SBExpress-RM6

Rackmount Gen6 NVMe SSD Test System

User Guide

Version 11.0

October 2025

Copyright

Copyright© SANBlaze Technology, Inc., 2025. All Rights Reserved.

Contact Information:
SANBLaze Technology, Inc.
One Monarch Drive
Suite 204
Littleton, Ma (USA) 01460
1-978-679-1400

The information in this document is subject to change without notice and should not be construed as a commitment by SANBlaze Technology, Inc. SANBlaze Technology, Inc. assumes no responsibility for any errors that may appear in this document.

The software, if any, described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license. SANBlaze Technology, Inc. and its affiliated companies assume no responsibility for the use or reliability of software or equipment not supplied by SANBlaze.

The following are trademarks of SANBlaze Technology, Inc.: SANBlaze™, *Certified by SANBlaze*™, VirtuaLUN™, VLUN™, iRiser™, the slogan *We Test NVMe over Everything*™, and the SANBlaze logo. All other trademarks and registered trademarks are the property of their respective holders.

SANBlaze holds multiple patents on its technology. Visit https://www.sanblaze.com/patents for more information.

For Technical Support:

SANBlaze Help Center: https://sanblaze.atlassian.net/servicedesk/customer/portals
Email: support@sanblaze.com
Website: https://www.sanblaze.com/contact-storage-testing-support

Table of Contents

Contents

Copyright 2

LIST OF FIGURES **7**

GENERAL FEATURES **1**

KEY APPLICATIONS **2**

SBEXPRESS VIRTUALUN NVME SOFTWARE **2**

NVME TEST PLATFORM FEATURES **2**

SOFTWARE-ONLY KIT VS. FULL HOST SYSTEM **2**

Software-Only Kit and Full Host System MATRIX 3

SANBlaze Software-Only Kit (on Customer-Provided Hardware) 3

SANBlaze Software and SBExpress-RM6 3

SANBlaze Host with SBExpress-RM6 3

Turnkey NVMe Test System 3

HARDWARE COMPONENTS (RM6) **4**

SANBlaze NVMe iRiser6 Intelligent Device 4

..... 4

SANBlaze NVMe iRiser6SE (Standard Edition) Intelligent Device 5

LEDs (Gen 5 Hardware) 6

Power Supply 7

Fans 8

DT5 Fan Support 8

Connectors 8

CHECKING THE STATUS OF FANS **8**

STATUS OF THE FAN SERVICE **9**

CONFIGURING FAN TEMPERATURE AND SPEED **9**

Fan Temperature 9

Fan Speed 10

Testing the Temperature of the PLX Chip 10
Reporting the Retrain (Recovery) Count 10
Fan Airflow 10
SBExpress-RM6 10
Step 1: Remove all contents from the shipping containers. 13
Step 2: Install the adapters in the host system. 13
Step 3: Connect the PCI Cables Between Host System and SBExpress-RM6 NVMe SSD Test System 14
Step 4: Install Drives in Front 14
Step 5: Connect the USB Cable Between the Host System and SBExpress-RM6 NVMe SSD Test System 14
Step 6: Connect the Power Cords 15
Step 7: Configuring the SBExpress-RM6 15
Connect to the Device 15
Changing the IP Address 15
Connecting via SSH 15
Connecting via the Command Line 16
Changing the IP Address at the Command Line: 16
Step 8: Enable SBExpress NVMe Support 16
Step 9: Setup Tests 16
If a Test Fails 17
When you Remove or Replace a Drive 17
Adding SANBlaze Certified Test Scripts 18
How to Get Your Test Results 19
Getting Test Results Data in CSV Format 20

REMOVE POWER FROM THE DRIVE **24**

SIMULATE THAT THE DRIVE IS PHYSICALLY GONE **24**

BRINGING THE DRIVE BACK **25**

GLITCHING PERST **25**

CHAPTER 4: 2NVME **26**

NVME VERSUS SCSI TERMINOLOGY **26**

NVME PORT STATUS **27**

NVME PORT STATUS SUB-PAGE **27**

NVME TESTS 28

NVME CONFIGURATION 29

INITIATOR PARAMETERS FOR NVME 29

NVME ACTIONS 31

SETTING PORT ACTIONS 32
Injecting T10 DIF Errors 32
T10 DIF Error Injection in Progress 33
Actions 34

NVME CONTROLLER 34

NVME CONTROLLER MAINTENANCE 34

NVME CONTROLLER STATUS 35

NVME CONTROLLER TESTS 36

NVME CONTROLLER ACTIONS 36

NVME NAMESPACE STATUS 37

NVME NAMESPACE ACTIONS TAB 37

SETTING CONTROLLER ACTIONS 39

CONTROLLER MAINTENANCE OPERATIONS 39

NVME NAMESPACE GENERIC I/OS 40

NVME GENERIC I/O PARAMETERS 42

RESERVATION HELPER FUNCTIONS 45
Reservation 45

DEFINITIONS OF PERSISTENT RESERVATION ACTIONS 46

APPENDIX A: SBEXPRESS-RM RISER CARD INSTALLATION 47

IRISER GEN 5 PRODUCT DESIGN CHARACTERISTICS 47
Controlling the FPGA on the iRiser 47

iRiser5 (standard) 48
iRiser5G (upgrade) 49
Toggling Dual Port on the Webpage 49
DT5V2 Configuration File 49

INSTALLATION STEPS **50**

CONTACT SANBLAZE **53**

APPENDIX B: TROUBLESHOOTING & PRODUCT SUPPORT **54**

EXPORT THE LOG FILES **54**

OPEN A SUPPORT (JIRA) TICKET **56**

CONTACTING SANBLAZE **56**

List of Figures

Figure 1: SANBlaze SBExpress-RM6 NVMe Test Solution 1
Figure 2: SANBlaze NVMe iRiser6 Intelligent Device 4
Figure 3: SANBlaze NVMe iRiser6 Intelligent Device 5
Figure 4: Gen 5 Riser LEDs 6
Figure 6: NVMe MI and System Environmental Commands for the CLI 12
Figure 7: SANBlaze NVMe Gen5 Host System without and with cables (note port numbering) 13
Figure 8: Connect the 2 PCIe cables from the Gen5 Host to the SBExpress-RM6 NVMe System 14
Figure 9: USB from Host to the SBExpress-RM6 NVMe System 14
Figure 10: Select a Drive Slot 16
Figure 11: SANBlaze SBExpress Automated Tests 17
Figure 12: Certified by SANBlaze Automated Test Scripts 18
Figure 13: Running Tests 19
Figure 14: SBExpress Manager Testing Progress 20
Figure 15: Overall Visual of the SBExpress RM/RM6 NVMe Tester 21
Figure 16: SBExpress-RM6 Slot and Port Numbering 22
Figure 17: Certified by SANBlaze Manual 23
Figure 18: NVMe PCI-e Status Sub-page 27
Figure 19: NVMe Port Configuration Page 29
Figure 20: NVMe Port Actions Page 32
Figure 21: NVMe Controller Status Sub-page 35
Figure 22: NVMe Controller Actions Page 36
Figure 23: NVMe Namespace Status 37
Figure 24: The NVMe Namespace Actions Page 38
Figure 25: The NVMe Namespace Generic I/O Page 40

CHAPTER 1: Introduction

Figure 1: SANBlaze SBExpress-RM6 NVMe Test Solution

The SANBlaze SBExpress-RM6 full feature set provides a unique set of functions applicable in all aspects of a product lifecycle; from development to design validation to test and QA. The ability to drive NVMe SSDs with a wide range of configurable attributes provides engineers with a flexible, scalable tool to simulate real disk and memory access environments and issues. Development, qualification, and certification test cycles can be highly automated, thus reducing overall test time by rapidly surfacing errors and non-conformance.

The SBExpress hardware also features full power margining under software control and continuous Voltage, Current and Power measurement at each device while testing.

General Features

- High bandwidth NVMe 2.3 testing (24GB/sec)
- Voltage margining
- Power monitoring at each drive
- Automated stress testing for validation
- Software integration for automation test environments REST/XML API and CLI test infrastructure
- Performance over time, power over time
- Latency testing
- UNH conformance test support
- Dual and Single Port SSD support
- m.2 SSD adapter available
- i2c connection to Management Interface (MI) and VDM
- Supports sixteen drives, single or dual port, all front accessible
- Optimal thermal design, with unobstructed air flow
- Six fans, all independently speed controlled
- Voltage margining +/- 15%
- Hot plug, slot power, and drive presence under software control

©Copyright 2025 SANBlaze. All Rights Reserved.

Key Applications

- Simultaneous signal integrity test control with data verification
- Extensive NVMe command control including firmware download, format, namespace management, T10 DIF
- Generic IO for all NVMe low level and custom commands
- Performance testing
- Error handling
- Multi-port failover testing
- Full validation test capability beyond “UNH conformance” with a fully customizable solution to match end-to-end departmental test requirements from engineering development to validation to QA to manufacturing floor.

SBExpress VirtuaLUN NVMe Software

The VLUN NVMe software provides full control and programmability of parameters, providing unique storage test conditions for NVMe drive testing, development and conformance testing. VLUN software includes:

- Test coverage for all aspects of the 2.3 version of the NVMe specification
- UNH Conformance Test Suite with Dual Port drive support
- SGL, SR-IOV, full namespace control and reservations
- Drive multiple ports of traffic simultaneously
- Send specific or custom op codes in an easy-to-use scriptable format
- Read/write/compare testing
- Error injection
- Custom command builder
- Drive and test single or multiple NVMe target devices

NVMe Test Platform Features

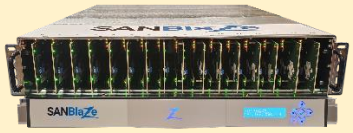
- Supports 4-lane (Single) and 2×2-lane (Dual) PCIe devices
- Supports a total of 16 single or 16 dual port drives
- Internal power source – 110V/240V AC, 50-60Hz, 8.5A
- NVMe connector savers (interposer boards)
- Hot swappable devices under test
- Status and power indicators
- Power margining
- Power measurement at each slot

Software-Only Kit vs. Full Host System

SANBlaze offers a software-only kit that runs on a customer’s hardware, as well as a full host system where SANBlaze provides the hardware and software bundle. Below is a matrix that shows the differences between the two.

©Copyright 2025 SANBlaze. All Rights Reserved

Software-Only Kit and Full Host System MATRIX

SANBlaze Software-Only Kit (on Customer-Provided Hardware) For customers qualifying host systems with NVMe drive bays or NVMe drive chassis, use the SANBlaze Software-Only Kit to qualify the complete system and NVMe drive configuration. Storage vendors producing NVMe expansion racks and JBODs can use the SANBlaze Software-only Kit to qualify drives in the vendor's own NVMe chassis.		SANBlaze Software and SBExpress-RM6	SANBlaze Host with SBExpress-RM6 Turnkey NVMe Test System Best out-of-box experience 
	Without SBExpress-RM6 (Software-only)	**With SBExpress-RM6 chassis**	**SANBlaze VLUN Host with SBExpress-RM6**
Run NVMe Tests	Yes	Yes	Yes
SBExpress Manager GUI for automated tests and power monitoring	No	Yes	Yes
Test Manager GUI	Yes	Yes	Yes
Hot Plug automated button press	No	Yes	Yes
Voltage Margining	No	Yes	Yes
Power Monitoring	No	Yes	Yes
Gen5 drive testing at Gen4 speed on a Gen4 host	No	Yes	Yes
PERST control	No	Yes	Yes
PWRDN/ClkReq signal control	No	Yes	Yes
System must be on SANBlaze compatibility Matrix	Yes	Yes	N/A
Benchmarked performance	No	No	Yes
Warrantied server	No	No	Yes

©Copyright 2025 SANBlaze. All Rights Reserved

Hardware Components (RM6)

In addition to the chassis, the following hardware components are available with the SBExpress-RM6 NVMe SSD Test System:

- iRiser6 Intelligent Device
- iRiser6SE (Standard Edition) Intelligent Device

The iRiser6 includes additional MACOM switches that enable glitching functionality and is offered at an additional cost. The iRiser6SE is the standard configuration of the iRiser card intended for use with the SBExpress-RM6 platform.

SANBlaze NVMe iRiser6 Intelligent Device

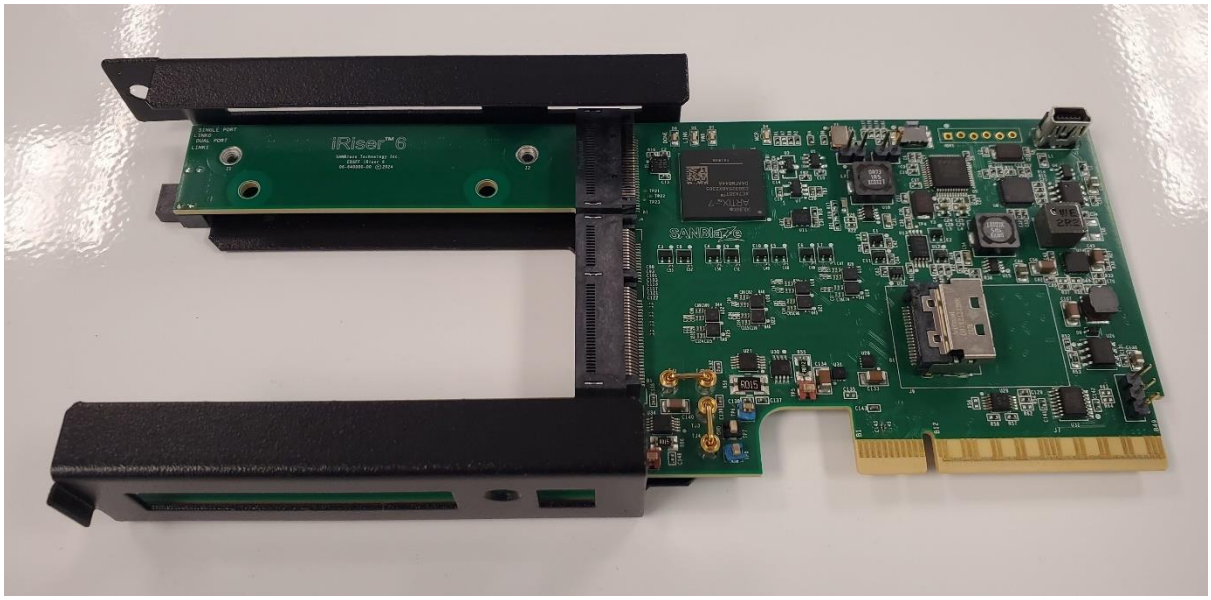


Figure 2: SANBlaze NVMe iRiser6 Intelligent Device

©Copyright 2025 SANBlaze. All Rights Reserved

SANBlaze NVMe iRiser6SE (Standard Edition) Intelligent Device

Figure 3: SANBlaze NVMe iRiser6 Intelligent Device

©Copyright 2025 SANBlaze. All Rights Reserved

LEDs (Gen 5 Hardware)

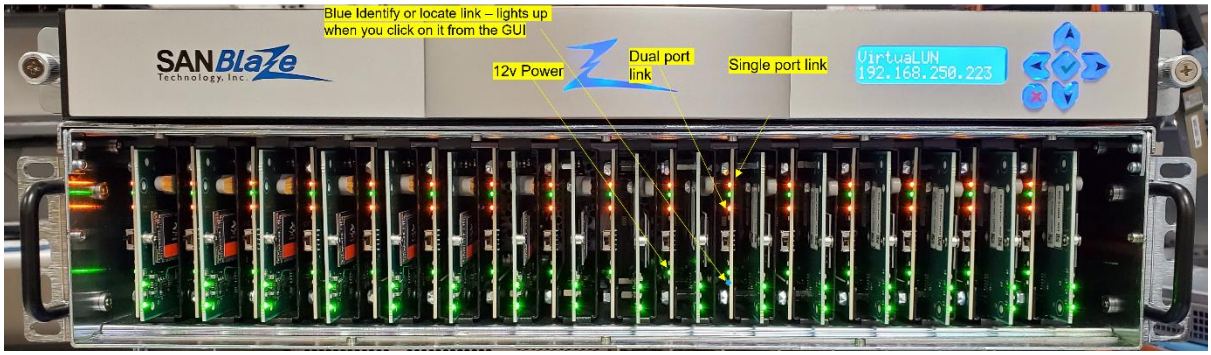


Figure 4: Gen 5 Riser LEDs

Single Port vs. Dual Port LEDs: The number of orange LEDs indicates single or dual port. One orange LED indicates a single port riser. Two orange LEDs indicate a dual port riser (see figure above.)

Link LEDs: Below the single port orange LED is a green LED. This lights up green when the link is up. There is one green LED on the single port riser, and two on the dual port riser.

Drive Activity: The Drive Activity LED becomes active when the NVMe drive is installed.

Power LED: The green power LED indicates that 12V power is “on” for that drive.

Locate LED: The Locate LED is blue when active. In the software, you can click on the “Ident” button to light up the LED on the system.

©Copyright 2025 SANBlaze. All Rights Reserved

Power Supply

The functions you can perform using the power supply include:

- **Read Power**

  `sb_i2c -n 1 -M` (SBExpress-RM **standard** power supply; uppercase **M**)

  ```
  Measuring power from power supply at 0x59 reg 0x8d
  INFO: Power supply temperature = 42.8750
  INFO: Power supply power = 87.0W
  INFO: Power supply voltage = 12.2793V
  INFO: Power supply current = 7.1250A
  ```

  `sb_i2c -n 1 -m` (SBExpress-RMI **industrial** power supply; lower case **m**)

  ```
  [root@vlun-111 ~]# sb_i2c -n 1 -m

  INFO: System 1 SBExpress-RM  SN=920A8110006 Rev=R03 i2c=/dev/i2c-0
  MI_i2c=/dev/i2c-1 VLUN_Port=0

  INFO: restdir = /virtualun/webs/web/rest/sanblazes/1/sn
  INFO: 01[-3] load12V=12378mV, max12V=14498mV, min12V=9653mV

  sb_i2c -n 1 -d -3 -M

  INFO: Command came from foreground process CLI or with -t to force
  INFO: System 1 device=600920000 i2c -y 0 (/dev/i2c-0), vlunport=0,file=8,
  boardtype=16, mux0addr=0x70
  INFO: Found MI bus hardware at /dev/i2c-1, file=10
  INFO: Measuring power from power supply at 0x59 reg 0x8d
  INFO: Power supply temperature = 38.9375C
  INFO: Power supply power = 75.0W
  INFO: Power supply voltage = 12.4883V
  INFO: Power supply current = 6.0781A
  ```

  `sb_i2c -n 1 -d -3 -M`

  ```
  INFO: Command came from foreground process CLI or with -t to force
  INFO: System 1 device=600920000 i2c -y 0 (/dev/i2c-0), vlunport=0, file=8,
  boardtype=16, mux0addr=0x70
  INFO: Found MI bus hardware at /dev/i2c-1, file=10
  INFO: Found Industrial Power Supply -M option not supported, use -m to
  measure power
  ```

- **Set 12V rail to X power +- 10% of 12Volts**

  ```
  sb_i2c -n 1 -v 12000
  ```

  Use `-v N` to set the voltage at the power supply to N mV.

  Use `-v 0` to recalibrate the power supply (max/min/nominal)

  Caution: Do not initiate a calibration with drives populated. The power supply will be set to Min and Max and may adversely affect devices under test.

- Operating temperature

  Commercial: -5C to 40C

  Industrial: -5C to 70C

Fans

- Six variable speed fans are controlled by ***sb_fans.service***
- Fans are controlled through two zones – temperature and speed – via a configuration file called ***/etc/sb_fans.conf***

```
See sections below for more information on sb_fans.service and the
sb_fans.conf configuration file.
```

DT5 Fan Support

The DT5 is self-contained, and no longer supports more than one system. With this, use sb_sdb -s to identify the system, which is the preferred method for all Gen5 systems.

With this support for FT260 controller has been added. The fans still use mcp2221, but need to know that the system may have FT260 to get the bus number correct. Address definitions for fan chips for DT5 have been added. The switch names and locations for DT5 need to be defined.

Define the switches in the config file called /etc/ioexpander_gen5.cfg. The source for

/etc/ioexpander_gen5.cfg is in ./sb_i2c2/ioexpander_gen5.cfg and is added to the kit. sb_fans remake.sh will copy the file to /etc during build.

Connectors

- 1 x 16 IPASS
- 1x16 QSFP-DD
- USB-IN (Cable must be connected to this port from the VLUN for the VLUN software to operate correctly.)
- USB-Out (additional USB port)

Checking the Status of FANs

To check the status of the fans, you can run the commands that are shown after entering:

```
cd /rest/sanblazes/1/
ls fan*
```

For example:

```
[vlun@vlun-111 ~]$ cd /rest/sanblazes/1/
[vlun@vlun-111 ~]$ ls fan*

fan1fail fan1temp fan2rpm fan3fail fan3temp fan4rpm fan5fail fan5temp
fan6rpm

fan1rpm fan2fail fan2temp fan3rpm fan4fail fan4temp fan5rpm fan6fail
fan6temp
```

These commands check the fan temperature, RPM, and fan failure (indicates the fan isn't running.)

Note that the "1" in the example above indicates there is one SBExpress-RM/RM6 device.

Use the individual commands to check for temperature, RPM, or fan failure:

```
[vlun@vlun-111 ~]$ cat /rest/sanblazes/1/fan1temp
26
```

©Copyright 2025 SANBlaze. All Rights Reserved

```
[vlun@vlun-111 ~]$ cat /rest/sanblazes/1/fan1rpm
5221
[vlun@vlun-111 ~]$ cat /rest/sanblazes/1/fan1fail
0
```

Use "*" to batch the command. For example, to see the temperatures of all the fans, run:

```
cat /rest/sanblazes/1/fan*temp
```

For example:

```
[vlun@vlun-111 ~]$ cat /rest/sanblazes/1/fan*temp
26
26
28
28
30
30
[vlun@vlun-111 ~]$
```

In the above example, there are six fans with the output showing the temperature of each fan 1-6.

Status of the FAN Service

To get the status of the service that runs the fans, use this command:

```
[vlun@vlun-111 ~]$ systemctl status sb_fans.service
sb_fans.service - SANBlaze Express NVMe Fan Control
    Loaded: loaded (/lib/systemd/system/sb_fans.service)
    Active: active (running) since Wed, 10 Oct 2018 11:36:14 -0400; 1 day
and 23h ago
        Main PID: 7106 (sb_fans)
          CGroup: name=systemd:/system/sb_fans.service
                  └ 7106 /usr/bin/sb_fans
[vlun@vlun-111 ~]$
```

Note that the service must be running for the above commands to contain accurate data.

Configuring Fan Temperature and Speed

Fans are controlled using two variables – temperature and speed – via a configuration file called ***/etc/sb_fans.conf***. There are four temperature variables T0-T3 and four speed variables S0-S3.

Fan Temperature

Fans will speed up or slow down as the temperature crosses into each temperature variable. At T4 or higher, fans will run at full speed. The default settings for temperature (in Celsius) are as follows:

T0 30
T1 40
T2 50
T3 65

©Copyright 2025 SANBlaze. All Rights Reserved

Fan Speed

Speed zones are triggered by the temperature zones. As the temperature increases, speeds will be moved to the next speed zone. At temperatures of T4 or higher, fans will run at full speed, regardless of the speed zone settings. Default values of the speed zones are as follows:

S0 30
S1 40
S2 60
S3 80

All values are % of MAX RPM (100% = full speed, 0% = stopped fan).

Testing the Temperature of the PLX Chip

To test the temperature of the chip on the PLX switch, use the following command:

```
[root@x10-vlf ~]# sb_sdb /dev/ttyACM1 -T
Temp = 41C
```

Reporting the Retrain (Recovery) Count

The retrain (recovery) count is part of -i and -l, or it can be shown on its own by using the register address bc4.

For example, to report the retrain (recovery) count, use the following command:

```
[root@x10-vlf ~]# sb_sdb -d 2 bc4
0x60810bc4: 0403005d
```

Fan Airflow

By controlling the RPM of the fans, you can match the air flow in various chassis where the NVMe device is deployed. This is a valuable tool to characterize the thermal behavior of the device prior to deployment, or to simulate a device in a user's environment.

The table below shows the approximate LFM (linear feet per minute) air flow rating based on different fan speed configurations (set in the configuration file) for the industrial and commercial chassis. These measurements were taken with 16 same–height drives installed.

SBExpress-RM6		
% Speed set in /etc/sb_fans.conf	**INLET LFM 15mm Drive**	**INLET LFM 7mm Drive**
100	985	627
90	985	627
80	900	573
70	795	506

©Copyright 2025 SANBlaze. All Rights Reserved

60	708	451
50	622	396
40	511	326
30	381	243

©Copyright 2025 SANBlaze. All Rights Reserved

NVMe MI Commands and System Environmental Commands Using the sb_i2c Program

The SANBlaze SBExpress enclosure for testing NVMe drives can be monitored and controlled via a built-in program called **sb_i2c.** The **sb_i2c** program lets you monitor system environmental conditions such as voltage measure and margin, resets, hot plug, and PCIe status.

The **sb_i2c** program can also be used to build custom NVMe Management Interface (MI) commands (for smbus transport only. The "vdm" program is used for vdm transport, but both are called via the "mi" program.)

To build **sb_i2c** commands to go out on the wire, you can use the **mi** program.

In the **CLI User Guide (Initiator)** manual, *Chapter 5* documents how to use the **sb_i2c** program for all supported parameters, including NVMe MI. See below to quickly locate this document.

Figure 5: NVMe MI and System Environmental Commands for the CLI

CHAPTER 2: Quick Start

Step 1: Remove all contents from the shipping containers.

Make sure the following is included in the shipping container:

- SANBlaze host server (1) (not included if you are using your own host server)
- Two host cards required for the system:
  - X16 PCIe host interface module (1)
  - 4-port USB interface (1)
    These cards are shipped with the system or as add-on cards to be installed in the field.
- PCIe cables (2)
- Power cord (2)
- USB cable (1)
- SBExpress-RM6 NVMe SSD Test System Enclosure

Note: The SBExpress-RM6 is a rackmount unit designed to fit into a 19" rack. Refer to the Installation Manual provided in the rail kit included with your system. Also, when mounting the unit, be sure to route your cables so that you can easily access the Power Switch.

Step 2: Install the adapters in the host system.

If you are using a SANBlaze host system, the adapters will already be installed, and you can skip to Step 3. The figure below is an example of a host system populated with cables.

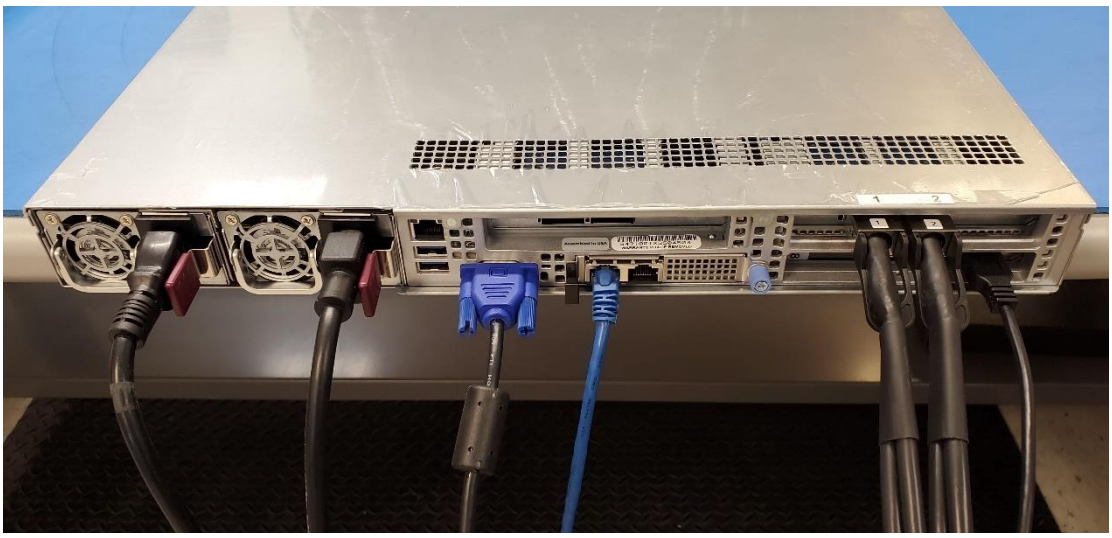


Figure 6: SANBlaze NVMe Gen5 Host System without and with cables (note port numbering)

©Copyright 2025 SANBlaze. All Rights Reserved

Step 3: Connect the PCI Cables Between Host System and SBExpress-RM6 NVMe SSD Test System

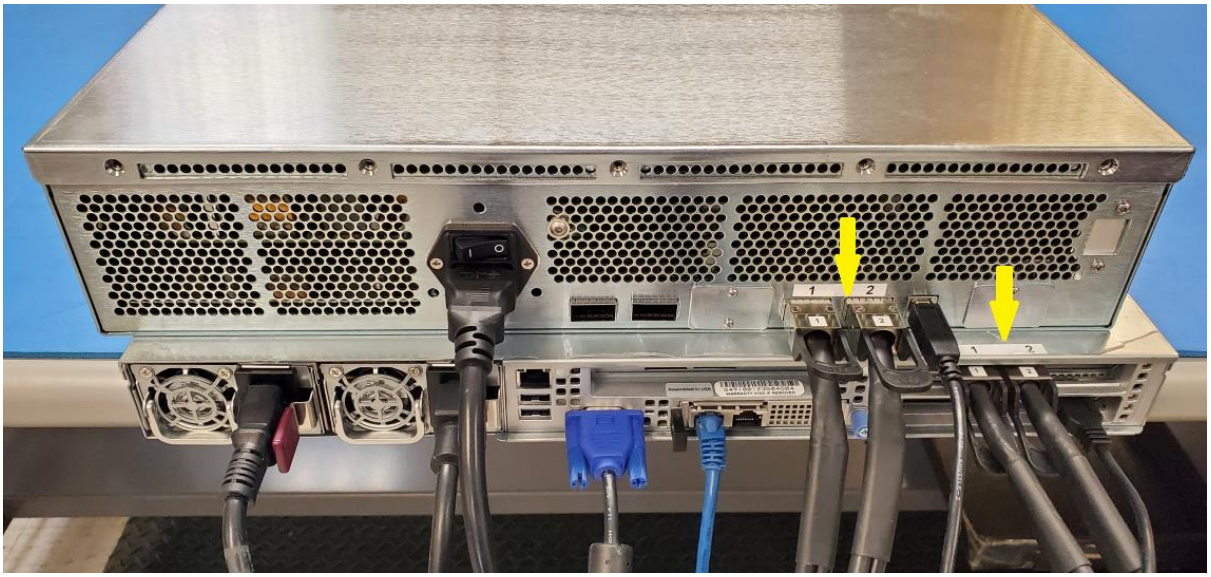


Figure 7: Connect the 2 PCIe cables from the Gen5 Host to the SBExpress-RM6 NVMe System

Connect the two PCIe cables from the host system to the SBExpress-RM NVMe SSD Test System. The two ports on the right correspond with the 2 PCIe lanes per port (port 1 = PCIe lanes 0-3 and Port 2 = PCIe lanes 4-7, for a total of 8 lanes). The two ports on the left-hand side are inactive and designated for future releases.

Step 4: Install Drives in Front

All drives are installed through the front panel. Up to sixteen single or dual port drives can be installed in the SBExpress-RM6 NVMe SSD Test System.

Step 5: Connect the USB Cable Between the Host System and SBExpress-RM6 NVMe SSD Test System

Figure 8: USB from Host to the SBExpress-RM6 NVMe System

©Copyright 2025 SANBlaze. All Rights Reserved

The USB-IN cable is used to communicate and control various features of the SBExpress-RM6 NVMe SSD Test System when used with the SANBlaze software.

Step 6: Connect the Power Cords

1. Connect the power cord and power on the SBExpress-RM6 system by setting the power switch on the back of the system to the **ON** position.
2. Connect the power cord of the host server and turn the unit on (by pressing the green checkmark if using a SANBlaze host system).

   The SBExpress-RM6 NVMe SSD Test System will power up when the host system powers up. The host system will handshake over the PCIe cable to turn on the power to the SBExpress-RM6 NVMe SSD Test System.

Step 7: Configuring the SBExpress-RM6

Configure the SANBlaze SBExpress NVMe SSD Test System through a web browser or Telnet session, using the Ethernet port (10/100/1000 auto sensing) on the front panel. The software uses the 'eth0' interface provided under Linux. Connect to the **eth0** interface of your target machine.

If the ports are not labeled, note that the left Ethernet port is **eth0** and the one to the right of it is **eth1** (see figure below).

Connect to the Device

Use a Web Browser to connect to the SBExpress system.

Use these parameters to connect:

IP Address:	192.168.1.222
Default Gateway:	192.168.1.1
User Name:	system
Password:	SANBlaze *(case sensitive)*

Changing the IP Address

Once the VirtuaLUN system has been accessed, the IP address, system name, and gateway can be changed, using the main web page.

Connecting via SSH

Enter the command:

```
ssh 192.168.1.222
```

Note: If your host is not in the VLUN's /etc/hosts file, the system will take a few seconds to reply.

User Name: vlun
Password: SANBlaze (*case sensitive*)

Upon successful log-in, issue the **su** command to get superuser access.

```
# su
Password: SANBlaze
```

Connecting via the Command Line

In addition to the Ethernet based connectivity methods, the VirtuaLUN can be accessed via the command line if desired. Connecting a monitor and keyboard to the product will allow direct access to the command line.

Once booted, the system will prompt for user name/password:

```
User Name: vlun
Password: SANBlaze (case sensitive)
```

Upon successful login, issue the **su** command to get superuser access.

```
# su
Password: SANBlaze
```

Changing the IP Address at the Command Line:

Run the network config script:

```
# /virtualun/scripts/config_network.sh
```

The script prompts you for all the necessary network settings.

Step 8: Enable SBExpress NVMe Support

Once connected to the device, set up NVMe Support completing the following steps:

1. Select the IP Address on the left-hand side of the panel.
2. Check the box **SBExpress NVMe Support**.
3. Click **Apply**.

Step 9: Setup Tests

To begin running tests, click on a populated drive to get started (later, you will have the option of running tests on all drives.)

In the example below, the drive in Slot 4 was selected by clicking on it. You can also click on the number (104:1) Tab located below the drive view to select the slot (see below):

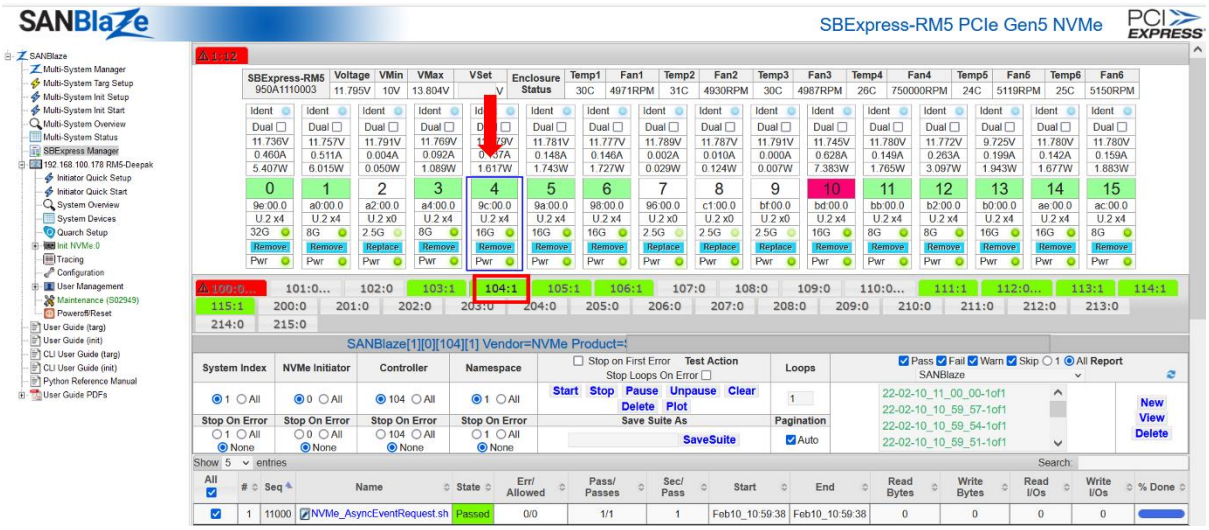


Figure 9: Select a Drive Slot

The background colors indicate the following:

- Green: A drive is inserted into the slot and is linked on PCI.
- Yellow: A drive is inserted into the slot and is not linked on PCI.
- White: No drive is inserted into the slot.
- Pink: The drive is linked on PCI but the NVMe driver did not attach correctly (potentially the drive is broken, or something has gone wrong.)
- Red: a test failed on this drive.

Note: if you do not have any dual port drives in your system, only one row of drives will be lit up.

If a Test Fails

If a test fails, the tab will display as red for that port and the system will show a red tab at the top left of the screen, which indicates the "worst" failure on the system (if there was more than one). You can hover over the tabs to quickly view where the failure occurred and click on the tab to view the tests running on that port. If you clear the tests that failed, the tabs will return to green.

When you Remove or Replace a Drive

If there are existing test results for a slot, and the drive in that slot is going to be removed and replaced by another drive, it is strongly recommended to save the test results first and then clear the test results before replacing the drive. This will be addressed in a future release.

After you have selected a populated drive, scroll down from this screen until you see "Certified by SANBlaze." See figure below.

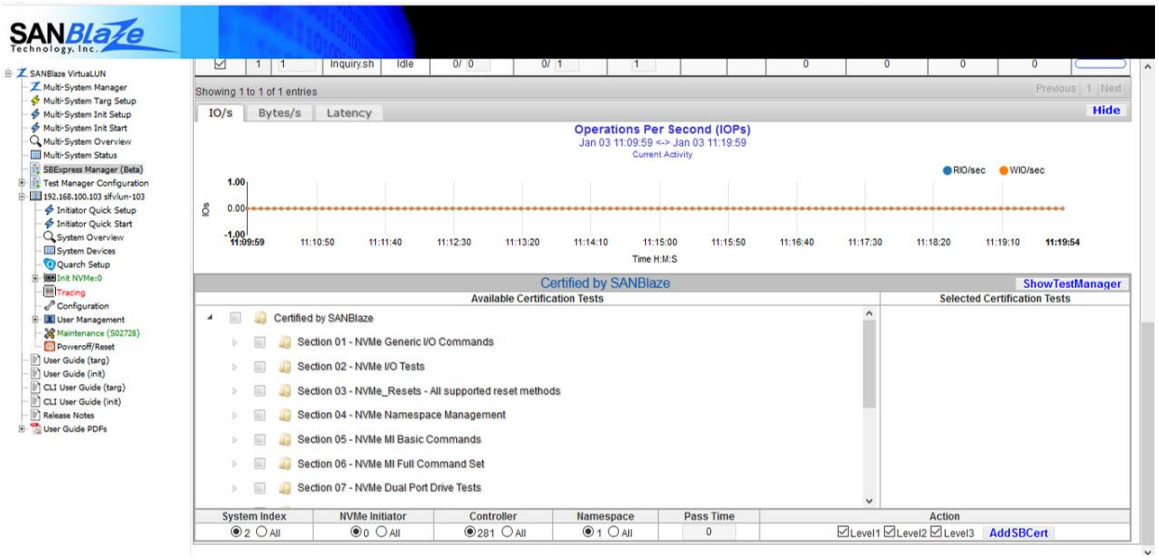


Figure 10: SANBlaze SBExpress Automated Tests

The full set of automated tests include the following test suites:

- NVMe Generic I/O Commands
- NVMe I/O Tests
- NVMe Resets (all supported reset methods)
- NVMe Namespace Management
- NVMe MI Basic Management Commands
- NVMe-MI Full Command Set
- NVMe Dual Port Drive Tests
- NVMe SBExpress Hotplug and Link Testing
- NVMe Quarch Testing Pull/Plug Glitch
- NVMe Miscellaneous Commands (e.g. SR-IOV)
- IOL NVMe Certification
- IOL NVMe-MI Certification
- NVMe ZNS SBCert
- NVMe TCG Opal/Ruby
- SSD Endurance JEDEC Spec. (long runtime)

Adding SANBlaze Certified Test Scripts

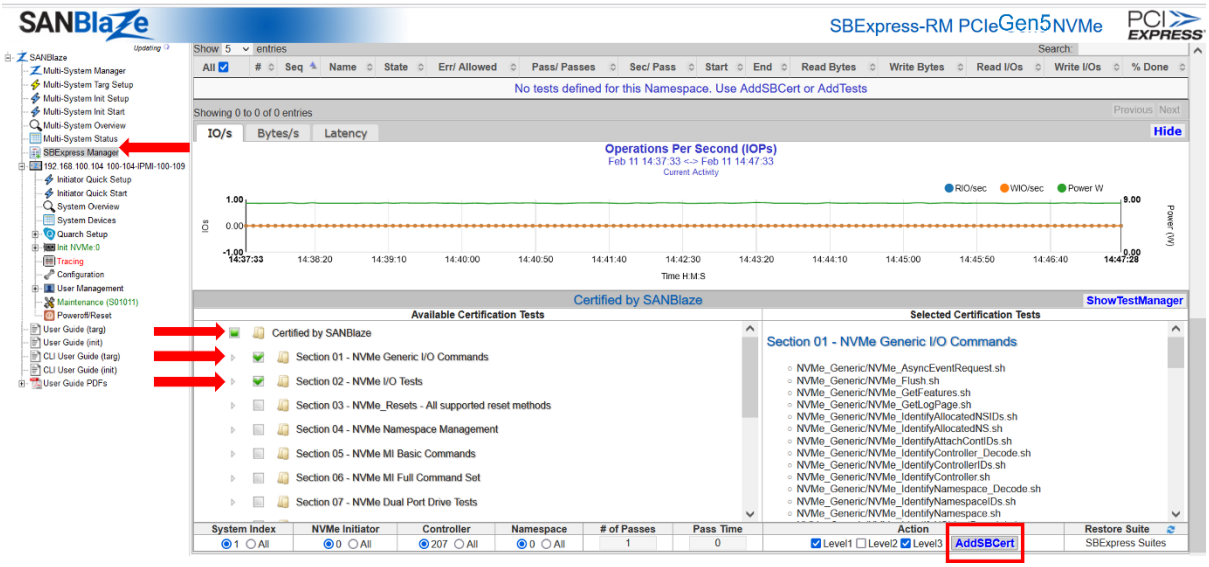


Figure 11: Certified by SANBlaze Automated Test Scripts

To add SANBlaze Certified Test Scripts:

1. In the left-hand panel, select SBExpress Manager.
2. The software shows all the drives in the SBExpress-DT or RM hardware.
3. Select a populated drive that you want to run tests on.
4. Scroll down the screen until you see the *Certified by SANBlaze* section.
5. Select the tests by using the checkboxes next to each suite of tests.
6. Once you have chosen the tests to run, they are displayed in the Selected Certification Tests section of the right panel of the screen (see figure below).
7. Select the range (one or all) at the bottom. You can select **All** for the System Index, NVMe Initiator, Controller, or Namespace. You can also select the Level of tests to run. **Level 1** is

Basic, **Level 2** is Intermediate, and **Level 3** is for Advanced tests that may take a while to run. Depending on which level you check, some of the tests will be grayed out if they belong to a level that is unchecked.

8. Click **AddSBCert** to assign the tests.

9. Scroll up toward the middle of the screen to see the tests you have just added and click the **Start** button to execute the tests (see below).

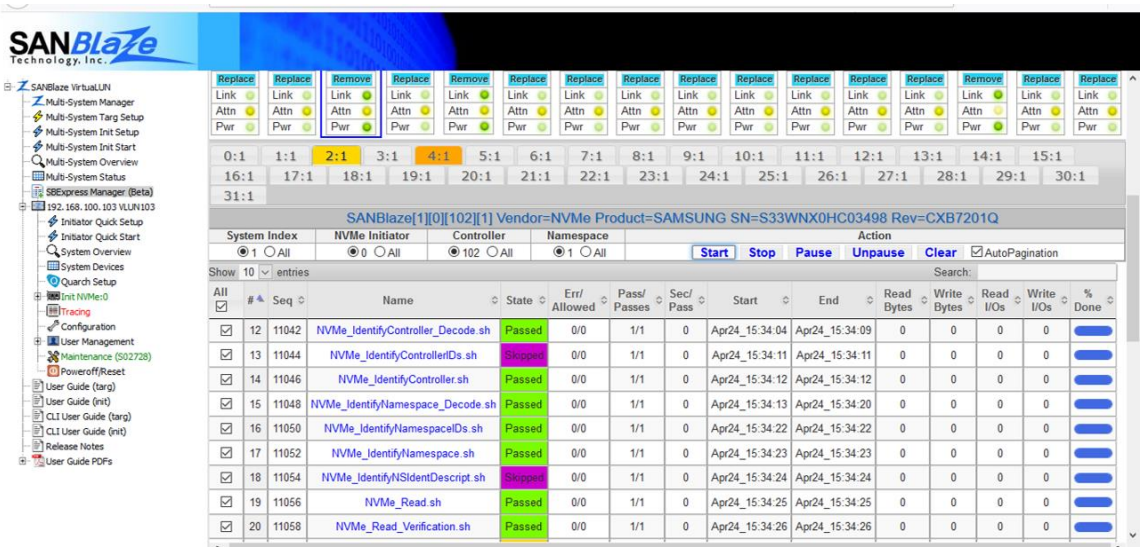


Figure 12: Running Tests

How to Get Your Test Results

Once you have assigned tests to Namespaces, the **SBExpress Manager** page will display the selected tests in the middle section of the page.

Note that the graphing area does not populate until IO testing begins.

Use the **Start** button to start all selected tests. The tests are run in sequence starting from the lowest selected sequence number in the Seq (Sequence) column.

To run one single test alone, uncheck the **All** box, select the test and use **Start** to start it.

While testing is in progress, the page will update automatically, with a refresh rate of 5 seconds. Tests are marked as follows:

Green – Passed
Yellow – Running
Salmon – Paused
Red – Failed
Blue – Stopped by user
Orange – Warning
Purple – Skipped (e.g., the controller does not support the test command running)

The following is an example of a test sequence in progress:

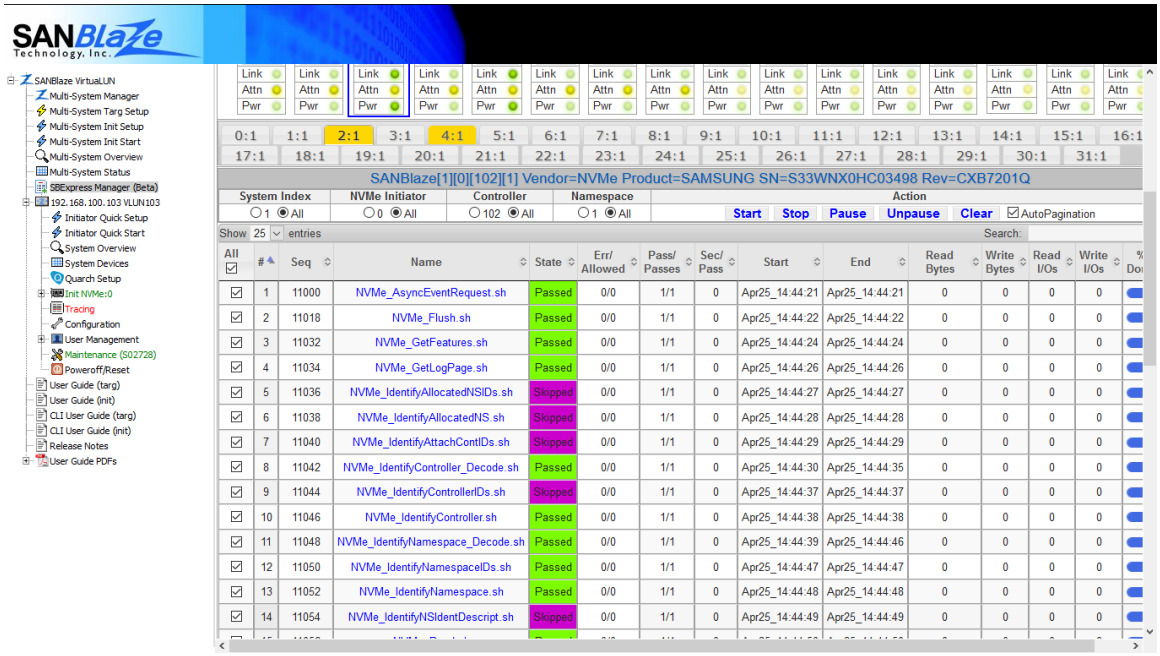


Figure 13: SBExpress Manager Testing Progress

You may pause testing at any time during the test run. This pauses the currently running tests and deducts the time paused from the total testing time. Use **Unpause** to resume testing or **Stop** to complete testing. **Unpause** will continue from the point the test was paused. **Stop** and **Start** will start the test from the beginning.

To act on any single test, for example to re-run a failed test alone, uncheck the **All** checkbox and select the individual test.

You must clear the state of a test before restarting it. **Clear** will reset the test to the Idle state and remove the test results. Currently results are not archived, and therefore you must manually move any results you want to archive.

Getting Test Results Data in CSV Format

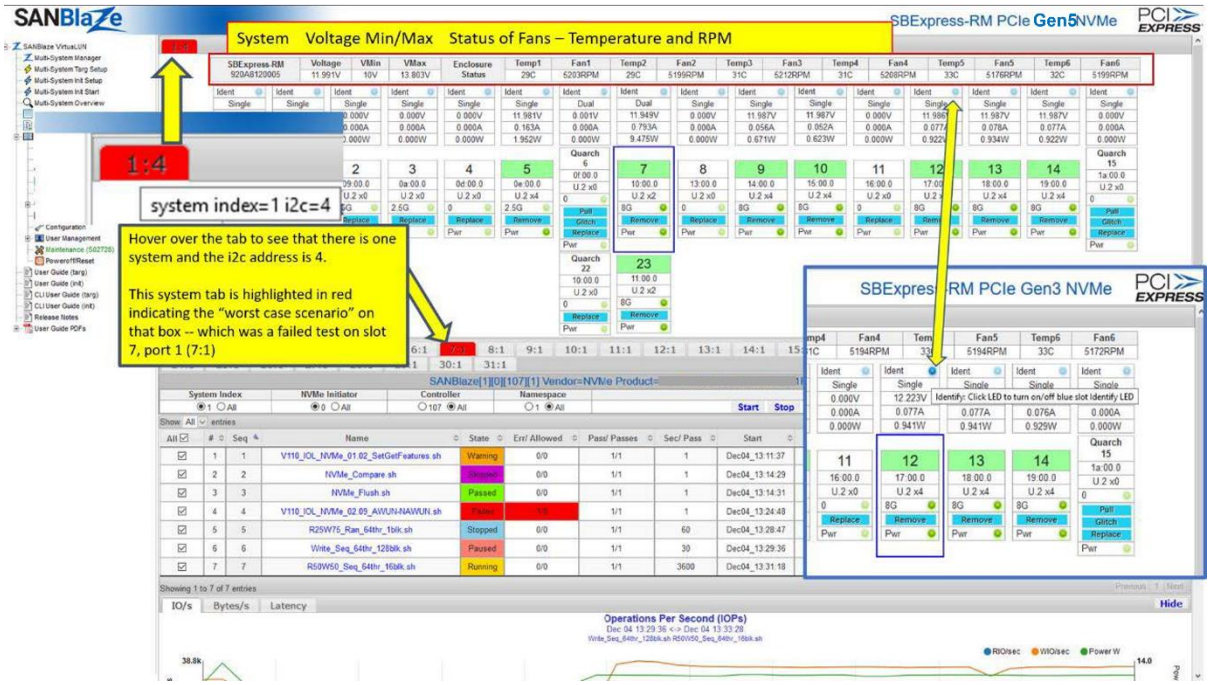
For test results in CSV format, refer to the *SBExpress Environmental Control Program and Remote Control of SBExpress Test Page (REST API) Guide*. You can download it here: https://www.sanblaze.com/rest-interface

Get an Overall Visual of the RM6 Unit Status

When you click on SBExpress Manager on the left-hand menu, it displays a visual of the SBExpress-RM/RM6 unit. You can view overall details of the box such as the system (can be one or more) and its sb_i2c address, the serial number of the unit, and individual fan temperatures and voltage.

You can also click on the blue LED to identify a controller and see its status, and quickly assess the "worst case scenario" on this box once you have some tests running. For example, if a test fails for any reason, the system tab will highlight in red to show you something failed on this test run. If a test is skipped, the system tab will highlight purple, letting you know right away that the controller doesn't support certain tests you are running on it. See the figure below for an example of a system with a variety of tests running and the overall information and details that are displayed.

Figure 14: Overall Visual of the SBExpress RM/RM6 NVMe Tester



SBExpress-RM6 (Gen5) Internal Port Numbering

The slot and port numbering for the SBExpress-RM6 Gen4 NVMe SSD Test System is shown below. Note that Port 2 only applies if the slot is populated with a Dual Port Device.

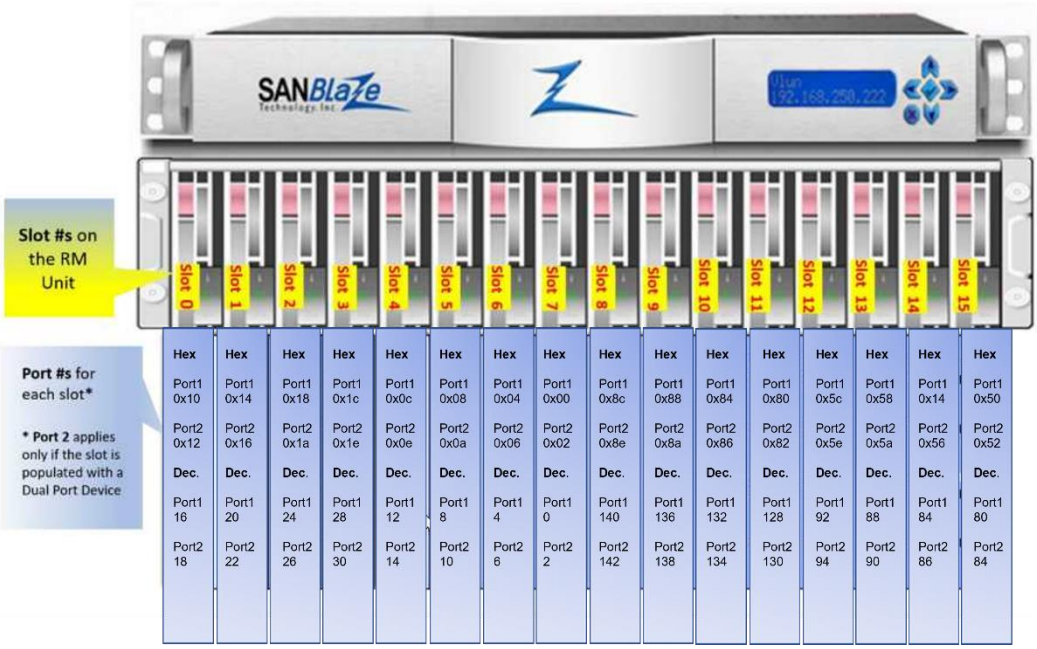


Figure 15: SBExpress-RM6 Slot and Port Numbering

For more information on running tests, refer to the *Certified by SANBlaze User Guide*.

©Copyright 2025 SANBlaze. All Rights Reserved

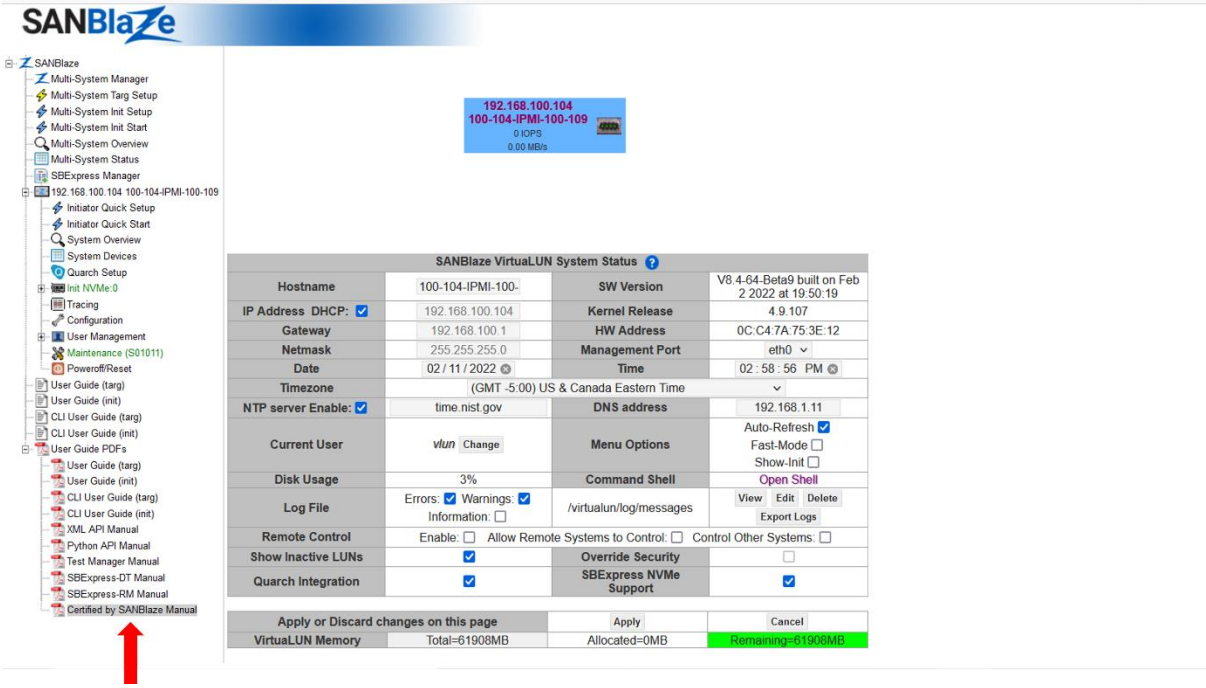


Figure 16: Certified by SANBlaze Manual

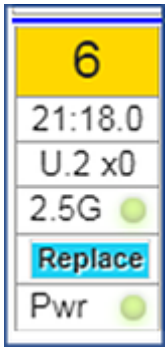
CHAPTER 3: Testing Surprise Removal

When testing your drives, you can make it more convenient to test “surprise removal” without having to physically remove the drive. With SANBlaze you can do this remotely, right from your keyboard, without ever walking into the server room.

Remove Power from the Drive

With SANBlaze, you can kill the power to any drive through the software. The following command removes power from the drive (surprise removal):

sb_i2c -d 6 -f power -w 1



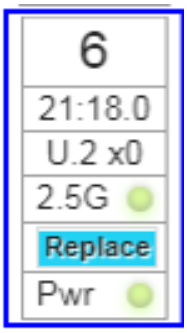
Notice that the LED for **Pwr** on Drive 6 is off (not lit). To see other available commands, use:

sb_i2c -d 6 -f -?

Simulate that the Drive is Physically Gone

This command removes the PRESENT signal (as if the drive is now physically not there) in the SANBlaze-RM and RM6:

sb_i2c -d 6 -f Drive_PRSNT_L -w 0



©Copyright 2025 SANBlaze. All Rights Reserved

Notice that the background color behind slot number 6 is now white not yellow, indicating there is no present signal. According to the software, the drive is “physically” gone.

Bringing the Drive Back

Now, to bring the drive back, you can enable present:

sb_i2c -d 6 -f Drive_PRSNT_L -w 1

This command (assuming there is an actual drive in slot 6) will hot plug "button" the drive (graceful removal/replacement):

sb_i2c -d 6 -b

The drive is now back. Note the color changes to GREEN indicating Present and Link, and Pwr LED is Green indicating the drive has power.

Glitching PERST

You can also glitch PERST using **-w 3** which does a quick on/off, like this:

```
sb_i2c -d 6 -f PORT0_PERST_L -w 3
INFO: System 1 SBExpress-RM6 SN=930A9050002 Rev=R03 i2c=/dev/i2c-4
AtlasPort=/dev/ttyACM0 MI_i2c=/dev/i2c-5 VLUN_Port=0
Feature IOX_PORT0_PERST_L value=3, oneshot action
INFO: 06L[0x62] 0x06[01:00] IOX_PORT0_PERST_L Def=0x01 Cur=0x00
INFO: 06L[0x62] 0x06[01:00] IOX_PORT0_PERST_L Def=0x01 Cur=0x01
```

Or **-w 0** will hold the device in reset and **-w 1** will release reset. As you can see, SANBlaze makes it easy to emulate physical circumstances through the software so that you can easily negative test a variety of scenarios that could happen in the real world with your drives.

©Copyright 2025 SANBlaze. All Rights Reserved

CHAPTER 4: 2NVMe

This chapter describes the NVMe components of the SBExpress NVMe SSD Test System.

NVMe versus SCSI Terminology

Because NVMe does not use SCSI as a base storage protocol, the historical SCSI "view of the world" used on traditional SANBlaze implementations has been replaced with an NVMe centric nomenclature.

While transitioning from SCSI to NVMe testing, on SANBlaze software the following functional comparisons may be helpful:

SCSI		NVMe	NVMe notes
Port	⇔	NVMe PCI-e	One per SANBlaze system
Target	⇔	Controller	One per NVMe device, may be many per system
LUN	⇔	Namespace	May be many per controller

Select an NVMe port by clicking on the port in the left-hand menu. On the top of the page are the following tabs:

Status

Tests

Configuration

Actions

©Copyright 2025 SANBlaze. All Rights Reserved

NVMe Port Status

Selecting the **Status** tab displays port statistics on the **Port Status** sub-page. Below is an example of the NVMe port status sub-page; other protocols may look slightly different.

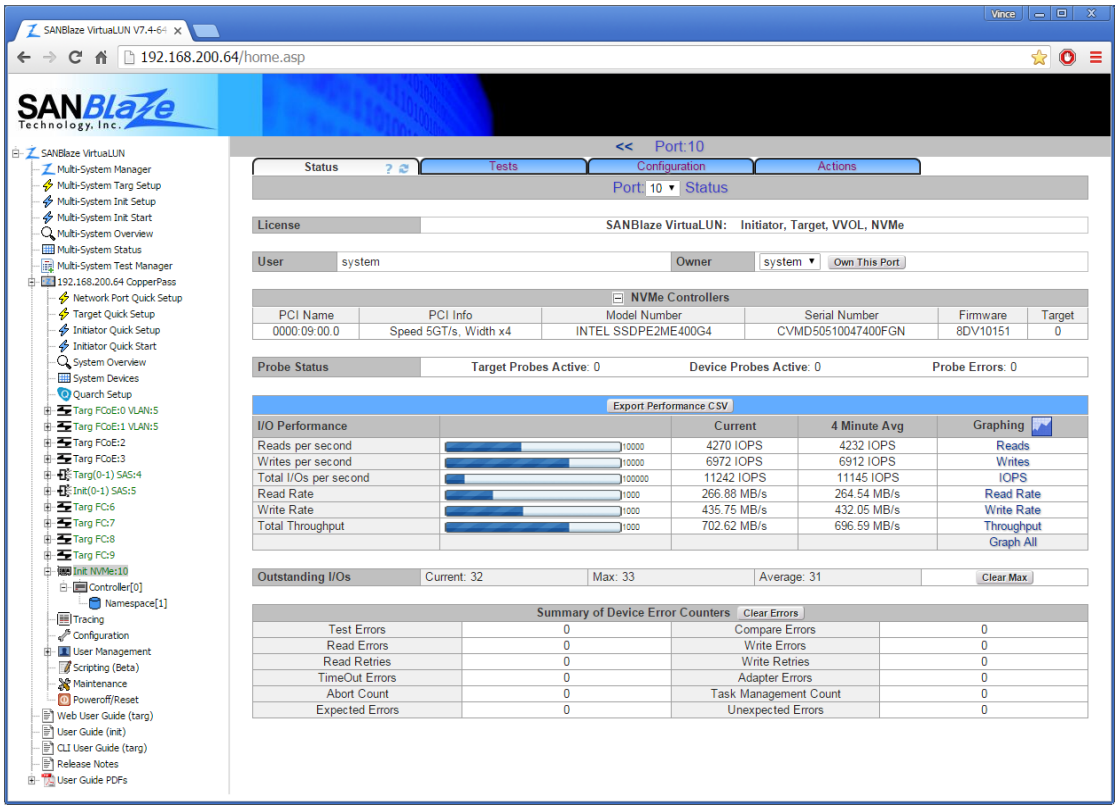


Figure 17: NVMe PCI-e Status Sub-page

NVMe Port Status Sub-Page	
License	Displays VirtuaLUN license information for the port
User	Displays current user of the port, allows change of ownership
NVMe Controllers	Lists information for each NVMe controller found
I/O performance counters	The I/O Performance Counters display the current throughput, the 4-minute average IOPs and throughput, and a graph of the following counters: Current Read I/Os per second Current Write I/Os per second Current Total I/Os per second Current Read Rate Current Write Rate Current Total Throughput
Graphing	Displays NVMe performance in graphical format.

©Copyright 2025 SANBlaze. All Rights Reserved

Outstanding I/Os	Displays the current, maximum, and average outstanding I/Os. The Outstanding I/Os section displays the following information: **Current** - The current depth of the outstanding I/O queue **Max** - The maximum depth of the outstanding I/O queue **Average** - The average depth of the outstanding I/O queue **ClearMax** - Reset the max IO count
Summary of Device Error Counters	Displays summary of errors detected for this port. Accumulates errors detected on all controllers and namespaces: **Test Errors** – Total number of SANBlaze IO Test Errors **Compare Errors** – Total number of compare errors detected by SANBlaze IO Tests **Read Errors** – Total number of failed reads during SANBlaze IO Tests **Write Errors** – Total number of failed writes during SANBlaze IO Tests **Read Retries** – Number of times a Read retries during SANBlaze IO Tests **Write Retries** – Number of times a Write retries during SANBlaze IO Tests **TimeOut Errors** – Number of times an I/O has timed out **Adapter Errors** – Number of errors reported by all NVMe controllers **Abort Count** – Number of times an I/O has been aborted **Task Management Count** – Number of times Task Management has been invoked **Expected Errors** – Number of I/Os that complete in error because they are expected to (due to ABTS or TM) **Unexpected Errors** – Number of I/Os that complete in error but were not expected to (i.e., Adapter Errors plus TimeOut Errors minus Expected Errors) Use **Clear Errors** to clear all counters

NVMe Tests

The *NVMe Port Tests* page displays the test status, configuration, new disk test and I/O latency settings and allows you to set specific parameters where appropriate.

Operation of SANBlaze Tests from this page is common to all port types and is documented in the “Initiator Tests” section in the SANBlaze VirtuaLUN™ *Initiator Emulator User’s Guide*. You can access this guide from the SANBlaze main page under the “PDFs” section.

©Copyright 2025 SANBlaze. All Rights Reserved

NVMe Configuration

Click on the **Configuration** button to display link parameters for the current port, as shown in the figure below. Use the **Apply** or **Cancel** button to apply or cancel configuration changes to a specific port.

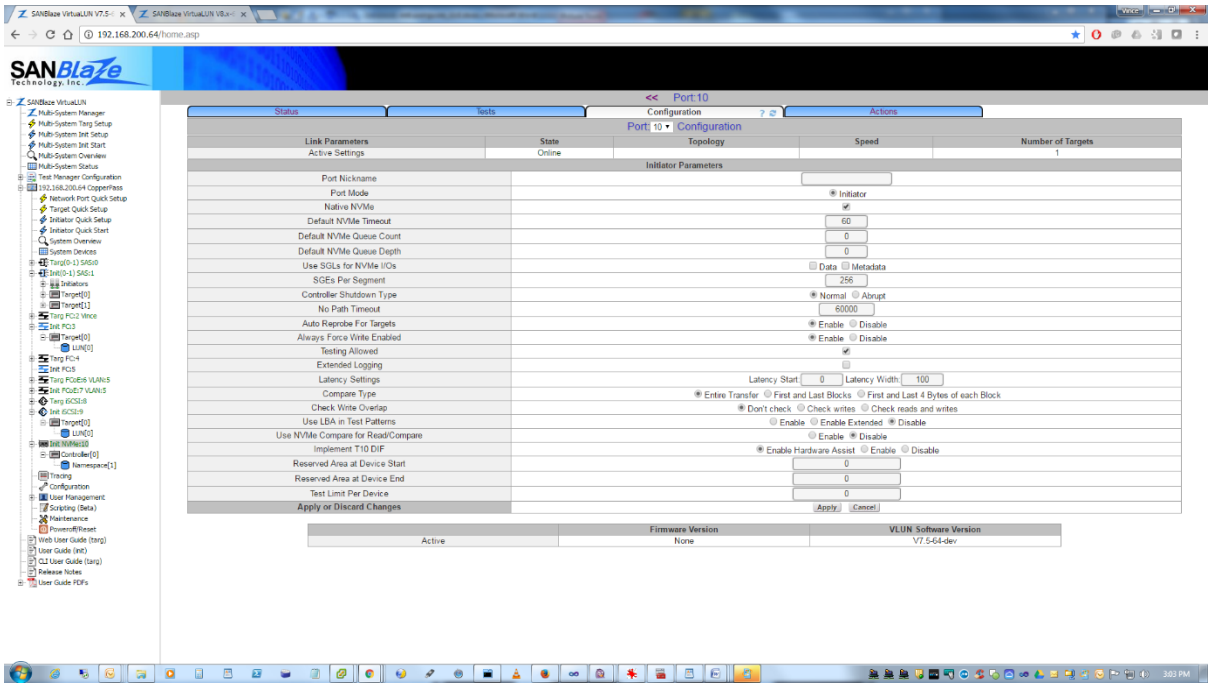


Figure 18: NVMe Port Configuration Page

Active Settings show the current state of the port and are read only.

Initiator Parameters for NVMe	
Port Nickname	Optional user "friendly" name for port.
Port Mode	Currently only Initiator mode testing is available for NVMe.
Native NVMe	**True** – Hide SCSI Compatibility mode on GenericIO page (Default). **False** – Show SCSI Compatibility mode on GenericIO page and NVMe native.
Default NVMe Queue Count / Queue Depth	The number of queues and depth per controller. Default of zero uses current driver values as follows: **queue count -** driver default is number of CPUs max 128 **queue depth -** driver default is 1024 **Note:** the NVMe controller may limit number of queues and depth.
Use SGLs for NVMe I/Os	Use the NVMe scatter/gather format for I/Os. Choose data, metadata or both.
Controller Shutdown Type	Select from: Normal

©Copyright 2025 SANBlaze. All Rights Reserved

	Abrupt
No Path Timeout	Wait up to this time (in milliseconds) for the path to a device to return.
Auto Reprobe For Targets	Automatically re-probe for targets after link events. Default is **Enabled**.
Always Force Write Enabled	Selecting this radio button will change the default for all LUNs on the port from write protected to write enabled. Use with care, be certain no real devices are discovered on the port. Default is **Disabled**.
Testing Allowed	Allow testing of targets and LUNs on this port.
Extended Logging	Create new log files, one per device, that are written with test failure messages and path change messages whenever extended logging is enabled for a given device. These are the same messages that are written to the trace file, but sorted per device. The files are located here:*/virtualun/log/test_device_<id>.*
Compare Type	For write / read / compare type testing, the options are: **Entire Transfer** – the most complete data integrity test, but the slowest. **First and Last Blocks** – compares the entire first and last block of the transfer. **First and Last 4 Bytes of each Block** – the fastest compare test, with the least data checking.
NVMe Queue Select	NVMeQueueSelect=x Allows priority-based queue selection: -3 – means urgent -4 – means high -5 – means medium -6 – means low
Check Write Overlap	Can be set to ensure writes are done to unique LBAs. Options are: 0 – don't check overlap 1 – check that writes don't overlap other writes 2 – check that writes don't overlap other reads and writes
Use LBA in Test Patterns	Enable, Enable Extended, or Disable LBA (Logical Block Address) for use in test patterns. "Enable Extended" option seeds additional information into the user block (device ID, timestamp, pass count).
Use NVMe Compare for Read/Compare	Use the NVMe command to do the Compare operation.
Implement T10 DIF	If a given device is formatted with T10 DIF (Types 0, 1, or 2) then if enabled, use read and write commands that enable checking of the device's T10 DIF protection information; if disabled, use read and write commands that disable checking of the protection information. If a given device is NOT formatted with T10 DIF of any type, this option does nothing.

©Copyright 2025 SANBlaze. All Rights Reserved

	Enable Hardware Assist – Use hardware assist if the device is capable. **Enable** – Use software generated T10 DIF. **Disable** – Don't do T10 DIF even if the hardware is capable.
Enable HW Assist T10 DIF	Use hardware assist if the device is capable. **PRACT**=0, **PRCHK**=111 **Enable** – Use software generated T10 DIF (**PRACT**=1, **PRCHK**=111). **Disable** – Do not do T10 DIF, even if the hardware is capable (**PRACT**=0, **PRCHK**=000).
Reserved Area at Device Start	When testing this device, reserve the specified number of MBytes at the beginning of the device. The default is zero, no reserved area at the start of media.
Reserved Area at Device End	When testing this device, reserve the specified number of MBytes at the end of the device. The default is zero, no reserved area at the end of media.
SGEs per Segment	Segment size in pages for each SGE.
Test Limit Per Device	Capacity limit in MB to be used for IO tests. All seeks will be at or below this specified limit. Default is zero (use entire device).
Firmware Version	Displays the firmware version.
VLUN Software Version	Displays the VLUN software version.

NVMe Actions

Click on the **Actions** tab to display the Port Actions for the port (as shown in

Figure 19).

©Copyright 2025 SANBlaze. All Rights Reserved

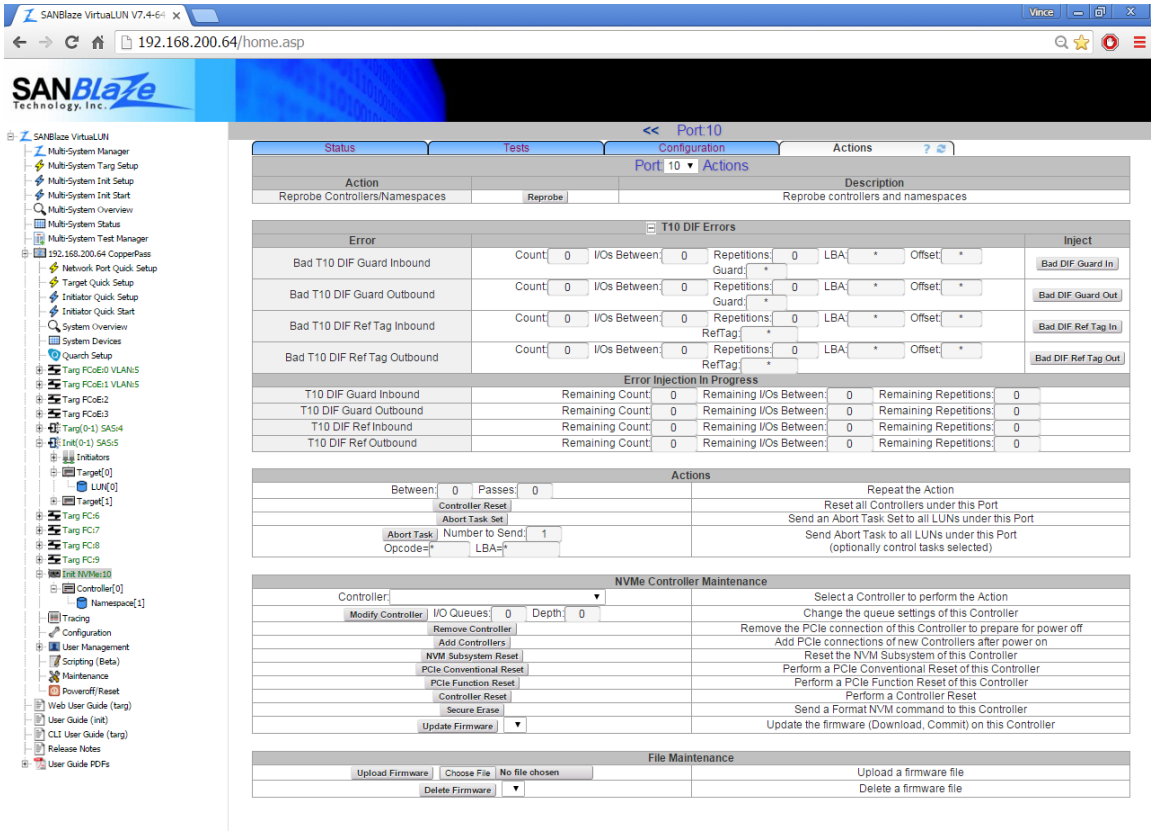


Figure 19: NVMe Port Actions Page

Setting Port Actions

The buttons and drop-down menus on the *NVMe Port Actions* page allow you to set the following actions for a specific port:

Reprobe – Reprobes the PCI-e bus for a current list of controllers and namespaces

Injecting T10 DIF Errors

Selecting the + for T10 DIF Errors will expose the T10 DIF menu as shown above.

Use the buttons and drop-down menus on the *Port Actions* page allows you to insert the following errors for a specific port:

Bad T10 DIF Guard Inbound - Insert Bad T10 DIF Guard Tag to incoming data.

Bad T10 DIF Guard Outbound - Insert Bad T10 DIF Guard Tag to outgoing data.

Bad T10 DIF Ref Inbound - Insert Bad T10 DIF Reference Tag to incoming data.

Bad T10 DIF Ref Outbound - Insert Bad T10 DIF Reference Tag to outgoing data.

For all T10 DIF Errors, the following optional parameters apply:

LBA - 0xN, or 0xN-0xM range or LBAs in hex

Offset - Offset to first LBA with error

DIF - 0xN, a hex number to use for the DIF value

©Copyright 2025 SANBlaze. All Rights Reserved

The arguments above are all optional; they can be omitted or replaced with "*" to signify that the default behavior should be used.

If **LBA** is given, it represents an LBA value (or range) that must be present in the I/O for the I/O to be chosen for error insertion. If **LBA** is 7, then an I/O which reads 2 LBAs starting at 4 does not match, but an I/O which reads 10 LBAs starting at 4 does match. If **LBA** is not given, then the starting **LBA** in an I/O is ignored.

If **offset** is given, then only I/Os that read or write more than that many LBAs (otherwise, trying to insert the error on the LBA with the desired offset will fail, as that **LBA** offset doesn't exist). If offset is not given, then the size of the I/O is ignored.

If **DIF** is given, then when inserting the error, that value is used instead of the actual guard value or the actual reference tag value. If **DIF** is not given, then use the actual guard value XOR'd with 7759, or the actual reference tag value XOR'd with 07071959.

Count - Number of errors to insert

I/Os Between - Number of I/Os before next (set of) error(s) is injected

Passes - Number of times to repeat the error sequence

T10 DIF Error Injection in Progress

Numbers in the table reflect current state of T10 Error Injection.

©Copyright 2025 SANBlaze. All Rights Reserved

Actions

Actions can be taken once, or repeated as follows:

Between – Number of seconds between repeats; 0 for don't repeat

Passes – Number of times to repeat the action; 0 with a non-zero number "between" repeats forever

Controller Reset – Sends a reset request to all NMVe controllers on this port

Abort Task Set – Sends an **Abort Task Set** to all Namespaces on this port

Abort Task – Sends an **Abort Task** to all Namespaces on this port

Opcode – Sends a specific opcode with **Abort Task** above

LBA – Sends a specific Logical Block Address with **Abort Task** above

NVMe Controller

One controller will be presented in the left hand menu for each NVMe controller discovered on the PCIe subsystem.

On the top of the page are the following sub-page tabs:

Status

Tests

Actions

NVMe Controller Maintenance

Maintenance commands can be directed to a specific NVMe controller or All. Select the target controller from the dropdown.

Modify Queues – I/O Queues / Depth, the number of queues and depth per controller. The default of zero uses the current driver values as follows:

Queue count – the driver default is number of CPUs max 128

Queue depth – the driver default is 1024

Note: The NVMe controller may limit the number of queues and depth.

Remove Controller – Remove this controller from the system. You can safely remove an NVMe device after selecting **Remove Controller**.

Add Controllers – Any controllers physically added or removed by using **Remove Controller** will be found and added to the configuration.

NVM Subsystem Reset – Reset the entire NVMe subsystem.

PCIe Conventional Reset – Issues a PCI-e reset; parent is instructed to reset all downstream devices.

PCIe Function Reset – Issues a PCI-e function reset to the selected controller.

©Copyright 2025 SANBlaze. All Rights Reserved

Controller Reset – Reset the NVMe controller.

Secure Erase – Sends a secure erase command to the NVMe drive.

Update Firmware – Load Firmware to a single NVMe device or all devices:

1. Use the File Maintenance menu.
2. Choose file to select your firmware file.
3. Upload Firmware to the system.

   Note: The uploaded firmware file will be available in the dropdown menu. Use **Update Firmware** to download to the NVMe drive and commit.

Commit Firmware – Commit the firmware downloaded to the controller. Options are:

Downloaded image; Do not activate.

Downloaded image; Activate next reset.

Existing image; Activate next reset.

Existing image; Activate immediately.

Delete Firmware – Deletes a firmware file that has been previously uploaded as described above.

NVMe Controller Status

Selecting the Status tab displays NVMe controller statistics on the *Controller Status* sub-page, as shown below.

Figure 20: NVMe Controller Status Sub-page

The *Controller Status* sub-page contains the following sections:

Controller General Information - displays information about the discovered controller

Current Controller Performance - displays performance for this controller

Average Controller Performance - displays average I/O performance counters

Summary of Device Error Counters - displays summary of errors detected for this port

Summary of Path Failures - displays summary of path errors and transitions

NVMe Controller Tests

The *NVMe Controller Tests* page displays the test status, configuration, new disk test and I/O latency settings and allows you to set specific parameters where appropriate.

Operation of SANBlaze tests from this page is common to all port types and is documented in the "Initiator Tests" section in the SANBlaze VirtuaLUN™ *Initiator Emulator User's Guide*. You can access this guide from the SANBlaze main page under the "PDFs" section.

NVMe Controller Actions

If the NVMe controller supports namespace management, click on the controller in the left hand menu, and click the **Controller Actions** tab. At the bottom of the page, **the NVMe Controller Namespace Management** table displays.

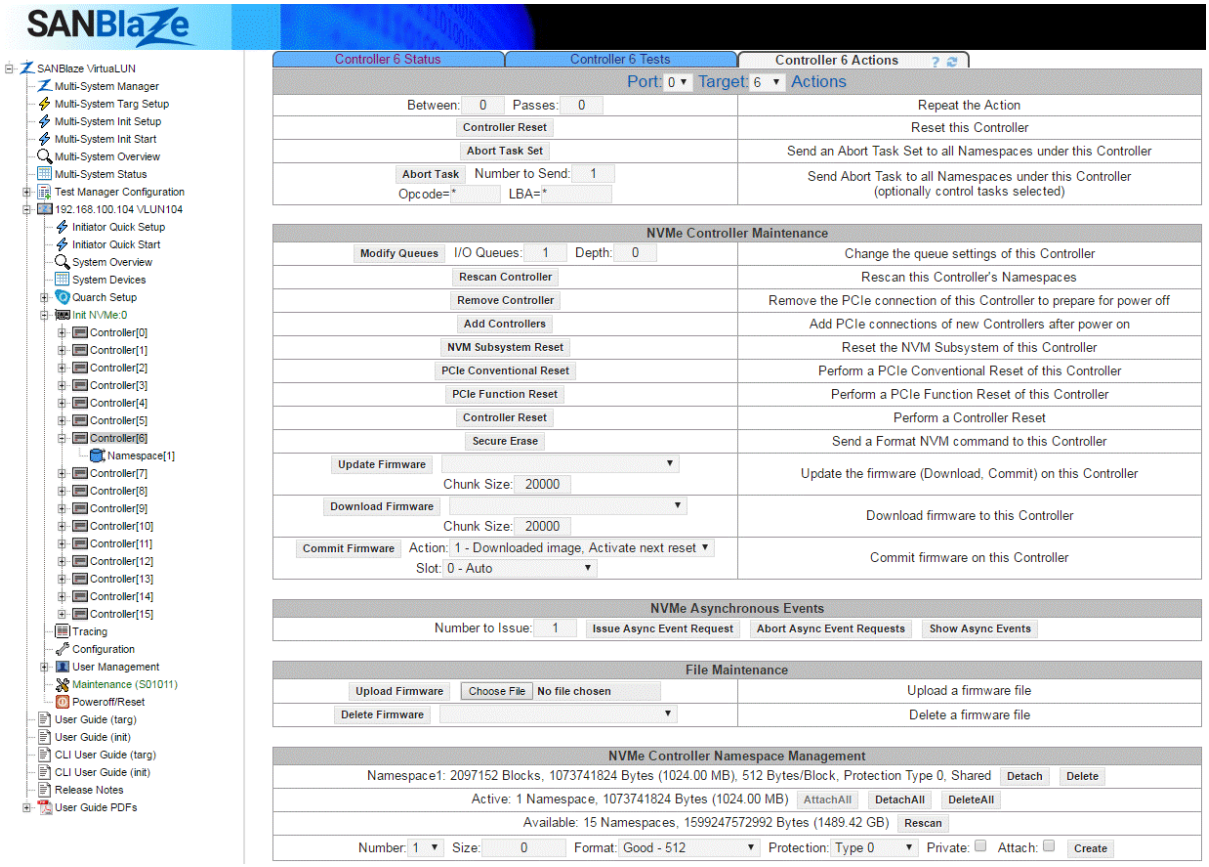


Figure 21: NVMe Controller Actions Page

This table shows all namespaces currently created (if any), along with the size of each one, the LBA format size, protection being used, and whether it's shared or private to the controller. To attach the namespace, click the **Attach** button. To detach, click the **Detach** button. To delete it, click the **Delete** button.

The *Active* line shows a summary of how many namespaces currently exist along with the combined size being used on the controller. To attach all namespaces at the same time, click **Attach All** button.

©Copyright 2025 SANBlaze. All Rights Reserved

To detach them all at the same time, click **Detach All** button. To delete them all at the same time, click **Delete All** button.

The *Available* line shows how many namespaces are still available on the system for creation, along with the total size left on the controller. The **Rescan** button also resides here; it should be clicked any time there is a change to any of the namespaces (create/delete/attach/detach), so that there is an updated view of the NVMe controllers and namespaces in the left hand menu.

To create a new namespace, select the number of namespaces to create using the dropdown in the **Number** parameter. The size can be specified in "kb", "mb", "gb" or "tb" (for example, "100gb").

Alternatively, a number of blocks can be specified using "b" (for example, "1000b"). If you want to use all of the remaining size left available, use the default value of 0. Select **Format** and **Protection** using the dropdowns, then set the **Private** and **Attach** checkboxes as desired, and click the **Create** button. A new namespace will be created and will appear in the table. Click **Rescan** from the left-hand menu to update with the new namespace underneath the controller.

NVMe Namespace Status

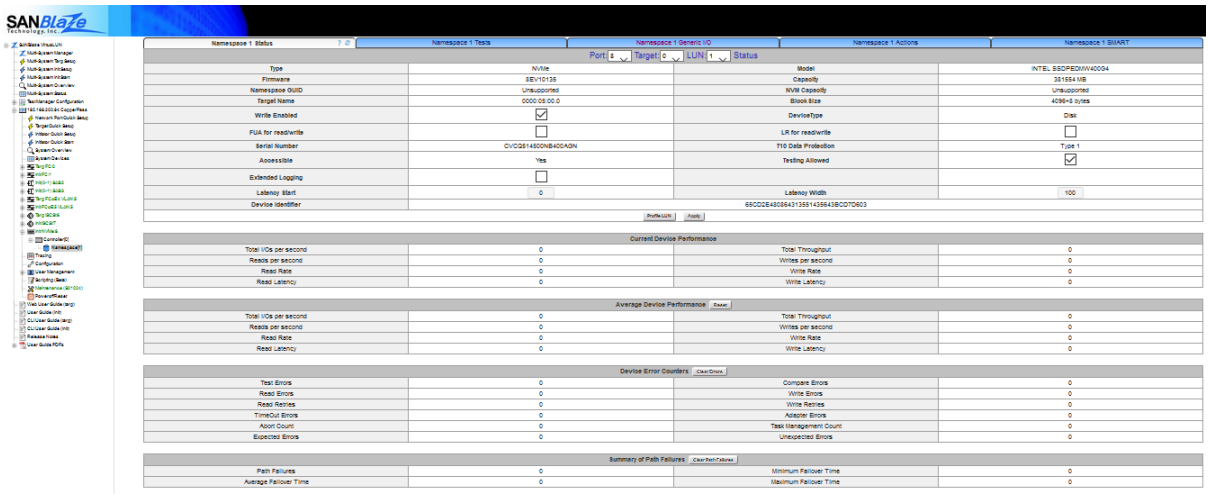


Figure 22: NVMe Namespace Status

Included in the *NVMe Namespace Status* are two specific NVMe Namespace parameters:

Namespace GUID - Namespace Global Unique Identifier from Identify Namespace

NVM Capacity - Total NVMs allocated to this namespace, from Identify Namespace

NVMe Namespace Actions Tab

In the left hand menu, click on the namespace underneath the controller and go to **Namespace Actions** tab. A table called *NVMe Namespace Maintenance* displays, as shown below.

It shows the current details of the namespace, and lets you re-format the namespace. Select the **Format**, **Protection** and **Erase** options using the dropdowns, then click the **Format** button.

©Copyright 2025 SANBlaze. All Rights Reserved

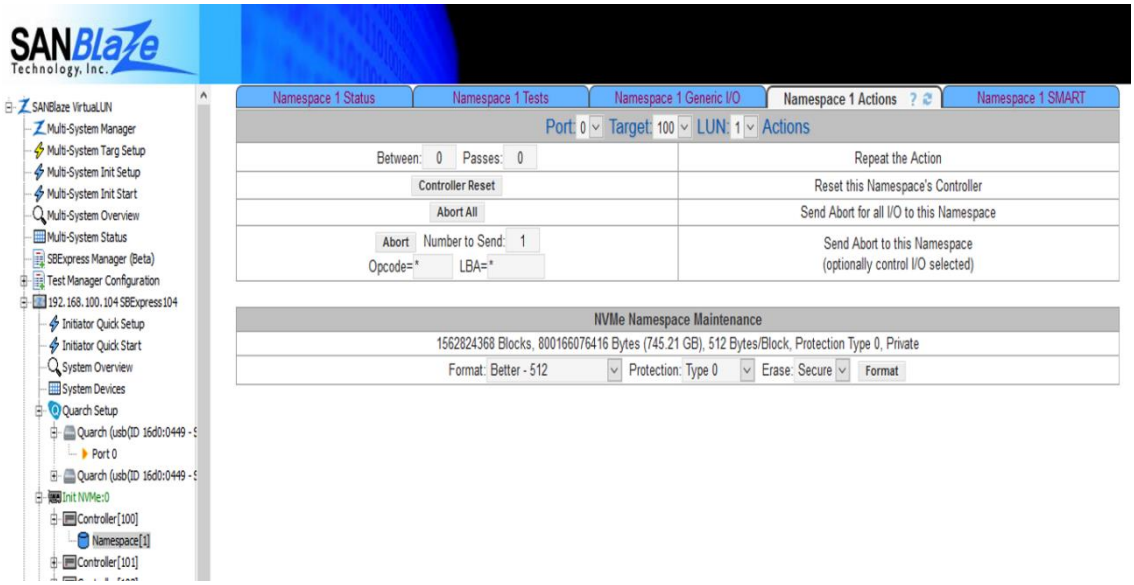


Figure 23: The NVMe Namespace Actions Page

A number of NVMe Namespace actions can be issued directly from this page for the purpose of testing the response by a single Namespace. These actions include:

Controller Reset

Abort All

Abort - Specify the number to send (default 1) specific opcode (default *, all) and LBA (default *, any)

The actions above can be repeated using the "Between" and "Passes" method.

NVMe Namespace Maintenance allows for namespace formatting with multiple options for Block layout, Protection and Erase type.

Setting Controller Actions

These actions can be repeated using the Passes and Between fields. The following actions are provided:

Controller Reset – Sends a controller reset request to the NVMe controller

Abort All – Sends **Abort Task Set** to the controller

Abort – Abort a specific task. The I/O must be outstanding. You can specify the number to send and Opcode or LBA to filter.

NVMe Asynchronous Events:

Issue Async Event Request – Issue the specified number of asynchronous event requests.

Abort Async Event Requests – Send Abort for all outstanding asynchronous event requests.

Show Async Event Requests – Print response for all completed asynchronous event requests.

Controller Maintenance Operations

The following actions can be used to affect various operating parameters of the NVMe controller:

Rescan Controller – Re-scan this controller's namespaces.

Remove Controller – Remove PCIe connection of this Controller to prepare for power off.

Add Controllers – Add PCIe connections of new Controllers after power on.

Secure Erase – Send a Format NVM command to the controller.

Update Firmware – Download then commit the selected firmware.

Download Firmware – Download the selected firmware and don't commit.

Commit Firmware – Commit a download firmware image to the selected slot then take the selected action.

©Copyright 2025 SANBlaze. All Rights Reserved

NVMe Namespace Generic I/Os

A method is provided for low level I/O access to NVMe Namespaces. The page is called *Namespace Generic I/O* and is accessible by selecting any of the discovered namespaces as shown in the figure below.

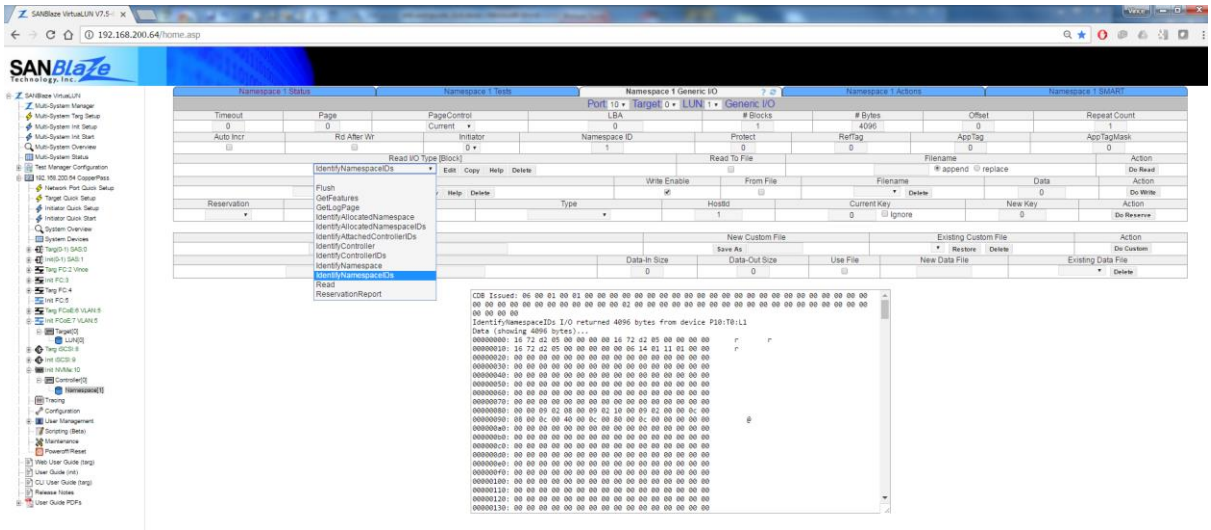


Figure 24: The NVMe Namespace Generic I/O Page

A number of NVMe commands can be issued directly from this page for the purpose of testing the response by a target, or to read and write data to a target device. Coupled with "trace mode" the *Generic I/O Test* page makes a powerful low level test tool, incorporating functions such as **Flush**, **Get Features**, **Get Log Page**, **Identify Namespace**, **Read/Write** a specific **LBA** and **Reservations**.

Basic operation of this page is to simply choose an I/O to issue. For example, under the **Read I/O** dropdown, **Identify Controller** was selected in the example above. Then simply select the **Do Read** button. The response from the target device will be displayed in the text box, as shown in the above example.

If a data **Read** is selected, the data at the selected **LBA** will be read from the device and displayed on the screen. There are no restrictions placed on your use of the parameters, such that it is possible to attempt to read (and display) up to 4TB of data on the screen. Such a large read will create havoc with your local web browser, and it is recommended you keep your reads small.

Not all fields apply to all I/Os. Use the table below to determine which fields are relevant to your selected command. Fields that are not required for a command are ignored. It is only required to fill in the fields that apply to the I/O you are currently issuing.

©Copyright 2025 SANBlaze. All Rights Reserved

Use the followiing tables as guides for required and optional fields for NVMe Read and Write I/O options.

KEY:

M=Mandatory
O=Optional
Blank=Not Applicable (ignored)

Read I/O	Page	Page Control	LBA	# Blocks	# Bytes	Offset	Namespace ID	T10 DIF
Flush								
GetFeatures	*M*	*M*					*O*	
GetLogPage	*M*				*M*		*M*	
IdentifyController								
IdentifyNamespace							*M*	
IdentifyNamespaceIDs							*M*	
IdentifyAllocatedNamespace							*M*	
IdentifyAllocatedNamespaceIDs							*M*	
IdentifyAttachedControllerIDs								
IdentifyControllerIDs							*M*	
Read			*M*	*M*				*O*
ReservationReport					*M*			

Read I/O Generic I/O Options

KEY:

M=Mandatory
O=Optional
Blank=Not Applicable (ignored)

Write I/O	Page	Sub Page	Page Control	LBA	# Blocks	# Bytes	Offset	Namespace ID	T10 DIF	Data
Compare				*M*	*M*				*O*	
DatasetDeallocate				*M*	*M*					
DatasetReadHint				*M*	*M*					
DatasetWriteHint				*M*	*M*					

©Copyright 2025 SANBlaze. All Rights Reserved

FormatNVM								*M*		
FormatNVMCryptoErase								*M*		
FormatNVMSecureErase								*M*		
SetFeatures	*M*							*O*		*M*
Write										
WriteUncorrectable										
WriteZeros										
ReservationAquire	See Reservation Helper Functions Described Below									
ReservationRegister	See Reservation Helper Functions Described Below									
ReservationRelease	See Reservation Helper Functions Described Below									

Write Generic I/O Options

NVMe Generic I/O Parameters

Fields that do not apply to the command you are issuing are ignored, and have no effect on the CDB issued. See the tables above for the applicability of each of these fields.

Timeout	Overrides internal timers of 10 seconds for read and 20 seconds for write with the value specified. Timeout is useful for commands known to take a long time, like rewind.
FID/LID	Hexadecimal number to read or write. For example, GetLogPage with Page set to 0x80 reads the device serial number page.
Select	Set for use with ModeSense/LogSense commands to indicate what kind of page to fetch. Select from one of the following values: Current Default Saved Supported
LogicalBlock	The starting logical block address.
# Blocks	Number of blocks to read/write (decimal integer).
# Bytes	Number of bytes (decimal integer) to read or write. Max and default are block size, min is 1. **Note:** Offset (see below) + *number of bytes* must not exceed the number of bytes in a block. For example, to read only byte 7, use # Bytes=1, Offset=7.

©Copyright 2025 SANBlaze. All Rights Reserved

Offset	Byte offset into a block (decimal integer). Max is BlockSize – 1, default is zero. This is the byte offset into the block where the read or write will start. For example, to read the last byte of a 512 byte block, use # Bytes=1, Offset=511. **Note:** Offset + *number of bytes* (see above) must not exceed the number of bytes in a block.

©Copyright 2025 SANBlaze. All Rights Reserved

Repeat Count	Number of times (decimal integer) to do this I/O. Default is 1, no max imposed, but use caution.
Auto Increment	After an I/O, set the next starting LBA to the ending LBA + 1. Combined with RepeatCount this feature can be used to access large ranges of data. For example, to read the first 1000 blocks of a drive, check AutoIncrement, and set RepeatCount to 1000. Using these two options it is possible to pre-set all data bytes on a device, or to set any N bytes in N blocks to a value. RepeatCount and AutoIncrement can also be combined with the "ReadToFile" and "WriteFromFile" options to read ranges to a file and write a file repeatedly over a range of LBAs.
Read After Write	Reads the range of LBAs just written, using the same CDB size (Write6 – Read6). This is a convenient way to display the data written; can be combined with "ReadToFile" and "WriteToFile".
Namespace ID	Defaults to the Namespace ID of the Namespace you have selected in the Left-hand Menu. -1 selects all Namespaces for commands that support this behavior 0 is for commands that do not require a namespace The Namespace ID varies with device implementation; you must read the specification for the device under test to determine behavior. If the device returns Invalid_NS, the value provided was incorrect for the command issued. See the tables above for which commands require NamespaceID.
Protect	T10 DIF Protect
RefTag(Hex)	T10 DIF RefTag
AppTag(Hex)	T10 DIF AppTag
AppTagMask(Hex)	T10 Dif AppTagMask
Read I/O Type	See Table 1 above for available Ccmmands and required parameters
Read To File	Results of a Read are written to a file rather than displayed on the web page.
Write I/O Type	See Table 2 above for available commands and required parameters
Write Enable	Enables writes to the selected namespace. If a write test is to be performed, this checkbox must be selected. The checkbox defaults to unchecked. **CAUTION!** No additional checking is done to verify the disk is available for writing. If you select the checkbox, writes will be done to the selected namespace. Select this checkbox only if you are sure the selected device is a "scratch disk"
Write From File	The data to be written should be fetched from a file.

©Copyright 2025 SANBlaze. All Rights Reserved

Filename	Selects the file that the Write results are written to. Select from one of the following values: random zero a file that was previously read from a device using read to file a custom binary file placed in */virtualun/lundata/initiator/FILENAME*
Data Byte	A single hex byte (for example, FF) that will be written over the range described by the values above. Using this option it is possible to write the same byte to an entire device, or a single byte anywhere in the read capacity size of the device. For example to write FF to the first 10 blocks of a device, set DataByte to FF, # of blocks to 10, select Write6 and then Do Write. To confirm, read the same range, or use the "ReadAfterWrite" feature.

Reservation Helper Functions

Reservation testing can be performed using the "Do Reserve" action, which is described below.

Note: You must do "SetHostID" first, before using any other reservation command.

Also, not all NVMe devices implement all reservation functions.

Reservation

Select the type of reservation to issue, then select an appropriate Reserve Action, Type, HostID, Current Key (or Ignore), and New key.

Use the **Do Reserve** button to issue the command.

Select from one of the following values:

SetHostId - (Must be done first)

Report - Read basic reservation information from the device

Register - Register a new key from the specified host ID

Acquire - Aquire a key (Reserve)

Release - Release a key

KEY:
M=Mandatory
O=Optional
Blank=Not Applicable (ignored)

Reservation	Reserve Action	Type	HostID	Current Key	New Key
SetHostID			*M*		
Report					
Register	*Register Key*		*M*		*M*
Register	*Unregister Key*		*M*	*M*	
Register	*Replace Key*		*M*	*M*	*M*
Acquire	*Acquire Reservation*	*M*	*M*	*M*	
Acquire	*Preempt Reservation*	*M*	*M*	*M*	
Acquire	*Preempt and Abort*	*M*	*M*	*M*	
Release	*Release Reservation*	*M*	*M*	*M*	
Release	*Clear Reservation*		*M*	*M*	

Persistent Reservation Actions

Definitions of Persistent Reservation Actions	
Reserve Action	List of actions as shown in the table above
Type Code	Select from one of the following values if Mandatory in table above: Write Exclusive Exclusive Access Write Exclusive Registrant Only Exclusive Registrant Only Write Exclusive All Registrants Exclusive All Registrants
HostID	Hex value uniquely identifying this registrant.
Current Key	Hex value of registered key you are changing. Use Ignore for the value while modifying keys.
New Key	Hex value for new key.
Action	Do Reserve, executes the reservation action.

©Copyright 2025 SANBlaze. All Rights Reserved

APPENDIX A: SBExpress-RM Riser Card Installation

The SANBlaze SBExpress-RM systems use Riser Cards to match specific functionality of each of the 16 slots to the drive type under test. The risers are Field Replaceable Units (FRUs) and are accessible by the user. Please follow the procedure outlined in this document to replace the risers.

Two supported Riser types include EDSFF to M.2 and U.2 to M.2:

60091500x - EDSFF to M.2
60092300x - U.2 to M.2
600913003 - Minimum 003 is compatible for Gen4 drives and NOT recommended

iRiser Gen 5 Product Design Characteristics

Controlling the FPGA on the iRiser

Extract the tar file to the system, and execute the following commands:

```
tar xzf iRiser.tz
```

This will extract the iRiser replacement program "iRiser"

```
cp iRiser /bin/iRiser
```

This will copy the iRiser program to /bin, overwriting the original program.

```
iRiser -d -3 show gpio -F 0
```

This will run the program with the following options:

`-d -3` (Selects the MB)
`show gpio` (Shows the current state of the 32 GPIOs with signal names)
`-F 0` (Force IO to go to the FPGA via the USB port)

You should see the following output:

```
Action: Sequence 512 address 0x200

Direction: I/O (UC) user writable signal, i/o (LC) signal locked

GPIO|31           24|23           16|15            8|7             0|

|B|P|D|D|T|T|T|D|S|S|S|S|S|S|S|S|D|D|D|D|D|D|D|D|D|D|D|D|D|D|D|D|

|i|w|i|i|P|P|P|r|W|W|W|W|W|W|W|W|u|u|u|u|u|u|u|u|u|u|u|u|u|u|u|u|

|t|r|s|s|3|2|1|v|8|7|6|5|4|3|2|1|a|a|a|a|a|a|a|a|a|a|a|a|a|a|a|a|
```

©Copyright 2025 SANBlaze. All Rights Reserved

```
    |3|O|U|W| | |
|R|D|S|S|F|C|U|E|E|1|1|1|1|1|1|1|1|1|1|1|1|1|1|1|1|1|
    |1|n|s|t| | |
|s|i|S|R|r|h|s|E|E|1|1|1|1|1|1|0|0|0|0|0|0|0|0|0|0|0|
    |N|R|b|c| | |
|t|s|C|I|c|a|l|S|S|5|4|3|2|1|0|9|8|7|6|5|4|3|2|1|0|
    |/|s|L|h| | |
|L|R|L|S|P|i|L|e|e|L|L|L|L|L|L|L|L|L|L|L|L|L|L|L|L|
    |C|t| |D| | | | |s| |L|w|n| |l|l| | |
| | | | | | | | | | | | | |
    | | | |g| | | | |t| | |r| | |1|0| | |
| | | | | | | | | | | | | |

├┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┤

|o|o|o|o|o|o|o|O|I|I|I|I|I|I|I|I|O|O|O|O|O|O|O|O|O|O|O|O|O|O|O|O|Dir:[0x214]0xff00ffff

├┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┤

|1|1|1|0|1|1|1|1|1|1|1|1|1|1|0|1|1|1|1|1|1|1|1|1|1|1|1|1|1|1|1|1|Val:[0x210]0xeffdffff

└┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┴┘
```

Your GPIO signals will differ, so you may not see 0xeffdffff for the value, but you should not see zero.

Once connection has been established as above, issue the following command to reset the FPGA back to factory defaults:

```
iRiser -d -3 show gpio -F 0 -f /etc/iRiser/RM6_GPIO_default.cfg
```

And you can issue the show gpio command again to verify:

```
iRiser -d -3 show gpio -F 0
```

Power the system off and back on again to make certain the new values are being used. When the system reboots, you should now see the FPGA on the PCIe bus, like this:

```
lspci |grep SANBlaze
2a:00.0 Signal processing management: SANBlaze Technology, Inc. Device 2005
(rev 61)
```

iRiser5 (standard)

+ Very fast control over PCIe for configuration and triggering, scalable with no contention

+ High performance power monitoring

- 12-bit A to D sampling power at 1M samples per second
- Ability to average any number of samples over any time (e.g., average for 250ms then upload)
- PCIe based direct data placement to host via DMA for very low host overhead.

+ High performance programmable signal control

+ Glitch Power, PERST, Clock as fast as 5ns

+ Fully programmable sequences with a period of 20ns and as long as hours per transition

- CLI and GUI based control for setting up complex power and reset sequences.
- Programmable power control for surprise/graceful removal at 20ns intervals

+ Multiple signal sequences can be pre-programmed for trigger via PCIe.

+ New sequences can be loaded from configuration files or from the GUI on the fly.

©Copyright 2025 SANBlaze. All Rights Reserved

- Single / Dual port mode fully under software control
- Nonvolatile configuration for power-up/start of day, no physical switch for initial state.
- Control via PCIe, a separate connection to each iRiser, up to 16 in the system

iRiser5G (upgrade)

All of the functionality above PLUS:

+ PCIe lane control

- Lanes can be disabled to test x1, x2, x4
- Lane drop to or from device
- Simulated drive removal/replacement

+ PCIe Error Injection

- PCIe lanes can be "glitched" as fast as 5ns
- Glitch control to or from the device
- Complex glitch sequences, any line any glitch length

Follow the procedure outlined in this document to replace the risers.

Toggling Dual Port on the Webpage

Clicking the dual/single check box leaves the device in a partially linked state.

- From single to dual:

  lower port linked x2
  upper port not linked (but present)

- From Dual to single:

  lower port linked x2
  upper port not linked not present

If the power to the slot is OFF, set/clear the dual bit in the iRiser. When you enable power, the drive links correctly.

If the power to the slot is ON, and the mode is changing (single <-> dual), shut the power to the slot off and then set/clear the dual bit in iRiser. Turn the power to the slot back on.

If you want to test the old behavior of just setting the dual/single bit to the drive, use the following:

- Single port to dual port:

  `iriser clear Dual00L (DualNNL where NN=slot number)`

- Dual port to single port:

  `iriser set Dual00L (DualNNL where NN=slot number)`

DT5V2 Configuration File

In Version 2, of the DT5, a new default config file has been created. The syntax for this config file is:

```
iriser -f /etc/iRiser/DT5_GPIO_default.cfg
```

©Copyright 2025 SANBlaze. All Rights Reserved

This file sets the default GPIOs correctly and writes them to the eeprom behind the FPGA.

iRiser also defines four signals as INPUTS (bits 4-7) but also allows you to force them to outputs and set/clear them.

The new syntax "input" (or in) and "output" (or out) sets a bit to the input or output. This is used for top slot power, where the control bits must be set to Input by default or the iRiser will override the present and enable signals.

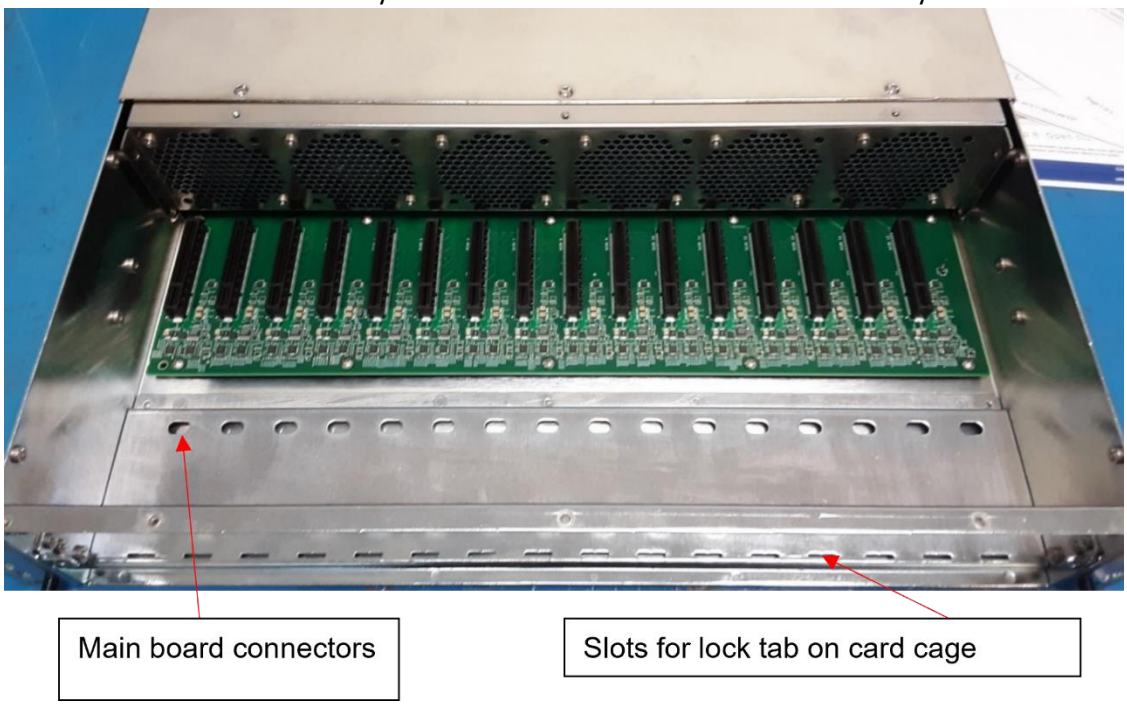
You can force PRESENT and/or POWER for the top slot by defining bits 4 and 5 to outputs and clearing them.

Like set and clear, you can specify a bit number or name, and the name can be UC or LC, and only needs to be unique (can be abbreviated as long as it is unique). The following are syntax examples:

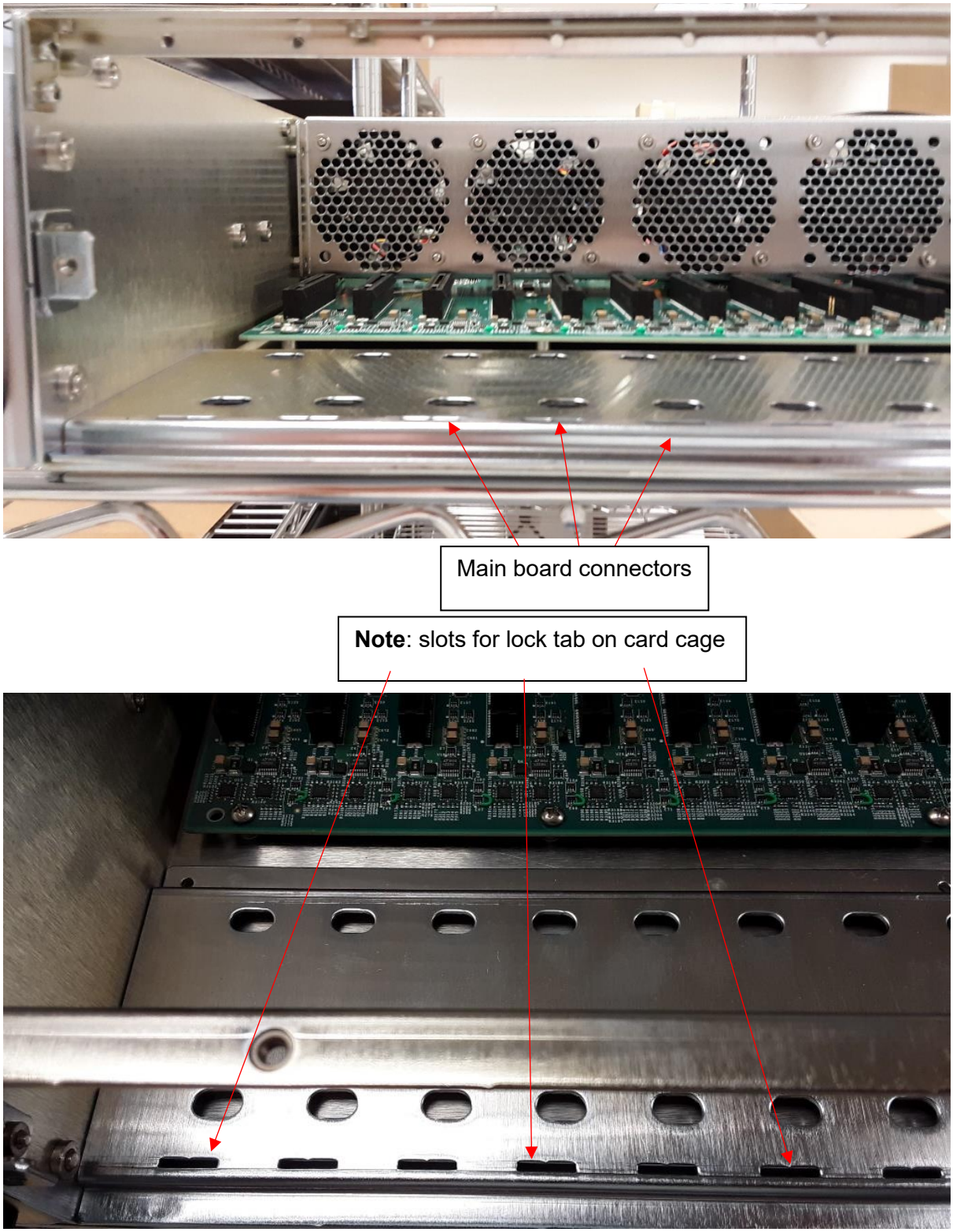
```
iriser out d3powerl

iriser out d3po

iriser out 4

iriser in 4
```

Installation Steps

1. On the back of the SBExpress-RM unit, turn the power switch to the OFF position and unplug the power cord.
2. To install a Riser card into the SBExpress-RM6 it is necessary to remove the top cover and the front cover by removing the screws securing the covers. ***** **Note: Use hand tools ONLY to remove screws*****
3. Once the cover is removed you will have access to the main PCB assembly.

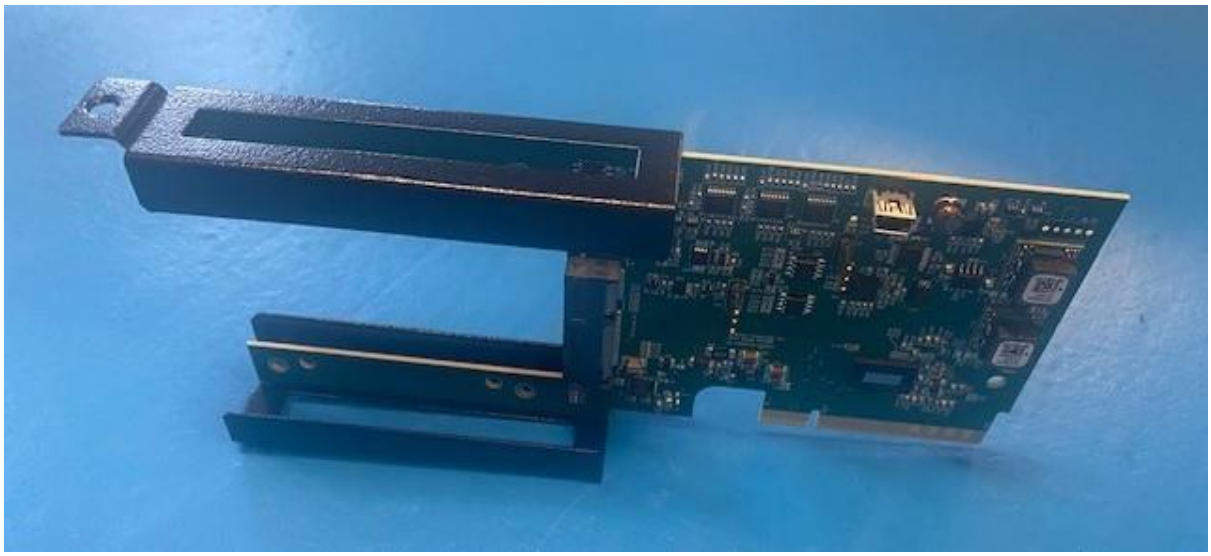


4. Slots in chassis to accept card cage lock tab.



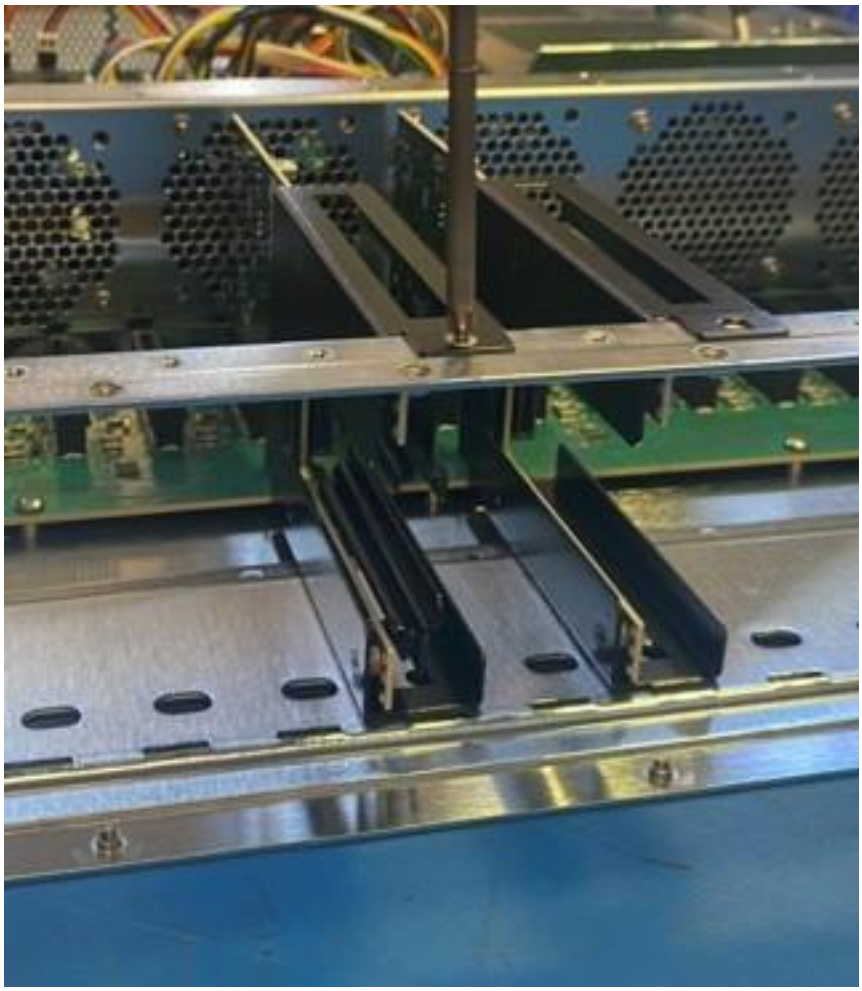
©Copyright 2025 SANBlaze. All Rights Reserved

5. Riser card assembly.



SBExpress-RM6 Riser Card

6. Position the riser card over the board connector as shown. Apply even downward pressure to insert the riser board into the main board connector.



©Copyright 2025 SANBlaze. All Rights Reserved

7. Install 2-56 x 3/16” Philips panhead screw thru the chassis frame.to secure each riser card. ***** Note: Use hand tools ONLY to install the screws. Hand tighten ONLY*****

8. Re-install front cover using (10) 6-32x1/4” Phillips panhead screw; 6pc on the top, 2pc each on the side. ***** Hand Tighten ONLY*****

Contact SANBlaze

If you have any questions, contact SANBlaze technical support by emailing support@sanblaze.com or call us @ (978) 679-1400.

©Copyright 2025 SANBlaze. All Rights Reserved

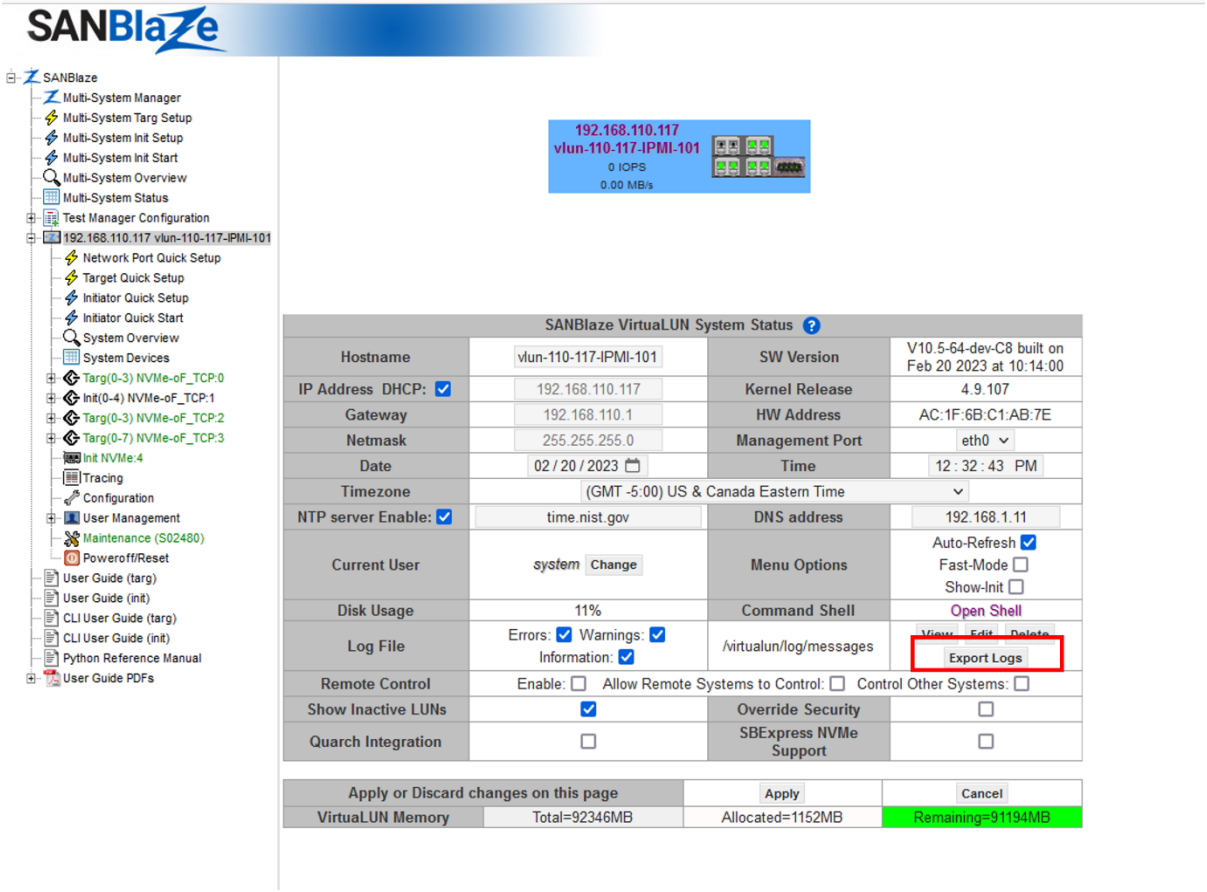
APPENDIX B: Troubleshooting & Product Support

If you run into any issues, you can export the log files on the system and/or log a support ticket to request help from SANBlaze support.

Export the Log Files

Export the log files on the system to view the system history. By reviewing the log files, it may become evident what the issue is and/or, the log files can help you narrow it down. On the main SBExpress GUI page, click Export Logs (see below).

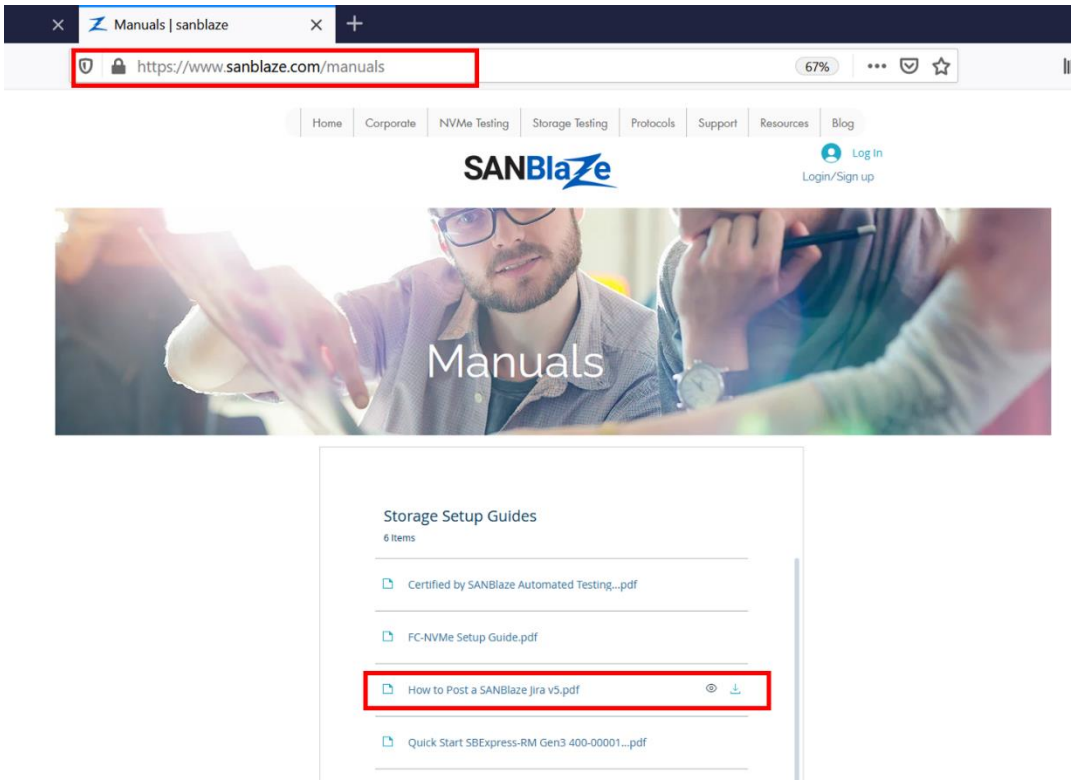
©Copyright 2025 SANBlaze. All Rights Reserved



Export Log Files

Open a Support (Jira) Ticket

If looking at the log files is not helpful, you can open a Jira ticket and SANBlaze support will respond. See our post "How to Post a SANBlaze Jira" for more information (location shown in the figure below).



How to Post a SANBlaze Jira

Contacting SANBlaze

Need more help? Contact us through phone or email (see below) or visit our website at https://www.sanblaze.com/

SANBlaze Technology, Inc.
One Monarch Drive, Suite 204
Littleton, MA 01460
(978) 679-1400
info@sanblaze.com
Help Center | Sales | Support
Technical Support is available from SANBlaze in four forms:

Visit our Help Center:

Visit our Help Center to log an issue, request a feature, or request help for a specific issue.

Email: Send an email detailing your issue to: support@sanblaze.com

Web: Visit our website at https://www.sanblaze.com/storage-testing-support

Phone: Call SANBlaze at 1-978-679-1400 from 8:30 AM to 5:00 PM EST Monday through Friday.

©Copyright 2025 SANBlaze. All Rights Reserved