|  |  |  |  |
| --- | --- | --- | --- |
|  | **Doc** **Number:** P1060972-ATP | | |
| **Revision:** 1.0 | | **Date:** 12-31-2024 |
| **Original** **Issue:** 12-31-2024 | | |
| **Author:** Steven Zhang | |  |
| **Page:** 1 | | **Of:** **27** |
| **ACCEPTANCE** **TEST** **PROCEDURE:** DUAL MOTOR DRIVER, Assembly # P1060972-XXX | | | |
| **Customer:** Varian Medical Systems | |  |  |
| **Approval:** |  | **Title:** |  |

*Content:* *This* *document* *describes* *the* *procedure* *for* *functional* *testing* *the* *Dual* *Motor* *Driver* *Assy.,* *P/N* *P1060972-XXX.*

**REVISION** **HISTORY**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | ***REVISION*** ***HISTORY*** | |  |
|  | **Rev**. | **Date** | **Originator** | **Description** |  |
|  | 1.0 | 12/31/2024 | Steven Zhang | Original Draft – Prepared from 100046950 ATP, submitted to VMS for approval |  |
|  |  |  |  |  |  |

**TABLE** **OF** **CONTENTS**

**Section/Title Page**

**1** **PURPOSE 3**

**2** **SCOPE 3**

**3** **REFERENCE DOCUMENTS 3**

**4** **DEFINITIONS and CONVENTIONS 3**

4.1 Definitions 3

**5** **EQUIPMENT REQUIRED 4**

5.1 Test Equipment 4

5.2 Test Software/Firmware 4

**6** **PRE-TEST PROCEDURES 4**

6.1 Visual Inspection 4

**7** **TEST SETUP 4**

7.1 ATE/Fixture Connections and Software Loading - Performed once for each lot. 4

7.2 UUT/Fixture Connections - Performed for each board. 5

7.3 Programming Test Firmware 5

**8** **TEST PROCEDURE 5**

8.1 Initial Power-Up Test 5

8.2 SPD DMD Interface Test 10

8.3 SPD EMOPS Interface Test 11

8.4 Stand Controller Interface Test 13

8.5 Gantry Brake And Motor Hall Sensor If Test 13

8.6 CCHL Interface Test 14

8.7 Lift Motor Hall Sensor If Test 15

8.8 Service Pendant Test 16

8.9 Gantry 96V Motor Test 16

8.10 lift 96V Motor Test 21

8.11 Program Customer Firmware 24

8.12 Power Down 24

# **1 PURPOSE**

This document provides step-by-step functional testing methods for automatic testing of the Dual Motor Driver PCB Assembly on VMS SOCRATES-1 Automatic Test Equipment Rack located at the Tempe, Arizona plant. The following tests and inspections are performed:

* INITIAL POWER-UP TEST
* SPD DMD INTERFACE TEST
* SPD EMOPS INTERFACE TEST
* STAND CONTROLLER INTERFACE TEST
* CCHL INTERFACE TEST
* LIFT MOTOR HALL SENSOR IF TEST
* SERVICE PENDANT TEST
* GANTRY 96V MOTOR TEST
* LIFT 96V MOTOR TEST
* PROGRAM CUSTOMER FIRMWARE (TBD)

# **2 SCOPE**

This document constitutes the functional test plan for the Dual Motor Driver PCB Assy., P/N P1060972. This document describes the testing procedures in a high-level format without making detailed references to the test software, instrument configuration or ATE stimulus and response signal routing. Units that meet all of the requirements of this document shall be accepted as functional.

# **3 REFERENCE DOCUMENTS**

* Schematics, P1060973
* Schematic, Test Fixture- P/N P1060972-FX
* VMS SOCRATES-1 Test System Documentation

# **4 DEFINITIONS and CONVENTIONS**

## ***4.1 Definitions***

ATE Automatic Test Equipment

ESD Electro-Static Discharge

GND Ground/UUT 0V Reference

GPIB General Purpose Instrument Bus

KHz kilohertz

MHz megahertz

MSB Most Significant Bit   
LSB Least Significant Bit   
P/N Part number

PCB Printed Circuit Board   
UUT Unit under Test

VDC Volts Direct Current

VAC Volts Alternating Current

GP General Purpose

DIO Digital Input / Output

HIGH TTL Logic State ONE

LOW TTL Logic State ZERO

# **5 EQUIPMENT REQUIRED**

The following list contains the required test equipment for functional testing of the Dual Motor Driver PCB assembly on the VMS SOCRATES-1 Test System ATE.

## ***5.1 Test Equipment***

### 5.1.1 AG34401 DMM

### 5.1.2 N6701A Power Supply Main Frame

### 5.1.3 N6733B 20V @ 2.5A DC#2 Power Supply

### 5.1.4 N6674A 60V @ 35A DC#5 Power Supply

### 5.1.5 N6674A 60V @ 35A DC#6 Power Supply

### 5.1.6 N6769A 100V @ 15A DC#7 Power Supply

### 5.1.7 DSO6014L 100MHz Scope

### 5.1.8 PXI-1042 8-Slot PXI Chassis

### 5.1.9 PXI-8360 MXI-Express

### 5.1.10 PXI-6704 16 Voltage AO/16 Current AO; 8-DIO

### 5.1.11 PXI-6509 96 Channel DIO

### 5.1.12 PXI-2527 32-Channel MUX

### 5.1.13 PXI2566 16-Channel High Current Relay

## ***5.2 Test Software/Firmware***

### 5.2.1 Test Software; P/N P1060972-TSW Revision 1.0 developed with Microsoft Visual Studio .NET 2010.

### 5.2.2 Test Firmware; P/N DSP.HEX Revision 1.0 developed with CCStudio\_V3.1 tool chain for C2000 DSP.

### 5.2.3 Test Firmware; P/N P1060972\_FPGA-TFW.PDB Revision 1.0 developed with ACTEL LIBERO IDE v9.1.

# **6 PRE-TEST PROCEDURES**

***Warning: ESD Sensitive devices present -*** Ensure proper grounding procedures are followed while handling/testing this and all ESD sensitive assemblies as defined in IPC-A-610.

## ***6.1 Visual Inspection***

### 6.1.1 Visually inspect the PCB to be tested for obvious signs of wrong, missing or improperly oriented parts (Refer to Assembly Drawing).

### 6.1.2 Also inspect for signs of contamination and poor workmanship including soldering defects (bridges, splashes, balls, unsoldered pins, flux build-up, etc.) and improper mounting of parts. Pay specific attention to ensure that through-hole parts and sockets are mounted flush to the board.

### 6.1.3 Inspect polarity on all aluminum and tantalum capacitors.

### 6.1.4 Any discrepancies must be corrected before proceeding.

# **7 TEST SETUP**

## ***7.1 ATE/Fixture Connections and Software Loading- Performed once for each lot.***

### 7.1.1 Ensure that all instruments on the VMS SOCRATES-1 ATE Test Rack have been reset.

### 7.1.2 Ensure that the engaging handle on the ATE Interface is in the up (disengaged) position.

### 7.1.3 Place the Test Fixture, P/N P1060972-FX, on the ATE interface with the fixture hinges facing the instruments.

### 7.1.4 Once the fixture is properly seated push the interface handle down to engage the fixture.

### 7.1.5 On the VMS SOCRATES-1 ATE Test Computer, load the VMS test program P1060972-TSW by clicking on the START button on the taskbar. Select the Production folder to bring up all available test sub-folders. Select the VMS sub-folder from the list and click on the P1060972-TSW icon to launch the test program.

## ***7.2 UUT/Fixture Connections - Performed for each board.***

### 7.2.1 Place the UUT on to the test fixture such that on the UUT is closest to the front of test fixture.

### 7.2.2 On the Bottom side of the UUT, connect cable assemblies J1, J5 to the UUT connectors J1, and J5 respectively.

### 7.2.3 On the Top side of the UUT, connect the ACTEL Programming Emulator to UUT connector J11.

### 7.2.4 On the Top side of the UUT mate the DSP Board to UUT connector J26.

### 7.2.5 On the Top side of the UUT, connect cable assemblies J18, J17, J24, J8, J12, J22, J23, J6, J10, J9, and J7 to the UUT connectors J18, J17, J24, J8, J12, J22, J23, J6, J10, J9, and J7 respectively.

## **7.3 Work Instructions.**

### 7.3.1 Click on the Instruction tab on the Test Platform to access to the Work Instructions.

# **8 TEST PROCEDURE**

The following procedure contains step-by-step instructions for testing the UUT. Although this test was developed exclusively to be performed on a specific ATE, it does not detail the specific ATE related operations such as signal routing. This information could be gathered by inspection of the test software and fixture drawing.

The test is performed with the utilization of a test firmware application developed specifically to implement the test steps described below. This test firmware application must reside inside the DSP before the test can be executed. The test firmware will work in conjunction with the test program executing on the ATE test computer. The two applications communicate with each other over the Bus Interface established between the UUT and the DSP board.

Click on the START TEST command button on the test program GUI to start the test.

## ***8.1 Initial Power-up Test***

TEST DESCRIPTION:

The power input to the UUT will be applying +24Vdc between J18-1, 4 and J18-3, 6. While power is ramping up the current draw of the UUT will be monitored for any spikes. Once power to the UUT has been established, the following group tests will be accomplished.

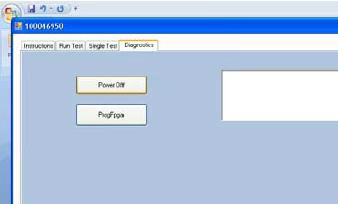
* The FPGA U54 will be programmed with test application firmware.
* DSP/FPGA communication test. A register will reside inside the FPGA to be used to provide a target for a communication test between DSP and FPGA. Bit patterns will be written to target, read back from target, and verified to be correct value.
* U55 100MHz output CLK test. A counter implemented inside the FPGA to verify the output frequency of U55.
* The FPGA test firmware will generate a waveform that toggle the WDI input pin of U45 every 2mS. The signal output WDOK of U45 will be monitored with FPGA test firmware.
* The UUT output voltages will be verified to be within tolerance through the on-board ADC U43. The conversion results will be read by the DSP on the DSP daughter Board and compared to the expected results over the SPI Bus. The transfer function for the on-board ADC is as follows: VADC = (ADC\_CNT) \* (5.0/65535.0).
* Temperature Sensor circuit. For the board temperature sensor measurement an ambient temperature between 20C to 32C will be measured.

TEST PROCEDURE:

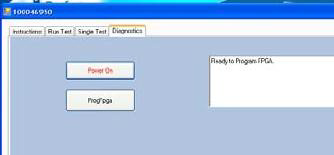
***8.1.1 Program the FPGA with Test Firmware***

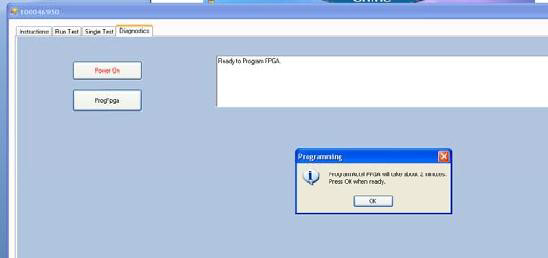
* Program the FPGA with Test Firmware:

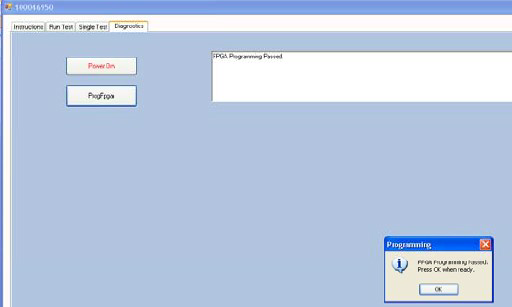
1. Connect the 10 pins header connector of the Actel FlashPro4 programming emulator to UUT J5.
2. Click on Diagnostics tab to open the FPGA programming window.
3. Click on Power Off button to apply powers to UUT.



1. Click on ProgFpga button to start programming the FPGA.



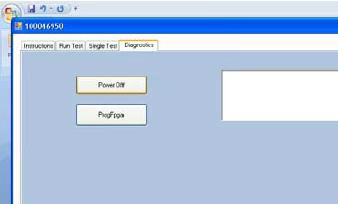
1. Click on OK to load test firmware into FPGA. Program the FPGA will take about 2 minutes.
2. Check the Diagnostic message window for the status of the FPGA programming. Click OK when FPGA programming is done.



1. Click on Power On button to turn off the power.



1. Click on Run Test Tab to start testing.



***8.1.2 24V DC input.***

* Connect a DC source DC#5 to between J9-1, 3, 5 and J9-2, 4, 6. Configure DC#5 to 24.0V @ 1.0A.
* Connect a DC source DC#6 to between J18-1, 4 and J18-3, 6. Configure DC#6 to 24.0V @ 2.93A.
* Power up the fixture interface board with 12V @ 1A DC from power supply DC#2.
* Turn DC#6 on. Monitor the input Voltage and Current. Turn off the Power input with an exception thrown if the input current is over the limit. Observe the status D8 for nominal condition of DMD\_24; LED D8 must be on.
* Reduce the voltage of DC#6 to 15V. Observe the status of the LED for under voltage condition test, LED D8 must be OFF.
* Reconfigure the voltage of DC#6 back to 24V.

***8.1.3 Reset Circuit Test.***

* Verify the following LEDs D8 (24V), D30 (5V), D12 (-5VA), and D13 (+5VA) are ON.
* Verify the LED D34 (POWER\_GOOD) is ON.
* Press and hold the switch SW1
* Verify D34 (POWER\_GOOD) is OFF.
* Release the switch SW1.

***8.1.4 DSP to FPGA communication Test.***

* Write a test pattern 0xAA55 to FPGA’s scratchpad register. Perform a read back and verify that the received test pattern is 0xAA55.
* Write a test pattern 0x55AA to FPGA’s scratchpad register. Perform a read back and verify that the received test pattern is 0x55AA.
* Send powerIfRd Command to FPGA to verify that the status of signal 24V\_GOOD# is low.

***8.1.5 Test Clock output of U55 and Watchdog U45***

* Send oscCounter Command to FPGA to verify the clock frequency of U55 to be 100MHz±0.1MHZ.
* Send powerIfRd Command to FPGA to verify that the status of Watchdog output signal WDOK is high.

***8.1.6 Monitor UUT output voltages.***

* The 12Viso output voltage will be monitored via of signal 12V\_ISO\_MON to be 12.0V±3%.
* The +5VA output voltage will be monitored via of signal 5VA\_POS\_MON to be 5V±3%.
* The -5VA output voltage will be monitored via of signal 5VA\_NEG\_MON to be -5V±3%.
* The 24V output voltage will be monitored via of signal 24V\_MON to be 24V±5%.
* The 5V output voltage will be monitored via of signal 5V\_MON to be 5V±3%.
* The 5Viso output voltage will be monitored via of signal 5V\_ISO\_MON to be 5V±3%.
* The 3.3V output voltage will be monitored via of signal 3.3V\_MON to be 3.3V±2%.
* The 2.5V output voltage will be monitored via of signal 2.5V\_MON to be 2.5V±5%.
* The 1.8V output voltage will be monitored via of signal 1.8V\_MON to be 1.8V±5%.
* The 1.0V output voltage will be monitored via of signal 1.0V\_MON to be 1.0V±5%.

***8.1.8 Test Temperature Sensor Circuit U70.***

* Read the output voltage of the Temperature Sensor U70 at ambient temperature to be within 600mV to 960mV.

***8.1.9 DAC\_OUT signal Test at U62 Pin 5.***

* Send dacWr command to FPGA to set DAC U62 output pin 5 to 0V. Send adcMux and adcSel to FPGA to select signal DAC\_OUT to be the input of the ADC U43. Send adc Acquire command to FPGA to read the ADC output voltage to be between 0V±15mV.
* Send dacWr command to FPGA to set DAC U62 output pin 5 to 5V. Send adcAcquire command to FPGA to read the ADC output voltage to be between 5V±5%.

***8.1.10 Miscellaneous Test.***

* Send gpiRd Command to FPGA to query the DMD Dash Number

configuration to be 0x4. (To set the real expected value.)

* Turns on DMD\_24V power input. Send gpiRd Command to FPGA to query the status of signal SPD2\_24V\_STATUS# to be low.
* Turns off DMD\_24V power input. Send gpiRd Command to FPGA to query the status of signal SPD2\_24V\_STATUS# to be high.

***8.2 SPD DMD Interface Test***

TEST DESCRIPTION:

The SPD DMD Interface consists of J9, inverter, and Source Channel Driver devices.

The outputs of the source channel device will be measured and verified. The loads will be connected to output 1 of U30 and these load conditions will be verified at signal SPDIO\_FLT#.

The output signals of U30 will be connected to J9-7, 9, 13, and 15 respectively to form loopbacks. The loopbacks will provide ability to detect for short and open conditions on the following signals.

|  |  |
| --- | --- |
| Input Signals | Output Signals |
| GNT\_HW\_EN\_MON | GANT\_HW\_EN# |
| GNT\_ST\_DISB\_MON | GANT\_ST\_DISB |
| LFT\_HW\_EN\_MON | LIFT\_HW\_EN# |
| LFT\_ST\_DISB\_MON | LIFT\_ST\_DISB |

TEST PROCEDURE:

8.2.1 Connect J9.8, J9.10, J9.14 and J9.16 to J9.7, J9.9, J9.13 and J9.15 respectively.

8.2.2 Send gpoWr Command to FPGA to assert logic 1 to the following signals GNT\_HW\_EN\_MON, LFT\_HW\_EN\_MON, and DMD\_PWR\_OK.

8.2.3 Measure the voltage at J9.8 to be between 23.5V and 24.5V.

8.2.4 Measure the voltage at J9.10 to be 0V±50mV.

8.2.5 Measure the voltage at J9.14 to be between 23.5V and 24.5V.

8.2.6 Measure the voltage at J9.16 to be 0V±50mV.

8.2.7 Measure the voltage at J9.19 to be between 23.5V and 24.5V.

8.2.8 Verify LEDs D81, D72, D71 and D70 are ON.

8.2.9 Send spdDmdRd command to FPGA to read and verify the status of signal SPDDIO\_FLT# is high.

8.2.10 Send spdDmdRd command to FPGA to read and verify the status of SPD DMD IF to be 0x14.

8.2.11 Send gpoWr Command to FPGA to output logic 1 at signals LFT\_ST\_DISB\_MON and GNT\_ST\_DISB\_MON.

8.2.12 Measure the voltage at J9.8 to be 0V±50mV.

8.2.13 Measure the voltage at J9.10 to be between 23.5V and 24.5V.

8.2.14 Measure the voltage at J9.14 to be 0V± 50mV.

8.2.15 Measure the voltage at J9.16 to be between 23.5V and 24.5V.

8.2.16 Measure the voltage at J9.19 to be 0V± 50mV.

8.2.17 Verify LEDs D81, D72, D71 and D70 are OFF.

8.2.18 Send spdDmdRd command to FPGA to read and verify the status of SPD DMD IF to be 0xA.

8.2.19 Disconnect J9.8, J9.10, J9.14 and J9.16 from J9.7, J9.9, J9.13 and J9.15 respectively.

8.2.20 Send command gpoWr command to FPGA to output logic 1 to signal GNT\_HW\_EN\_MON.

8.2.21 Measure the voltage at J9.8 to be between 23.5V and 24.5V.

8.2.22 Connect 30Ohms load to U30.20. Measure the voltage at J9.8 to be between -500mV and 500mV.

8.2.23 Send spdDmdRd command to FPGA to verify the status of signal SPDIO\_FLT# is low.

8.2.24 Disconnect 30 Ohms load from U30.20.

8.2.25 Send spdDmdRd command to FPGA to verify the status of signal SPDIO\_FLT# is back to high.

***8.3 SPD EMOPS Interface Test***

TEST DESCRIPTION:

The SPD EMOPS Interface consists of J10, Power switch, and inverter device.

The output voltage of the Power switch devices will be measured and verified with the DMM. The status of SPD\_EMPOS\_IF LEDs D16, D17, D22 and D24 will be verified.

The output of the Power Switches will be connected to J10-6, 14, 16, 18, and 19 respectively to form loopbacks. The loopbacks will provide ability to detect for short and open conditions on the following signals.

The signal SITE\_24V\_PWR\_OK# will be tested by applying 0V and 24V at J10-18. The signal status of SITE\_24V\_PWR\_OK# will be monitored by the FPGA test firmware.

The signal status of SPD\_EMOPS\_IF will be monitored by the FPGA test firmware.

|  |  |
| --- | --- |
| Input Signals | Output Signals |
| EMOPS\_STAT1 | EMO\_GOOD# |
| EMOPS\_STAT2 | CCH\_LAT\_LNG\_FLOAT# |
| LFT\_ROT\_BRK\_RLS | CCH\_LFT\_DWN# |
| LAT\_LNG\_BRK\_RLS | CCH\_LFT\_MOT\_EN# |
| EM\_CCH\_24V\_EN | SPD\_EM\_24V\_CCH |

TEST PROCEDURE:

8.3.1 Connect J10.6, J10.14, J10.16, J10.18, and J10.19 to J10.1, J10.2, J10.24, DMD\_24V and J10.25 respectively.

8.3.2 Send gpoWr command to FPGA to output logic 1 at signals EMPOS\_STAT1, and

LFT\_ROT\_BRK\_RLS. Verify LEDs D16 and D22 are ON.

8.3.3 Measure the voltage at signal SPD\_EMSTAT1 to be between 23.5V and 24.5V.

8.3.4 Measure the voltage at signal SPD\_EMSTAT2 to be 0V±50mV.

8.3.5 Measure the voltage at signal SPD\_LFT\_ROT\_BRK\_OVRD to be between 23.5V and 24.5V.

8.3.6 Measure the voltage at signal SPD\_LAT\_LNG\_BRK\_OVRD to be 0V±50mV.

8.3.7 Send gpoWr command to FPGA to output logic 1 at signals EMOPS\_STAT2, and

LAT\_LNG\_BRK\_RLS. Verify LEDs D17 and D24 are ON.

8.3.8 Measure the voltage at signal SPD\_EMSTAT1 to be 0V±50mV.

8.3.9 Measure the voltage at signal SPD\_EMSTAT2 to be between 23.5V and 24.5V.

8.3.10 Measure the voltage at signal SPD\_LFT\_ROT\_BRK\_OVRD to be 0V±50mV.

8.3.11 Measure the voltage at signal SPD\_LAT\_LNG\_BRK\_OVRD to be between 23.5V and 24.5V.

8.3.12 Send gpoWr command to FPGA to output logic 0 at signals EMOPS\_STAT1, EMOPS\_STAT2, LFT\_ROT\_BRK\_RLS, and LAT\_LNG\_BRK\_RLS. Verify LEDs D16, D17, D22 and D24 are OFF.

8.3.13 Measure the voltage at signals SPD\_EMSTAT1, SPD\_EMSTAT2, SPD\_LFT\_ROT\_BRK\_OVRD and SPD\_LAT\_BRK\_OVRD to be 0V±50mV.

8.3.14 Send gpoWr command to FPGA to output logic 1 at signals EMOPS\_STAT2 and

LAT\_LNG\_BRK\_RLS.

8.3.15 Send spdEmopsRd command to FPGA to read and verify the status of signals

CCH\_LFT\_MOT\_EN# and CCH\_LAT\_LNG\_FLOAT# are low, while CCH\_LFT\_DWN# and EMO\_GOOD# are high.

8.3.16 Verify D42 and D44 are ON.

8.3.17 Send gpoWr command to FPGA to output logic 1 at signals EMOPS\_STAT1 and

LFT\_ROT\_BRK\_RLS.

8.3.18 Send spdEmopsRd command to FPGA to read and verify the status of signals

CCH\_LFT\_MOT\_EN# and CCH\_LAT\_LNG\_FLOAT# are high, while CCH\_LFT\_DWN# and EMO\_GOOD# are low.

8.3.19 Verify D43 and D45 are ON.

8.3.20 Send spdEmopsRd command to FPGA to read and verify the status of signal

SITE\_24V\_PWR\_OK# is low.

8.3.21 Verify LED D46 is ON.

8.3.22 Disconnect J10.6, J10.14, J10.16, J10.18, and J10.19 from J10.1, J10.2, J10.24, DMD\_24V and J10.25.

8.3.23 Send spdEmopsRd to FPGA to read and verify status of signal SITE\_24V\_PWR\_OK# is high.

8.3.24 Assert logic 1 to signal EM\_CCH\_24V\_EN to enable U28. Measure the voltage at J10.26 (SPD\_EM\_24V\_CCH) to be 24V±5%.

8.3.25 Assert logic 0 to signal EM\_CCH\_24V\_EN to disable U28. Measure the voltage at J10.26 (SPD\_EM\_24V\_CCH) to be 0V±0.5V.

8.3.26 Connect J10.5 to J10.1. Assert logic 1 to signal EMOPS\_STAT1. Verify the status of signal EM\_UP\_LIMIT is low.

8.3.27 Verify D41 is on.

8.3.28 Assert logic 0 to signal EMOPS\_STAT1, verify the status of signal EM\_UP\_LIMIT is high. Disconnect J10.5 from J10.1.

8.3.29 Connect J10.8 to J10.1. Assert logic 1 to signal EMOPS\_STAT1, verify the status of signal EM\_DOWN\_LIMIT is low.

8.3.30 Verify D35 is on.

8.3.31 Assert logic 0 to signal EMOPS\_STAT1, verify the status of signal EM\_DOWN\_LIMIT is high. Disconnect J10.8 from J10.1.

8.3.32 Send command gpoWr command to FPGA to output logic 1 to signal EMOPS\_STAT1.

8.3.33 Measure the voltage at J10.1 to be between 23.5V and 24.5V.

8.3.34 Connect 30Ohms load to U27.20. Measure the voltage at J10.1 to be between -50mV and 50mV.

8.3.35 Send spdDmdRd command to FPGA to verify the status of signal SPD\_EMOPS\_FLT# is low.

8.3.36 Disconnect 30 Ohms load from U27.20.

8.3.37 Send spdDmdRd command to FPGA to verify the status of signal SPD\_EMOPS\_FLT# is back to high.

***8.4 Stand Controller Interface Test***

TEST DESCRIPTION:

The Stand Controller Interface consists of J1, invertor U74 and STN MSSB transceiver U86.

The signal CW\_LIMIT\_24V# and CCW\_LIMIT\_24V# will be tested by applying 0V and 24V at J1.24 and J1.22, the status of signal GNT\_CW\_LIMIT and GNT\_CCW\_LIMIT will be verified with FPGA test firmware and LED D82 and D83.

MSSB will be tested by FPGA packets communication with Tx/Rx loopback connection on J1. Perform loop-back test where connect the TX output to RX input and check FPGA registers.

FPGA continuously checks for communication errors - transmission rate is around 500K packets per second, with continuous back-to-back TX/RX.

There are three registers to check at address 24 (for stand-controller) and 25 (for service panel).

Good packets (bits 15..0)

Bad Packets (bits 27..16)

Aligned (bit 31)

Criteria:

Good Packets should continuously increment (spinning).

Bad Packets should remain static

Aligned bit should be '1'

[MSSB communication test section needs the test firmware function support, the command used to transit packets and check register bits is To Be Decided.]

TEST PROCEDURE:

8.4.1 Send gpoWr command to FPGA to apply logic 1 at signal EXOPS\_GNT\_24V\_EN to enable U28. Measure and verify the output voltage of U28 at J1.1 to be 24V±5%.

8.4.2 Send gpoWr command to FPGA to apply logic 0 at signal EXOPS\_GNT\_24V\_EN to disable U28. Measure and verify the output voltage of U28 at J1.1 to be 0V±0.5V.

8.4.3 Apply +24V to J1.24 (CW\_LIMIT\_24V#) and 0V to J1.22 (CCW\_LIMIT\_24V#).

8.4.4 Send standContRd command to FPGA to verify the status of signal GANT\_CW\_LIMIT# is low, while signals GANT\_CCW\_LIMIT# is high.

8.4.5 Verify D82 is OFF while D83 is ON.

8.4.6 Apply 0V to J1.24 (CW\_LIMIT\_24V#) and +24V to J1.22 (CCW\_LIMIT\_24V#).

8.4.7 Send standContRd command to FPGA to verify the status of signal GANT\_CW\_LIMIT# is high, while signals GANT\_CCW\_LIMIT# is low.

8.4.8 Verify D82 is ON while D83 is OFF.

8.4.9 Remove voltage from J1.24 (CW\_LIMIT\_24V#) and J1.22 (CCW\_LIMIT\_24V#).

8.4.10 Loop J1.10 (MSSB\_TX\_P), J1.11(MSSB\_TX\_N) and J1.14(MSSB\_RX\_P), J1.15 (MSSB\_RX\_N).

8.4.11 Send command to make FPGA transits the packets, transmission rate is around 500K packets per second, with continuous back-to-back TX/RX.

8.4.12 Send command to check FPGA registers bits for Good Packets, Bad Packets and Aligned. Good Packets expect continuous increment, Bad Packets expect remain static and Aligned expect ‘1’.

***8.5 Gantry Brake and Motor Hall Sensor IF Test***

N/A

***8.6 CCHL Interface Test***

TEST DESCRIPTION:

The CCHL Interface consists of Analog Interface DAC, Source Channel Driver, and Inverters devices.

The output voltage of DAC will be set to 0V/2.5V/5V and to be measured and verified with the DMM.

The Source Channel Driver will be tested with nominal and overload conditions. The voltage of channel outputs will be measured and verified with the DMM. Signal LFT\_SERIO\_FLT# will be used to monitor the status of the load conditions.

The U85 is configured at scale 1:4. The analog voltages will be applied at the input of U15 and the output of U85 will be measured and verified with the DMM.

The output signals of U77 will be connected to J12-10, 11, 14, and 6 respectively to form loopbacks. The loopbacks will provide ability to detect for short and open conditions on the flowing signals:

|  |  |
| --- | --- |
| Inverter’s Input Signals | Inverter’s Output Signals |
| LFT\_SER\_CLK | LFT\_MTN\_EN# |
| LFT\_SER\_SYNC | LFT\_SER\_PAGE\_SEL# |
| LFT\_SER\_DAC0 | LFT\_DOWN\_LIMIT |
| LFT\_SER\_DAC1 | LFT\_UP\_LIMIT |

TEST PROCEDURE:

8.6.1 Send motgpoWr command to FPGA to output logic 1 to signals LFT\_SER\_CLK, LFT\_SER\_SYNC, LFT\_SER\_DATA [1:0].

8.6.2 Connect J12.1, J12.2, J12.3, and J12.4 to J12.6, J12.10, J12.11, and J12.14 respectively.

8.6.3 Measure the voltage at signals LFT\_GPI [4:1] to be within 23.5V to 24.5V.

8.6.4 Verify LEDs D60 and D61are ON.

8.6.5 Send cchlRd command to FPGA to read and verify the status of signal LFT\_SERIO\_FLT# is high.

8.6.6 Send motgpoWr command to FPGA to output logic 1 to signals LFT\_SER\_CLK, LFT\_SER\_DATA0, and logic 0 to signals LFT\_SER\_SYNC, LFT\_SER\_DATA1.

8.6.7 Send cchlRd command to FPGA to read and verify the status of CHHL\_IF to be 0x5.

8.6.8 Send motgpoWr command to FPGA to output logic 1 to signals LFT\_SER\_SNYC, LFTSER\_DATA1 and logic 0 to LFT\_SER\_CLK, LFT\_SER\_DATA0.

8.6.9 Send cchlRd command to FPGA to read and verify the status of CCHL\_IF to be 0xA.

8.6.10 Send motgpoWr command to FPGA to output logic 1 signals LFT\_SER\_CLK, LFT\_SER\_DATA0, and logic 0 to signals LFT\_SER\_SYNC, LFT\_SER\_DATA1. Measure the voltage at signal LIFT\_GPI1 to be within 23.5V and 24.5V.

8.6.11 Connect 30Ohms resistive load to J12.1. Measure the voltage at LIFT\_GPI1 to be 0V±50mV.

8.6.12 Send cchlRd command to FPGA to read and verify the status of signal LFT\_SERIO\_FLT# to be low.

8.6.13 Disconnect 30Ohms resistive load from J12.1.

8.6.14 Send cchlRd command to FPGA to read and verify the status of signal LFT\_SERIO\_FLT# is back to high.

8.6.15 Disconnect J12.1, J12.2, J12.3, and J12.4 from J12.6, J12.10, J12.11, and J12.14.

8.6.16 Connect 100KOhms resistive between J12.23 and J12.24.

8.6.17 Send dacWr command to FPGA to set the output of LFT\_DAC to 0V. Measure the voltage at signal LIFT\_CURR\_FB\_A to be 0V±50mV.

8.6.18 Set the output voltage of LFT\_DAC to 2.5V. Measure the voltage at signal LIFT\_CURR\_FB\_A to be 2.5V±4%.

8.6.19 Set the output voltage of LFT\_DAC to 5.0V. Measure the voltage at signal LIFT\_CURR\_FB\_A to be 5V±2%.

8.6.20 Set the output voltage of LFT\_DAC to 0V. Disconnect 100KOhms resistive load from J12.23 and J12.24.

8.6.21 Apply 0V analog input between J12.21 and J12.22. Select signal LFT\_CURR\_CMD to be the input of ADC. Send adcAquire command to FPGA to query and verify the voltage at signal LFT\_CURR\_CMD to be 0V±50mV.

8.6.22 Set the analog input voltage to 2.5V. Send adcAquire command to FPGA to query and verify the voltage at signal LFT\_CURR\_CMD to be 0.625V±20mV.

8.6.23 Set the analog input voltage to 5.0V. Send adcAquire command to FPGA to query and verify the voltage at signal LFT\_CURR\_CMD to be 1.25V±20mV.

8.6.24 Set the analog input voltage to -5V. Send adcAquire command to FPGA to query and verify the voltage at signal LFT\_CURR\_CMD to be -1.25V±20mV.

8.6.25 Set the analog input voltage to -2.5V. Send adcAquire command to FPGA to query and verify the voltage at signal LFT\_CURR\_CMD to be -0.625V±20mV.

8.6.26 Set the analog input voltage to 0V. Send adcAquire command to FPGA to query and verify the voltage at signal LFT\_CURR\_CMD to be 0V±50mV.

***8.7 Lift Motor Hall Sensor IF Test***

TEST PROCEDURE:

The Lift Motor Hall Sensor interface consists of a Power Switch IC U13, inverters U14. Power Switch IC will be tested with loads connect to the output. The output voltage of the power switch IC will be measured and verified. The signal LFT\_HALL\_PWR\_OK is used to monitor the status of the load conditions.

The inputs of the Inverter will be driven with digital test patterns and the outputs will be read and verified with FPGA test firmware.

|  |  |
| --- | --- |
| Inverter’s Input Signals | Inverter’s Output Signals |
| HALL\_LFT\_A | LFT\_HALL\_SNS1# |
| HALL\_LFT\_B | LFT\_HALL\_SNS2# |
| HALL\_LFT\_C | LFT\_HALL\_SNS3# |

TEST PROCEDURE:

8.7.1 Apply 20 Ohms resistive load between HALL\_LFT\_+5V and HALL\_LFT\_GND.

8.7.2 Send gpoWr command to FPGA to output logic 1 to signal LFT\_HALL\_PWR\_EN#.

8.7.3 Measure the voltage at HALL\_LFT\_+5V to be 0V±50mV.

8.7.4 Send gpoWr command to FPGA to output logic 0 to signal LFT\_HALL\_PWR\_EN#. Measure the voltage at signal HALL\_LFT\_+5V to be 5V±2%.

8.7.5 Send liftMotRd command to FPGA to read and verify the status of signal LFT\_HALL\_PWR\_OK is high.

8.7.6 Replace the 20 Ohms resistive load with 2 Ohms resistive load.

8.7.7 Measure the voltage at HALL\_LFT\_+5V to be 0V±50mV.

8.7.8 Send liftMotRd command to FPGA to read and verify the status of signal LFT\_HALL\_PWR\_OK is low.

8.7.9 Disconnect 2 Ohms resistive load from J6.5. Send gpoWr command to FPGA to output logic 1 to signal LFT\_HALL\_PWR\_EN#.

8.7.10 Connect 20 Ohms resistive load across HALL\_LFT\_+5V and HALL\_LFT\_GND.

8.7.11 Send gpoWr command to FPGA to output logic 0 to LFT\_HALL\_PWR\_EN#.

8.7.12 Measure the voltage at HALL\_LFT\_+5V to be 5V±10%.

8.7.13 Send liftMotRd command to FPGA to read and verify the status of signal LFT\_HALL\_PWR\_OK is high.

8.7.14 Send gpoWr command to FPGA to output logic 1 to signal LFT\_HALL\_PWR\_EN#.

8.7.15 Remove 20 Ohms resistive load between HALL\_LFT\_+5V and HALL\_LFT\_GND.

8.7.16 Apply DIO logic 1 to signals IN110, and IN112.

8.7.17 Send liftMotRd command to FPGA to read and verify the status of signals LFT\_HALL\_SNS [3:1] to be 0x2.

8.7.18 Verify LED D14 and D6 are ON.

8.7.19 Apply DIO logic 1 to signal IN111.

8.7.20 Send liftMotRd command to FPGA to read and verify the status of signals LFT\_HALL\_SNS# [3:1] to be 0x5

8.7.21 Verify LED D7 is ON. Set the digital input at signal IN111 to logic 0.

***8.8 Service Pendant Test***

TEST PROCEDURE:

The Service Pendant consists power SRV\_PANEL\_24V and MSSB transceiver U100. SRV\_PANEL\_24V will be measured and verified with the DMM. MSSB will be tested by FPGA packets communication with Tx/Rx loopback connection on J8, the test method is similar as describe in section 8.4.

[MSSB communication test section needs the test firmware function support, the command used to transit packets and check register bits is To Be Decided.]

TEST DESCRIPTION:

8.8.1 Send gpoWr command to FPGA to apply logic 1 at signal PAN\_24V\_SW to enable U28. Measure and verify the output voltage of U28 at J8.1 to be 24V±5%.

8.8.2 Send gpoWr command to FPGA to apply logic 0 at signal PAN\_24V\_SW to disable U28. Measure and verify the output voltage of U28 at J8.1 to be 0V±0.5V.

8.8.3 Loop J8.5 (SRV\_MSSB\_RS422\_TX+), J8.10 (SRV\_MSSB\_RS422\_TX-) and J8.4 (SRV\_MSSB\_RS422\_RX+), J8.9 (SRV\_MSSB\_RS422\_RX-).

8.8.4 Send command to make FPGA transits the packets, transmission rate is around 500K packets per second, with continuous back-to-back TX/RX.

8.8.5 Send command to check FPGA registers bits for Good Packets, Bad Packets and Aligned. Good Packets expect continuous increment, Bad Packets expect remain static and Aligned expect ‘1’.

***8.9 Gantry Motor Test***

TEST DESCRIPTION:

This test group verifies the operation of Gantry motor Driver Circuit in two modes, EMOPS Mode and 96V\_GNT mode. During the test, the FPGA will apply three-phases PWM to the input of the motor drivers. The PWM phase and amplitude will be measured and verified with an oscilloscope.

The Gantry overcurrent condition will be verified by the method following. LIFT and BRAKES will be similar.

1. Driving the Current to ~62A:
   * Use a PWM duty cycle of ~96% (max duty) to reduce stress on the board (minimizing output current Iout = Iin/Duty).
   * Select a load with an L/R time constant that keeps current ripple low yet still allows the current to reach the threshold in a few milliseconds. If the time constant is too small (taking a long time to rise), the board can trigger a HSWAP fault after ~30ms at >40A. Which is to be avoided since we are trying to verify the overcurrent comparator circuit.
2. FPGA Configuration and Dynamic Braking:
   * A comparator monitors the current. Once overcurrent is detected, comparator output toggles high. A 1µs digital filter in FPGA helps avoid false triggers from noise – which is significant during PWM.
   * When triggered, the FPGA turns OFF the upper FETs and turns ON the lower FETs, creating a dynamic braking path that circulates current to ground. The test records the maximum current just before shutdown – or maximum value reached.
3. Short Pulse Test (~70A for 100ms):
   * To ensure overcurrent detection is reliable, apply a brief pulse of about 70A for 100ms.
   * If the comparator fails to trigger during this higher-current pulse, the HSWAP will eventually fault. The HSWAP’s FLT# output will latch LOW.
     + Care must be taken in this case - as the HSWAP faults, it turns the input MOSFETs OFF immediately. This creates almost a step load response from the perspective of the power supply which might cause a large input voltage spike, potentially damaging the board. This depends on the power supply’s load regulation and other variables – e.g. cable length etc.
   * A passing condition requires no latched fault (FLT# remains HIGH indicating HSWAP did not trigger) and a peak current measured at ~62A (the threshold) ± margin.

The scope captures below shows the current pulse rising, hitting the comparator threshold, then being clamped by the dynamic braking action. This method confirms both the accuracy of overcurrent detection and the proper response from the hardware protection circuitry.

CH3 – Input current, CH2 – Comparator output

A screen shot of a graph

Description automatically generated A screen shot of a computer

Description automatically generated

Furthermore, the test also measures the current of phase A and phase B. The output voltages of VMOPS\_24V and 96V\_GNT also will be verified with FPGA test firmware. The conversion results will be read by the DSP on the DSP daughter Board and compared to the expected results over the Bus Interface. The transfer function for the on-board ADC is as follows: VADC = (ADC\_CNT) \* (5.0/65535).

The 96V Protection Shunt Circuitry will be tested by verify by GNT\_96VSHUNT- voltage remains low for the GNT\_SHUNT\_EN\_OUT pulse duration, this measurement done with 25ohms resistor connect across J14.

The output voltage of U70 GANTRY\_DRVR\_TEMP also will be read and verified with FPGA test firmware.

The Gantry Brake drivers U98, U96 and U95 will be tested, PWM current on all three brakes will be measured simultaneously with resistive load on each Brake output.

Connect 5ohms between the outputs of each brake (BRK+ and BRK-).

Verify PWM current of 3A on all three brakes simultaneously - for a 5s pulse.

Verify PWM current of 7.5A on all three brakes simultaneously - for a 0.5s pulse.

[Shunt enable pulse, overcurrent detection to trigger dynamic brake action and GNT Brakes PWM signals need the test firmware function support, the FPGA commands used to output pulse and the GNT\_BRK\_CUR spec are To Be Decided.]

TEST PROCEDURE:

8.9.1 Apply 24V to EMOS\_24V. Connect J14.4 (GNT\_96VSHUNT-) to scope CH3, connect 25ohms resistor across J14.3 and J14.4.

8.9.2 Apply 96VDC between signal 96V\_GNT and MGND.

8.9.3 Send gantry96VoltsEn command to FPGA to output logic 1 to signal 24V\_GNT\_EMOPS\_EN and GNT\_MOT\_PWR\_EN, measure the voltage at J14.3 to be 24V±5%.

8.9.4 Send gantry96VoltsEn command to FPGA to output logic 0 to signal 24V\_GNT\_EMOPS\_EN and logic 1 to signal GNT\_MOT\_PWR\_EN. Measure the voltage at J14.3 and GND to be 96V±5%.

8.9.5 Send gantry96VoltsEn command to FPGA to output logic 0 to signal 24V\_GNT\_EMOPS\_EN and GNT\_MOT\_PWR\_EN, measure the voltage at J14.3 to be 0Vdc to 0.5Vdc.

8.9.6 Send gantry96VoltsEn command to FPGA to output logic 1 to signal GNT\_MOT\_PWR\_EN. Measure the voltage at J14.3 and GND to be 96V±5%.

8.9.7 Send command to FPGA to output logic 1 to GNT\_SHUNT\_EN\_OUT for 2.5s.

8.9.8 Measure the voltage at J14.4 (GNT\_96VSHUNT-) from Scope CH3, it should be a 96V-0V pulse for 2.5s period.

8.9.9 Disconnect 25ohms resistor from J14.3, disconnect CH3 from J14.4.

8.9.10 Apply 24VDC to signal EMOPS\_24V and GND. Send powerIfRd command to FPGA to read and verify the status of signal 24V\_GNT\_EMOPS\_PG is low.

8.9.11 Send gantry96VoltsEn to FPGA to output logic 1 to 24V\_GNT\_EMOPS\_EN.

8.9.12 Send powerIfRd command to FPGA to read and verify the status of signal 24V\_GNT\_EMOPS\_PG is high.

8.9.13 Select signal 24V\_IN\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 24V\_IN to be 24V±5%.

8.9.14 Select signal 24V\_GNT\_EMOPS\_S\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 24V\_GNT\_EMOPS\_S to be 24V±5%.

8.9.15 Select signal 96V\_24V\_GNT\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 96V\_24V\_GNT 21.5V and 24.5V.

8.9.16 Verify the output voltage of U99 is between 0.6V and 0.96V.

8.9.17 Send gantMotPwmSel and gantMotEn command to PFGA to output 333.3Hz pulses to signals GNT\_PWM\_PHA\_HI/LO, GNT\_PWM\_PHB\_HI/LO and GNT\_PWM\_PHC\_HI/LO.

8.9.18 Send gantMotGpio command to FPGA to output logic 1 at signal GNT\_PWM\_EN.

8.9.19 Measure the signal amplitude of GNT\_DRV\_PHASE\_A, B, and C to be 24V±10%.

8.9.20 Measure the signal frequency of GNT\_DRV\_PHASE\_A, B, and C to be 333.3Hz±5%.

8.9.21 Measure the phase between GNT\_DRV\_PHASE\_A, B, and C to be within 115 and 125 degrees.

8.9.22 Send gantMotGpio command to FPGA to output logic 0 to signal GNT\_PWM\_EN.

8.9.23 Connect 25Ohms resistive loads delta connection to GNT\_DRV\_PHASE\_A, B, and C.

8.9.24 Send gantMotGpio command to FPGA to output logic 1 to signal GNT\_PWM\_EN.

8.9.25 Select the signal 96V\_24V\_GNT\_IMON\_V to be an input of the ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 96V\_24V\_GNT\_IMON\_V to be within 87.5mV and 112.5mV. (Spec limit to be settled with KGB.)

8.9.26 Send gantry96VoltsEn command to FPGA to output logic 0 to signal GNT\_MOT\_PWR\_EN.

8.9.27 Connect Scope CH3 to 96V\_24V\_GNT current sample, CH2 to TP118 (OC\_V\_GNT\_MOT\_DRV).

8.9.28 Send command to FPGA to set PWM duty cycle to 96%.

8.9.29 Adjust resistive loads to 0.47 Ohms delta connection to GNT\_DRV\_PHASE\_A, B, and C. (To get 96V\_24V\_GNT\_DRV current ~70A)

8.9.30 Send command to FPGA to output 100ms pulse at signal GNT\_MOT\_PWR\_EN to generate a brief high current pulse for 100ms.

8.9.31 Send powerIfRd command to FPGA to read and verify the status of signal GNT\_MOT\_PWR\_FLT# is HIGH.

8.9.32 Scope will measure the peak current 62.9A±5% on CH3 and positive pulse of 4uS±5% on CH2.

8.9.33 Disconnect Scope CH3 and CH2. Send command to FPGA to set PWM duty cycle back as before.

8.9.34 Recover the PWM pulse input like 8.9.17 and adjust the resistive loads back to 25 ohms delta connection to GNT\_DRV\_PHASE\_A, B, and C.

8.9.35 Send gantMotGpio command to FPGA to output logic 1 to signal GNT\_MOT\_PWR\_EN.

8.9.36 Send powerIfRd command to FPGA to read and very the status of signal OC\_V\_GNT\_MOT\_DRV is low.

8.9.37 Send gantMotGpio command to FPGA to output logic 0 to signal GNT\_PWM\_EN.

8.9.38 Disconnect the loads from GNT\_DRV\_PHASE\_A, B, and C.

8.9.39 Send gantry96VoltsEn command to FPGA to output logic 0 to signal 24V\_GNT\_EMOPS\_EN.

8.9.40 Send powerIfRd command to FPGA to read and verify the status of signal 24V\_GNT\_EMOPS\_PG is low.

8.9.41 Remove 24V source from 24V\_IN.

8.9.42 Apply 96VDC between 96V\_GNT and MGND.

8.9.43 Send gantry96VoltsEn command to FPGA to output logic 0 to signal GNT\_EMOPS\_EN.

8.9.44 Verify the LED D73 is ON.

8.9.45 Select signal 24V\_IN\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 24V\_IN to be 0V±500mV.

8.9.46 Select signal 24V\_GNT\_EMOPS\_S\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 24V\_GNT\_EMOPS\_S to be 0V±500mV.

8.9.47 Select signal 96V\_24V\_GNT\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 96V\_24V\_GNT to be 96V±10%.

8.9.48 Select signal 96V\_GNT\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 96V\_GNT to be 96V±10%.

8.9.49 Send gantMotEn and gantMotPwmSel command to FPGA to output 333.3Hz pulses to GNT\_PWM\_PHA\_HI/LO, GNT\_PWM\_PHB\_HI/LO and GNT\_PWM\_PHC\_HI/LO.

8.9.50 Send gantMotGpio command to FPGA to output logic 1 to signal GNT\_PWM\_EN

8.9.51 Measure the signal amplitude of GNT\_DRV\_PHASE\_A, B, and C to be 96V±10%.

8.9.52 Measure the signal frequency of GNT\_DRV\_PHASE\_A, B, and C to be 333.3HZ±5%.

8.9.53 Measure the phase of GNT\_DRV\_PHASE\_A, B, and C to be 115 degrees to 125 degrees.

8.9.54 Connect 100Ohms resistive loads WYE connection to GNT\_DRV\_PHASE\_A, B, and C.

8.9.55 Select 96V\_24V\_GNT\_IMON\_V to be input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal Select 96V\_24V\_GNT\_IMON\_V to be within 25mV and 50mV. (Spec limit to be settled with KGB.)

8.9.56 Select signal GNT\_DRV\_CURA to be the input of the ADC. Send adc1024Samples command to FPGA to read and verify the voltage at signal GNT\_DRV\_CURA to be between 20mV to 76.6mV. (Spec limit to be settled with KGB.)

8.9.57 Select signal GNT\_DRV\_CURB to be the input of the ADC. Send adc1024Samples command to FPGA to read and verify the voltage at signal GNT\_DRV\_CURB to be between 20mV to 76.6mV. (Spec limit to be settled with KGB.)

8.9.58 Disable 96V\_GNT, remove the loads from J22.

8.9.59 Connect 5ohms between J13.1 (GNT\_BRK1) and J13.10 (GNT\_BRK1\_RET), J13.5 (GNT\_BRK2) and J13.12 (GNT\_BRK2\_RET), J13.7 (GNT\_BRK3) and J13.14 (GNT\_BRK3\_RET).

8.9.60 Send gantry96VoltsEn command to FPGA to output logic 1 to signal GNT\_BRK\_PWR\_EN.

8.9.61 Send command to FPGA to output 5s pulses to GNT\_BRK1\_PWM\_HI/LO, GNT\_BRK1\_RET\_PWM\_HI/LO; GNT\_BRK2\_PWM\_HI/LO, GNT\_BRK2\_RET\_PWM\_HI/LO; GNT\_BRK3\_PWM\_HI/LO, GNT\_BRK3\_RET\_PWM\_HI/LO.

8.9.62 Select signal GNT\_BRK1\_CUR to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage to be 1.7V±10%. (Spec limit TBD.)

8.9.63 Select signal GNT\_BRK2\_CUR to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage to be 1.7V±10%. (Spec limit TBD.)

8.9.64 Select signal GNT\_BRK3\_CUR to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage to be 1.7V±10%. (Spec limit TBD.)

8.9.65 Send command to FPGA to output 0.5s pulses to above GNT\_BRK\_PWM location.

8.9.66 Select signal GNT\_BRK1\_CUR to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage to be 3.5V±10%. (Spec limit TBD.)

8.9.67 Select signal GNT\_BRK2\_CUR to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage to be 3.5V±10%. (Spec limit TBD.)

8.9.68 Select signal GNT\_BRK3\_CUR to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage to be 3.5V±10%. (Spec limit TBD.)

8.9.69 Send command to FPGA to disable pulses from above GNT\_BRK\_PWM location. Remove 5ohm from J13.

8.9.70 Send motgpoWr command to FPGA to output logic 1 to signal GNT\_BRK\_SW\_MON.

8.9.71 Measure J13.2, J13.4 and J13.6 (GNT\_BRK\_SW\_24V) to be between 23.5V and 24.5V.

8.9.72 Send motgpoWr command to FPGA to output logic 0 to signal GNT\_BRK\_SW\_MON.

8.9.73 Measure J13.2, J13.4 and J13.6 (GNT\_BRK\_SW\_24V) to be 0V±0.5V.

8.9.74 Apply 24V to J13.11 (GNT\_BRK1\_SW\_FB), J13.13 (GNT\_BRK2\_SW\_FB) and J13.15 (GNT\_BRK3\_SW\_FB).

8.9.75 Send command to FPGA to verify the status of signals GNT\_BRK1\_FB#, GNT\_BRK2\_FB# and GNT\_BRK3\_FB# to be logic low.

8.9.76 Check D84, D85 and D86 are illuminated.

8.9.77 Apply 0V to J13.11 (GNT\_BRK1\_SW\_FB), J13.13 (GNT\_BRK2\_SW\_FB) and J13.15 (GNT\_BRK3\_SW\_FB).

8.9.78 Send command to FPGA to verify the status of signals GNT\_BRK1\_FB#, GNT\_BRK2\_FB# and GNT\_BRK3\_FB# to be logic high.

8.9.79 Check D84, D85 and D86 are not illuminated.

8.9.80 Send gantry96VoltsEn command to FPGA to output logic 0 to signal GNT\_BRK\_PWR\_EN.

8.9.81 Connect Scope CH3 to 96V\_GNT\_BRK\_DRV current sample, CH2 to TP39 (OC\_V\_GNT\_BRK\_DRV).

8.9.82 Send command to FPGA to set PWM duty cycle to 96% and apply PWM signal to GNT\_BRK1\_PWM\_HI/LO, GNT\_BRK2\_PWM\_HI/LO and GNT\_BRK3\_PWM\_HI/LO.

8.9.83 Adjust resistive loads to 0.98 ohms delta connection to J13.1 (GNT\_BRK1), J13.5 (GNT\_BRK2) and J13.7 (GNT\_BRK3). (To get 96V\_GNT\_BRK\_DRV current ~34A)

8.9.84 Apply 24VDC between signal 96V\_GNT and MGND. Send command to FPGA to output 100ms pulse at signal GNT\_BRK\_PWR\_EN to generate a brief high current pulse for 100ms.

8.9.85 Send powerIfRd command to FPGA to read and verify the status of signal GNT\_BRK\_PWR\_FLT# is HIGH.

8.9.86 Scope will measure the peak current 30.26A±5% on CH3 and positive pulse of 4uS±5% on CH2.

8.9.87 Disconnect Scope CH3 and CH2. Remove resistive load from J13. Send command to FPGA to disable PWM signals.

8.9.88 Send gantMotGpio command to FPGA to output logic 0 to signal GNT\_PWM\_EN.

8.9.89 Send gantry96VoltsEn command to FPGA to output logic 0 to signal GNT\_MOT\_PWR\_EN.

***8.10 Lift 96V Motor Test***

TEST DESCRIPTION:

This test group verifies the operation of Lift motor Driver Circuit in two modes, EMOPS Mode and 96V\_LFT mode. During the test, the FPGA will apply three-phases PWM to the input of the motor drivers. The PWM phase and amplitude will be measured and verified with an oscilloscope. The over current condition will be verified by the similar method described in section 8.9. Furthermore, the test also measures the current of phase A and phase B. The output voltages of VMOPS\_24V and 96V\_LFT also will be verified with FPGA test firmware. The conversion results will be read by the DSP on the DSP daughter Board and compared to the expected results over the Bus Interface. The transfer function for the on-board ADC is as follows: VADC = (ADC\_CNT) \* (5.0/65535).

The 96V Protection Shunt Circuitry will be tested by verify by LFT\_96VSHUNT- voltage remains low for the LFT\_SHUNT\_EN\_OUT pulse duration, this measurement done with 50ohms resistor connect across J14.

The output voltage of U4 LIFT\_DRVR\_TEMP also will be read and verified with FPGA test firmware.

TEST PROCEDURE:

8.10.1 Apply +24VDC to 24V\_IN. Connect J14.2 (LFT\_96VSHUNT-) to scope CH3.

8.10.2 Apply 96VDC between signal 96V\_LFT and MGND. Connect 50ohms resistor across J14.1 and J14.2.

8.10.3 Send lift96VoltsEn command to FPGA to output logic 0 to signal LFT\_EMOPS\_EN and logic 1 to LFT\_MOT\_PWR\_EN, measure the voltage at J14.1 and GND to be 96V±5%.

8.10.4 Send command to FPGA to output logic 1 to LFT\_SHUNT\_EN\_OUT for 2.5s.

8.10.5 Measure the voltage at J14.2 (LFT\_96VSHUNT-) from Scope CH3, it should be a 96V-0V pulse for 2.5s period.

8.10.6 Disconnect 50ohms resistor from J14.1, disconnect CH3 from J14.2.

8.10.7 Send lift96VoltsEn command to FPGA to output logic 1 to signal LFT\_EMOPS\_EN, and LFT\_SHUNT\_EN\_OUT.

8.10.8 Apply 24VDC to signal 24V\_IN and GND.

8.10.9 Send powerIfRd command to FPGA to read and verify the status of signal 24V\_LFT\_EMOPS\_PG is low.

8.10.10 Send lift96VoltsEn to FPGA to output logic 1 to 24V\_LFT\_EMOPS\_EN.

8.10.11 Send powerIfRd command to FPGA to read and verify the status of signal 24V\_LFT\_EMOPS\_PG is high.

8.10.12 Select signal 24V\_LFT\_EMOPS\_S\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 24V\_LFT\_EMOPS\_S to be 24V±5%.

8.10.13 Select signal 96V\_24V\_LFT\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 96V\_24V\_LFT to be between 21.5V and 24.5V.

8.10.14 Verify the output voltage of U8 is between 0.6V and 0.96V.

8.10.15 Send lifttMotPwmSel and liftMotEn command to PFGA to output 333.3Hz pulses to signals LFT\_PWM\_PHA\_HI/LO, LFT\_PWM\_PHB\_HI/LO and LFT\_PWM\_PHC\_HI/LO.

8.10.16 Send liftMotGpio command to FPGA to output logic 1 at signals LFT\_PWM\_EN.

8.10.17 Measure the signal amplitude of LFT\_DRV\_PHASE\_A, B, and C to be 24V±5%.

8.10.18 Measure the signal frequency of LFT\_DRV\_PHASE\_A, B, and C to be 333.3Hz±5%.

8.10.19 Measure the phase between LFT\_DRV\_PHASE\_A, B, and C to be within 115 and 125 degrees.

8.10.20 Send liftMotGpio command to FPGA to output logic 0 to signals LFT\_PWM\_EN.

8.10.21 Connect 25Ohms resistive loads delta connection to LFT\_DRV\_PHASE\_A, B, and C.

8.10.22 Send liftMotGpio command to FPGA to output logic 1 to signals LFT\_PWM\_EN.

8.10.23 Select the signal 96V\_24V\_LFT\_IMON\_V to be an input of the ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 96V\_24V\_LFT\_IMON\_V to be within 87.5 mV and 112.5mV. (Spec limit to be settled with KGB.)

8.10.24 Send gantry96VoltsEn command to FPGA to output logic 0 to signal LFT\_MOT\_PWR\_EN.

8.10.25 Connect Scope CH3 to 96V\_24V\_LFT current sample, CH2 to TP38 (OC\_V\_LFT\_MOT\_DRV).

8.10.26 Send command to FPGA to set PWM duty cycle to 96%.

8.10.27 Adjust resistive loads to 1.23Ohms delta connection to LFT\_DRV\_PHASE\_A, B, and C. (To get 96V\_24V\_LFT\_DRV current ~27A)

8.10.28 Send command to FPGA to output 100ms pulse at signal LFT\_MOT\_PWR\_EN to generate a brief high current pulse for 100ms.

8.10.29 Send powerIfRd command to FPGA to read and verify the status of signal LFT\_MOT\_PWR\_FLT# is HIGH.

8.10.30 Scope will measure the peak current 24.608A±5% on CH3 and positive pulse of 4uS±5% on CH2.

8.10.31 Disconnect Scope CH3 and CH2. Send command to FPGA to set PWM duty cycle back as before.

8.10.32 Recover the PWM pulse input like 8.10.15, adjust resistive loads back to 25 Ohm delta connection to LFT\_DRV\_PHASE\_A, B, and C.

8.10.33 Send gantMotGpio command to FPGA to output logic 1 to signal LFT\_MOT\_PWR\_EN.

8.10.34 Send powerIfRd command to FPGA to read and very the status of signal LFT\_OVER\_CURR is low.

8.10.35 Send liftMotGpio command to FPGA to output logic 0 to signal LFT\_PWM\_EN.

8.10.36 Disconnect the loads from LFT\_DRV\_PHASE\_A, B, and C.

8.10.37 Send lift96VoltsEn command to FPGA to output logic 0 to signal 24V\_LFT\_EMOPS\_EN.

8.10.38 Send powerIfRd command to FPGA to read and verify the status of signal 24V\_LFT\_EMOPS\_PG is low.

8.10.39 Remove 24V from 24V\_IN.

8.10.40 Apply 96VDC between 96V\_LFT and MGND.

8.10.41 Send lift96VoltsEn command to FPGA to output logic 0 to signal LFT\_EMOPS\_EN.

8.10.42 Verify the LED D36 is ON.

8.10.43 Select signal 24V\_LFT\_EMOPS\_S\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 24V\_LFT\_EMOPS\_S to be 0V±500mV.

8.10.44 Select signal 96V\_24V\_LFT\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 96V\_24V\_LFT to be 96V±10%.

8.10.45 Select signal 96V\_LFT\_MON to be the input of ADC. Send adcAcquire command to FPGA to query and verify the voltage of signal 96V\_LFT to be 96V±10%.

8.10.46 Send liftMotEn and liftMotPwmSel command to FPGA to output 333.3Hz pulses to LFT\_DRV\_PHASE\_A, B, and C.

8.10.47 Send liftMotGpio command to FPGA to output logic 1 to signal LFT\_PWM\_EN.

8.10.48 Measure the signal amplitude of LFT\_DRV\_PHASE\_A, B, and C to be 96V±5%.

8.10.49 Measure the signal frequency of LFT\_DRV\_PHASE\_A, B, and C to be 333.3HZ±5%.

8.10.50 Measure the phase of LFT\_DRV\_PHASE\_A, B, and C to be 115 degrees to 125 degrees.

8.10.51 Connect 100Ohms resistive loads WYE connection to LFT\_DRV\_PHASE\_A, B, and C.

8.10.52 Select 96V\_24V\_LFT\_IMON\_V to be input of ADC. Send adcAcquire command to FPGA to query and verify the voltage at signal 96V\_24V\_LFT\_IMON\_V to be within 25mV and 50mV. (Spec limit to be settled with KGB.)

8.10.53 Select signal LFT\_DRV\_CURA to be the input of the ADC. Send adc1024Samples command to FPGA to read and verify the voltage at signal LFT\_DRV\_CURA to be between 20mV to 76.6mV. (Spec limit to be settled with KGB.)

8.10.54 Select signal LFT\_DRV\_CURB to be the input of the AD C. Send adc1024Samples command to FPGA to read and verify the voltage at signal LFT\_DRV\_CURB to be between 20mV to 76.6mV. (Spec limit to be settled with KGB.)

8.10.55 Disable 96V\_LFT, remove the loads from J23.

8.10.56 Send liftMotGpio command to FPGA to output logic 0 to signal LFT\_PWM\_EN.

8.10.57 Send gantry96VoltsEn command to FPGA to output logic 0 to signal LFT\_MOT\_PWR\_EN.

**8.11 Program FPGA**

The test program will program the FPGA U54 with the customer provided firmware,

Part#xxxxxxx(TBD).stp in the final if the board passes the test.

The following steps are performed to verify further if the FPGA of the board under test is properly programmed with customer application:

8.11.1 After the programming sequence, the FPGA content has to be read back using “DEVICE\_INFO” mode of FlashPro tool; compare the checksum with programming log (reference), saved in local drive. Checksums should match.

**8.12 Power Down**

8.12.1 Turns off the UUT power inputs.

8.12.2 Disconnect all cable assemblies from the UUT connectors.