Innovative Thinking for Power, Control and Instrumentation

OZDSP3000
FPGA Functional Specification

FS-0044

11 Continental Blvd Merrimack NH 03054
v. (603) 546-0090 f. (603)386-6366 oztekcorp.com

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

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.

Date and Revision

October 2011 Rev G

Part Number

FS-0044

Contact Information

USA
Telephone: 603-546-0090
Fax: 603-386-6366
Email techsupport@oztekcorp.com

Table of Contents

Table of Figures

Table of Tables

1. Introduction

The purpose of this document is twofold – it is meant to convey both the design requirements for the FPGA on the OZDSP3000 Control Board as well as to provide a functional description of the final implementation.

1.1 Referenced Documents

Ref.	Document	Description
[1]	10741-xx	OZDSP3000 Control Board Schematic
[2]	HB1002	MachXO Family Handbook, Lattice Semiconductor Corp.

1.2 Definitions

AFE	Active Front End
CAN	Controller Area Network
DSP	Digital signal processor
EEPROM	Electrically Erasable Programmable Read Only Memory
EMC	Electro-magnetic compatibility
EMI	Electro-magnetic interference
GND	Ground, low side of input power supply
GTI	Grid Tied Inverter
GUI	Graphical User Interface
HMI	Human Machine Interface
IPM	Intelligent Power Module
N.C.	Not connected
PCB	Printed Circuit Board
PCC	Power Control Center
PLC	Programmable Logic Controller
PLL	Phase Locked Loop
POR	Power On Reset
PWM	Pulse width modulation
SVM	Space Vector Modulator

2. Overview

The OZDSP3000 Control Board FPGA is implemented using a Lattice MachXO LCMXO1200C device in a 144-pin QPF package. This design provides an interface between the DSP and several peripheral circuits on the control board. The following is a high-level summary of the functions provided by this FPGA:

- Parallel read/write interface to the TMS320F28335 DSP for internal register access and control
- Provides fault latching and interrupt generation to the DSP for various inverter and system related faults
- Provides an interface to the ADS7863. This is a 12-bit, 3+3 channel analog-to-digital converter used for measuring high voltages (grid voltages and output voltages) on the control board.
- Provides an interface to the TPIC44H01 high-side FET driver IC's, used to provide a drive for 4 relay drive outputs.
- Provides an interface to the TLV5620. This is an 8-bit, 4 channel digital-to-analog converter that is used to generate debugging analog outputs.
- Provides support for the following general purpose pins:
    - 4 isolated general purpose inputs
    - 4 isolated general purpose outputs
    - 6 general purpose spare I/O connected to the DSP for future expansion
- Provides an interface to the DS1302 Real Time Clock IC
- Provides control for LEDs
    - 4 debug/heartbeat LEDs connected to the spare I/O pins
    - 4 warning LEDs for the two inverters, one half bridge, and a system-level warning
- Provides support for miscellaneous status pins (hardware revision)
- Provides selection of the analog mux signal
    - 3 inverter temperatures
    - 3 auxiliary thermistor temperatures
    - 4 voltage monitors
- Provides four interrupt signals (two inverters, one half bridge, and system)
- Provides chip select decoding between the Anybus and Oztek Expansion Board interfaces

The figure below shows a high-level block diagram for this FPGA. Each block is described in more detail in later sections of this document.

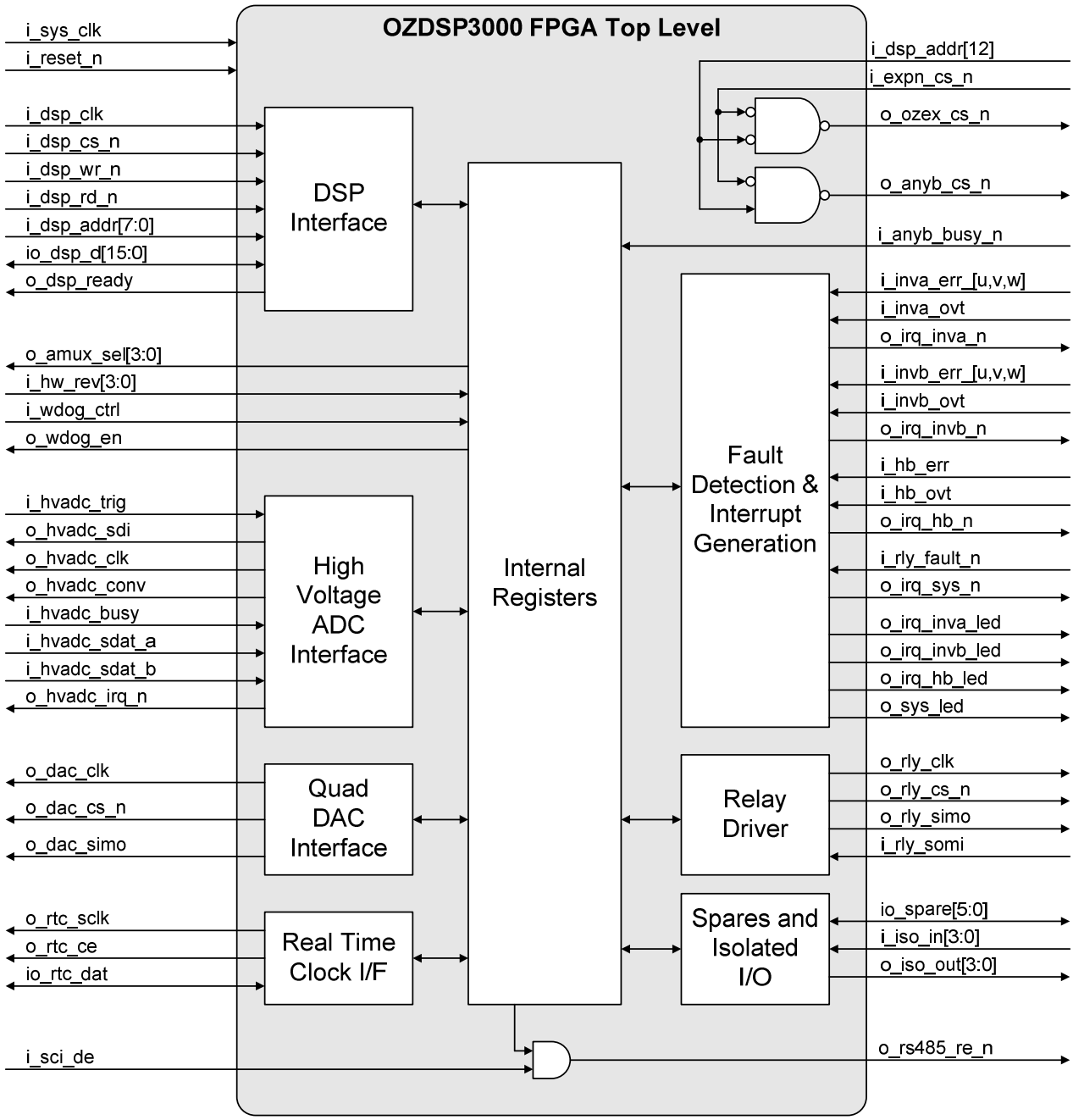


Figure 1 – FPGA Top-level Block Diagram

The following table lists the I/O pins used for this FPGA:

Table 1 – FPGA Pin Descriptions

Pin Name	Direction	Description
i_sys_clk	IN	25 MHz System Clock
i_reset_n	IN	Board Reset (active low)
DSP Interface:		
i_dsp_clk	IN	DSP clock output, up to 150MHz
i_dsp_cs_n	IN	FPGA chip select from DSP (active low)
i_dsp_wr_n	IN	DSP write enable (active low)
i_dsp_rd_n	IN	DSP read enable (active low)
i_dsp_addr[12,7:0]	IN	DSP address bus (bit 12 for Expansion chip select decoding)
io_dsp_d[15:0]	IN/OUT	DSP data bus
o_dsp_ready	OUT	DSP ready, used to generate extra bus wait states
io _spare[5:0]	IN/OUT	Spare pins to DSP. Pins 0-3 are connected to debug LEDs
General Purpose I/O:		
i_iso_in[3:0]	IN	Isolated General Purpose Digital Inputs
o_iso_out[3:0]	OUT	Isolated General Purpose Digital Outputs
Expansion Board Interface:		
i_expn_cs_n	IN	Anybus Chip Select from DSP
o_anyb_cs_n	OUT	Decoded Anybus Chip Select
o_ozex_cs_n	OUT	Decoded Oztek Expansion Board Chip Select
i_anyb_busy_n	IN	Anybus Access Busy Indicator (not presently used)
High Voltage ADC:		
i_hvadc_trig	IN	Trigger from DSP to initiate ADC conversion
o_hvadc_clk	OUT	Serial clock to ADC
o_hvadc_conv	OUT	Convert Start pulse to ADC
o_hvadc_sdi	OUT	Selects ADC channels to be read
i_hvadc_busy	IN	Conversion busy indicator (1=busy) from ADC
i_hvadc_sdat_a	IN	Serial data ‘A’ from ADC
i_hvadc_sdat_b	IN	Serial data ‘B’ from ADC
o_hvadc_irq_n	OUT	End of conversion interrupt to DSP
Inverter Faults:		
i_inva_err_u/v/w	IN	Error inputs (active high) from inverter A interface
i_inva_ovt	IN	Over temp fault (active high) from inverter A interface
o_irq_inva_n	OUT	Inverter A fault interrupt to DSP (active low)
o_irq_inva_led	OUT	Inverter A fault LED control – lit when fault exists
i_invb_err_u/v/w	IN	Error inputs (active high) from inverter B interface
i_invb_ovt	IN	Over temp fault (active high) from inverter B interface
o_irq_invb_n	OUT	Inverter B fault interrupt to DSP (active low)
o_irq_invb_led	OUT	Inverter B fault LED control – lit when fault exists

Pin Name	Direction	Description
i_hb_err	IN	Error input (active high) from half bridge interface
i_hb_ovt	IN	Over temp fault (active high) from half bridge interface
o_irq_hb_n	OUT	Half Bridge fault interrupt to DSP (active low)
o_irq_hb_led	OUT	Half Bridge fault LED control – lit when fault exists
o_sys_led	OUT	System fault LED control – lit when fault exists
Quad DAC Interface:		
o_dac_clk	OUT	Serial clock to DAC
o_dac_cs_n	OUT	Chip select (active low) to DAC
o_dac_simo	OUT	Serial data to DAC
Relay Driver Interface:		
o_rly_clk	OUT	Serial clock to driver
o_rly_cs_n	OUT	Chip select to driver (active low)
o_rly_simo	OUT	Serial data out to driver
i_rly_somi	IN	Serial data in from driver
Real Time Clock Interface:		
o_rtc_clk	OUT	Serial clock to Real Time Clock IC
o_rtc_ce	OUT	Chip enable to Real Time Clock IC
io_rtc_dat	IN/OUT	Real Time Clock IC serial data
System Faults:		
i_rly_fault_n	IN	Fault indicator (active low) from Relay Driver
o_irq_sys_n	OUT	System interrupt to DSP (active low)
Miscellaneous Pins:		
o_amux_sel[3:0]	OUT	Controls select lines to the analog signal MUX
i_hw_rev[3:0]	IN	Hardwired PCB hardware revision level
i_wdog_ctrl	IN	Hardware jumper control for watchdog (1=enable, 0=disable)
o_wdog_dis	OUT	Watchdog disable
i_sci_de	IN	Serial Comm. I/F drive enable from DSP
o_rs485_re_n	OUT	Serial RS485 receive enable to transceiver

3. Logic Description

3.1 DSP Interface

The DSP Interface logic provides a read/write hardware interface to the TMS320F28335 DSP's data bus. All of the data and configuration registers implemented in this FPGA can be accessed via this interface as memory mapped I/O. This interface is designed to operate with the maximum DSP clock output of 150MHz. The FPGA requires specific read and write wait states to meet timing when using the maximum clock rate. This is achieved by adjusting the DSP's external memory interface timing registers just for the memory zone occupied by the FPGA.

This allows the other parallel devices on the bus to operate at their optimal timing without being affected by the FPGA timing requirements.

Lower-level blocks in this design may use the 'o_dsp_ready' FPGA output to dynamically extend the read or write time by inserting external wait states. Any interface that requires the use of additional wait states will document this in the respective sections of this document. The next two sections detail the write and read access timing required by the FPGA.

3.1.1 Writing to the FPGA

The figure below shows the functional waveforms required to write to the FPGA. As the figure shows, the write access takes 8 DSP clock cycles to complete.

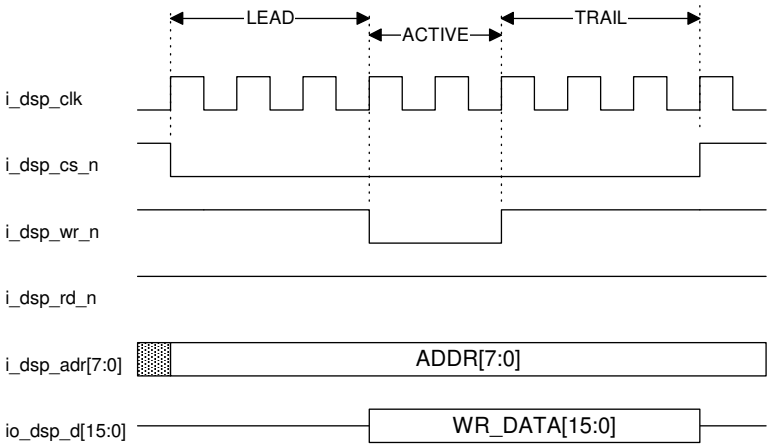


Figure 2 – Functional waveforms for writing to the FPGA

3.1.2 Reading from the FPGA

The figure below shows the functional waveforms required to read from the FPGA. As the figure shows, the read access takes 8 DSP clock cycles to complete.

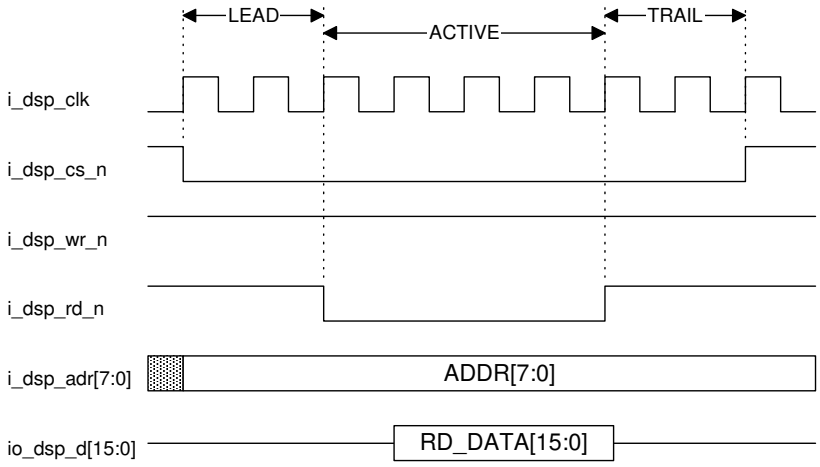


Figure 3 – Functional waveforms for reading from the FPGA

3.1.3 DSP Bus Timing Configuration Registers

The FPGA addressable memory space resides in Region 0 of the DPS's External Memory Interface (i.e. FPGA uses DSP CS0n). In order to generate the write and read timing and the corresponding Lead, Active, and Trail periods shown above, the DSP's timing registers should be programmed as follows:

- The XTIMING0 register should be set to 0x00036746 so that:
  - X2TIMING = 0 (1:1 timing for fastest clock rate)
  - XSIZE = 3 (for a 16-bit interface)
  - READYMODE = 0 (so READY input is synchronous)
  - USEREADY = 1 (so READY input may be used)
  - XRDLEAD = 2 (for a Read Lead period of 2 cycles)
  - XRDACTIVE = 3 (for a Read Active period of 4 cycles)
  - XRDTRAIL = 2 (for a Read Trail period of 2 cycles)
  - XWRLEAD = 3 (for a Write Lead period of 3 cycles)
  - XWRACTIVE = 1 (for a Write Active period of 2 cycles)
  - XWRTRAIL = 3 (for a Write Trail period of 3 cycles)

- Set the following fields in the XINTCNF2 register:
  - XTIMCLK = 0 (so XTIMCLK = SYSCLKOUT)
  - CLKOFF = 0 (so XCLKOUT is enabled)
  - CLKMODE = 0 (so XCLKOUT = XTIMCLK)

3.2 High Voltage ADC Interface

The FPGA provides an interface to an ADS7863 analog-to-digital converter. This is a six channel 12-bit 3x3 simultaneously sampled ADC device (simultaneously samples the first two inputs at the same time, then the next two, then the last two). This interface is used on the control

board for measuring the high-voltage grid phase voltages as well as the high-voltage output phase voltages.

Conversions are initiated by pulsing the 'i_adc_trig' high and then low from the DSP. Upon seeing a valid trigger, a state machine is kicked off that will drive the correct sequence of clock, convert, and SDI select signals to the ADC. The ADC has two serial outputs that are essentially clocked into separate 12-bit shift registers inside the FPGA. The state machine is used to determine when shifting occurs and when the shift register data is latched into the corresponding results registers. When the process is done, the state machine drives the ADC interrupt request line back to the DSP to indicate that the converted data is available for reading.

The ADC's maximum clock input rate is 32MHz. The FPGA will run the ADC clock at 6.25 MHz using a divided version of the 25MHz system clock. It takes approximately 48 clock cycles to convert and read all 6 channels (16 clocks to perform each of the three conversions). This gives a trigger to interrupt time of approximately 48 * (1/6.25MHz) = 7.68µs, for an effective maximum sampling rate of 130 kHz.

Note that there are digital isolator devices between the FPGA and the ADC (Analog Devices AduM1300). These devices come in a few versions to support various data rates (1Mbps, 10Mbps, and 100Mbps). Running the ADC clock at 6.25 MHz allows the use of the middle part. If the ADC clock speed is increased to 12.5 MHz or 25 MHz, the fastest part must be used. The 1 Mbps part cannot be used in either configuration.

3.3 Quad DAC Interface

The FPGA provides an interface to a TLV5620 octal digital-to-analog converter IC. This is used on the control board to provide four general purpose analog outputs. The DAC outputs are controlled by four separate registers within the FPGA (one data register for each of the DAC channels). Any given register is immediately loaded into the DAC whenever it is updated by the CPU.

Data is clocked into the DAC at an approximately 961.5 kHz rate (25MHz/26). It takes 12 clocks to update a single DAC channel, or 12.48µs. Software may quickly update more than one DAC register at a rate faster than this. If more than one register is updated, this controller will update the DAC in a round robin fashion (once finished updating the present channel the controller moves on to the next higher channel that is pending (looping from channel 3 back to 0).

It takes approximately 4 x 12.48µs or 49.92µs to update all four DAC channels, for an effective overall maximum update rate of roughly 20 kHz.

3.4 Real Time Clock Interface

The FPGA provides a serial interface to a DS1302 real time clock IC. This is a three-wire interface that consists of a clock and chip enable to the device and a bi-directional I/O line for writing and reading to the device. The internal interface to the RTC device consists of three registers: a control, write data, and read data register. The control register contains the address, read/write, and ram/calendar control lines that are shifted out to the device. Any time the control register is written the FPGA will automatically initiate a serial read or write to the device based on the read/write bit. For writes, the 8-bit data contained in the write data register will be shifted out to the device. For a read operation, the serial data from the RTC device is shifted in and latched in the read data register.

To satisfy the serial timing for the RTC IC, the serial interface is clocked at 0.5MHz. Both reads and writes consist of shifting 16 serial bits at this rate (2µs x 16). Also, the chip enable setup time is specified at 4µs, and also must be inactive for at least 4µs between operations. Given this, it takes approximately 40µs to complete a serial operation. The control register contains a busy bit that is set any time a serial operation is in progress. It is the responsibility of software to monitor this bit prior to attempting any new updates to the device; otherwise the current serial operation may be corrupted and unpredictable behavior may occur.

See the DS1302 data sheet for the details on the devices internal registers.

3.5 Faults and Interrupts

As the FPGA Pin Description table showed in the beginning of this document, there are several possible fault and hardware interrupt sources tied to the FPGA. These sources are tied to one of four possible interrupt lines to the DSP as follows:

- “o_irq_inva_n” conveys faults from the inverter “A” interface as follows:
  - A single over temperature fault (OVT)
  - 3 driver error faults (ERR_U, ERR_V, ERR_W)
- “o_irq_invb_n” conveys faults from the inverter “B” interface (similar faults as inverter “A”)
- “o_irq_hb_n” conveys faults from the half bridge interface as follows:
  - Over temperature fault (OVT)
  - Driver error fault (ERR)
- “o_irq_sys_n” conveys non-inverter based faults and interrupt sources as follows:
  - a relay driver channel fault
  - a general purpose input pin transition

For all fault/interrupt inputs, a simple digital filter is used to reduce the chance of mid to high frequency noise spikes from causing erroneous trips. The filter is a simple 3 flip-flop serial chain clocked at the system clock rate of 25MHz. A transition on the input will be considered valid if

the same value is sampled three times in a row (i.e. all three flip-flops are the same value and are the opposite from the current internal stored state for the input). This allows for filtering noise spikes up to 3 * 1/25MHz = 120ns wide, but also means that real faults will have a delay of 120ns before being considered valid.

When any of these faults/interrupts occur, they will be latched and stored in an internal status register that can be read by the DSP. For all interrupt outputs, each of the contributing fault inputs can be selectively enabled or disabled using an internal interrupt enable register bit.

These faults are also linked to four fault LEDs through the “o_irq_inva_led”, “o_irq_invb_led”, “o_irq_hb_led”, and “o_sys_led” pins. These pins will always be the complement of the interrupt pins to the DSP so each LED will be lit only when an interrupt is occurring. The “o_sys_led” signal can also be statically asserted by the control software to indicate any system-level faults that occurred within the application itself.

All four interrupt outputs are active low signals. The DSP should be configured for negative-edge triggered interrupts. Upon servicing one or more of the interrupt sources for any given interrupt line, software will read the associated interrupt status register to determine which source(s) caused the interrupt. At the very end of the service routine, the software should then clear the associated interrupt bits in the FPGA’s interrupt status register. Upon clearing one or more interrupt bits, the FPGA will automatically de-assert the interrupt output pin for a minimum of one 25MHz clock cycle (40ns). If additional interrupt sources are still active for the given interrupt output, the output pin will then be asserted again to cause another negative-edge interrupt at the DSP input.

3.6 Relay Drive Interface

The FPGA provides a serial interface to a TPIC44H01 FET driver. This is a quad driver that is used to provide four general purpose relay drive outputs from the control board. The interface to this device is controlled using two internal registers: an 8-bit configuration register and a 9-bit status register. Any time the configuration register is written the FPGA will automatically shift all 8 bits out to the device. At the same time, the FPGA will shift in the fault status from the device into the internal status register.

To satisfy the serial timing for the driver IC, the serial interface is clocked at 1.56MHz (25MHz/16). Given this, it takes approximately 8 x (1/1.56MHz) or about 5.1µs to complete a serial operation. The ninth bit of the status register is a busy bit that is set any time a serial operation is in progress. It is the responsibility of software to monitor this bit prior to attempting any new updates to the device; otherwise the current serial operation may be corrupted and unpredictable behavior may occur.

See the TPIC44H01 datasheet for specific definitions of the configuration and status bits and for details on the fault reading/clearing protocol.

3.7 General Purpose I/O

The FPGA provides several general purpose Digital I/O pins. There are four dedicated general purpose output pins and four dedicated input pins.

The dedicated general purpose inputs are sent through a digital filter prior to being driven to the register interface. This filter simply requires the input to stay in the same state for three consecutive 25 MHz clock cycles before it considers it a valid transition. This is solely meant to provide some basic high-frequency noise filtering. Any expected low-frequency noise or switch bounce should be additionally filtered by the DSP.

The four input pins are capable of interrupting the DSP upon either a rising or falling edge. This function can also be disabled individually for each input.

3.8 Watchdog Enable

The FPGA facilitates disabling the watchdog input to the power supply monitor IC on the control board. The watchdog can be disabled in one of two ways – either by using a hardware jumper on the board to force the “i_wdog_ctrl” input low or via software by use of the watchdog control register. Note that specific multi-bit values are required to be written to the control register to cause the watchdog to be enabled or disabled via software. The present state of the watchdog enable output pin can be read back using the watchdog control register.

3.9 Expansion Address Decoding

The FPGA facilitates addressing the two expansion modules (Anybus and Oztek Expansion) on the control board. The DSP provides a single expansion chip select along with a DSP address line (bit 12). The FPGA uses these two to determine which expansion module is being selected – it then drives the appropriate chip select line active for the selected expansion module.

The Anybus “busy” indicator is tied to the FPGA for possible future use – the present design does not use it.

3.10 Spare I/O

The FPGA provides six spare pins to the DSP for provisional future use. These six pins are configurable as either inputs or outputs from the FPGA using the internal register interface. The first four pins are connected to debug LEDs.

3.11 Programmable Pulse Generator

The FPGA provides a programmable pulse generator that can be driven out on the “io_spare[0]” pin. This pulse generator is intended to be used for creating an external ADC Start

of Conversion (SOC) pulse interrupt to the DSP. The pulse generator consists of a timer that counts at the 25MHz internal clock rate and a PERIOD register that defines the number of clocks between pulses. When enabled, the pulse output takes precedence over bit zero in the SPARES_DIR register. Care should be taken to insure that the DSP is configured to treat the "io_spare[0]" connection as an input and that the external daughter board interface is not driving this signal.

4. Internal Registers

The table below shows the registers that are implemented in the FPGA along with their corresponding addresses. Writes to unused addresses and to unused bits within the specified registers will be ignored. Data read from unused addresses and unused register bits will return all 0's. Each of the registers shown in the table below is then described in greater detail in the following sections. The usage for each field is described as either "R" for read-only, "W" for write-only, or "R/W" for read/write registers.

Table 2 – FPGA Internal Register Summary

Register Name	Address[7:0]
INVA_INT_STAT	0x00
INVA_INT_EN	0x01
INVB_INT_STAT	0x02
INVB_INT_EN	0x03
HB_INT_STAT	0x04
HB_INT_EN	0x05
SYS_INT_STAT	0x06
SYS_INT_EN	0x07
ISO_OUT	0x08
ISO_IN	0x09
ISO_IN_POL	0x0A
SPARES_DIR	0x0B
SPARES_IN	0x0C
SPARES_OUT	0x0D
HVADC_CTRL	0x0E
HVADC_HI_A	0x0F
HVADC_HI_B	0x10
HVADC_HI_C	0x11
HVADC_LO_A	0x12
HVADC_LO_B	0x13
HVADC_LO_C	0x14
RTC_CTRL	0x15

Register Name	Address[7:0]
RTC_WDATA	0x16
RTC_RDATA	0x17
RLY_CFG	0x18
RLY_STAT	0x19
WDOG_CTRL	0x1A
BRD_REV	0x1B
FPGA_REV	0x1C
DAC_CH0	0x1D
DAC_CH1	0x1E
DAC_CH2	0x1F
DAC_CH3	0x20
DAC_BUSY	0x21
UART_MODE	0x22
AMUX_CTRL	0x23
TEST_INPUTS	0x24
TEST_REG	0x25
FORCE_LEDS	0x26
FORCE_INTS	0x27
TEST_ADDR	0x28
ERR_PIN_POL	0x29
PULSE_CTRL	0x2A

4.1 INVA_INT_STAT: Address 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												ERR_U	ERR_V	ERR_W	OVT

This register is used to report the over temperature and error faults associated with the inverter "A" interface. When a given fault occurs, the corresponding bit in this register is set to a '1' and remains set until cleared. Writing a '0' to the corresponding location clears a fault bit. This register resets to 0x0. The individual faults are:

- **OVT** (R/W) Over Temperature: This bit is used to report an over temperature condition from the IGBT module.
- **ERR_U/V/W** (R/W) Inverter Error: These bits are used to report driver errors for each of the three individual phases on the IGBT module.

4.2 INVA_INT_EN: Address 0x01

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												ERR_U_EN	ERR_V_EN	ERR_W_EN	OVT_EN

This register is used to individually enable the various interrupt sources for the "o_irq_inva_n" interrupt to the DSP. There is an enable bit for each of the faults associated with INVA_INT_STAT register. Setting an enable bit to a '1' allows that fault to cause an interrupt to the DSP. Setting enable bits to a '0' disables particular faults from creating interrupts. Note that the fault status register will continue to latch and report faults even if particular bits are not enabled to generate an interrupt. All enable bits are (R/W). This register resets to 0x0.

4.3 INVB_INT_STAT: Address 0x02

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												ERR_U	ERR_V	ERR_W	OVT

This register is used to report the over temperature and error faults associated with the inverter “B” interface. When a given fault occurs, the corresponding bit in this register is set to a ‘1’ and remains set until cleared. Writing a ‘0’ to the corresponding location clears a fault bit. This register resets to 0x0. The individual faults are:

- **OVT** (R/W) Over Temperature: This bit is used to report an over temperature condition from the IGBT module.
- **ERR_U/V/W** (R/W) Inverter Error: These bits are used to report driver errors for each of the three individual phases on the IGBT module.

4.4 INVB_INT_EN: Address 0x03

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												ERR_U_EN	ERR_V_EN	ERR_W_EN	OVT_EN

This register is used to individually enable the various interrupt sources for the “o_irq_invb_n” interrupt to the DSP. There is an enable bit for each of the faults associated with INVB_INT_STAT register. Setting an enable bit to a ‘1’ allows that fault to cause an interrupt to the DSP. Setting enable bits to a ‘0’ disables particular faults from creating interrupts. Note that the fault status register will continue to latch and report faults even if particular bits are not enabled to generate an interrupt. All enable bits are (R/W). This register resets to 0x0.

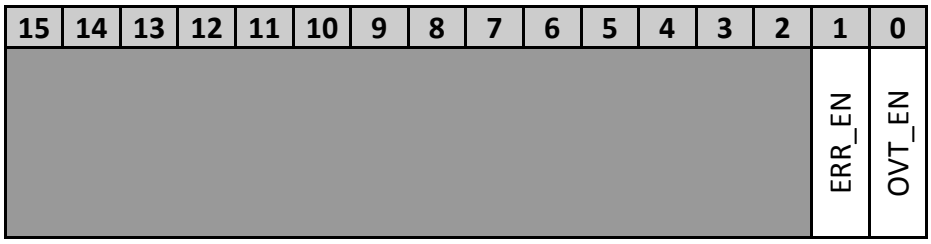
4.5 HB_INT_STAT: Address 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
														ERR	OVT

This register is used to report the over temperature and error faults associated with the inverter “B” interface. When a given fault occurs, the corresponding bit in this register is set to a ‘1’ and remains set until cleared. Writing a ‘0’ to the corresponding location clears a fault bit. This register resets to 0x0. The individual faults are:

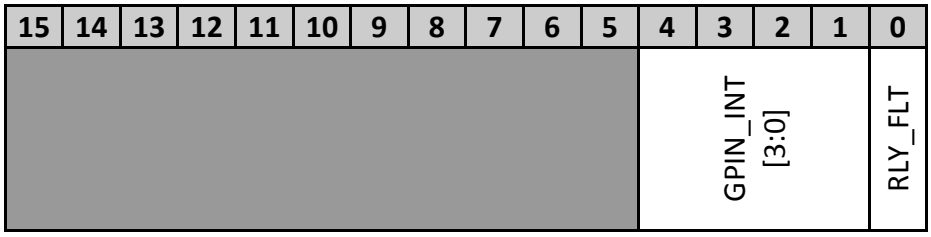
- **OVT** (R/W) Over Temperature: This bit is used to report an over temperature condition from the IGBT module.
- **ERR** (R/W) Inverter Error: This bit is used to report driver errors on the IGBT module.

4.6 HB_INT_EN: Address 0x05



This register is used to individually enable the various interrupt sources for the “o_irq_hb_n” interrupt to the DSP. There is an enable bit for each of the faults associated with HB_INT_STAT register. Setting an enable bit to a ‘1’ allows that fault to cause an interrupt to the DSP. Setting enable bits to a ‘0’ disables particular faults from creating interrupts. Note that the fault status register will continue to latch and report faults even if particular bits are not enabled to generate an interrupt. All enable bits are (R/W). This register resets to 0x0.

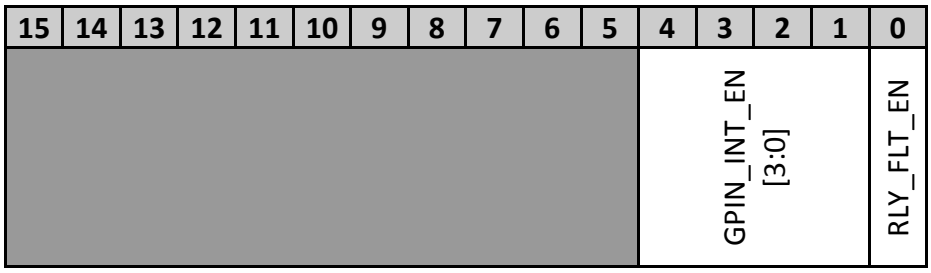
4.7 SYS_INT_STAT: Address 0x06



This register is used to report non-inverter based faults and interrupt sources. When a given fault or interrupt source occurs, the corresponding bit in this register is set to a ‘1’ and remains set until cleared. Writing a ‘0’ to the corresponding location clears a bit. This register resets to 0x0. The individual faults are:

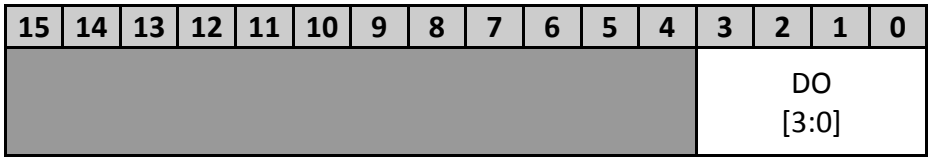
- **RLY_FLT** (R/W) Relay Driver Fault: This bit is used to report a fault condition from the relay driver IC. See RLY_STAT for fault details from the driver IC.
- **GPIN_INT [3:0]** (R/W) General Purpose Input Interrupts: These bits are used to report edge-sensitive interrupts generated from the dedicated general purpose inputs. See the GPIN_INT_POL register to determine whether a rising or falling edge causes an interrupt.

4.8 SYS_INT_EN: Address 0x07



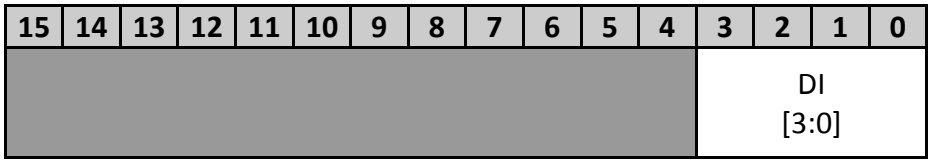
This register is used to individually enable the various interrupt sources for the "o_irq_sys_n" interrupt to the DSP. There is an enable bit for each of the bits associated with SYS_INT_STAT register. Setting an enable bit to a '1' allows that source to cause an interrupt to the DSP. Setting enable bits to a '0' disables particular sources from creating interrupts. Note that the status register will continue to latch and report faults even if particular bits are not enabled to generate an interrupt. All enable bits are (R/W). This register resets to 0x0.

4.9 ISO_OUT: Address 0x08



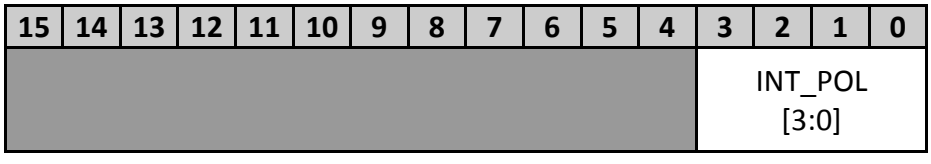
- **DO [3:0]** (R/W) Output Data For "o_iso_out[3:0]": The "o_iso_out[3:0]" dedicated output pins will be driven with the data contained in these bits. These bits reset to 0x0.

4.10 ISO_IN: Address 0x09



- **DI [3:0]** (R) Input Data From "i_iso_in[3:0]": These bits will return the present state of the "i_iso_in[3:0]" dedicated input pins.

4.11 ISO_IN_POL: Address 0x0A



- **INT_POL [3:0]** (R/W) Interrupt Polarity for Isolated General Purpose Inputs: These bits determine which edge of the dedicated input pins will be used for generating a system interrupt. If a bit is set to a 0, the corresponding "i_iso_in[n]" will generate an interrupt on a low-to-high transition. If a bit is set to a 1, the corresponding input pin will generate an interrupt on a high-to-low transition. Note that the SYS_INT_EN register is used to determine if an interrupt is actually enabled for each input pin. These bits reset to 0x0.

4.12 SPARES_DIR: Address 0x0B

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
										SPARE_DIR [5:0]					

- **SPARE_DIR [5:0]** (R/W) Spare Debug Pin Direction: These bits are used to set the direction for the six spare pins. Bits set to '1' indicate those pins that are outputs from the FPGA. Bits set to '0' indicate those pins that are inputs to the FPGA. The SPARES_IN and SPARES_OUT registers are used for reading the input state and setting the output states for the various pins. This field resets to 0x0.

4.13 SPARES_IN: Address 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
										SPARE_DIN [5:0]					

- **SPARE_DIN [5:0]** (R) Spare Debug Pin State: These read-only bits report the present state of the six spare pins (regardless of whether the pins are configured as inputs or outputs).

4.14 SPARES_OUT: Address 0x0D

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
										SPARE_DOUT [5:0]					

- **SPARE_DOUT [5:0]** (R/W) Spare Debug Pin Output Data: These bits contain the data to be driven out on the spare pins when they are configured as outputs (see SPARES_DIR register). This field resets to 0x0.

4.15 HVADC_CTRL: Address 0x0E

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
														BUSY	INT_EN

- **INT_EN** (R/W) Interrupt Enable: When this bit is set to '1', ADC interrupts are enabled and the "o_hvadc_irq_n" output will be toggled low and high again at the end of a conversion. When '0', ADC interrupts are disabled. This bit resets to 1.
- **BUSY** (R) Conversion Busy: This bit is set to '1' any time an ADC conversion is in progress (i.e. the time between receiving a trigger and when the 6 result registers are updated). This bit is '0' when the ADC interface is not in use.

4.16 HVADC_HI_A: Address 0x0F

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				HI_A [11:0]											

- **HI_A [11:0]** (R) Line Voltage (A to Neutral): This field is updated at the end of an ADC conversion with the line-to-neutral voltage for phase A on the high-voltage input (the circuit with the on-board high-voltage resistor divider).

4.17 HVADC_HI_B: Address 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				HI_B [11:0]											

- **HI_B [11:0]** (R) Line Voltage (B to Neutral): This field is updated at the end of an ADC conversion with the line-to-neutral voltage for phase B on the high-voltage input (the circuit with the on-board high-voltage resistor divider).

4.18 HVADC_HI_C: Address 0x11

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				HI_C [11:0]											

- **HI_C [11:0]** (R) Line Voltage (C to Neutral): This field is updated at the end of an ADC conversion with the line-to-neutral voltage for phase C on the high-voltage input (the circuit with the on-board high-voltage resistor divider).

4.19 HVADC_LO_A: Address 0x12

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				LO_A [11:0]											

- **LO_A [11:0]** (R) Line Voltage (A to Neutral): This field is updated at the end of an ADC conversion with the line-to-neutral voltage for phase A on the low-voltage input (the circuit with the off-board high-voltage resistor divider/buffer).

4.20 HVADC_LO_B: Address 0x13

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				LO_B [11:0]											

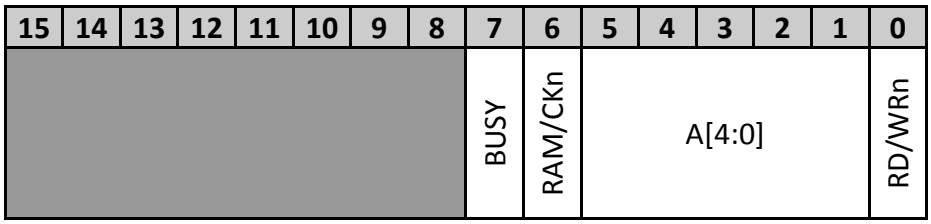
- **LO_B [11:0]** (R) Line Voltage (B to Neutral): This field is updated at the end of an ADC conversion with the line-to-neutral voltage for phase B on the low-voltage input (the circuit with the off-board high-voltage resistor divider/buffer).

4.21 HVADC_LO_C: Address 0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				LO_C [11:0]											

- **LO_C [11:0]** (R) Line Voltage (C to Neutral): This field is updated at the end of an ADC conversion with the line-to-neutral voltage for phase C on the low-voltage input (the circuit with the off-board high-voltage resistor divider/buffer).

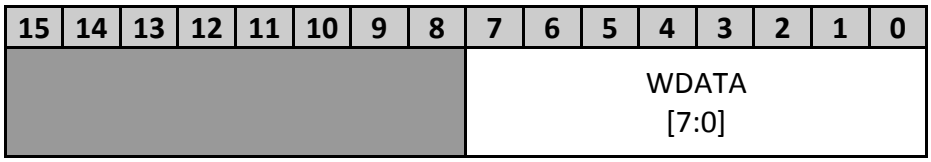
4.22 RTC_CTRL: Address 0x15



The lower 7 bits of this register correspond to the same bits in the Real Time Clock IC (DS1302) command byte. Writes to this register automatically kick off a read or write operation to the RTC device depending on the setting of the RD/WRn bit. For writes, the data stored in the RTC_WDATA register is written to the RTC device. For reads, the return value from the RTC device is stored in the RTC_RDATA register at the end of the read operation. Software *must* monitor the BUSY bit to determine if a new operation is allowed. The RTC_CTRL register must not be updated while the BUSY bit is set; otherwise unpredictable behavior may result. Similarly, for write operations in progress, the RTC_WDATA register must not be updated while BUSY is set. For read operations, the RTC_RDATA register data is invalid while the BUSY bit is set.

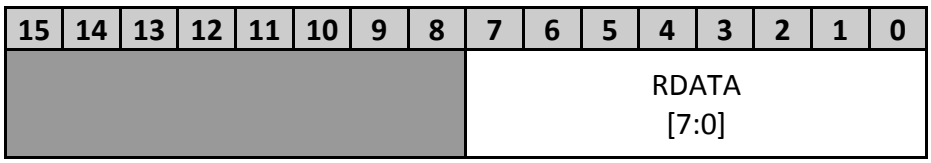
- **RD/WRn** (R/W) Read/Write Command Bit. This bit resets to 0.
- **A [4:0]** (R/W) Device Address. This field resets to 0x00.
- **RAM/CKn** (R/W) RAM/Clock Memory Space Select This bit resets to 0.
- **BUSY** (R) RTC Read or Write in Progress. This read-only bit is set to a 1 any time a read or write operation is in progress. This bit will return a 0 if the interface to the RTC device is not in use.

4.23 RTC_WDATA: Address 0x16



- **WDATA [7:0]** (R/W) RTC Write Data. The data stored in this register is written to the RTC device when a write is initiated (see RTC_CTRL for details). This field resets to 0x00.

4.24 RTC_RDATA: Address 0x17



- **RDATA [7:0]** (R) RTC Read Data. This register stores read data gathered from the RTC device at the completion of a read operation (see RTC_CTRL for details).

4.25 RLY_CFG: Address 0x18

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								SLEEP	DG [3:1]			IN [4:1]			

The 8 bits in this register correspond to the same bits in the relay drive IC (TPIC44H01) command byte. Writes to this register automatically kick off a serial update of the drive IC – the bits in this register are shifted out to the device while at the same time the fault status from the device is shifted into the RLY_STAT register. Software *must* monitor the BUSY bit in the RLY_STAT register to determine if a new operation is allowed. This register must not be updated while the BUSY bit is set in RLY_STAT; otherwise unpredictable behavior may result.

- **IN [4:1]** (R/W) Relay Output Control. Setting a bit to a 0 turns off the corresponding relay drive output. Setting a bit to a 1 will turn on the selected output. This field resets to 0x00.
- **DG [3:1]** (R/W) Deglitch Timer Control. See the TPIC44H01 datasheet for details. This field resets to 0x00.
- **SLEEP** (R/W) Sleep Mode. Setting this bit to a 1 enables sleep mode in the TPIC44H01, setting this bit to 0 will maintain normal operation. This bit resets to 0.

4.26 RLY_STAT: Address 0x19

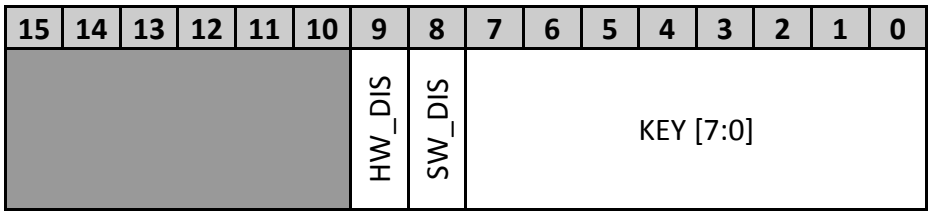
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							BUSY	F4 [1:0]		F3 [1:0]		F2 [1:0]		F1 [1:0]	

- **F*n* [1:0]** (R) Relay Drive Faults. There are four sets of 2-bit fault feedback (one for each output channel) encoded as follows (see TPIC44H01 datasheet for principle of operation):
  - **00 :** Over Voltage
  - **01 :** Open load
  - **10 :** Over Current
  - **11 :** Short to Ground

- **BUSY** (R) Relay Update in Progress. This read-only bit is set to a 1 any time a serial operation is in progress. The F*n* [1:0] fields are not valid while the BUSY bit is set. This bit will return a 0 if the interface to the relay device is not in use.

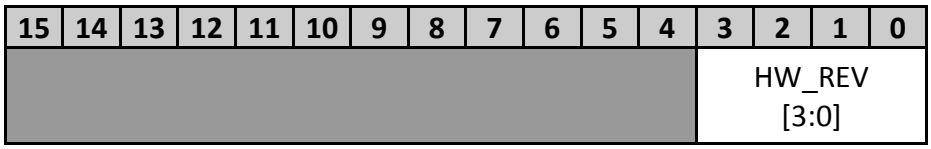
It is important to note that the RLY_STAT register is only updated when a write to the RLY_CFG register is performed; this register contains the data sent back from the relay driver following a write. The RLY_FLT bit of SYS_INT_STAT will indicate when a relay fault has occurred. A write to the RLY_CFG register should then be performed in order to update this register, which will clear the faults from the relay driver. Please see the TPIC44H01 datasheet for more information on this behavior.

4.27 WDOG_CTRL: Address 0x1A



- **KEY [7:0]** (R/W) Software-controlled Watchdog Enable Key. This field is used to enable or disable the external watchdog timer. Writing a value of 0xAA to this field will disable the external watchdog timer. Writing a value of 0x55 to this field will clear the S/W controlled disable bit. This field resets to 0xAA.
- **SW_DIS** (R) Software Watchdog Disable Status. This read-only bit indicates the present state of the software-controlled disable bit. A value of 0 indicates that software is not disabling the watchdog timer. A value of 1 indicates that the software disable function is active and the watchdog timer is disabled.
- **HW_DIS** (R) Hardware Watchdog Disable Status. This read-only bit indicates the present state of the hardware-controlled disable bit. A value of 0 indicates that hardware is not disabling the watchdog timer (i.e. jumper is *not* installed on the board). A value of 1 indicates that the hardware disable function is active and the watchdog timer is disabled (i.e. jumper *is* installed).

4.28 BRD_REV: Address 0x1B



- **HW_REV [3:0]** (R) This read-only register is used to report the current control board design revision level as specified by the pull-up/pull-down resistors populated on the PCB.

4.29 FPGA_REV: Address 0x1C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAJ_REV [7:0]								MIN_REV [7:0]							

- **MAJ/MIN REV[7:0]** (R) This read-only register is used to report the current FPGA design revision level. Revisions are reported as <major>.<minor>; with the major level indicating changes to the overall functionality of the FPGA (or major overhauls) and the minor level reflecting small changes or bug fixes to the existing features. The initial design revision will be 1.0.

4.30 DAC_CH0 – DAC_CH3: Addresses 0x1D – 0x20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								DAC_DATA [7:0]							

- **DAC_DATA [7:0]** (R/W) DAC Output Data: These registers are used to specify the output voltage for the corresponding channel on the quad DAC (TLV5620). Writing to a particular register will automatically initiate an update to the DAC. Multiple DAC registers may be updated by software back-to-back – the FPGA will update the DAC channels in a round-robin fashion starting from the current channel being written and followed by any higher channel that is pending (looping from channel 3 back to 0). Care should be taken to not overwrite any registers that may be pending a DAC update (use the DAC_BUSY register to determine if there are any channels pending). These registers reset to 0x00. The DAC outputs power-up to ~0V.

4.31 DAC_BUSY: Address 0x21

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												BUSY [3:0]			

- **BUSY [3:0]** (R) DAC Busy Indicators: These bits are used to report any DAC_CH*n* registers that have been updated by software but not yet updated in the DAC. Any bit that reads a 1 indicates that the corresponding channel is pending a DAC update. Any bit that reads a 0 indicates that the DAC output has been updated to the corresponding value specified in the DAC_CH*n* register.

4.32 UART_MODE: Address 0x22

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
															MODE

- **MODE (**R/W) Serial Comm. Mode: There are locations for two different serial transceivers on the control board – one supports RS232 and one supports RS422/485. Only one will be soldered on as the board only provides a single serial port and connector. When the board contains the RS485/RS422 transceiver, this register is used to select which protocol is used as follows: **1 =** RS422, **0** = RS485. When the board contains the RS232 transceiver, this register has no affect on the serial port operation. This register resets to 0x0.

4.33 AMUX_CTRL: Address 0x23

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												AMUX_SEL [3:0]			

- **AMUX_SEL [3:0]** (R/W) Analog Mux Control: These bits are used to select a channel for the analog mux. This field resets to 0x00. The analog channels are selected as follows:
  - **0x0:** Inverter A temperature
  - **0x1:** Inverter B temperature
  - **0x2:** Half bridge temperature
  - **0x3:** Thermistor A auxiliary temperature
  - **0x4:** Thermistor B auxiliary temperature
  - **0x5:** Thermistor C auxiliary temperature
  - **0x6:** Voltage monitoring of 24V (Vsense = Vin*0.09091)
  - **0x7:** Voltage monitoring of 15V (Vsense = Vin*0.16701)
  - **0x8:** Voltage monitoring of 5V (Vsense = Vin*0.50000)
  - **0x9:** Voltage monitoring of 3.3V (Vsense = Vin*0.75518)
  - **0xA:** Voltage monitoring of -15V (Vsense = Vin*0.04649 + 3.14658)
  - **0xB:** Test point
  - **0xC:** No connection
  - **0xD:** No connection
  - **0xE:** No connection
  - **0xF:** No connection

4.34 TEST_INPUTS: Address 0x24

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		SCI_DE	RLY_FLT_N	HB_OVT	HB_ERR	INVB_OVT	INVB_ERR_W	INVB_ERR_V	INVB_ERR_U	INVA_OVT	INVA_ERR_W	INVA_ERR_V	INVA_ERR_U	HVAD_BUSY	ANYB_BUSY

This read-only (R) register is provided solely for board diagnostic purposes. It provides direct connections to the input pins listed in the bit map table above.

4.35 TEST_REG: Address 0x25

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEST_DATA [15:0]															

- **TEST_DATA [15:0]** (R/W) Test Data: This generic 16-bit test register is provided for diagnostic purposes only. It does not connect to any other functional interface within the FPGA. This field resets to 0x00.

4.36 FORCE_LEDS: Address 0x26

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												SYS_LED	HB_LED	INVB_LED	INVA_LED

This RW register is provided for board diagnostic purposes as well as a means for the software application to force a particular LED to be lit. Writing a '1' to any of these bits will force the corresponding LED to be lit. Setting a bit to '0' will remove the software-forced condition, and the LED will otherwise be driven by the corresponding INT_STAT and INT_EN registers. This register resets to 0x00.

- **INVA_LED** (R/W) Inverter A LED
- **INVB_LED** (R/W) Inverter B LED
- **HB_LED** (R/W) Half-bridge LED
- **SYS_LED** (R/W) System Fault LED

4.37 FORCE_INTS: Address 0x27

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
										FORCE_EN	DSP_READY	SYS_INT	HB_INT	INVB_INT	INVA_INT

This RW register is provided for board diagnostic purposes only. Writing a '1' the FORCE_EN bit will force the values in bits [4:0] to be applied to the corresponding FPGA output pins. Writing a '0' to the FORCE_EN bit disables the forcing function and the pins listed above will not be affected by this register. This register resets to 0x00.

4.38 TEST_ADDR: Address 0x28

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								DSP_ADDR [7:0]							

This read-only (R) register is provided for board diagnostic purposes only. Any time the DSP performs a write to the FPGA (i.e. the FPGA chip select and the DSP write signals are asserted), the value of the address pins are written to this register. This allows for testing of unused address inputs.

4.39 ERR_PIN_POL: Address 0x29

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						HB_OVT_POL	HB_ERR_POL	INVB_OVT_POL	INVB_ERR_W_POL	INVB_ERR_V_POL	INVB_ERR_U_POL	INVA_OVT_POL	INVA_ERR_W_POL	INVA_ERR_V_POL	INVA_ERR_U_POL

- ***_POL** (R/W) Input Error Pin Polarity: These bits are used to determine the logic level on the error input pins that equates to an error condition. Each bit in this register corresponds to the phase and temperature error pins reported by the Semikron module interface. Setting a bit to a '1' indicates that the error input is active low (low = error, high = OK). Setting a bit to a '0' indicates that the error input is active high (low = OK, high = error). This register resets to 0x0000.

4.40 PULSE_CTRL: Address 0x2A

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN	PRD[14:0]														

- **PRD [14:0]** (R/W) Programmable Pulse Period: These bits are used to define the period count used to generate the programmable output pulses. An internal free running timer will count from zero up to this value at the internal 25MHz clock rate. Once the timer reaches this value, it is reset to zero and will continues to count up. This field resets to 0x0000.
- **EN** (R/W) Programmable Pulse Enable: This bit is used to enable the pulse output. When set to '0', the pulse output is disabled and the "io_spare[0]" pin operates as a spare pin. When set to a '1', the pulse output is driven out on the "io_spare[0]" pin (this feature takes precedence over the SPARE_DIR bit 0 setting). This bit resets to 0.

5. Physical Information

As was mentioned in the Overview section, this FPGA is implemented using a Lattice MachXO LCMXO1200C device in a 144-pin QPF package. Unless otherwise specified, all I/O are +3.3V LVCMOS with a nominal drive strength of +/- 12mA. The device pinout is given in the table below.

Table 3 – FPGA Pinout

Pin	Pin Name
1	io_spare_0
2	io_spare_1
3	io_spare_2
4	io_spare_3
5	io_spare_4
6	io_spare_5
7	N/C
8	N/C
9	N/C
10	VCCIO7
11	GNDIO7
12	N/C
13	N/C
14	i_reset_n
15	N/C
16	GND1
17	i_anyb_busy_n
18	N/C
19	o_amux_sel_0
20	o_amux_sel_1
21	VCC1
22	o_amux_sel_2
23	o_amux_sel_3
24	N/C
25	o_sys_led
26	VCCIO6
27	GNDIO6
28	N/C
29	N/C
30	o_hvadc_clk
31	o_hvadc_conv
32	o_hvadc_sdi
33	i_hvadc_sdat_a
34	i_hvadc_sdat_b
35	i_hvadc_busy
36	N/C
37	GNDIO5
38	VCCIO5
39	TMS
40	o_iso_out_0

Pin	Pin Name
41	o_iso_out_1
42	TCK
43	o_iso_out_2
44	o_iso_out_3
45	i_sci_de
46	o_rs485_re_n
47	TDO
48	o_rtc_clk
49	io_rtc_dat
50	o_rtc_ce
51	TDI
52	VCC2
53	VCCAUX1
54	N/C
55	o_dac_clk
56	o_dac_simo
57	o_dac_cs_n
58	N/C
59	GND2
60	o_rly_clk
61	o_rly_simo
62	o_rly_cs_n
63	VCCIO4
64	GNDIO4
65	i_rly_somi
66	i_rly_fault_n
67	N/C
68	i_iso_in_0
69	i_iso_in_1
70	SLEEP
71	i_iso_in_2
72	i_iso_in_3
73	i_inva_ovt
74	i_inva_err_u
75	i_inva_err_v
76	i_inva_err_w
77	i_invb_ovt
78	i_invb_err_u
79	i_invb_err_v
80	i_invb_err_w

Pin	Pin Name
81	i_expn_cs_n
82	VCCIO3
83	GNDIO3
84	o_anyb_cs_n
85	o_ozex_cs_n
86	i_hw_rev_0
87	i_hw_rev_1
88	GND3
89	i_hw_rev_2
90	i_hw_rev_3
91	o_dsp_ready
92	i_hvadc_trig
93	VCC3
94	i_dsp_cs_n
95	i_dsp_wr_n
96	i_dsp_rd_n
97	o_irq_inva_n
98	VCCIO2
99	GNDIO2
100	o_irq_invb_n
101	o_irq_sys_n
102	o_hvadc_irq_n
103	o_irq_hb_n
104	o_irq_inva_led
105	o_irq_invb_led
106	i_hb_err
107	o_irq_hb_led
108	i_hb_ovt
109	io_dsp_d_13
110	io_dsp_d_14
111	io_dsp_d_15
112	i_dsp_addr_0
113	i_dsp_addr_1
114	i_dsp_addr_2
115	i_dsp_addr_3
116	i_dsp_addr_4
117	VCCIO1
118	GNDIO1
119	i_dsp_addr_5
120	i_dsp_addr_6

Pin	Pin Name
121	i_dsp_addr_7
122	i_dsp_addr_12
123	GND4
124	i_dsp_clk
125	i_wdog_ctrl
126	o_wdog_dis
127	i_sys_clk
128	VCCAUX2
129	VCC4
130	io_dsp_d_0
131	io_dsp_d_1
132	io_dsp_d_2
133	io_dsp_d_3
134	io_dsp_d_4
135	VCCIO0
136	GNDIO0
137	io_dsp_d_5
138	io_dsp_d_6
139	io_dsp_d_7
140	io_dsp_d_8
141	io_dsp_d_9
142	io_dsp_d_10
143	io_dsp_d_11
144	io_dsp_d_12

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@oztekcorp.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.