) and the Inverter B current will return zero. For dual interleaved inverter operation, each inverter's current is reported separately in the respective register.

#### 4.2.2.8 Brake State

The present state of the brake controller is reported as follows:

Value	State
0	Disabled
1	Off
2	On
3	Fault
4-255	<i>Reserved for future use</i>

#### 4.2.2.9 Brake Temperature

This register reports the measured brake chopper temperature if a sensor has been enabled in the **Brake Temp Enable** configuration parameter (PID 0x80CA). This register will return zero when a temperature sensor is not enabled.

#### 4.2.2.10 Brake Temperature

This register reports the measured brake chopper temperature if a sensor has been enabled in the **Brake Temp Enable** configuration parameter (PID 0x80CA). This register will return zero when a temperature sensor is not enabled.

#### 4.2.2.11 Inverter A Temperature

This register reports the measured temperature for inverter A if a sensor has been enabled in the **Inverter Temp Enable** configuration parameter (PID 0x80B8). This register will return zero when a temperature sensor is not enabled.

#### 4.2.2.12 Inverter B Temperature

This register reports the measured temperature for optional secondary interleaved inverter B power module if a sensor has been enabled in the **Inverter Temp Enable** configuration parameter (PID 0x80B8). This register will return zero when a temperature sensor is not enabled or if dual interleaved inverter operation is disabled.

#### 4.2.2.13 DC Link Voltage

This register reports the measured voltage on the DC link.

#### 4.2.2.14 Warning Status

Warning bits are active when set to '1', and not present when set to '0'. See section 2.5 for details on each warning. The warning bits are mapped into the register as follows.

Bit	Warning
0	Inverter A High Temperature
1	Inverter B High Temperature
2	Inverter A High Current
3	Inverter B High Current
4	Motor High Temperature
5	Motor High Speed
6	Torque Limited to Maximum
7	Torque Limited by Brake Fault
8	Torque Limited by High Temp
9	Iq Current Foldback
10	Id Current Foldback
11	DC Link High Voltage
12	Reserved
13	Brake Switch High Temperature
14-26	Reserved
27	24V Supply Out of Tolerance
28	15V Supply Out of Tolerance
29	5V Supply Out of Tolerance
30	3.3V Supply Out of Tolerance
31	-15V Supply Out of Tolerance

#### 4.2.2.15 Fault Status

Fault bits are active when set to a '1' and not present when set to a '0'. If a fault occurs, the corresponding bit is set to a '1' and remains set until a "1" is written to the **Fault Reset** command register (PID 0x0004). When a fault occurs, the controller will go to the FAULT state and the drive will stop operating. The controller will stay in the FAULT state until the **Fault Reset** command is received. See section 2.5 for details on each fault condition. The fault bits are mapped as follows:

Bit	Fault
0	Inverter A H/W Error – Phase U
1	Inverter A H/W Error – Phase V
2	Inverter A H/W Error – Phase W
3	Inverter A H/W Error – Over Temp
4	Inverter B H/W Error – Phase U
5	Inverter B H/W Error – Phase V
6	Inverter B H/W Error – Phase W
7	Inverter B H/W Error – Over Temp
8	Inverter A S/W Over Temp
9	Inverter A S/W Over Current
10	Inverter B S/W Over Temp
11	Inverter B S/W Over Current
12	Motor Over Temp

Bit	Fault
13	Motor Over Speed
14	Motor Align Error
15	Reserved
16	Brake Switch H/W Error
17	Brake Switch H/W Over Temp
18	Brake Switch S/W Over Temp
19	Reserved
20	DC Link Over Voltage
21	Configuration Error
22	Calibration Error
23	Communications Timeout
24-31	Reserved

#### 4.2.2.16 Register Operation Status

This register is updated after every parameter read or write operation and indicates whether the operation was completed successfully. The status is reported as follows:

Value	State
0	Operation completed successfully
1	Error – Illegal/unsupported Parameter ID was supplied by the user
2	Error – A write was attempted to a Read-Only Parameter
3	Error – A read was attempted from a Write-Only Parameter
4	Error – User-provided data is not within legal range
5	Error – Configuration Memory Hardware Error
6	Error – Configuration Memory CRC Mismatch
7	Error – Invalid password provided for operation
8	Error – Operation not allowed when the drive is “ON”
9-255	<i>Unknown: Reserved for future use</i>

#### 4.2.2.17 DSP Software Revision – Major/Minor

These registers report the major and minor revisions of the controller software presently loaded in the DSP’s FLASH memory.

#### 4.2.2.18 FPGA Firmware Revision – Major/Minor

These registers report the major and minor revisions of the FPGA on the control board.

#### 4.2.2.19 PCB Variant

This value represents the hardware variant of the DSP control board running the application.

#### 4.2.2.20 Contactor State

This register reports the state of the optional auxiliary contactor as follows:

- 0 – Contactor is OPEN
- 1 – Contactor is CLOSED



### 4.3 Non-Volatile Configuration Registers

Many operating parameters in this application have been made configurable so as to support various motor drive topologies as well as to support other similar products in the future. As such, the software contains provisions for storing these configuration parameters in an external non-volatile EEPROM device.

Once one or more configuration parameters have been updated by writing to the applicable configuration register, the actual operating configuration variables remain unaffected until one of two events occur: either the user cycles power on the control board or the **Configuration Reload** command register (PID 0x0007) is written to. In the second case, the reload is only allowed if the drive is **not** enabled. Attempts to reload the system configuration while the drive is in operation will result in the command being ignored and an error being reported in the **Register Operation Status** register (PID 0x400F).

The **Configuration Reset** command register (PID 0x0006) is used to reset the EEPROM back to the original factory default values. The user should take care when using this command as any custom configuration settings will be lost when the entire contents of the configuration memory is overwritten with the factory defaults. This command is only allowed if the drive is **not** enabled. Attempts to reset the EEPROM data while the drive is in operation will result in the command being ignored and an error being reported in the **Register Operation Status** register.

There are multiple scenarios in which the firmware may automatically program the configuration memory at startup with the factory default values. First, when the drive boots following a reset, it attempts to read a predefined read-only header stored in the EEPROM. If the header does not match the expected value, the memory is considered uninitialized and the firmware will automatically write the default factory configuration values into the memory. This would be the case if a new control board were just being loaded with the firmware and operated for the first time, in which case the configuration memory wouldn't contain valid data.

The second scenario in which the firmware will automatically update the configuration memory occurs when the firmware has been updated on the control board and its associated configuration memory map is not compatible with the previously programmed version of the firmware. The configuration memory contains two factory revision values; a major and a minor revision. At startup the firmware will read the major revision value and compare it against the default major revision for that particular build of the firmware. If the two do not match, the firmware will automatically reset the configuration memory to the factory defaults that are hard coded into the newly installed firmware. Differences in the minor revision will not cause the memory to be reset. Using this factory configuration revision scheme, the minor revision number is expected to change for minor changes that do not change the layout and format of the data in the configuration memory (i.e. a simple change to a default value for a particular parameter or its legal data range). The major revision number is required to change any time

new parameters are added or parameter locations or formats are changed. The user must take care when updating the firmware to understand whether or not the configuration memory will be reset so as not to lose any custom settings previously stored. The present factory revision of the configuration memory is stored in the **Factory Configuration Revision** registers (PIDs 0x8001 – 0x8002).

### 4.3.1 Configuration Control Parameters

**Table 11 – Configuration Control Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8000	U16	EEPROM Header	Integer	0x0159	0	0xFFFF	R
0x8001	U16	Factory Configuration Revision – Major	Integer	1	0	0xFFFF	R
0x8002	U16	Factory Configuration Revision – Minor	Integer	0	0	0xFFFF	R
0x8003	U16	Application Configuration Data Revision	Integer	0	0	0xFFFF	RWP
0x8004	U16	Hardware Configuration	Integer	0	0	0xFFFF	R
0x8005	U16	User Configuration Revision	Integer	0	0	0xFFFF	RW
0x8006	U16	Configuration Password	Integer	0x1111	0	0xFFFF	RWP

#### 4.3.1.1 EEPROM Header

This is a read-only header word that is used to indicate whether or not the configuration memory contains valid configuration data. This header word is a fixed constant and is not expected to change for any revision of the drive firmware. At startup the firmware attempts to read this value and if it does not match the expected value the firmware assumes the configuration memory does not contain valid configuration data (i.e. not previously programmed). If this is the case, the firmware will automatically reset the contents of the configuration memory to the factory defaults.

#### 4.3.1.2 Factory Configuration Revision – Major

This is a read-only value that represents the major revision of the factory configuration stored in the configuration memory. Major revision changes to the default factory configuration are those that are not compatible with previous configurations, such as when new parameters are added to the memory that are required for proper drive operation, or if existing parameters change locations or formats. At startup the firmware will read this value from the memory and compare it against the factory default for the present build of the firmware. If the two values do not match, the firmware will automatically reset the memory to the factory default values built into that version of the firmware. Updates to the major factory revision value are expected to increment the previous value by +1.

#### 4.3.1.3 Factory Configuration Revision – Minor

This is a read-only value that represents the minor revision of the factory configuration stored in the configuration memory. Minor revisions are those that do not require reloading the memory to the factory default values. This could be a result of a minor value change to the default value for a particular parameter or the addition of a new parameter that is not needed

for proper drive operation. Updates to the minor factory revision value are expected to increment the previous value by +1.

#### **4.3.1.4 Application Configuration Data Revision**

This value is used to represent the revision of the application-specific configuration data as programmed by the factory. This field is meant to store the revision of any custom configuration settings programmed at the factory for a specific end-user application. The factory default (prior to customization) for this field is zero, indicating that no custom settings have been made to the configuration memory. Application-specific updates to the configuration data are expected to increment this parameter by +1. This parameter is password protected to prevent accidental modification by the end-user.

#### **4.3.1.5 Hardware Configuration**

This read-only register is provided for future use only. It is presently not used by the motor drive application and will return a value of zero when read.

#### **4.3.1.6 User Configuration Revision**

This is a generic parameter that is provided to allow the user or a higher-level controller to maintain revision information for custom settings to the configuration memory. The firmware does not use this value. The protocol for numbering and maintaining custom configurations is left up to the user.

#### **4.3.1.7 Configuration Password**

This parameter defines the configuration password stored in the configuration memory. To access any parameter that is marked as “password-protected”, the user must provide a password that matches the value stored in this parameter. See the Parameter Read/Write messages described in FS-0066 (OzCan Motor Controller Device Profile) for further details on providing this password value.

Note that this parameter is password protected as well. The firmware has its own unpublished “master password” that can be used to override the password stored in this register. In the event that the password is changed from the factory default listed and then subsequently lost, contact Oztek for the “master password” or for other alternatives to reset the value in the configuration memory.

### **4.3.2 CAN Interface Parameters**

The following set of parameters are provided to allow for customizing the CAN interface for the end-user’s application.

**Table 12 – CAN Interface Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8010	U16	CAN Group ID	Integer	3	1	15	RW
0x8011	U16	CAN Module ID	Integer	1	1	31	RW
0x8012	U16	CAN Baud Rate	ENUM	250kbps	50kbps	1Mbps	RW
0x8013	U16	CAN Status Destination Group ID	Integer	1	0	15	RW
0x8014	U16	CAN Status Destination Module ID	Integer	1	0	31	RW
0x8015	U16	CAN Timeout	1 ms	0	0	65535	RW
0x8016	U16	CAN Automatic Alarm Transmit Enable	boolean	TRUE	FALSE	TRUE	RW
0x8017	U16	CAN Update Rate – Motor Status	1 ms	100	0	65535	RW
0x8018	U16	CAN Update Rate – Brake Status	1 ms	100	0	65535	RW
0x8019	U16	CAN Update Rate – System Status	1 ms	100	0	65535	RW
0x801A	U16	CAN Update Rate – Alarm Status	1 ms	100	0	65535	RW

**4.3.2.1 CAN Group ID**

This parameter specifies the Group ID used by the firmware. For more information see FS-0047.

**4.3.2.2 CAN Module ID**

This parameter specifies the Module ID used by the firmware. For more information see FS-0047.

**4.3.2.3 CAN Baud Rate**

This parameter is used to configure the serial baud rate for the CAN interface. The legal values are as follows:

- 0 = 1 Mbps
- 1 = 500 kbps
- 2 = 250 kbps
- 3 = 125 kbps
- 4 = 100 kbps
- 5 = 50 kbps

**4.3.2.4 CAN Status Destination Group ID**

This parameter specifies the Destination Group ID that this firmware will use when sending the Status messages. Note that the Illegal CAN Message and Configuration Response messages use the Group/Module information from the sending device as the Destination Group/Module when responding, so this parameter is not used for those messages. For more information see FS-0047.

**4.3.2.5 CAN Status Destination Module ID**

This parameter specifies the Destination Module ID that this firmware will use when sending the Status messages. Note that the Illegal CAN Message and Configuration Response messages use the Group/Module information from the sending device as the Destination Group/Module

when responding, so this parameter is not used for those messages. For more information see FS-0047.

#### 4.3.2.6 CAN Timeout

This parameter specifies the timeout period to use when checking for CAN communications errors. This value specifies the timeout period in terms of milliseconds. Setting this parameter to zero disables checking for CAN communications timeouts. When CAN timeout checking is enabled (parameter is set to a legal non-zero value), a timeout counter is used to time the period of inactivity on the CAN bus. This timer is reset upon the receipt of any of the valid receive messages supported by this application.

#### 4.3.2.7 CAN Automatic Alarm Transmit Enable

This is a Boolean parameter that is used to enable automatic transmission of the Alarm Status message upon a change of value of any warning or fault bit. Note that when enabled (parameter is set to *true*), the checks for whether or not an automatic transmission should be sent occur on 1ms boundaries, so there may be up to 1ms of latency between when the offending event occurs and when the Alarm message is sent. When disabled (parameter is set to *false*), a change in any warning or fault bit does not cause an automatic transmission of the Alarm Status message.

#### 4.3.2.8 CAN Update Rate – Motor Status Message

This parameter specifies the rate at which the Motor Status CAN message will be automatically transmitted by the firmware. This value specifies the period between message transmissions in terms of milliseconds. Setting this parameter to zero disables automatic/periodic transmission of this message. All other values (1ms – 65,535ms) are valid.

#### 4.3.2.9 CAN Update Rate – Brake Status Message

This parameter specifies the rate at which the Brake Status CAN message will be automatically transmitted by the firmware. This value specifies the period between message transmissions in terms of milliseconds. Setting this parameter to zero disables automatic/periodic transmission of this message. All other values (1ms – 65,535ms) are valid.

#### 4.3.2.10 CAN Update Rate – System Status Message

This parameter specifies the rate at which the System Status CAN message will be automatically transmitted by the firmware. This value specifies the period between message transmissions in terms of milliseconds. Setting this parameter to zero disables automatic/periodic transmission of this message. All other values (1ms – 65,535ms) are valid.

#### 4.3.2.11 CAN Update Rate - Alarm Status Message

This parameter specifies the rate at which Alarm Status CAN message will be automatically transmitted by the firmware. This value specifies the period between message transmissions in terms of milliseconds. Setting this parameter to zero disables automatic/periodic transmission of this message. All other values (1ms – 65,535ms) are valid.

### 4.3.3 Instrumentation and Measurement Scaling Parameters

**Table 13 – Instrumentation and Measurement Scaling Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8020	U16	Instrumentation Interrupt Rate	1 Hz	2000	500	10000	RW
0x8021	U16	DC Link Voltage Measurement – Full Scale	0.1 V	10000	100	15000	RW
0x8022	U16	Motor Current Measurement – Full Scale	0.1 A	5000	100	65535	RW
0x8023	U16	Motor Current Measurement Polarity Select	ENUM	0	0	1	RW
0x8024	U16	Low Pass Filter Cutoff - Motor Current	1 Hz	100	1	1000	RW
0x8025	U16	Low Pass Filter Cutoff - DC Link	1 Hz	100	1	1000	RW
0x8026	U16	Low Pass Filter Cutoff - Temperatures	1 Hz	5	1	1000	RW

#### 4.3.3.1 Instrumentation Interrupt Rate

This parameter specifies the rate at which the instrumentation interrupt is executed in the controller. This is the rate that the various measurements are read and filtered by the drive software.

#### 4.3.3.2 DC Link Voltage Measurement – Full Scale

This parameter defines the full scale value of the voltage sensor used to measure the DC link voltage. This full scale value is relative to the signal at the DSP's ADC input that would cause a full scale ADC reading (i.e. ADC reading of 0xFFFF).

#### 4.3.3.3 Motor Current Measurement – Full Scale

This parameter defines the full scale value of the motor phase current sense measurements. This full scale value is relative to the signal at the DSP's ADC input that would cause a full scale ADC reading (i.e. ADC reading of 0xFFFF).

#### 4.3.3.4 Motor Current Measurement Polarity Select

This parameter defines the polarity of the measured phase currents. A value of zero (0) indicates that current flowing from the power switches to the motor is positive. A value of one (1) indicates that current flowing from the power switches to the motor is negative.

#### 4.3.3.5 Low Pass Filter Cutoff – Motor Current

This parameter defines the cutoff frequency for the digital low pass filter used to calculate the RMS motor current.

#### 4.3.3.6 Low Pass Filter Cutoff – DC Link

This parameter defines the cutoff frequency for the digital low pass filter used to calculate the DC link voltage used for instrumentation purposes.

#### 4.3.3.7 Low Pass Filter Cutoff – Temperatures

This parameter defines the cutoff frequency for the digital low pass filter used to calculate the various temperatures reported by the controller.

#### 4.3.4 Fault and Warning Parameters

Table 14 – Fault and Warning Parameter Summary

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8030	U16	DC Link Over Voltage Fault Threshold	0.1 V	7600	0	12000	RW
0x8031	U16	DC Link Over Voltage Warning Threshold	0.1 V	7550	0	12000	RW
0x8032	U16	DC Link Over Voltage Recover Threshold	0.1 V	7500	0	12000	RW
0x8033	U16	Inverter Temperature Fault Threshold	°C	115	0	150	RW
0x8034	U16	Inverter Temperature Warning Threshold	°C	110	0	150	RW
0x8035	U16	Inverter Temperature Recover Threshold	°C	105	0	150	RW
0x8036	U16	Brake Temperature Fault Threshold	°C	115	0	150	RW
0x8037	U16	Brake Temperature Warning Threshold	°C	110	0	150	RW
0x8038	U16	Brake Temperature Recover Threshold	°C	105	0	150	RW
0x8039	U16	Motor Temperature Fault Threshold	°C	115	0	150	RW
0x803A	U16	Motor Temperature Warning Threshold	°C	105	0	150	RW
0x803B	U16	Motor Temperature Recover Threshold	°C	100	0	150	RW
0x803C	U16	Motor Over Current Fault Threshold	0.1 A <sub>rms</sub>	3250	0	65535	RW
0x803D	U16	Motor Over Current Warning Threshold	0.1 A <sub>rms</sub>	2900	0	65535	RW
0x803E	U16	Motor Over Current Recover Threshold	0.1 A <sub>rms</sub>	2800	0	65535	RW
0x803F	U16	Motor Over Speed Fault Threshold	RPM	6250	0	65535	RW
0x8040	U16	Motor Over Speed Warning Threshold	RPM	6000	0	65535	RW
0x8041	U16	Motor Over Speed Warning Recover Threshold	RPM	5750	0	65535	RW
0x8042	U16	Inverter Error Pin Active High	boolean	TRUE	FALSE	TRUE	RW
0x8043	U16	Inverter Over Temp Pin Active High	boolean	TRUE	FALSE	TRUE	RW
0x8044	U16	Brake Hardware Error Pin Active High	boolean	TRUE	FALSE	TRUE	RW
0x8045	U16	Brake Hardware Over Temp Pin Active High	boolean	TRUE	FALSE	TRUE	RW

##### 4.3.4.1 DC Link Over Voltage Fault Threshold

This parameter defines the DC link over voltage fault threshold. If the DC link voltage rises above this value the application will automatically transition to the FAULT state and operation of the inverter will be disabled (and forced OFF).

##### 4.3.4.2 DC Link Over Voltage Warning Threshold

This parameter defines the DC link voltage threshold above which the firmware will report a high voltage warning. Once above this warning threshold, the voltage must drop below the corresponding recover threshold before the firmware will clear the high voltage warning.

##### 4.3.4.3 DC Link Over Voltage Recover Threshold

See warning threshold description above.

##### 4.3.4.4 Inverter Temperature Fault Threshold

This parameter defines the inverter temperature fault threshold. If the temperature rises above this value the controller will automatically transition to the FAULT state and drive operation will be disabled (and forced OFF).

#### 4.3.4.5 Inverter Temperature Warning Threshold

This parameter defines the inverter temperature threshold above which the firmware will report a high temperature warning. Once above this warning threshold, the temperature must fall below the corresponding recover threshold before the firmware will clear the high temperature warning.

#### 4.3.4.6 Inverter Temperature Recover Threshold

See warning threshold description above.

#### 4.3.4.7 Brake Temperature Fault Threshold

This parameter defines the brake switch temperature fault threshold. If the **Brake Temperature Sensor Enable** configuration parameter (PID 0x80CA) is set to TRUE and the temperature rises above this value the controller will automatically transition to the FAULT state and drive operation will be disabled (and forced OFF).

#### 4.3.4.8 Brake Temperature Warning Threshold

This parameter defines the brake switch temperature threshold above which the firmware will report a high temperature warning (if the **Brake Temperature Sensor Enable** configuration parameter is set to TRUE). Once above this warning threshold, the temperature must fall below the corresponding recover threshold before the firmware will clear the high temperature warning.

#### 4.3.4.9 Brake Temperature Recover Threshold

See warning threshold description above.

#### 4.3.4.10 Motor Temperature Fault Threshold

This parameter defines the motor temperature fault threshold. If the **Motor Temperature Sensor Enable** configuration parameter (PID 0x80C1) is set to TRUE and the temperature rises above this value the controller will automatically transition to the FAULT state and drive operation will be disabled (and forced OFF).

#### 4.3.4.11 Motor Temperature Warning Threshold

This parameter defines the motor temperature threshold above which the firmware will report a high temperature warning (if the **Motor Temperature Sensor Enable** configuration parameter is set to TRUE). Once above this warning threshold, the temperature must fall below the corresponding recover threshold before the firmware will clear the high temperature warning.

#### 4.3.4.12 Motor Temperature Recover Threshold

See warning threshold description above.

#### 4.3.4.13 Motor Over Current Fault Threshold

This parameter defines the motor over current fault threshold. If the current rises above this value the controller will automatically transition to the FAULT state and drive operation will be disabled (and forced OFF).



For dual interleaved inverter operation, this value is divided in half and used by each inverter's current controller. If either inverter exceeds 50% of the value stored in this register the controller will transition to the FAULT state.

The over current fault condition is latched and reported separately for each inverter in the **Fault Status** instrumentation register (PID 0x400E).

#### 4.3.4.14 Motor Over Current Warning Threshold

This parameter defines the motor current threshold above which the firmware will report a high current warning. Once above this warning threshold, the current must fall below the corresponding recover threshold before the firmware will clear the high current warning.

For dual interleaved inverter operation, this value is divided in half and used by each inverter's current controller. If either inverter exceeds 50% of the value stored in this register the controller will indicate a warning condition for the corresponding inverter. The over current warning condition is reported separately for each inverter in the **Warning Status** instrumentation register (PID 0x400D).

#### 4.3.4.15 Motor Over Current Recover Threshold

See warning threshold description above.

#### 4.3.4.16 Motor Over Speed Fault Threshold

This parameter defines the motor speed fault threshold. If the motor speed rises above this value the controller will automatically transition to the FAULT state and drive operation will be disabled (and forced OFF).

#### 4.3.4.17 Motor Over Speed Warning Threshold

This parameter defines the motor speed threshold above which the firmware will report a high speed warning. Once above this warning threshold, the speed must fall below the corresponding recover threshold before the firmware will clear the high speed warning.

#### 4.3.4.18 Motor Over Speed Recover Threshold

See warning threshold description above.

#### 4.3.4.19 Inverter Error Pin Active High

This parameter determines the polarity of the hardware error input pins from the power module interface ("INV\_ERR\_U/V/W" listed in section 3.1.1). When this parameter is set to TRUE, the input pins are treated as active high, meaning a high input is considered a fault condition.

#### 4.3.4.20 Inverter Over Temp Pin Active High

This parameter determines the polarity of the hardware over temperature input pin from the power module interface ("INV\_OVR\_TEMP" listed in section 3.1.1). When this parameter is set to TRUE, the input pin is treated as active high, meaning a high input is considered a fault condition.

#### 4.3.4.21 Brake Hardware Error Pin Active High

This parameter determines the polarity of the hardware error input pin from the half bridge power module interface (“HB\_ERR” listed in section 3.1.6). When this parameter is set to TRUE, the input pins are treated as active high, meaning a high input is considered a fault condition.

#### 4.3.4.22 Brake Hardware Over Temp Pin Active High

This parameter determines the polarity of the hardware over temperature input pin from the half bridge power module interface (“HB\_OVR\_TEMP” listed in section 3.1.6). When this parameter is set to TRUE, the input pin is treated as active high, meaning a high input is considered a fault condition.

### 4.3.5 Inverter Control Parameters

**Table 15 – Inverter Control Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8050	U16	Pulse Width Modulation Frequency	1 Hz	5000	1000	25000	RW
0x8051	U16	Pulse Width Modulation Deadband Time	1 ns	2100	0	65535	RW
0x8052	S32	Pulse Width Modulation Max Duty Cycle	Q16	0.95	0	0.9999	RW
0x8053	S32	Pulse Width Modulation Min Duty Cycle	Q16	-0.95	-1	0	RW
0x8054	U16	Inverter Power Stage Dead Time	1 ns	0	0	65535	RW

#### 4.3.5.1 Pulse Width Modulation Frequency

This parameter defines the PWM switching rate as well as the frequency at which the control ISR will be executed.

#### 4.3.5.2 Pulse Width Modulation Deadband Time

This parameter defines the amount of dead band time to use between switching the complementary top and bottom switches in the half bridges (i.e. the time that both PWM outputs are OFF).

#### 4.3.5.3 Pulse Width Modulation Max/Min Duty Cycle

These parameters specify the minimum and maximum duty cycles allowed on the PWM outputs to the power switches. The controller will clamp the PWM pulses to these values, preventing any pulses narrower than the minimum specified value or greater than the maximum specified value. These parameters can be used to guarantee minimum pulse widths if required by the power switches. The dead band time (either enforced by the power module hardware or by the controller software parameter) should be taken into account when setting these parameters.

The duty cycle is encoded such that -1.0 is equivalent to 0% duty cycle, 0.0 is equivalent to 50% duty cycle, and +1.0 is equivalent to 100% duty cycle. For example, setting the max duty cycle parameter to 0.95 would be equivalent to clamping the duty cycle to be no greater than 97.5%.

#### 4.3.5.4 Inverter Power Stage Dead Time

This parameter is used to specify the interlock dead time enforced by the inverter's power stage drivers. This is the time between turning off one switch and turning on the complementary switch in the same half bridge. In some systems, particularly those with high PWM switching rates, this hardware-enforced dead time may add a small but noticeable distortion to the inverter's current waveforms. The controller provides a dead time compensator that adds a correction to the PWM output signals to remove these distortions. For proper operation, the value programmed in this register must match the actual dead time enforced by the power stage driver hardware. For systems where the dead time distortion is not an issue or to otherwise disable the compensator in the controller, simply set this parameter to zero.

#### 4.3.6 Drive Configuration Parameters

**Table 16 – Drive Configuration Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8060	U16	Motor Drive Type	ENUM	1	0	3	RW
0x8061	U16	Motor Speed Feedback Device Select	ENUM	1	0	2	RW
0x8062	U16	Torque Command Slew Rate	0.1 %/s	1000	1	65535	RW
0x8063	U16	Torque Command Maximum	0.1 Nm	10000	0	65535	RW
0x8064	S16	Torque Command Default at Turn On	0.1 %	0	-1000	1000	RW
0x8065	U16	Speed Command Slew Rate	RPM/s	100	1	65535	RW
0x8066	U16	Speed Command Maximum	RPM	3000	0	65535	RW
0x8067	S16	Speed Command Default at Turn On	0.1 %	0	-1000	1000	RW
0x8068	U16	Torque/Speed Mode Default at Power On	ENUM	1	0	1	RW

##### 4.3.6.1 Motor Drive Type

This parameter is used to indicate the type of motor drive being implemented by the controller as follows:

- 0 = None (drive disabled)
- 1 = Closed-loop Field Oriented AC Induction Motor Controller
- 2 = Closed-loop Field Oriented Permanent Magnet Motor Controller
- 3 = Open-loop AC Induction Volts/Hertz Controller

##### 4.3.6.2 Motor Speed Feedback Device Select

This parameter is used to configure the type of speed feedback device being used by the controller as follows:

- 0 = None
- 1 = Digital Quadrature Encoder
- 2 = Resolver

#### 4.3.6.3 Torque Command Slew Rate

This parameter defines the slew rate to use when user changes the commanded torque value. This parameter is only used when the drive is being operated in direct torque control mode. This parameter is specified as a percent of the **Torque Command Maximum** (see next parameter) per second. For example, if this parameter is set to 1000, this would indicate that the torque command slew rate will be zero to 100% of the specified maximum torque command in one second.

#### 4.3.6.4 Torque Command Maximum

This parameter defines the maximum torque command used by the drive. When operating in direct torque control mode, the user commands a torque value as a percentage (-100% to +100%) of this value.

#### 4.3.6.5 Torque Command Default at Turn On

This parameter defines the default torque command to use when turning on the drive in direct torque control mode. The drive will initially turn on with a zero torque command and will then slew to this value at a rate defined by the **Torque Command Slew Rate** parameter (see above).

#### 4.3.6.6 Speed Command Slew Rate

This parameter defines the slew rate to use when user changes the commanded speed value. This parameter is only used when the drive is being operated in speed control mode.

#### 4.3.6.7 Speed Command Maximum

This parameter defines the maximum speed command used by the drive. When operating in speed control mode, the user commands a speed value as a percentage (-100% to +100%) of this value.

#### 4.3.6.8 Speed Command Default at Turn On

This parameter defines the default speed command to use when turning on the drive in speed control mode. When the drive initially turns on, it will sense the present motor speed. It will then slew the speed command from the sensed speed to the value specified in this parameter at a rate defined by the **Speed Command Slew Rate** parameter (see above).

#### 4.3.6.9 Torque/Speed Mode Default at Power On

This parameter defines the default control mode (0 = speed control, 1 = torque control) following a power-on reset of the controller, assuming the user has not sent a Mode command via the corresponding CAN message. Once the mode has been changed by the user (via the corresponding CAN message), the drive will always turn on in the mode that was last specified.

### 4.3.7 General Motor Parameters

**Table 17 – General Motor Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8070	U16	Motor Pole Pair Count	Integer	3	1	65535	RW
0x8071	U16	Motor Rated Voltage	0.1 Vrms	4140	1	65535	RW
0x8072	U16	Motor Rated Frequency	0.1 Hz	1016	1	65535	RW
0x8073	U16	Motor Maximum Current	0.1 Arms	2900	1	65535	RW
0x8074	U16	Motor Maximum Frequency	0.1 Hz	3500	1	65535	RW

#### 4.3.7.1 Motor Pole Pair Count

This parameter is used to indicate the number of pole pairs for the motor being controlled.

#### 4.3.7.2 Motor Rated Voltage

This parameter is used to indicate the rated voltage for the motor being controlled.

#### 4.3.7.3 Motor Rated Frequency

This parameter is used to indicate the rated frequency for the motor being controlled.

#### 4.3.7.4 Motor Maximum Current

This parameter is used to indicate the maximum allowable current for the motor being controlled. When operating in closed-loop speed or torque mode, the controller will attempt to clamp the operating current to this value. Torque or speed commands that would otherwise result in a current above this value will result in the motor current being clamped to this value. In this case, the resulting torque will be reduced or the speed will fall out of regulation depending on the direction of the load.

#### 4.3.7.5 Motor Maximum Frequency

This parameter is used to indicate the maximum allowable electrical frequency for the motor being controlled. This value is used by the firmware as the full scale frequency for normalization purposes. This parameter should be set above the maximum expected operating frequency for the motor in order to provide some margin for headroom when operating at high speeds. As a rule of thumb, a reasonable margin for headroom would be around 10% to 20%. For example, if the expected maximum operating frequency of the motor was 300Hz, then a reasonable value for this parameter would be 350Hz.

### 4.3.8 AC Induction Motor Parameters

The parameters described in this section are only used if the controller is configured to operate as a Closed-loop Field Oriented AC Induction Motor Controller as specified in the **Motor Drive Type** parameter (PID 0x8060). Otherwise, these parameters are not used by the controller.

**Table 18 – AC Induction Motor Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8080	U32	AC Motor Mutual Inductance	$\mu\text{H}$	4841	0	1E+08	RW
0x8081	U32	AC Motor Stator Leakage Inductance	$\mu\text{H}$	175	0	1E+08	RW
0x8082	U32	AC Motor Rotor Leakage Inductance	$\mu\text{H}$	175	0	1E+08	RW
0x8083	U32	AC Motor Rotor Resistance	$\mu\Omega$	19400	0	1E+08	RW
0x8084	U16	AC Motor Magnetizing Current	0.1 Arms	620	1	65535	RW
0x8085	U16	AC Motor Initial Magnetizing Current	0.1 Arms	1360	1	65535	RW
0x8086	U16	AC Motor Minimum Magnetizing Current	0.1 Arms	100	1	65535	RW
0x8087	U16	AC Motor Magnetizing Current Slew Rate	Arms/s	1400	1	65535	RW
0x8088	U16	AC Motor Field Weakening Enable	boolean	TRUE	FALSE	TRUE	RW
0x8089	U16	AC Motor Field Weakening Minimum Speed	RPM	100	1	65535	RW
0x808A	U16	AC Motor Temperature Coefficient	0.001%/°C	400	1	65535	RW

**4.3.8.1 AC Motor Mutual Inductance**

This parameter is used to specify the motor's mutual inductance.

**4.3.8.2 AC Motor Stator Leakage Inductance**

This parameter is used to specify the motor's stator leakage inductance.

**4.3.8.3 AC Motor Rotor Leakage Inductance**

This parameter is used to specify the motor's rotor leakage inductance.

**4.3.8.4 AC Motor Rotor Resistance**

This parameter is used to specify the motor's rotor resistance.

**4.3.8.5 AC Motor Magnetizing Current**

This parameter is used to specify the nominal magnetizing current for the motor. When field weakening is not in effect, the magnetizing current to the motor will be controlled to this value.

**4.3.8.6 AC Motor Initial Magnetizing Current**

This parameter is used to specify the initial magnetizing current to use when first turning the drive on. For motors with long rotor time constants, it may be desirable to set this parameter to a value higher than the nominal magnetizing current parameter as this will shorten the time it takes to induce the magnetic field in the rotor. Once the drive has estimated that the rotor field has reached the desired level, the magnetizing current will then be reduced to the nominal magnetizing current specified by the parameter above. This parameter should be set to the nominal magnetizing current for applications that do not require a faster initial magnetization period.

**4.3.8.7 AC Motor Minimum Magnetizing Current**

This parameter is used to specify the minimum magnetizing current ( $I_d$ ) to use when calculating the necessary torque producing current ( $I_q$ ) based on the commanded torque value. In general

this parameter is only pertinent when operating at very high motor speeds that require significant field weakening (i.e. at a very low  $I_d$  current). This parameter is used mainly to simplify the firmware and to avoid divide-by-zero issues with the  $I_q$  calculation. This parameter should be set at or below the anticipated minimum field-weakened magnetizing current for the drive.

#### 4.3.8.8 AC Motor Magnetizing Current Slew Rate

This parameter is used to set the magnetizing current slew rate when turning on the drive. The drive initially turns on with a regulated current of zero amps. It will then ramp up the magnetizing current at a rate defined by this parameter.

#### 4.3.8.9 AC Motor Field Weakening Enable

This parameter is used to enable the field weakening controller. If this parameter is set to FALSE, the field weakening feature will be disabled and the magnetizing current will be held at the nominal specified magnetizing current. If this parameter is set to TRUE, the field weakening feature will be enabled, allowing the magnetizing current to be reduced at high motor speeds. Note that the maximum available motor torque and/or motor speed may be limited if field weakening is disabled (depending on the motor ratings and the DC link voltage).

#### 4.3.8.10 AC Motor Field Weakening Minimum Speed

This parameter is used to define the minimum speed at which field weakening will be calculated. If field weakening is disabled, this parameter is not used. This parameter is provided mainly to simplify the firmware and to avoid divide-by-zero issues with the field weakening calculations. This parameter should be set to a value below the minimum speed at which field weakening is expected to occur.

#### 4.3.8.11 AC Motor Temperature Coefficient

This parameter is used to specify the thermal coefficient of the motor's rotor material. The controller's slip estimator uses this parameter along with the measured motor temperature to estimate the change in rotor resistance as the temperature increases above the nominal 20°C. If a motor temperature sensor is not enabled, then this parameter will not be used.

This parameter is specified in units of 0.001 % / °C. For example, the thermal coefficient of copper is approximately +0.4 %/°C, meaning that if the temperature increases by 1°C, the resistance will increase by 0.4%. This value would be stored as 400. The rotor resistance adjustment can be disabled by setting this parameter to zero.

### 4.3.9 Permanent Magnet Motor Parameters

The parameters described in this section are only used if the controller is configured to operate as a Closed-loop Field Oriented Permanent Magnet Motor Controller as specified in the **Motor Drive Type** parameter (PID 0x8060). Otherwise, these parameters are not used by the controller.

Table 19 – Permanent Magnet Motor Parameter Summary

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8090	U16	PM Torque Constant	0.001 Nm / Arms	1000	1	65535	RW
0x8091	U16	PM Align At Startup Enable	boolean	FALSE	FALSE	TRUE	RW
0x8092	U16	PM Align Current	0.1 Arms	100	0	65535	RW
0x8093	U16	PM Align Time	10 ms	1000	1	65535	RW
0x8094	U16	PM Align Maximum Speed Threshold	0.1 RPM	50	1	65535	RW
0x8095	U16	PM Align Angle Offset	0.01 Degrees	0	0	65535	RW

#### 4.3.9.1 PM Torque Constant

This parameter is used to specify the torque constant of the permanent magnet motor. The firmware uses this value to convert the user's commanded torque value to the corresponding current command for the motor.

For applications where the torque constant is unknown, or if it is desirable to simply command current instead of torque, this can be achieved by doing the following: set this parameter to a value of 1000 (i.e. 1 Nm = 1 Arms), and set the **Torque Command Maximum** parameter (PID 0x8063) to the desired maximum current command. By doing this, the firmware will take the user's normalized torque command value (-100% to +100%) and convert this to a percentage of the maximum desired torque current.

#### 4.3.9.2 PM Align At Startup Enable

This parameter is used to enable automatic alignment of a permanent magnet motor at startup. When set to true, the first time a motor is started after power-up the drive will energize the windings with a DC current and wait for the shaft of the motor to align to the generated field. This process is only done once following a power-up reset. If set to false, no alignment is performed and it is assumed that the speed and position device resets to a known alignment point. When this feature is enabled and the alignment is attempted following a power-on reset of the controller, the shaft of the motor **MUST** be allowed to rotate freely for proper position alignment to occur.

#### 4.3.9.3 PM Align Current

When automatic alignment is enabled, this parameter is used to specify the amount of current to regulate through the windings of the motor during the alignment process.

#### 4.3.9.4 PM Align Time

This parameter is used to determine how long to apply the align current for and wait for the rotor to settle into the align position. Once this time has expired, the current is then set to zero amps and the drive is then ready for use.



#### 4.3.9.5 PM Align Maximum Speed Threshold

This parameter indicates the maximum speed at which the drive will attempt to align the motor. If the motor is presently spinning faster than this speed, the alignment process will be aborted and a system “Alignment Error Fault” will be generated.

#### 4.3.9.6 PM Align Angle Offset

Once alignment is complete, the present “aligned” motor position is read by the firmware and then adjusted by 90 degrees to create the proper torque alignment for the motor.

This parameter is used to allow the user to enter an additional angle offset (in addition to the 90 degrees automatically applied by the drive). This may be useful for calibration purposes or for non-standard PM motor arrangements that requires additional alignment offsets – for example, if multiple motors are connected to the same rotor shaft, only one is used during the alignment process, and the phase of the other motors is intentionally offset from the one used for alignment.

### 4.3.10 Volts/Hertz Drive Parameters

The parameters described in this section are only used if the controller is configured to operate as an Open-loop Volts/Hertz Controller for AC induction motors as specified in the **Motor Drive Type** parameter (PID 0x8060). Otherwise, these parameters are not used by the controller. See section 2.2.1 for more details on general Volts/Hertz operation and configuring the desired Volts/Hertz operating profile.

**Table 20 – Volts/Hertz Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80A0	U16	V/Hz Minimum Frequency	0.1 Hz	10	0	10000	RW
0x80A1	U16	V/Hz Maximum Frequency	0.1 Hz	600	0	10000	RW
0x80A2	U16	V/Hz Inflection Frequency	0.1 Hz	100	0	10000	RW
0x80A3	U16	V/Hz Zero Hertz Voltage	0.1 Vrms	60	1	65535	RW
0x80A4	U16	V/Hz Synchronous Startup Enable	boolean	TRUE	FALSE	TRUE	RW
0x80A5	U16	V/Hz Startup Frequency	0.1 Hz	100	0	10000	RW
0x80A6	U16	V/Hz Startup Pause Time	1 ms	1000	0	65535	RW

#### 4.3.10.1 V/Hz Minimum Frequency

This parameter is used to set the minimum allowable frequency to apply to the motor. This parameter takes precedence over the start frequency - if the start frequency is less than this parameter it will be ignored and instead this parameter is used as the start frequency when first turning on. Care should be taken to not set this too high when using both positive and negative speed commands, otherwise, when the speed command changes direction, there will be a noticeable jump (discontinuity) in frequency when transitioning through zero. Applications that expect to support bipolar speed commands should set this parameter to zero (or very small).

This parameter is mainly intended for unipolar operation where a minimum electrical frequency is required.

#### 4.3.10.2 V/Hz Maximum Frequency

This parameter is used to set the maximum allowable frequency to apply to the motor. The motor speed will be clamped to this maximum frequency whenever the user's speed command would require a frequency greater than this parameter.

#### 4.3.10.3 V/Hz Inflection Frequency

This parameter is used to set the inflection point on the Volts/Hertz profile curve. Above this the point the controller will use the nominal Volts/Hertz constant determined by the rated motor voltage and frequency. Below this point the Volts/Hertz relationship is set by the slope between the Zero Hertz Voltage (see below) and this point.

#### 4.3.10.4 V/Hz Zero Hertz Voltage

This parameter is used to set the voltage to apply to the motor at zero speed (0 Hz), thus establishing the low speed Volts/Hertz slope between this point and the **V/Hz Inflection Frequency** point defined above.

#### 4.3.10.5 V/Hz Synchronous Startup Enable

When this parameter is set to TRUE, the controller will turn on at the measured motor speed. When this parameter is set to FALSE, the controller will turn on at the specified starting frequency (see below).

#### 4.3.10.6 V/Hz Startup Frequency

This parameter sets the starting frequency to apply when first turning on the motor. This parameter will be ignored if **V/Hz Synchronous Startup** has been enabled or if the **V/Hz Minimum Frequency** is greater than the value specified by this parameter.

#### 4.3.10.7 V/Hz Startup Pause Time

This parameter determines the length of time to maintain the start frequency when first turning on the motor. Once this time has expired, the controller will then go to the commanded motor speed set by the user.

### 4.3.11 Maximum Torque Profile Parameters

The parameters described in this section are only used if the controller is configured as a Closed-loop Field Oriented Motor Controller (for either PM or AC Induction motors) as specified in the **Motor Drive Type** parameter (PID 0x8060) and the drive is operated in direct torque control mode. Otherwise, these parameters are not used by the controller. See section 2.2.5 for more details on configuring the desired maximum torque operating profile.

**Table 21 – Maximum Torque Profile Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80B0	U16	Torque Profile Enable	boolean	FALSE	FALSE	TRUE	RW
0x80B1	U16	Torque Profile Maximum Torque	Nm	1000	1	65535	RW
0x80B2	U16	Torque Profile Maximum Power	10 W	10000	1	65535	RW
0x80B3	U16	Torque Profile Power*Speed Region Start	RPM	3000	0	65535	RW

**4.3.11.1 Torque Profile Enable**

This parameter is used to enable the Maximum Torque Profile feature. When this parameter is set to FALSE, the torque profile parameters are ignored and no torque limiting is performed. When set to TRUE, the motor drive enforces the maximum allowable torque versus speed profile as defined by the parameters below.

**4.3.11.2 Torque Profile Maximum Torque**

This parameter defines the maximum allowable torque when operating at the lower speed ranges (Region 1).

**4.3.11.3 Torque Profile Maximum Power**

This parameter defines the maximum allowable power. The speed at which the torque profile switches from torque limiting to power limiting (Region 1 to Region 2) is determined by the combination of this parameter and the specified maximum torque described in the previous section.

**4.3.11.4 Torque Profile Power\*Speed Region Start**

This parameter defines the speed at which the torque profile switches from power limiting to power\*speed limiting (Region 2 to Region 3).

**4.3.12 Temperature Measurement and Derating Parameters****Table 22 – Temperature Measurement and Derating Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80B8	U16	Inverter Temp Enable	boolean	TRUE	FALSE	TRUE	RW
0x80B9	S32	Inverter Temp Coefficient C0	Q16	19.67	-10000.0	10000.0	RW
0x80BA	S32	Inverter Temp Coefficient C1	Q16	120.6	-10000.0	10000.0	RW
0x80BB	S32	Inverter Temp Coefficient C2	Q16	-28.31	-10000.0	10000.0	RW
0x80BC	S32	Inverter Temp Coefficient C3	Q16	6.679	-10000.0	10000.0	RW
0x80BD	U16	Inverter Temp Derating Enable	boolean	FALSE	FALSE	TRUE	RW
0x80BE	U16	Inverter Temp Derating Low Threshold	°C	90	0	150	RW
0x80BF	U16	Inverter Temp Derating High Threshold	°C	100	0	150	RW
0x80C0	U16	Inverter Temp Min Derating Factor	%	10	0	100	RW
0x80C1	U16	Motor Temp Enable	boolean	FALSE	FALSE	TRUE	RW

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80C2	S32	Motor Temp Coefficient C0	Q16	-3.929	-10000.0	10000.0	RW
0x80C3	S32	Motor Temp Coefficient C1	Q16	152.8	-10000.0	10000.0	RW
0x80C4	S32	Motor Temp Coefficient C2	Q16	-13.88	-10000.0	10000.0	RW
0x80C5	S32	Motor Temp Coefficient C3	Q16	13.96	-10000.0	10000.0	RW
0x80C6	U16	Motor Temp Derating Enable	boolean	FALSE	FALSE	TRUE	RW
0x80C7	U16	Motor Temp Derating Low Threshold	°C	90	0	150	RW
0x80C8	U16	Motor Temp Derating High Threshold	°C	100	0	150	RW
0x80C9	U16	Motor Temp Min Derating Factor	%	10	0	100	RW
0x80CA	U16	Brake Temp Enable	boolean	FALSE	FALSE	TRUE	RW
0x80CB	S32	Brake Temp Coefficient C0	Q16	148.2	-10000.0	10000.0	RW
0x80CC	S32	Brake Temp Coefficient C1	Q16	-212.8	-10000.0	10000.0	RW
0x80CD	S32	Brake Temp Coefficient C2	Q16	212.3	-10000.0	10000.0	RW
0x80CE	S32	Brake Temp Coefficient C3	Q16	-142.5	-10000.0	10000.0	RW

#### 4.3.12.1 Inverter Temp Enable

This parameter is used to determine if there is a temperature sensing device present from the IGBT inverter interface and if it should be used to measure the inverter temperature. If this parameter is set to FALSE, the inverter temperature is not monitored or calculated and the reported inverter temperature will default to 0°C. When set to TRUE, the temperature is calculated according to the user-provided temperature coefficients (see next section) and is also monitored for possible over-temperature conditions according to the ***Inverter Temperature Fault/Warning Threshold*** configuration parameters (PIDs 0x8033 – 0x8034).

#### 4.3.12.2 Inverter Temp Coefficients (C0 through C3)

These parameters define the coefficients used by the 3<sup>rd</sup> order polynomial fitting routine within the firmware to convert raw ADC readings to °C. The temperature is calculated using the normalized ADC measurements (values in the range of 0 to 1, corresponding to the 12-bit ADC input range of 0x000 to 0xFFF) and the coefficient parameters as follows:

$$\text{Temperature} = C3 \cdot X^3 + C2 \cdot X^2 + C1 \cdot X + C0$$

Where: X is the normalized ADC reading in the range of 0 – 1  
C0 – C3 are the coefficient parameters

The factory default values for these coefficients are based on the temperature sensors contained within a typical Semikron SKiiP power module. When non-SKiiP based temperature sensing is employed, it is left up to the user to generate these coefficients for the specific sensor's temperature-to-voltage transfer function. The temperature transfer function can be derived using a common tool like Microsoft Excel in which the temperature vs. voltage data from the device's datasheet are entered, any scaling performed based on the analog signal conditioning prior to the control board connection, and then normalizing the resulting voltage

by dividing by the control boards voltage input range. The resulting ADC-to-temperature curve can be plotted and a 3<sup>rd</sup> order polynomial trend line can be generated to match the curve.

#### 4.3.12.3 Inverter Temp Derating Enable

This parameter enables the thermal derating feature where the user's motor torque command can be reduced based on high inverter temperatures. If this parameter is set to FALSE, the inverter temperature will not be used to adjust the commanded torque. If this parameter is set to TRUE and the ***Inverter Temp Enable*** parameter is TRUE (i.e. the sensor is present and being used), then the thermal derating feature will be enabled and will operate as discussed in section 2.2.4.

#### 4.3.12.4 Inverter Temp Derating Low Threshold

This parameter specifies the lower inverter temperature threshold at which thermal derating will begin. Temperatures below this value will not cause torque derating. Temperatures above this value will cause the user's torque command to be reduced as described in section 2.2.4.

#### 4.3.12.5 Inverter Temp Derating High Threshold

This parameter specifies the upper inverter temperature threshold at which the user's torque command will be reduced to the minimum derating value specified in the ***Inverter Temp Min Derating Factor*** parameter (PID 0x80C0), as discussed in section 2.2.4.

#### 4.3.12.6 Inverter Temp Min Derating Factor

This parameter is used to define the minimum derating factor to use when reducing the user's torque command due to high inverter temperatures. Normally, at low temperatures with no torque reduction the thermal derating factor is 1.0 (or 100%). This parameter specifies the maximum amount of torque reduction allowed. This parameter should be set to a value between zero (0% being the maximum derating, which reduces the torque command to zero at the maximum temperature) and 100 (100% being no derating).

#### 4.3.12.7 Motor Temp Enable

This parameter is used to determine if there is a motor temperature sensing device present and if it should be used to measure the motor temperature. If this parameter is set to FALSE, the motor temperature is not monitored or calculated and the reported motor temperature will default to 0°C. When set to TRUE, the temperature is calculated according to the user-provided temperature coefficients (see next section) and is also monitored for possible over-temperature conditions according to the ***Motor Temperature Fault/Warning Threshold*** configuration parameters (PIDs 0x8039 – 0x803A).

#### 4.3.12.8 Motor Temp Coefficients (C0 through C3)

These parameters define the coefficients used by the 3<sup>rd</sup> order polynomial fitting routine within the firmware to convert raw ADC readings to °C. See section 4.3.12.2 (***Inverter Temp Coefficients***) for a description of these polynomial coefficients.

#### 4.3.12.9 Motor Temp Derating Enable

This parameter enables the thermal derating feature where the user's motor torque command can be reduced based on high motor temperatures. If this parameter is set to FALSE, the motor temperature will not be used to adjust the commanded torque. If this parameter is set to TRUE and the **Motor Temp Enable** parameter is TRUE (i.e. the sensor is present and being used), then the thermal derating feature will be enabled and will operate as discussed in section 2.2.4.

#### 4.3.12.10 Motor Temp Derating Low Threshold

This parameter specifies the lower motor temperature threshold at which thermal derating will begin. Temperatures below this value will not cause torque derating. Temperatures above this value will cause the user's torque command to be reduced as described in section 2.2.4.

#### 4.3.12.11 Motor Temp Derating High Threshold

This parameter specifies the upper motor temperature threshold at which the user's torque command will be reduced to the minimum derating value specified in the **Motor Temp Min Derating Factor** parameter (PID 0x80C9), as discussed in section 2.2.4.

#### 4.3.12.12 Motor Temp Min Derating Factor

This parameter is used to define the minimum derating factor to use when reducing the user's torque command due to high motor temperatures. Normally, at low temperatures with no torque reduction the thermal derating factor is 1.0 (or 100%). This parameter specifies the maximum amount of torque reduction allowed. This parameter should be set to a value between zero (0% being the maximum derating, which reduces the torque command to zero at the maximum temperature) and 100 (100% being no derating).

#### 4.3.12.13 Brake Temp Enable

This parameter is used to determine if there is a temperature sensing device present in the brake chopper drive interface and if it should to be used to measure the switch temperature. If this parameter is set to FALSE, the brake temperature is not monitored or calculated and the reported brake temperature will default to 0°C. When set to TRUE, the temperature is calculated according to the user-provided temperature coefficients (see next section) and is also monitored for possible over-temperature conditions according to the **Brake Temperature Fault/Warning Threshold** configuration parameters (PIDs 0x8036 – 0x8037).

#### 4.3.12.14 Brake Temp Coefficients (C0 through C3)

These parameters define the coefficients used by the 3<sup>rd</sup> order polynomial fitting routine within the firmware to convert raw ADC readings to °C. See section 4.3.12.2 (**Inverter Temp Coefficients**) for a description of these polynomial coefficients.

### 4.3.13 Encoder Parameters

**Table 23 – Encoder Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80D8	U32	Encoder Line Count	Integer	2048	0	1E+08	RW
0x80D9	U16	Encoder Positive Direction	ENUM	1	0	1	RW
0x80DA	U16	Encoder Clock Timer Prescaler	ENUM	6	0	7	RW
0x80DB	U16	Encoder Event Counter Prescaler	ENUM	2	0	11	RW
0x80DC	U16	Encoder Debounce Prescaler	ENUM	4	0	255	RW
0x80DD	U16	Encoder Debounce Select	ENUM	2	0	2	RW

#### 4.3.13.1 Encoder Line Count

This parameter is used to specify the line count for the encoder being used.

#### 4.3.13.2 Encoder Positive Direction

This parameter is used to define the phase relationship between A and B that results in a positive speed calculation by the motor controller. The legal values for this parameter are as follows:

- 0 = A LEADS B – a positive speed is calculated when A leads B by 90°
- 1 = B LEADS A – a positive speed is calculated when B leads A by 90°

#### 4.3.13.3 Encoder Clock Timer Prescaler

This parameter is used to configure the clock that is used to time the QEP input events. The clock will be set to  $\text{SYSCLK} / 2^N$ , where  $N$  is the value specified by this parameter and  $\text{SYSCLK}$  is the internal clock rate of the DSP on the control board (150MHz). The legal values for this parameter are 0 through 7, yielding a range of clocks from  $\text{SYSCLK} / 2^0 = 150\text{MHz}$  to  $\text{SYSCLK} / 2^7 = 1.17\text{MHz}$ .

This clock timer is used when the motor speed is being calculated at low speeds. In this mode, the speed is calculated as  $\Delta \text{position} / \Delta \text{time}$ , where the change in position is fixed and based on the **Encoder Line Count** and the **Encoder Event Counter Prescale** and the change in time is measured using the clock timer specified by this parameter.

#### 4.3.13.4 Encoder Event Counter Prescaler

This parameter is used to determine the encoder event (an edge on the A or B input) pre-scale value to use when the motor speed is being calculated at low speeds. With no pre-scale, an encoder event is defined as any edge on the encoder input (either A or B), and so the controller would be timing the duration between any two adjacent edges. The event pre-scale is set to  $\text{EVENT} / 2^N$ , where  $N$  is the value specified by this parameter. The legal values for this parameter are 0 through 11, yielding a range of event periods from  $\text{EVENT} / 2^0 = \text{time the}$

duration between every event to  $EVENT / 2^{11} = EVENT / 2048 =$  time duration between every 2048<sup>th</sup> event.

#### 4.3.13.5 Encoder Debounce Prescaler

When encoder pin debounce is enabled (see next parameter), this parameter is used to define the clock period of the debounce clock. The debounce clock is set to  $SYSCLK / (2 * N)$ , where N is the value specified by this parameter. The legal range for this parameter is N=0 (debounce clock =  $SYSCLK = 150MHz$ ) to N=0xFF (debounce clock =  $SYSCLK/510 = 294kHz$ ).

For electrically noisy environments that could cause false noise-induced edges on the encoder inputs, the user may wish to configure aggressive debounce (a slower debounce clock and a higher number of debounce edges). One side effect of doing so is that the resulting encoder signals used by the controller will be quantized by the configured debounce period. This quantization effect could be undesirable if a precise speed measurement is required by the end application. For systems where noise is not an issue or where it is important to preserve the exact timing of the encoder pulses, minimal debounce should be used.

#### 4.3.13.6 Encoder Debounce Select

This parameter is used to select the encoder input debounce event period as follows:

- 0 = OFF – no debounce is performed
- 1 = X3 – three consecutive samples must be the same value to be considered a valid transition (as clocked by the prescaled clock, configured in the previous parameter)
- 2 = X6 – six consecutive samples must be the same value to be considered a valid transition (as clocked by the prescaled clock, configured in the previous parameter)

The user should take care to ensure that the debounce time (# consecutive samples \* debounce clock period) does not exceed the minimum expected period of the encoder input signals (determined by the maximum motor speed and encoder line count).

#### 4.3.14 Motor Speed Calculation Parameters

The following set of parameters affects how the controller calculates the motor speed when using a digital quadrature encoder device for speed feedback. If an encoder is not being used, these parameters will not be used by the controller.

**Table 24 – Motor Speed Calculation Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80E0	U16	Motor Speed Low-to-High Crossover Mode	ENUM	0	0	1	RW
0x80E1	S32	Motor Speed Low-to-High RPM	RPM	1000	1	65535	RW
0x80E2	S32	Motor Speed Low-to-High Hysteresis	RPM	100	1	65535	RW
0x80E3	S32	Motor Speed Low Pass Filter Cutoff Freq	Hz	1000	1	65535	RW
0x80E4	U16	Motor Speed Deadband Period (Low Speed)	Counts	0xC000	0x0064	0xFFFF	RW



PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x80E5	U16	Motor Speed Update Prescale (High Speed)	Counts	8	1	16	RW

#### 4.3.14.1 Motor Speed Low-to-High Crossover Mode

This parameter is used to determine when the speed calculator switches between low-speed and high-speed calculation techniques as follows:

- 0 = AUTO – the firmware automatically determines where the speed resolution of the two techniques is equal and uses this as the switchover speed
- 1 = MANUAL – the user specifies the desired switchover speed using the Motor Speed Low-to-High RPM parameter (see next parameter)

Using manual mode may be useful in situations where the encoder edge accuracy or signal timing is poor.

#### 4.3.14.2 Motor Speed Low-to-High RPM

When manual speed crossover mode is selected (see previous parameter), this parameter is used to specify the motor speed at which the speed calculator will switch between the low and high speed calculation techniques. When using manual crossover mode, the user will generally set this value to be a lower value than the automatically calculated crossover point to avoid noise or edge accuracy affects on the speed resolution when using the low speed calculation technique.

#### 4.3.14.3 Motor Speed Low-to-High Hysteresis

This parameter determines how much hysteresis to use when switching between the low and high speed calculation techniques. This hysteresis value is used in both AUTO and MANUAL crossover mode. The speed calculator will switch from low speed to high speed mode when the motor speed is at the switchover speed plus the value defined by this parameter. Similarly, the speed calculator switches from high speed to low speed mode when the motor speed is at the switchover speed minus the value defined by this parameter. This behavior results in a hysteresis band equal to twice the value specified by this parameter.

#### 4.3.14.4 Motor Speed Low Pass Filter Cutoff Freq

This parameter is used to set the cutoff frequency of the first order low pass digital filter that is used to filter the measured motor speed. The measured speed is used for both instrumentation purposes (sent in the Motor Status message) as well as for control purposes when operating in torque mode or closed-loop speed control mode. The low pass filter is updated at the PWM rate specified in the **Pulse Width Modulation Frequency** parameter (PID 0x8050), so this parameter should be set to a value less than this PWM rate.

#### 4.3.14.5 Motor Speed Deadband Period (Low Speed)

This parameter is used to determine the encoder event timer count dead band period above which the speed is considered zero. This parameter is expressed in terms of the number of encoder timer clock periods above which the speed is forced to zero.

This parameter is only used when using the low-speed method for calculating motor speed. In this mode of operation, the calculated speed is updated only at encoder event boundaries or when the event timer saturates (which could be a long time). This dead band period parameter allows the user to shorten the effective “saturation” timer count so that zero speed can be determined sooner.

#### 4.3.14.6 Motor Speed Update Prescale (High Speed)

This parameter is used when the motor speed is being calculated at high speeds. In this mode, the speed is calculated as  $\Delta position / \Delta time$ , where the change in position is determined using the encoder position counter and the change in time is normally set to the control ISR period, which is equal to the configurable PWM rate. This parameter is used to pre-scale the time interval to be a multiple of the PWM period from 1 to 16, to allow for adjustable speed resolution.

The controller stores the position counter every PWM period in a 16-deep buffer. The speed calculator compares the last position count value to the counter value from N cycles ago where N is the value specified in this parameter. This effectively produces a larger window of time over which to look at the change in the position count, which in turn increases the effective resolution of the calculated speed. Note that using this implementation method, the speed calculation is still updated every PWM period (not every Nth PWM period).

#### 4.3.15 Current Regulator Parameters

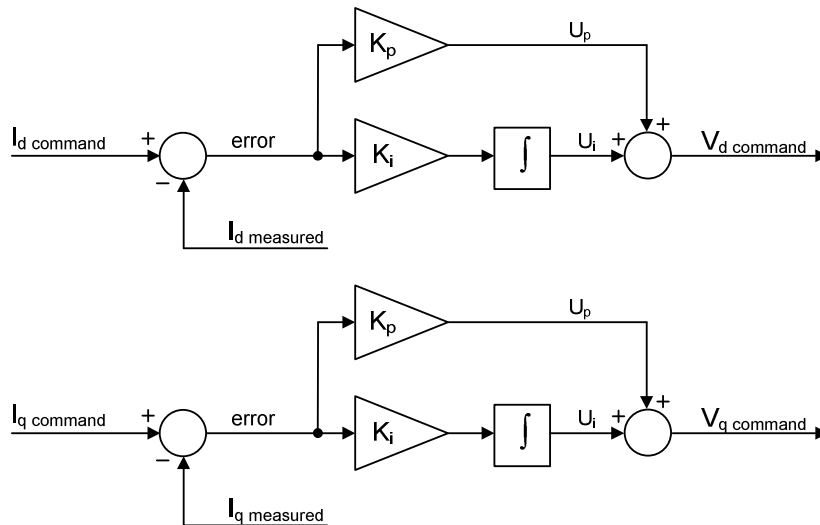
Table 25 – Current Regulator Parameter Summary

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x00F0	S32	Current Controller Kp	Q16	1.0	0.0	32767.99	RW
0x00F1	S32	Current Controller Ki	Q16	1000	0.0	32767.99	RW
0x00F2	U16	Id Current Foldback Enable	boolean	TRUE	FALSE	TRUE	RW
0x00F3	U16	Id Current Foldback Mod Index Threshold	0.1 %	960	0	1000	RW
0x00F4	S32	Id Foldback Controller Kp	Q16	0.2	0.0	32767.99	RW
0x00F5	S32	Id Foldback Controller Ki	Q16	200	0.0	32767.99	RW
0x00F6	U16	Iq Current Foldback Enable	boolean	TRUE	FALSE	TRUE	RW
0x00F7	U16	Iq Current Foldback Mod Index Threshold	0.1 %	950	0	1000	RW
0x00F8	S32	Iq Foldback Controller Kp	Q16	0.2	0.0	32767.99	RW
0x00F9	S32	Iq Foldback Controller Ki	Q16	200	0.0	32767.99	RW

##### 4.3.15.1 Current Controller Gain Constants (Kp, Ki)

These parameters define the gain constants for the PI controllers that are used to regulate the direct field-producing current ( $I_d$ ) and the quadrature torque-producing ( $I_q$ ) current when the control loops are updated. The integral gain ( $K_i$ ) parameter should be entered as the continuous

gain (or sometimes referred to as the “analog” gain). The firmware handles converting this to the discrete-time gain by automatically dividing this by the sample frequency at which the controller is updated (at the specified PWM frequency).



**Figure 13 – Current Control PI Regulators**

As the figure above shows, the PI topology uses separate proportional and integral correction terms that are then summed together to create the desired voltage command ( $V_d$  or  $V_q$ ). There are two separate PI blocks, one used to regulate the  $I_d$  current and one to regulate the  $I_q$  current. The torque-producing current ( $I_q$ ) is either commanded indirectly by the user when operating in torque mode or commanded by the outer speed loop when operating in closed-loop speed control mode. The field-producing current ( $I_d$ ) is commanded to the magnetizing current or the lower field weakened current (if enabled) when driving AC induction motors, or to zero when driving permanent magnet motors.

The current commands and current feedback terms as seen by the PI regulators are normalized relative to the full scale motor current specified by the **Motor Phase Current Measurement – Full Scale** configuration parameter (PID 0x8022). The output of each PI block is then used to command an appropriate  $V_d/V_q$  applied voltage command to the Space Vector Modulator (as shown earlier in this document in Figure 3), with a scale of 0 to 100% modulation index. The  $K_p$  and  $K_i$  gain constants should be scaled appropriately based on this normalization.

#### 4.3.15.2 Id/Iq Current Foldback Enable

These parameters are used to enable the  $I_d$  or  $I_q$  foldback mechanism. When set to FALSE, the corresponding current foldback controller is disabled. When set to TRUE, the corresponding current foldback controller is enabled.

#### 4.3.15.3 Id/Iq Current Foldback Modulation Index Threshold

These parameters specify the modulation index level at which the corresponding controller should start folding back the commanded current. For proper foldback operation, this

parameter should be set slightly below 100% (generally in the mid 90's). Also, the intention of the foldback mechanism is to foldback  $I_q$  prior to folding back  $I_d$ , so the  $I_d$  threshold should be set slightly higher than the  $I_q$  foldback reference (see section 2.2.6 for more details).

#### 4.3.15.4 $I_d/I_q$ Current Foldback Controller Gain Constants ( $K_p$ , $K_i$ )

These parameters set the proportional and integral gain constants for the PI regulator that is used by the foldback controllers. The PI topology is the same as the main current regulators shown above with the exception of a clamp on the integrator and output. The calculated error input to the regulator is the difference between the maximum modulation index and the actual modulation index. The output from the compensator is a number between 0 and 1.0 that is multiplied by the corresponding current command, where a value of 1.0 indicates no foldback, a value of 0.9 would indicate that the corresponding current command has been reduced by 10%. See section 2.2.6 for more details on the current foldback mechanism.

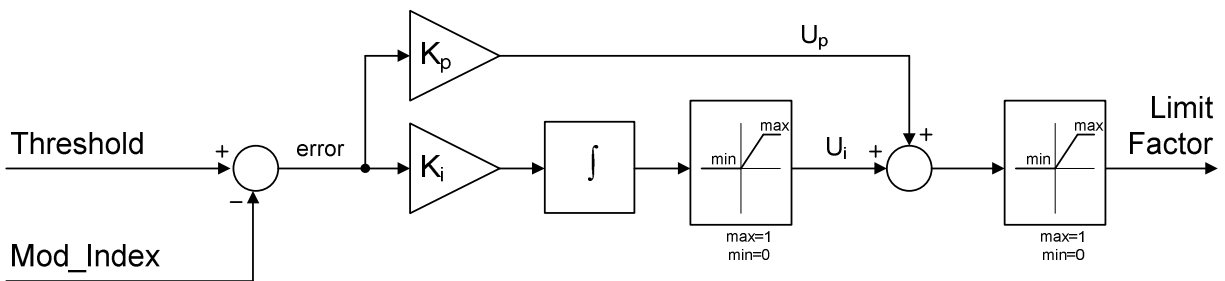


Figure 14 – Current Fold Back PI Regulator

#### 4.3.16 Speed Regulator Parameters

The parameters described in this section are only used if the controller is configured as a Closed-loop Field Oriented Motor Controller (for either PM or AC Induction motors) as specified in the **Motor Drive Type** parameter (PID 0x8060) and the drive is operated in speed control mode. Otherwise, these parameters are not used by the controller.

Table 26 – Speed Regulator Parameter Summary

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8100	U16	Speed Control Enable	boolean	FALSE	FALSE	TRUE	RW
0x8101	S32	Speed Controller $K_p$	Q16	0.2	0.0	32767.99	RW
0x8102	S32	Speed Controller $K_i$	Q16	0.0	0.0	32767.99	RW
0x8103	S16	Speed Controller Maximum Current	0.1 Arms	100	-32768	32767	RW
0x8104	S16	Speed Controller Minimum Current	0.1 Arms	-100	-32768	32767	RW

##### 4.3.16.1 Speed Control Enable

This parameter is used to enable the use of the closed-loop speed controller. Setting this parameter to TRUE will allow the user to operate the drive in either direct torque-control mode

or in speed control mode, depending on the user's Mode command sent to the controller over the CAN bus. Setting this parameter to FALSE will disable the use of the speed controller and will only allow the drive to be operated in direct torque control mode. Any attempts to change the operating mode to speed control via CAN will be ignored and the drive will remain in direct torque control mode.

#### 4.3.16.2 Speed Controller Gain Constants ( $K_p$ , $K_i$ )

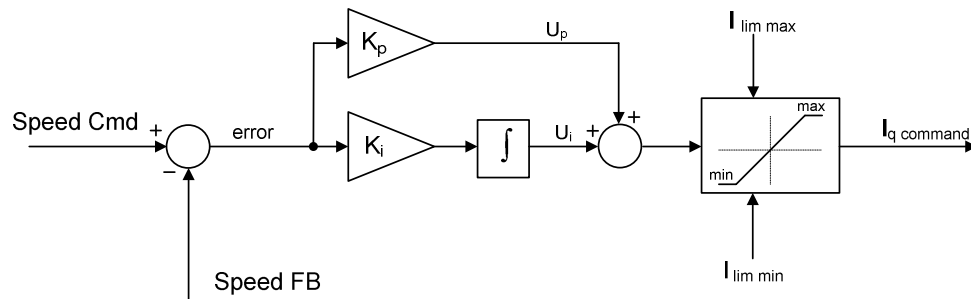


Figure 15 – Speed Control PI Regulator

These parameters define the gain constants for the PI controller that is regulating the motor speed. The integral gain ( $K_i$ ) parameter should be entered as the continuous gain (or sometimes referred to as the “analog” gain). The firmware handles converting this to the discretized gain by automatically dividing this by the sample frequency at which the controller is updated (specified by the **Pulse Width Modulation Frequency** parameter – PID 0x8050).

As above illustrates, the PI topology used sums the proportional and integral correction terms and then clamps the output to the specified current limits based on the **Speed Controller Maximum/Minimum Current** configuration parameters (see next section).

It is important to note that the speed setpoint and measured speed feedback used at the input to the speed regulator are actually normalized relative to the maximum motor frequency value specified by the **Motor Maximum Frequency** configuration parameter (PID 0x8074). Similarly, the resulting current command output is also a normalized current relative to the full scale motor current specified by the **Motor Phase Current Measurement – Full Scale** configuration parameter (PID 0x8022). The  $K_p$  and  $K_i$  gain constants should be scaled appropriately based on this normalization.

#### 4.3.16.3 Speed Controller Maximum/Minimum Current

These parameters specify the current limits used by the speed controller. For flexibility, there are separate parameters provided for the positive and negative current limits. The output of the speed loop will be clamped to be within these current limits.

### 4.3.17 Dual Interleaved Inverter Parameters

**Table 27 – Dual Interleaved Inverter Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8108	U16	Dual Interleaved Inverter Enable	boolean	FALSE	FALSE	TRUE	RW
0x8109	U16	Zero Sequence Current Control Enable	boolean	TRUE	FALSE	TRUE	RW
0x810A	S32	Kp – Zero Sequence Current Controller	Q16	1.0	0.0	32767.99	RW
0x810B	S32	Ki – Zero Sequence Current Controller	Q16	1000	0.0	32767.99	RW

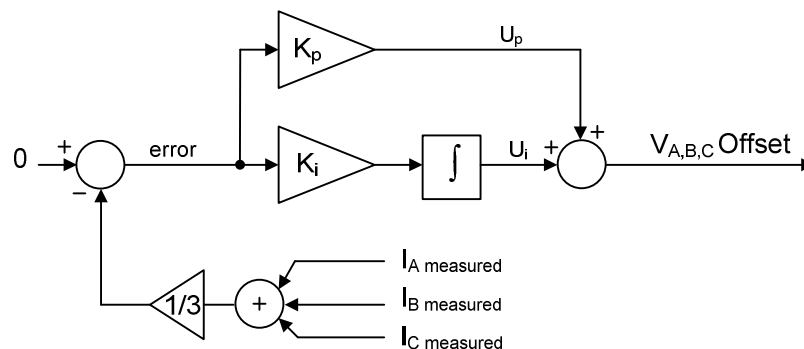
#### 4.3.17.1 Dual Interleaved Inverter Enable

When set to TRUE, this parameter enables controls for a second inverter in parallel with the main inverter for use in a dual inverter interleaved power stage. When set to FALSE, the controller will only drive one inverter interface and all other parameters in this section are ignored.

#### 4.3.17.2 Zero Sequence Current Control Enable

Setting this parameter to TRUE will enable a PI regulator to be used to force the zero sequence current on the second inverter to zero. This is intended to prevent a DC current offset from developing and circulating between the two parallel inverters.

#### 4.3.17.3 Zero Sequence Current Controller Gain Constants (Kp, Ki)



**Figure 16 – Zero Sequence PI Regulator**

These parameters define the gain constants for the PI controller that is used to regulate the zero sequence current in the second inverter. The integral gain ( $K_i$ ) parameter should be entered as the continuous gain (or sometimes referred to as the “analog” gain). The firmware handles converting this to the discrete-time gain by automatically dividing this by the sample frequency at which the controller is updated (the PWM frequency).

The PI topology used by the zero sequence current controller is the same as that used by the main current controllers in that the proportional and integral correction terms are summed together (see the figure above). The output of the PI block is then summed into the PWM

modulator outputs, effectively acting as a DC offset to the commanded  $V_{A,B,C}$  voltage waveforms.

The current feedback signals ( $I_{A,B,C}$  in the figure above) are normalized relative to the full scale inverter current specified by the **Motor Phase Current Measurement – Full Scale** configuration parameter (PID 0x8022). The output of the PI controller is added to the PWM duty cycle commands, effectively increasing or decreasing the overall PWM modulation index. The Kp and Ki gain constants should be scaled appropriately based on this normalization. In general these gains can be set to the same values as the main  $I_d/I_q$  current loops; these parameters are provided to increase system-level tuning flexibility.

#### 4.3.18 Brake Controller Parameters

**Table 28 – Brake Controller Parameter Summary**

PID	Data Type	Description	Units	Factory Default	Min	Max	Access Level
0x8110	U16	Brake Controller Enable	boolean	false	false	true	RW
0x8111	U16	Brake Controller Voltage Threshold	0.1 V	7250	0	15000	RW
0x8112	U16	Brake Controller Recover Voltage Threshold	0.1 V	7000	0	15000	RW
0x8113	U16	Brake Controller Fault Motor Response	ENUM	1	0	2	RW
0x8114	U16	Brake Gate Driver Signal Active High	boolean	true	false	true	RW

##### 4.3.18.1 Brake Controller Enable

This parameter is used to enable the brake chopper controller. When this parameter is set to FALSE, brake chopper operation and all related features are disabled. In this case the brake controller state will always read as OFF, the switch output will always be inactive, and the temperature measurement block will be disabled. When this parameter is set to TRUE, the brake chopper is allowed to operate as specified in section 2.3

##### 4.3.18.2 Brake Controller Voltage Threshold

This parameter specifies the DC Link voltage above which the brake chopper should engage the brake resistor. Once the brake resistor is engaged, it remains connected until the DC Link voltage drops below the value specified by the **Brake Controller Recover Voltage Threshold**.

##### 4.3.18.3 Brake Controller Recover Voltage Threshold

This parameter specifies the DC Link voltage at which the brake chopper should disconnect the brake resistor if it is currently engaged. This value should be set to a value less than the **Brake Chopper Voltage Threshold** in order to provide some hysteresis between turning the brake resistor on and off.

##### 4.3.18.4 Brake Controller Fault Motor Response

This parameter is used to specify whether or not the motor controller is affected by a brake controller fault. The allowable settings for this parameter are:

- 0 = MOTOR FAULT – when this mode is selected the motor controller will go to the FAULT state any time a brake controller fault occurs.
- 1 = NO BRAKING TORQUE – when this mode is selected and the motor drive is in direct torque control mode, the motor drive is allowed to continue operating with the exception that any braking torque commands are ignored and instead the torque command is set to 0. This behavior is only supported when the motor drive is in torque control mode – if the drive is in speed mode and this mode is selected, a brake controller fault will force the motor controller to go to the FAULT state.
- 2 = DO NOTHING – when this mode is selected, brake faults have no affect on the motor controller.

#### 4.3.18.5 Brake Gate Driver Signal Active High

This parameter specifies the logic level that is used to turn ON the gate driver. If the value is set to TRUE, the drive signal out of the control board is driven low when the Brake Chopper is off (i.e. the signal is treated as active high to connect the brake resistor). If this value is set to FALSE, the signal is driven high when the Brake Chopper is off.

## 5. Oztek PCC Tool

The Oztek Power Control Center (PCC) is a Microsoft Windows based Graphical User Interface (GUI) that provides the user a simple graphical interface for interacting with Oztek's embedded Power Control products. The PCC provides the ability to control, monitor, and configure the motor controller. For detailed information and operating instructions, please refer to UM-0036 – Power Control Center GUI User's Manual.

## 6. Maintenance and Upgrade

The firmware image on the OZDSP3000 can be upgraded in-system using the resident CAN bootloader. For detailed information on how to upgrade the firmware or directly interface with the bootloader, please reference UM-0015 Oztek TMS28x CAN Bootloader Users Manual.



# Warranty and Product Information

## Limited Warranty

**What does this warranty cover and how long does it last?** This Limited Warranty is provided by Oztek Corp. ("Oztek") and covers defects in workmanship and materials in your OZDSP3000 controller. This Warranty Period lasts for 18 months from the date of purchase at the point of sale to you, the original end user customer, unless otherwise agreed in writing. You will be required to demonstrate proof of purchase to make warranty claims. This Limited Warranty is transferable to subsequent owners but only for the unexpired portion of the Warranty Period. Subsequent owners also require original proof of purchase as described in "What proof of purchase is required?"

**What will Oztek do?** During the Warranty Period Oztek will, at its option, repair the product (if economically feasible) or replace the defective product free of charge, provided that you notify Oztek of the product defect within the Warranty Period, and provided that through inspection Oztek establishes the existence of such a defect and that it is covered by this Limited Warranty.

Oztek will, at its option, use new and/or reconditioned parts in performing warranty repair and building replacement products. Oztek reserves the right to use parts or products of original or improved design in the repair or replacement. If Oztek repairs or replaces a product, its warranty continues for the remaining portion of the original Warranty Period or 90 days from the date of the return shipment to the customer, whichever is greater. All replaced products and all parts removed from repaired products become the property of Oztek.

Oztek covers both parts and labor necessary to repair the product, and return shipment to the customer via an Oztek-selected non-expedited surface freight within the contiguous United States and Canada. Alaska, Hawaii and locations outside of the United States and Canada are excluded. Contact Oztek Customer Service for details on freight policy for return shipments from excluded areas.

**How do you get service?** If your product requires troubleshooting or warranty service, contact your merchant. If you are unable to contact your merchant, or the merchant is unable to provide service, contact Oztek directly at:

USA  
Telephone: 603-546-0090  
Fax: 603-386-6366  
Email [techsupport@oztekc corp.com](mailto:techsupport@oztekc corp.com)

Direct returns may be performed according to the Oztek Return Material Authorization Policy described in your product manual.

**What proof of purchase is required?** In any warranty claim, dated proof of purchase must accompany the product and the product must not have been disassembled or modified without prior written authorization by Oztek. Proof of purchase may be in any one of the following forms:

- The dated purchase receipt from the original purchase of the product at point of sale to the end user
- The dated dealer invoice or purchase receipt showing original equipment manufacturer (OEM) status
- The dated invoice or purchase receipt showing the product exchanged under warranty

**What does this warranty not cover?** Claims are limited to repair and replacement, or if in Oztek's discretion that is not possible, reimbursement up to the purchase price paid for the product. Oztek will be liable to you only for direct damages suffered by you and only up to a maximum amount equal to the purchase price of the product. This Limited Warranty does not warrant uninterrupted or error-free operation of the product or cover normal wear and tear of the product or costs related to the removal, installation, or troubleshooting of the customer's electrical systems. This warranty does not apply to and Oztek will not be responsible for any defect in or damage to:

- a) The product if it has been misused, neglected, improperly installed, physically damaged or altered, either internally or externally, or damaged from improper use or use in an unsuitable environment
- b) The product if it has been subjected to fire, water, generalized corrosion, biological infestations, or input voltage that creates operating conditions beyond the maximum or minimum limits listed in the Oztek product specifications including high input voltage from generators and lightning strikes
- c) The product if repairs have been done to it other than by Oztek or its authorized service centers (hereafter "ASCs")
- d) The product if it is used as a component part of a product expressly warranted by another manufacturer
- e) The product if its original identification (trade-mark, serial number) markings have been defaced, altered, or removed
- f) The product if it is located outside of the country where it was purchased
- g) Any consequential losses that are attributable to the product losing power whether by product malfunction, installation error or misuse.

## Disclaimer

### Product

THIS LIMITED WARRANTY IS THE SOLE AND EXCLUSIVE WARRANTY PROVIDED BY OZTEK IN CONNECTION WITH YOUR OZTEK PRODUCT AND IS, WHERE PERMITTED BY LAW, IN LIEU OF ALL OTHER WARRANTIES, CONDITIONS, GUARANTEES, REPRESENTATIONS, OBLIGATIONS AND LIABILITIES, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE IN CONNECTION WITH THE PRODUCT, HOWEVER ARISING (WHETHER BY CONTRACT, TORT, NEGLIGENCE, PRINCIPLES OF MANUFACTURER'S LIABILITY, OPERATION OF LAW, CONDUCT, STATEMENT OR OTHERWISE), INCLUDING WITHOUT RESTRICTION ANY IMPLIED WARRANTY OR CONDITION OF QUALITY, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE TO THE EXTENT REQUIRED UNDER APPLICABLE LAW TO APPLY TO THE PRODUCT SHALL BE LIMITED IN DURATION TO THE PERIOD STIPULATED UNDER THIS LIMITED WARRANTY. IN NO EVENT WILL OZTEK BE LIABLE FOR: (a) ANY SPECIAL, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, LOST REVENUES, FAILURE TO REALIZE EXPECTED SAVINGS, OR OTHER COMMERCIAL OR ECONOMIC LOSSES OF ANY KIND, EVEN IF OZTEK HAS BEEN ADVISED, OR HAD REASON TO KNOW, OF THE POSSIBILITY OF SUCH DAMAGE, (b) ANY LIABILITY ARISING IN TORT, WHETHER OR NOT ARISING OUT OF OZTEK'S NEGLIGENCE, AND ALL LOSSES OR DAMAGES TO ANY PROPERTY OR FOR ANY PERSONAL INJURY OR ECONOMIC LOSS OR DAMAGE CAUSED BY THE CONNECTION OF A PRODUCT TO ANY OTHER DEVICE OR SYSTEM, AND (c) ANY DAMAGE OR INJURY ARISING FROM OR AS A RESULT OF MISUSE OR ABUSE, OR THE INCORRECT INSTALLATION, INTEGRATION OR OPERATION OF THE PRODUCT. IF YOU ARE A CONSUMER (RATHER THAN A PURCHASER OF THE PRODUCT IN THE COURSE OF A BUSINESS) AND PURCHASED THE PRODUCT IN A MEMBER STATE OF THE EUROPEAN UNION, THIS LIMITED WARRANTY SHALL BE SUBJECT TO YOUR STATUTORY RIGHTS AS A CONSUMER UNDER THE EUROPEAN UNION PRODUCT WARRANTY DIRECTIVE 1999/44/EC AND AS SUCH DIRECTIVE HAS BEEN IMPLEMENTED IN THE EUROPEAN UNION MEMBER STATE WHERE YOU PURCHASED THE PRODUCT. FURTHER, WHILE THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, YOU MAY HAVE OTHER RIGHTS WHICH MAY VARY FROM EU MEMBER STATE TO EU MEMBER STATE OR, IF YOU DID NOT PURCHASE THE PRODUCT IN AN EU MEMBER STATE, IN THE COUNTRY YOU PURCHASED THE PRODUCT WHICH MAY VARY FROM COUNTRY TO COUNTRY AND JURISDICTION TO JURISDICTION.

## Return Material Authorization Policy

Before returning a product directly to Oztek you must obtain a Return Material Authorization (RMA) number and the correct factory "Ship To" address. Products must also be shipped prepaid. Product shipments will be refused and returned at your expense if they are unauthorized, returned without an RMA number clearly marked on the outside of the shipping box, if they are shipped collect, or if they are shipped to the wrong location.

When you contact Oztek to obtain service, please have your instruction manual ready for reference and be prepared to supply:

- The serial number of your product
- Information about the installation and use of the unit
- Information about the failure and/or reason for the return
- A copy of your dated proof of purchase

### Return Procedure

Package the unit safely, preferably using the original box and packing materials. Please ensure that your product is shipped fully insured in the original packaging or equivalent. This warranty will not apply where the product is damaged due to improper packaging. Include the following:

- The RMA number supplied by Oztek clearly marked on the outside of the box.
- A return address where the unit can be shipped. Post office boxes are not acceptable.
- A contact telephone number where you can be reached during work hours.
- A brief description of the problem.

Ship the unit prepaid to the address provided by your Oztek customer service representative.

**If you are returning a product from outside of the USA or Canada** - In addition to the above, you **MUST** include return freight funds and you are fully responsible for all documents, duties, tariffs, and deposits.

### Out of Warranty Service

If the warranty period for your product has expired, if the unit was damaged by misuse or incorrect installation, if other conditions of the warranty have not been met, or if no dated proof of purchase is available, your unit may be serviced or replaced for a flat fee. If a unit cannot be serviced due to damage beyond salvation or because the repair is not economically feasible, a labor fee may still be incurred for the time spent making this determination.

To return your product for out of warranty service, contact Oztek Customer Service for a Return Material Authorization (RMA) number and follow the other steps outlined in "Return Procedure".

Payment options such as credit card or money order will be explained by the Customer Service Representative. In cases where the minimum flat fee does not apply, as with incomplete units or units with excessive damage, an additional fee will be charged. If applicable, you will be contacted by Customer Service once your unit has been received.