Western Digital.

User Guide

RapidFlex™ A2000 Target, Fabric Bridge Device

Part Number(s) 1K00041, 1K00043
Firmware A2000Tfw_2.0.0
Document D018-000613-000
Revision 02
February 2024

Table of Contents

Revision History

Date	Revision	Notes
October 2023	01	Initial release
February 2024	02	• Added Logging to Remote Server *(page 181)* • Updated Bridge Status *(page 70)* CLI output, object format, and field definitions • Updated Bridge Version *(page 73)* object format and field definitions • Updated Authorization Config *(page 118)* CLI syntax, object format, and field definitions • Updated Async Notification *(page 122)* object format and field definitions • Added Connection Recovery *(page 20)* • Added Downloading Firmware *(page 209)*

Notices

Western Digital Technologies, Inc. or its affiliates' (collectively “Western Digital”) general policy does not recommend the use of its products in life support applications wherein a failure or malfunction of the product may directly threaten life or injury. Per Western Digital Terms and Conditions of Sale, the user of Western Digital products in life support applications assumes all risk of such use and indemnifies Western Digital against all damages.

This document is for information use only and is subject to change without prior notice. Western Digital assumes no responsibility for any errors that may appear in this document, nor for incidental or consequential damages resulting from the furnishing, performance or use of this material.

Absent a written agreement signed by Western Digital or its authorized representative to the contrary, Western Digital explicitly disclaims any express and implied warranties and indemnities of any kind that may, or could, be associated with this document and related material, and any user of this document or related material agrees to such disclaimer as a precondition to receipt and usage hereof.

Each user of this document or any product referred to herein expressly waives all guaranties and warranties of any kind associated with this document any related materials or such product, whether expressed or implied, including without limitation, any implied warranty of merchantability or fitness for a particular purpose or non-infringement. Each user of this document or any product referred to herein also expressly agrees Western Digital shall not be liable for any incidental, punitive, indirect, special, or consequential damages, including without limitation physical injury or death, property damage, lost data, loss of profits or costs of procurement of substitute goods, technology, or services, arising out of or related to this document, any related materials or any product referred to herein, regardless of whether such damages are based on tort, warranty, contract, or any other legal theory, even if advised of the possibility of such damages.

This document and its contents, including diagrams, schematics, methodology, work product, and intellectual property rights described in, associated with, or implied by this document, are the sole and exclusive property of Western Digital. No intellectual property license, express or implied, is granted by Western Digital associated with the document recipient's receipt, access and/or use of this document or the products referred to herein; Western Digital retains all rights hereto.

Western Digital, the Western Digital design, the Western Digital logo, and RapidFlex are registered trademarks or trademarks of Western Digital Corporation or its affiliates in the US and/or other countries. Broadcom is among the trademarks of Broadcom. Cisco and Nexus are registered trademarks or trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries. NVIDIA, Mellanox, ConnectX, and Spectre are registered trademarks or trademarks of NVIDIA Corporation. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries. The NVMe, NVM Express, and NVMe-oF word marks are trademarks of NVM Express, Inc. PCIe® is a registered trademark and/or service mark of PCI-SIG in the United States and/or other countries. Windows is a trademark or registered trademark of Microsoft Corporation in the United States and/or other countries. All other marks are the property of their respective owners.

One terabyte (TB) is equal to one trillion bytes. One megabyte (MB) is equal to one million bytes. Actual user capacity may be less due to operating environment. Not all products are available in all regions of the world. Pictures shown may vary from actual products. Product specifications subject to change without notice.

Western Digital
5601 Great Oaks Parkway
San Jose, CA 95119

Points of Contact

For further assistance with a Western Digital product, contact Western Digital Datacenter Platforms technical support. Please be prepared to provide the following information, as applicable: part number (P/N), serial number (S/N), product name and/or model number, software version, and a brief description of the issue.

Website:
https://portal.wdc.com/s/

Email:
enteprisesupport@wdc.com

UK Import Representation Contact

Western Digital UK Limited
PO Box 471
Leatherhead KT22 2LU
UK

Telephone: +44 1372 366000

EU Import Representation Contact

Western Digital EU Limited
PO Box 13379
Swords, Co
Dublin, Ireland

Overview

In This Chapter:

1.1 Purpose

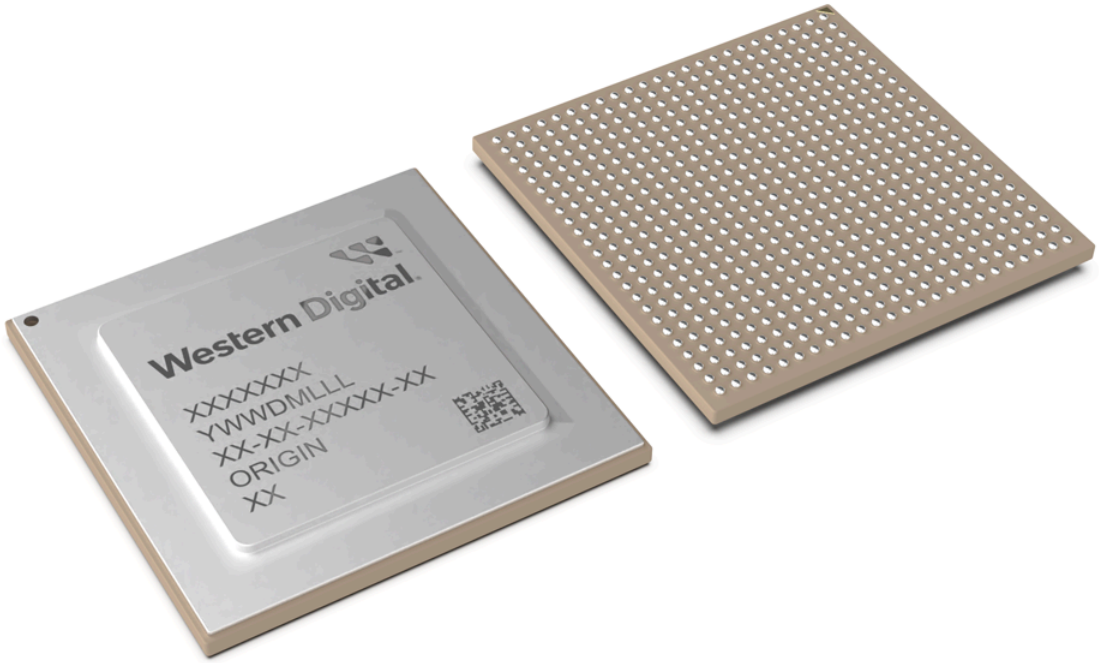
This guide provides information and use instructions for the following device(s):

- RapidFlex A2000 Target, Fabric Bridge Device

Note: The RapidFlex C2000 Target (add-in-card) contains the A2000 Target (ASIC). For C2000 Target use instructions, see the *RapidFlex C2000 Target, Fabric Bridge Adapter User Guide*.

1.2 RapidFlex A2000 Target Description

Figure 1: RapidFlex A2000 Target



Western Digital's RapidFlex A2000 Target is a second generation fabric bridge device that implements the NVMe-oF™ protocol. As an NVMe-oF target in a storage appliance, the device receives NVMe-oF commands from network clients over Ethernet, translates them into NVMe® commands, and transmits them to PCIe® attached NVMe devices (SSDs).

When used in a storage appliance that is connected to an Ethernet Switch, the A2000 Target can support many to many connectivity. Many clients can connect through the A2000 Target to the same namespace on a single SSD, to one or multiple namespaces on the same SSD, or to multiple namespaces on multiple SSDs. This characteristic differentiates Ethernet attached storage enclosures from traditional SAS attached enclosures, which can only connect to a fixed number of hosts limited by the number of SAS ports on the enclosure.

The NVMe-oF protocol defines two methods of transport over Ethernet: RDMA and TCP. The A2000 Target supports both RDMA (RoCE v2) and TCP. Each of these transport protocols are implemented completely in silicon logic, without any firmware code in the high-speed data path. A complete state machine-based implementation provides extremely low latency performance, minimizing per-I/O network overhead.

The A2000 Target is an NVMe-oF standards-compliant solution, which helps enable interoperability with any other industry devices that also strive to be standards-compliant.

1.3 RapidFlex A2000 Target Features

System Features

- NVM Express® base specification revision 1.4 support
- NVMe over Fabrics specification revision 1.1 support
- NVMe-oF on RoCE or TCP (not both at the same time)
- Performance monitors for Ethernet ports

NVMe Features

- Supports up to 120 virtual controllers, allowing up to 120 hosts to share the NVMe device
- Up to 3776 RoCE or 832[1] NVMe-TCP Fabric I/O connections
- Up to 64 entries per NVMe-oF connection (depends on selected profile)
- Up to 32 NVMe SSDs
- Up to 32 namespaces per SSD[2]
- Supports metadata pass-through
- Supports 512- and 4096-byte logical block sizes
- Supports 8-, 16-, 64-, and 128-byte metadata[3]
- Persistent Discovery Controller Connection

RDMA Features

- All hardware logic RoCE (v2) implementation
- Up to 1568 total memory regions: 1536 IO and 32 Admin
- RoCE Reliable Connection Transport

TCP Features

- All hardware logic NVMe-TCP implementation (TCP Offload Engine)
- Implements *NewReno* version of TCP
- Embedded CPU processing of slow path traffic

1. NVMe-TCP supports 1024 total connections. With 832 IO connections in NVMe/TCP mode, the A2000 Target can only support 192 virtual controllers/hosts.
2. Limited to 16 if Reservation or Host-Authorization is enabled
3. 8- and 16-byte metadata is supported for 512B blocks; 8-, 16-, 64-, and 128-byte metatdata is supported for 4096B blocks.

Ethernet Features

- Two 100 Gbps Ethernet ports
- Reduced power when using a single Ethernet port
- One MAC address per port
- One of 4K VLAN addresses per port (ID range: 1-4095)
- One IPv4 / one IPv6 address per port
- Hardware Link Aggregation
- Jumbo Packets up to 4500 bytes

Network Services

- ARP
- ICMP
- SNTP
- LLDP

Flow Control / Congestion Management Features

- Supports ECN for TCP and RoCEv2
- Supports DC-QCN
- Global Pause, PCP-PFC, and DSCP-PFC
- Includes a programmable congestion notification engine

PCIe Features

- PCIe Gen4 x16
- Reduced power when using only x8 or x4 lanes
- Supports external PCIe fabric switching

Peripherals

- 1G Ethernet port for connection to a management network. External interface is SGMII
- SPI Flash
- One optional MIIM interface for 1G PHY
- One master I2C for QSFP management
- One slave I2C for BMC and EEPROM access
- Two general purpose master/slave I2C
- 32 GPIO pins

Reliability, Availability, Serviceability (RAS) Support

- Internal memories ECC protected
- All internal data / control path parity protected
- Soft error retry / recovery
- First time debug data capture on error

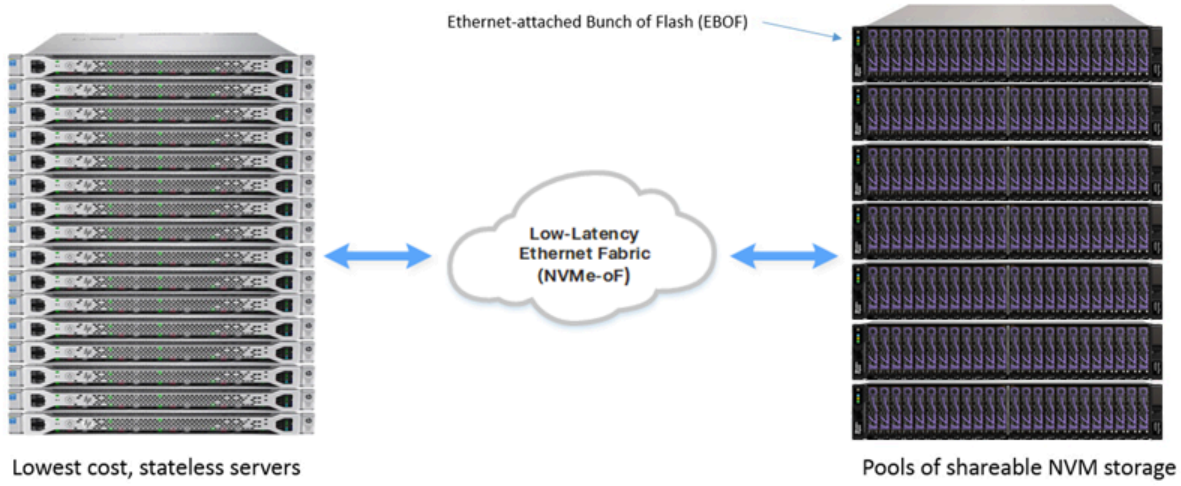
Use Cases

This chapter presents a typical use case for the A2000 Target.

In This Chapter:

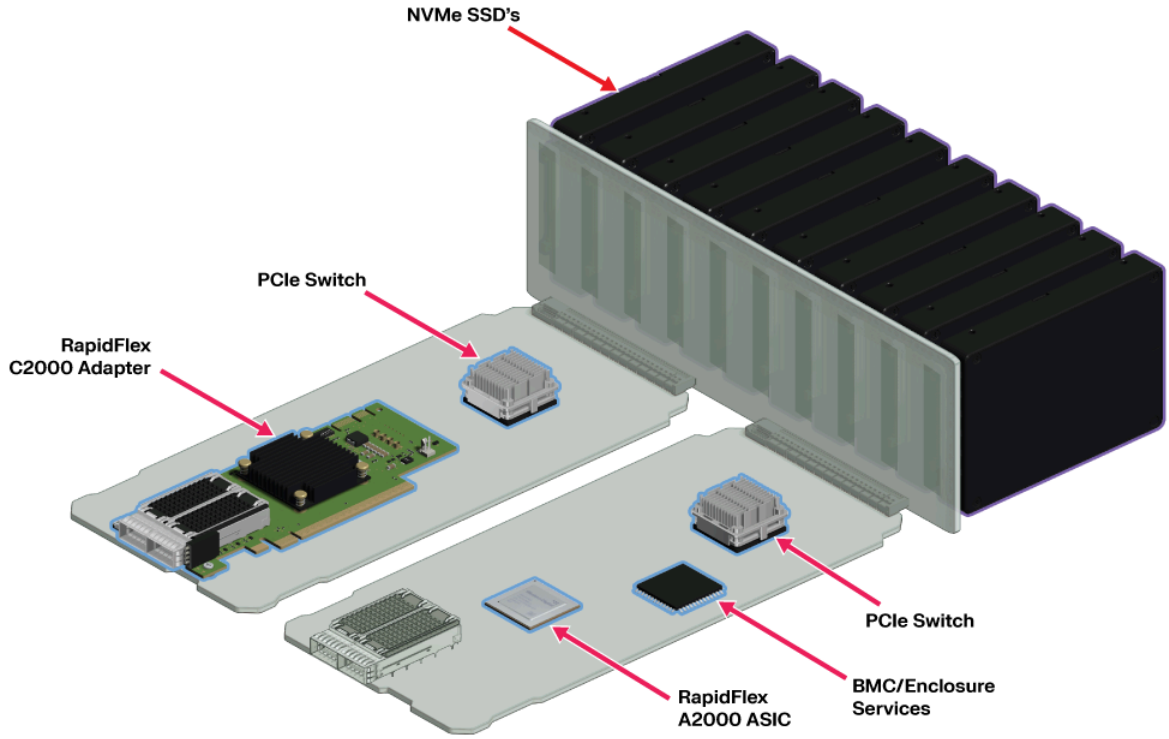
2.1 Typical Disaggregated Data Center Architecture

Figure 2: Example of Composable Infrastructure Architecture



2.2 Ethernet-Attached Bunch of Flash (EBOF)

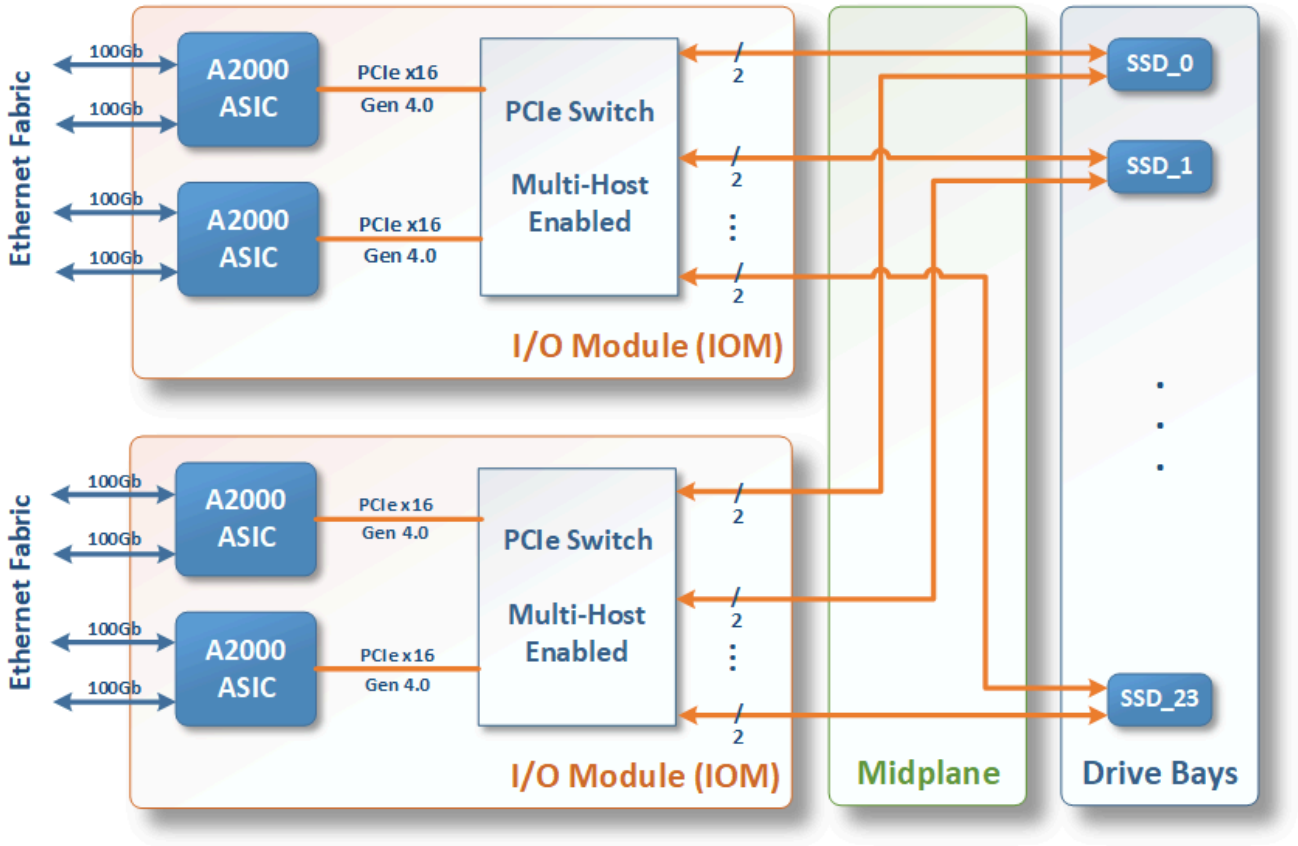
Figure 3: Generic EBOF Electronics



This image image shows the C2000 Target (or the A2000 Target) implementing the bridging function between an Ethernet-based fabric and NVMe SSDs, typically through one or more PCIe switches.

2.3 I/O Block Diagram - HA Configuration

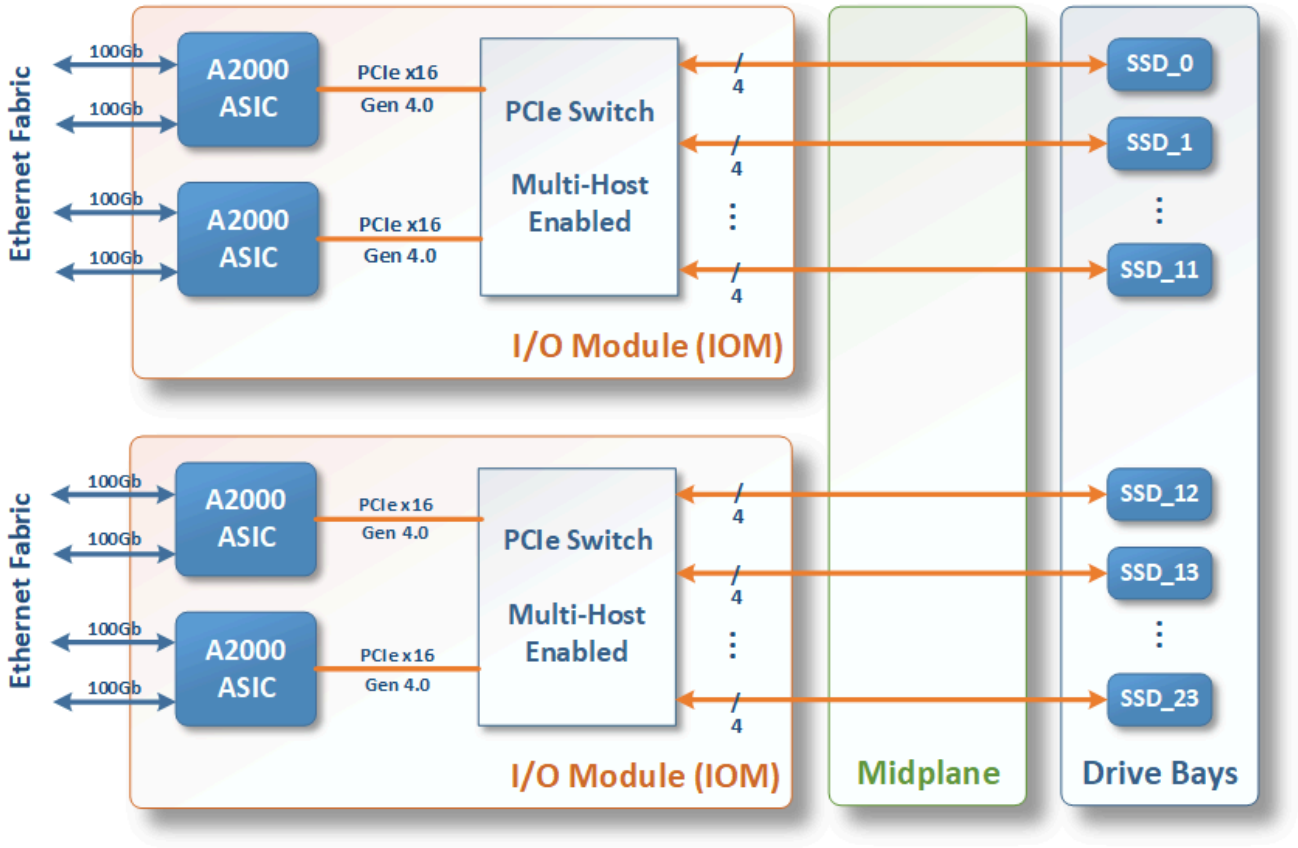
Figure 4: Generic EBOF HA Block Diagram



In a Highly Available (HA) configuration, typically a x2 PCIe connection is made from each I/O Module (IOM) to each SSD. In this topology, the failure of any component on a single IOM (or the IOM itself) can be compensated for by rerouting traffic via the second IOM. However, there may be a compromise in performance, as only two PCIe lanes are typically active with traffic to each SSD.

2.4 I/O Block Diagram - Non-HA Configuration

Figure 5: Generic EBOF Performance Block Diagram



In this non-HA configuration, a full x4 PCIe connection is made to each SSD, enabling higher performance. In this topology, the failure domain is likely to be at the shelf level rather than at a component level.

Compatibility

In This Chapter:

3.1 Compatible Initiators

The following table lists initiator devices compatible with the A2000 Target.

Table 2: Compatible Initiators

Manufacturer	Type	Model	Protocol(s)
NVIDIA®	Mellanox® ConnectX®-5	MCX556A-ECAT	TCP
		MCX516A-CCAT	RoCE v2, TCP
		MCX516A-CDAT	RoCEv2
	Mellanox ConnectX-6	MCX614106A-CCAT	RoCE v2, TCP
		MCX623106AN-CDAT	RoCE v2, TCP
Western Digital	RapidFlex	A2000 Initiator	RoCE v2, TCP
		C2000 Initiator	RoCE v2, TCP
SANBlaze	N/A	V10.5-64-Beta15-C8	RoCE v2, TCP
Broadcom®	N/A	P2100G	RoCE v2

3.2 Enclosure Requirements

This section provides requirements and recommendations for the storage enclosure containing the A2000 Target.

Enclosure Features

Feature	Requirement
PCIe Compatibility	Gen3, Gen4
Power	~10W

3.3 Compatible Operating Systems

The following table lists operating systems compatible with the A2000 Target.

Table 4: Compatible Operating Systems

OS	Version
Ubuntu® Server	**20.04.3 LTS** Kernel: 6.6.x

3.4 Compatible Ethernet Switches

The following table lists Ethernet switches compatible with the A2000 Target.

Table 5: Compatible Ethernet Switches

Manufacturer	Model	FW Version
NVIDIA®	Spectrum® SN2100	3.10.4302
	Spectrum SN2700	3.10.4302
	Spectrum SN3700	3.10.4302
	Spectrum SN4700	3.10.4302
Cisco®	Nexus® 3000 C3232C	NXOS: 9.3.3
	Nexus 9000 C9332D-GX2B	NXOS: 10.2(3)
Arista®	7050CX3-32S-R	4.27.0F

3.5 Compatible Cables

The following table lists cables compatible with the A2000 Target.

Table 6: Compatible Ethernet Cables

Manufacturer	Type	Model	Length
NVIDIA / Mellanox	Active	MFA1A00-C003	3m
		MFA1A00-C005	5m
		MFA1A00-C010	10m
		MFA1A00-C050	50m
	Passive	MCP1600-C001E30N	1m
		MCP1600-C002E30N	2m
		MCP1600-C003E30L	3m
		MCP1600-C005	5m

Table 7: Serial Console Port Cable Requirements

Supported Cable Type	Supported Standard
micro USB / micro-b	USB 2.0

3.6 Compatible Software

The following table lists software compatible with the A2000 Target.

Table 8: Compatible Software

Software	Version
NVMe-CLI	1.16 and later

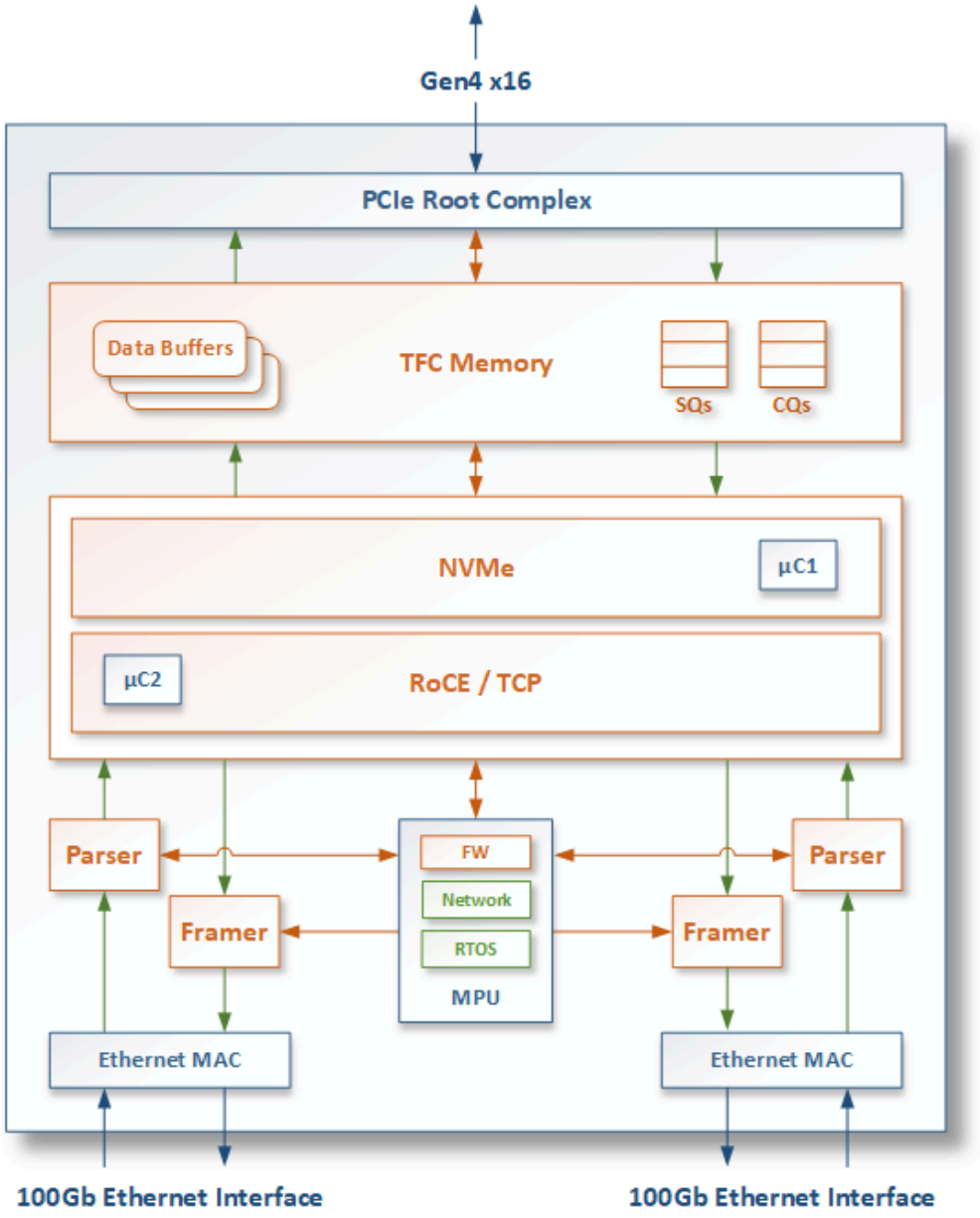
Architecture and Command Processing

In This Chapter:

4.1 A2000 Target Internal Block Diagram

A2000 Target Block Diagram

Figure 6: A2000 Target Internal Block Diagram



Block Descriptions

- **PCIe Root Complex** – Performs the industry standard functions of a PCIe Root Port, including interfacing to PCIe switches or SSDs directly.

- **TFC Memory** – The Target Fabric Controller memory block contains several different memories, including data buffer memory for I/Os in-flight, and the Submission and Completion circular queues associated with the NVMe devices.
- **NVMe** – This protocol block emulates the NVMe Target functionality required to transform host NVMe commands into NVMe-oF commands to remotely attached NVMe devices, allowing them to work in a standard, plug-and-play manner. This block also contains some programmability via the embedded microcontroller (µC1). See Programming *(page 215)* for more information.
- **RoCE / TCP** – These virtual blocks represent the Transport and RDMA functional blocks within the A2000 Target. All protocols are implemented in hardware, such that the entire I/O flows through the A2000 Target without encountering any software- or firmware-based paths (green arrows). Error and exception processing is done via the embedded microcontroller (µC2).
- **MPU** – The internal Management Processing Unit implements various management and non-performance tasks, such as:
  - Start-of-the-day: Initialization of the A2000 Target, self-test, networking configuration, PCIe bus configuration
  - Connection to NVMe-oF device(s)
  - SSD SMART event monitoring
  - Management of all host Admin queues
  - Management of TCP or RDMA connections from host to NVMe-oF Target controllers
  - Networking services (e.g. ARP, ping, ICMP)
  - Error and exception processing, including logging functions
- **Parser** – This inbound parsing block performs an internal routing function within the A2000 Target, directing expected I/O frames to the protocol blocks and other frames to the internal MPU.
- **Framer** – This outbound framing block transmits frames onto the Ethernet link as requested by the protocol blocks or the internal MPU.
- **Ethernet MAC** – The Ethernet Media Access Controller implements the 100/50/25Gb Ethernet interface to the fabric. This block includes other standard Ethernet functions such as the Physical Coding Sublayer (PCS) and Forward Error Correction (FEC), as needed for efficient operation.

4.2 A2000 Target Implementation

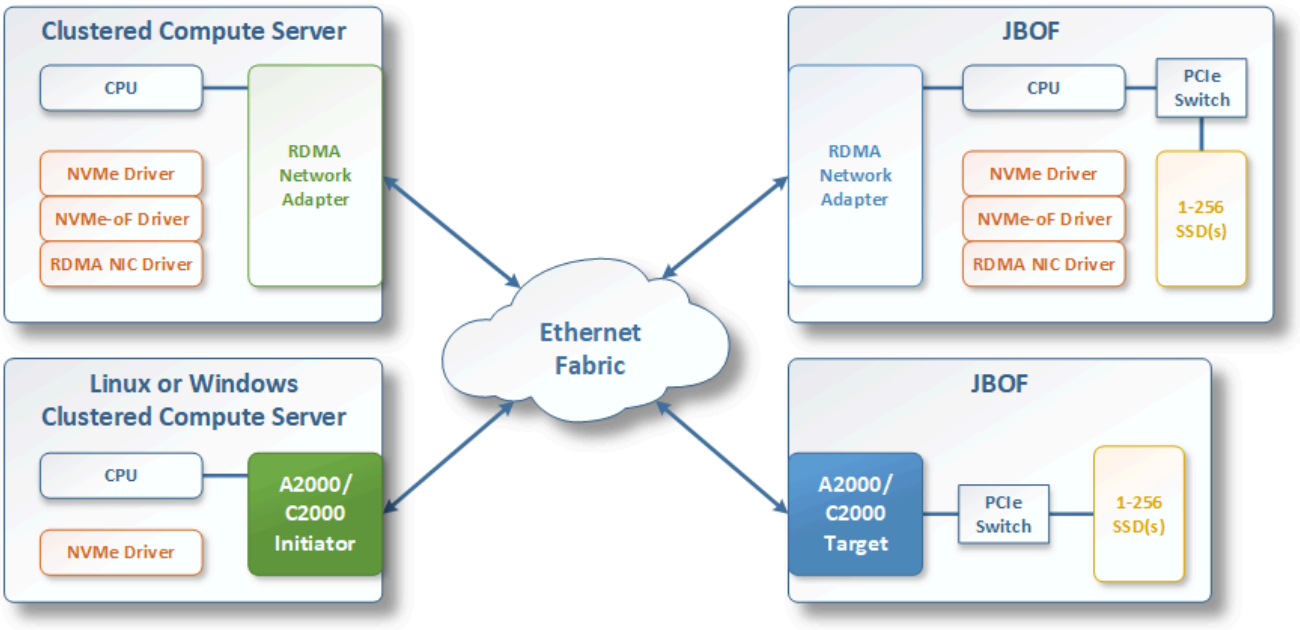
JBOF Shelf of Drives with a Software Target

In the following diagram, the top-right JBOF implementation uses an RDMA network adapter as a software target, connecting to multiple SSDs and requiring multiple drivers and use of the CPU.

JBOF Shelf of Drives with Target Bridge

The bottom-right section of the following diagram depicts an implementation using the A2000 Target, interfacing through a PCIe switch to many (up to 256) SSDs. This can amortize the cost of the A2000 Target across several drives. However, for full JBOF performance, 4-8 SSDs are usually mapped to a single A2000 Target. And if there are more SSDs in the enclosure, it is expected that a number of A2000 Targets are deployed in the enclosure to match the network to SSD bandwidths.

Figure 7: A2000-Based Disaggregated Storage



4.3 Functional Overview

The topics in this section provide a functional overview of the data flow, initialization, and connection process for the A2000 Target.

4.3.1 Basic Data Flow

The A2000 Target has two basic data flows:

- NVMe traffic running on an established RDMA connection
- Everything else, including the RDMA connection establishment traffic

The non-NVMe connection traffic is received by the MAC and routed directly to the MPU, which is responsible for processing all non-NVMe Ethernet traffic. This includes frames used for all the various Ethernet Network services and the connection frames used to establish an RDMA connection.

The MPU handles RDMA connection establishment. Once an RDMA connection exists, the parser will route all frames associated with that connection to the upper layer protocol engines. These engines handle the TCP flow control for the connection, perform the appropriate RDMA processing, and manage the NVMe device interactions via TFC Memory.

4.3.2 Initialization

Since the A2000 Target is designed to operate as a standalone device, it does not require any external initialization or programming. When it powers up, it will go through the following steps to bring the various modules and functions online:

1. Load MPU code from external Flash

Note: Flash code is checksum protected.

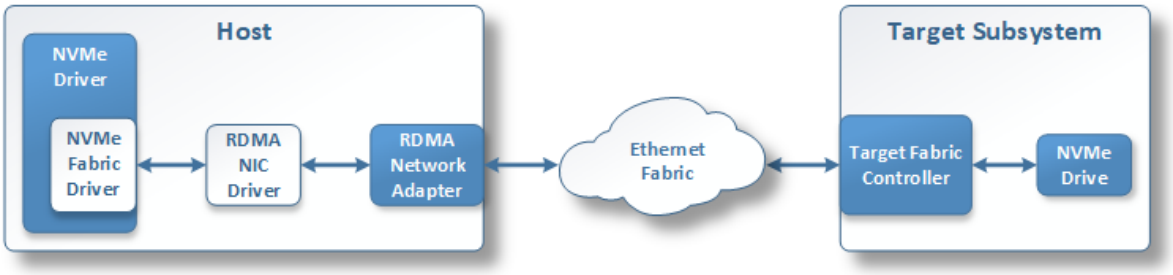
2. Self-test all memories
3. Initialize all memories
4. Load network addresses into Rx Parser/Router

Note: IP address(es) may be assigned after the MPU comes up and talks with the DHCP server, if not statically assigned.

5. Bring up PCIe bus and perform enumeration
6. MPU starts its main code and is ready to accept commands
7. MPU may communicate with NVMe drive(s)

Once the A2000 Target has finished initialization, it waits for the first communication from the host. The MPU may perform a number of operations to establish itself on the Ethernet fabric, such as registering itself with the local switch and obtaining an IP address from a DHCP server. This is all under the control of the A2000 Target's firmware and does not affect hardware initialization.

Figure 8: Fabric-Attached NVMe Systems



4.3.3 Connect Process

The Connect process is used by an NVMe-oF host to connect to the NVMe-oF target. It establishes a network and RDMA or NVMe/TCPconnection to the target and creates an NVMe Submission/Completion queue pair (either Admin or I/O).

The process of establishing connections is accomplished with the following steps:

Table 9: RDMA & NVMe/TCP Process

Steps	RDMA	NVMe/TCP
1		The NVMe-oF host sends a TCP sychronize (SYN) request to the A2000 Target.
2		The A2000 Target replies with a TCP synchronize-acknowledge (SYN-ACK) to the host.
3		The host completes handshake by sending an acknowledge (ACK) to the target.

Steps	RDMA	NVMe/TCP
4	The NVMe-oF host sends a RoCE Management Datagram (MAD) Connection Request to the A2000 Target.	The host sends an Initialize Connection Request (ICReq) protocol data unit (PDU) to the target.
5	The A2000 Target replies with a RoCE MAD Connection Response to the host.	The target responds with an Initialize Connection Response (ICRsp) PDU to the host.
6	The host completes the connection by sending a RoCE MAD Ready to Use (RTU) to the target.	The host completes the connection by acknowledging the ICRsp.
7	The host sends a Fabric Connect request to the A2000 Target.	
8	The A2000 Target replies with a Fabric Connect response. This completes the NVMe Admin QP.	
9	If the host wants one or more I/O Queue pairs, it performs Connect commands for each one.	

When the host creates an Admin Queue pair and at least one I/O Queue pair, it can then start performing data reads and writes to the device by sending NVMe commands.

Note: The entire Connect process is accomplished by A2000 Target's internal MPU and associated firmware. Once one or more I/O Queue Pairs is established, communication for each queue pair is entirely handled by the ULP hardware.

4.3.4 Connection Recovery

As per the NVM Express Base Specification[4] related to Set Features "Number of Queues (FID 0x7)": "when a host application requests a number of queues, the value allocated may be smaller or larger than the number of queues requested, often in virtualized implementations. The controller may not have as many queues to allocate as are requested."

The chances of getting a smaller number of queues may be more likely in configurations that approach the maximum allowed Admin and I/O connections for a given target. The reason is that when connections are interrupted, the resources on both the target and initiator begin their recovery process. At this point, when the initiator begins to reconnect, the target may still be in the process of cleaning up previous connection resources, so the initiator "Number of Queues" request may return a smaller number of queues than was expected.

Therefore, if there is a host application requirement to retain the same number of I/O queues, per admin queue, across connection loss events, here are some options to help minimize the issue described above:

- Reduce the number of Admin and I/O connections to allow headroom for recovery, per the following table:

4. See NVM Express Base Specification, Revision 1.4, June 10, 2019

Table 10: Reduced Connections

Profile	Admin Conns	I/O Conns	KATO	Host io_timeout
1	64	64	12 seconds (with A2000 Initiator)	30 seconds
2	64	448	If AdCon > 64: 60 sec Else: 5 sec (default)	RoCE: 12 seconds NVM/TCP: 30 seconds
3	120	832	If AdCon > 64: 60 sec Else: 5 sec (default)	30 seconds
4	120	3776	If AdCon > 64: 60 sec Else: 5 sec (default)	30 seconds

Caution: For larger profiles (i.e. #2, 3, or 4), if the number of Admin connections is expected to grow beyond 64 in the future, set KATO to 60 seconds to begin with.

In some cases (e.g. heavy I/O load conditions), even this number of connections may experience a reduced number of queues after recovery. The only way to ensure the same number of connections after recovery is to reduce connections to half the maximum number of allowed connections - this gives plenty of headroom for cleaning up previous connections in a recovery scenario.

- Use shorter KeepAlive times to ensure quicker cleanup of target connection resources after loss of connection, and to better synchronize initiator/target cleanup. However, this creates the risk of not allowing sufficient time for normal operations.

4.4 High-Availability Features

The A2000 Target's architecture is based on enterprise-level design principles for RAS (Reliability, Availability and Serviceability). This includes the following:

- All data and control paths in the chip are (even) parity protected.
- All data paths have full overlapping data protection throughout the ASIC.
- All memories are protected by ECC (Error Correcting Code) unless byte- or bit-level access is required, in which case parity protection will be used.
- CAMs (Context-Addressable Memories) are parity protected from false hits and misses.
- The device is designed to recover from soft errors, ensuring that errors will not result in data corruption.
- All errors are logged in an internal log buffer. This log buffer is periodically written to Flash for permanent record of errors. This log buffer will also be flushed to Flash in the event of a non-recoverable error.
- Error injection is implemented for all memories and interfaces to verify correct recovery behavior from a soft error.
- Code images in external (local) Flash are checksum-protected.
- Fail-safe FW download: Two code images are stored in the local Flash to allow for one to be corrupted and still enable the A2000 Target to boot successfully. Additionally, a third, non-writable "golden" image is present.
- Local Flash is used to dump error logs to for failure analysis in a factory return.

4.5 DIF Support

The A2000 Target supports the use of data integrity fields (DIF) in order to protect user data from undetected corruption. The A2000 Target does not check or insert DIF values, but rather allows them to pass from host to drive and back.

The following block and metadata combinations have been successfully tested:

- 512B logical block
  - Metadata size = 8B
  - Metadata size = 16B
- 4KB logical block
  - Metadata size = 8B
  - Metadata size = 16B
  - Metadata size = 64B
  - Metadata size = 128B

4.6 NVMe I/O Command Set

This section describes the processing of I/O commands in the NVM command set by the A2000 Target.

The A2000 Target behaves as a transparent bridge for NVMe commands posted by the host to a device/drive. Internal hardware-implemented protocol engines modify the NVMe-oF data pointer fields from keyed SGLs to a single PRP entry in the command posted to the drive.

Other fields in the command are not modified with the following exception: When the length of an I/O is greater than 4KB long, the A2000 Target will break the request up into a number of 4KB transactions to the drive. As part of this process, the A2000 Target modifies the PRP entry as well as the LBA and NLB fields to account for this reduced length. In addition, if the format of the drive is using meta-data—see DIF Support *(page 21)*—the A2000 Target modifies the meta-data fields to account for the segmentation of the data stream.

4.6.1 Vendor-Specific I/O Commands

If vendor-specific commands are used, the A2000 Target's protocol engines require standard formatting of these commands, even though they are vendor defined. Specifically, hardware expects the opcodes to use the standard definition (the least significant two bits of opcode) to reflect whether there is data being transferred and in what direction the data is being transferred.

Additional limitations when using vendor-specific commands:

- Data transfers for vendor-specific commands are limited to 4KB.
- The NLB field for these commands must be set to zero.
- The MPTR field cannot be used as the A2000 Target replaces this field with zeros.

4.6.2 Firmware Snooping of I/O Commands

The A2000 Target has the capability to snoop opcodes in SQEs submitted to the chip and re-direct commands with specific (programmed) opcode values to an internal CPU. This CPU can modify various fields in the SQE, create one or more SQEs to be submitted to the drive(s), or even generate new commands to be sent back out on the network. This allows new or different support for new commands or existing commands which a user wishes to be handled in a possible non-standard way.

As an example, the A2000 Target supports NVMe Reservations—see NVMe Reservations *(page 184)*—at the bridge level, such that Reservations can be used even if the drives connected to the A2000 Target don't support this functionality. A filter exists which intercepts Reservation commands and passes them to the FW running in this CPU; it then performs the necessary hardware reconfiguration to enforce Reservations at the bridge level without involving the drive. This same capability could be used for other commands as well.

A second example would be to intercept NVMe Write commands, park them for later execution, and then submit them to the drives in a time-sliced manner such that reads are executed without the delays that writes cause, except at very specific time slots. This could be used as a method of creating more consistent read latencies.

4.7 NVMe Admin Command Processing

This section describes the processing of commands in the Admin Command Set. The purpose is to describe any operations which may be performed in addition to, or in place of, passing these commands through to the SSD.

This section references the following specifications:

- NVM Express base specification revision 1.4
- NVMe over Fabrics specification revision 1.1

Table 11: Admin Commands

OpCode	Optional/ Mandatory	Command
0x00	N/A	Delete I/O Submission Queue[5]
0x01	N/A	Create I/O Submission Queue [1 (page 24)]
0x02	M	Get Log Page (GLP) *(page 26)*
0x03	N/A	Reserved
0x04	N/A	Delete I/O Completion Queue [1 (page 24)]
0x05	N/A	Create I/O Completion Queue [1 (page 24)]
0x06	M	Identify *(page 28)*
0x07	N/A	Reserved
0x08	M	Abort
0x09	M	Set Features *(page 29)*
0x0A	M	Get Features *(page 31)*
0x0B	N/A	Reserved
0x0C	M	Asynchronous Event Request *(page 33)*
0x0D	O	Namespace Management (NS) *(page 33)*
0x0E	N/A	Reserved
0x0F	N/A	Reserved
0x10	O	Firmware Commit *(page 33)*
0x11	O	Firmware Image Download *(page 34)*
0x12	N/A	Reserved
0x13	N/A	Reserved
0x14	O	Device Self-Test
0x15	O	Namespace Attachment *(page 34)*
0x16	N/A	Reserved

5. NVMe-oF does not support the Admin commands associated with I/O Queue creation and deletion (Create I/O Completion Queue, Create I/O Submission Queue, Delete I/O Completion Queue, Delete I/O Submission Queue) defined in the NVMe Base specification. This is as specified in the NVM Express over Fabrics Revision 1.1.

OpCode	Optional/ Mandatory	Command
0x17	N/A	Reserved
0x18	M	Keep Alive (KA) *(page 35)*
0x19	O	NVMe-MI Send *(page 35)*
0x1A	O	NVMe-MI Receive *(page 35)*
0x1B	N/A	Reserved
0x1C	O	Virtualization Management
0x1D	O	NVMe-MI Send[6]
0x1E	O	NVMe-MI Receive
0x1F-0x7B	N/A	Reserved
0x7C	O	Doorbell Buffer Config
0x7D	N/A	Reserved
0x7E	N/A	Reserved
0x7F	M	Fabric Commands *(page 35)*
0x80	O	Format NVM *(page 36)*
0x81	O	Security Send
0x82	O	Security Receive
0x83	N/A	Reserved
0x84	O	Sanitize
0x85-0xBF	N/A	Reserved
0xC0-FF	O	Vendor Specific *(page 36)*

4.7.1 Admin Command Descriptions

The following sections describe each of the Admin commands. Each section includes:

- A description as to how the command is processed (via A2000 Target, SSD, or both). Specified as:
  - A2000 Target (processed solely by A2000 Target)
  - SSD (processed solely by SSD, e.g. passed-thru)
  - A2000 Target/SSD (processed by both A2000 Target and SSD)
- A table that describes the flow of the command from the host to/through the A2000 Target and the operations that are performed by the A2000 Target and the SSD.

For example, in the following table, the flow of the `Req` (request) is as follows (from left to right):

1. `<Sample Command>`: Admin Command is first sent from the host to A2000 Target via Ethernet
2. `<Sample Command> to SSD`: A2000 Target will just pass this command through to the SSD using NVMe via PCIe

The flow of the `Rsp`(response) is as follows (from right to left):

6. NVMe-MI Send/Receive can be passed through to an MI device on the PCIe bus or to an SSD. The mechanism to discover a specific MI device needs to be agreed upon/defined with Western Digital.

1. `NVMe Write to RapidFlex A2000`: SSD will do an NVMe Write to A2000 Target via PCIe (sending data needed to respond to the host). At this point, the data will be in A2000 Target's memory.
2. `Completion to RapidFlex A2000`: Completion will then be sent from the SSD to A2000 Target.
3. `RDMA Write to Host`: A2000 Target will perform an RDMA write over Ethernet to the Host (pushing data from A2000 Target memory).
4. `Completion to Host`: Completion will then be sent from A2000 Target to the Host.

Dir	Host	A2000 Target	SSD
Req	→ `<Sample Command>`	→ `<Sample Command>` to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 ← Completion to A2000

Note: From the SSD's perspective, the A2000 Target acts as the host. So, if any memory is needed for a given transaction (NVM Read/Write), A2000 Target memory will be used as the **Host** memory. In this example, when the SSD does the NVMe Write, the data will be staged in A2000 Target memory. When the A2000 Target then sends the RDMA Write to the Host, the data in A2000 Target memory will be moved to Host memory via the RDMA operation.

4.7.2 Get Log Page (GLP)

Discovery - (LID 0x70)

Dir	Host	A2000 Target	SSD
Req	→ GLP Discovery	→ Retrieve NVM Subsys Inventory	
Rsp		← RDMA Write to Host (prev cached info) ← Completion to Host	

The A2000 Target implements a Discovery Controller which supports retrieving NVM subsystems via Discovery Log Page (Log ID 0x70).

Note: There is also work underway for supporting referral to a separate Discovery Service (contact Western Digital directly for more information).

As defined in the NVMe over Fabrics specification, the Discovery Log Page provides an inventory of NVM subsystems with which a host may attempt to form an association.

A2000 Target will build up the discovery response based on the information it has cached for each drive. If Authorization is configured to restrict which targets a given host may access, the list returned will reflect the targets to which the given host has access.

A2000 Target supports Persistent connection Discovery controller. If the host sends a non-zero Keep Alive Timer value in the Fabric Admin Connect Request and enables Asynchronous Event Notifications from the Discovery controller, the host will receive an AEN completion when the Discover Log Page (Log ID 0x70) has been changed.

The following values are returned for the Discovery Log Header:

Log	Value
Generation Counter	0x00
Number of Records	<number of log page entries for which the given host has access>
Record Format	0x00

The following values are returned for each record:

Log	Value
Transport Type	0x01: RDMA
Address Family	0x01: IPv4 0x02: IPv6
Subsystem Type	0x02: Note: This can also report 0x01 for the A2000 Target firmware image which supports Referral to separate Discovery Service.
Transport Requirements	0x00: Not Specified
Port ID	0x00
Controller ID	0xFFFF: Dynamic Controller Model
Admin Max SQ Size	32
Transport Service ID	4420 in ASCII
NQN	<nqn from target NQN Table>
Transport Address	<IPv4 Address> <IPv6 Address>
Transport Specific Addr Subtype	031: RoceV1 041: RoceV2

Western Digital Core Dump (0xD1, 0xD2, 0xD3, 0xD5)

This provides the ability to pull an A2000 Target core dump via NVMe-CLI.

- 0xD1-D3=HW/FW core dump 1-3
- 0xD5=Pull FW log previously written to flash by setting RLMP Bridge Control bit 3

Dir	Host	A2000 Target	SSD
Req	→ GLP WDC Cmd	→ Retrieve Core Dump	
Rsp		← RDMA Write to Host ← Completion to Host	

Misc Get Log Page (LID 0x1-0x4, 0xC0-0xD0, 0xD6-0xFF)

Dir	Host	A2000 Target	SSD
Req	→ GLP Other	→ GLP Other to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 Target ← Completion to A2000 Target

Other Get Log Page

Dir	Host	A2000 Target	SSD
Req	→ GLP Other		
Rsp		← Completion to Host w/ Generic Invalid Field (0x02)	

4.7.3 Identify

Identify (CNS 0x00) - A2000 Target/SSD

The following table lists bit fields modified by the A2000 Target:

Bit Field	Meaning
NAWUN (Byte 35:34) NAWUPF (Byte 37:36) NACWU (Byte 39:38)	Set based on Block Size: Block Size 512: 0x7 Block Size 4096: 0x0
CTRATT (Byte 96)	Bit 6: Set

Dir	Host	A2000 Target	SSD
Req	→ ID Namespace	→ Retrieve ID NS Info	
Rsp		← RDMA Write to Host (previously cached information) ← Completion to Host	

Identify Controller Discovery - A2000 Target

Dir	Host	A2000 Target	SSD
Req	→ ID Cmd	→ ID Cmd to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 ← Completion to A2000

Identify Controller (CNS 0x1) - A2000 Target/SSD

The following table lists bit fields modified by the A2000 Target

Bit Field	Meaning
CMIC (Byte 76)	Bit 0: set if PCIe width=0x2 Bit1: Set Bits 2-3: Clear
CTRATT (Byte 96)	Bit 6: Set
OACS (Byte 257:256)	Clear bits 5, 7-9
ACL (Byte 258)	0x03
AERL (Byte 259)	0x03
LPA (Byte 261)	Clear Telemetry Bit (0x3)

Bit Field	Meaning
KAS (Byte 321:320)	0x0A
HCTMA (Byte 323:322)	0x00
SANICAP (Byte 331:328)	0x00
SQES (Byte 512)	0x66
CQES (Byte 513)	0x44
MAXCMD (Byte 515:514)	Max Bridge WQE Count
NN (Byte 519:516)	Supported NS based on Profile
AWUN (Byte 527:526) AWUPF (Byte 529:528) ACWU (Byte 533:532)	7 for 512 block size 0 for 4k block size
SGLS (Byte 539:536)	0x0010_0005
IOCCSZ (Byte 1795:1792)	0x0000_0023
IORCCSZ (Byte 1799:1796)	0x0000_0001
ICDOFF (Byte 1801:1800)	0x0000
FCATT (Byte 1802)	0x00
MSDBD (Byte 1803)	0x01

Dir	Host	A2000 Target	SSD
Req	→ ID Cmd	→ ID Cmd to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 ← Completion to A2000

Identify Other - SSD

Dir	Host	A2000 Target	SSD
Req	→ ID Cmd	→ ID Cmd to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 ← Completion to A2000 Target

Abort - A2000 Target

Dir	Host	A2000 Target	SSD
Req	→ Abort	→ Currently do not abort commands	
Rsp		← Completion to Host w/ Command Not aborted in DW10	

4.7.4 Set Features

Arbitration/Power Management/Error Recovery/Volatile Write Cache Memory/ Write Atomicity Normal – SSD

Dir	Host	A2000 Target	SSD
Req	→ SF Cmd	→ SF to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

LBA Range Type/Autonomous Power State Transition/Timestamp - SSD

Dir	Host	A2000 Target	SSD
Req	→ SF Cmd	← RDMA Read from Host → SF Cmd to SSD	← NVMe Read from A2000 Target
Rsp		← Completion to Host	← Completion to A2000 Target

Temp Threshold – A2000 Target/SSD

Dir	Host	A2000 Target	SSD
Req	→ SF Temp Thresh	→ SF Temp Thresh to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

Number of Queues

Dir	Host	A2000 Target	SSD
Req	→ SF #Queues	→ Calculate the # of queues to be reserved	
Rsp		← Completion to Host	

Async Event Config – A2000 Target/SSD

Dir	Host	A2000 Target	SSD
Req	→ SF Async	Store Event Config → SF Async to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

Keep Alive (KA)

Dir	Host	A2000 Target	SSD
Req	→ SF KA	→ Store KA value	
Rsp		← Completion to Host	

Host Identifier

Dir	Host	A2000 Target	SSD
Req	→ SF Host ID	→	
Rsp		← Completion to Host w/ Generic Command Sequence Error (0x0C)	

Interrupt Coalescing/Interrupt Vector Configuration/Host Memory Buffer

Dir	Host	A2000 Target	SSD
Req	→ SF Cmd	→	
Rsp		← Completion to Host w/ Generic Invalid Field (0x02)	

Other SF

Dir	Host	A2000 Target	SSD
Req	→ Other SF	→	
Rsp		← Completion to Host w/ Generic Invalid Field (0x02)	

4.7.5 Get Features

Arbitration/Power Management/Temp Threshold/Error Recovery/ Volatile Write Cache Memory/Write Atomicity Normal - SSD

Dir	Host	A2000 Target	SSD
Req	→ GF Cmd	→ GFCmd to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

LBA Range Type/Autonomous Power State Transition/Timestamp - SSD

Dir	Host	A2000 Target	SSD
Req	→ GF Cmd	→ GF Cmd to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 Target ← Completion to A2000 Target

Number of Queues

Dir	Host	A2000 Target	SSD
Req	→ GF# Queues	→ Retrieve # of Queues for Ctrl	
Rsp		← Completion to Host	

Async Event Config

Dir	Host	A2000 Target	SSD
Req	→ GF Async	→ Retrieve Async Event Cfg	
Rsp		← Completion to Host	

Keep Alive (KA)

Dir	Host	A2000 Target	SSD
Req	→ GF KA	→ Retrieve Keep Alive Timeout value	
Rsp		← Completion to Host w/ KA	

Host Identifier

Dir	Host	A2000 Target	SSD
Req	→ GF Host ID	→ Retrieve Host Identifier	
Rsp		← RDMA Write to Host ← Completion to Host	

Interrupt Coalescing/Interrupt Vector Configuration/Host Memory Buffer

Dir	Host	A2000 Target	SSD
Req	→ GF Cmd	→	
Rsp		← Completion to Host w/ Generic Invalid Field (0x02)	

Other GF

Dir	Host	A2000 Target	SSD
Req	→ Other GF	→	
Rsp		← Completion to Host w/ Generic Invalid Field (0x02)	

4.7.6 Reservation Notification Mask/Reservation Persistence

The A2000 Target can be configured to process NVMe 1.4 Reservation commands instead of passing them through to the NVMe drive. This is useful if the NVMe device does not support Reservations natively. See Authorization Config *(page 118)* for details on configuring Reservations). The following is a description of both modes of operation.

Pass-thru Mode - SSD

Dir	Host	A2000 Target	SSD
Req	→ GF Resv Cmd	→ GF Resv Cmd	→
Rsp		← Completion to Host	← Completion to A2000 Target

Bridge Processed Reservations

Dir	Host	A2000 Target	SSD
Req	→ GF Resv Cmd	→ GF Resv Cmd	
Rsp		← Completion to Host	

4.7.7 Asynchronous Event Request

Dir	Host	A2000 Target	SSD
Req	→ Async Event	→ Store ASYNC Event	
Rsp		← Completion to Host During Event Handling	

4.7.8 Namespace Management (NS)

Create - Namespace

Dir	Host	A2000 Target	SSD
Req	→ NS Mgmt Create	← RDMA Read from Host → NS Mgmt Create to SSD	← NVMe Read from bridge
Rsp		← Completion to Host	← Completion to A2000 Target

Delete - Namespace

Dir	Host	A2000 Target	SSD
Req	→ NS Mgmt Delete	→ NS Mgmt Delete to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

Other NS command

Dir	Host	A2000 Target	SSD
Req	→ Other NS Mgmt Cmd	→	
Rsp		← Completion to Host Invalid Cmd (0x01)	

4.7.9 Firmware Commit

Firmware Commit Slot 1-5 – SSD

Dir	Host	A2000 Target	SSD
Req	→ Firmware Commit	→ Firmware Commit to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

Firmware Commit Slot 6 or 7

Dir	Host	A2000 Target	SSD
Req	→ Firmware Commit	→ A2000 Target performs commit action for bridge firmware image	
Rsp		← Completion to Host	

4.7.10 Firmware Image Download

Firmware Image w/ A2000 Target Header

Dir	Host	A2000 Target	SSD
Req	→ Firmware Download	← RDMA Read from Host Store A2000 Target firmware image to scratch area	
Rsp		← Completion to Host	

Firmware Image w/o A2000 Target Header - SSD

Dir	Host	A2000 Target	SSD
Req	→ Firmware Download	← RDMA Read from Host → Firmware Download to SSD	← NVMe Read from A2000 Target
Rsp		← Completion to Host	← Completion to A2000 Target

4.7.11 Namespace Attachment

Attach/Detach – SSD

Dir	Host	A2000 Target	SSD
Req	→ NS Attach/Detach	← RDMA Read from Host → NS Attach/Detach to SSD	← NVMe Read from A2000 Target
Rsp		← Completion to Host	← Completion to A2000 Target

Other NS Attach

Dir	Host	A2000 Target	SSD
Req	→ Other NS Attach	→	
Rsp		← Completion to Host Invalid Cmd (0x01)	

4.7.12 Keep Alive (KA)

Dir	Host	A2000 Target	SSD
Req	→ KA	→ Reset KA Timer	
Rsp		← Completion to Host	

4.7.13 NVMe-MI Send

Table 56: NVMe-MI Send - MI Device or SSD

Dir	Host	A2000 Target	SSD
Req	→ MI Send	← RDMA Read from Host → MI Send to MI Device or SSD	← NVMe Read from bridge
Rsp		← Completion to Host	← Completion to A2000 Target

4.7.14 NVMe-MI Receive

NVMe-MI Receive - MI Device or SSD

Dir	Host	A2000 Target	SSD
Req	→ MI Receive	→ MI Receive to MI Device or SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 Target ← Completion to A2000 Target

4.7.15 Fabric Commands

Property Set/Get/Connect

Dir	Host	A2000 Target	SSD
Req	→ Fabric Cmd	→ Perform Operation	
Rsp		← Completion to Host	

Auth Send/Recv

Dir	Host	A2000 Target	SSD
Req	→ Auth Cmd	→	
Rsp		← Completion to Host w/ Generic Invalid Cmd (0x01)	

4.7.16 Format NVM

Table 60: Format NVM - SSD

Dir	Host	A2000 Target	SSD
Req	→ Format NVM	→ Format NVM to SSD	
Rsp		← Completion to Host	← Completion to A2000 Target

4.7.17 Vendor Specific

The A2000 Target will pass Opcodes 0xC0-0xFF to the target SSD. Some Opcodes may utilize a memory buffer to configure or return attributes, whereas others only utilize a DWord in the command or completion queue entry.

The low 2 bits in the Opcode will be used to determine if there is a memory buffer and the data transfer direction (if a buffer is to be transferred). These bits are defined as follows:

- 00: No Data to transfer
- 01: Data is transferred from Host to Controller
- 10: Data is transferred from Controller to Host
- 11: Data is transferred in both directions (not supported)

In addition to determining data direction, the A2000 Target will also need to know the size of the data to be transferred. It is assumed that the size is at a fixed location for all pass-thru commands. The location of the size is in DWord 8 (SGL Length) as the lower 14 bits (to account for a size of up to 8192 bytes). It is assumed that the size will not exceed 8192 bytes.

Table 61: No Data Transfer (low-bits: 00)

Dir	Host	A2000 Target	SSD
Req	→ Vendor Specific (No Data)	→ Vendor Specific to SSD	→
Rsp		← Completion to Host	← Completion to A2000 Target

Table 62: Data is Transferred from Host (low bits: 01)

Dir	Host	A2000 Target	SSD
Req	→ Vendor Specific (Write)	← RDMA Read from Host → Vendor Specific to SSD	← NVMe Read from A2000 Target
Rsp		← Completion to Host	← Completion to A2000 Target

Table 63: Data is Transferred from Controller (low bits: 10)

Dir	Host	A2000 Target	SSD
Req	→ Vendor Specific (Read)	→ Vendor Specific to SSD	→
Rsp		← RDMA Write to Host ← Completion to Host	← NVMe Write to A2000 Target ← Completion to A2000 Target

Configuration

The topics in this section provide instructions for configuring the A2000 Target.

In This Chapter:

5.1 Configuring the A2000 Target

This procedure provides instructions for configuring the A2000 Target for basic networking functions using UART CLI commands.

Note: For detailed information about the UART CLI commands and options used in this procedure, see Management Objects *(page 64)*.

Step 1: Use the `set-ip <Port> <DHCP> <IPv4> <Msk4> <GW4> <Host>` command to configure the network settings.

For example:

```
t-wdc> set-ip 1 0 192.168.1.52 255.255.255.0 192.168.1.1 c2000-1
```

Step 2: Use the `set-portset <Port> <Flow> <Prio> <LLDP> <DCBX> <MTU>` command to configure the Ethernet port settings.

For example:

```
t-wdc> set-portset 1 2 3 0 0 5000
```

Step 3: Use the `set-bridgeset <Mgmt> <DegC> <KATO> <RPID> <Dual> <DMI> <nTCP> <#DHP> <FWDL>` command to configure the network protocol settings.

For example:

```
t-wdc> set-bridgeset 1 85 1 0 0 0 1 3 0
```

Step 4: Use the `set-autoneg <Port> <FEC> <SPDS>` command to configure the auto-negotiation settings.

For example:

```
t-wdc> set-autoneg 1 2 8
```

Step 5: Use the `set-ccmode <MODE>` command to configure the congestion control mode.

For example:

```
t-wdc> set-ccmode 0
```

Step 6: Use the `set-profile <Prof> <#Hst>` command to configure the profile settings.

For example:

```
t-wdc> set-profile 2 0
```

Step 7: If required, use the `set-vlan <Port> <VLAN> <TRNK>` command to configure the VLAN settings.

Note: Use a VLAN value of 4095 to leave the interface untagged.

For example:

```
t-wdc> set-vlan 1 4095 0
```

Step 8 : Use the `show-drive <Slot>` command with a maximum slot value to verify the configuration.

For example:

```
t-wdc> show-drive 0xff
```

If the command returns correct information for all drives, discovery and connection commands from the initiator should work.

Result: The A2000 Target is now configured for basic operation.

Management

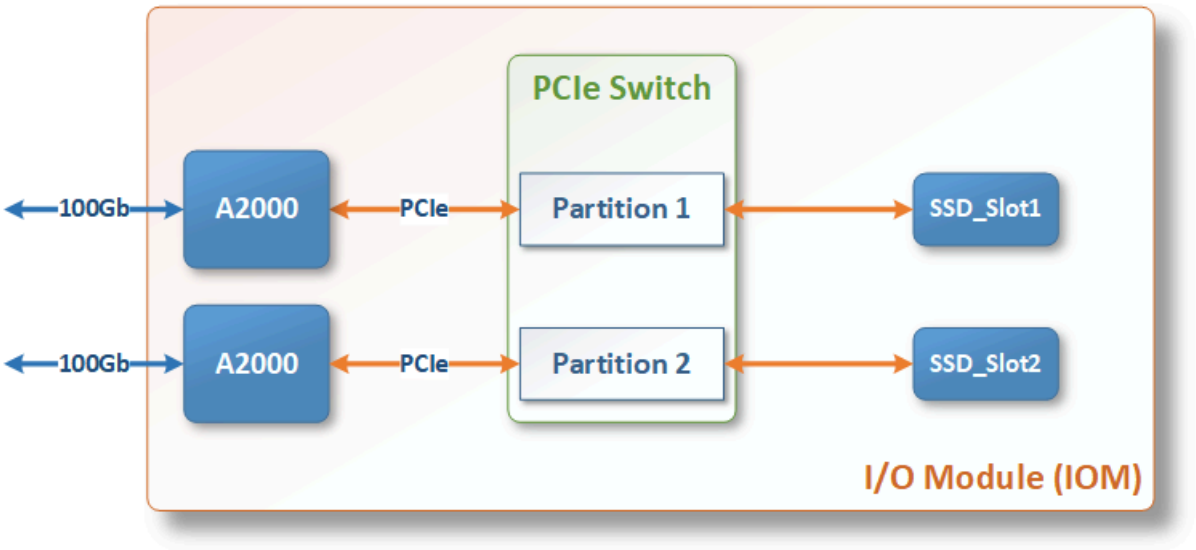
In This Chapter:

6.1 Access

There are four principal ways in which management of the A2000 Target device is connected. Each access method is detailed in the following sections:

- In-Band Access Through the Ethernet Interface *(page 42)*
- In / Out-of-Band Management Access via 1G Ethernet *(page 42)*
- Out-of-Band Access via the I2C Management Port *(page 44)*
- Out-of-Band Access via a PCIe Port *(page 45)*

6.1.1 In-Band Access Through the Ethernet Interface

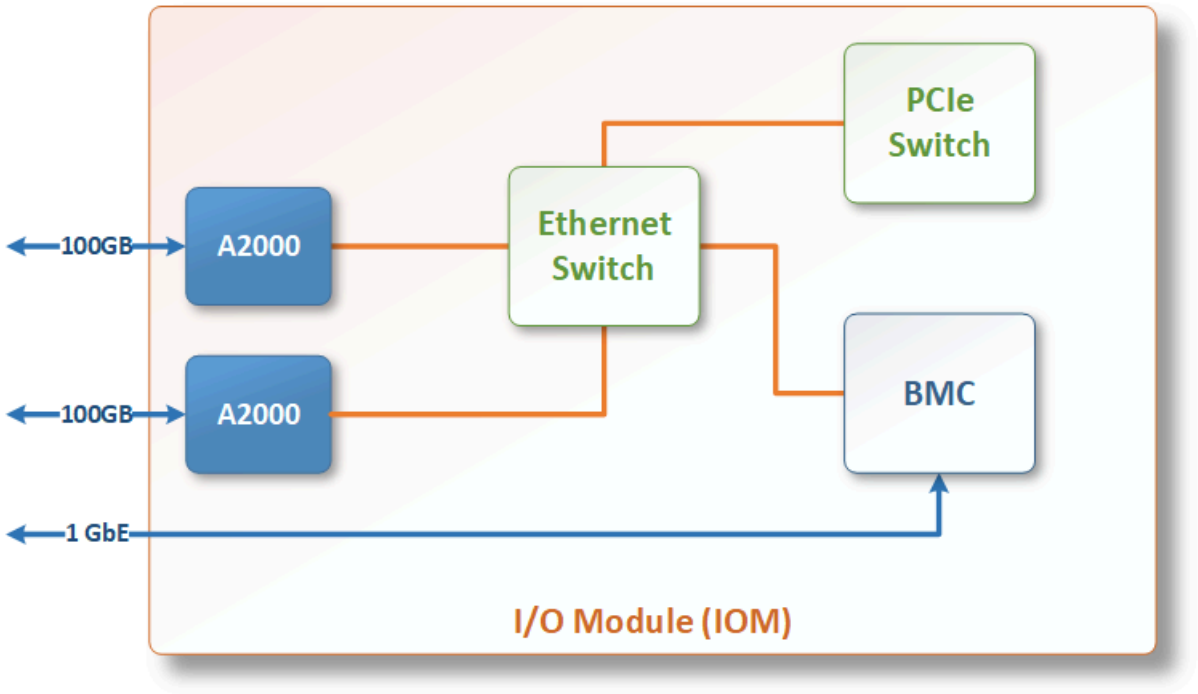


In the case of Ethernet in-band management connectivity, management traffic may be sent via the primary Ethernet connection (shown as 100Gb in this diagram) and is then directed to the A2000 Target's internal Management Processing Unit.

NVMe Pass-Through to Targets

NVMe-oF Host(s) can use a utility, such as NVMe-CLI, for issuing NVMe commands to the NVMe-oF Target (EBOF). In this case, A2000 Target will act as a proxy, passing the command to the target (Enclosure Manager, SSD, PCIe Switch, etc.) and forwarding the target's response back to the host.

6.1.2 In / Out-of-Band Management Access via 1G Ethernet



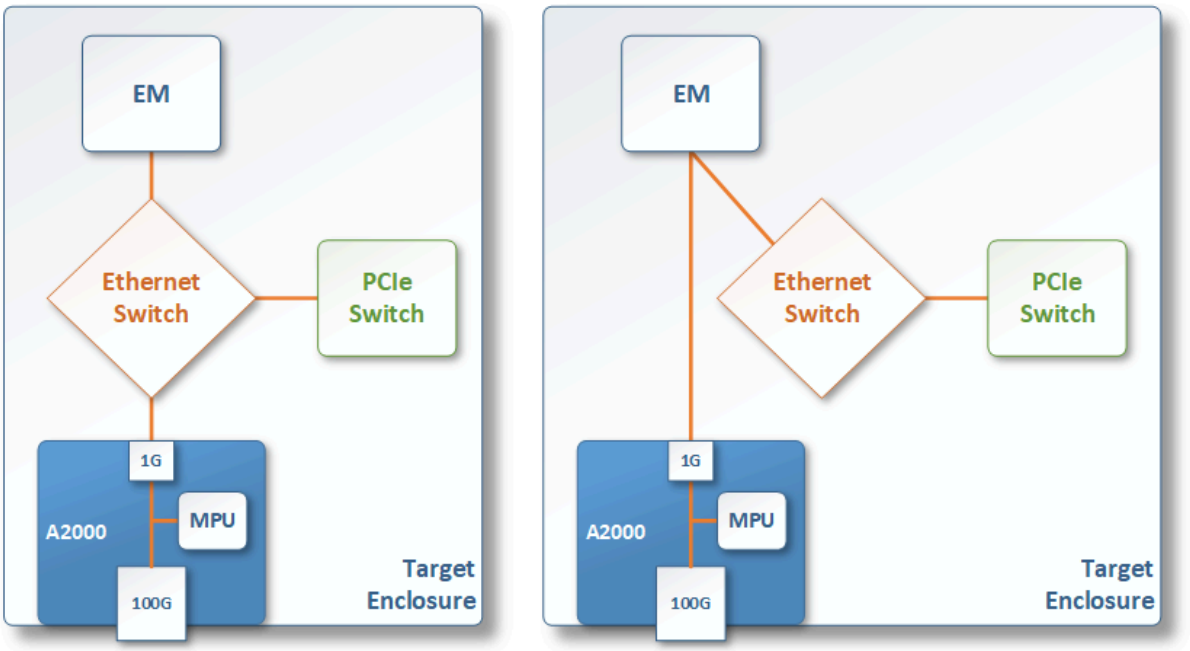
1GB Connectivity From BMC to Enclosure Ethernet Management Port (Independent of A2000 Target)

In this case, all management traffic to/from the BMC from an external management entity is accomplished via a dedicated and separate Ethernet port (shown above as a 1 GbE port).

1 GbE Connectivity From BMC to A2000 Target/100G Network

The A2000 Target contains a 1 Gb (SGMII) Ethernet port, which can be connected to a management entity such as the BMC (as in the previous diagram, connected to an Ethernet Switch to allow access via the BMC). This allows the BMC to talk RLMP directly to the A2000 Target via UDP. It also could allow the BMC to send/receive traffic to/from the 100G port connected to A2000 Target (as in the following diagram):

Figure 11: 1G Interface for A2000 Target

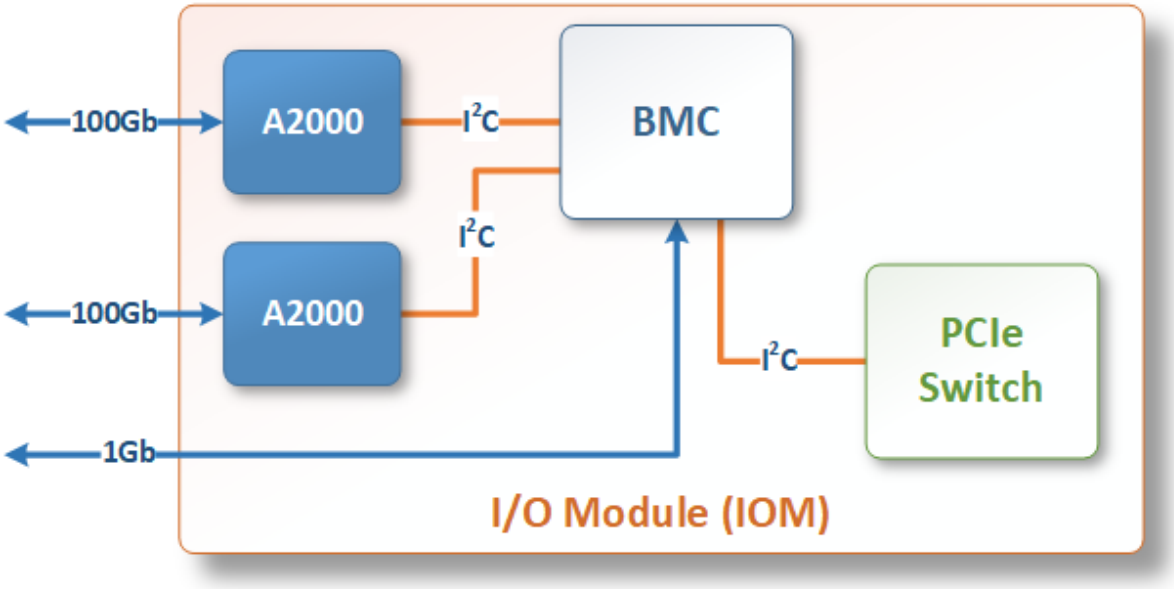


Treat 1G Interface like a standard Linux® Interface from the perspective of the EM/BMC (Unique MAC Address for 1G port which is different than A2000 Target MPU)

Packets Received from 100G Interface	Packets Received from 1G Interface
Fwd Bcast pkts to 1G and MPU	Fwd all Bcast pkts to MPU and 100G
Fwd Ucast pkts which match: • 1G MAC Addr to 1G only • MPU MAC Addr to MPU only	Fwd all Ucast pkts which match MPU MAC Addr to MPU only
Fwd Mcast pkts which match: • 1G Mcast Addr entries to 1G only • MPU Mcast Addr entries to MPU only	Fwd all Mcast pkts to 100G. All that match MPU Mcast MAC Addrs will also be sent to MPU
If Unknown DA, drop frame	If Unknown DA, Fwd to 100G

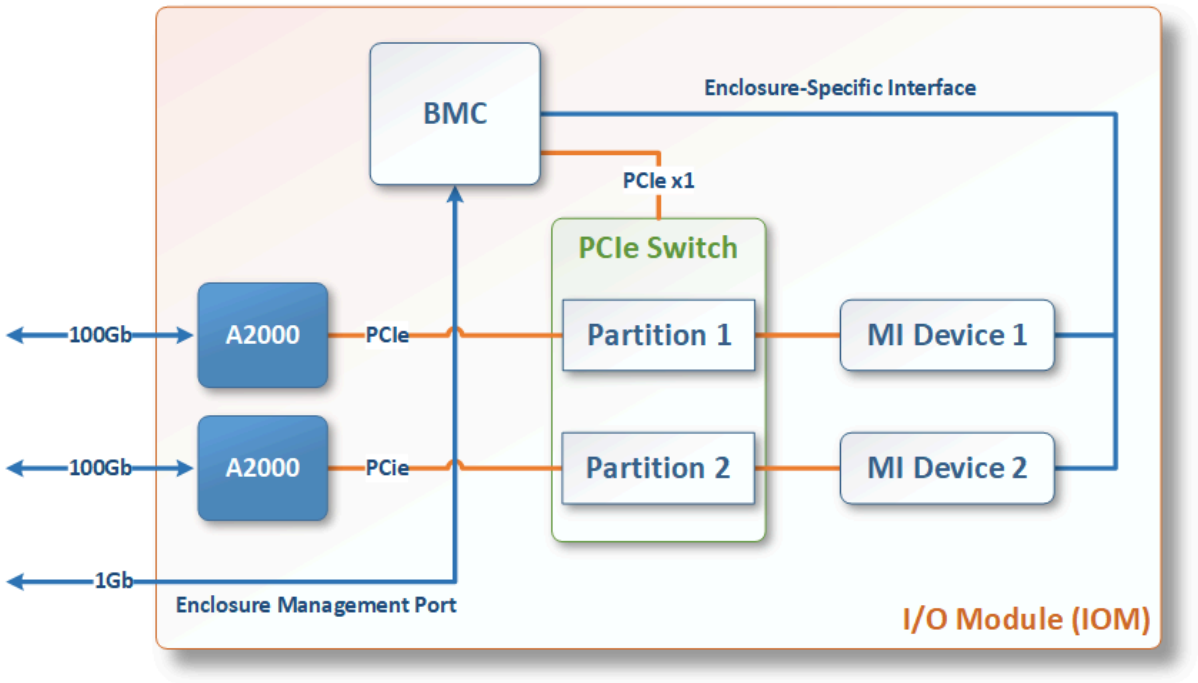
Note: there is an option to disable 1G to/from 100G traffic.

6.1.3 Out-of-Band Access via the I2C Management Port



In the case of Ethernet out-of-band management connectivity, all management traffic to/from the BMC from the external management entity is accomplished via a dedicated and separate Ethernet port. In this case, the BMC may access A2000 Target's internal management objects via I2C in order to read or write pertinent management information.

6.1.4 Out-of-Band Access via a PCIe Port



Similar to the previous case, all management traffic to/from the BMC from the external management entity is accomplished via a dedicated and separate Ethernet port. In this case, the BMC may access A2000 Target's internal management objects via PCIe in order to read or write pertinent management information.

6.2 Management Overview

Management is provided by firmware running on the A2000 Target's internal Management Processing Unit (MPU). Device Management provides four basic functions:

- Device configuration
- Device status
- Firmware image downloads
- Core dump uploads

Device Management can be accessed via any of the four methods previously described in Access *(page 42)*.

For the purposes of configuration and status reporting, the MPU creates a set of management objects. These objects can be accessed via any of the above-mentioned methods. For example, these objects can provide status of the device, the attached drives, or the connections and queue pairs used to communicate with the device and the drives.

The following sections describe the device management methods:

- UART Command Line Interface (CLI) *(page 47)*
- NVMe Admin Queues *(page 47)*
- BMC *(page 50)*

- Management Objects *(page 64)*

6.3 UART Command Line Interface (CLI)

The UART CLI, accessible through the A2000 Target's serial port, provides low-level access to the management functions of the A2000 chip.

6.4 NVMe Admin Queues

The NVMe Admin Queues are primarily used to manage the NVMe drives connected to the A2000 Target device, but can also be used to communicate with the device management controller to retrieve controller specific log pages and to download device firmware.

6.4.1 NVMe-CLI

NVMe-CLI is a publicly available NVMe management command line interface utility that provides standard-compliant management for NVMe drives. For more details, refer to https://github.com/linux-nvme/nvme-cli.

The following table lists the A2000 Target support status for NVMe-CLI commands (version 1.16 and later).

Note: RapidFlexCLI is the preferred utility for configuring the A2000 Target. The following NVMe-CLI commands are expected to work but have not been extensively tested. If they fail or do not produce the desired results, please use RapidFlexCLI.

Table 65: NVMe-CLI Command Support On A2000 Target

Command	A2000 Target Status
`nvme admin-passthru`	Supported
`nvme ana-log`	Unupported
`nvme attach-ns`	Supported
`nvme boot-part-log`	Unupported
`nvme capacity-mgmt`	Unupported
`nvme changed-ns-list-log`	Unupported
`nvme cmdset-ind-id-ns`	Unupported
`nvme compare`	Unupported
`nvme connect`	Supported
`nvme connect-all`	Supported
`nvme copy`	Unupported
`nvme create-ns`	Supported
`nvme delete-ns`	Supported
`nvme detach-ns`	Supported
`nvme device-self-test`	Supported
`nvme disconnect`	Supported

Command	A2000 Target Status
`nvme disconnect-all`	Supported
`nvme discover`	Supported
`nvme dsm`	Supported
`nvme effects-log`	Supported
`nvme endurance-event-agg-log`	Unupported
`nvme endurance-log`	Unupported
`nvme error-log`	Supported
`nvme flush`	Unupported
`nvme format`	Supported  **Note:** When using the `format` command to change the block size, the command may indicate that it has failed even when the change is successful. This issue occurs more often with versions older than 1.16.
`nvme fw-activate`	Supported  **Note:** This command was deprecated after NVMe-CLI version 1.2. See `nvme fw-commit` below.
`nvme fw-commit`	Supported
`nvme fw-download`	Supported
`nvme fw-log`	Supported
`nvme gen-dhchap-key`	Unupported
`nvme get-feature`	Supported
`nvme get-lba-status`	Unupported
`nvme get-log`	Supported
`nvme get-ns-id`	Supported
`nvme get-property`	Supported
`nvme help`	Supported
`nvme id-ctrl`	Supported
`nvme id-domain`	Unupported
`nvme id-iocs`	Unupported
`nvme id-ns`	Supported
`nvme id-nvmset`	Unupported
`nvme io-passthru`	

Command	A2000 Target Status
nvme lba-status-log	Unupported
nvme list	Supported
nvme list-ctrl	Supported
nvme list-endgrp	Unupported
nvme list-ns	Supported
nvme list-subsys	Supported
nvme ns-descs	Supported
nvme ns-rescan	Supported
nvme nvm-id-ctrl	Unupported
nvme persistent-event-log	Unupported
nvme predictable-lat-log	Unupported
nvme pred-lat-event-agg-log	Unupported
nvme primary-ctrl-caps	Unupported
nvme read	Supported
nvme reset	Supported
nvme resv-acquire	Supported
nvme resv-register	Supported
nvme resv-release	Supported
nvme resv-report	Supported
nvme rpmb	Unupported
nvme sanitize-log	Supported
nvme security-recv	Unupported
nvme security-send	Unupported
nvme self-test-log	Supported
nvme set-feature	Supported
nvme set-property	Supported
nvme show-hostnqn	Supported
nvme show-regs	Supported
nvme smart-log	Supported
nvme subsystem-reset	Unupported
nvme supported-log-pages	Unupported
nvme telemetry-log	Supported
nvme verify	Unupported
nvme write	Supported
nvme write-uncor	Unupported

Command	A2000 Target Status
`nvme write-zeroes`	Supported

6.5 BMC

A baseboard management controller (BMC) implemented within a storage enclosure can communicate with the A2000 Target device's MPU using a Western Digital-specific protocol called RapidFlex Lightweight Management Protocol (RLMP). This management encapsulation can be used for setting/getting management objects to/from A2000 Target's internal MPU via I2C, PCIe, or UDP/Ethernet 1G or 100G.

6.5.1 Endianness

The following sections describe the various headers used to pass commands between a management controller and an endpoint. Each of the protocols are defined by different governing bodies, and each has chosen one type of endianess in their documentation. For consistency, the following header data structures are shown in Little Endian format, where the least significant byte of a multi-byte field is shown in the lower addressed byte. It is assumed that Byte 0 is the least significant byte and will be sent first on the serial interface.

6.5.2 RLMP

A baseboard management controller (BMC) implemented within a storage enclosure can communicate with the A2000 Target's MPU using Western Digital's RapidFlex Lightweight Management Protocol (RLMP). This protocol is defined to provide a thin encapsulation format to minimize the number of bytes needed per transaction. It is used for all CRUD (Create/Read/Update/Delete) operations against any of the Management Objects. The details of the management objects are described in Management Objects *(page 64)*. Management commands can complete in 1 of 3 ways:

- **Complete successfully**: In this case, the Status (in the RLMP header) returned will be 0 indicating success, and the In-Progress bit (in the RLMP header) will be 0.
- **Complete with a failure**: In this case, the Status (in the RLMP header) returned will be non-zero indicating failure, and the In-Progress bit (in the RLMP header) will be 0.
- **Complete with an In-Progress state**: In this case, the Status (in the RLMP header) returned will be 0, and the In-Progress bit (in the RLMP header) will be set to 1.

Note: Only a single management command from the BMC to the MPU may be outstanding at any point.

6.5.3 In-Progress Command

Some commands may not be able to complete in the time needed to be able to respond to the I2C request from the BMC. If this is the case, the command will be completed, but it will return with the In-Progress bit set to 1 in the RLMP header (and Error Status of 0). If this occurs, the BMC will need to poll the In-Progress Status Command. When polling this command, the BMC needs to check the state of the In-Progress command. The state will indicate:

- Command is still in progress (in which case the BMC needs to continue polling)
- Command failed
- Command completed successfully

In addition to the state, the In-Progress command will also return the command code of the command which was last in progress.

Note: If a command is in-progress, the only MI command that will be accepted from the BMC is the In-Progress command.

The most recent in-progress command status will be returned when this command is executed, even if another (non In-Progress) MI command has been executed.

6.5.4 RLMP via I2C

RLMP can be sent between the BMC and the MPU via I2C, which is described in this section.. RLMP via PCIe, an alternative method of access, is described in RLMP via PCIe *(page 53)*.

I2C RLMP Packet Encapsulation Format – Get/Set

This is the RLMP encapsulation format used to send packets over an I2C interface from the BMC to A2000 Target's internal MPU. It is used for all CRUD (Create/Read/Update/ Delete) operations which the BMC would like to perform against any of the management objects to/from the MPU.

The BMC will initiate all transactions (BMC is the I2C Master) and the MPU will act as the Slave. A transaction can take one of 2 forms:

- **Single Start**: Perform a Write from the BMC, which begins with a Start condition and completes with a Stop condition. Then perform a Start condition and complete with a Read from the BMC.
- **Repeated Start**: Perform a Write from the BMC, which begins with a Start condition. Without sending a Stop, send another Start and Read.

Note: For completeness, the I2C Slave Address and Direction bit are shown below, but they are not counted in the DWords specified.

Note: DWords 0-N are sent from the BMC to the MPU. DWords (N+1)-M are returned from the MPU back to the BMC.

Table 66: I2C RLMP Packet Encapsulation Fields

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Slave Addr							0																								
0	Rsvd		CRUD						Cmd Code																Byte Count							
1 to N	Object Data																															
	Checksum																								Reserved							
	Slave Addr							1																								
N+1	R	P	Status						Cmd Code																Byte Count							
N+2 to M	Object Data																															
	Checksum																								Reserved							

Table 67: I2C RLMP Packet Encapsulation Field Descriptions

DWord	Bits	Description
	31:25	**Slave Addr**: Slave Addr 0x22 - I2C Slave Address of MPU
	24	**Direction**: 0: Write
0	31:30	**Reserved**
0	29:28	**CRUD**: Operation to perform: 0: Create Object 1: Read Object 2: Update Object 3: Delete Note: Each Object will define which CRUD operations are allowed for that object
0	27:16	**Cmd Code**: Unique Command Code for the specific Management Object
0	15:0	**Byte Count**: This field is a count of the bytes in the ObjectData
1-(N-1)	31:0	**Object Data**: Management Object Data – see Management Objects *(page 64)*
N	31:16	**Checksum**
N	15:0	**Reserved**
	31:25	**Slave Addr**: 0x22 - I2C Slave Address of MPU
	24	**Direction: 1**: Read
N+1	31	**Reserved**
N+1	30	**In-Progress (P)**: 0 Command complete. Check status field for success/failure. 1 Command in progress. Poll In-Progress command to determine when complete.
N+1	29:28	**Status**: Returns 0 for Success, else Error
N+1	27:16	**Cmd Code**: Echo the Unique Command Code for the specific Management Object
N+1	15:0	**Byte Count**: This field is a count of the bytes in the ObjectData
(N+2)(M-1)	31:0	**Object Data**: Management Object Data (See Management Object Formats)
M	31:16	**Checksum**
M	15:0	**Reserved**

I2C RLMP Sample Packet Exchange

What follows is an example of a Read operation of the Network Port MAC Address Management Object.

The following table breaks out the different fields for the Management Object which is sent from a BMC to the MPU (DWords 0-2) and the response from the MPU (DWords 3-6).

Table 68: I2C RLMP Sample Packet Exchange

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Rsvd		1						0x008								4															
1	Reserved																														Enet Port	
2	Checksum																Reserved															
3	Rsvd		0						0x008								8															
4	MAC Addr (Hi)																Reserved														Enet Port	
5	MAC Addr (Lo)																															
6	Checksum																Reserved															

The following diagrams show a sample of the actual data sent on the I2C for this transaction:

Figure 14: Command Sent from BMC:

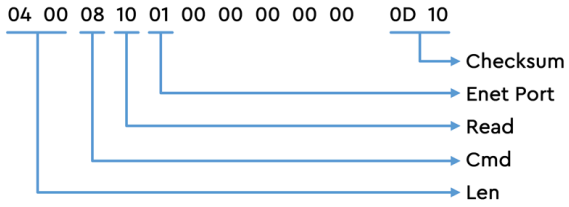
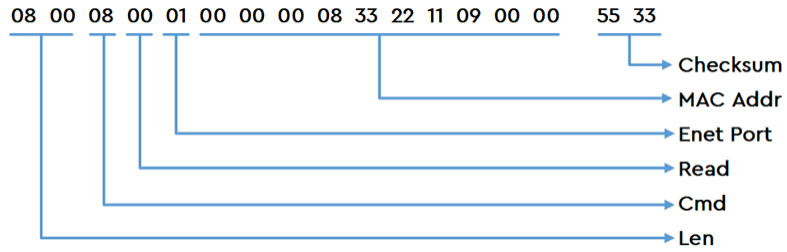


Figure 15: Response from MPU:



Note: The 16-bit checksum is computed as follows:

- Add the 16-bit values from the beginning of the packet (Command Length field) up through the reserved field just prior to the Checksum – this yields value `cksum`.
- Using the cksum value computed above:
  - cksum=(cksum>>16) + (cksum&0xFFFF)
  - cksum=(cksum>>16) + (cksum&0xFFFF)

Note: This operation must be performed twice as shown above.

6.5.5 RLMP via PCIe

RLMP can be sent between the BMC and the MPU via PCIe. This format is used to send packets over PCIe between a Western Digital PCIe port and a PCIe Management Endpoint (ME). RLMP over PCIe works in a similar manner to RLMP over I2C, as documented in RLMP via I2C *(page 51)*. Currently, the maximum payload size (MPS) is 32 bytes. All packets will be DWord aligned.

Note: Sending multiple unrelated Request messages outstanding at a time is not supported.

PCIe Message Format

The Vendor Defined Type 1 PCIe Message will be used to transport the RLMP packets. The message header and the fields are described in the following tables.

Table 69: PCIe Message Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Fmt			Type					R	TC			R	Attr	R	TH	TD	EP	Attr		AT		Length										
1	Requestor ID																Tag								Message Code - Vendor_Defined							
2	Bus								Device					Function			Vendor ID															
3	For Vendor Definition																															

Note: All fields are set to zero except the ones described in the following table.

Table 70: PCIe Message Format Descriptions

DWord	Bits	Description
0	31:29	**Fmt**: Set to 011b indicating 4 DWord header with data
0	28:24	**Type**: [4:3] set to 10b [2:0] set to 010b for route by ID
0	9:0	**Length**: Data in DWords
1	31:16	**Requester ID**: Bus/Device/Function number of the sender
1	7:0	**Message Code**: Set to 0111_1111b to indicate a Type 1 VDM
2	31:24	**Bus**: Bus Number of the target
2	23:19	**Device**: Device Number of the target
2	18:16	**Function**: Function Number of the target
2	15:0	**Vendor ID**: 0x70B3D576 (can be replaced with a partner vendor id)

DWord	Bits	Description
3	31:0	For Vendor Definition: [31] Set to 1b indicating RLMP protocol [30] More: Used when RLMP Object > MPL [29:16] Reserved [15:08] API Version – Current version is 0 [07:00] Sequence Number: Used when RLMP Object > MPL

PCIe RLMP Message Format

Table 71: PCIe RLMP Message Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	R	P	CRUD/ Status						Cmd Code								Byte Count															
1	Object Data																															
N																																

Note: DWord 0 is the RLMP Header. Description of the fields are as follows:

Table 72: PCIe RLMP Packet Encapsulation Field Descriptions

Field	Description
R	Reserved
P	**For a Request**: Reserved **For a Response**: In-Progress (P): 0 Command complete. Check status field for success/failure. 1 Command in progress. Poll In-Progress command to determine when complete.
CRUD/Status	**For a Request**: 0: Create Object 1: Read Object 2: Update Object 3: Delete Note: Each Object will define which CRUD operations are allowed for that object **For a Response**: Returns 0 for Success, else Error
Cmd Code	Unique ID specifying Management Object
Byte Count	This field is a count of the bytes in the Object Data
Object Data	For requests, parameters are passed using these fields. For example, the Ethernet port number for Network Status Object.

More Bit Usage

If RLMP Object is more than Maximum Payload Size (MPS) bytes, then the More bit should be used to indicate multiple messages, and Sequence Number should be used to detect out-of-order delivery. Table 5-9 summarizes the related field usage.

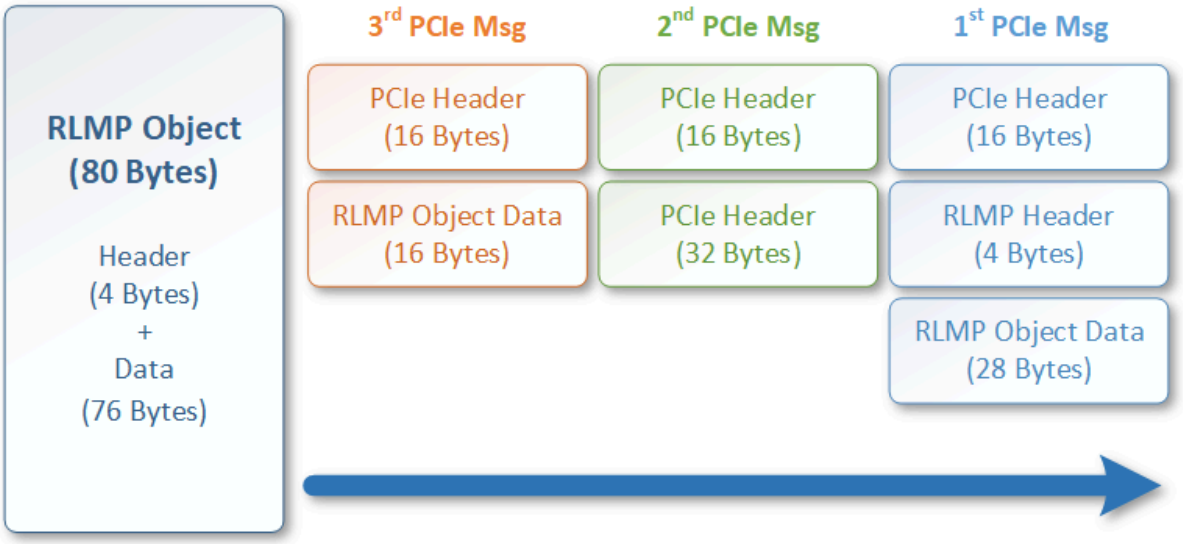
Table 73: More Bit Usage

PCIe Msg Payload Length (PMPL) (byte)	PCIe Hdr Length Field (PHL) (DWord)	More Bit (M)	Seq No (SN)	Has RLMP Hdr	RLMP Hdr Byte Count Field (KHBC)
<=MPS	PMPL/4	No	0	Yes	PMPL-4
>MPS First Msg	PMPL/4	Yes	1	Yes	PMPL-4
>MPS Middle Msg	PMPL/4	Yes	n	No	N/A
>MPS Last Msg	PMPL/4	No	N	No	N/A

For example, if the RLMP Object is 80 bytes, then the RLMP will send multiple PCIe messages:

- First message will have the PHL=8 (DWords), M=1, SN=1. It will have the RLMP Header and KHBC=76 (bytes).
- Second message will have the PHL=8 (DWords), M=1, SN=2. It will not have the RLMP Header.
- Third message will have the PHL=4 (DWords), M=0, SN=3. It will not have the RLMP Header.

Figure 16: RLMP Request/Response Example



This is applicable for both request and response messages since the request can be a Create/Update operation with data.

RLMP via PCIe Example

Table 74: Network Status Read Request for Ethernet Port 1

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0x002													4															
1	0																														1	

Table 75: RLMP via PCIe Example

Field	Description
R	0: Reserved
P	0: In-Progress
CRUD	1: Read Object
Object ID	2 - Network (Ethernet) Status
Byte Count	4
Object Data	1 - Ethernet Port #1

Table 76: Network Status Read Response for Ethernet Port 1

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0x002													152															
1	MTU																Rsvd		D	Speed					LinkState					EnetPort		
2	MAC Addr (Hi)																VLAN															
3	Mac Addr (Lo)																															
4-19	IPv4 Hostname																															
20	IPv4 Address																															
21	IPv4 Mask																															
22	IPv4 GW																															
23	IPv6 Addr (Hi)																															
24	IPv6 Addr (MidHi)																															
25	IPv6 Addr (MidLo)																															
26	IPv6 Addr (Lo)																															
27	SLAAC IPv6 Addr (Hi)																															
28	SLAAC IPv6 Addr (MidHi)																															
29	SLAAC IPv6 Addr (MidLo)																															
30	SLAAC IPv6 Addr (Lo)																															
31	Prefix IPv6 (Hi)																															
32	Prefix IPv6 (MidHi)																															
33	Prefix IPv6 (MidLo)																															
34	Prefix IPv6 (Lo)																															
35	IPv6 GW (Hi)																															
36	IPv6 GW (MidHi)																															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
37	IPv6 GW (MidLo)																															
38	IPv6 GW (Lo)																															

6.5.6 RLMP via UDP

RLMP can be sent between a management entity (such as a BMC) and the MPU via the User Datagram Protocol (UDP), which is described in this section. RLMP via UDP can be used via the 1G Mgmt port to the A2000 Target as well as in-band via the 100G data port.

UDP RLMP Packet Encapsulation Format - Get/Set

The following table shows the RLMP encapsulation format used to send packets over an Ethernet interface through A2000 Target's internal MPU. It is used for all CRUD (Create/Read/Update/Delete) operations from a BMC (or external management entity) against any of the management objects to/from the MPU.

The BMC will initiate all transactions and the MPU will act as the Slave.

Table 77: UDP RLMP Packet Encapsulation Format - Request

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Dst MAC Address Upper																															
1	Dst MAC Address Lower																															
2	Src MAC Address Upper																															
3	Ethertype																Src MAC Address Lower															
4	Total Length																TOS								IHL				Version			
5	Flags																Fragment ID															
6	Header Checksum																Protocol								TTL							
7	Src IP Address																															
8	Dst IP Address																															
9	Dst Port																Src Port															
10	Checksum																Length															
11	Rsvd		CRUD			Cmd Code											Byte Count															
12 to N	ObjectData																															
	Checksum																Reserved															

Table 78: UDP RLMP Packet Encapsulation Field Descriptions - Request

DWord	Bits	Description
0	31:16	**Dst MAC Address Upper**: Upper 16 bits of 48 bit globally unique Ethernet MAC address of packet destination
1	31:0	**Dst MAC Address Lower**: Lower 32 bits of 48 bit globally unique Ethernet MAC address of packet destination

DWord	Bits	Description
2	31:0	**Src MAC Address Upper**: Upper 32 bits of 48 bit globally unique Ethernet MAC address of packet source
3	31:16	**Ethertype**: Always IPv4 (0x0800)
3	15:0	**Src MAC Address Lower**: Lower 16 bits of 48 bit globally unique Ethernet MAC address of packet source
4	31:16	**Total Length**: Length of IP packet in bytes including IP Header
4	15:18	**TOS**: Type of Service. Diffserv/ECN. Always 0x00
4	7:4	**IHL**: Internet Header Length
4	3:0	**Version**: Protocol Version. Always 4
5	31:16	**Flags**: Fragment/Don't Fragment. Always 0x0000
5	15:0	**Fragment ID**: Sequence number for fragments of an IP packet. Only used if IP has fragmented the ULP packet, which is not supported with this ASIC.
6	31:16	**Header Checksum**: A one's complement of just the IP header.
6	15:8	**Protocol**: Always UDP (0x11)
6	7:0	**TTL**: Time to Live
7	31:0	**Src IP Address**: The 32 bit IP address of this node.
8	31:0	**Dst IP Address**: The32 bit IP address of the remote node.
9	31:16	**Dst Port**: Always 44257
9	15:0	**Src Port**: Always 44257
10	31:16	**Checksum**: The one's complement checksum of the UDP header and data payload
10	15:0	**Length**: Length of the UDP datagram in bytes, including the UDP header.
11	29:28	**CRUD**: Operation to Perform: 0: Create Object 1: Read Object 2: Update Object 3: Delete Note: Each Object will define which CRUD operations are allowed for that object.
11	27:16	**Cmd Code**: Unique Command Code for the specific Management Object
11	15:0	**Byte Count**: This field is a count of the bytes in the ObjectData
N-1:12	31:0	**ObjectData**: Management Object Data – see Management Objects *(page 64)*
N	31:16	**Checksum**
N	15:0	**Reserved**

Table 79: UDP RLMP Packet Encapsulation Format - Response

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Dst MAC Address Upper																															
1	Dst MAC Address Lower																															
2	Src MAC Address Upper																															
3	Ethertype																Src MAC Address Lower															
4	Total Length																TOS								IHL				Version			
5	Flags																Fragment ID															
6	Header Checksum																Protocol								TTL							
7	Src IP Address																															
8	Dst IP Address																															
9	Dst Port																Src Port															
10	Checksum																Length															
11	R		P	Status					Cmd Code								Byte Count															
12 to N	ObjectData																															
	Checksum																Reserved															

Table 80: UDP RLMP Packet Encapsulation Field Descriptions - Response

DWord	Bits	Description
0	31:16	**Dst MAC Address Upper**: Upper 16 bits of 48 bit globally unique Ethernet MAC address of packet destination
1	31:0	**Dst MAC Address Lower**: Lower 32 bits of 48 bit globally unique Ethernet MAC address of packet destination
2	31:0	**Src MAC Address Upper**: Upper 32 bits of 48 bit globally unique Ethernet MAC address of packet source
3	31:16	**Ethertype**: Always IPv4 (0x0800)
3	15:0	**Src MAC Address Lower**: Lower 16 bits of 48 bit globally unique Ethernet MAC address of packet source
4	31:16	**Total Length**: Length of IP packet in bytes including IP Header
4	15:18	**TOS**: Type of Service. Diffserv/ECN. Always 0x00
4	7:4	**IHL**: Internet Header Length
4	3:0	**Version**: Protocol Version. Always 4
5	31:16	**Flags**: Fragment/Don't Fragment. Always 0x0000
5	15:0	**Fragment ID**: Sequence number for fragments of an IP packet. Only used if IP has fragmented the ULP packet, which is not supported with this ASIC.
6	31:16	**Header Checksum**: A one's complement of just the IP header.
6	15:8	**Protocol**: Always UDP (0x11)
6	7:0	**TTL**: Time to Live

DWord	Bits	Description
7	31:0	**Src IP Address**: The 32 bit IP address of this node.
8	31:0	**Dst IP Address**: The32 bit IP address of the remote node.
9	31:16	**Dst Port**: Always 44257
9	15:0	**Src Port**: Always 44257
10	31:16	**Checksum**: The one's complement checksum of the UDP header and data payload
10	15:0	**Length**: Length of the UDP datagram in bytes, including the UDP header.
11	30	**In-Progress (P)**: 0 Command complete. Check status field for success/failure. 1 Command in progress. Poll In-Progress command to determine when complete
11	29:28	**Status**: Returns 0 for Success, else Error
11	27:16	**Cmd Code**: Unique Command Code for the specific Management Object
11	15:0	**Byte Count**: This field is a count of the bytes in the ObjectData
N-1:12	31:0	**ObjectData**: Management Object Data – see Management Objects *(page 64)*
N	31:16	**Checksum**
N	15:0	**Reserved**

UDP RLMP Sample Packet Exchange

The following tables are examples examples of a READ operation of the Host NQN Table Management Object. Table 31: UDP RLMP Packet Sample Packet Exchange - Request *(page 61)* breaks out the different fields for the UDP header and the Management Object which is sent from a BMC to the MPU. Table 32: UDP RLMP Packet Encapsulation Format - Response *(page 62)* breaks out the different fields for the UDP header and the Management Object which is the response from the MPU to the BMC.

Table 81: UDP RLMP Packet Sample Packet Exchange - Request

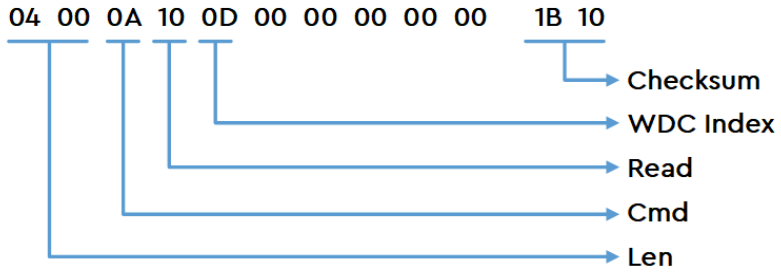
DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0x40B0 (Dst MAC Addr Upper)																															
1	0x341354C1 (Dst MAC Addr Lower)																															
2	0x70BED576 (Src MAC Addr Upper)																															
3	0x0800 (Ethertype)																0x8363 (Src MAC Addr Lower)															
4	0x0024 (Total Length)																0x0000 (TOS)								0x5 (IHL)				0x4 (Vers)			
5	0x0000 (Flags)																0x0133 (Fragment ID)															
6	Header Checksum																0x11 (Protocol)								0xFF (TTL)							
7	0xC0A80564 (Src IP Address)																															
8	0xC0A80565 (Dst IP Address)																															
9	0xACE1 (Dst Port)																0xACE1 (Src Port)															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
10	Checksum																0x0010 (Length)																
11	Rsvd		1 (Read)						0x00A (Command Code)								0x0004 (Byte Count)																
12	Reserved																								0x0D (WDC Index)							
13	0x1B10 (Checksum)																Reserved															

Table 82: UDP RLMP Packet Encapsulation Format - Response

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0x70BE (Dst MAC Addr Upper)																															
1	0xD5768363 (Dst MAC Addr Lower)																															
2	0x40b02413 (Src MAC Addr Upper)																															
3	0x0800 (Ethertype)																0x54C1 (Src MAC Addr Lower)															
4	0x0028 (Total Length)																0x0000 (TOS)								0x5 (IHL)				0x4 (Vers)			
5	0x0000 (Flags)																0x01fD (Fragment ID)															
6	Header Checksum																0x11 (Protocol)								0x80 (TTL)							
7	0xC0A80565 (Src IP Address)																															
8	0xC0A805654 (Dst IP Address)																															
9	0xACE1 (Dst P ort)																0xACE1 (Src Port)															
10	Checksum																0x0010 (Length)															
11	Rsvd		1 (Read)						0x00A (Command Code)								0x0108 (Byte Count)															
12	Reserved																0x03 (EMsk)								0x0D (WDC Index)							
13	Reserved																0x3FFF (Cmd Msk)															
14	0x3173742e (Upper NQN DWord)																															
	0x53535353																															
	0x53535353																															
	...																															
	0x53535353																															
77	0x53535353 (Lowest NQN DWord)																															
78	0xC555 (Checksum)																Reserved															

The following is a sample of the actual data sent on the wire for the RLMP payload for this transaction:



The 16-bit checksum is computed as follows:

- Add the 16 bit values from the beginning of the packet (Command Length field) up through the reserved field just prior to the Checksum – this yields value **cksum**.
- Using the cksum value computed above:
  - cksum=(cksum>>16) + (cksum&0xFFFF)
  - cksum=(cksum>>16) + (cksum&0xFFFF)

Note: This operation must be performed twice as shown above.

6.6 Management Objects

The following table provides an overview of the management objects provided by the RapidFlex A2000 Target, Fabric Bridge Device. Each object can be accessed by an RLMP command or CLI command and has defined operations it can perform: CREATE, READ, UPDATE, DELETE (CRUD). Additional details are provided for each management object in a subsequent section; click the link in the left column to visit that section.

Table 83: Management Objects

Object	Description				
	RLMP ID	INI/ TGT	C R U D	UART CLI Command(s)	
Bridge Status *(page 70)*	Current operating characteristics of the A2000 (temp, #conn, #core dumps, etc.)				
	0x000	I/T	R	`show-bridge <INST>`	
Bridge Version *(page 73)*	High-level config (max #network ports, max #drives, max #I/O conn, etc.)				
	0x001	I/T	R	`show-version <INST>`	
Network Status *(page 77)*	Network port status (link, speed, etc.)				
	0x002	I/T	R	`show-port <Port>`	
PCIe Root Port Config / Status *(page 82)*	PCI Express root ports status (link, speed, width, #drives, etc.)				
	0x003	I/T	R	`show-pcie <Port>`	
Connection Status *(page 84)*	Active network connection and I/O queue pair status (RDMA protocol, connection host name, connection drive(s), etc.)				
	0x004	I/T	R	`show-con <CID>`	
Drive Status *(page 86)*	Physical drives status (PCIe address, capabilities of drive, drive temperature, etc.)				
	0x005	T	R	`show-drive <Slot>`	
Controller Statistics *(page 90)*	Various network and I/O statistics (frame counts, I/O counts, I/O latency, etc.)				
	0x006	T	R	`show-stats <INDX>`	
Namespace Status *(page 93)*	Drive Namespace information (NS size, LBAF, etc.)				
	0x007	T	R	`show-ns <Slot> <NS>`	
Network Port MAC Address *(page 95)*	Ethernet port MAC address				
	0x008	I/T	R	`show-mac <Port>`	

Object	Description			
	RLMP ID	INI/ TGT	C R U D	UART CLI Command(s)
Network Port IPV4 Address *(page 97)*	Ethernet port IPv4 info (address, mask, GW, etc.)			
	0x009	I/T	R	`show-ip <Port>`
			U	`set-ip <Port> <DHCP> <IPv4> <Msk4> <GW4> <Host>`
Host NQN Table *(page 99)*	Map host NQN to Western Digital index. Western Digital index to be used for the authorization tables			
	0x00A	T	C	`add-hostnqn <EMsk> <CMsk> <HNQN>`
			R	`show-hostnqn <HIdx>`
			U	`set-hostnqn <HIdx> <EMsk> <CMsk> <HNQN>`
			D	`del-hostnqn <HIdx>`
Bridge Control *(page 102)*	Allows defaulting the configuration and resetting the A2000			
	0x00B	I/T	U	`set-control <CDFW> <CD1> <Dflt> <RST>`
Bridge Settings *(page 104)*	General A2000 settings (over-temp threshold, #DHCP retries, MR size, etc.)			
	0x00C	I/T	R	`show-bridgeset`
			U	`set-bridgeset <Mgmt> <DegC> <KATO> <RPID> <Dual> <DMI> <nTCP> <#DHP> <FWDL>`
Network Port VLAN / LAG *(page 106)*	Ethernet port VLAN address and LAG configuration			
	0x00D	I/T	R	`show-vlan <Port>`
			U	`set-vlan <Port> <VLAN> <TRNK>`
Network Port Settings *(page 108)*	Ethernet port parameters (MTU, flow control, etc.)			
	0x00E	I/T	R	`show-portset <Port>`
			U	`set-portset <Port> <Flow> <Prio> <LLDP> <DCBX> <MTU>`
Board Serial Number *(page 110)*	The configured serial number for this A2000			
	0x00F	I/T	R	`show-sn <INST>`
Target NQN Table *(page 111)*	Map target NQN to drive slot number			
	0x010	T	R	`show-targetnqn <Slot>`
			U	`set-targetnqn <Slot> <TNQN>`
Host / Target Auth Table *(page 113)*	Define which targets can be accessed by which hosts			
	0x011	T	C	`add-auth <HIdx> <Slot> <EIdx> <NS>`

Object	Description			
	RLMP ID	INI/ TGT	CRUD	UART CLI Command(s)
			R	`show-auth <WIdx> <EIdx>`
			U	`set-auth <WIdx> <HIdx> <Slot> <EIdx> <NS>`
			D	`del-auth <WIdx>`
Target Slot Map Table *(page 116)*	Map PCIe bus / device / function to target drive slot #			
	0x012	T	R	`show-slot <Slot>`
			U	`set-slot <Slot> <Port> <Bus> <Dev> <Func>`
Authorization Config *(page 118)*	Disable / enable host / target authorization			
	0x013	T	R	`show-authcfg`
			U	`set-authcfg <DHT> <Slot> <TGT> <Resv> <DNQN>`
Async Notification *(page 122)*	Allow A2000 to notify BMC that a condition occurred, which may require action			
	0x014	I/T	R	`show-async <Clr>`
			U	`set-async <Clr> <RMsk>`
Time Settings *(page 125)*	Configure Simple Network Time Protocol (SNTP) parameters			
	0x015	I/T	R	`show-time`
			U	`set-time <Time> <SNTP> <IPV4>`
Present/Shutdown Drives *(page 127)*	On READ: Bitmap, indicating if drive for the given slot# is currently present. On UPDATE: Allows setting bitmap to indicate on which drives to initiate a shutdown			
	0x016	T	R	`show-present <INST>`
			U	`set-present <INST> <Driv>`
Active Connections *(page 129)*	Bitmap, indicating if the given CID contains an active connection			
	0x017	I/T	R	`show-active <Type> <Grp>`
Format NVM *(page 131)*	Format a drive with standard NVMe parameters			
	0x018	T	U	`set-format <Slot> <NS#> <SES> <PIL> <PI> <MSET> <LBAF>`
Identify *(page 133)*	Issue Identify Namespace or Identify Controller command to a drive			
	0x019	T	R	`show-iden <Slot> <Parm>`

Object	Description			
	RLMP ID	INI/ TGT	C R U D	UART CLI Command(s)
In-Progress Status *(page 136)*	Query to get the status of a currently in-progress management command			
	0x01A	I/T	R	`show-cmdstatus <EX>`
FW Activate *(page 138)*	Verify/commit MPU firmware image			
	0x01B	I/T	U	`set-activate <Slot> <ACTN>`
FW Download *(page 139)*	Download firmware image to staging area			
	0x01C	I/T	U	`set-download <Len> <OFST> <IMGE>`
Network Port Auto-Negotiation *(page 140)*	Determine auto-negotiation parameters to advertise (FEC / pause / speed)			
	0x01D	I/T	R	`show-autoneg <Port>`
			U	`set-autoneg <Port> <FEC> <SPDS>`
MAC Interface Counters *(page 142)*	Transmit and receive MAC interface counters			
	0x01E	I/T	R	`show-macif <EnetPort>`
Read Core Dump *(page 146)*	Pull ASIC core dump or firmware debug messages from flash			
	0x01F	I/T	R	`show-core <Slot> <Ofst>`
Profile Selection *(page 148)*	Override EEPROM value to determine which connection/drive/ NS profile to use			
	0x020	I/T	R	`show-profile`
			U	`set-profile <Prof> <#Hst>`
VUC Opcode *(page 150)*	Allows user to set which Vendor Unique Command (VUC) Opcodes to use			
	0x022	T	R	`show-vucop <Command>`
			U	`set-vucop <Command> <Opcode>`
Locate LED *(page 152)*	Allows host to issue command to illuminate drive LED			
	0x023	T	R	`show-locateled`
			U	`set-locateled <LED Low> <LED Hi>`
Read VPD *(page 153)*	Allows the host to read the VPID information in EEPROM			
	0x024	I/T	R	`show-vpd <offset> <length>`
Uptime *(page 154)*	Number of seconds since the last boot/reset			
	0x025	I/T	R	`show-uptime`
Read I2C Xcvr *(page 155)*	Allows the host to read the Xcvr information via I2C			
	0x026	I/T	R	`show-xcvr <poff> <len>`

Object	Description			
	RLMP ID	INI/ TGT	C R U D	UART CLI Command(s)
MAC Etherstats Counters *(page 157)*	RX and TX MAC interface Etherstats counters			
	0x027	I/T	R	`show-ethstats <RXTX> <Port>`
DSCP *(page 160)*	Allows reading/updating the Differentiated Services Field Codepoints (DSCP) setting			
	0x02A	I/T	R	`show-dscp`
			U	`set-dscp <ENAB> <DSCP>`
Namespace Attachment *(page 162)*	Pass-thru Namespace attach/detach commands to a specific drive			
	0x02B	T	U	`set-nsattach <NS# > <Slot> <ATER>`
Namespace Management *(page 164)*	Pass-thru Namespace create/delete commands to a specific drive			
	0x02C	T	C	`add-nsmgmt <Slot> <LSZ > <USZ > <LCAP> <UCAP> <LBAS> <DPS > <NMIC> <AGID> <STID>`
			U	`set-nsmgmt <Slot>`
			D	`del-nsmgmt <NS#> <Slot>`
Debug Level *(page 167)*	Current ASIC and firmware logging levels			
	0x02D	I/T	R	`show-level <HW> <MID>`
			U	`set-level <MODE> <MID> <LVL> <BIT>`
Namespace Info Bitmap *(page 169)*	Bitmap of active / existing Namespaces for a given slot			
	0x02E	T	R	`show-nsinfo <Type> <Slot>`
NNS Status *(page 170)*	NNS target agent status			
	0x02F	T	R	`show-nns <Port>`
Identify Passthru *(page 172)*	Send `Identify` command to specific drive / Namespace / controller			
	0x030	T	R	`show-idpt <NSSl> <Opts>`
Perst Setting *(page 175)*	Override the `perstdelay` and `perstholdlo` settings in EEPROM			
	0x038	T	R	`show-perst`
			U	`set-perst <DLAY> <HOLD>`
SGMII *(page 176)*	Configure SGMII attributes			
	0x03C	T	R	`show-sgmii`
			U	`set-sgmii <EN> <PT>`

Object	Description			
	RLMP ID	INI/ TGT	C R U D	UART CLI Command(s)
Congestion Control Mode *(page 177)*	Determines the algorithm that A2000 runs for congestion control			
	0x03E	I/T	R	`show-ccmode`
			U	`set-ccmode <MODE>`
Data Center QCN Settings *(page 178)*	Values to optimize the DCQCN algorithm			
	0x03F	I/T	R	`show-dcqcn`
			U	`set-dcqcn <CnpT> <Time> <WC> <aI> <aT> <F> <g> <AddI>`
Logging to Remote Server *(page 181)*	Streaming of firmware logs to remote server			
	0x045	T	R	`show-logging`
			U	`set-logging <EN/DIS> <IPV4 ADDRESS>`

6.6.1 Bridge Status

The Bridge Status object provides information about the current operating characteristics of the A2000 Target, including internal temperature, how many connections are active, and how many core dumps exist in flash.

Supported CRUD Operations: READ

RLMP ID

- `0x000`

UART CLI Syntax & Example

- **READ** (Show Bridge Status MI): `show-bridge <INST>`

```
t-wdc> show-bridge 1
```

```
Bridge Status MI:
+ INST: 1
+ Stat: 1
+ CDmp: 0
+ CMap: 0
+ DSmd: 0
+ Drvs: 8
+ AdCn: 8
+ IOCn: 128
+ ioCM: 128
+ DegC: 55
+ MaxC: 56
+ POST: 0x00000000
+ Actn: 0x00000000
```

6.6.1.1 Bridge Status Object Format and Field Definitions

Table 84: Bridge Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																														0x1	
1	#Drives																DS	Reserved					CDMap		#Dumps				State			
2	#I/O Connections																#Admin Connections															
3	Chip Temperature																															
4	Max Chip Temperature																															
5	Power on Self-Test Results																															
	T	Rsvd						F	Reserved																							
6	Action Required (Blinking Amber LED)																															
	Reserved								DP	DA	DM	CF	SE	Reserved							CG	PL	FB	FC	PC	CD	R	P	GF	R		

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
7	Reserved																								ioCM							

Table 85: Bridge Status Field Definitions

DWord	Bits	Description
0	31:4	**Reserved**
0	3:0	**0x1** Must be set to 0x1 for all CRUD operations
1	31:16	**# Drives**: Number of drives attached to bridge
1	15	**DS** (PCIe Device Sharing Mode): 0 = Non-DS Mode 1 = DS Mode
1	14:12	**Reserved**
1	11:8	**CDMap** (Core Dump Bitmap): 1000b: Reserved 0100b: Slot 2 0010b: Slot 1 0001b: Slot 0 Note: 0x0 = invalid, 0x1 = valid
1	7:4	**# Dumps**: Number of core dumps resident in flash
1	3:0	**State**: Current state of bridge. 0x1=Active, 0x0=Halted
2	31:16	**# I/O Connections**: # of established I/O connections
2	15:0	**# Admin Connections**: # of established Admin connections
3	31:0	**Chip Temperature**: Internal die temperature in degrees Centigrade +/- 3 degrees
4	31:0	**Max Chip Temperature**: The maximum die temperature in degrees Centigrade since the last boot operation
5	31	**T**: Power-on self-test timeout 0 = Test complete 1 = Test failed due to timeout
5	30:28	**Reserved**
5	27	**F**: Power-on self-test failed
5	26:0	**Reserved**
6	31:0	**Action Required**: Bitmask to indicate why the status LED on the module is flashing amber
	31:20	**Reserved**
	19	**DP**: Device Sharing mode PSX Mem access error
	18	**DA**: Device Sharing mode AUTH config error
	17	**DM**: Device Sharing mode max drive error

DWord	Bits	Description
	16	**CF**: Fatal chip error
	15	**SE**: Serious error
	14:10	**Reserved**
	9	**CG**: Configuration does not match hardware
	8	**PL**: PCIe interface did not come up
	7	**FB**: Fallback image booted
	6	**FC**: FLASH configuration CRC error
	5	**PC**: PCIe link is up, but it is not configured as Gen3 x4, Gen3 x8, or Gen3 x16 on at lease one of the PCIe ports
	4	**CD**: Core Dump. New core dump occurred on the A2000 Target. Should read core dump from A2000 Target with NVMe CLI. Used for support
	3	**R**: Reboot Required. Firmware needs to be rebooted in order for a configuration change to complete
	2	**P**: POST Failure. Set power-on selftest results above.
	1	**GF**: Golden firmware image is loaded. May need to update the updateable firmware image and reboot
	0	**R**: Reserved
7	31:8	**Reserved**
7	7:0	**ioCM**: Represents the number of I/O queues committed to open Admin connections (as a result of NVMe “Set Features – number of queues” command response). IOCn is the current number of successfully established I/O queues. When connections are being opened, ioCM will be incremented before IOCn is incremented. When connections are stable, IOCn should equal ioCM.

6.6.2 Bridge Version

The Bridge Version object provides information about the high-level configuration of the A2000 Target, such as how many network ports exist, how many drives can be attached, and how many I/O connections are supported.

Supported CRUD Operations: READ

RLMP ID

- `0x001`

UART CLI Syntax & Example

- **READ** (Show Version - Bridge Config MI): `show-version <INST>`

```
t-wdc> show-version 1
```

```
Version - Bridge Config MI:
+ INST: 1
+ BdSN: USALP22290006
+ AcHW: ys_0000 (586-0xf9ffffd9)
+ GlFW: A2000Tfw_1.0.0
+ TUse: 0x00000000
+ LRst: 0x0000000A
+ AcFW: A2000Tfw_2.0.0.rc8 (Aug  9 2023-13:02:48) Slot:7
+ SLT6: A2000Tfw_2.0.0.rc8
+ SLT7: A2000Tfw_2.0.0.rc8*
+ Mgmt: 1
+ Init: 0
+ ExUa: 0
+ Pres: 0
+ PrLo: 0
+ Prst: 0
+ Prfl: 1
+ Port: 3
+ PCIe: 1
+ SSDs: 32
+ IOQs: 128
+ Cons: 192
+ VCtl: 65
+ Dual: 0
```

6.6.2.1 Bridge Version Object Format and Field Definitions

Table 86: Bridge Version Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																														0x1	
1	Reserved																															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	Reserved																															
3	Reserved																															
4-19	HW Version																															
20-35	Golden Firmware Image Version																															
36	Reserved																								TUse							
37	Reserved																LRst															
38-51	Reserved																															
52-67	Active Firmware Image Version																															
68-83	Downloadable Firmware Image Version (Slot 6)																															
84-99	Downloadable Firmware Image Version (Slot 7)																															
100	M	I	EU	R	Pres				R	PrLo		Prst					Profile								DP	Reserved						
101	#I/O Queues																#SSDs								#PCIePorts				#ENetPorts			
102	# Virtual Controllers																# Transport Conns															
103-107	Board S/N																															

Table 87: Bridge Version Field Definitions

DWord	Bits	Description
0	31:4	**Reserved**
0	3:0	**Must be set to 0x1** for all CRUD operations
1	31:0	**Reserved**
2	31:0	**Reserved**
3	31:0	**Reserved**
4-19	31:0	**Hardware Version**: ASIC hardware version
20-35	31:0	**Golden Firmware Image Version**: Golden (non-updateable) FW version in Flash
36	31:11	**Reserved**
36	10:0	**TUse**: Same as k_preset_to_use in EEPROM
37	31:16	**Reserved**
37	15:0	**LRst**: Reason for previous Reset of A2000 Target
38-51	31:0	**Reserved**
52-67	31:0	**Active Firmware Image Version**: Version of the firmware image currently running
68-83	31:0	**Downloadable Firmware Image Version**: User updateable (slot 6) Version of the firmware image in flash
84-99	31:0	**Downloadable Firmware Image Version**: User updateable (slot 7) Version of the firmware image in flash

DWord	Bits	Description
100	31	Returns current state of proprietary Western Digital Management Application access 0: Listening on TCP Port 9999 is disabled (Default) 1: Listening on TCP Port 9999 is enabled. See also Bridge Settings *(page 104)* management object
100	30	**Initiator**: This bit is set if A2000 Target is running as Initiator, else it is 0
100	29	**ExUa**: Same as UART Direction in EEPROM
100	28	**Reserved**
100	27:24	**Pres**: Same as k_qeupreset in EEPROM
100	23	**Reserved**
100	22:20	**PrLo**: Same as Ho.d PRST in EEPROM
100	19:16	**Prst**: Same as PRST Delay in EEPROM
100	15:8	**Profile**: Same as Profile in EEPROM
100	7	**DP** (DualPort2): If set, indicates this A2000 Target connects to the 2nd port of dual-ported drives. Used in HA Multi-Path Chassis configurations. If set, will cause the Controller IDs for this A2000 Target to have the upper bit of the 16-bit Controller ID to be set. This is used to make sure the Controller IDs for the 2 ports on a Dual ported drive end up with a unique controller ID. This indicates the active value during this boot cycle.
100	6:0	**Reserved**
101	31:16	**Max # I/O Queues**: Maximum number of I/O queues supported. This number is selected based on the Profile Selected (See Profile Selected object).
101	15:8	**Max # SSDs**: Maximum number of SSDs that can be connected to this bridge. If an MI device is discovered, it is included in the count of SSDs. **Note**: these drives may be connected in any combination across the available PCIe root ports and across the available Ethernet ports. This number is selected based on the profile selected - see Profile Selection *(page 148)*.
101	7:4	**#PCIePorts**: Number of PCIe root ports implemented in chip
101	3:0	**#EnetPorts**: Number of Ethernet Ports implemented in chip. Note that when the chip is running at its maximum port speed, only one MAC is used. At lower speeds, one or two MACs may be used.
102	31:16	**Max # Virtual Controllers**: Maximum number of Virtual Controllers. Since Virtual Controllers are mapped directly to Admin Queues, this also defines the maximum number of Admin Queues supported.
102	15:0	**Max # Transport Connections**: Maximum number of TCP / RoCE connections supported

DWord	Bits	Description
103-107	31:0	**Board S/N**: Unique Serial Number for the board as a string

6.6.3 Network (Ethernet) Status

The Network Status object provides information about the Ethernet port(s) of the A2000 Target.

Supported CRUD Operations: READ

RLMP ID

- `0x002`

UART CLI Syntax & Example

- **READ** (Show Port Status (Ethernet) MI): `show-port <Port>`

```
t-wdc> show-port 1
```

```
Port Status (Ethernet) MI:
+ Port: 1
+ Link: 1
+ Xcvr: 2
+ Sped: 100Gbps
+ DHCP: 0
+ FEC : RS FEC
+ MTU : 5000
+ MAC : 00:0c:ca:12:21:78
+ VLAN: untagged
+ HOST: p1
+ IPv4: 10.10.10.233
+ Msk4: 255.255.255.0
+ GW4 : 10.10.10.1
+ IPv6: fe80:0000:0000:0000:020c:caff:fe12:2178
+ IPv6: 0000:0000:0000:0000:0000:0000:0000:0000
+ PRF6: 0000:0000:0000:0000:0000:0000:0000:0000
+ GW6 : 0000:0000:0000:0000:0000:0000:0000:0000
```

Note: This example includes the encapsulation used with the RLMP over I2C.

6.6.3.1 Network Status Object Format and Field Definitions

Table 88: Network Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	MTU																FEC		D		Speed				Xcvr		Link		EnetPort			
1	Mac Addr High																VLAN															
2	Mac Addr Low																															
3-18	IPv4 HostName																															
19	IPv4 Address																															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
20	IPv4 Mask																															
21	IPv4 GW																															
22	IPv6 Addr (Lo)																															
23	IPv6 Addr (MidLo)																															
24	IPv6 Addr (MidHi)																															
25	IPv6 Addr (Hi)																															
26	SLAAC IPv6 Addr (Lo)																															
27	SLAAC IPv6 Addr (MidLo)																															
28	SLAAC IPv6 Addr (MidHi)																															
29	SLAAC IPv6 Addr (Hi)																															
30	Prefix IPv6 Addr (Lo)																															
31	Prefix IPv6 Addr (MidLo)																															
32	Prefix IPv6 Addr (MidHi)																															
33	Prefix IPv6 Addr (Hi)																															
34	IPv6 GW (Lo)																															
35	IPv6 GW (MidLo)																															
36	IPv6 GW (MidHi)																															
37	IPv6 GW (Hi)																															

Table 89: Network Status Format - RLMP Read

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Rsvd		1						0x002								4															
1	Reserved																														EnetPort	
2	Checksum																Reserved															
3	R	P	0						0x002								152															
4	MTU																FEC	D	Speed						Xcvr				Link		EnetPort	
5	Mac Addr High																VLAN															
6	Mac Addr Low																															
7-22	IPv4 HostName																															
23	IPv4 Address																															
24	IPv4 Mask																															
25	IPv4 GW																															
26	IPv6 Addr (Lo)																															
27	IPv6 Addr (MidLo)																															
28	IPv6 Addr (MidHi)																															
29	IPv6 Addr (Hi)																															
30	SLAAC IPv6 Addr (Lo)																															
31	SLAAC IPv6 Addr (MidLo)																															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
32	SLAAC IPv6 Addr (MidHi)																															
33	SLAAC IPv6 Addr (Hi)																															
34	Prefix IPv6 Addr (Lo)																															
35	Prefix IPv6 Addr (MidLo)																															
36	Prefix IPv6 Addr (MidHi)																															
37	Prefix IPv6 Addr (Hi)																															
38	IPv6 GW (Lo)																															
39	IPv6 GW (MidLo)																															
40	IPv6 GW (MidHi)																															
41	IPv6 GW (Hi)																															
42	Checksum																Reserved															

Table 90: Network Status Field Definitions from BMC

DWord	Bits	Description
0	31:30	**Reserved**
0	29:28	**CRUD**: 1 = Read
0	27:16	**Cmd Code**: 0x002
0	15:0	**Byte Count**: 4
1	31:4	**Reserved**
1	3:0	**EnetPort**: Network port for which this status is given 1 for the 1st port 2 for the 2nd port 3 for the 1G SGMII port **Note**: This field must be set for all CRUD operations.
2	31:16	**Checksum**
2	15:0	**Reserved**
3	31	**Reserved**
3	30	**P**: 0 = Not in progress
3	29:28	**Status**: Returns 0 for success, else Error
3	27:16	**Cmd Code**: 0x002
3	15:0	**Byte Count**: 152
4	31:16	**MTU**: Currently active Ethernet Maximum Transmit Unit
4	15:13	**FEC**: **Bit** **CLI Value** **FEC** 0 1 FC-FEC (default) 1 2 RS-FEC 2 4 10Gb/s per lane FEC (Reserved: Not supported)
4	12	**D**: DHCP State

DWord	Bits	Description
4	11:8	Speed: **Value** **Definition** 0x0 Reserved 0x1 25Gbps 0x2 50Gbps 0x3 100Gbps 0x4-0xc Reserved 0xd 10Mbps 0xe 100Mbps 0xf Reserved
4	7:6	**Xcvr**: Current Xcvr Type **Value** **Definition** 0x0 No Xcvr Present 0x1 Copper QSFP 0x2 Optical QSFP 0x3 Reserved
4	5:4	**Link** (current state): **Value** **Definition** 0x0 Link Down 0x1 Link Up 0xF Unknown
4	3:0	**Port**: Ethernet Port
5	31:16	**MAC Address High**: Currently active upper 16 bits of 6 byte MAC address
5	15:0	**VLAN**: Currently active VLAN this port is assigned to **Value** **Definition** 0 Priority Tag used, No-VLAN 1-4094 Tagged VLAN >4094 Untagged VLAN. If set above 4094, VLAN tagging will not be used.
6	31:0	**MAC Address Low**: Currently active lower 32-bits of 6 byte MAC address
7-22	31:0	**IPv4 Hostname**: Currently active IPv4 Hostname
23	31:0	**IPv4 Address**: Currently active 4 byte IPv4 address for this port
24	31:0	**IPv4 Mask**: Currently active 4 byte IPv4 Gateway for this port
25	31:0	**IPv4 GW**: Currently active 4 byte binary IPv4 Gateway for this port
26-29	31:0	**IPv6 Address**: Currently active 16 byte IPv6 address for this port
30-33	31:0	**SLAAC IPv6 Address**: Stateless Autoconfigured 16 byte IPv6 addr for this port
34-37	31:0	**Prefix IPv6**: Rout-able Auto-configured 16 byte IPv6 prefix/subnet for this port
38-41	31:0	**IPv6 GW**: Currently active 16 byte IPv6 Gateway for this port
42	31:16	**Checksum**

DWord	Bits	Description
42	15:0	**Reserved**

6.6.4 PCIe Root Port Config / Status

The PCIe Root Port Config/Status object provides information about the PCI Express Root Port of the A2000 Target, such as whether the link is up, at what speed/width, and the number of drives detected.

Supported CRUD Operations: READ

RLMP ID

- `0x003`

UART CLI Syntax & Example

- **READ** (Show PCIe Root Status MI): `show-pcie <Port>`

```
t-wdc> show-pcie 0
```

```
PCIe Root Status MI:
+ Port: 0
+ Link: 1
+ Gen : 4
+ Wdth: x16
+ Drvs: 0
+ VID : 0x00001B96
```

6.6.4.1 PCIe Root Port Config / Status Object Format and Field Definitions

Table 91: PCIe Root Port Config / Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved								#Drives								Width				Speed				LinkState				PCIe Port			
1	Device/ Vendor ID																															

Table 92: PCIe Root Port Config / Status Field Definitions

DWord	Bits	Description	
0	31:24	**Reserved**	
0	23:16	**# Drives**: Number of drives detected on this root port	
0	15:12	**Width**:	
		Value	**Definition**
		0x0	Link down
		0x1	Port is x1
		0x2	Port is x2
		0x3	Port is x4

DWord	Bits	Description
		0x4 Port is x8
		0x5 Port is x16
0	11:8	**Speed**: **Value** **Definition** 0x0 Link down 0x1 Gen1 0x2 Gen2 0x3 Gen3 0x4 Gen4 0x5 - 0xf Reserved
0	7:4	**LinkState**: Current state of PCIe Port **Value** **Definition** 0x0 Link down 0x1 Link up
0	3:0	**PCI Port**: PCIe Port for which this status is given 0 for the 1st port **Note**: This field must be set for all CRUD operations.
1	31:16	**Device / Vendor ID**: Of switch or drive connected to root port

6.6.5 Connection Status

The Connection Status object provides information about the active network connections and I/O queue pairs. This includes things like the RDMA protocol of the connection, the host name using the connection, and what drive(s) the connection is accessing. Command is only valid for CIDs with an active connection. Refer to the Active Connections *(page 129)* command for a list of currently active connections.

Supported CRUD Operations: READ

RLMP ID

- `0x004`

UART CLI Syntax & Example

- **READ** (Show Con (Connection) Status MI): `show-con <CID>`

```
t-wdc> show-con 1
```

```
Con (Connection) Status MI:
+ CID : 1
+ Slot: 0
+ EPrt: 1
+ Host: 0x00000000
+ TTYP: 1
+ NTYP: 1
+ QPID: 0x00000000
+ QSiz: 0
+ QTyp: 0
+ Stat: 1
+ CtID: 1
+ MSS : 4096
+ HIP : 10.10.10.231
+ HID1: 0x00000000
+ HID2: 0x00000000
+ HID3: 0x00000000
+ HID4: 0x00000000
+ HNQN:
```

6.6.5.1 Connection Status Object Format and Field Definitions

Table 93: Connection Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Rsvd				EnetPort				Slot#								CID															
1	Reserved								NType								TType								Host Transport Port							
2	State				QType				Qsize								Drive QPID															
3	MSS																Controller ID															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
4-7	Host IP Address																															
8-11	Host ID																															
12-75	Host NQN																															

Table 94: Connection Status Field Definitions

DWord	Bits	Description
0	31:28	**Reserved**
0	27:24	**EnetPort**: Network port for which this status is given 1 for the 1st port 2 for the 2nd port
0	23:16	**Slot#**: Chassis / Shelf Slot Number for the drive for this Connection Entry
0	15:0	**CID**: Internal ID for Transport Connection **Min Conn** **Max Conn** 1 # Transport Conns. See Bridge Version *(page 73)* MI Command for number Transport Con **Note**: This field must be set for all CRUD operations. This field is used as an index to retrieve this object.
1	31:24	**Reserved**
1	23:20	**NType** (Network/IP Type): 0 = GRH (RoCEv1) | 1 = IPv4 | 2 = IPv6
1	19:16	**TType**: 0 = RoCEv1 | 1 = RoCEv2/UDP | 2 = TCP
1	15:0	**Host Transport Port#**: TCP Port Number for NVMe over TCP, UDP Port Number for RoCEv2
2	31:28	**State**: 0 = Closed | 1 = Connected/Established | 2 = Disconnecting
2	27:24	**QType**: 0 = Admin QP | 1 = I/O QP
2	23:16	**QSize**: number of entries in NVMe SQ/CQ
2	15:0	**Drive QPID**: The NVMe queue pair ID associated with the connected drive
3	31:16	**MSS**: MSS used for this connection. **RoCE**: Path Maximum Transfer Unit (PMTU). PMTU defines the payload size and can be 256, 512, 1024, 2048, or 4096 bytes.
3	15:0	**Controller ID**: Internal Index used into Controller Table for this connection
4-7	31:0	**Host IP Address**: 16 byte binary IP address. If IPv4, upper 12 bytes will be zero
8-11	31:0	**Host ID**: HOSTID from Fabric Connect.
12-75	31:0	**Host NQN**: HOSTNQN from Fabric Connect.

6.6.6 Drive Status

The Drive Status object provides information about the physical drives connected to the A2000 Target (either directly or via a PCIe switch network). This information includes various PCIe addressing information assigned to the drive, the make/model/capabilities of the drive, and the current drive status. Command is only valid for drives that are currently present in the system. Refer to the Present/Shutdown Drives *(page 127)* command for a list of currently present drives.

Supported CRUD Operations: READ

RLMP ID

- `0x005`

UART CLI Syntax & Example

- **READ** (Show Drive Status MI): `show-drive <Slot>`

```
t-wdc> show-drive 1
```

```
Drive Status MI:
 + Slot: 1
 + Gen : 3
 + Wdth: x4
 + VID : 0x00031C58
 + SN : STM0001B3C9F
 + MN : HUSPR3216ADP301
 + FW : KMGNP120
 + NSs : 1
 + IOQs: 128
 + BDF : 0x00000400
 + Bar : 0x01200000
 + MxSz: 0x00000081
 + DSTR: 0x00000000
 + RSTS: 0x00000000
 + CSS : 0x00000001
 + MPSm: 0x00000000
 + MPSx: 0x0000000F
 + WARN: 0x00000000
 + MQES: 0x000003FF
 + CQ : 0x00000001
 + AMS : 0x00000000
 + TO : 0x00000050
 + Vers: 0x00010100
 + degC: 40
 + MaxC: 40
```

6.6.6.1 Drive Status Object Format and Field Definitions

Table 95: Drive Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Namespaces								Slot#								Bus/Device/Function															
1	BAR																															
2	Rsvd				Width				Speed				MSI-X Size												#I/O Queues							
3	Drive Temperature																															
4	Max Drive Temperature																															
5	Drive Vendor ID																															
6	Rsvd				Warn				MPSMAX				MPSMIN				Rsvd				CSS							R	DSTRD			
7	TO								Reserved						AMS	CQ	MQES															
8-23	Firmware Version																															
24-55	S/N																															
56-87	Model #																															
88	NVMe Version																															

Table 96: Drive Status Field Definitions

DWord	Bits	Description
0	31:24	**Namespaces**: Number of Namespaces available on this drive
0	23:16	**Slot#**: Chassis/Shelf Slot Number for this drive. **Note**: This field must be set for all CRUD operations.
0	15:0	**Bus/Device/Function**: PCIe bus ID Bus: (value>>8) & 0xFF Device: (value >> 3) & 0x7) Function: value & 0x7
1	31:0	**BAR**: Assigned Base Address Register for drive
2	31:28	**Reserved**
2	27:24	**Width**: 0 = Link Down | 1 = x1 | 2 = x2 | 3 = x4 | 4 = x8 | 5 = x16. All values reserved.
2	23:20	**Speed**: 0 = Link Down | 1 = Gen1 | 2 = Gen2 | 3 = Gen3 | 4 = Gen4. All other values reserved
2	19:8	**MSI-X Size**: MSI-X Table field from the MSI-X Message Control register
2	7:0	**# I/O Queues**: Number of I/O queues allocated to drive
3	31:0	**Drive Temperature**: Temperature in degrees Centigrade **Note**: When this value is returned, a new Smart Log request is sent to the drive to update this temperature for the next GET of this object. So, to get the current temperature, this object should be run a second time.
4	31:0	**Max Drive Temperature**: Max drive temperature since boot in degrees Centigrade.

DWord	Bits	Description
5	31:0	**Drive Vendor ID**: PCIe Vendor ID
6	31:29	**Reserved**
6	28:24	**Warn**: Returns the Critical Warning Field from the SMART/ Health Information Log **Bit Definition** 0 Avail spare space below threshold 1 Over/Below Temp Threshold 2 NVM Subsystem degraded due to media/internal errors 3 Media in read only mode 4 Volatile memory backup device failed
6	23:20	**MPSMAX** (Memory Page Size Max): Maximum host memory page size that the controller supports (2 ^ (12 + MPSMAX)).
6	19:16	**MPSMIN**: Memory Page Size Min
6	15:13	**Reserved**
6	12:5	**CSS** (Command Sets Supported): I/O Command Set(s) that the controller supports **Bit Definition** 0 NVM command set 7:1 Reserved
6	4	**R**: NVM Subsystem Reset Supported: Set to '1' if the controller supports the NVM Subsystem Reset feature.
6	3:0	**DSTRD** (Doorbell Stride): The stride in bytes between submission/ completion doorbell registers (2 ^ (2 + DSTRD)). A value of 0h indicates a stride of 4 bytes, where the doorbell registers are packed without reserved space between each register.
7	31:24	**TO** (Timeout): Worst case time that host software shall wait for CSTS.RDY to transition from: a) '0' to '1' after CC.EN transitions from '0' to '1'; or b) '1' to '0' after CC.EN transitions from '1' to '0'. This worst case time may be experienced after events such as an abrupt shutdown or activation of a new firmware image; typical times are expected to be much shorter. This field is in 500 millisecond units.
7	23:19	**Reserved**
7	18:17	**AMS** (Arbitration Mechanism Supported): Optional arbitration mechanisms supported by the controller
7	16	**CQ** (Contiguous Queues Required): Set if the controller requires that I/O Submission/Completion Queues are required to be physically contiguous
7	15:0	**MQES** (Maximum Queue Entries Supported): Maximum individual Submission queue size that the controller supports. This is a 0's based value. The minimum value is 1h, indicating two entries.
8-23	31:0	**Drive Firmware Version**: 16 character string for firmware version
24-55	31:0	**Serial Number (S/N)**: 32 character string for drive serial number

DWord	Bits	Description
56-87	31:0	**Model #**: 32 character string for drive model number
88	31:0	**NVMe Version**: Indicates the major/minor version of the NVMe specification that the controller implementation supports. Valid versions of the specification are currently: 1.0, 1.1, 1.2, 1.2.1, 1.3, and 1.4. 31:16 0001h Major Version Number (MJR): Major version is 1 15:08 00h Minor Version Number (MNR): can be 00, 01, 02 07:00 00h Reserved

6.6.7 Controller Statistics

The Controller Statistics object provides various network and I/O statistics from A2000 Target such as frame counts, I/O counts, I/O latency, and bandwidth. All counters roll over to 0x0000 after reaching 0xFFFF.

Supported CRUD Operations: READ

RLMP ID

- `0x006`

UART CLI Syntax & Example

- **READ** (Show Stats for Controller MI): `show-stats <INDX>`

```
t-wdc> show-stats 1
```

```
Stats for Controller MI:
    + INDX: 1
    + IOs : 1830
    + CIOP: 0
    + MIOP: 1034
    + BWLO: 0
    + BWHI: 0
    + LAT0: 0
    + RSVD: 0
    + Rx   : 111676
    + Tx   : 7189
    + RxD : 2
    + FRx1: 1748905
    + FTx1: 1644352
    + FRx2: 0
    + FTx2: 0
```

6.6.7.1 Controller Statistics Object Format and Field Definitions

Table 97: Controller Statistics Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																												Index			
	Index 1																**Index 2**															
1	Total I/Os																Host Con Request Total Count															
2	Current IOPs																Host Con Request Success Count															
3	Max IOPs																Con Unexpected Disc Count															
4	Current BW Lo																Backend Total I/O Count															
5	Current BW Hi																I/O Frames Received															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	Current I/O Latency[7]																I/O Frames Transmitted															
7	Reserved																Port 1 Count Enet Link Up/Down															
8	Management Frames Received																Port 2 Count Enet Link Up/Down															
9	Management Frames Transmitted																Port 1 Last Link Up/Down Duration															
10	Received Frames Dropped																Port 2 Last Link Up/Down Duration															
11	Port 1 Frames Received																MIF Dropped															
12	Port 1 Frames Transmitted																CRC Error Counter															
13	Port 2 Frames Received																Port 1 Dropped Frame Count															
14	Port 2 Frames Transmitted																Port 2 Dropped Frame Count															

Table 98: Controller Statistics Field Definitions

DWord	Bits	Description	
0	31:4	**Reserved**	
0	3:0	**Index**: 1: Return counters shown in left column of object format table 2: Show counters in right column of object format table **Note**: This field must be set for all CRUD operations.	
		Index 1	**Index 2**
1	31:0	Total I/Os	Host Con Request Total Count: Total Host Admin and I/O connection requests
2	31:0	Current IOPs: Latest sample period value.	Host Con Request Success Count: Total Successful host Admin and I/O connection requests. Unsuccessful connection request count can be deduced by the difference between **Host Con req total count** and **Host Con req success count**. An example of an unsuccessful connect request is the Host attempting to connect to a non-existent target NQN.
3	31:0	Max IOPs	Connection Unexpected Disc Count: Total unexpected host disconnects
4	31:0	**Current BW Lo**: Lower 32-bits of MB/s, Latest sample period value	Backend Total I/O Count
5	31:0	**Current BW Hi**: Upper 32-bits of MB/s, Assigned Base Address Register for drive	I/O Frames Received

7. The performance counters reported in the show-stats response (IOPs, BW, and latency) are close approximations when the Host does 4K IO workload. For any other IO issued from the Host(s), counters will not report useful information.

DWord	Bits	Description	
6	31:0	Current I/O Latency: ns, Latest sample period value	I/O Frames Transmitted
7	31:0	Reserved	Port 1 Count Enet Link Up/Down: Count bumped each time Ethernet link goes up or down
8	31:0	Mgmt Frames Rx'd: number frames routed to MPU for processing	Port 2 Count Enet Link Up/Down: Count bumped each time Ethernet link goes up or down
9	31:0	Mgmt Frames Tx'd: number frames sent by MPU.	Port 1 Last Link Up/Down Duration: # of seconds since last link up/ down
10	31:0	Rx Frames Dropped: Total number of frames dropped in the Rx path for any reason.	Port 2 Last Link Up/Down Duration: # of seconds since last link up/down
11	31:0	Port 1 Frames Received: number frames received on port 0, including Mgmt frames.	MIF Dropped: # non-I/O frames dropped by hardware due to full queues or firmware due to resource constraints
12	31:0	Port 1 Frames Transmitted: number of frames sent on Port 0, including Mgmt frames.	CRC Error Counter: Frames dropped due to CRC errors
13	31:0	Port 2 Frames Received: number frames received on port 1, including Mgmt frames.	Port 1 Dropped Frame Count
14	31:0	Port 2 Frames Transmitted: number of frames sent on Port 1, including Mgmt frames.	Port 2 Dropped Frame Count

6.6.8 Namespace Status

The Namespace Status object provides drive Namespace info such as Namespace Size, LBAF, etc.

Command is only valid for drives/Namespaces which are currently present in the system. Refer to Present/Shutdown Drives *(page 127)* command for a list of currently present drives. Refer to the Drive Status *(page 86)* Namespaces field to determine the number of valid Namespaces for a given drive.

Supported CRUD Operations: READ

RLMP ID

- `0x007`

UART CLI Syntax & Example

- **READ** (Show NS Status MI): `show-ns <Slot> <NS>`

```
t-wdc> show-ns 1 1
```

```
NS Status MI:
+ Slot: 1
+ NS : 1
+ lbaf: 00090000
++ MS :0000
++ LBADS:512
++ RPERF:0
+ NSHi: 0x00000000
+ NSLo: 0xBA4D4AB0
```

6.6.8.1 Namespace Status Object Format and Field Definitions

Table 99: Namespace Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	NS#								Slot#								Reserved															
1	LBAF																															
2	Namespace Size High																															
3	Namespace Size Low																															

Table 100: Namespace Status Field Definitions

DWord	Bits	Description
0	31:24	**NS#**: Namespsace **Note**: This field must be set for all CRUD operations.

DWord	Bits	Description
0	23:16	**Slot#**: Chassis/Shelf Slot Number for this drive. **Note**: This field must be set for all CRUD operations.
0	15:0	**Reserved**
1	31:0	**LBAF**: LBA Format
2	31:0	Namspace Size High
3	31:0	Namespace Size Low

6.6.9 Network Port MAC Address

The Network Port MAC Address object allows reading of current Ethernet Port MAC address. A READ operation will return the configured values in persistent store.

Supported CRUD Operations: READ

RLMP ID

- `0x008`

UART CLI Syntax & Example

- **READ** (Show MAC Port (Ethernet) MI): `show-mac <Port>`

```
t-wdc> show-mac 1
```

```
MAC Port (Ethernet) MI:
+ Port: 1
+ MAC : 00:0c:ca:12:21:78
```

Note: This example shows the encapsulation used with RLMP over I2C.

6.6.9.1 Network Port MAC Address Object Format and Field Definitions

Table 101: Network Port MAC Address Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Mac Addr High																Reserved												EnetPort			
1	Mac Addr Low																															

Table 102: Network Port MAC Address Field Definitions

DWord	Bits	Description
0	31:16	**Mac Address High**: Upper 16-bits of 6 byte binary MAC address
0	15:4	**Reserved**
0	3:0	**EnetPort**: Network port for which this status is given 1 = 1st port 2 = 2nd port 3 = 1G SGMII port **Note**: This field must be set during a READ operation.
1	31:0	**MAC Address Low**: Lower 32 bits of 6-byte binary MAC address

Table 103: Read Network Port MAC Address Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Rsvd		1						0x008								4															
1	Reserved																														EnetPort	
2	Checksum																Reserved															
3	R	P	0						0x008								8															
4	Mac Addr High																Rsvd														EnetPort	
5	Mac Addr Low																															
6	Checksum																Reserved															

6.6.10 Network Port IPV4 Address

The Network Port IPV4 Address object allows reading/updating the current Ethernet port IPv4 address, IPv4 mask, IPv4 gateway, etc.

Supported CRUD Operations: READ UPDATE

A READ operation will return the configured values in persistent store.

An UPDATE operation will set the values in persistent store & make them active.

RLMP ID

- `0x009`

UART CLI Syntax & Examples

- **READ** (Show IP v4 Address MI): `show-ip <Port>`

```
t-wdc> show-ip 1
```

```
IP v4 Address MI:
+ Port: 1
+ DHCP: 0
+ IPv4: 10.10.10.233
+ Msk4: 255.255.255.0
+ GW4 : 10.10.10.1
+ Host: p1
```

- **UPDATE** (Set IP v4 Address MI): `set-ip <Port> <DHCP> <IPv4> <Msk4> <GW4> <Host>`

  To set the IP address of Ethernet port 1 to 10.10.10.12 and disable DHCP:

```
t-wdc> set-ip 1 0 10.10.10.12 255.255.248.0 10.10.10.1 Port1
```

  To enable DHCP for Ethernet port 1 (static IP address not used):

```
t-wdc> set-ip 1 1 10.10.10.12 255.255.248.0 10.10.10.1 Port1
```

6.6.10.1 Network Port IPV4 Address Object Format and Field Definitions

Table 104: Network Port IPV4 Address Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																D	Reserved											TIdx	EnetPort		
1	IPV4 Address																															
2	IPv4 Mask																															
3	IPv4 Gateway																															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
4-19	IPv4 Hostname																															

Table 105: Network Port IPV4 Address Field Definitions

DWord	Bits	Description
0	31:13	**Reserved**
0	12	**D** (DHCP): DHCP address assignment 0 = disabled 1 = enabled (default)
0	11:4	**Reserved**
0	3	**TIdx**: If set on a READ operation, truncate the Hostname length based on its actual size. If clear, always return 64 bytes for the Hostname. This is to allow minimizing the size of data transferred.
0	2:0	**EnetPort**: Network port for which this status is given 1 = 1st port 2 = 2nd port 3 = 1G SGMII port **Note**: DHCP is not supported on port 3. **Note**: This field must be set for all CRUD operations.
1	31:0	**IPv4 Address**: 4 byte binary IPv4 address assigned to this port
2	31:0	**IPv4 Mask**: 4 byte binary IPv4 subnet mask assigned to this port
3	31:0	**IPv4 GW**: 4 byte binary IPv4 gateway used by this port
4-19	31:0	**IPv4 Hostname**: the length of this field can vary but must be a multiple of 4-bytes (i.e. the entire 4x64 bytes are not required to be sent if the Hostname is shorter than 64 bytes).

6.6.11 Host NQN Table

The Host NQN Table object maps the Host NQN to a WDC Index. The WDC Index is used for the auth tables. Each Host NQN entry is provided by the BMC. WDC Index is returned by the MPU (WDC A2000 Target).

Supported CRUD Operations: CREATE READ UPDATE DELETE

Note: CREATE and UPDATE operations require clearing the Disable Authorization Bit in the Authorization Config *(page 118)* object.

RLMP ID

- `0x00A`

UART CLI Syntax & Example

- **CREATE** (Add HostNQN MI): `add-hostnqn <EMsk> <CMsk> <HNQN>`
- **READ** (Show HostNQN MI): `show-hostnqn <HIdx>`

```
t-wdc> show-hostnqn 0
```

```
HostNQN MI:
+ HIdx: 0
+ EMsk: 0x00000001
+ CMsk: 0x000043E4
+ HNQN: nqn.2014-08.org.nvmexpress:NVMf:uuid:840c5f47-3f45-4040-9fa3-83ec5c8bfbc3
```

- **UPDATE** (Set HostNQN MI): `set-hostnqn <HIdx> <EMsk> <CMsk> <HNQN>`
- **DELETE** (Del HostNQN MI): `del-hostnqn <HIdx>`

Table 106: Host NQN Table

WDC Index	NQN
0	<empty>
1	nqn.host1.com
2	nqn.host2.com
3	nqn.host3.com
...	...

Table 107: In-Band Admin Command Table

Bit	Default	Command
0	0	Delete IO SQ
1	0	Create IO SQ
2	1	Get Log Page

Bit	Default	Command
3	0	Delete IO CQ
4	0	Create IO CQ
5	1	Identify
6	1	Abort
7	1	Set Features
8	1	Get Features
9	1	Async Event Req
10	0	Namespace Mgmt
11	0	FW Commit
12	0	FW Image Download
13	0	Namespace Attach
14	1	Keep Alive
15	0	Format NVM
16	0	Security Send
17	0	Security Receive

6.6.11.1 Host NQN Table Object Format and Field Definitions

Table 108: Host NQN Table Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																EMsk				Rsvd		TI		Host Index							
1	Reserved																In-Band Admin Command Mask															
2-65	Host NQN																															

Table 109: Host NQN Table Field Definitions

DWord	Bits	Description
0	31:16	**Reserved**
0	15:12	**Emsk**: If set, this host is allowed to communicate over Ethernet Port.
0	11:10	**Reserved**
0	9	**TIdx**: If set on a READ operation, truncate the Host NQN length based on its actual size. If clear, always return 256 bytes for the Host NQN. This is to allow minimizing the size of data transferred.
0	8:0	**Host Index** (HIdx): Unique handle into Host NQN Table **Note**: This field must be set for all CRUD operations. **Note**: Host Index field is Reserved on a CREATE from the BMC, and returned by the A2000 Target on the CREATE response.
1	31:18	**Reserved**

DWord	Bits	Description
1	17:0	**In-Band Admin Command Mask**: See Table 57: In-Band Admin Command Table *(page 99)*. If a given bit is set, then this host is allowed to execute the given Admin Command.
2-65	31:0	**Host NQN**: When CREATE / READ / UPDATE are used on this object, the length of this field can vary, but must be a multiple of 4-bytes (i.e. the entire 4x64 bytes are not required to be sent if the NQN is shorter than 256 bytes).

6.6.12 Bridge Control

The Bridge Control object allows setting general Bridge Control values.

Supported CRUD Operations: UPDATE

RLMP ID

- `0x00B`

UART CLI Syntax

- **UPDATE** (Set Control Used to reset RTL MI): `set-control <CDFW> <CD1> <Dflt> <RST>`

6.6.12.1 Bridge Control Object Format and Field Definitions

Table 110: Bridge Control Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																											FW	C	D	R	

Table 111: Bridge Control Field Definitions

DWord	Bits	Description
0	31:5	**Reserved**
0	4:3	**FW** (Firmware): Bit 0 – If set, force the firmware Debug Messages (CLI RAM Buffer) to be written to Flash Bit 1 – If set, and Default Config is set, save Network Information (DHCP, VLAN, IPv4, SNTP, FEC, nTCP, DSCP, MTU) when going back to factory defaults
0	2	**C** (Core Dump): If set, force core dump to persistent store

DWord	Bits	Description
0	1	**D** (Default Config): If set to 1, set Config back to factory defaults This will clear: • Config • Core dumps • Authorization tables This will not clear: • MAC address • Serial number A reboot is required after this bit is set, so it is recommended to set bit 0 (Reset) to 1 when defaulting the Config.
0	0	**R** (Reset): If set to 1, reset the firmware/ASIC

6.6.13 Bridge Settings

The Bridge Settings object allows setting general Bridge Setting values.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x00C`

UART CLI Syntax & Example

- **READ** (Show BridgeSet MI): `show-bridgeset`

```
t-wdc> show-bridgeset
```

```
BridgeSet MI:
+ Mgmt: 0
+ DegC: 95
+ KATO: 1
+ RPID: 0
+ Dual: 0
+ DMI : 0
+ nTCP: 0
+ #DHP: 3
+ FWDL: 0
```

- **UPDATE** (Set BridgeSet MI): `set-bridgeset <Mgmt> <DegC> <KATO> <RPID> <Dual> <DMI> <nTCP> <#DHP> <FWDL>`

6.6.13.1 Bridge Settings Object Format and Field Definitions

Table 112: Bridge Settings Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	M	Bridge Thresh °C							RPID			KATO Timeout			Rsvd		DP	DMI	DHCP Retries						nTCP	FWDL	Reserved					

Table 113: Bridge Settings Field Definitions

DWord	Bits	Description
0	31	**M**: Enable proprietary Western Digital management application access. 0 = Disable listening on TCP Port 9999 (Default) 1 = Enable listening on TCP Port 9999
0	30:24	**Bridge Thresh °C**: When this Bridge Temperature Threshold is exceeded, a notification is sent. Default: 95°C

DWord	Bits	Description
0	23:21	**RPID**: Root Port ID used for the PCIe Root Port on this ASIC
0	20:18	**KATO Timeout**: Number of Keep Alive TimeOuts before closing connection 0 = The 1st time KATO fires, the connection will be closed. 1 = The 2nd time KATO fires, the connection will be closed. Default: 1
0	17:16	**Reserved**
0	15	**DP** (Dual Port2): If set, indicates this A2000 Target connects to the 2nd port of dual-ported drives. Used in HA multi-path chassis configurations. If set, will cause the Controller IDs for this A2000 Target to have the upper bit of the 16-bit Controller ID to be set. This is used to make sure the Controller IDs for the 2 ports on a dual-ported drive end up with a unique controller ID. **Note**: A2000 Target must be reset for this value to take effect.
0	14	**DMI**: Disable MI device support. See Target Management Interface (MI) Device Support *(page 201)* for more information.
0	13:8	**DHCP Retries**: Number of DHCP retries before defaulting to Auto IP for IPv4 configuration. Default: 3 Auto IP is implemented via RFC 3927 - Dynamic Configuration of IPv4 Link-Local Addresses
0	7	**nTCP**: 0 = Enable RoCE Mode 1 = Enable TCP Mode Takes effect on reboot
0	6	**FWDL** (Firmware Download Disable): 0 = Enable (default) 1 = Disable When this bit is cleared, A2000 firmware will look at any NVMe `fw-download` commands to see if the image is meant for its slots. Additionally, the `fw-commit` command for slot 6 and 7 will be intercepted by A2000 and used for itself. When the bit is set, A2000 will assume that incoming `fw-download` and `fw-commit` commands are meant for the drive and pass them along.
0	5:0	**Reserved**

6.6.14 Network Port VLAN / LAG

The Network Port VLAN / LAG object allows reading / updating current Ethernet port VLAN address and enabling / disabling Ethernet port Link Aggregation (LAG).

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x00D`

UART CLI Syntax & Examples

- **READ** (Show VLAN Port (Ethernet) VLAN MI): `show-vlan <Port>`

```
t-wdc> show-vlan 1
```

```
VLAN Port (Ethernet) VLAN MI:
+ Port: 1
+ VLAN: untagged
+ TRNK: 0
```

```
t-wdc> show-vlan 2
```

```
VLAN Port (Ethernet) VLAN MI:
+ Port: 2
+ VLAN: 4
+ TRNK: 0
```

- **UPDATE** (Set VLAN Port (Ethernet) VLAN MI): `set-vlan <Port> <VLAN> <TRNK>`

```
t-wdc> set-vlan 2 4 0
```

6.6.14.1 Network Port VLAN / LAG Object Format and Field Definitions

Table 114: Network Port VLAN / LAG Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	VLAN																TR	Reserved							EnetPort							

Table 115: Network Port VLAN / LAG Field Definitions

DWord	Bits	Description
0	31:16	**VLAN**: VLAN this port is assigned to. Default: 4095 **0 = Priority Tag used, No VLAN** 1-4094 = Tagged VLAN >4094 = Untagged VLAN. If set above 4094, VLAN tagging will not be used.

DWord	Bits	Description
0	15	**TR/TRNK** (LAG): If set, Enable Trunking/Link Aggregation on this port. **Note**: Both ports must have Trunking enabled in order for trunking to be configured. **Default**: Disabled
0	14:4	**Reserved**
0	3:0	**EnetPort**: Network port for which this status is given 1 for the 1st port 2 for the 2nd port 3 for the 1G SGMII port **Note**: This field must be set for all CRUD operations.

6.6.15 Network Port Settings

The Network Port Settings object allows reading / updating the current Ethernet port parameters, such as MTU, Flow Control, etc.

Supported CRUD Operations: READ UPDATE

A READ operation will return the configured values in Persistent Store.

An UPDATE operation will set the values in Persistent Store and make them active. See Persistent Store *(page 186)* for more details.

RLMP ID

- `0x00E`

UART CLI Syntax & Example

- **READ** (Show PortSet (Ethernet) Parameters MI): `show-portset <Port>`

```
t-wdc> show-portset 1
```

```
PortSet (Ethernet) Parameters MI:
+ Port: 1
+ Flow: 1
+ Prio: 3
+ LLDP: 0
+ DCBX: 0
+ MTU : 5000
```

- **UPDATE** (Set PortSet (Ethernet) Parameters MI): `set-portset <Port> <Flow> <Prio> <LLDP> <DCBX> <MTU>`

6.6.15.1 Network Port Settings Object Format and Field Definitions

Table 116: Network Port Settings Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	MTU																R	D	L	Flow Ctl					Reserved							
																				Priority			Enbl									
																									EnetPort							

Table 117: Network Port Settings Field Definitions

DWord	Bits	Description
0	31:16	**MTU**: Maximum Transmission Unit (bytes). Default = 5000 The A2000 Target will infer the MTU for each connection based on host/initiator parameters exchanged during the RDMA connection establishment for RoCEv2, and TCP connection establishment for NVMe/TCP mode, and apply that value for operations conducted over that connection.The user must not change this default value.
0	15	**Reserved**
0	14	**D** (DCBX): Enable or Disable. Default: Disabled
0	13	**L** (LLDP): Enable or Disable. Default: Disabled
0	12:10	**Flow Ctl** (Priority): PFC Priority Level. Default: 3
0	9:8	**Flow Ctl** (Enbl): 00b = Disable Flow Control 01b = Enable Global Pause Flow Control (Default) 10b = Enable PFC Flow Control
0	7:4	**Reserved**
0	3:0	**EnetPort**: Network port for which this status is given 1 = 1st port 2 = 2nd port 3 = 1G SGMII port **Note**: This field must be set for all CRUD operations.

6.6.16 Board Serial Number

The Board Serial Number object allows reading of the device's serial number.

Supported CRUD Operations: READ

A READ operation will return the configured values in persistent store.

RLMP ID

- `0x00F`

UART CLI Syntax & Example

- **READ** (Show SN Serial # MI): `show-sn <INST>`

```
t-wdc> show-sn 1
```

```
SN Serial # MI:
+ INST: 1
+ BdSN: USALP22390002
```

6.6.16.1 Board Serial Number Object Format and Field Definitions

Table 118: Board Serial Number Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																								IID							
1-4	Serial Number																															

Table 119: Board Serial Number Field Definitions

DWord	Bits	Description
0	31:0	**Reserved**
0	3:0	**Instance ID**: Must be set to `0x1` for all CRUD operations
1-4	31:0	**Serial Number**: The configured serial number for this A2000 Target. Assumed to be set once at the time of production.

6.6.17 Target NQN Table

The Target NQN Table object maps the Target NQN to a drive slot number. The NQN is provided by the BMC.

Supported CRUD Operations: READ UPDATE

An UPDATE operation requires clearing the Auto-Populate Target NQN Table Bit in the Authorization Config *(page 118)* object.

RLMP ID

- `0x010`

UART CLI Syntax

- **READ** (Show TargetNQN MI): `show-targetnqn <Slot>`
- **UPDATE** (Set TargetNQN MI): `set-targetnqn <Slot> <TNQN>`

Table 120: Target NQN Table

Target Slot#	Target NQN
0	<empty>
1	nqn.wdc-networks.unique1.com
2	nqn.wdc-networks.unique1.com
3	nqn.wdc-networks.unique1.com
...	...

6.6.17.1 Target NQN Table Object Format and Field Definitions

Table 121: Target NQN Table Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																								TIdx	Slot#						
1-64	Target NQN																															

Table 122: Target NQN Table Field Definitions

DWord	Bits	Description
0	31:8	**Reserved**

DWord	Bits	Description
0	7	**TIdx** (Target Index): If set on a READ operation, truncate the Target NQN length based on it actual size. If clear, always return 256 bytes for the Target NQN. This is to allow minimizing of the size of data transferred.
0	6:0	**Slot#**: Slot number in the chassis/shelf where this drive resides. **Note**: This field must be set for all CRUD operations.
1-64	31:0	**Target NQN**: When READ / UPDATE are used on this object, the length of this field can vary, but must be a multiple of 4-bytes (i.e. the entire 4x64 bytes are not required to be sent if the NQN is shorter than 256 bytes).

Caution: If the same Host connects to drives across more than one A2000 Target, it is possible for the Host to detect duplicate Target NQN names if the Auto-Populate Target NQN Table bit is set in both A2000 Targets. Certain Linux kernel versions will not allow connections to drives that have the same Target NQN name. To avoid this situation, Target NQN names can be made unique by using the `set-targetnqn` command.

6.6.18 Host / Target Auth Table

The Host / Target Auth Table object defines which targets can be accessed by which hosts. Host Index, Target Slot#, Ctlr#, and NS Mask are provided by BMC. WDC Index is returned by the A2000 Target. The A2000 Target enforces at the time of Fabric Connect.

Supported CRUD Operations: CREATE READ UPDATE DELETE

RLMP ID

- `0x011`

UART CLI Syntax

- **CREATE** (Add Auth Host/Tgt MI): `add-auth <HIdx> <Slot> <EIdx> <NS>`
- **READ** (Show Auth Host/Tgt MI): `show-auth <WIdx> <EIdx>`
- **UPDATE** (Set Auth Host/Tgt MI): `set-auth <WIdx> <HIdx> <Slot> <EIdx> <NS>`
- **DELETE** (Del Auth Host/Tgt MI): `del-auth <WIdx>`

Table 123: Host Target Authorization Table

WDC Auth Index	WDC Host Index	Target Slot#	NA Mask (bit for each NS)
0	1	1	0x0001 (NS 1)
1	1	2	0x0003 (NS 1 & 2)
2	5	4	0x0011 (NS 1 & 5)
3	4	3	
...		...	

6.6.18.1 Host / Target Auth Table Object Format and Field Definitions

Table 124: Host / Target Auth Table Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	EIdx								Target Slot#								Host Index								WDC Index							
1	Reserved																NS Mask															
...	DWords 0 and 1 will repeat for the number of entries specified in EIdx.																															

Table 125: Host / Target Auth Table Field Definitions

DWord	Bits	Description
0	31:26	**EIdx**: Number of entries to pack into a CRUD READ: When sending a request: number of entries to return. When receiving a response: count, which starts as 0 for the first entry and increments for each additional entry. If `EIdx` is larger than the actual number of entries, the "empty" entries will return 0xFFs. Ignored for everything but a CRUD READ. Ignored for CLI. A value of 0 defaults to 1. Max: 48
0	25:20	**Target Slot#**: Slot number in the chassis / shelf where the drive resides
0	19:10	**Host Index**: Unique handle from Host NQN Table
0	9:0	**Western Digital Index**: This field is used as an index to retrieve this object **Note**: WDC Index field is reserved on a CREATE from the BMC. Returned by the A2000 Target on the CREATE response.
1	31:16	**Reserved**
1	15:0	**NS Mask**: Mask to determine which Namespaces a given host has access to for this Target NQN Bit 0 = NS 1 Bit 1 = NS 2 ...

6.6.18.2 Setting Access Controls For Drive / Namespace Visibility

This procedure provides an example of setting access controls in the A2000 Target to control drive / namespace visibility to Hosts.

This example implements the following access policy:

- Host #1 to see only drive #1
- Host #2 to see only drive #2

Step 1: Use the `set-authcfg` command to enable authorization:

```
# t-wdc> set-authcfg 0 1 1 0 0
```

Step 2: Use the `set-control` command to reboot the A2000 Target. This will allow the authorization to take effect:

```
# t-wdc> set-control 0 0 0 1
```

See Bridge Control *(page 102)* for details on the object and its commands.

Step 3: Use the `add-host` command to add Host #1 to the internal Host-NQN table:

```
# t-wdc> add-hostnqn 0x3 0x3FFFF host1.nqn
```

This example assumes that Host #1's NQN is `host1.nqn`. The value of the Host NQN is provided to the A2000 Target via the `nvme discover` and/or `nvme connect` commands using the `-q` option (`-q=host1.nqn`). This command will return a WDC Index, which is assumed to be 0 for this example. See Host NQN Table *(page 99)* for details on the object and its commands.

Step 4: Use the `add-host` command to add Host #2 to the internal Host-NQN table:

```
# t-wdc> add-host nqn 0x3 0x3FFFF host2.nqn
```

This example assumes that Host #2's NQN is `host2.nqn`. The value of the Host NQN is provided to the A2000 Target via the `nvme discover` and/or `nvme connect` commands using the `-q` option (`-q=host2.nqn`). This command will return a WDC Index which is assumed to be 1 for this example.

Step 5: Use the `add-auth` command to map these hosts to the targets they are allowed to access:

```
# t-wdc> add-auth 0 1 0 0xFFFF
```

This enables the host at index 0—host #1 from step 3 *(page 114)*—access to all namespaces for drive #1. The mask field can be modified to control which namespaces are visible, if so desired, per the Host / Target Auth Table *(page 113)* section. This example assumes that drive #1 is in slot 1 in the enclosure.

```
# t-wdc> add-auth 1 2 0 0xFFFF
```

This enables the host at index 1—host #2 from step 4 *(page 115)*—access to all namespaces for drive #2. The mask field can be modified to control which namespaces are visible, if so desired. This example assumes that drive #2 is in slot 2 in the enclosure.

Step 6: Issue an `nvme discover` command from each host. Each command must provide the Host NQN name using the `-q` option.

Host #1 will see only drive #1, and Host #2 will see only drive #2. The drive slot values of `1` and `2` shown in the previous examples are reported in the response to the `show-drive 0xff` console command.

6.6.19 Target Slot Map Table

The Target Slot Map Table object maps a PCIe Bus / Device / Function to a Target Drive Slot#. The Bus / Device / Function is provided by the BMC.

Supported CRUD Operations: READ UPDATE

A READ operation will read the values in persistent store.

An UPDATE operation will set the values in persistent store, which will become active on the next reboot.

Note: An UPDATE operation requires clearing the Auto-Populate Slot Map Table Bit in the Authorization Config *(page 118)* object.

RLMP ID

- `0x012`

UART CLI Syntax

- **READ** (Show Slot Map MI): `show-slot <Slot>`
- **UPDATE** (Set Slot Map MI): `set-slot <Slot> <Port> <Bus> <Dev> <Func>`

Table 126: Target Slot Map Table

Target Slot#	Bus/Device/Func
0	0xFFFFFFFF
1	0x03000000
2	0x04000000
3	0x06000000
...	

6.6.19.1 Target Slot Map Table Object Format and Field Definitions

Table 127: Target Slot Map Table Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Bus								Device					Func			Reserved					PCIe Port			Slot#							

Table 128: Target Slot Map Table Field Definitions

DWord	Bits	Description
0	31:24	**Bus**: PCIe bus where this drive resides
0	23:19	**Device**: PCIe device where this drive resides

DWord	Bits	Description
0	18:16	**Func**: PCIe Function where this drive resides
0	15:12	**Reserved**
0	11:8	**PCIe Port**: PCIe port for which this status is given. Must always be 0.
0	7:0	**Slot#**: Slot number in the chassis / shelf where this drive resides **Note**: This field must be set for all CRUD operations.

6.6.20 Authorization Config

The Authorization Config object allows disabling / enabling host / target authorization.

Supported CRUD Operations: READ UPDATE

A READ operation will retreive the values from persistent store; an UPDATE operation will set the values in persistent store, which will become active upon the next reboot.

RLMP ID

- `0x013`

UART CLI Syntax & Examples

- **READ** (Show AuthCfg MI): `show-authcfg`

```
t-wdc> show-authcfg
```

```
AuthCfg MI:
+ DHT : 0
+ Slot: 1
+ TGT : 1
+ Resv: 0
+ DNQN: 0
```

- **UPDATE** (Set AuthCfg MI): `set-authcfg <DHT> <Slot> <TGT> <Resv> <DNQN>`

Below is an overview of how the different Authentication tables relate to each other. When the Disable bit in the Authorization Config is cleared, the following tables will be in effect:

- Host NQN Table
- Host Target Authentication Table
- Target Slot Map tables

Figure 18: Authorization Config

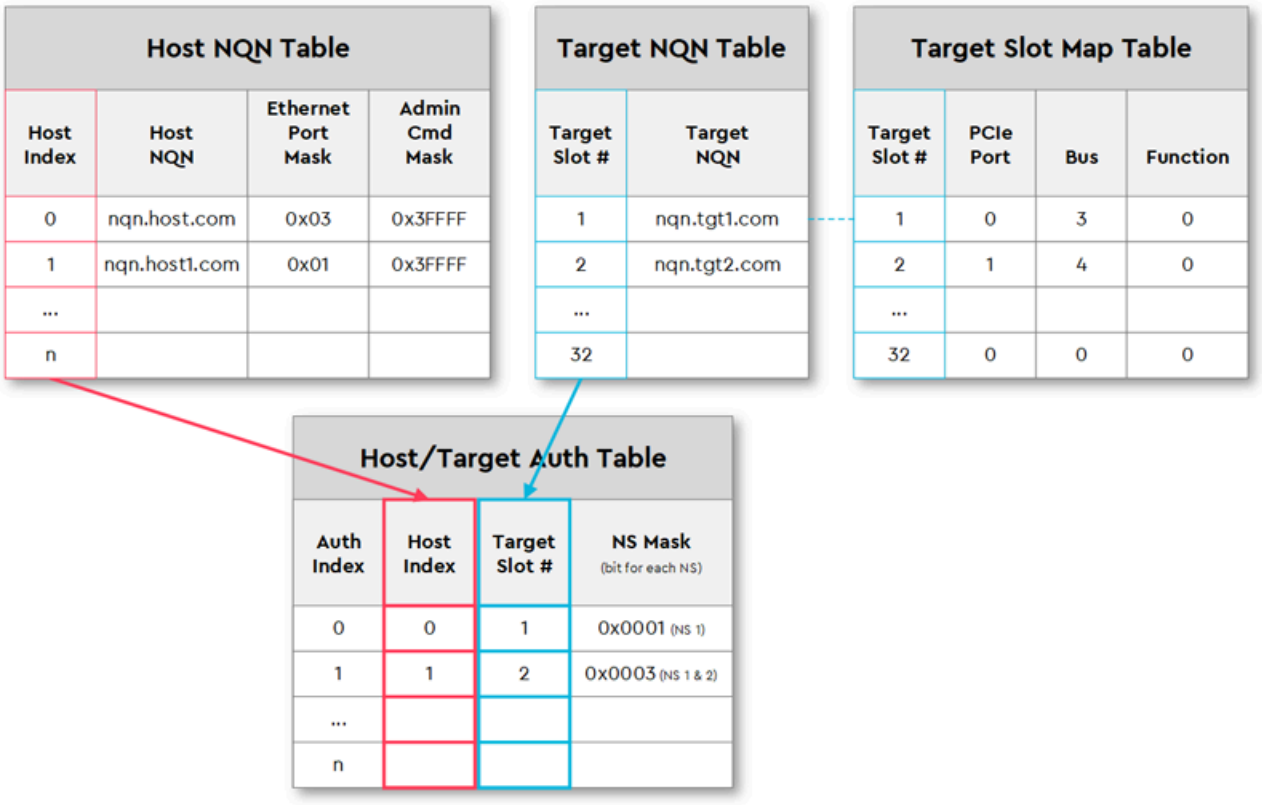


Table 129: Authorization Config Object Field Definitions

Profile #	Description				Host NQN Table	Target NQN Target Slot Map	Host/Target Auth Table
	I/O Conns	Admin Conns	Drives	NSs			
1	128	64	32	128	0-63	0-32	0-63
2	512	128	32	128	0-127	0-32	0-127
3	896	128	32	128	0-127	0-32	0-127
4	3840	256	32	128	0-255	0-32	0-255
5	32	32	8	128	0-31	0-8	0-31

Note: The Target NQN and Slot Map Table sizes can also be retrieved programmatically from the **#SSDs** field in the Bridge Version *(page 73)* command. The Host NQN and Host/Target NQN Auth Table sizes can be retrieved from the **#Virtual Controllers** field in the Bridge Version *(page 73)* command.

6.6.20.1 Authorization Config Object Format and Field Definitions

Table 130: Authorization Config Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																										N	R		T	S	D

Table 131: Authorization Config Field Definitions

DWord	Bits	Description
0	31:6	**Reserved**
0	5	**N** (DNQN): Populate Target NQN Table with Drive NQN 0 = Target NQN Table population is controlled by TGT field. 1 = Target NQN Table will be populated by passing through the NQN reported by the drive in each slot. **Note:** DNQN and TGT can't be enabled at the same time – these fields are mutually exclusive. If TGT is 0, DNQN must be 0 (and vice versa). Setting DNQN to 1 (enabled) when TGT is 0 (enabled) will result in an error. TGT must be 1 when DNQN is set to 1 (enabled).
0	4:3	**R** (Reservations): Enable Reservations 0 = Reservations disabled 1 = The A2000 Target will forward all Reservation requests to the drive (pass-through). 2 = The A2000 Target will emulate reservations.
0	2	**T** (TGT): Auto-Populate Target NQN Table 0 = Target NQN Table will be manually populated 1 = Target NQN Table will be populated as follows: (Default) Slot 1, will contain nqn `nqn.2015-09.com.wdc:nvme.1` Slot 2, will contain nqn `nqn.2015-09.com.wdc:nvme.2` and so forth …
0	1	**S** (Slot#): Auto-Populate Slot Map Table 0 = Target Slot Map Table will be manually populated 1 = Target Slot Map Table will be auto-populated as drives are discovered/enumerated via PCIe by the MPU. In this mode, on Instance 1, the first drive discovered will be put in slot 1, 2nd in slot 2, and so on. (Default)

DWord	Bits	Description
0	0	**D** (Disable Authorization) 0 = Target Slot Map Table, Host NQN Table, and Target NQN tables must be populated. No Hosts will be able to connect unless there is an entry in the Host/Target Auth Table for that host. 1 = the A2000 Target will not enforce Host/Target Authorization Tables. (Default)

6.6.21 Async Notification

The Async Notification object allows the A2000 to asynchronously notify the BMC that a condition occurred, which might require action from the BMC. This command is used to both enable notifications and report which conditions caused the last notification. To report the notification, the A2000's MPU will assert the `GPIO_24 – ATTN REQUIRED` signal from the ASIC. See GPIO Signals *(page 239)* for more information.

Supported CRUD Operations: READ UPDATE

A READ operation shows which condition(s) are causing the notification to occur.

An UPDATE operation is used to determine which conditions will cause an Async notification to occur to the BMC. If the given bit is set, this condition will cause the notification. If the bit is cleared, the condition will not cause a notification to occur. By default, all bits are cleared.

RLMP ID

- `0x014`

UART CLI Syntax & Examples

- **READ** (Show Async Notification MI): `show-async <Clr>`

```
t-wdc> show-async 0
```

```
Async Notification MI:
+ Clr : 0
+ RMsk: 0x00000001
++ Bridge Up
```

- **UPDATE** (Set Async Notification MI): `set-async <Clr> <RMsk>`

```
t-wdc> set-async 0 0x61
```

```
00 00 00 00 61 00 00 00
```

6.6.21.1 Async Notification Object Format and Field Definitions

Table 132: Async Notification Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																															Clr
1	Reserved														DC	DH	NS	Rs	SF	SD	SS	FW	L	W	D	E	T	CD	Rsvd			U

Table 133: Async Notification Field Definitions

DWord	Bits	Description
0	31:1	**Reserved**

DWord	Bits	Description
0	0:0	**Clr** (Clear) On a READ operation, allows BMC to determine when to clear. If Set: • The condition bits will be cleared after the read completes. • The async notification will also be cleared. If Clear: • The condition bits will remain set, and the async notification will remain unchanged. On an UPDATE operation: this field is ignored
	Note:	For any of the bits below, if an UPDATE is being performed: • If the given bit is set, the A2000 Target will monitor the given condition; the Async Notification will be given if a change occurs. Default: All bits set. • If the given bit is cleared, the A2000 Target will not cause an Async Notification to occur if a change in the condition occurs.
1	31:17	**Reserved**
1	16	**DC**: Device Sharing mode Config error
1	15	**DH**: Bad Drive State/Health Read Present Drives object (Type: 3) to determine which drive(s) are in a bad state
1	14	**NS Created**: A new NS was created
1	13	**Reserved**
1	12	**SF**: Sanititize Failed
1	11	**SD**: Sanitize Done
1	10	**SS**: Sanitize Started
1	9	**FW**: FW Download Activated
1	8	**L** (Locate LED): Read `Locate LED` bitmap to determine which drive LED has changed
1	7	**W** (Drive Critical Warning): Read `Drive Status` for each drive to determine which drive(s) have the Warn field
1	6	**D**: Drive was Hot Plugged (in/out). Read `Present Drives` object in order to determine current state of the drives
1	5	**E**: Ethernet Port Link Transitioned (up/down). Read `Network (Ethernet) Status` to determine current Link state.
1	4	**T**: Indicates that the Bridge Temperature Threshold has been exceeded. Read `Bridge Status` to determine current Temperature and Max Temperature

DWord	Bits	Description
1	3	**CD**: Indicates that a core dump has occurred. Read `Bridge Status` to determine the current number of core dumps on the system. NVMe-CLI can be used to retrieve the core dumps.
1	2:1	**Reserved**
1	0	**U**: A2000 Target is fully up and operational

6.6.22 Time Settings

The Time Settings object allows configuring Simple Network Time Protocol (SNTP) Parameters. If SNTP is disabled, this object can be used to set the current time.

Supported CRUD Operations: READ UPDATE

A READ operation will read the values in persistent store.

An UPDATE operation will set and save the values in persistent store:

- If the Enable bit is set, the Time field is ignored on an Update.
- If the Enable bit is cleared, SNTP is disabled and the Time value will be configured as the current time in seconds since.

RLMP ID

- `0x015`

UART CLI Syntax & Examples

- **READ** (Show Time Settings MI): `show-time`

```
t-wdc> show-time
```

```
Time Settings MI:
+ Time: Thu Jan  1 05:14:41 1970
+ SNTP: 1
+ IPV4: 0.0.0.0
```

- **UPDATE** (Set Time Settings MI): `set-time <Time> <SNTP> <IPV4>`

```
t-wdc> set-time 0 0 0.0.0.0
```

```
00 00 00 00 00 00 00 00 00 00 00 00
```

6.6.22.1 Time Settings Object Format and Field Definitions

Table 134: Time Settings Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Time																															
1	Reserved																															E
2	IPv4 SNTP Server Address																															

Table 135: Time Settings Field Definitions

DWord	Bits	Description
0	31:0	**Time**: Current Time reported as GMT For UPDATE operation: • If SNTP enabled: the value specified should be 0 (it is ignored). • Else: value specified will be configured as the current time. The value is the number of seconds since 0h on 1 January 1900 (as per RFC 4330).
1	31:1	**Reserved**
1	0	**E**: Enable SNTP. Default: 1
2	31:0	**IPv4 SNTP Server Address**

6.6.23 Present/Shutdown Drives

The Present/Shutdown Drives object indicates the presence of a drive for a given slot number or which drives to shutdown.

Supported CRUD Operations: READ UPDATE

On a READ operation: indicates if the drive for the given slot number is currently present.

On an UPDATE operation: indicates on which drives to initiate a shutdown.[8] Allows for sending "Normal" or "Abrupt" shutdown to the drive(s). Steps taken are described in the Type of operation. Once shut down, drive link must drop and then be reapplied in order to make the drive accessible again.

Note: There should be one bit set for each `#Drives` count in the Bridge Status *(page 70)* management object. Also, for each bit set, there should be an entry in the Drive Status *(page 86)* management object.

RLMP ID

- `0x016`

UART CLI Syntax & Example

- **READ** (Show Present Drives MI): `show-present <INST>`

```
t-wdc> show-present 1
```

```
Present Drives MI:
+ INST: 1
+ Driv: 0x000001FE
```

- **UPDATE** (Set Present Drives MI): `set-present <INST> <Driv>`

6.6.23.1 Present/Shutdown Drives Object Format and Field Definitions

Table 136: Present/Shutdown Drives Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																														Type	
1	Drive Present / Shutdown / SSD Health Bitmap																															

8. This command may take some time to complete. Therefore, its status must be polled via the In-Progress Status *(page 136)* object. Until the UPDATE command completes, all management commands (besides the In-Progress command) will be blocked.

Table 137: Present/Shutdown Drives Field Definitions

DWord	Bits	Description
0	31:3	**Reserved**
0	2:0	**Type**: READ: • Must be set to 01b or 11b • 0x1: Show present drives bitmap • 0x3: Show drive bitmap for drives with BAD state UPDATE: • 0x0: No action taken • 0x1: Normal Shutdown. Will cause the following steps: **1.** Stop submitting any new I/O commands to the controller. **2.** All connections to the given drive(s) will be closed and have their I/O Submission and Completion queues deleted. **3.** The Shutdown Notification (CC.SHN) field will be set to 1 to indicate normal shutdown operation. • 0x2: Abrupt Shutdown. Will cause the following steps: **1.** Stop submitting any new I/O commands to the controller. **2.** The Shutdown Notification (CC.SHN) field will be set to 1 to indicate normal shutdown operation. • 0x3: Set all drives indicated in the bitmap to the DOWN state. This allows FW to re-poll the specified drives for their link and state information.
1	31:0	READ: **Drive Present Mask**: Bit mask indicating which drives are present. If bit 0 is set, then slot number 0 contains a drive; if bit 1 is set, then slot number 1 contains a drive; etc. UPDATE: **Drive Shutdown/Drive Down Bitmap**: Bitmap indicating which drives to shutdown. If bit 0 is set, then slot number 0 is shut down; if bit 1 is set, then slot number 1 is shut down; etc.

6.6.24 Active Connections

The Active Connections object is a bitmap, indicating if the given CID (Connection ID) currently contains an Active Connection.

Note: There should be one bit set for each `#I/O Connections` and `#Admin Connections` count in the Bridge Status *(page 70)* management object. Also, for each bit set, there should be an entry in the Connection Status *(page 84)* management object.

Supported CRUD Operations: READ

RLMP ID

- `0x017`

UART CLI Syntax & Example

- **READ** (Show Active Connections MI): `show-active <Type> <Grp>`

```
t-wdc> show-active 1 0
```

```
Active Connections MI:
+ Type: 1
+ Grp : 0
+ Msk1: 0xFFFF5555
+ Msk2: 0xFFFFFFFF
+ Msk3: 0xFFFFFFFF
+ Msk4: 0xFFFFFFFF
+ Msk5: 0x00000000
+ Msk6: 0x00000000
+ Msk7: 0x00000000
+ Msk8: 0x00000000
```

6.6.24.1 Active Connections Object Format and Field Definitions

Table 138: Active Connections Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																				Group				Reserved				Type			
1-8	Active Connection Mask																															

Table 139: Active Connections Field Definitions

DWord	Bits	Description
0	31:12	**Reserved**

DWord	Bits	Description
0	11:8	**Group**: Due to the large number of potential connections, a group will be specified to determine which set of CID indices will be returned. Each read can return up to 256 bits (CIDs). **Group** = **CID Range** 0 = 0-255 1 = 256-511 2 = 512-767 3 = 768-1023 4 = 1024-1279 **Note**: This field must be set for all CRUD operations
0	7:4	**Reserved**
0	3:0	**Type**: Type of Connection 0: Return bit mask of all Active Connections 1: Return bit mask of all Active Admin Connections 2: Return bit mask of all Active I/O Connections **Note**: This field must be set for all CRUD operations
1-8	31:0	**Active Connection Mask**: Bit mask indicating which connections (CIDs) are currently active. If bit 0 of DWord 1 is set, then CID 0 contains an active connection. If bit 0 of DWord 2 is set, then CID 32 contains an active connection.

6.6.25 Format NVM

The Format NVM object allows formatting a drive with standard NVMe parameters. Drive formatting should be done via the NVMe CLI when possible. This command is provided if there is no way to access the drive via the NVMe CLI.

Supported CRUD Operations: UPDATE

Note: This command may require a substantial amount of time to complete. Therefore, its status must be polled via the In-Progress Status Command (0x1A). Until this command completes, all management commands, besides the In-Progress command, will be blocked.

RLMP ID

- `0x018`

UART CLI Syntax

- **UPDATE** (Set Format NVM MI): `set-format <Slot> <NS#> <SES> <PIL> <PI> <MSET> <LBAF>`

6.6.25.1 Format NVM Object Format and Field Definitions

Table 140: Format NVM Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																NS#								Slot#							
1	Reserved																				SES			PIL	PI			M	LBAF			

Table 141: Format NVM Field Definitions

DWord	Bits	Description
0	31:16	**Reserved**
0	15:8	**NS#**: Namespace to Format
0	7:0	**Slat#**: Chassis / shelf slot number for the drive to be formatted
1	31:12	**Reserved**
1	11:9	**Secure Erase Settings (SES)**: Specifies if a secure erase & type should be performed. The erase applies to all user data, regardless of location. **Value** **Definition** 0 No secure erase operation requested 1 User Data Erase: All user data shall be erased 2 Crypto Erase: All user data shall be erased cryptographically 3-7 Reserved

DWord	Bits	Description
1	8	**Protection Information Location (PIL)** If set, and protection info is enabled, protection info is transferred as the first eight bytes of metadata. If cleared, and protection info is enabled, protection info is transferred as the last eight bytes of metadata. This setting is reported in the Formatted LBA Size field of the Identify *(page 133)* data structure.
1	7:5	**Protection Information (PI)**: Specifies if end-to-end data protection is enabled and the type of protection information. **Value** **Description** 0 Protection information is not enabled 1 Protection information is enabled, Type 1 2 Protection information is enabled, Type 2 3 Protection information is enabled, Type 3 4-7 Reserved
1	4	**Metadata Settings (M)**: If set, metadata is transferred as part of an extended data LBA.
1	3:0	**LBA Format (LBAF)**: This field specifies the LBA format to apply to the NVM media. Only supported LBA formats shall be selected. See Identify *(page 133)* command (LBAF0-LBAF15) to determine the format to use.

6.6.26 Identify

The Identify object allows an Identify Namespace or Identify Controller command to be issued to a drive.

Supported CRUD Operations: READ

Note: This command may require a substantial amount of time to complete. If the data is available, the command will return immediately with the data shown below. If the data must be read from the drive, an `In-Progress` status will be returned and the data will be all zeros. In this case, this command should be called until `In-Progress` is not returned, indicating the data is now valid. If `In-Progress`, the status can also be polled via the In-Progress Status *(page 136)* command (`0x1A`). Until this command completes, all management commands, other than the In-Progress command, will be blocked.

RLMP ID

- `0x019`

UART CLI Syntax

- **READ** (Show IDEN MI): `show-iden <Slot> <Parm>`
  - where `<Parm>` is defined as (`<format> << 23`) | (`<offset> << 16`) | (`<CNS> <<8`) | `<NS#>`
  - `show-iden 1 1` means Slot 1, Format 0, offset 0, CNS 0, NS# 1
  - `show-iden 1 0x800001` means Slot 1, Format 1, offset 0, CNS 0, NS# 1
  - `show-iden 1 0x810100` means Slot 1, Format 1, offset 1, CNS 1, NS# 0

6.6.26.1 Identify Object Format and Field Definitions

Note: The following table is for Format bit 0. Presently, it is only for CNS 0.

Table 142: Identify Format bit 0

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	F			Offset					CNS								NS#								Slot#							
1	Namespace Size Lo																															
2	Namespace Size Hi																															
3-6	See Spec by subtracting 1 DWord offset*																															
7	Metadata Capabilities								Formatted LBA Size								# of LBA Formats								See Spec*							
8-19	See Spec by subtracting 1 DWord offset*																															
20-22	Reserved (we consume 1 DWord for following items to have no DWord offset)																															
23-31	See Spec*																															
32-47	LBA Format 0-15 Support (LBAF0-15)																															

* See NVM Express Revision 1.4 Identify Namespace Data Structure

Note: The following table is for Format bit 1.

Table 143: Identify Format bit 1

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	F				Offset				CNS								NS#								Slot#							
1-64	See Spec* ((offset * 64 - 1) + current DWord)																															

* See NVM Express Revision 1.4 Identify Namespace Data Structure

Table 144: Identify Field Definitions

DWord	Bits	Description
0	31	**F** (Format): SET: Support new format of 65 DWords CLEAR: Support old format of 48 DWords as was defined for CNS 0 in *RapidFlex A1000 NVMe-oF ASIC DeviceUser Guide*: • Dwords 0 to 18 are at 1 Dword offset in *RapidFlex A1000 NVMe-oF ASIC Device User Guide* "Identify Namespace Format". DWord 19 to DWord 22 are Reserved • Dwords 23 to 47 are at 0 Dword offset in *RapidFlex A1000 NVMe-oF ASIC Device User Guide* "Identify Namespace Format"
0	27:24	**Offset**: Offset represents 64 Dwords of Identify Data Structure. Offset range is from 0 to 15 and we refer NVMe spec using ((offset * 64 - 1) + DWord). For example, if offset is 5 then: • Dword 1 corresponds to (5 * 64) – 1 + 1 = 320 Dword • Dword 2 corresponds to (5 * 64) – 1 + 2 = 321 Dword
0	23:16	**CNS**: • CNS 0: Identify Namespace data structure • CNS 1: Identify Controller data structure (NS# not used)
0	15:8	**NS#**: Namespace to Identify **Note**: This field must be set for all CRUD operations.
0	7:0	**Slot#**: Chassis / Shelf Slot Number for the drive to be Identified **Note**: This field must be set for all CRUD operations.
1	31:0	**Namespace Size Lo** (NSIZE): See Spec*
2	31:0	**Namespace Size Hi** (NSIZE): See Spec*
3-6	31:0	See Spec
7	31:24	**Metadata Capabilities** (MC): See Spec*
7	23:16	**Formatted LBA Size** (FLBAS): See Spec*

DWord	Bits	Description
7	15:8	**Number of LBA Formats** (NLBAF): See Spec*
7	7:0	See Spec*
8-31	31:0	See Spec*
32-47	31:0	**LBA Format**: Indicate the LBA formats 0-15 that are supported by the controller. Fields are defined in the LBA Format Data Structure.[9]

* See NVM Express Revision 1.4 Identify Namespace Data Structure

9. See NVM Express Revision 1.4 Identify - LBA Format Data Structure

6.6.27 In-Progress Status

If a command takes some time to execute, the In-Progress Status object can be polled to determine when the command completes.

Supported CRUD Operations: READ

Note: The following commands use In-Progress Status:

- Format NVM *(page 131)*
- Identify *(page 133)*
- FW Activate *(page 138)*
- FW Download *(page 139)*
- Read Core Dump *(page 146)*
- Namespace Attachment *(page 162)*
- Namespace Management *(page 164)*
- Identify Passthru *(page 172)*

RLMP ID

- `0x01A`

Command Syntax & Example

- **READ** (Show CmdStatus MI): `show-cmdstatus <EX>`

```
t-wdc> show-cmdstatus 0
```

```
CmdStatus MI:
+ Stte: 0x00000000
+ Cmd  : 0x00000000
+ Csrc: 0x00000004
+ EX   : 0
+ OPT1: 0x00000000
+ OPT2: 0x00000000
```

6.6.27.1 In-Progress Status Object Format and Field Definitions

Table 145: In-Progress Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Cmd Source								State								EX	Rsvd								Cmd Code							
1	Optional 1																															
2	Optional 2																															

Table 146: In-Progress Status Field Definitions

DWord	Bits	Description
0	31:24	**Cmd Source**: RLMP commands can arrive to the MPU from different sources. This byte indicates the source protocol of the most recent RLMP command. 1: PCIe 2: I2C 3: UDP 4: WDC CLI
0	23:16	**State**: 0: Command completed successfully 1: Command is still in progress, continue to poll this command 2: Command completed with a failure
0	15	**EX** (Extended Reporting) If this bit is set when a READ is performed, 2 optional DWords will be returned, which provide more details around the specific `In-Progress` command. This is used to provide additional error reporting & info for pass-thru commands to the drive, such as Namespace Management *(page 164)* CREATE command, where there is a need to report back the NSID. If this bit is cleared when a READ is performed, then DWords 1 & 2 are not returned (for backward compatibility).
0	14:13	**Reserved**
0	12:0	**Cmd Code**: Unique Code for the Command for which `In-Progress` is reporting state.
1-2	31:0	**Optional**: These 2 DWords are only returned if the EX bit is set on the READ request. Definitions for these DWords are provided in each command, which can return `In-Progress`. If not documented for a specific command, assume these values are reserved and will return `0x0`s. This field is only valid if the State is not `In-Progress` (`0x1`).

6.6.28 FW Activate

The FW Activate object is used to verify / commit an MPU firmware image.

Supported CRUD Operations: UPDATE

Note: This command may take some time to complete. Therefore, its status must be polled via the In-Progress Status *(page 136)* command (`0x1A`). Until the command completes, all management commands, besides the In-Progress Status *(page 136)* command, will be blocked.

RLMP ID

- `0x01B`

UART CLI Syntax

- **UPDATE** (Set Activate FW MI): `set-activate <Slot> <ACTN>`

6.6.28.1 FW Activate Object Format and Field Definitions

Table 147: FW Activate Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																Action								FW Slot#							

Table 148: FW Activate Field Definitions

DWord	Bits	Description
0	31:16	**Reserved**
0	15:8	**Action** / **Slot#** / **Action Taken** 0 / 6/7 / Image currently in the Staging location replaces the image currently in the specified slot number, provided the CRC is valid. The image is not activated. 1 / 6/7 / Image currently in Staging location replaces the image currently in the specified slot number and is activated at the next reset. 2 / 6/7 / Image in the specified slot number is activated at the next reset (this Action is executed without downloading).
0	7:0	**FW Slot#**: Firmware slot used to store the firmware image. Valid values are 6 or 7

6.6.29 FW Download

The FW Download object allows downloading of a firmware image to a staging area. See Firmware Images Overview *(page 209)*.

Supported CRUD Operations: UPDATE

Note: This command may take some time to complete. Therefore, its status must be polled via the In-Progress Status *(page 136)* command (`0x1A`). Until the command completes, all management commands, besides the In-Progress Status *(page 136)* command, will be blocked.

RLMP ID

- `0x01C`

UART CLI Syntax

- **UPDATE** (Set Download FW MI): `set-download <Len> <OFST> <IMGE>`

6.6.29.1 FW Download Object Format and Field Definitions

Table 149: FW Download Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Length																															
1	Offset																															
2-65	Image																															

Table 150: FW Download Field Definitions

DWord	Bits	Description
0	31:0	**Length**: Number of firmware image bytes for this transaction.
1	31:0	**Offset**: Offset in staging area in which to write the bytes in this transaction. **Note**: The firmware staging area will be erased if this write is to Offset 0.
2-65	31:0	**Image**: Bytes to write for this transaction.

6.6.30 Network Port Auto-Negotiation

The Network Port Auto-Negotiation object allows determining which auto-negotiation parameters to advertise per network port.

Supported CRUD Operations: READ UPDATE

Note: 100G is currently the only speed supported.

RLMP ID

- `0x01D`

UART CLI Syntax & Example

- **READ** (Show AutoNeg (Ethernet) Parameters MI): `show-autoneg <Port>`

```
t-wdc> show-autoneg 1
```

```
AutoNeg (Ethernet) Parameters MI:
+ Port: 1
+ FEC : 0x00000002
+ SPDS: 0x00000008
```

- **UPDATE** (Set AutoNeg (Ethernet) Parameters MI): `set-autoneg <Port> <FEC> <SPDS>`

To set the Ethernet FEC mode and port speed of the A2000 Target using the CLI, execute the appropriate two commands from the following table:

Note: 100G is currently the only speed supported.

Table 151: Port Speed and FEC Configuration Commands

Port Speed	FEC	Port #1 Command	Port #2 Command
100G	RS-FEC (required)	`set-autoneg 1 2 8`	`set-autoneg 2 2 8`
50G	RS-FEC	`set-autoneg 1 2 4`	`set-autoneg 2 2 4`
50G	FC-FEC	`set-autoneg 1 1 4`	`set-autoneg 2 1 4`
25G	RS-FEC	`set-autoneg 1 2 2`	`set-autoneg 2 2 2`
25G	FC-FEC	`set-autoneg 1 1 2`	`set-autoneg 2 1 2`
Disable ports 1 & 2		`set-autoneg 1 0 0`	`set-autoneg 2 0 0`

6.6.30.1 Network Port Auto-Negotiation Object Format and Field Definitions

Table 152: Network Port Auto-Negotiation Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																					FEC			Rsvd				EnetPort			
1	Reserved																								Speed							

Table 153: Network Port Auto-Negotiation Field Definitions

DWord	Bits	Description		
0	31:11	**Reserved**		
0	10:8	**FEC**: 0 = No FEC 1 = FC-FEC 2 = RS-FEC		
0	7:4	**Reserved**		
0	3:0	**EnetPort**: Network port for which this status is given 1 = 1st port 2 = 2nd port **Note**: This field must be set for all CRUD operations.		
1	31:8	**Reserved**		
1	7:0	**Speed**: Bitmask of technologies to advertise. Bit settings are as follows:		
		Bit Value	**CLI**	**Speed**
		-	0	Port disabled
		1	2	25Gb
		2	4	50Gb
		3	8	100Gb (default)
		Note: If multiple bits are set, only the highest speed will be used for configuration		

6.6.31 MAC Interface Counters

The MAC Interface Counters object allows transmitting and receiving MAC interface counters.

Supported CRUD Operations: READ

Most counters are 64-bit and contain a Lo and Hi Dword for the lower and upper 32 bits, respectively. The last few counters are related to congestion control, and are only 32 bits.

RLMP ID

- `0x01E`

UART CLI Syntax & Example

- **READ** (Show MACIF Counters MI): `show-macif <EnetPort>`

```
t-wdc> show-macif 1
```

```
MACIF Counters MI:
+ Port: 1
+ RTOL: 735601
+ RTOH: 0
+ ROOL: 735601
+ ROOH: 0
+ RAEL: 0
+ RAEH: 0
+ RPFL: 0
+ RPFH: 0
+ RTLL: 0
+ RTLH: 0
+ RREL: 0
+ RREH: 0
+ ROFL: 9165
+ ROFH: 0
+ RFEL: 0
+ RFEH: 0
+ ROVL: 0
+ ROVH: 0
+ RTEL: 0
+ RTEH: 0
+ RUPL: 9123
+ RUPH: 0
+ RMPL: 38
+ RMPH: 0
+ RBPL: 4
+ RBPH: 0
+ TTOL: 898250
+ TTOH: 0
+ TOOL: 898250
+ TOOH: 0
+ TPFL: 0
+ TPFH: 0
+ TOFL: 9121
+ TOFH: 0
```

```
+ TOVL: 0
+ TOVH: 0
+ TTEL: 0
+ TTEH: 0
+ TUPL: 9112
+ TUPH: 0
+ TMPL: 7
+ TMPH: 0
+ TBPL: 2
+ TBPH: 0
+ CECT: 0
+ CNPC: 0
+ CSCP: 0
```

6.6.31.1 MAC Interface Counters Object Format

Table 154: MAC Interface Counters Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																								EnetPort							
1	RX Interface Total Octets Lo																															
2	RX Interface Total Octets Hi																															
3	RX Interface OK Octets Lo																															
4	Rx Interface OK Octets Hi																															
5	RX Interface Alignment Errors Lo																															
6	RX Interface Alignment Errors Hi																															
7	RX Interface Pause Frames Lo																															
8	RX Interface Pause Frames Hi																															
9	RX Interface Too Long Lo																															
10	RX Interface Too Long Hi																															
11	RX Interface In Range Errors Lo																															
12	RX Interface In Range Errors Hi																															
13	RX Interface OK Frames Lo																															
14	RX Interface OK Frames Hi																															
15	RX Interface FCS Errors Lo																															
16	RX Interface FCS Errors Hi																															
17	RX Interface OK VLAN Lo																															
18	RX Interface OK VLAN Hi																															
19	RX Interface Total Errors Lo																															
20	RX Interface Total Errors Hi																															
21	RX Interface Unicast Packets Lo																															
22	RX Interface Unicast Packets Hi																															
23	RX Interface Multicast Packets Lo																															
24	RX Interface Multicast Packets Hi																															

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
25	RX Interface Broadcast Packets Lo																															
26	RX Interface Broadcast Packets Hi																															
27	TX Interface Total Octets Lo																															
28	TX Interface Total Octets Hi																															
29	TX Interface OK Octets Lo																															
30	TX Interface OK Octets Hi																															
31	TX Interface Pause Frames Lo																															
32	TX Interface Pause Frames Hi																															
33	TX Interface OK Frames Lo																															
34	TX Interface OK Frames Hi																															
35	TX Interface OK VLAN Lo																															
36	TX Interface OK VLAN Hi																															
37	TX Interface Total Errors Lo																															
38	TX Interface Total Errors Hi																															
39	TX Interface Unicast Packets Lo																															
40	TX Interface Unicast Packets Hi																															
41	TX Interface Multicast Packets Lo																															
42	TX Interface Multicast Packets Hi																															
43	TX Interface Broadcast Packets Lo																															
44	TX Interface Broadcast Packets Hi																															
45	RX Interface ECN Marked Packets																															
46	RX Interface CNP Packets (RoCE) / RX Interface ECE Packets (TCP)																															
47	TX Interface CNP Packets (RoCE) / RX Interface CWR Packets (TCP)																															

Table 155: MAC Interface Counters Field Definitions

DWord	Bits	Description
0-44	31:0	See Table 104: MAC Interface Counters Format *(page 143)*
45	31:0	**RX Interface ECN Marked Packets**: When ECN is enabled, this counter counts the number of IP Header Congestion Encountered (CE) marked packets received on a given port. CE marked packets are an indication that this connection is overloading the fabric and a congestion notification needs to be sent back to the transmitter of this packet so it can reduce its transmit rate.

DWord	Bits	Description
46	31:0	**RX Interface CNP Packets (RoCE)** **RX Interface ECE Packets (TCP)**: When RoCE ECN is enabled, this counter counts the number of RoCE CNP (Congestion Notification Packet) received on a given port. Receiving a CNP packet is an indication that this connection is overloading the downstream path and it needs to reduce its transmit rate. For NVMe-TCP, this counter counts the number of ECE marked packets. This does not count every ECE marked packet, but only the first one during a CE event. This is also a rough count of the number of CWR packets sent, as a CWR is sent when the first ECE is received.
47	31:0	**TX Interface CNP Packets (RoCE)** **RX Interface CWR Packets (TCP)**: When RoCE ECN is enabled, this counter counts the number of RoCE CNP packets sent on a given port. Normally this is sent as a reaction to receiving CE marked packets. A2000 Target does throttle the number of CNP packets if a large number of CE marked packets are received on a given connection back-to-back. For NVMe-TCP, this counts the number of CWR marked packets received. This does not count every CWR marked packet, but only the first one during a CE event.

6.6.32 Read Core Dump

The Read Core Dump object can be used to pull ASIC core dumps or FW Debug Messages from Flash. This information is used when troubleshooting issues with the A2000 Target.

Supported CRUD Operations: READ

Note: The Bridge Control *(page 102)* object can be used to force the FW Debug Messages to be written to Flash. FW Debug Messages are also included when polling an ASIC core dump.

Note: Reading the last entry for a given core dump may require a substantial amount of time to complete. Therefore, status should be polled via the In-Progress Status *(page 136)* command (`0x1A`). Until this command completes, all management commands, besides the In-Progress Status *(page 136)* command, will be blocked.

RLMP ID

- `0x01F`

UART CLI Syntax

- **READ** (Show Core Dump Read MI): `show-core <Slot> <Ofst>`

6.6.32.1 Read Core Dump Object Format and Field Definitions

Table 156: Read Core Dump Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Offset in Core																								Core Dump Slot							
1-64	Image																															

Table 157: Read Core Dump Field Definitions

DWord	Bits	Description
0	31:8	**Offset in Core**: Offset in core dump in which to read the bytes in this transaction. This offset is in terms of 256-byte multiples from the base address of the specified core dump. Example: offset := 5 corresponds to 5*256 = 1280 bytes offset from base address of core dump. When reading an ASIC core dump, this value should range from `0x0-0xEFFF` to pull an entire core dump. When pulling FW Debug Messages, this value should range from `0x0-0x2F00`. **Note**: For the ASIC core dump, pulling entry at `0xEFFF` will cause the given core dump to be erased from Flash. Due to erasing the flash, when reading this entry, an `In-Progress` status will be returned. **Note**: This field must be set for all CRUD operations.
0	7:0	**Core Dump Slot Number**: HW: Use values 0-2 to pull the ASIC core dump(s) FW: Use value 7 to pull the FW Log Messages **Note**: This field must be set for all CRUD operations.
1-64	31:0	**Image**: Core dump values read from this transaction.

6.6.33 Profile Selection

If Profile is set to `0xFF`, then the value in Profile in the EEPROM is used. Any other value set for the Profile in this object will override the EEPROM value. Value will become active on next reboot.

Also allows for adjusting the maximum number of I/O connections a given host can acquire. The default mode of operation is to allow a given Host to open as many connections up to the limit of maximum connections supported by the active profile. In a multi-Host configuration, this **Num Expected Hosts** field can allow the user the option to provide equitable distribution of connection resources across all Hosts. If the **Num Expected Hosts** field is configured to a non-zero value, the A2000 will determine the maximum connections allowed by any Host by dividing the maximum connections supported for the active profile by **Num Expected Hosts**. This value will be rounded down. A Host is identified by the Host-NQN value specified in the Fabric Connect request parameters.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x020`

UART CLI Syntax

- **READ** (Show Profile Selection MI): `show-profile`
- **UPDATE** (Set Profile Selection MI): `set-profile <Prof> <#Hst>`

6.6.33.1 Profile Selection Object Format and Field Definitions

Table 158: Profile Selection Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved								Num Expected Hosts																Profile							

Table 159: Profile Selection Field Definitions

DWord	Bits	Description
0	31:20	**Reserved**
0	19:8	**Num Expected Hosts**: A2000 Target will determine the maximum connections allowed by any Host by dividing the maximum connections supported for the active profile by Num Expected Hosts. **Num Expected Hosts** **Description** 0x00 or > maxIOConns Feature disabled (Default) 1-maxIOConns The # of I/O Connections allowed per host will be maxIOConns divided by this value. Note: This field must be set for all CRUD operations.

DWord	Bits	Description	
		Profile#	**Description**
0	7:0	0-254	See EEPROM Profile description
		255	Use Profile value in EEPROM

6.6.34 VUC Opcode

The VUC Opcode object allows the user to set which Vendor Unique Command (VUC) Opcode to use for the following A2000 Target Functionality:

- Locate LED
- Bridge Control
- Set Vendor Unique Bidirectional command to Read/Write

Note: The valid Opcode range for VUC is 0xC0-0xFF.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x022`

UART CLI Syntax

- **READ** (Show VucOp admin cmd opcode MI): `show-vucop <Command>`
- **UPDATE** (Set VucOp admin cmd opcode MI): `set-vucop <Command> <Opcode>`

6.6.34.1 VUC Opcode Object Format and Field Definitions

Table 160: VUC Opcode Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved								Opcode																Command							

Table 161: VUC Opcode Field Definitions

DWord	Bits	Description
0	31:24	**Reserved**

<table>
<tr><th>DWord</th><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>23:8</td><td>Opcode: Based on the Command specified below, this value will represent the VUC used for that command.

When used with command of 0x01 and 0x02: Valid values are 0xC0-0xFF. If a value less than 0xC0 is used on an Update, the given command will be disabled. When used with command of 0xC1, 0xC2, and 0xC3: Valid values are a 16-bit array representing bi-directional commands (bit 0 is NVMe Opcode 0xC3, bit 1 is NVMe Opcode 0xC7, ..., and bit 15 is NVMe Opcode 0xFF). Bits enabled on 0xC1/0xC2 set the corresponding bi-directional command to Read/Write. Bits enabled for 0xC3 disables the command from being processed (i.e. they will be set to bi-directional). On showvucop, 0xC1 shows which bits are set to Read, 0xC2 shows which bits are set to Write, and 0xC3 shows which bits are bidirectional.</td></tr>
<tr><td>0</td><td>7:0</td><td>Command Description
0x1 Locate LED
0x2 Bridge Control
0xC1 Set Bidirectional to Write
0xC2 Set Bidirectional to Read
0xC3 Set Bidirectional

Bridge Control allows for resetting Bridge or forcing FW Debug Messages (CLI RAM Buffer) to be written to Flash (see Bridge Control object bits 0 and 3). For example, if Opcode is set to 0x7:

To Reset Bridge:
<code>nvme admin-passthru /dev/nvme0n1 -o 0xC7 -- cdw10=1</code>
To Force Firmware Debug to Flash:
<code>nvme admin-passthru /dev/nvme0n1 -o 0xC7 -- cdw10=8</code>

Note: Only values of 1, 8, & 9 are currently valid

Note: This field must be set for all CRUD operations.</td></tr>
</table>

6.6.35 Locate LED

The Locate LED object allows for a Host to issue a command that results in the LED of the specified drive to be lit, thereby having the Host be able to locate a specific drive. Refer to the Locate LED *(page 205)* section for a detailed description of this functionality.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x023`

UART CLI Syntax

- **READ** (Show LocateLed slotmap 0-off/1-on MI): `show-locateled`

```
t-wdc> show-locateled
```

```
LocateLed slotmap 0-off/1-on MI:
+ LEDL: 0x00000001
+ LEDH: 0x00000000
```

- **UPDATE** (Set LocateLed slotmap 0-off/1-on MI): `set-locateled <LED Low> <LED Hi>`

6.6.35.1 Locate LED Object Format and Field Definitions

Table 162: Locate LED Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	LED Low																															
1	LED Hi																															

Table 163: Locate LED Field Definitions

DWord	Bits	Description
0	31:0	**LED Low**: Bitmask indicating if the LED is on (1) or off (0) for each drive slot from 0-31.
1	31:0	**LED Hi**: Bitmask indicating if the LED is on (1) or off (0) for each drive slot from 32-63.

6.6.36 Read VPD

The Read VPD object allows the host to read the VPD information in EEPROM.

Note: This management object is for the ASIC only.

Supported CRUD Operations: READ

RLMP ID

- `0x024`

UART CLI Syntax

- **READ** (Show vpd data MI): `show-vpd <offset> <length>`

6.6.36.1 Read VPD Object Format and Field Definitions

Table 164: Read VPD Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Len																Offset															
1-100	VPD																															

Table 165: Read VPD Field Definitions

DWord	Bits	Description
0	31:16	**Len**: # of bytes of VPD data to read **Note**: This field must be set for all CRUD operations.
0	15:0	**Offset**: Offset from the beginning of the VPD data **Note**: This field must be set for all CRUD operations.
1-100	31:0	**VPD**: VPD data read from EEPROM at Offset for a Length of Len.

6.6.37 Uptime

The Uptime object returns the number of seconds since the last boot/reset.

Supported CRUD Operations: READ

RLMP ID

- `0x025`

UART CLI Syntax & Example

- **READ** (Show Uptime since last boot/reset MI): `show-uptime`

```
t-wdc> show-uptime
```

```
Uptime since last boot/reset MI:
+ uptm: 000:00:21:18
```

6.6.37.1 Uptime Object Format and Field Definitions

Table 166: Uptime Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Uptime																															

Table 167: Uptime Field Definitions

DWord	Bits	Description
0	31:0	**Uptime**: # of seconds since last boot/reset. Format is 000:00:00:00 (day/hour/min/sec).

6.6.38 Read I2C Xcvr

The Read I2C Xcvr object allows the host to read the Xcvr information via I2C.

Supported CRUD Operations: READ

RLMP ID

- `0x026`

UART CLI Syntax

- **READ**: `show-xcvr <poff> <len>`

A couple of examples from SFF-8636:

Table 168: Read I2C Xcvr

Description	PG	Off	Len	CLI	Decode
Device Technology	0	147	1	`t-wdc> show-xcvr 0x9300 1 xcvr I2c data MI: + poff: 0x00009300 + len: 1 000: 00`	850 nm VCSEL
Date Code	0	212	8	`t-wdc> show-xcvr 0xD400 8 xcvr I2c data MI: + poff: 0x0000D400 + len: 8`	190921

6.6.38.1 Read I2C Xcvr Object Format and Field Definitions

Table 169: Read I2C Xcvr Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Port		Len														Offset								Page							
1-64	Xcvr Data																															

Table 170: Read I2C Xcvr Field Definitions

DWord	Bits	Description
0	31:30	**Port**: 0 = Ethernet port 1 1 = Ethernet port 2
0	29:16	**Len**: # of bytes of I2C Xcvr data to read **Note**: This field must be set for all CRUD operations.

DWord	Bits	Description
0	15:8	**Offset**: Offset from the beginning of the I2C Xcvr page **Note**: This field must be set for all CRUD operations.
0	7:0	**Page**: Xcvr Upper Page to read. This is the Page Select Byte (Byte 127) as specified in SFF-8636. **Note**: This field must be set for all CRUD operations.
1-64	31:0	**Xcvr Data**: Xcvr data read via I2C. Variable length up to 64 DWords.

6.6.39 MAC Etherstats Counters

The MAC Etherstats Counters object includes RX and TX MAC interface Etherstats counters. All counters reported in the response to this command are for the Ethernet port specified in the command. The only exception is the **TX discarded packets** counter which is cumulative for both Ethernet ports (or for Port #1 in 1x100G mode). Additionally, the **TX discarded packets** counter is a 32- bit value. All other counters are 64-bit values.

Supported CRUD Operations: READ

RLMP ID

- `0x027`

UART CLI Syntax & Example

- **READ** (Show ethstats Statistics MI): `show-ethstats <RXTX> <Port>`

```
t-wdc> show-ethstats 1 1
```

```
ethstats Statistics MI:
+ RXTX: 1
+ Port: 1
+ STOL: 866532
+ STOH: 0
+ P64L: 16
+ P64H: 0
+ P65L: 10874
+ P65H: 0
+ P12L: 44
+ P12H: 0
+ P25L: 22
+ P25H: 0
+ P51L: 0
+ P51H: 0
+ P10L: 0
+ P10H: 0
+ P15L: 16
+ P15H: 0
+ SPKL: 10972
+ SPKH: 0
+ CRAL: 0
+ CRAH: 0
+ DREL: 0
+ DREH: 0
+ UNPL: 0
+ UNPH: 0
+ OVPL: 0
+ OVPH: 0
+ JBRL: 0
+ JBRH: 0
+ FRAL: 0
+ FRAH: 0
+ DISL: 0
```

```
+ DISH: 0
```

6.6.39.1 MAC Etherstats Counters Object Format and Field Definitions

Table 171: MAC Etherstats Counters Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																								EnetPort				RX/TX			
1	RX/TX etherStatsOctets Lo																															
2	RX/TX etherStatsOctets Hi																															
3	RX/TX etherStatsPkts64Octets Lo																															
4	RX/TX etherStatsPkts64Octets Hi																															
5	RX/TX etherStatsPkts65to127Octets Lo																															
6	RX/TX etherStatsPkts65to127Octets Hi																															
7	RX/TX etherStatsPkts128to255Octets Lo																															
8	RX/TX etherStatsPkts128to255Octets Hi																															
9	RX/TX etherStatsPkts256to511Octets Lo																															
10	RX/TX etherStatsPkts256to511Octets Hi																															
11	RX/TX etherStatsPkts512to1023Octets Lo																															
12	RX/TX etherStatsPkts512to1023Octets Hi																															
13	RX/TX etherStatsPkts1024to1518Octets Lo																															
14	RX/TX etherStatsPkts1024to1518Octets Hi																															
15	RX/TX etherStatsPkts1519toMaxOctets Lo																															
16	RX/TX etherStatsPkts1519toMaxOctets Hi																															
17	RX/TX etherStatsPkts Lo																															
18	RX/TX etherStatsPkts Hi																															
19	RX etherStatsCRCAlignErrors Lo																															
20	RX etherStatsCRCAlignErrors Hi																															
21	RX etherStatsDropEvents Lo																															
22	RX etherStatsDropEvents Hi																															
23	RX etherStatsUndersizePkts Lo																															
24	RX etherStatsUndersizePkts Hi																															
25	RX etherStatsOversizePkts Lo																															
26	RX etherStatsOversizePkts Hi																															
27	RX etherStatsJabbers Lo																															
28	RX etherStatsJabbers Hi																															
29	RX etherStatsFragments Lo																															
30	RX etherStatsFragments Hi																															
31	Rx/TX discarded packets(Port1/2 same) Lo																															
32	Rx discarded packets Hi																															

Table 172: MAC Etherstats Counters Field Definitions

DWord	Bits	Description
0	31:8	**Reserved**
0	7:4	**EnetPort**: Network port for which these counters are given 1 = 1st port 2 = 2nd port (not used in 100Gb mode) **Note**: This field must be set for all CRUD operations.
0	3:0	**RX/TX**: 1 = Retrieve the Receive (RX) counters 2 = Retrieve the Transmit (TX) counters **Note**: This field must be set for all CRUD operations.
1-32	31:0	See Table 121: MAC Etherstats Counters Format *(page 158)*

6.6.40 DSCP

The Differentiated Services (Diffserv) architecture specifies use of the DS field in the IPv4 and IPv6 packet headers to carry one of 64 distinct differentiated services field codepoint (DSCP) values.

The Differentiated Services Field Codepoints (DSCP) object allows reading/updating the setting to enable/disable learning the DSCP codepoint set in received packets and including that value in sent packets.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x02A`

UART CLI Syntax & Examples

- **READ** (Show DSCP Mapping): `show-dscp`

```
t-wdc> show-dscp
```

```
DSCP Mapping:
+ ENAB: 0
+ DSCP: 63
```

- **UPDATE** (Set DSCP Mapping): `set-dscp <ENAB> <DSCP>`

  To enable DSCP support:

```
t-wdc> set-dscp 1 63
```

  To disable DSCP support:

```
t-wdc> set-dscp 0 63
```

6.6.40.1 DSCP Object Format and Field Definitions

Table 173: DSCP Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	EN	Reserved																							DSCP							

Table 174: DSCP Field Definitions

DWord	Bits	Description
0	31	**ENAB** **0**: Disable DSCP support. If disabled, the PFC priority defined in Network Port Settings *(page 108)* will be used.[10] **1**: Enable DSCP support. If enabled, the DSCP value—learned from the latest incoming packet with a DSCP value in the IP header—will override the PFC priority defined in Network Port Settings *(page 108)* and will be used for the IP header in outgoing packets. **Note:** If multiple connections use different DSCP values, the configured PFC priority will be the last one received. Therefore, all connecting hosts should use the same DSCP value.
0	30:6	**Reserved**
0	5:0	**DSCP** 0-63: DSCP value to use in outgoing packets (ignored in A2000 Target)

10. Unless another PFC priority is offered by the peer port in the DCBX exchange, in which case the offered priority will be used.

6.6.41 Namespace Attachment

The Namespace Attachment object is used to pass-thru NS Attach/Detach commands to a specific drive. Refer to NVM Express 1.4 Spec for details. The attach/detach will occur relative to the Controller in which the bridge is connected to the SSD. So the attach/detach will occur to a single Controller only.

Supported CRUD Operations: UPDATE

Note: This command will initiate passing the command through to the drive. Once issued, the user needs to poll the In-Progress object (0x1A) to wait for command to complete. See In-Progress Status *(page 136)*. Until this command completes, all management commands, other than the In-Progress command, will be blocked.

RLMP ID

- `0x02B`

Command Syntax

- **UPDATE** (Set NSAttach MI): `set-nsattach <NS# > <Slot> <ATER>`

6.6.41.1 Namespace Attachment Object Format and Field Definitions

Table 175: Namespace Attachment Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	NS#								Slot#								Reserved															
1	Attach																															

Table 176: Namespace Attachment Field Definitions

DWord	Bits	Description
0	31:24	**NS#**: Namespace **Note**: This field must be set for all CRUD operations.
0	23:16	**Slot#**: Chassis/Shelf Slot Number for this drive **Note**: This field must be set for all CRUD operations.
0	15:0	**Reserved**
1	31:0	**Attach**: 0 = Controller detach 1 = Controller attach

Note: After an Update is performed for this object, the caller should poll the In-Progress object to determine when the command is complete, and to provide Error status. For this command, if the "EX" bit is set when reading the in-progress command, the info returned will be as follows:

Table 177: Namespace Attachment In-Progress Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	As defined for In-Progress Command																															
1	Extended Status																															
2	NS#								Slot#								Reserved															

Table 178: Namespace Attachment In-Progress Field Definitions

DWord	Bits	Description
0	31:0	See In-Progress object (0x1A): In-Progress Status *(page 136)*
1	31:0	**Extended Status**: Contains Completion Queue Entry DWord 3 (Refer to NVM-Express Spec for more details). Bit Definition: 31: Do Not Retry (DNR) 30: More (M) 29:28: Reserved 27:25 Status Code Type (SCT) 24:17: Status Code (SC) 16: Phase Tag (P) 15:0: Command Identifier (CID)
2	31:24	**NS#**: Namespace
2	23:16	**Slot#**: Chassis/Shelf Slot Number for this drive
2	15:0	**Reserved**

6.6.42 Namespace Management

The Namespace Management object is used to pass-thru NS Create/Delete commands to a specific drive. Refer to NVM Express 1.4 Spec for details.

Supported CRUD Operations: CREATE UPDATE DELETE

Note: This command returns a payload for the CREATE & DELETE operations. CREATE will create a NS for the given slot/NS. The NS# is reserved on a CREATE command from the BMC and will be returned from the bridge on the response. DELETE will detach the given NS from the controller for the given slot/NS. UPDATE is used to cause the bridge to re-scan the NS info for a given slot.

Note: The DELETE command will initiate passing the command through to the drive. Once issued, the user needs to poll the In-Progress object (0x1A) to wait for command to complete. See In-Progress Status *(page 136)*. Until this command completes, all management commands, other than the In-Progress command, will be blocked.

RLMP ID

- `0x02C`

UART CLI Syntax

- **CREATE** (Add NSMgmt MI): `add-nsmgmt <Slot> <LSZ > <USZ > <LCAP> <UCAP> <LBAS> <DPS > <NMIC> <AGID> <STID>`
- **UPDATE** (Set NSMgmt MI): `set-nsmgmt <Slot>`
- **DELETE** (Del NSMgmt MI): `del-nsmgmt <NS#> <Slot>`

6.6.42.1 Namespace Management Object Format and Field Definitions

Table 179: Namespace Management Create Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	NS#								Slot#								Reserved															
1	NS Size (lower 32)																															
2	NS Size (upper 32)																															
3	NS Capacity (lower 32)																															
4	NS Capacity (upper 32)																															
5	Reserved								Formatted LBA Size								Reserved															
6	Reserved								NMIC								DPS								Reserved							
7	ANA Group ID																															
8	Reserved								STID								Reserved															

Table 180: Namespace Management Update Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved								Slot#								Reserved															

Table 181: Namespace Management Delete Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	NS#								Slot#								Reserved															

Table 182: Namespace Management Field Definitions

DWord	Bits	Description
0	31:24	**NS#**: Namespace Passed in from BMC for DELETE This field reserved for CREATE **Note**: This field must be set for all READ operations.
0	23:16	**Slot#**: Chassis/Shelf Slot Number for this drive **Note**: This field must be set for all CRUD operations.
0	15:0	**Reserved**
1-2	31:0	**NS Size**: See spec*
3-4	31:0	**NS Capacity**: See spec*
5	31:24	**Reserved**
5	23:16	**Formatted LBA Size**: See spec*
5	15:0	**Reserved**
6	31:24	**Reserved**
6	23:16	**NMIC**: See spec*
6	15:8	**DPS**: See spec*
6	7:0	**Reserved**
7	31:0	**ANA Group ID**: See spec*
8	31:24	**Reserved**
8	23:16	**STID**: NVM Set ID. See spec*
8	15:0	**Reserved**

* See NVM Express Revision 1.4 for definition

Note: After a CREATE or DELETE is performed for this object, the caller should poll the In-Progress object to determine when the command is complete, and to provide Error status. For this command, if the “EX” bit is set when reading the in-progress command, the info returned will be as follows:

Table 183: Namespace Management In-Progress Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	As defined for In-Progress Command																															
1	Extended Status																															
2	NS#								Slot#								Reserved															

Table 184: Namespace Management In-Progress Field Definitions

DWord	Bits	Description
0	31:0	See In-Progress object (0x1A): In-Progress Status *(page 136)*
1	31:0	Extended Status: Contains Completion Queue Entry DWord 3 (Refer to NVM-Express Spec for more details). Bit Definition: 31: Do Not Retry (DNR) 30: More (M) 29:28: Reserved 27:25 Status Code Type (SCT) 24:17: Status Code (SC) 16: Phase Tag (P) 15:0: Command Identifier (CID)
2	31:24	**NS**#: Namespace For **Create**: This value will be the newly created NSID returned from the drive For **Delete**: This is the NSID which was deleted (or attempted to be deleted)
2	23:16	**Slot**#: Chassis/Shelf Slot Number for this drive
2	15:0	**Reserved**

6.6.43 Debug Level

The Debug Level object allows reading / updating current ASIC and firmware logging levels.

Supported CRUD Operations: READ UPDATE

READ: will return the configured values in persistent store.

UPDATE: will make the values active immediately. For firmware, the values will be kept in persistent store.

RLMP ID

- `0x02D`

UART CLI Syntax & Example

- **READ** (Show level HW/FW Log MI): `show-level <HW> <MID>`

```
t-wdc> show-level 1 0xe
```

```
level HW/FW Log MI:
+ HW  : 1
+ MID : 0x0000000E
+ LVL : 1
```

- **UPDATE** (Set level HW/FW Log MI): `set-level <MODE> <MID> <LVL> <BIT>`

6.6.43.1 Debug Level Object Format and Field Definitions

Table 185: Debug Level Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Bit								LVL								MID								Reserved							M

Table 186: Debug Level Field Definitions

DWord	Bits	Description
0	31:24	**Bit**: Location (lower 18 bits of location in debug entry) **Note**: only used if Mode is set to "bit" (i.e. this value is always 0 for HW or FW Modes).
0	23:16	**LVL**: HW: 0-7: ASIC Module Log Level FW: 0-6: FW Module Log Level Bit: 0x0 = disable the given mid/bit debug message 0x81 = If upper bit (of byte) is set, and lower bit is set to 1, will cause coredump to occur

DWord	Bits	Description
0	15:8	**MID**: HW: ASIC Module ID: 0x00-0x1A FW: FW File ID: 0x00-0x36 If 0xFF is used for MID, then all ASIC/FW IDs will be set to eLVL Bit: FileID (upper 8 bits of location in debug entry)
0	7:1	**Reserved**
0	0	**M** (Mode): 0: FW: Set Log Level for FW 1: HW: Set Log Level for ASIC 2: Bit: Enable/disable printing of an individual Debug location message

6.6.44 Namespace Info Bitmap

The Namespace Info Bitmap object is a bitmap, indicating if the given Namespace exists or is active.

Supported CRUD Operations: READ

RLMP ID

- `0x02E`

UART CLI Syntax

- **READ** (Show NSInfo bitmap MI): `show-nsinfo <Type> <Slot>`

6.6.44.1 Namespace Info Bitmap Object Format and Field Definitions

Table 187: Namespace Info Bitmap Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved								Count								Slot								Type							
1-4	NS Info Mask																															

Table 188: Namespace Info Bitmap Field Definitions

DWord	Bits	Description
0	31:12	**Reserved**
0	23:16	**Count**: Count of the total number of NS's created/attached
0	15:8	**Slot**: Chassis/Shelf Slot Number for the drive **Note**: This field must be set for all CRUD operations.
0	7:0	**Type**: Type of Mask 0: Return bit mask of all created NSs 1: Return bit mask of all attached NSs **Note**: This field must be set for all CRUD operations.
1-4	31:0	**NS Info Mask**: Bit mask indicating which NSs are currently created (Type 0) or attached (Type 1) If bit 0 of DWord 1 is set, then NS 1 is created/active. If bit 0 of DWord 2 is set, then NS 32 is created/active.

6.6.45 NNS Status

The NNS Status object provides the NNS target agent status.

Supported CRUD Operations: READ

RLMP ID

- `0x02F`

UART CLI Syntax

- **READ** (Show NNS Status MI): `show-nns <Port>`

6.6.45.1 NNS Status Object Format and Field Definitions

Table 189: NNS Status Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																								R	R	Mode		EnetPort			
1	IPv4/IPv6 Addr (Hi)																															
2	IPv6 Addr (MidHi)																															
3	IPv6 Addr (MidLo)																															
4	IPv6 Addr (Lo)																															

Table 190: NNS Status Field Definitions

DWord	Bits	Description
0	31:8	**Reserved**
0	7	**Run**: NNS is running
0	6	**Reserved**
0	5:4	**Mode**: 0: Not connected to NNS 1: Connected to NNS via IPv4 2: Connected to NNS via IPv6
0	3:0	**EnetPort**: 1 for the 1st port 2 for the 2nd port (not used in 100Gb mode) **Note**: This field must be set for all CRUD operations.

DWord	Bits	Description
1-4	31:0	**IPv4/IPv6 Address**: If Mode: 0: field will be all 0x0s 1: DWord 1 contains IPv4 address of the connected NNS 2: 16 byte IPv6 address of the connected NNS

6.6.46 Identify Passthru

The Identify Passthru object is used to pass-thru Identify Commands to a specific drive/namespace/Controller. The CNS value is as defined in the NVM-Express Base Specification (Revision 1.4).

Note: NS# and CntlID should be 0 if not required for the given CNS.

Supported CRUD Operations: READ

Note: This command may require a substantial amount of time to complete. If the data is available, the command will return immediately with the data shown below. If the data must be read from the drive, an 'In-Progress' status will be returned and the data will be all zeros. In this case, this command should be called until 'In-Progress' is not returned, indicating the data is now valid. If 'In-Progress', the status can also be polled via the In-Progress Status Command (0x1A). Until this command completes, all management commands, other than the In-Progress command, will be blocked.

RLMP ID

- `0x030`

UART CLI Syntax & Example

- **READ** (Show IDPT MI): `show-idpt <NSSl> <Opts>`
  - `NSSl` = `<NS#<<5) | Slot#>`
  - `Opts` = `<(CntlID<<12) | (CNS<<4) | Offset>`
  - For example, to show bytes 256-511 from the Identify Controller Data Structure for drive 8:

```
t-wdc> show-idpt 0x008 0x00011
```

6.6.46.1 Identify Passthru Object Format and Field Definitions

Table 191: Identify Passthru Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CntlID								CNS								Offset						NS#						Slot#			
1-64	Data[11]																															

Table 192: Identify Passthru Field Definitions

DWord	Bits	Description
0	31:24	**CntlID**: Controller Identifier of the primary controller

11. See NVM Express Revision 1.4 Identify Namespace Data Command Data Structures

DWord	Bits	Description
0	23:16	**CNS**: CNS as per the NVM-Express Base Specification. Only valid values from the spec should be used. **Note**: This field must be set for all CRUD operations.
0	15:12	**Offset**: The Identify command will return 256 bytes of data. The offset is the # of 256 byte chunks to skip before pulling the 256 bytes to return. So, a value of: 0: Returns bytes 0-255 1: Returns bytes 256-511 ... The max value supported is 15 (offset of 3840 bytes) **Note**: This field must be set for all CRUD operations.
0	11:5	**NS#**: Namespace to Identify. This value must be 0 for those Identify commands which do not require a Namespace to be specified. **Note**: This field must be set for all CRUD operations.
0	4:0	**Slot#**: Chassis/Shelf Slot Number for the drive to be Identified. **Note**: This field must be set for all CRUD operations.
1-64	31:0	**Data**: Resulting data from the Identify command passed through to the drive. The data presented is offset from the beginning of the data returned from the drive by the "Offset" parameter above.

Table 193: Identify Passthru In-Progress Field Definitions

DWord	Bits	Description
0	31:0	See In-Progress Status *(page 136)* object (0x1A)
1	31:0	Extended Status: Contains Completion Queue Entry DWord 3 (Refer to NVM-Express Spec for more details). Bit Defn: 31: Do Not Retry (DNR) 30: More (M) 29:28: Reserved 27:25 Status Code Type (SCT) 24:17: Status Code (SC) 16: Phase Tag (P) 15:0: Command Identifier (CID)
0	27:24	**Offset**: The Identify command will return 256 bytes of data. The offset is the # of 256 byte chunks to skip before pulling the 256 bytes to return. So, a value of: 0: Returns bytes 0-255 1: Returns bytes 256-511 ... The max value supported is 15 (offset of 3840 bytes)
2	19:16	**CNS**: CNS as per the NVM-Express Base Specification. Only valid values from the Spec should be used.

DWord	Bits	Description
2	15:8	**NS#**: Namespace to Identify. This value must be 0 for those Identify commands which do not require a Namespace to be specified.
0	7:0	**Slot#**: Chassis/Shelf Slot Number for the drive to be Identified.

6.6.47 Perst Setting

The Perst Setting object allows overriding the `perstdelay` and `perstholdlo` settings in EEPROM.

If the value is set to 0xFF for either/both fields, then the value in the EEPROM is used for these values. Any other valid values set will override the EEPROM value. Value will become active on next reboot.

Note: The EEPROM will retain its default value (i.e. if the VPD is read via I2C/EEPROM, it will still contain its original values).

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x038`

UART CLI Syntax

- **READ** (Show PERST MI): `show-perst`
- **UPDATE** (Set PERST MI): `set-perst <DLAY> <HOLD>`

6.6.47.1 Perst Setting Object Format and Field Definitions

Table 194: Perst Setting Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																hold								delay							

Table 195: Perst Setting Field Definitions

DWord	Bits	Description
0	31:16	**Reserved**
0	15:8	**hold** 0-7: Number of 100ms periods to assert PERST# before de-asserting 255: Default: Use persholdlo value in EEPROM
0	7:0	**delay** 0-15: seconds to delay from boot before driving PERST# Low A value of 0 will cause PERST# to not be drive by the MPU 255: Default: Use perstdelay value in EEPROM

6.6.48 SGMII

The SGMII object allows configuration of SGMII attributes.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x03C`

UART CLI Syntax

- **READ** (Show SGMII Config MI): `show-sgmii`
- **UPDATE** (Set SGMII Config MI): `set-sgmii <EN> <PT>`

6.6.48.1 SGMII Object Format and Field Definitions

Table 196: SGMII Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																														PT	EN

Table 197: SGMII Field Definitions

DWord	Bits	Description
0	31:3	**Reserved**
0	2:1	**PT** (100G Passthru Mode) – If enabled, allows SGMII 1G traffic to flow between 1G port and either 100G port #1 or 100G port #2 0 = 100G passthru disabled (default) 1 = 100G passthru enabled to 100G port #1 2 = 100G passthru enabled to 100G port #2
0	0	**EN** (1G Port Enable) – If set, enables SGMII 1G port. If cleared, disables SGMII 1G port. Default: 1

6.6.49 Congestion Control Mode

The Congestion Control Mode object determines the congestion control mechanism or algorithm used to respond to conditions in the network fabric.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x03E`

Command Syntax & Example

- **READ** (Show CCMode (Congestion Control Mode) MI): `show-ccmode`

```
t-wdc> show-ccmode
```

```
CCMode (Congestion Control Mode) MI:
+ MODE: 0
```

- **UPDATE** (Set CCMode (Congestion Control Mode) MI): `set-ccmode <MODE>`

6.6.49.1 Congestion Control Mode Object Format and Field Definitions

Table 198: Congestion Control Mode Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																												Mode			

Table 199: Congestion Control Mode Field Definitions

DWord	Bits	Description
0	31:4	**Reserved**
0	3:0	**Mode** – Determines the algorithm that A2000 runs for congestion control: 0 = Default (disables congestion notification for RoCE, defaults to Reno for TCP, **enables ECN for TCP**) 1 = DCQCN (RoCE only) **Note**: Setting an algorithm that does not match the nTCP setting in `set-bridgeset` will result in default (0) mode behavior.

6.6.50 Data Center QCN Settings

The Data Center QCN Settings object holds the values needed by A2000 to optimize the DCQCN algorithm.

Supported CRUD Operations: READ UPDATE

RLMP ID

- `0x03F`

UART CLI Syntax

- **READ** (Show DCQCN Values MI): `show-dcqcn`

```
t-wdc> show-dcqcn
```

```
DCQCN Values MI:
+ CnpT: 50
+ Time: 1
+ WC  : 0
+ aI  : 100
+ aT  : 1
+ F   : 5
+ g   : 6
+ AddI: 4
```

- **UPDATE** (Set DCQCN Values MI): `set-dcqcn <CnpT> <Time> <WC> <aI> <aT> <F> <g> <AddI>`

6.6.50.1 Data Center QCN Settings Object Format and Field Definitions

Table 200: Data Center QCN Settings Format

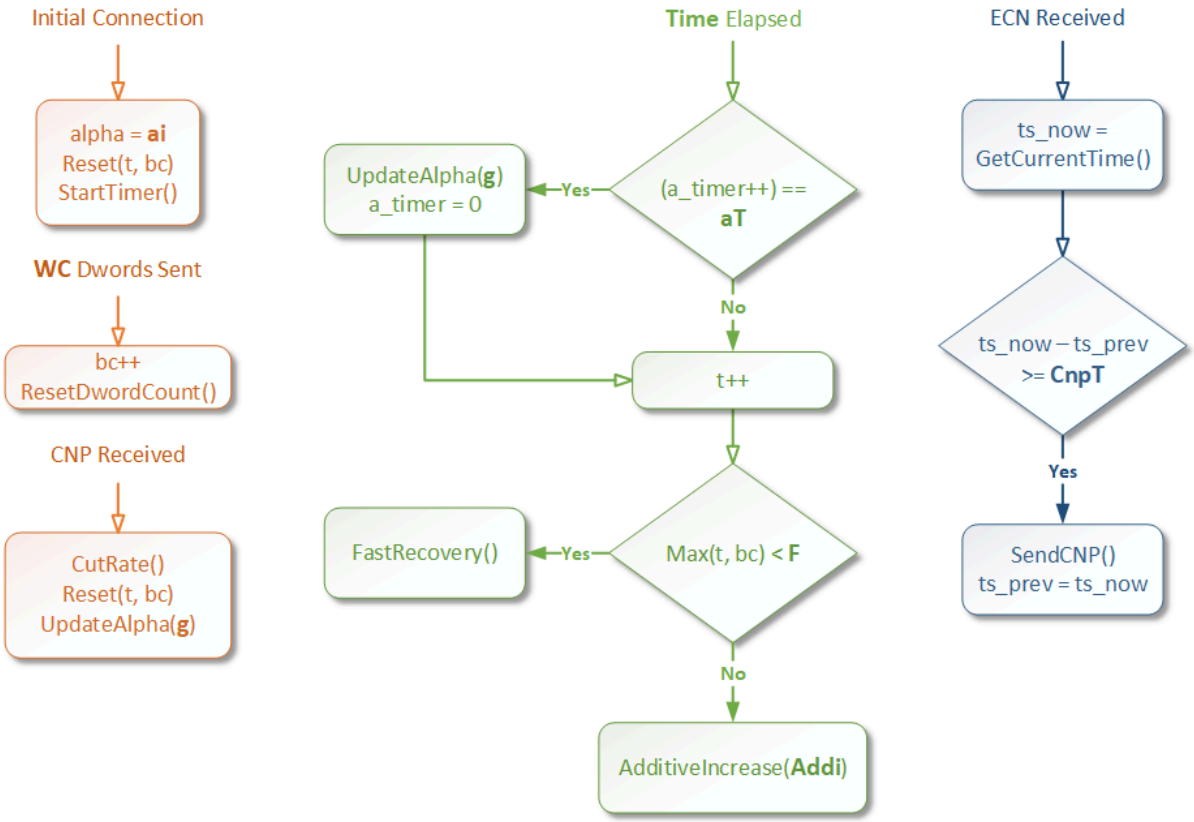
DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CnpT																															
1	Time																															
2	WC																															
3	g								F								aT								ai							
4	Reserved																								Addi							

Table 201: Data Center QCN Settings Field Definitions

DWord	Bits	Description
0	31:0	**CnpT** (CNP Timer) – Determines the minimum time (in microseconds) between A2000 sending CNP packets in response to a CN packet Default: 50

DWord	Bits	Description
1	31:0	**Time** (Algorithm Timer) – In the DCQCN algorithm, this is the main control loop timer that defines the speed at which the algorithm iterates. Units are milliseconds Default: 1
2	31:0	**WC** (DWord Count) – This number is used in place of the byte count (BC) value in the DCQCN algorithm Default: 0
3	31:24	**g** (Constant g) – Represents g in the algorithm, in terms of percent Range: 1-100 Default: 16
3	23:16	**F** (Constant F) – Represents F in the DCQCN algorithm, which determines how many iterations to count before transitioning recovery states Default: 5
3	15:8	**aT** (Alpha Timer Multiple) – An integer multiple of the "Time" variable that determines how often the alpha value is updated Default: 1
3	7:0	**ai** (Alpha Initial) – Initial value of alpha, in terms of percent Range: 1-100 Default: 100
4	31:8	**Reserved**
4	7:0	**Addi** (Additive Increase Percentage) – Percent to increase during the additive increase phase Default: 4

Figure 19: Usage Diagram of Data Center QCN Settings



6.6.51 Logging to Remote Server

The Logging to Remote Server object allows streaming of firmware logs from the A2000 Target to a remote server via a 1G link and UDP port (with port number fixed as 51401). The A2000 Target currently logs ~640 messages to a circular buffer in ASIC RAM. This feature is implemented to capture and store firmware buffer logs in a remote server to debug field issues, as the existing ~640 messages may not suffice. The remote server should be configured to accept messages from UDP port number 51401 with a feature similar to `rsyslog/syslog-ng`.

Supported CRUD Operations: READ UPDATE

Important: The 1G SGMII port (port 3) must be connected and show link up before this feature can be configured.

Note:

- This feature is not persistent across controller resets.
- Only firmware buffer log messages will be transmitted.
- The format of the messages destined for the remote log server complies to Syslog Message Format (reference RFC 5424).

RLMP ID

- `0x45`

UART CLI Syntax and Examples

- **READ** (Show Logging to remote server): `show-logging`

```
t-wdc> show-logging
```

```
Logging to remote server:
+ ENAB: 1
+ IPV4: <ip_address>
```

- **UPDATE** (Set Logging to remote server): `set-logging <EN/DIS> <IPV4 ADDRESS>`

```
t-wdc> set-logging 1 192.168.10.190
```

```
01 00 00 00 be 0a a8 c0
```

6.6.51.1 Logging To Remote Server Object Format and Field Definitions

Table 202: Logging To Remote Server Format

DWd	Byte 3								Byte 2								Byte 1								Byte 0							
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved																								Enable/Disable							
1	IPv4																															

Table 203: Logging To Remote Server Field Definitions

DWord	Bits	Description
0	31:8	**Reserved**
0	8:0	**Enable/Disable**: 0 = Disable logging (default setting) 1 = Enable logging
1	31:0	**IPv4** – IPv4 address of the logging destination server. Default value is: 0.0.0.0

6.7 Host/Target Authorization

The A2000 Target allows for enabling Host/Target Authorization which the administrator can use to specify which Hosts can access which Drives down to the Namespace level. The mechanism to do this is derived from the following tables:

- Host NQN Table
  - Specify unique Host NQN
  - Specify which Ethernet ports a given host can access the Bridge through
  - Specify which In-band Admin commands a given host can perform via Ethernet
- Target NQN Table
  - Specify unique Target NQN
- Host/Target Auth Table
  - Specify which hosts can talk to which targets (down to the Namespace level)

Figure 20: Host/Target NQN

6.8 Namespace Virtualization

The Host/Target Authorization Entry mentioned in Host/Target Authorization *(page 182)* also provides a Namespace Mask field. This allows the bridge to support Namespace Virtualization by restricting which Namespaces on a specific target that a given host can have access to. Refer to the **NS Mask** field in Host / Target Auth Table *(page 113)* management object for a description of the namespace bit settings.

6.8.1 Auto-Populate Auth Tables

The Target NQN Table and Target Slot Map Tables shown above can either be autopopulated by the Bridge, or the Administrator can manually set them. If manually populated, the values will be retained persistently across re-boots.

Manually-Populate Target NQN Table

The Administrator can configure any unique NQN name for each Slot#.

Auto-Populate Target NQN Table

The Target NQN Table will be populated as follows:

- Slot 1, will contain `nqn.2015-09.com.wdc:nvme.1`
- Slot 2, will contain `nqn.2015-09.com.wdc:nvme.2`

Manually-Populate Slot Map Table

The Administrator can map a given PCIe Port, Bus, Device, & Function to a given Target Slot#. This allows the Slot# used by the Bridge to remain consistent to a physical slot on the chassis regardless of which drives are currently populated.

A2000 Target Drive Enumeration

The A2000 Target PCIe port maps to a given partition on the switch.

When the A2000 Target MPU enumerates the PCIe interface, it starts with PCIe Core 0 and walks the ports in order on the switch (so it starts walking the ports in the Partition connected to PCIe Core 0).

For each downstream port discovered, a Bus:Device:Function (BDF) is assigned to each drive. A2000 Target will walk the switch partition in order starting with port 1. The order assigned is BDF 3:0:0, 4:0:0, 5:0:0, etc.

Note: For PCIe Core 1, the enumeration re-starts at BDF 3:0:0.

If the slot map table is not used, once the maximum # of drives is reached (as determined by the selected Profile – see section 6.2.5.1), remaining devices will not be added as potential drives/slots (however, A2000 Target will continue scanning downstream ports and assigning BDFs for up to 32 downstream ports per PCIe Core).

If the slot map table is used, as BDFs are discovered for the drives, it will match these to the slot map table. If it finds an entry in the table, then it will add this as a potential drive. As described above, the scan of downstream ports and assignment of BDFs continues for up to 32 downstream ports per PCIe Core.

With the information above, tools such as lnkstat can be used to view how the downstream ports (drives) are connected via the PCIe switch. If it is determined which partition a given PCIe core is connected to, it can then be determined which drives are also in that partition. By walking the drives for that partition in order, the first drive will start with BDF 3:0:0, the 2nd drive with 4:0:0, etc. From the lnkstat tool, it can be determined which ports have active downstream ports and determine which BDF is assigned to each of these ports. These are the BDFs to be added to the slot map table.

Auto-Populate Slot Map Table

The Slot Map Table will be auto-populated as the PCIe switches are enumerated and end-node downstream ports are discovered. In this mode, the first downstream port discovered will be assigned to slot 1, 2nd to slot 2, and so on.

6.9 NVMe Reservations

NVMe reservations allow for two or more hosts to coordinate access to a shared drive/name space. The A2000 Target supports the reservation feature in one of the following—mutually exclusive—ways:

- **Pass-through mode**: where the reservation commands are passed through to the drive. This mode can be used if the drive natively supports the reservation feature, and in the controlled configuration where a single host connects to the drive on each of the drive's ports.
- **Emulated mode**: where the A2000 Target intercepts the reservation commands and implements the full reservation feature. The optional support for persistence of reservations across a power-loss is not supported.

By default, the A2000 Target is configured to **not** process reservation commands. It can be enabled to process reservation commands by setting the Reservation bit in the Authorization Config *(page 118)* management object.

Persistent Store

In This Chapter:

7.1 Persistent Store Overview

The A2000 Target uses two methods to store operational data persistently:

1. **Flash**
    - Bootloader/Golden Firmware images programmed at manufacturing time
    - Updateable firmware images (Slot 6 & Slot 7 Images)
    - Configuration information such as:
        - IP addresses
        - Which firmware image to boot
        - Ethernet port configuration
        - Authentication parameters
    - Troubleshooting information such as:
        - Core dumps – see Pulling Core Dumps *(page 213)*

    Note: The Configuration information can be set in the field using RLMP/CLI commands for setting Management Objects. See Management Objects *(page 64)*.

2. **EEPROM/VPD**
    - Used to store bridge and SKU-specific information in Vital Product (VPD) format.

The remainder of this chapter documents the contents of the EEPROM / VPD portion of Persistent Store.

7.2 EEPROM/VPD

The A2000 Target typically utilizes an external EEPROM to store Vital Product Data (VPD) which can be configured at the time of manufacturing of the associated board level product. The VPD is accessible by a management entity through the BMC I2C interface. It is also accessible by the MPU on the same shared I2C interface.

7.2.1 EEPROM Access

The A2000 Target's internal MPU will access the EEPROM at boot time at I2C address 0x53 (0xA6 if including the direction bit). When accessing the EEPROM, the system containing A2000 Target needs to ensure that the only entities on the I2C 0 channel (connecting the MPU to the EEPROM) are the A2000 Target and the EEPROM.

If the system is designed to allow a BMC to also access the EEPROM, a hardware circuit should be implemented which allows the A2000 Target MPU to isolate this interface during this time.

To achieve this:

- Before the MPU accesses the EEPROM, it will assert GPIO 11 (Isolate EEPROM)
    - GPIO 11 asserted should cause the BMC to isolate from the I2C 0 channel
    - GPIO 11 asserted should also cause the EEPROM to be taken out of writeprotect mode
    - The MPU will then act as an I2C master to be able to access the EEPROM
- Once the MPU completes accessing the EEPROM, it will de-assert GPIO 11

- Prior to de-assertion, MPU will go back to I2C slave mode
- De-asserting GPIO 11 should put the BMC back on the I2C bus to access the MPU/EEPROM
- De-asserting GPIO 11 low should cause the EEPROM to be in write-protect mode

7.3 Data Format

The data format will comply with the IPMI Platform Management FRU Information Storage definition (v1.0 Document Revision 1.3). It includes the following tables:

- Common Header
- Product Info Area
- Multi-Record Area
- NVMe MultiRecord Area information
- NVMe PCIe Port MultiRecord Area information
- Board Info Area. The Board Area information will contain extra fields for MAC addresses and other unique configuration/FRU information.

7.3.1 Common Header

Table 204: EEPROM Common Header

Common Header (offset 0x000)		
Field Length	Value	Field
1	0x01	Version
1	0x00	Internal Use Area Offset
1	0x00	Chassis Info Area Offset
1	0x1B	Board Area Offset
1	0x01	Product Info Area Offset
1	0x0f	MultiRecord Area Offset
1	0x00	PAD
1	0xD4	Checksum

7.3.2 Product Information

Table 205: EEPROM Product Information

Product Info (offset 0x008)		
Field Length	Hex Value	Field
1	0x01	Version
1	0x0E	Length (in multiples of 8 bytes)

Product Info (offset 0x008)		
Field Length	Hex Value	Field
1	0x00	Language Code
1	0x08	Manufacturer Name type/len (8)
8	0x77 64 63 00 00 00 00 00	Manufacturer Name (**wdc**)
1	0x18	Product Name type/length (24)
24	0x41 32 30 30 30 46 61 62 72 69 63 42 72 69 64 67 65 44 65 76 69 63 65	Product Name (**A2000FabricBridgeDevice**)
1	0x28	Product Part/Model Number type/length (40)
40	0x31 4b 30 30 30 34 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 **or** 0x31 4b 30 30 30 34 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Product Part/Model Number (**1K00041, 1K00043**)[12]
1	0x20	Product Version type/length (2)
2	0x30 00	Product Version (0)
1	0x14	Product SN type/length (20)
20	(Ex: 0x38 35 31 4b 31 30 39 34 31 31 30 30 30 38 00 00 00 00 00 00)	Product SN (Ex: 851K1094110008)
1	0x00	Asset Tag type/length (0)
0	<empty>	Asset Tag
1	0x00	FRU File ID type/length (0)
0	<empty>	FRU File ID
1	0xC1	End of Fields
6	0x00 00 00 00 00 00	PAD
1	0xB2	Checksum

7.3.3 NVMe Multi-Record Area

Note: Values shown in italics are examples characteristic of the RapidFlex C2000 Fabric Bridge Adapter. These values will likely be different for unique PC board applications of the RapidFlex A2000 Fabric Bridge Device.

12. The listed number is the default SKU, but the number can be customized.

Table 206: NVMe Multi-Record Area

NVMe MultiRecord Area (offset 0x078)		
Field Length	Value	Field
1	0x0B	NVMe Record Type
1	0x02	Record Format Version
1	0x40	Record Length
1	<checksum>	Record Checksum
1	0x1E	Header Checksum
1	0x00	Version
1	0x21	Mgmt Endpoint Form Factor (Low Profile) (HHHL)
1	0x00	Reserved
1	0x00	Initial 1.8V Power Supply
1	0x00	Max 1.8V Power Supply
1	0x00	Initial 3.3V Power Supply
1	0x00	Max 3.3V Power Supply
1	0x00	Reserved
1	0x01	Max 3.3V aux Power Supply
1	0x00	Initial 5V Power Supply
1	0x00	Max 5V Power Supply
1	0x08	Initial 12V Power Supply
1	0x11	Max 12V Power Supply **Note:** C2000 only requires 12V power rail
1	0x11	Max Thermal Load
1	0x00	Total NVM Capacity (Not Supported)
1	0x00	Reserved

7.3.4 NVMe PCIe Port Multi-Record Area

Table 207: NVMe PCI Port Multi-Record Area

NVMe PCIe Port MultiRecord Area (PCIe 0/1) (offset 0x0B8/0x0C8)			
Field Length	Value Port 0	Value Port 1	Field
1	0x0C	0x0C	NVMe PCIe Port Record Type
1	0x02	0x82	Record Format Version (Format Version2; end of list for Port 1)

NVMe PCIe Port MultiRecord Area (PCIe 0/1) (offset 0x0B8/0x0C8)			
Field Length	Value Port 0	Value Port 1	Field
1	0x10	0x10	Record Length
1	0xE8	0xFF	Record Checksum
1	0xFA	0x63	Header Checksum
1	0x00	0x00	Version
1	0x00	0x01	PCIe Port #
1	0x00	0x00	Port Information
1	0x00	0x07	PCIe Link Speed (2.5/4.0/8.0 suported)
1	0x10	0x00	PCIe Max Link Width (16 port 0; 8 port 1)
1	0x00	0x00	MCTP Support (Not supported)
1	0x01	0x0	Ref Clk Capability (Common/Separate ReClk)
4	0x00 00 00 00	0x00	Reserved

7.3.5 Board Information Area

Table 208: Board Information

Board Info Area (offset 0x0D8)		
Field Length	Value	Field
1	0x04	Version
1	0x0B	Length (in multiples of 8 bytes)
1	0x00	Language Code
3	(Ex: 0xEE 59 B8)	Mfg Date
1	0x00	Board Mfg Type/Len
0	<empty>	Board Mfg
1	0x00	Board Product Name Type/Len
0	<empty>	Board Product Name
1	0x00	Board SN Type/Len
0	<empty>	Board SN
1	0x00	Board Part # Type/Len
0	<empty>	Board Part #
1	0x00	FRU File ID Type/Len
0	<empty>	FRU File ID
1	0x12	MAC Addr 1 Type/Len

Board Info Area (offset 0x0D8)				
Field Length	**Value**	**Field**		
18	<Board Specific>	MAC Addr 1		
1	0x12	MAC Addr 2 Type/Len		
18	<Board Specific>	MAC Addr 2		
1	0x12	MAC Addr 3 Type/Len		
18	<Board Specific>	MAC Addr 3 (SGMII MAC)		
1	0x01	Profile Type/Len		
1	<Board Specific>	Profile (See Profiles Table below)		
1	0x1	Direction Type/Len		
1	FRU Specific (Ex: 0x00)	Direction		
		Bit	**Values**	**Description**
		2-7	0x0	Reserved
		1	0: Default 1: SGMII autoneg as received	SGMII mode
		0	0: On-board (default) 1: Route via PCIe fingers	UART Direction
1	0x01	PCIe Type/Len		
1	FRU Specific (Ex: 0x10)	PCIe		
		Bit	**Values**	**Description**
		7:6	0: 100ms (Default) 1: 200ms 2: 300ms 3: 400ms	Hold PRST Low for a certain # of 100ms intervals. Only valid if PRST Delay > 0
		5	0	Reserved
		4	0	Reserved
		3:0	0: Don't drive PRST (Default) 1-15: #secs to delay	PRST Delay in secs before driving PRST out from MPU
1	0x01	PCIe Equalization Type/Len		
1	FRU Specific (Ex: 0x04)	PCIe Equalization		
		Bit	**Values**	**Description**
		7:4	0x00	Reserved

Board Info Area (offset 0x0D8)				
Field Length	**Value**	**Field**		
		3:0	0-15: 0x4: Default	Upstream port xmit preset (k_equpreset)
1	0x08	PCIe Equalization to use Type/Len		
8	FRU Specific (Ex: 0x70 00 00 00 00 00 00 00)	PCIe Equalization to use		
		Bit	**Values**	**Description**
		63:12	0x00	Reserved
		11:0	0x70 (Default)	k_preset_to_use
1	0xC1	End of Fields		
1	0x00	Pad		
1	0xED	Checksum		

Profiles

Table 209: Profiles

Profile #	NVMe-oF[13]		# Drives	# NS
	I/O Conns	Admin Conns		
1	128	64	32	128
2	512	128	32	128
3	896	128	32	128
4	3840	256	32	128
5	32	32	8	128

Note: Profile 2 is the default profile. Profile 4 is only for RoCE.

Note: When reservations are enabled, the reservation profile selection is clamped to profiles 1-2; a profile of greater than 2 becomes 2 on reset. It does not change the saved value, only the runtime value. Profiles 1 & 2 get the following values:

1. 128/32/8/16
2. 512/64/32/16

Note: Additional profiles may be provided based on customer requirements. Contact Western Digital directly in order to discuss specific configurations.

13. The number of I/O and Admin connections may be limited by firmware. Please see the latest firmware Release Notes for details.

In-Band Ethernet Connectivity

In This Chapter:

8.1 In-Band Ethernet Connectivity

The A2000 Target's primary Ethernet interface consists of two Media Access Controllers (MAC) and serves the following functions:

- Datapath to/from A2000 Target using RDMA (RoCE) or NVMe-TCP
- Networking services to/from A2000 Target

Ethernet Connectivity Options

Up to two Ethernet Ports (MACs) can be used to connect A2000 Target in-band. Currently supported connectivity options include:

- Cable Types
  - Active Optical: 1 meter, 2 meter, and 3 meter
  - Passive Copper: 1 meter, 2 meter, and 3 meter
- Port Speeds – see Network Port Auto-Negotiation *(page 140)*
  - 25G – Single Lane
  - 50G – Dual 25G Lanes
  - 100G – Four 25G Lanes
- FEC Modes – see Network Port Auto-Negotiation *(page 140)*
  - RS-FEC – Reed Solomon Forward Error Correction
  - FC-FEC – Fire Code Forward Error Correction
  - No FEC
- Pause – see Network Port Auto-Negotiation *(page 140)*
  - RX Pause Flow Control
  - TX Pause Flow Control
- VLAN per port – see Network Port Auto-Negotiation *(page 140)*
  - Priority Tag with no VLAN – Tagged VLAN (1-4094) – Untagged VLAN.
- MTU up to 5000
- Link Aggregation – see Network Port Auto-Negotiation *(page 140)*
  - When configuring 2 ports, Trunking/Link Aggregation can be used to aggregate the 2 ports as a single interface. As RoCE/NVMe-TCP connections are made, A2000 Target will assign each of these connections to a specific physical port in order to aggregate the traffic across both physical ports.
- IPv4/IPv6 Support
- NVMe-TCP/RoCEv1/RoCEv2

Ethernet ports can be configured in Manual Mode or they can be configured to AutoNegotiate. See the "Network Port Auto-Negotiation" section for details on how to configure the ports.

8.1.1 Ethernet Lane Configuration and Ordering

Additionally, when configured for 2-ports, the least significant lanes in that port must be mapped into lanes zero and two.

- 25Gb: A2000 Target lanes 0 or 2, or both for 2 x 25Gb
- 50Gb: A2000 Target lanes 0,1 or 2,3 with lowest order lanes mapped into lanes 0 or 2 respectively. All four lanes to be connected for 2 x 50Gb
- 100Gb: A2000 Target lanes 0,1,2,3 with the least significant lane mapped into lane 0

8.2 Network Services

Currently supported network services include:

- IPv4 DHCP – see Network Port IPV4 Address *(page 97)*
  - IP, Subnet Mask, Gateway, Hostname configuration
- ICMP (Ping) - admin mode only
- ARP - admin mode only
- IPv6 Neighbor Discovery
- IPv6 Router Discovery
- LLDP - Link Layer Discovery Protocol – see Network Port Settings *(page 108)*
- DCBX - Data Center Bridging eXchange – see Network Port Settings *(page 108)*
- Flow Control – Global Pause/PFC
- SNTP Client
  - Used for more accurate timestamps during logging

A2000 Target Operation

In This Chapter:

9.1 Supported PCIe Switch Topologies

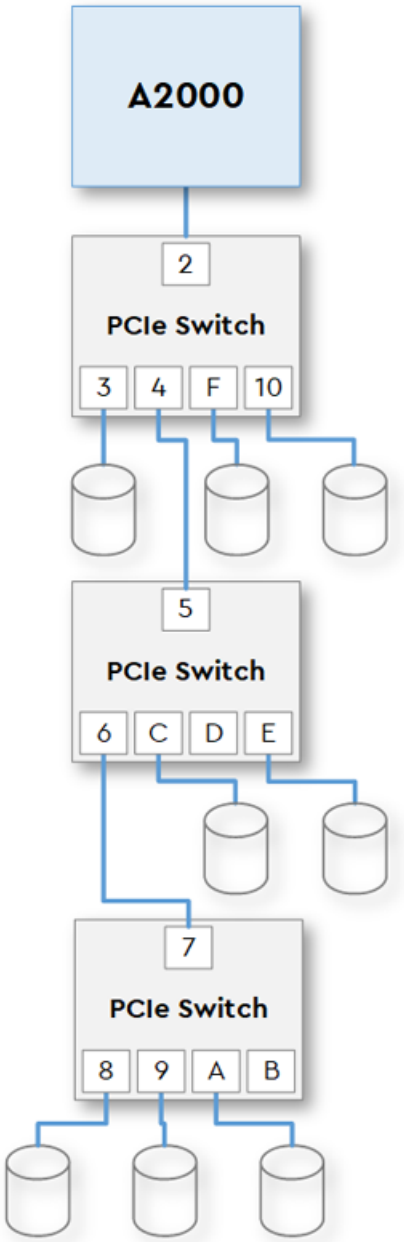
The A2000 Target utilizes a single PCIe Gen4 16-lane root complex to connect to NVMe storage devices. This root complex does not support bifurcation. It must be connected to a single PCIe device. That single PCIe device could be a single NVMe device (SSD), but it is more commonly expected that the A2000 Target will be directly connected to a PCIe switch.

PCIe Bus Enumeration

Prior to powering up the A2000 Target, all PCIe switches must be fully initialized. This is to ensure full PCIe switch enumeration can occur by the A2000 Target when first powering up. Hot swap of PCIe switches is not supported.

Bus enumeration is done depth first. A total of 32 buses can be enumerated by the A2000 Target. An example is shown in the following diagram:

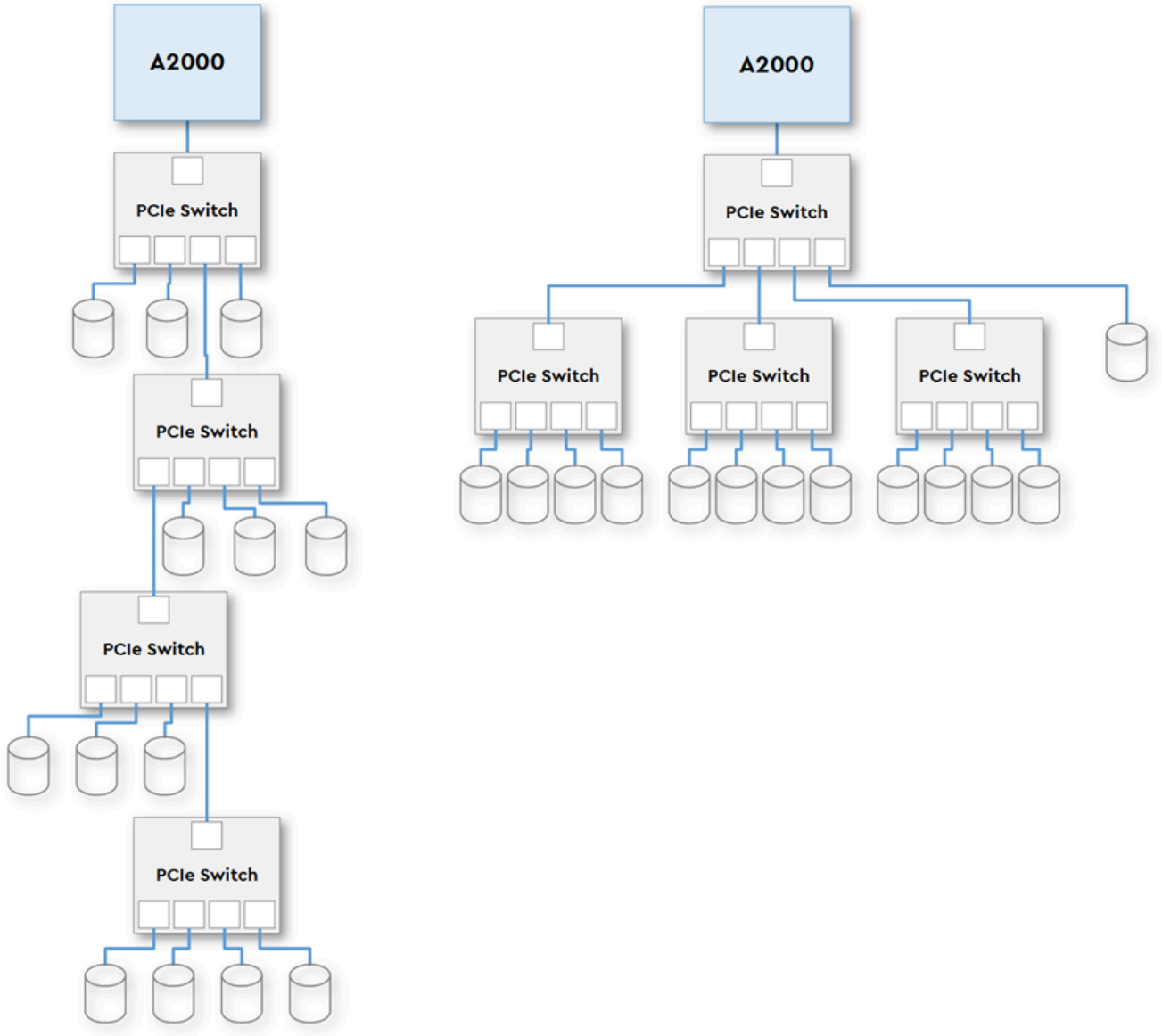
Figure 21: Example PCIe Switch Cascading



PCIe Switch Topology

As stated earlier, there is a limit of 4 PCIe switches for the A2000 Target PCIe Core. Two examples of how these switches may be interconnected are shown below:

Figure 22: Example Maximum PCIe Switch Cascading



9.2 Target Management Interface (MI) Device Support

In addition to the enumeration described in Supported PCIe Switch Topologies *(page 198)*, the A2000 Target can also discover a single MI Device connected to the PCIe fabric. Currently, support is provided when this MI device is embedded as a separate Function within a PCIe switch. From the perspective of the A2000 Target, the MI device hangs off the PCIe bus the same as any other downstream port (such as a port connected to an SSD). The solution is shown in the picture below:

Figure 23: Target MI Device Physical View

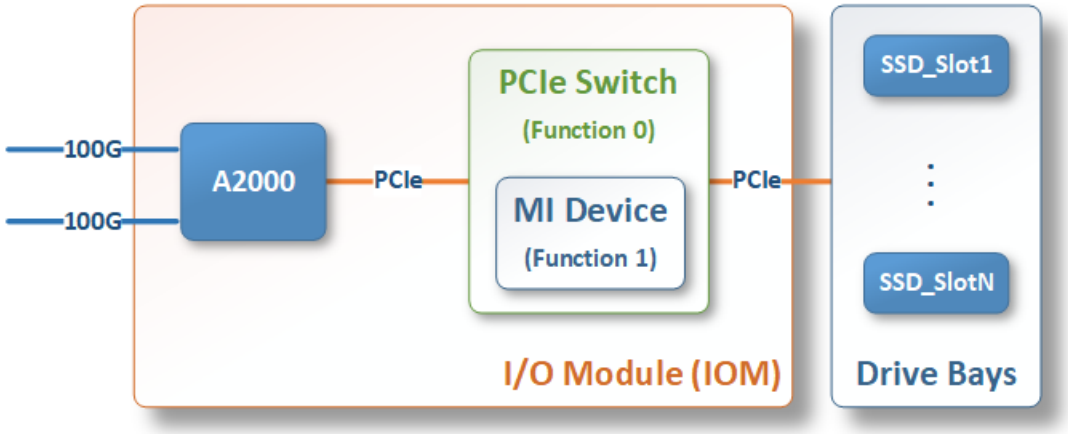
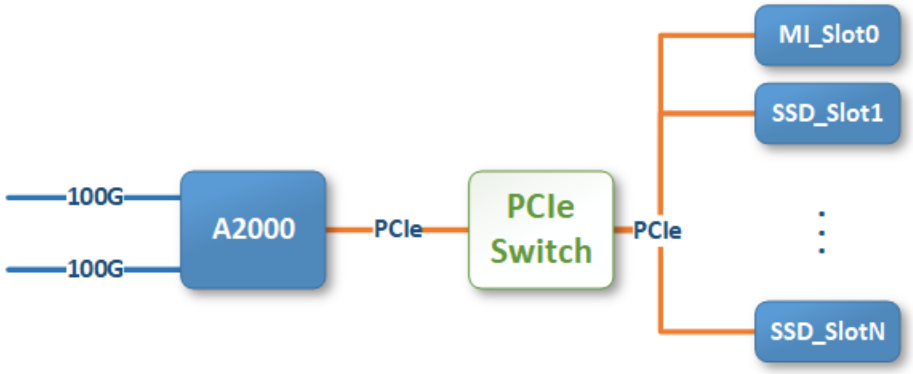


Figure 24: Target MI Device Logical View



Once discovered, the A2000 Target will create an NVMe Admin connection to the MI device. A host can then come in and create Virtual Admin connections to the MI device via the A2000 Target's 100G ports. The host can also create an I/O connection to the MI device via the A2000 Target (this model is similar to the model used for any SSD connected to the switch via PCIe).

Slot 0 is reserved within the A2000 Target for the MI device. Slots 1-n are reserved for downstream SSDs.

Among other uses, this MI device can be used for receiving MI Send/MI Receive commands. This allows for implementing functionality such as SCSI Enclosure Services (SES) within the MI device.

The implementation of the MI device itself is beyond the scope of this document. A host can create an Admin connection to slot 0 via the A2000 Target's IP address. It can then perform NVMe CLI commands to/from that MI device. The A2000 Target will pass these NVMe commands through from Ethernet to PCIe (i.e. from a Virtual Admin connection between the host and A2000 Target, to the physical Admin connection between the A2000 Target and the MI device).

9.3 Hot Plug Support

The A2000 Target supports managed and surprise hot plug events. The PCIe switch must also be capable of handling surprise removal/insertion of NVMe SSDs.

The primary difference between NVMe vs NVMe-oF from the Host perspective is that the host is NOT aware of the PCIe level activities during HOT plug operation. The A2000 Target handles the PCIe events.

After power up, the A2000 Target monitors the PCIe Root Port Link status. If link up is detected, the A2000 Target enumerates the PCIe bus/devices. The A2000 Target allocates resources for the available PCIe switch ports and the NVMe SSDs. It also pre-allocates resources for the SSDs not currently available.

PCIe hot plug is supported by monitoring PCIe Link events at the downstream ports. The A2000 Target uses a polling mechanism to detect the SSD insertion and removal events. This eliminates potential PCIe switch-dependent compatibility issues between chassis.

If a drive is removed while the host has I/O connections, the A2000 Target first sends an Asynchronous event with **Namespace Attribute Notice** and disconnects the host connections and releases/cleans up the resources.

If the host has established a persistent connection to the A2000 Target's Discovery Controller, ASYNC notifications will be sent to the host to indicate drive insertion/removal conditions. This will cause the host to automatically detect the change in the target configuration and present the change to upper-level applications.

Managed hot-swap events are well-orchestrated and the outstanding I/Os are stopped/completed prior to the time of the hot plug. It is not the focus of this document. These can be managed by specific vendor chassis management tools/ procedures. The A2000 Target provides a management interface to assist chassis manager for these operations.

It is recommended that only one hot plug operation is performed at a time.

Hot-Plug Assumptions

- A2000 Target is connected to a Chassis PCIe slot.
- Drives are hot-swap capable.
- Chassis HW and PCIe switches are hot-swap capable.
- Host and drive will handle data integrity.
- Hot plugging an entire PCIe switch (including bringing down/up upstream ports) is outside the scope of this document.

Hot-Plug Assumption Definitions

- **Hot-Plug**: Removing or inserting an NVMe SSD into a slot while the chassis is powered. This can be planned/managed or unplanned/surprise.
- **Hot-Swap**: Hot removal followed by a hot insertion of an NVMe SSD.
- **Managed Hot-Insertion**: Inserting an NVMe SSD to a powered chassis after applying the chassis' managed hot plug procedure.
- **Managed Hot-Removal**: Removing an NVMe SSD from powered chassis after stopping the IOs, removing the host(s) connections and applying the chassis' managed hot plug procedure.
- **Surprise Hot-Insertion**: Inserting an NVMe SSD to a powered chassis slot without applying the chassis' managed hot plug procedure.

- **Surprise Hot-Remove**: Removing an NVMe SSD from a powered chassis without applying the chassis' managed hot plug procedure. The drive may be in use by hosts at the time of removal.
- **Downstream Port Containment**: Downstream Port Containment, or DPC, is an optional capability for detecting uncorrectable errors/messages and disconnecting downstream ports to prevent the spread of data corruption.

9.3.1 Downstream Port Containment (DPC)

During PCIe enumeration, if the A2000 Target detects DPC, it configures the PCIe DPC Extended Capability registers.

During Hot Drive removal, the A2000 Target detects the DPC Trigger event from the DPC Status Register. The A2000 Target completes the drive removal processing and updates the DPC Status register to clear the event.

9.3.2 Hot Plug Scenarios

Adding a drive to an empty slot

A2000 Target will:

1. Detect the event.
2. Configure the drive PCIe registers and NVMe controller registers.
3. Identify the drive NVMe controller and Namespaces.
4. Configure the bridge HW with Namespace information (such as format of the namespace)
5. Allocate resources for the drive to be used by the host(s)
6. Trigger an MI Object Async Notification (0x014) to the BMC (via I2C or PCIe) indicating drive addition.

The `show-drive` CLI command can be used to see the drive information.

If integrated with A2000 Target, Administrators can use the Management tools—such as BMC—to configure the information necessary—such as Host Target Authorization Table—for hosts to access the drive.

Removing a drive from a slot with no host connection

A2000 Target will:

1. Detect the event.
2. Invalidate the drive and namespace resources.
3. Trigger an MI Object Async Notification (0x014) to the BMC (via I2C or PCIe) indicating drive removal.

Removing a drive from a slot while host has connections

A2000 Target will:

1. Detect the event.
2. Send an ASYNC event with **Namespace Attribute Notice** to host (Not necessary).
3. Send a DISCONNECT request on the Admin and I/O connections and free the resources.
4. Invalidate the drive and namespace resources.

The host will retry connecting to the drive. If the same drive is plugged back in to the same slot during the retry time, the host will re-establish the connections.

Removing a drive from a slot with host connected and running FIO

1. Same as section above, Removing a drive from a slot while host has connections
2. FIO stops and exits unless **continue_on_error=all** parameter is used. If this parameter is used, FIO will not stop. If the drive is hot inserted back, the host will re-establish the connections and IOs will continue.

Removing a drive from a slot and inserting a different drive to the same slot while Host has connections and/or IOs

This operation is strongly discouraged. It is likely for the Host to have information based on the old drive.

Note: The A2000 Target does NOT prevent the user from performing this operation.

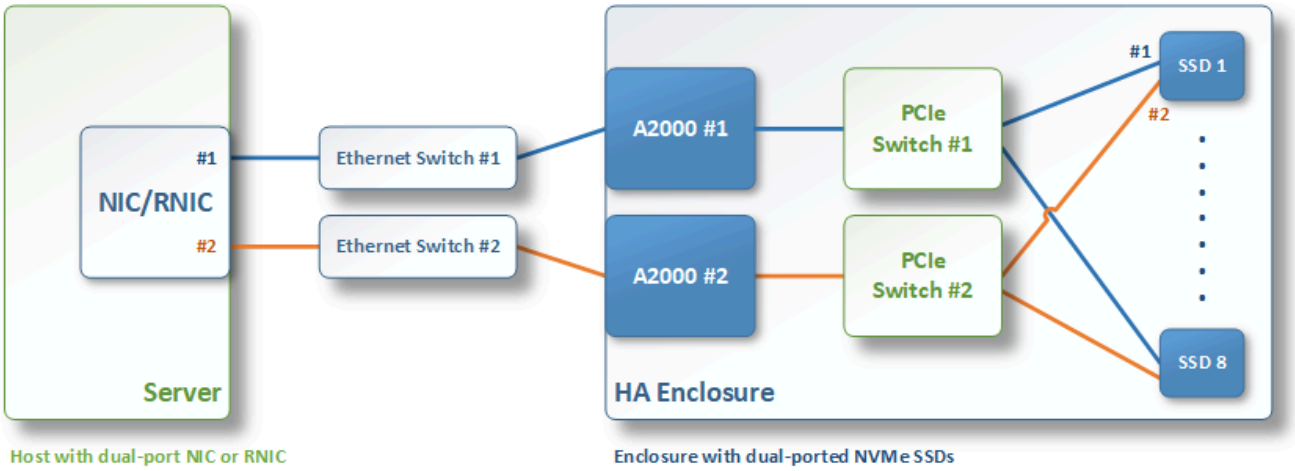
Hotplugging a PCIe switch

Caution: hot plugging of an entire PCIe switch is not supported.

9.4 High Availability Support

The figure below shows a typical configuration where the A2000 Target is deployed in a system with high availability, achieved by providing multiple paths from the host to the storage device. An important functionality of the A2000 Target in such an application is to provide unique Controller ID values for each connection established by the Host to the same end device via multiple paths.

Figure 25: System Configuration for High Availability



In this configuration, the Host establishes an NVMe-oF connection to SSD #1 using port #1 of the NIC/RNIC via Ethernet Switch #1, which is connected to A2000 Target #1. The second independent path to this same NVMe drive is enabled by establishing an NVMe-oF connection to SSD #1 using port #2 of the NIC/RNIC via Ethernet Switch #2, which is connected to A2000 Target #2. The requirement for multi-pathing to operate successfully is that the Controller IDs generated for these two connections across the two different A2000 Targets must be unique. The A2000 Target supports this requirement via the “Dual-Port” parameter Bridge Settings object. Refer to Bridge Settings *(page 104)* for details.

9.5 Locate LED

The A2000 Target provides the capability for a Host to issue a command that results in an LED of the specified drive to be lit, thereby having the Host be able to locate a specific drive. In order to make this functionality work, the following needs to occur:

1. Set Opcode: BMC needs to execute RLMP command to set the Vendor Unique Command opcode to be used for the Locate LED function.
2. Locate LED VUC: The host would execute a Locate LED Vendor Unique Command which uses the VUC above and specifies the drive effected and whether to enable/disable the LED.
3. Async Notification/Locate LED RLMP Object: Upon receiving the above Locate LED VUC, the A2000 Target will in turn cause an Async Event to occur and set the appropriate bit in the Async Notification RLMP object indicating Locate LED change. The BMC would then query the Locate LED RLMP opcode to determine which drive slots have the LED enabled/disabled.
4. Light Drive LED: BMC would then enable/disable the given drive LED.

9.5.1 Set Opcode

In order for the host to execute an `nvme admin-passthru` command to enable/disable the Locate LED, the BMC must first define the VUC opcode to use. The VUC Opcode RLMP Object allows the BMC to configure this opcode. Refer to VUC Opcode *(page 150)* for details. The BMC can set this opcode to be any value between `0xC0` and `0xFF`. If a value less than `0xC0` is used (which is the default), the Locate LED functionality will be disabled.

9.5.2 Locate LED VUC

The host must have a connection to a drive in the system. Over this connection, they have two options for how to enable/disable a given drive's Locate LED. Examples of each are shown below:

Option 1: Example where host wants to enable LED for drive on the current connection:

```
nvme admin-passthru /dev/nvme0n1 -o <locateledoc> --cdw10=1
```

Option 2: Example where host wants to disable LED for drive with nqn **nqn.wdc.com:nvme.1**. This can be executed from any drive connection:

```
echo nqn.wdc.com:nvme.1 | nvme admin-passthru /dev/nvme0n1 -o
<locateledoc> -w --cdw10=0 -l 256 -p 0x0
```

NVMe Admin-passthru Parameters	Functionality
Opcode (-o)	Determined by the VUC Opcode KLMP Object
DWord 10 (--cdw10)	0: Disable LED 1: Enable LED
-l	0/unused: No NQN specified 256: Length of NQN payload to pass in command.
-p	0x0: fill entire NQN payload (up to –l bytes) with 0x0. This is to make sure and zero-fill any bytes after the specified NQN

NVMe Admin-passthru Parameters	Functionality
echo **<NQN>**	Specify if A2000 Target needs to read the NQN from the host.

9.5.3 Async Notification/Locate LED RLMP Object

When the host updates any Locate LED field, the A2000 Target will cause an Async Notification to occur – see **Async Notification** *(page 122)*. The BMC can then read the Async Notification option to determine that a Locate LED field was updated.

The BMC will then read the Locate LED object to determine which drive LEDs should be enabled/disabled – see **Locate LED** *(page 152)*

9.5.4 Light Drive LED

Based on the LED mask from the Locate LED Object, the BMC will then enable/disable the appropriate drive LEDs (this is BMC dependent).

A2000 Initiator & A2000 Target Common Operation

In This Chapter:

10.1 Firmware Images Overview

The following table lists the images that are located in Flash for the A2000 Target C2xxx Device.

Table 211: A2000 Target C2xxx Device Flash Images

	Bridge
Bootloader Image	X
Recovery/(Golden) Firmware Image	X
Downloadable Firmware (Slot 6)	X
Downloadable Firmware (Slot 7)	X
Staging Firmware	X

Bootloader Image

This image is programmed in the factory only. It is the initial code booted when the A2000 Target comes up and determines which firmware image to load/run.

Recovery/Golden Firmware Image

This image is programmed in the factory only and can be booted if there is an issue booting any of the downloadable images.

Downloadable Firmware (Slot 6/7) Images

These firmware images are programmed in the factory and can be updated in the field. They allow for feature enhancements and fixes in the field for the firmware.

Staging Firmware Image

This location is used to temporarily hold the firmware image while it is downloaded. Once downloaded, the image can be activated, at which point the staging firmware area is freed up, and the image—which was previously downloaded—is assigned to a specific slot.

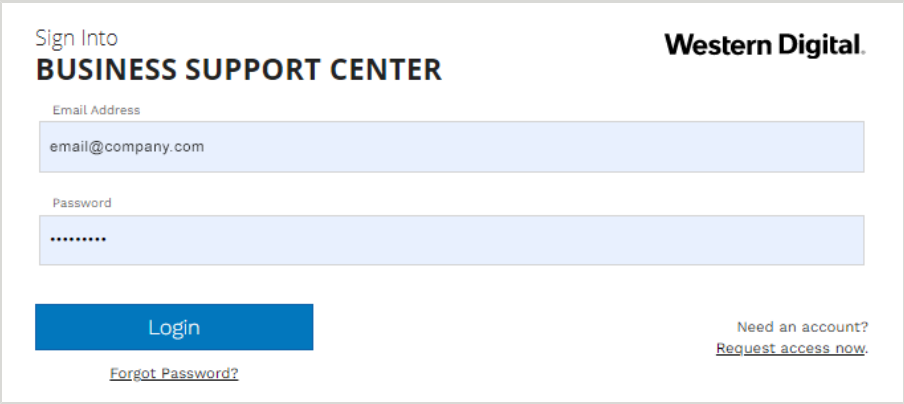
10.1.1 Downloading Firmware

This procedure provides instructions for downloading firmware for the RapidFlex A2000 Target from the Western Digital Business Support Center.

Step 1: Open a web browser and navigate to: https://portal.wdc.com/s/.

The login page for the **Western Digital Business Support Center** will be displayed:

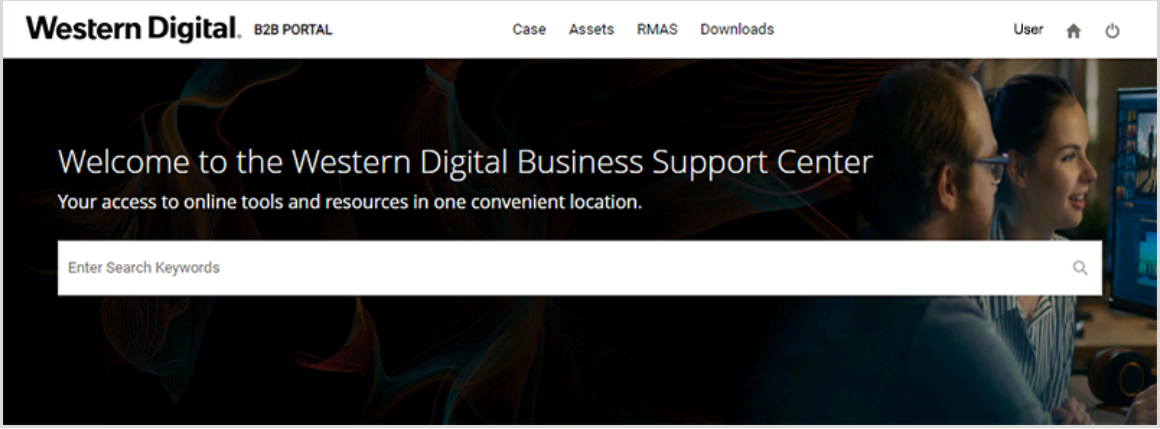
Figure 26: Western Digital Business Support Center Login Page



Step 2 : Enter a valid email address and password into the **Email Address** and **Password** fields. Then click the **Login** button.

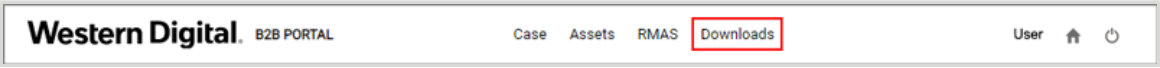
The **Western Digital B2B Portal** page will be displayed:

Figure 27: Western Digital B2B Portal



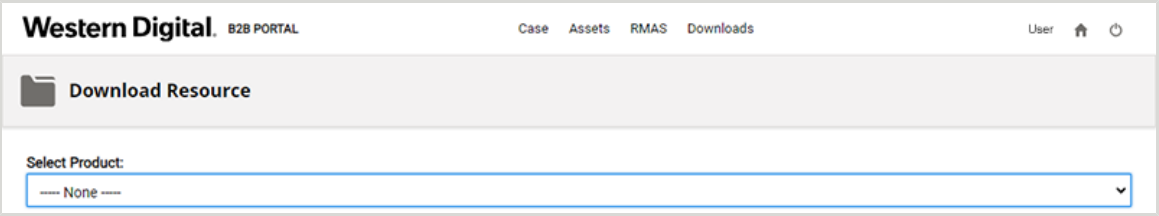
Step 3 : Click **Downloads** at the top of the page.

Figure 28: Downloads Link



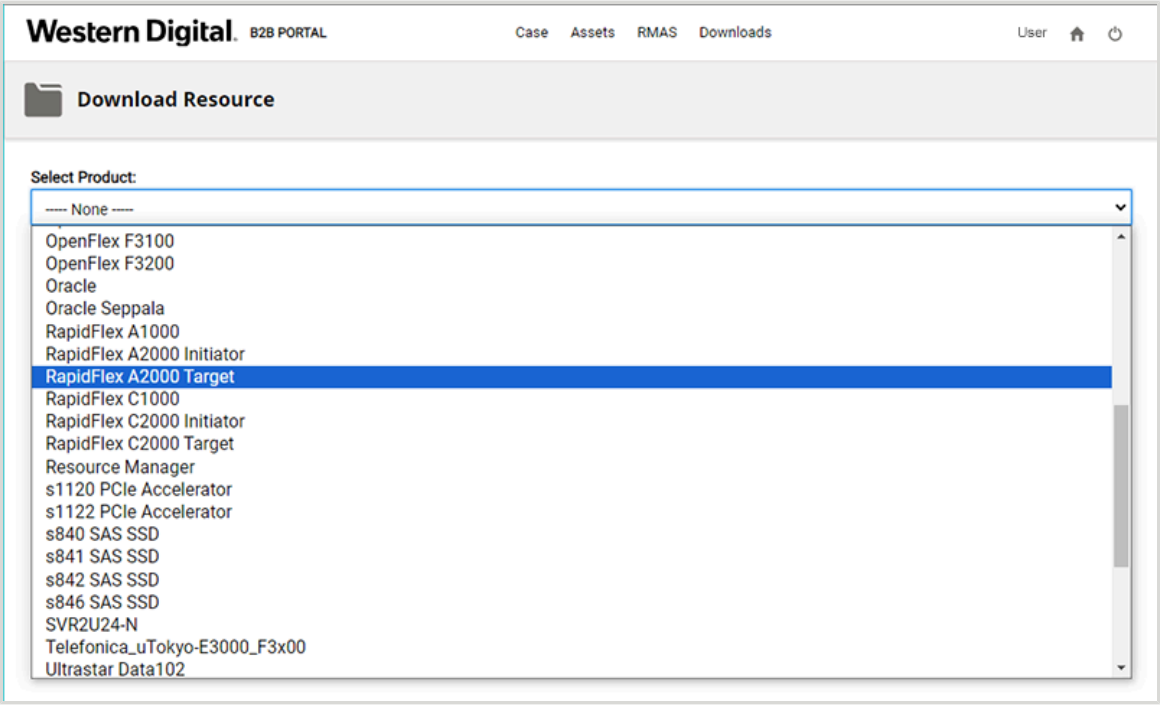
The **Download Resource** page will be displayed:

Figure 29: Download Resource Page



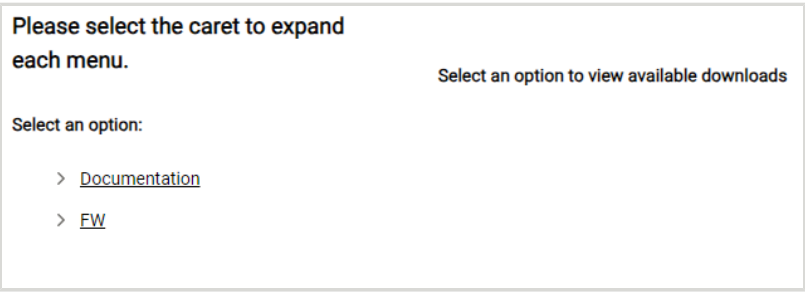
Step 4 : Use the **Select Product** drop-down list to choose the A2000 Target.

Figure 30: Select Product Menu



A **Select an option** menu will be displayed:

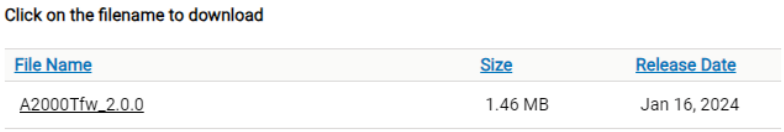
Figure 31: Select an Option Menu



Step 5 : Under **Select an option**, click the caret symbols to expand the menus for **FW** and **Current-<version>**. Then click the **Firmware** link.

A list of firmware files will be displayed:

Figure 32: Firmware Files



Step 6: From the available options, click to select the desired firmware file.
The firmware binary file will download to the host.

Result: The firmware file has now been downloaded from the Western Digital Business Support Center.

What to do next: Proceed to Updating Firmware *(page 212)*.

10.1.2 Updating Firmware

This procedure provides instructions for updating firmware on the RapidFlex A2000 device.

Before you begin:

Note: This procedure requires the use of NVMe-CLI version 1.16 or later.

Firmware Image Locations

There are two locations available in flash for holding new firmware images. These two locations are referred to by slot number. The first location is Slot 6 and the second location is Slot 7. These locations allow the user to maintain two firmware images in flash at all times. At any point, the user can specify which image to boot.

Step 1: To get the latest firmware for your RapidFlex A2000 device, log in to your Western Digital support account at https://portal.wdc.com/s/. See Downloading Firmware *(page 209)* for instructions.

Step 2: Ensure that the NVMe connection to the device is functioning correctly prior to starting the download.

Step 3: Use the `nvme fw-download` command to download the firmware image to a staging location in flash.

```
nvme fw-download /dev/nvme0 --fw=<path/filename.fw>
```

… where `<path/filename.fw>` is the full path to the firmware file on your system.

Note: The download operation can be performed on any connection to an active drive. While the `fw-download` command can also be used to update firmware on the drive, in this instance only the drive connection is being used. The RapidFlex device will look for a header signature in the first buffer payload that indicates the firmware file is for itself, not the drive.

When functioning correctly, firmware downloads usually complete within 10 seconds. After download, the image is held in the staging area of flash but has not yet been allocated to a given slot number.

Step 4 : Perform a cyclic redundancy check (CRC) to verify that the image in flash is valid:

a. Use the `nvme fw-commit` command to initiate a CRC on the firmware image and make it bootable. The `--slot` and `--action` options must be specified in the command. See Table 161: Actions & Slots *(page 213)* for a description of available actions and slots.

```
nvme fw-commit /dev/nvme0 --slot=7 --action=1
```

Table 212: Actions & Slots

Slot #	Commit Action	Action Taken
6 | 7	0	Image currently in the staging location replaces the image currently in the specified slot number if the CRC is valid. The image is not activated.
6 | 7	1	Image currently in staging location replaces the image currently in the specified slot number and is activated at the next reset.
6 | 7	2	The image in the specified slot number is activated at the next reset (this action is executed without downloading).

b. Use the `nvme get-feature` command to check the status of the CRC until the `Current value` returned is `0x00020` (Good CRC).

```
nvme get-feature /dev/nvme0 -f 0xd0
```

The output of this command could be any of the following options:

- Good CRC: `get-feature: 0xd0 (Unknown), Current value: 0x000020`
- In Progress: `get-feature: 0xd0 (Unknown), Current value: 0x000021`
- Internal Error: `get-feature: 0xd0 (Unknown), Current value: 0x000022`
- Bad CRC: `get-feature: 0xd0 (Unknown), Current value: 0x000023`
- No `fw-commit` has been processed on the image in the staging area: `get-feature: 0xd0 (Unknown), Current value: 0x000000`

Caution: Although the firmware is now bootable, it is still not running. Proceed to step 5 *(page 213)* to run the firmware.

Step 5 : To run the new firmware, reboot the A2000 device either by power-cycling it or issuing the following UART CLI command to its Bridge Control management object.

```
set-control 0 0 0 1
```

Result: The new firmware is now running on the A2000 device.

10.1.3 Pulling Core Dumps

For support purposes, the A2000 Initiator and A2000 Target have the ability to produce core dumps, which may provide detailed information about internal system events. There are up to three core dumps stored persistently in Flash. Each core dump image is 15M in size (0xF0_0000). When working with the Western Digital Support team, you may be asked to pull one of these core dumps.

Core Dump Scripts, A2000 Initiator with A2000 Target

The following bash scripts can be run from the host—in conjunction with the NVMe-CLI—to pull either an A2000 Initiator or A2000 Target core dump. These scripts are intended for deployments that use an A2000 Initiator in conjuction with an A2000 Target.

Note: Core dumps can also be pulled using the `readcoredumps` command of the RapidFlexCLI utility. See the *RapidFlexCLI User Guide* for more information and instructions.

Download Target Core Dump:

```
#!/bin/bash
echo Uploading coredump 0
if [ -f "tgt_log_page_d1.bin" ]
then
 rm tgt_log_page_d1.bin
fi
COUNTER=0
OFFSET=0
while [ $COUNTER -lt 3840 ]
do
 let OFFSET=COUNTER*4096
 nvme admin-passthru /dev/nvme2n2 --namespace-id=$OFFSET -o 0xD2 -r -l 0x1000 -
cdw10=0x03FF00D1 --cdw2=0x02800001 --cdw3=0x3FF --raw-binary >> tgt_log_page_d1.bin
 let COUNTER=COUNTER+1
done
echo Uploading coredump 0 completed
```

Note: The `cdw2` value in this example represents the physical target/SSD ID. The user should change this value based on their specific configuration.

Note: This example pulls a core dump from slot 0. To change the slot number, use the following `cdw10` values:

- **Slot 0**: `0x03FF00D1`
- **Slot 1**: `0x03FF00D2`
- **Slot 2**: `0x03FF00D3`

Download Initiator Core Dump:

```
#!/bin/bash
echo Uploading coredump 0
if [ -f "log_page_c0.bin" ]; then rm log_page_c0.bin; fi
COUNTER=0
OFFSET=0
while [ $COUNTER -lt 3840 ]; do
let OFFSET=COUNTER*4096
nvme get-log /dev/nvme0n1 --namespace-id=$OFFSET -log-id=0xD1 --log-len=0x1000 --raw-
binary >> log_page_c0.bin
```

```
let COUNTER=COUNTER+1
done
echo Uploading coredump 0 completed
```

Note: This example pulls a core dump from slot 0. To change the slot number, use the following `log-id` values:

- **Slot 0**: `0xD1`
- **Slot 1**: `0xD2`
- **Slot 2**: `0xD3`

Core Dump Script, Other Initiators with A2000 Target

The following bash script can be run from the host—in conjunction with the NVMe-CLI—to pull an A2000 Target core dump. This script is intended for a deployment that uses another manufacturer's initiator in conjuction with an A2000 Target.

Download Target Core Dump:

```
#!/bin/bash
echo Uploading coredump 0
if [ -f "log_page_c0.bin" ]; then rm log_page_c0.bin; fi
COUNTER=0
OFFSET=0
while [ $COUNTER -lt 3840 ]; do
let OFFSET=COUNTER*4096
nvme get-log /dev/nvme0n1 --namespace-id=$OFFSET -log-id=0xD1 --log-len=0x1000 --raw-
binary >> log_page_c0.bin
let COUNTER=COUNTER+1
done
echo Uploading coredump 0 completed
```

Note: This example pulls a core dump from slot 0. To change the slot number, use the following `log-id` values:

- **Slot 0**: `0xD1`
- **Slot 1**: `0xD2`
- **Slot 2**: `0xD3`

10.2 Programming

Console Command Overview

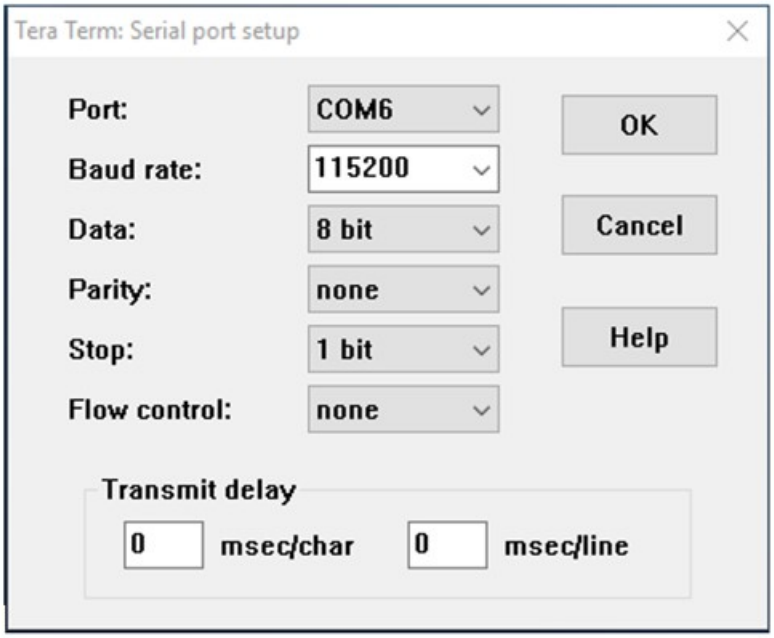
The A2000 Target provides a serial console port which can be used to interact with the MPU.

When in use, operational information will be displayed on the serial console. The serial console can also be used for configuration and displaying information.

Console Port Configuration

The serial port is configured to operate at 115200bps. An appropriate driver (e.g. Silicon Labs Dual CP210x USB to UART Bridge driver) needs to be installed on the system being used to gain console access to the A2000 Target. The console port settings should be set to match the following figure:

Figure 33: Required Console Port Settings



Note: The COM port number will be specific to the user's system.

Console Command Usage

CLI commands can be executed from the console to query and set various attributes and objects within the bridge. See Management Objects *(page 64)* for more information on specific management objects and CLI syntax.

The **?** command displays a list of all available commands.

Console Prompt

The console prompt consists of a letter to designate the product type (i=initiator, t=target), followed by the Manufacturer name from EEPROM, followed by a >. For example:

```
t-wdc>
```

Console Output

The A2000 Initiator outputs information related to both logging and CLI commands. The following information is directed out the UART:

- CLI command output
- Logging information associated with the ASIC's initialization sequence

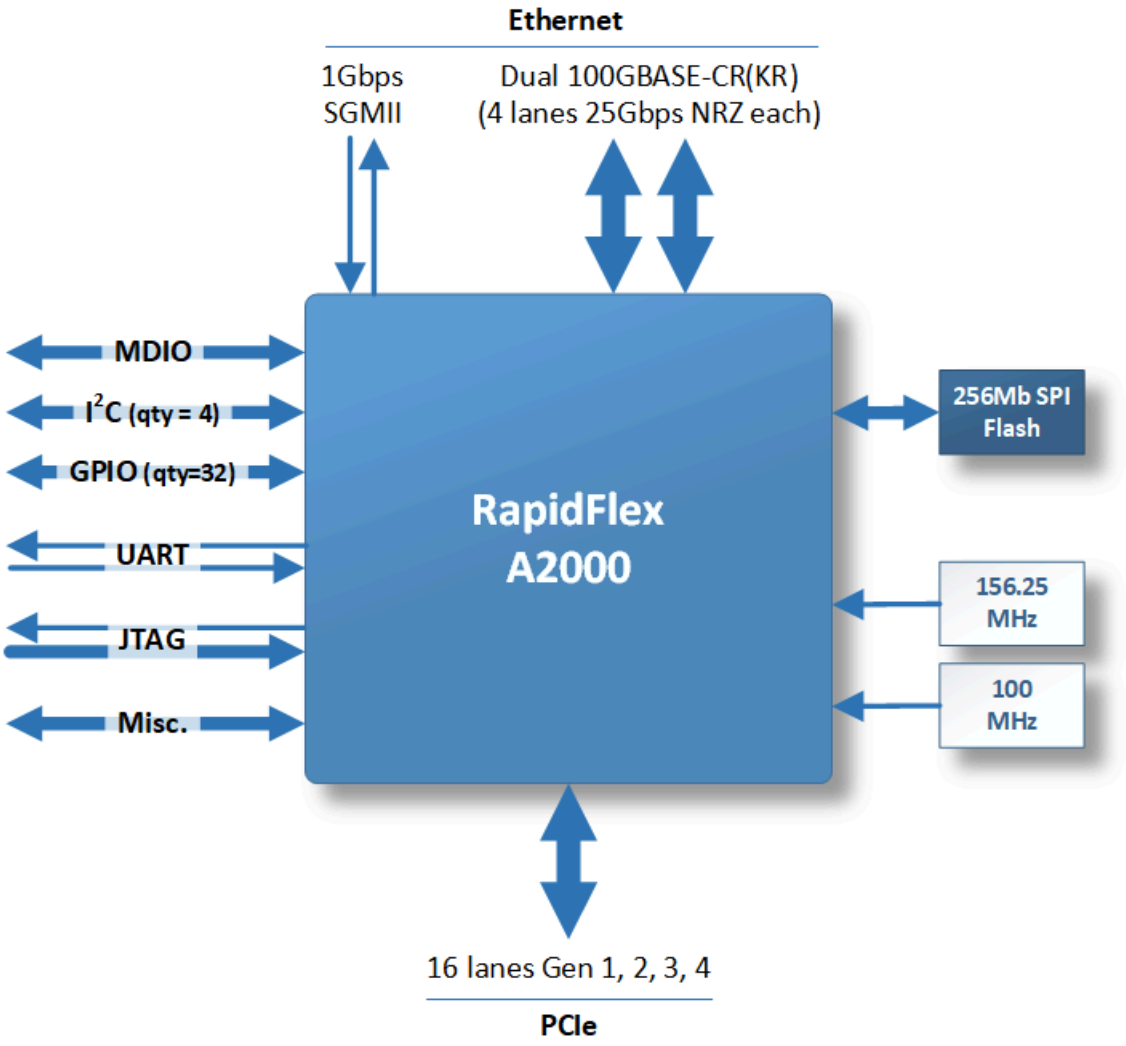
Once up and running, only the CLI command output will be sent to the UART; all other log information is sent to an internal memory buffer.

Electrical Characteristics

In This Chapter:

11.1 A2000 Electrical Interfaces

Figure 34: A2000 Target Electrical Interfaces



11.2 A2000 Operating Conditions

11.2.1 Absolute Maximum Ratings

Parameter	Description	Value	Units
VDD, VDDE, SGMII_VP, PCIE_VP, AVDD_PLL_575, AVDD_PLL_800	Core supply, SERDES Digital voltage range	-0.5 to 0.88	V
PCIE_VPH, VAA, AVDD_PVT, TS_AVDD, AVDD_PLL, ETH_VAA, ETH_VAA_AGCPLL	Analog Mixed Signal Supply voltage range	-0.5 to 1.98	V
VDDIO18	I/O supply voltage range	-0.5 to 1.98	V
VDDIO33	I/O supply voltage range	-0.5 to 3.63	V
VPAD (single ended IO)	Voltage at IO Pin (ball)	-0.5 to (VDDIO + 0.5)	V
TJ	Junction operating temperature range	-40 to 125	°C

11.2.2 Recommended Operating Conditions

Parameter	Description	Min	Nom	Max	Units
VDD, VDDE, SGMII_VP, PCIE_VP, AVDD_PLL_575, AVDD_PLL_800	Core supply, SERDES Digital voltage range	0.772	0.8	0.828	V
PCIE_VPH, VAA, AVDD_PVT, TS_AVDD, AVDD_PLL, ETH_VAA, ETH_VAA_AGCPLL, SGMII_VPH	Analog Mixed Signal Supply voltage range	1.71	1.8	1.84	V
VDDIO18	I/O supply voltage	1.62	1.8	1.89	V
VDDIO33	I/O supply voltage	2.97	3.3	3.46	V
TJ	Junction temperature	0	25	+110	°C
VPAD (single ended IO)	Voltage at IO Pin (ball)	-0.3		VDDIO + 0.3	V
VIH	High-level input voltage at IO pin	0.65 * VDDIO		VDDIO + 0.3	V
VIL	Low-level input voltage at IO pin	VSS - 0.3		0.35 * VDDIO	V

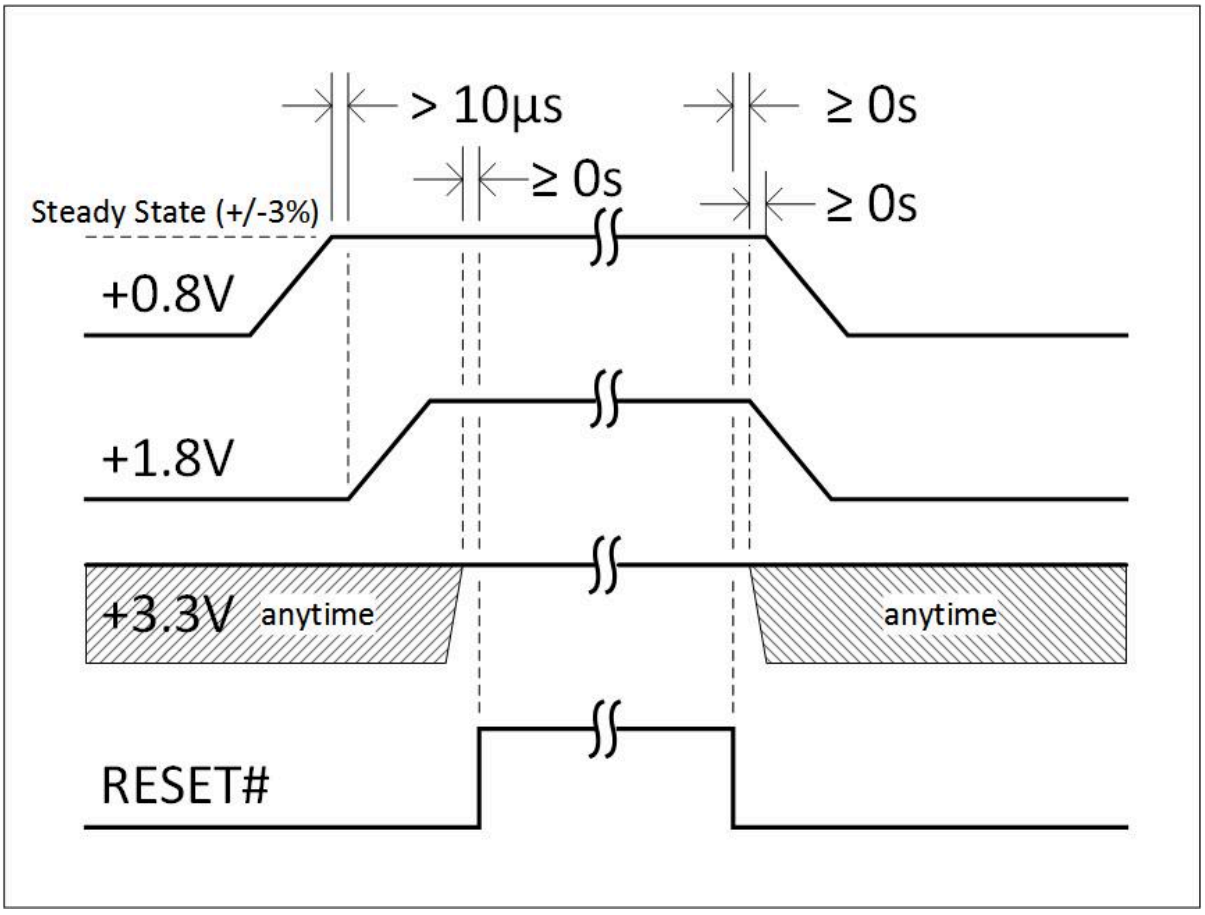
Parameter	Description	Min	Nom	Max	Units
VHYS	Input Hysteresis Voltage on all single ended inputs		0.1 * VDVDD		V

11.2.3 Power Supply Sequencing Requirements

Sequence Requirements

At power-up, the 0.8V supplies (VDD, VDDE, SGMII_VP, PCIE_VP) should establish their steady state value plus 10 uS before the 1.8V supplies begin to ramp. The 3.3 volt supply has no sequence requirements. The RESET# signal pin on the A2000 Target should not be deasserted until all the supplies have reached their steady state value.

At power-down, the 0.8V supplies (VDD, VDDE, SGMII_VP, PCIE_VP) should fall no sooner than the 1.8V supplies. These supplies can fall in phase. There are no requirements on when the 3.3-volt supply falls. The reset signal should assert before the supplies fall out of tolerance.



Power-Up Ramp Times

The minimum supply ramp up times (20%-80%) should be greater than 100 µs and less than 10mSec. Falling power supply ramp times should be greater than 100 µs.

11.2.4 Supply AC Requirements and Noise Sensitivity

The following AC noise requirements are part of the overall DC supply budget. They are in Volts peak-to-peak, and therefore half of the value should be subtracted from the + or – range when considering other sources for power supply accuracy.

Table 215: A2000 Target Power Noise Sensitivity

Supply Name	Value	Unit
VDD	26	mVpp
VDDE	20	mVpp
VAA	15	mVpp
AVDD_PVT	20	mVpp
AVDD_PLL	15	mVpp
TS_AVDD	7	mVpp
VP	20	mVpp
VPH	25	mVpp
VDDIO18	60	mVpp
VDDIO33	60	mVpp

11.3 A2000 Power Dissipation Estimates

Total Power Summary

The values in the following table are calculated estimates based upon characterized process and specialized IP (e.g. Memory, SERDES, IO, standard cell libraries).

Table 216: A2000 Target Total Power Dissipation Estimates

	Junction Temp = 25°C	Junction Temp = 90°C	Junction Temp = 110°C
Power Supply State	Nominal	+2.0%	Max
Estimated Power	7.3W	10.0W	11.8W

Detailed Maximum Power Requirements

The following power estimates represent the expected conditions with a device junction temperature of Tj= 110 C°, and the maximum voltage for each supply as noted in the table below.

Supply Name	Nom	Max	mW (max)	mA (max)
VDD	0.8V	0.84V	6200	7750
VDDE	0.8V	0.84V	908	1080
PCIE_VP	0.8V	0.84V	1596	1900
SGMII_VP	0.8V	0.84V	100	119

Supply Name	Nom	Max	mW (max)	mA (max)
AVDD_PLL_585	0.8V	0.84V	2	2.4
AVDD_PLL_800	0.8V	0.84V	2	2.4
PCIE_VPH	1.8V	1.89V	1307	692
SGMII_VPH	1.8V	1.89V	42.4	22.4
ETH_VAA	1.8V	1.89V	1160	614
ETH_VAA_AGCPLL	1.8V	1.89V	329	174
TS_AVDD	1.8V	1.89V	2	1
AVDD_PVT	1.8V	1.89V	2	1
VDE18*	1.8V	1.89V	76	40
VDE33*	3.3V	3.465V	70	20
Total			**11,796.4**	

* Actual power and currents are dependent upon the capacitive and DC loads on the PC board for all single-ended signals.

11.4 A2000 Ethernet Ports

11.4.1 Ethernet Features Overview

- Eight lanes of 2.5Gbps NRZ transceivers
- Insertion Loss handling greater than 35bB at Nyquist frequency
- Advanced Equalization
  - Receiver 3-tap DFE, and CTLE
  - Transmit pre-emphasis (5-taps, 2-pre, 1 center, 2-post)
- On-die Rx blocking capacitors
- Auto-negotiation
  - IEEE 802.3ap clause 72
  - IEEE 802.3bj clause 92/93
  - 25G Ethernet Consortium Specification
- Random Jitter <200 fs
- Built-in Self test features
  - RX Eye Monitor (non-destructive)
  - Pattern Generation/Checking (PRBS9, PRBS15, PRBS23, PRBS31, 32 UI customer defined)
  - Internal digital Rx to Tx
  - Internal analog Tx to Rx loopback
- Scan DFT
- Automatic Termination Resistance Tuning

11.4.2 Ethernet Transmitter Parameters

Parameters	Min	Typ	Max	Units
Differential Output pk-pk voltage	200		1000	mV (pk-pk)
Differential Return Loss	TBD			
Common Mode Return Loss	TBD			
DC output impedance termination	80	100	120	ohm
Total Jitter (TJ)	0.33	UI		
Random Jitter (RJ)			<200	fs (rms)
Deterministic Jitter (DJ)	TBD			

11.4.3 Ethernet Receiver

Ethernet Receiver Equalization and Adaption

The incoming signal passes through a configuarable analog CTLE (continuous time linear equalizer). The remaining ISI at the CTLE output is removed by a DFE equalizer. The DFE tap settings are constantly optimized by checking the eye opening of the equalized channel output.

Receiver Parameters

Parameters	Min	Typ	Max	Units
Differential Input pk-pk voltage	50		1200	mV (pk-pk)
Differential input common mode voltage	0.2		0.8	V
Differential Return Loss	TBD			
Common Mode Return Loss	TBD			
DC input impedance termination	80	100	120	ohm
Frequency offset tolerance	-100	0	100	ppm

11.4.4 Ethernet Ports ESD Rating

Ethernet SerDes I/O pins are designed to tolerate the ESD voltages shown in the following table.

Table 220: A2000 Target Ethernet ESD Ratings

Parameters	Tolerance	Units
HBM	1,000	V
CDM	250	V
MM	100	V

11.5 A2000 PCIe Ports

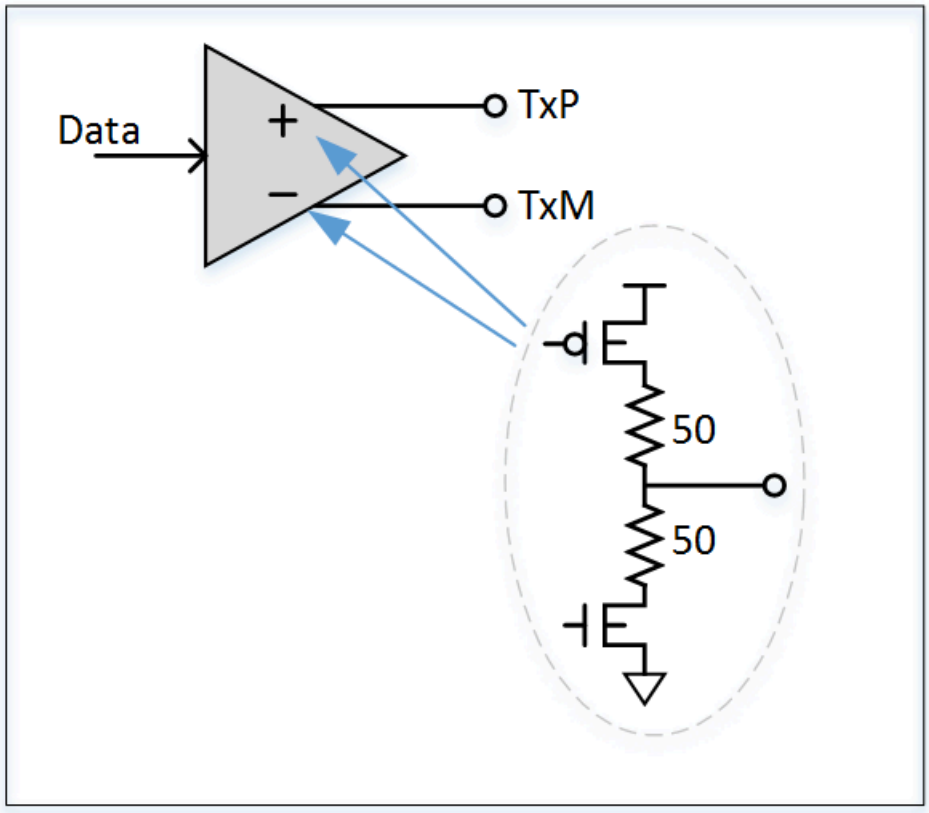
11.5.1 PCIe Features Overview

- Single x16 lanes of NRZ 2.5 Gbps, 5 Gbps, 8 Gbps, 16 Gbps PCIe 4.0
- Adaptive and configurable RX continuous time linear equalizer (CTLE), and 5-tap decision feedback equalizer (DFE)
- Adaptive and programmable 3-tap FFE TX equalization
- RX Eye Monitor (non-destructive)
- PRBS Generation/Verification (7, 9, 11, 15, 16, 23, 31 and user defined)
- Programmable 10-bit pattern generation with error injection capabiltiy
- Serial internal analog loopback (TX-to-RX) and Digital Reverse loop back (TX-toRX)
- IEEE 1149.1 and 1149.6 (ACJTAG)
- Termination Resistance Tuning (external reference resistor)
- Separate REFCLK, with or without independent SSC (SRIS)

11.5.2 PCIe Transmitter

PCIe Transmitter Termination

The Tx is a voltage mode driver with 50 ohm termination on both TxP and TxM pullup and pulldowns. This results in a 100 ohm differential source impedance. The termination is tuned by the PHY using an off-chip reference resistor.



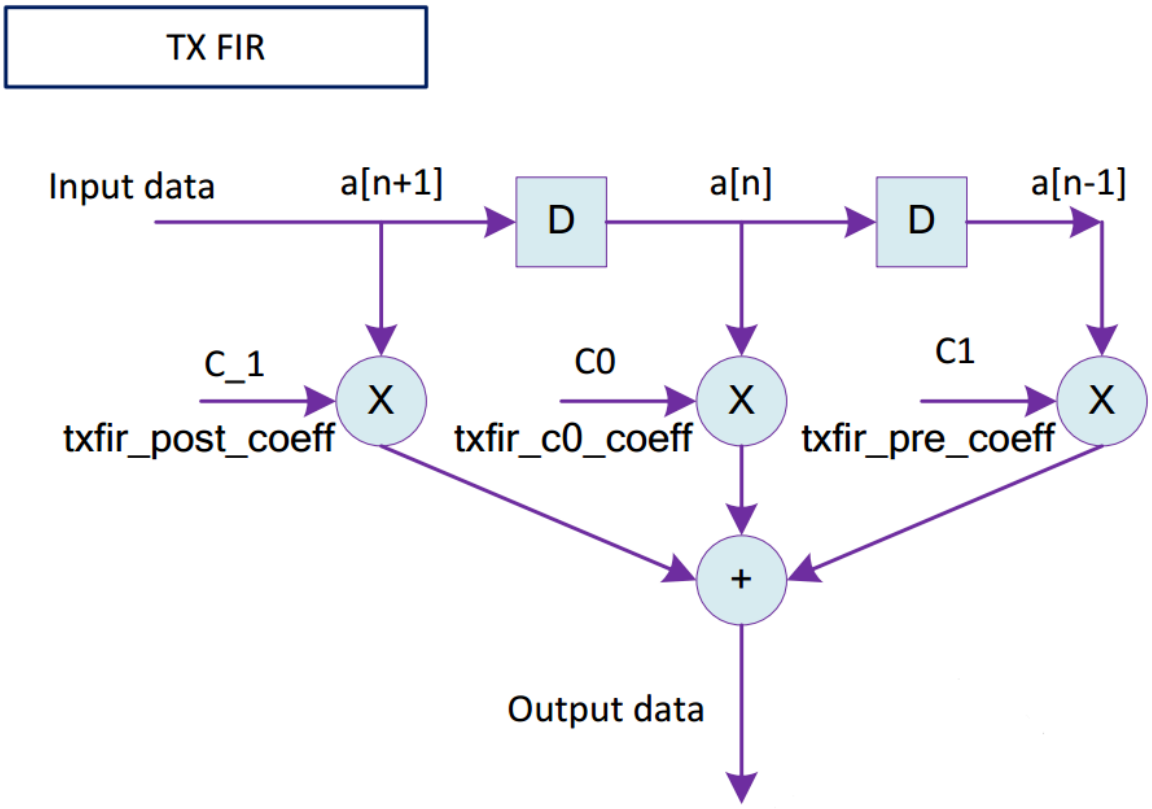
PCIe Transmitter Parameters

Parameters	Min	Typ	Max	Units
Differential Output pk-pk voltage (C0)	909	1000	mV (pk-pk)	
Differential Return Loss	TBD	Pending characterization report		
Common Mode Return Loss	TBD	Pending characterization report		
DC output impedance termination	90	100	110	ohm
Total Jitter (TJ)	TBD	UI		
Random Jitter (RJ)	TBD	fs (rms)		
Deterministic Jitter (DJ)	TBD			

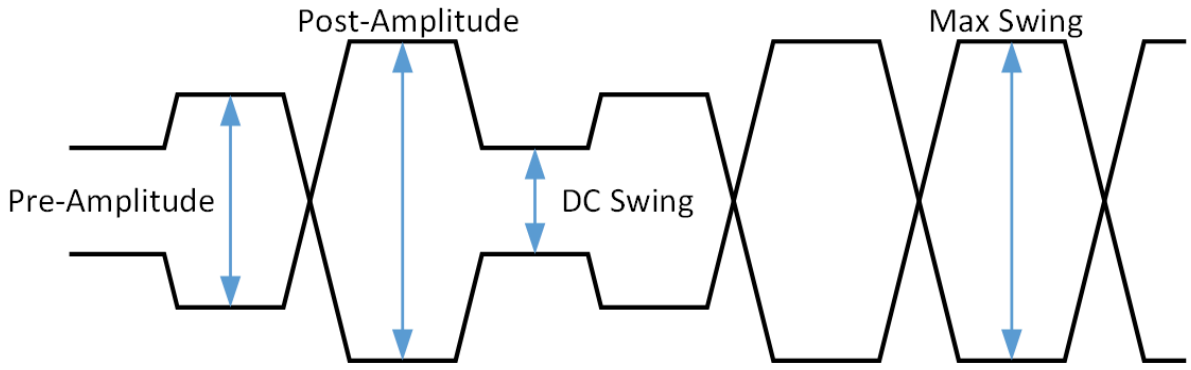
Transmitter Equalization

Transmit Equalization utilizes a 3-tap feed-forward equalization (FFE), with 1 precursor tap and 1 post-cursor tap.

Figure 37: Transmitter Equalization



C_0, C_{-1}, C_1 are Tx driver main tap, pre-cursor tap, and post-cursor tap coefficients
C_{-1} and C_1 are defined as negative values



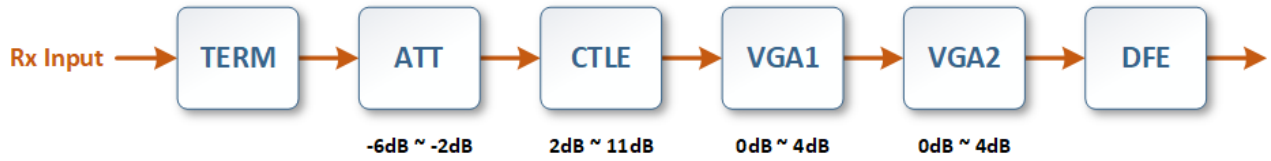
Pre-amplitude = Vswing * (C_0 +|C_{-1}| - |C_1|)/40
Post-amplitude = Vswing * (C_0 -|C_{-1}| + |C_1|)/40
DC_swing = Vswing * (C_0 -|C_{-1}| - |C_1|)/40
MAX_swing = Vswing * (C_0 +|C_{-1}| + |C_1|)/40

11.5.3 PCIe Receiver

11.5.3.1 Receiver Analog Front End

The analog front end (AFE) circuitry receives off-chip data and automatically optimizes link performance by adjusting the equalization parameters.

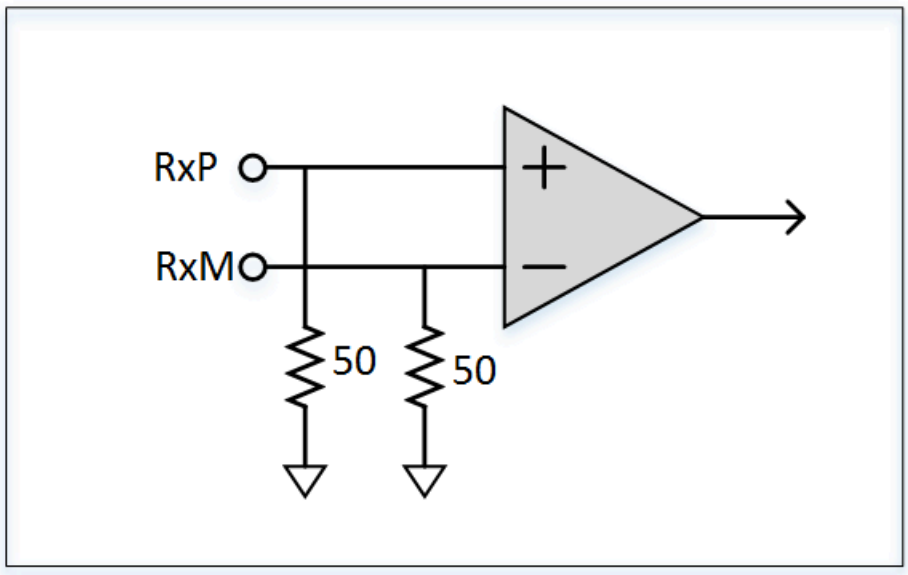
Figure 39: Rx Analog Front End Diagram



11.5.3.2 Rx Termination

Termination (TERM) – The TERM block is tuned by the PHY using an off-chip reference resistor. The termination is a single ended 50 ohm termination to ground on each of RXP and RXM.

Figure 40: Rx Termination



11.5.3.3 Attenuator (ATT)

The ATT block is responsible for attenuating the incoming signal to acceptable levels for the subsequent CTLE block, maintaining linearity in case of short reach links.

11.5.3.4 Continuous Time Linear Equalizer (CTLE)

The CTLE block provides high frequency boost range from 2dB to 11dB. The CTLE block has programmable boost levels, boost pole frequency, and boost bandwidth.

11.5.3.5 Voltage Gain Amplifiers (VGA)

The VGAs are implemented to maintain a constant signal swing.

11.5.3.6 Rx DFE (Decision Feedback Equalizer)

The post VGA output is summed by a 5-tap, continuously adaptive DFE. The output of the DFE is sampled by offset-compensated samplers, deserialized, and sent to the core along with the recovered clock.

11.5.3.7 Rx Internal Eye Monitor

Provided with the PHY is an internal eye monitor, which can provide data to produce a plot of time and voltage axis. The eye monitor is non-destructive and can be run in parallel with live traffic. This procedure can be used to characterize the input Rx eye margin as seen after the ATT and DFE summer.

11.5.4 PCIe Ports ESD Rating

PCIe SerDes I/O pins are designed to tolerate the ESD voltages shown in the following table.

Table 222: PCIe Port ESD Ratings

Parameter	Tolerance	Units	Characteristics
HBM	2,000	V	High voltage, low current Very long duration
CDM	6 A peak discharge	A	High voltage, high current Short duration
MM	100	V	Low voltage, medium current Long duration, bidirectional

11.6 A2000 External Reference Clock Requirements

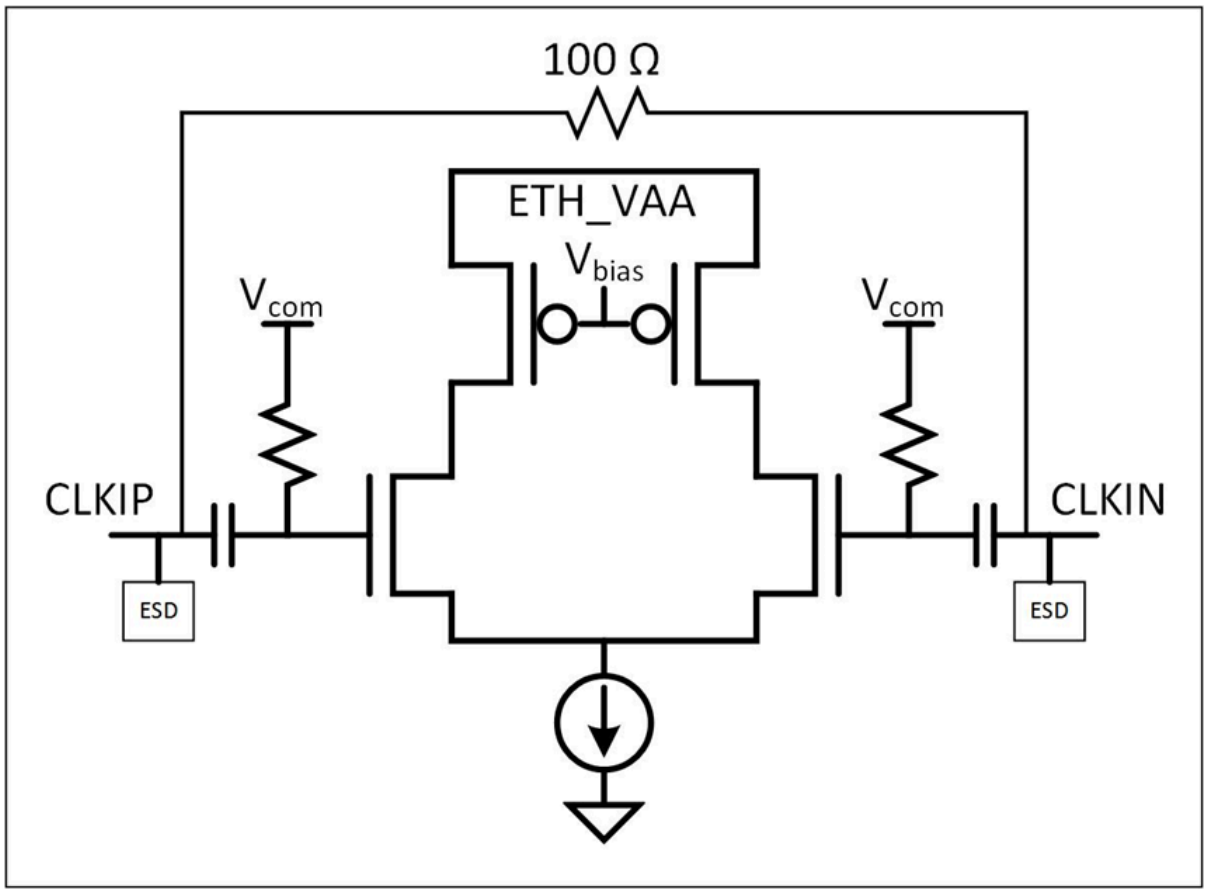
11.6.1 External 156.25MHz REFCLK Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
CLKDIV	CLK Input differential swing		600		1600	mVp-pd
CLKICM	CLK Input common mode		400		1300	mV
CKIPSEV	CLKIP Input voltage range		VSS+ 100mV		VAA-100mV	mV
CKISEV	CLKIN Input voltage range		VSS+ 100mV		VAA-100mV	mV

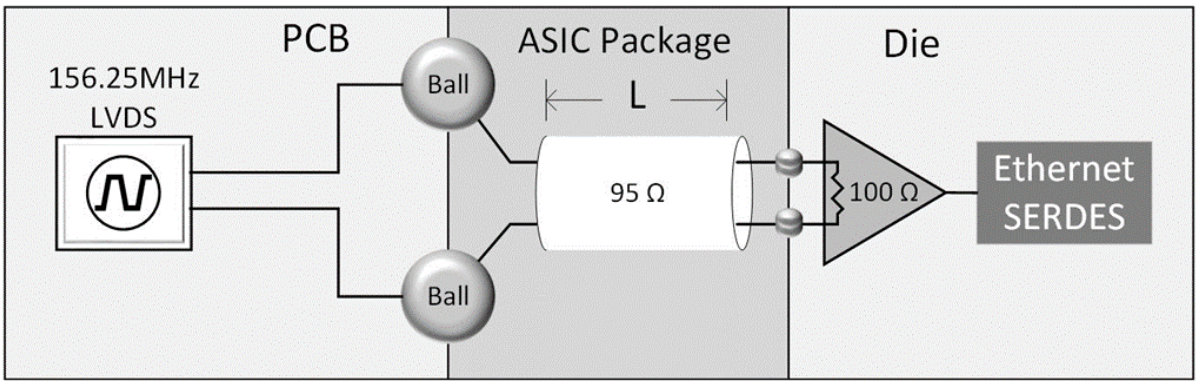
Symbol	Parameter	Conditions	Min	Typ	Max	Units
CLKTR	CLK Input differential transition time	-100mV to +100mV			120	pS
CLKIDC	CLK input duty cycle		45		55	%
CLKIRF	CLK Input Rise & Fall time	20% to 80%			1.0	nS
CLKIF	CLK Input Frequency	+/-50ppm		156.25		MHz
CLKPNM	Phase Noise Mask	@10KHz			-130	dBc/Hz
		@100KHz			-138	
		@1MHz			-147	
		@10MHz			-150	
CLKOFFSET	Reference Clock Accuracy		-50		+50	ppm
CLKDRES	Input differential resistance		70	100	130	ohm

11.6.2 156.25MHz REFCLK Receiver

The A2000 Target's differential receiver for the 156.25MHz clock includes an internal 100 Ohm termination resistor. This termination resistor has no internal common mode bias, and therefore the DC voltage of the supplied clock signal needs to be controlled by circuits external to the A2000 Target device. This DC bias requirement is not to establish the optimum operating point for the internal receiver but is meant to avoid forward bias in the internal ESD networks. Two possible methods for connecting an external clock source are, supply a DC coupled LVDS compliant signal, or alternatively provide an AC coupled clock signal with an external network to bias the common mode voltage halfwaybetween ETH_VAA (1.8V) and Ground. For example: ETH_VAA/2 or ~ 0.9v. These two options are shown in the following figures.

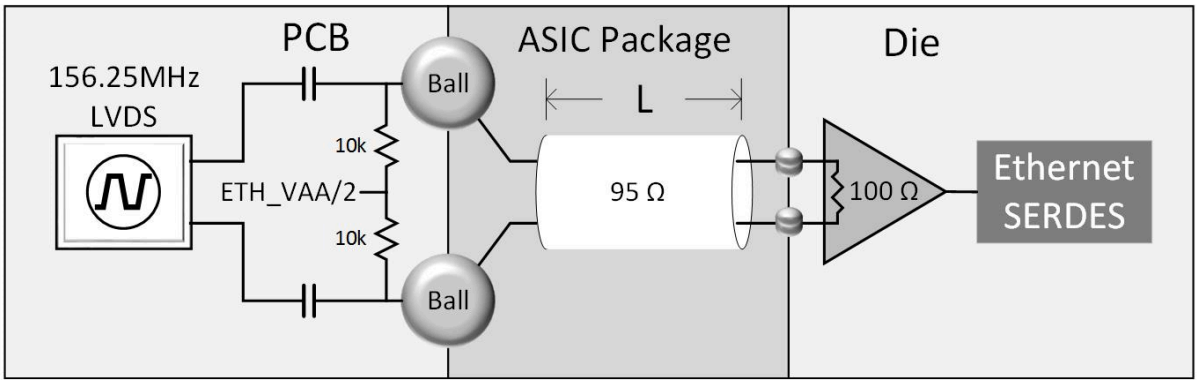


Recommended External Clock Circuit (DC Coupled)



Note: L = 14mm

Recommended External Clock Circuit (AC Coupled)



Note: L=14mm

11.6.3 100 MHz REFCLK External Clock Characteristics

External Reference Clock Requirements (PCIe – 100MHz)

In general, the external REFCLK requirements for the 100MHz clock source should meet the standards defined in the PCISIG for the data rates expected in the application of the A2000 Target.

Supported Standards	Reference Clock Jitter and Swing requirements
PCIe 1.1 - 2.5 Gbps	Refer to the PCI Express 3.0 Card Electromechanical specification
PCIe 2.1 - 5.0 Gbps	Refer to the PCI Express 3.1 base specification
PCIe 3.1 - 8.0Gbps	Refer to the PCI Express 3.1 base specification
PCIe 4.0 - 16.0Gbps	Refer to the PCI Express 4.0 base specification

Table 225: PCIe Reference Clock Requirements

Symbol	Parameter	Conditions	Min	Typ	Max	Units
REFCLK_F	Input Frequency from external Pins			100		MHz
FREF_OFFSET	Reference clock frequency offset		-300		+300	ppm
DCREF_CLK	Duty Cycle		40		60	%
VCMREF_CLK	Common Mode input level	Differential inputs	0		VP	V
VDREF_CLK	Differential Swing	Differential input must meet both VDREF_CLK and VDSEREF_CLK	300			mVppd
VDSEREF_CLK	Differential clock single ended voltage	Differential input must meet both VDREF_CLK and VDSEREF_CLK	-0.3		VP+0.3	V
SKEWDIF_CLK	Differential input skew				20%	%

Symbol	Parameter	Conditions	Min	Typ	Max	Units
SWREF_CLK	Differential input slew rate	20% - 80%	0.6		4	V/ns

External Reference Pad Input Load

Figure 44: Model of Reference Clock Pad Input Load

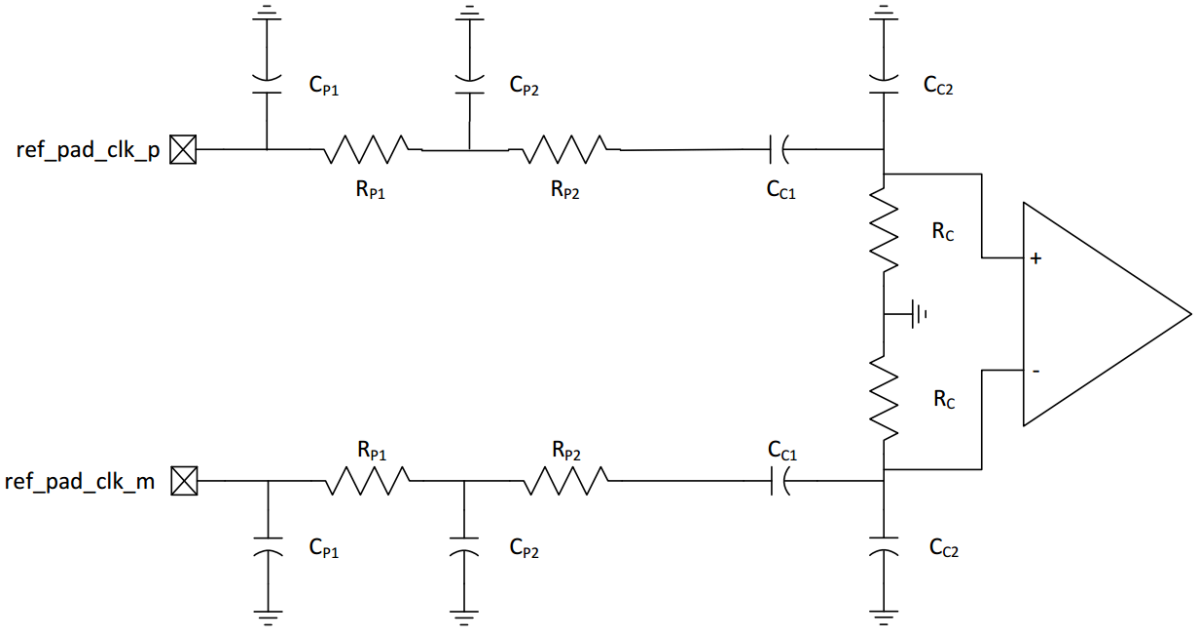
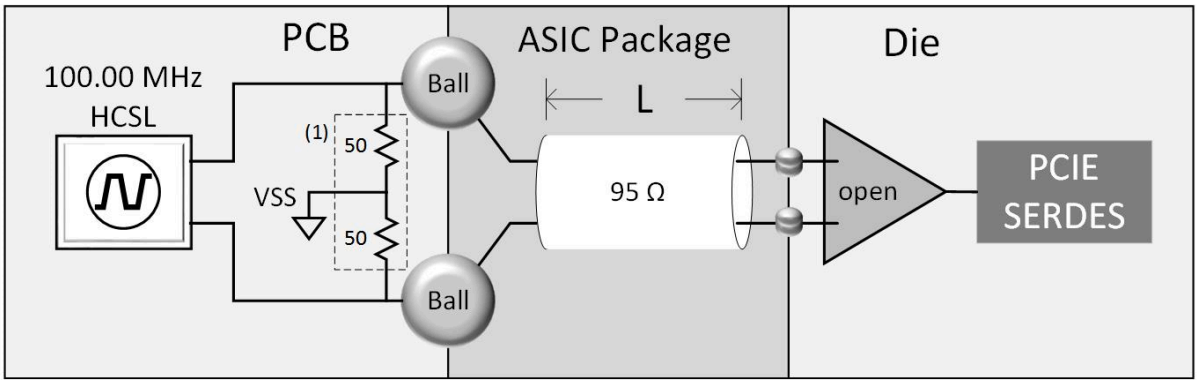


Table 226: PCIe Reference Clock Input Parasitic

Parasitic Device	Value	Units
CP1	1.4	pF
CP2	0.5	pF
RP1	500	Ohm
RP2	300	Ohm
CC1	0.594	pF
CC2	22	fF
RC	454	kOhm

External PCIE Reference Clock Recommended Circuit

Figure 45: PCIe Reference Clock Input



Note: Locate 50 ohm termination resistors close to clock generator, or close to A2000 Target package balls for either source or load impedance matching respectively. The termination resistors are not required for LV-HCSL source terminated type clock sources.

11.7 A2000 Signal Definitions

11.7.1 Power Supply Connections

Signal Name	Balls	Type	Filter	Description
VDD_SENSE	A22	0.8V	-	Isolated VDD connection on IC die that can be used as a remote sense bay an external power supply
VSS_SENSE	A21	GND	-	Isolated VSS connection on IC die that can be used as a remote sense bay an external power supply
Supply				

Signal Name	Balls	Type	Filter	Description
VDD	G12, G14, G16, G18, H11, H13, H15, H17, J12, J14, J16, J18, K11, K13, K15, K17, L12, L14, L16, L18, M7, M11, M13, M15, M17, N8, N10, N12, N14, N16, N18, P7, P9, P11, P13, P15, P17, R8, R10, R18, T9, U8, U10	0.8V	a	Digital core supply
VDDE	J10, L10	0.8V	b	Ethernet DSP
SGMII_VP	T11	0.8V	c	SGMII DSP
PCIE_VP	R12, R14, R16, T13, T17, U16	0.8V	d	PCIE DSP
AVDD_PLL_585	F11	0.8V	e	PLL analog
AVDD_PLL_800	F9	0.8V	e	PLL analog
SGMII_VPH	U12	1.8V	c	SGMII analog
PCIE_VPH	T15, U14	1.8V	f	PCIE analog
ETH_VAA_AGCPL	L8, J8	1.8V	f	Ethernet PLL analog
ETH_VAA	H9, K9, M9	1.8V	g	Ethernet analog
TS_AVDD	R20	1.8V	e	Temp sensor analog
AVDD_PVT	T19	1.8V	e	Temp sensor analog
VDD18	G10, E21, H19, K19	1.8V	h	1.8V IO supply
VDD33	D9, D15, D18, E16, E17, M20, U21	3.3V	i	3.3V IO supply
Ground (Vss)				
AVSS_PVT	U19	GND	-	PC board ground
AVSS_PLL_585	E12	GND	-	PC board ground
AVSS_PLL_800	E10	GND	-	PC board ground

Signal Name	Balls	Type	Filter	Description
VSS	A2, A3, A4, A7, A9, A23, B1, B4, B5, B6, B7, B9, B24, C1, C2, C3, C4, C7, C8, C9, C11, C14, C17, C22, D3, D4, D5, D6, D7, D8, D10, D11, D12, D13, D14, D16, D17, D19, D20, D23, E1, E2, E3, E6, E8, E9, E11, E13, E14, E15, E18, E19, E20, E22, F3, F4, F5, F6, F7, F8, F10, F12, F13, F14, F15, F16, F17, F18, F19, F20, F21, F23, G1, G2, G3, G6, G7, G8, G9, G11, G13, G15, G17, G19, G20, G21, G22, H3, H4, H5, H6, H7, H8, H10, H12, H14, H16, H18, H20, H21, J1, J2, J3, J6, J7, J9, J11, J13, J15, J17, J19, J20, J22, K3, K4, K5, K6, K7, K8, K10, K12, K14, K16, K18, K20, L1, L2, L3, L6, L7, L9, L11, L13, L15, L17, L19, L20, M3, M4, M5, M6, M8, M10, M12, M14, M16, M18, M21, N1, N2, N3, N6, N7, N9, N11, N13, N15, N17, N19, N21, P3, P4, P5, P6, P8, P10, P12, P14, P16, P18, P21, R1, R2, R3, R6, R7, R9, R11, R13, R15, R17, R19, R21, T3, T4, T5, T6, T8, T10, T12, T14, T16, T18, T20, T22, T23, T24, U1, U2, U3, U5, U7, U9, U11, U13, U15, U17, U18, U20, U22, V3, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14, V15, V16, V17, V18, V19, V20, V21, V22, V23, V24, W1, W2, W3, W4, W6, W8, W10, W12, W14, W16, W18, W20, W21, W22, Y3, Y4, Y6, Y8, Y10, Y12, Y14, Y16, Y18, Y20, Y23, Y24, AA1, AA2, AA3, AA5, AA7, AA9, AA11, AA13, AA15, AA17, AA19, AA21, AA22, AA23, AA24, AB3, AB5, AB7, AB9, AB11, AB13, AB15, AB17, AB19, AB21, AB22, AB24, AC1, AC2, AC4, AC6, AC8, AC10, AC12, AC14, AC16, AC18, AC20, AC22, AC24, AD2, AD4, AD6, AD8, AD10, AD12, AD14, AD16, AD18, AD20, AD22, AD23	GND	-	PC board ground

11.7.2 Ethernet 100G Differential High-Speed Signals

Signal Name	Ball	Type	Description
ENET_REFCLK_N	B8	I	Ethernet REFCLK input positive and negative side of differential pair. 156.25 MHz, LVDS levels, 100 ohm differential terminated
ENET_REFCLK_P	A8	I	
ENET0_0_RX_N	E5	I	Negative side of Ethernet Port 0 Receiver Differential pairs
ENET0_1_RX_N	H1	I	
ENET0_2_RX_N	G4	I	
ENET0_3_RX_N	F1	I	
ENET1_0_RX_N	N5	I	Negative side of Ethernet Port 1 Receiver Differential pairs
ENET1_1_RX_N	T1	I	
ENET1_2_RX_N	R4	I	
ENET1_3_RX_N	P1	I	
ENET0_0_RX_P	E4	I	Positive side of Ethernet Port 0 Receiver Differential pairs
ENET0_1_RX_P	H2	I	
ENET0_2_RX_P	G5	I	
ENET0_3_RX_P	F2	I	
ENET1_0_RX_P	N4	I	Positive side of Ethernet Port 1 Receiver Differential pairs
ENET1_1_RX_P	T2	I	
ENET1_2_RX_P	R5	I	
ENET1_3_RX_P	P2	I	
ENET0_0_TX_N	A6	O	Negative side of Ethernet Port 0 Transmitter Differential pairs
ENET0_1_TX_N	D2	O	
ENET0_2_TX_N	C6	O	
ENET0_3_TX_N	B2	O	
ENET1_0_TX_N	J5	O	Negative side of Ethernet Port 1 Transmitter Differential pairs
ENET1_1_TX_N	M2	O	
ENET1_2_TX_N	L5	O	
ENET1_3_TX_N	K1	O	
ENET0_0_TX_P	A5	O	Positive side of Ethernet Port 0 Transmitter Differential pairs
ENET0_1_TX_P	D1	O	
ENET0_2_TX_P	C5	O	
ENET0_3_TX_P	B3	O	
ENET1_0_TX_P	J4	O	Positive side of Ethernet Port 1 Transmitter Differential pairs
ENET1_1_TX_P	M1	O	
ENET1_2_TX_P	L4	O	

Signal Name	Ball	Type	Description
ENET1_3_TX_P	K2	O	

11.7.3 Ethernet 1G SGMII Differential High-Speed Signals

Signal Name	Ball	Type	Description
SGMII_RX_N	U4	I	Ethernet 1G SGMII Receiver Differential pairs
SGMII_RX_P	V4	I	
SGMII_TX_N	AA4	O	Ethernet 1G SGMII Transmitter Differential pairs
SGMII_TX_P	AB4	O	

11.7.4 PCIe Differential High-Speed Signals

Signal Name	Ball	Type	Description
PCIE_REFCLK_N	U24	I	PCIE REFCLK input negative side of differential pair. 100.00 MHz, HCSL levels
PCIE_REFCLK_P	U23	I	PCIE REFCLK input positive side of differential pair. 100.00 MHz, HCSL levels
PCIE_RX0_N	V2	I	Negative side of PCIE Port Receiver Differential pairs
PCIE_RX1_N	Y2	I	
PCIE_RX2_N	AB2	I	
PCIE_RX3_N	AD3	I	
PCIE_RX4_N	AD5	I	
PCIE_RX5_N	AD7	I	
PCIE_RX6_N	AD9	I	
PCIE_RX7_N	AD11	I	
PCIE_RX8_N	AD13	I	
PCIE_RX9_N	AD15	I	
PCIE_RX10_N	AD17	I	
PCIE_RX11_N	AD19	I	
PCIE_RX12_N	AD21	I	
PCIE_RX13_N	AC23	I	
PCIE_RX14_N	Y22	I	
PCIE_RX15_N	W24	I	
PCIE_RX0_P	V1	I	Positive side of PCIE Port Receiver Differential pairs
PCIE_RX1_P	Y1	I	

Signal Name	Ball	Type	Description
PCIE_RX2_P	AB1	I	
PCIE_RX3_P	AC3	I	
PCIE_RX4_P	AC5	I	
PCIE_RX5_P	AC7	I	
PCIE_RX6_P	AC9	I	
PCIE_RX7_P	AC11	I	
PCIE_RX8_P	AC13	I	
PCIE_RX9_P	AC15	I	
PCIE_RX10_P	AC17	I	
PCIE_RX11_P	AC19	I	
PCIE_RX12_P	AC21	I	
PCIE_RX13_P	AB23	I	
PCIE_RX14_P	Y21	I	
PCIE_RX15_P	W23	I	
PCIE_TX0_N	Y5	O	
PCIE_TX1_N	AB6	O	
PCIE_TX2_N	Y7	O	
PCIE_TX3_N	AB8	O	
PCIE_TX4_N	Y9	O	
PCIE_TX5_N	AB10	O	
PCIE_TX6_N	Y11	O	
PCIE_TX7_N	AB12	O	Negative side of PCIE Port Transmitter Differential pairs
PCIE_TX8_N	Y13	O	
PCIE_TX9_N	AB14	O	
PCIE_TX10_N	Y15	O	
PCIE_TX11_N	AB16	O	
PCIE_TX12_N	Y17	O	
PCIE_TX13_N	AB18	O	
PCIE_TX14_N	Y19	O	
PCIE_TX15_N	AB20	O	
PCIE_TX0_P	W5	O	
PCIE_TX1_P	AA6	O	
PCIE_TX2_P	W7	O	Positive side of PCIE Port Transmitter Differential pairs
PCIE_TX3_P	AA8	O	
PCIE_TX4_P	W9	O	

Signal Name	Ball	Type	Description
PCIE_TX5_P	AA10	O	
PCIE_TX6_P	W11	O	
PCIE_TX7_P	AA12	O	
PCIE_TX8_P	W13	O	
PCIE_TX9_P	AA14	O	
PCIE_TX10_P	W15	O	
PCIE_TX11_P	AA16	O	
PCIE_TX12_P	W17	O	
PCIE_TX13_P	AA18	O	
PCIE_TX14_P	W19	O	
PCIE_TX15_P	AA20	O	

11.7.5 GPIO Signals

Signal Name	Ball	In/Out	Volt Level	Weak Pull	Description (Default FW Function)	Default In/Out
GPIO_0	C13	I/O	3.3V	pu	FW Alert	I
GPIO_1	A15	I/O	3.3V	pu	FW LED Green	O
GPIO_2	B15	I/O	3.3V	pu	FW LED Yellow	O
GPIO_3	C15	I/O	3.3V	pu	Board Revision_3: Pulled high or low	I
GPIO_4	C16	I/O	3.3V	pu	Board Revision_4: Pulled high or low	I
GPIO_5	A17	I/O	3.3V	pu	QSFP_0 Module Present#	I
GPIO_6	B17	I/O	3.3V	pu	QSFP_0 Module Select#	O
GPIO_7	A18	I/O	3.3V	pu	QSFP_0 LPMode (low power mode)	O
GPIO_8	B18	I/O	3.3V	pu	QSFP_0 RESET#	O
GPIO_9	C18	I/O	3.3V	pu	Board Revision_9: Pulled high or low	I
GPIO_10	A19	I/O	3.3V	pu	Board Revision_10: Pulled high or low	I
GPIO_11	R23	I/O	3.3V	pu	Isolate EEPROM: Used to prevent third party access to EEPROM during A2000 Target boot activity. See EEPROM Access *(page 187)* for more information.	O

Signal Name	Ball	In/Out	Volt Level	Weak Pull	Description (Default FW Function)	Default In/Out
GPIO_12	B19	I/O	3.3V	pu	OTA Present: reserved for WDC internal use only	I
GPIO_13	C19	I/O	3.3V	pu	Board Revision_4: Pulled high or low	I
GPIO_14	B21	I/O	3.3V	pu	UART Dir: reserved for WDC internal use only	O
GPIO_15	C21	I/O	3.3V	pu	Board Revision_5	I
GPIO_16	A10	I/O	3.3V	pu	QSFP Present	I
GPIO_17	B10	I/O	3.3V	pu	QSFP LPM	O
GPIO_18	C10	I/O	3.3V	pu	QSFP Reset	O
GPIO_19	A11	I/O	3.3V	pu	QSFP Int	I
GPIO_20	B11	I/O	3.3V	pu	QSFP Mod Sel	O
GPIO_21	A12	I/O	3.3V	pu	Flash Reset: A2000 Target FW control of Boot and Configuration Flash reset signal	O
GPIO_22	B12	I/O	3.3V	pu	Connected to PCIE finger (pin) B82	I
GPIO_23	C12	I/O	3.3V	pu	Connected to PCIE finger (pin) A50	I
GPIO_24	A13	I/O	3.3V	pu	See Table 168: Logic Levels for GPIO_24 & GPIO_25 *(page 241)*	
GPIO_25	B13	I/O	3.3V	pu		
GPIO_26	L24	I/O	3.3V	pu	PCIe Clock Select: Output from the A2000 Target which allows FW to select the source of PCIe clock. **Note**: Bit 5 in the PCIe Field in Board Information Area *(page 191)* describes how this GPIO will be driven upon boot.	O
GPIO_27	M22	I/O	3.3V	pu	HW Reset Output: Used as an indicator to board circuits to assert A2000 Target RESET# pin for a prescribed time reset pulse(>1mSec)	O
GPIO_28	N22	I/O	3.3V	pu	Watchdog Event	O

Signal Name	Ball	In/Out	Volt Level	Weak Pull	Description (Default FW Function)	Default In/Out
GPIO_29	R24	I/O	3.3V	pu	Optional PERST#: Output from the A2000 Target to the system to control downstream resets. This GPIO signal is PUSH-PULL, but is tri-stated at power-up and will remain that way until a prescribed time after A2000 Target FW boot. At that time, the IO pin will drive low for a specified time (100ms, 200ms, 400ms… etc.). After this time has passed this GPIO will Drive HIGH. **Note**: Bits 3:0 and 7:6 in the PCIe Field in Board Information Area *(page 191)* describe how this GPIO will be driven upon boot.	I
GPIO_30	P22	I/O	3.3V	pu	Connected to PCIE finger (pin) A19	I
GPIO_31	R22	I/O	3.3V	pu	I2C Async (Attention): Output from A2000 Target FW to indicate attention by enclosure management is required. See Async Notification *(page 122)* for more information. Connected to PCIE finger (pin) A18	I

Table 232: Logic Levels for GPIO_24 & GPIO_25

GPIO	Logic Level			
GPIO_24	0	0	1	1
GPIO_25	0	1	0	1
(Optional) voltage margin control	-5%	nom	nom	5%

11.7.6 I2C Signals

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
SCL0	M24	I/O	3.3V	-	I2C Clock ports [3:0], open drain, Needs external resistor per I2C Fast Mode specification
SCL1	A16	I/O	3.3V	-	
SCL2	A20	I/O	3.3V	-	
SCL3	B22	I/O	3.3V	-	
SDA0	M23	I/O	3.3V	-	I2C Data ports [3:0], open drain, Needs external resistor per I2C Fast Mode specification
SDA1	B16	I/O	3.3V	-	
SDA2	B20	I/O	3.3V	-	
SDA3	C20	I/O	3.3V	-	

11.7.7 SPI Interface

Signal Name	Ball	Type	Vp-p[14]	Weak Pull	Description
SPI_CLK	C24	O	1.8V	-	SPI Clock
SPI_CS#	B23	O	1.8V	pu	SPI Chip Select
SPI_DQ0_MOSI	D24	I/O	1.8V	-	Master out Slave In or DQ0 in 4 bit mode
SPI_DQ1_MISO	C23	I/O	1.8V	-	Master In Slave Out or DQ1 in four bit mode
SPI_DQ2_WP#	E24	I/O	1.8V	pu	Write protect_n or DQ2 in 4 bit mode
SPI_DQ3_HOLD#	E23	I/O	1.8V	pu	Hold_n or DQ3 in 4 bit mode

11.7.8 Reference Signals

Signal Name	Ball	Type	Vp-p	Description
RESREF_PCIE	U6	O	analog	Reference 200 ohm, 1% resistor connection. Resistor is connected between this pin and Ground. Net capacitance < 2pf
RESREF_SGMII	T7	O	analog	Reference 200 ohm, 1% resistor connection. Resistor is connected between this pin and Ground. Net capacitance < 2pf

14. 1.8V tolerant

11.7.9 Reset

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
RESET#	K24	I	1.8V	pu	A2000 Target hardware reset input (>1msec)

11.7.10 UART / Console Signals

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
RXD	N23	I	3.3V	pd	UART received signal
TXD	N24	O	3.3V	-	UART transmitted signal

11.7.11 LED Signals

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
ENET0_LED	A14	O	3.3V	-	Activity LED for 100G Ethernet Port 0
ENET1_LED	B14	O	3.3V	-	Activity LED for 100G Ethernet Port 1

11.7.12 MIIM

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
MDC	P23	O	3.3V	-	Interface Clock: clock driven by the MAC device to the PHY.
MDIO	P24	I/O	3.3V	pu	Data: bidirectional, Needs external pull-up resistor (1.5k~10k)

11.7.13 Test Signals

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
TCK	F24	I	1.8V	pu	JTAG interface for Boundary scan/ Processor/ PCIe SERDES
TDI	G23	I	1.8V	pd	

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
TDO	G24	O	1.8V	-	
TMS	H24	I	1.8V	pu	
TRST#	H23	I	1.8V	pd	
TEST_0	D22	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_1	F22	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_2	H22	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_3	K22	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_4	K23	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_5	L22	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_6	L23	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_7	J21	I/O	1.8V	pd	Reserved for test, leave open for normal operation
TEST_8	K21	I/O	1.8V	pd	Reserved for test, leave open for normal operation
KMD_0	J23	I	1.8V	pd	Reserved for test, external pull-down resistor for normal operation
KMD_1	J24	I	1.8V	pd	Reserved for test, external pull-down resistor for normal operation
EFUSE_T_VQPS	M19	I	1.8V	-	Reserved for test, external 1K pull down resistor for normal operation
EFUSE_F_VQPS	P19	I	1.8V	-	Reserved for test, external 1K pull-down resistor for normal operation
PROC_STALL	D21	I	1.8V	pd	Reserved for test, leave unconnected
PLL_OBSRV	L21	O	1.8V	-	Reserved for test, external 1K pull-down resistor for normal operation
TS_AN_IO_0	P20	I/O	1.8V	-	Reserved for test, external 1K pull-down resistor for normal operation
TS_AN_IO_1	N20	I/O	1.8V	-	Reserved for test, external 1K pull-down resistor for normal operation

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
TSTOUT	E7	O	1.8V	-	Reserved for test, leave unconnected but accessible from a test point. 100G SERDES Analog debug
PVT_SENSE_TS	T21	I/O	1.8V	-	External temp sense diode, connect with external PNP BJT, or resistor pull down to logic ground

11.7.14 No Connection

Signal Name	Ball	Type	Vp-p	Weak Pull	Description
NC	A1	N/A	N/A	N/A	No connection
NC	A24	N/A	N/A	N/A	No connection
NC	AD1	N/A	N/A	N/A	No connection
NC	AD24	N/A	N/A	N/A	No connection

11.7.15 Ball Map

Figure 46: A2000 Target Ball Map

TOP VIEW

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
A	NC	VSS	VSS	VSS	ENET_P0_0_TX_P	ENET_P0_0_TX_N	VSS	ENET_REFCLK_P	VSS	GPIO16	GPIO19	GPIO21	GPIO24	ENET0_LED	GPIO1	SCL1	GPIO5	GPIO7	GPIO10	SCL2	VSS_SENSE	VDD_SENSE	VSS	NC	A
B	VSS	ENET_P0_3_TX_N	ENET_P0_3_TX_P	VSS	VSS	VSS	VSS	ENET_REFCLK_N	VSS	GPIO17	GPIO20	GPIO22	GPIO25	ENET1_LED	GPIO2	SDA1	GPIO6	GPIO8	GPIO12	SDA2	GPIO14	SCL3	SPI_CSn	VSS	B
C	VSS	VSS	VSS	VSS	ENET_P0_2_TX_P	ENET_P0_2_TX_N	VSS	VSS	VSS	GPIO18	VSS	GPIO23	GPIO0	VSS	GPIO3	GPIO4	VSS	GPIO9	GPIO13	SDA3	GPIO15	VSS	SPI_DQ1_MISO	SPI_CLK	C
D	ENET_P0_1_TX_P	ENET_P0_1_TX_N	VSS	VSS	VSS	VSS	VSS	VSS	GPIO_VDD_33	VSS	VSS	VSS	VSS	VSS	GPIO_VDD_33	VSS	VSS	GPIO_VDD_33	VSS	VSS	PROC_STALL	TEST0	VSS	SPI_DQ0_MOSI	D
E	VSS	VSS	VSS	ENET_P0_0_RX_P	ENET_P0_0_RX_N	VSS	TSTOUT	VSS	VSS	AVSS_PLL_800	VSS	AVSS_PLL_585	VSS	VSS	VSS	GPIO_VDD_33	GPIO_VDD_33	VSS	VSS	VSS	GPIO_VDD_18	VSS	SPI_DQ3_HOLDn	SPI_DQ2_WPn	E
F	ENET_P0_3_RX_N	ENET_P0_3_RX_P	VSS	VSS	VSS	VSS	VSS	VSS	AVDD_PLL_800	VSS	AVDD_PLL_585	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	TEST1	VSS	TCK	F
G	VSS	VSS	VSS	ENET_P0_2_RX_N	ENET_P0_2_RX_P	VSS	VSS	VSS	VSS	GPIO_VDD_18	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VSS	VSS	VSS	TDI	TDO	G
H	ENET_P0_1_RX_N	ENET_P0_1_RX_P	VSS	VSS	VSS	VSS	VSS	VSS	ETH_VAA	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	GPIO_VDD_18	VSS	VSS	TEST2	TRSTn	TMS	H
J	VSS	VSS	VSS	ENET_P1_0_TX_P	ENET_P1_0_TX_N	VSS	VSS	ETH_VAA_AGC_PLL	VSS	ETH_VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VSS	TEST7	VSS	KMD_0	KMD_1	J
K	ENET_P1_3_TX_N	ENET_P1_3_TX_P	VSS	VSS	VSS	VSS	VSS	VSS	ETH_VAA	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	GPIO_VDD_18	VSS	TEST8	TEST3	TEST4	RESETn	K
L	VSS	VSS	VSS	ENET_P1_2_TX_P	ENET_P1_2_TX_N	VSS	VSS	ETH_VAA_AGC_PLL	VSS	ETH_VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VSS	PLL_OBSRV	TEST5	TEST6	GPIO26	L
M	ENET_P1_1_TX_P	ENET_P1_1_TX_N	VSS	VSS	VSS	VSS	VDD	VSS	ETH_VAA	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	EFUSE_TEST_VQPS	GPIO_VDD_33	VSS	GPIO27	SDA0	SCL0	M
N	VSS	VSS	VSS	ENET_P1_0_RX_P	ENET_P1_0_RX_N	VSS	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	TS_ANIO_1	VSS	GPIO28	RXD	TXD	N
P	ENET_P1_3_RX_N	ENET_P1_3_RX_P	VSS	VSS	VSS	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	VDD	VSS	EFUSE_FUNC_VQPS	TS_ANIO_0	VSS	GPIO30	MDC	MDIO	P
R	VSS	VSS	VSS	ENET_P1_2_RX_N	ENET_P1_2_RX_P	VSS	VSS	VDD	VSS	VDD	VSS	PCIE_VP	VSS	PCIE_VP	VSS	PCIE_VP	VSS	VDD	VSS	AVDD_MOORTEC	VSS	GPIO31	GPIO11	GPIO29	R
T	ENET_P1_1_RX_N	ENET_P1_1_RX_P	VSS	VSS	VSS	VSS	RESREF_SGMII	VSS	VDD	VSS	SGMII_VP	VSS	PCIE_VP	VSS	PCIE_VPH	VSS	PCIE_VP	VSS	AVDD_PVTSENSE	VSS	PVTSENSE_TS_EXT	VSS	VSS	VSS	T
U	VSS	VSS	VSS	SGMII_RX_N	VSS	RESREF_PCIE	VSS	VDD	VSS	VDD	VSS	SGMII_VPH	VSS	PCIE_VPH	VSS	PCIE_VP	VSS	VSS	AVSS_PVTSENSE	VSS	GPIO_VDD_33	VSS	PCIE_REFCLK_P	PCIE_REFCLK_N	U
V	PCIE_RX0_P	PCIE_RX0_N	VSS	SGMII_RX_P	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	V
W	VSS	VSS	VSS	VSS	PCIE_TX0_P	VSS	PCIE_TX2_P	VSS	PCIE_TX4_P	VSS	PCIE_TX6_P	VSS	PCIE_TX8_P	VSS	PCIE_TX10_P	VSS	PCIE_TX12_P	VSS	PCIE_TX14_P	VSS	VSS	VSS	PCIE_RX15_P	PCIE_RX15_N	W
Y	PCIE_RX1_P	PCIE_RX1_N	VSS	VSS	PCIE_TX0_N	VSS	PCIE_TX2_N	VSS	PCIE_TX4_N	VSS	PCIE_TX6_N	VSS	PCIE_TX8_N	VSS	PCIE_TX10_N	VSS	PCIE_TX12_N	VSS	PCIE_TX14_N	VSS	PCIE_RX14_P	PCIE_RX14_N	VSS	VSS	Y
AA	VSS	VSS	VSS	SGMII_TX_N	VSS	PCIE_TX1_P	VSS	PCIE_TX3_P	VSS	PCIE_TX5_P	VSS	PCIE_TX7_P	VSS	PCIE_TX9_P	VSS	PCIE_TX11_P	VSS	PCIE_TX13_P	VSS	PCIE_TX15_P	VSS	VSS	VSS	VSS	AA
AB	PCIE_RX2_P	PCIE_RX2_N	VSS	SGMII_TX_P	VSS	PCIE_TX1_N	VSS	PCIE_TX3_N	VSS	PCIE_TX5_N	VSS	PCIE_TX7_N	VSS	PCIE_TX9_N	VSS	PCIE_TX11_N	VSS	PCIE_TX13_N	VSS	PCIE_TX15_N	VSS	VSS	PCIE_RX13_P	VSS	AB
AC	VSS	VSS	PCIE_RX3_P	VSS	PCIE_RX4_P	VSS	PCIE_RX5_P	VSS	PCIE_RX6_P	VSS	PCIE_RX7_P	VSS	PCIE_RX8_P	VSS	PCIE_RX9_P	VSS	PCIE_RX10_P	VSS	PCIE_RX11_P	VSS	PCIE_RX12_P	VSS	PCIE_RX13_N	VSS	AC
AD	NC	VSS	PCIE_RX3_N	VSS	PCIE_RX4_N	VSS	PCIE_RX5_N	VSS	PCIE_RX6_N	VSS	PCIE_RX7_N	VSS	PCIE_RX8_N	VSS	PCIE_RX9_N	VSS	PCIE_RX10_N	VSS	PCIE_RX11_N	VSS	PCIE_RX12_N	VSS	VSS	NC	AD
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	

3.3V Signal | 1.8V Signal

Mechanical Information

In This Chapter:

12.1 A2000 Package Information

Figure 48: A2000 Target Package Drawing, Top View

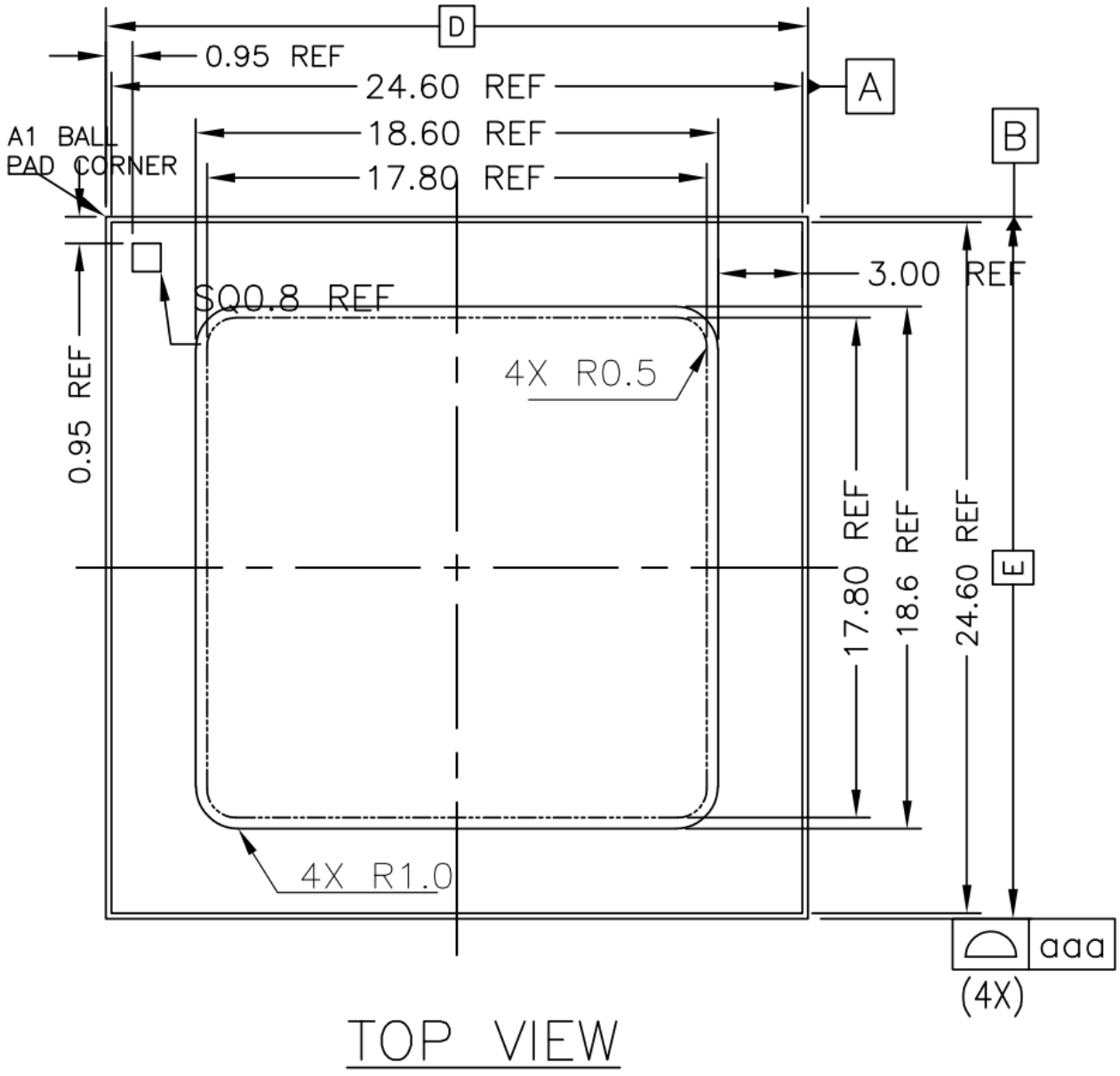


Figure 49: A2000 Target Package Drawing, Side View

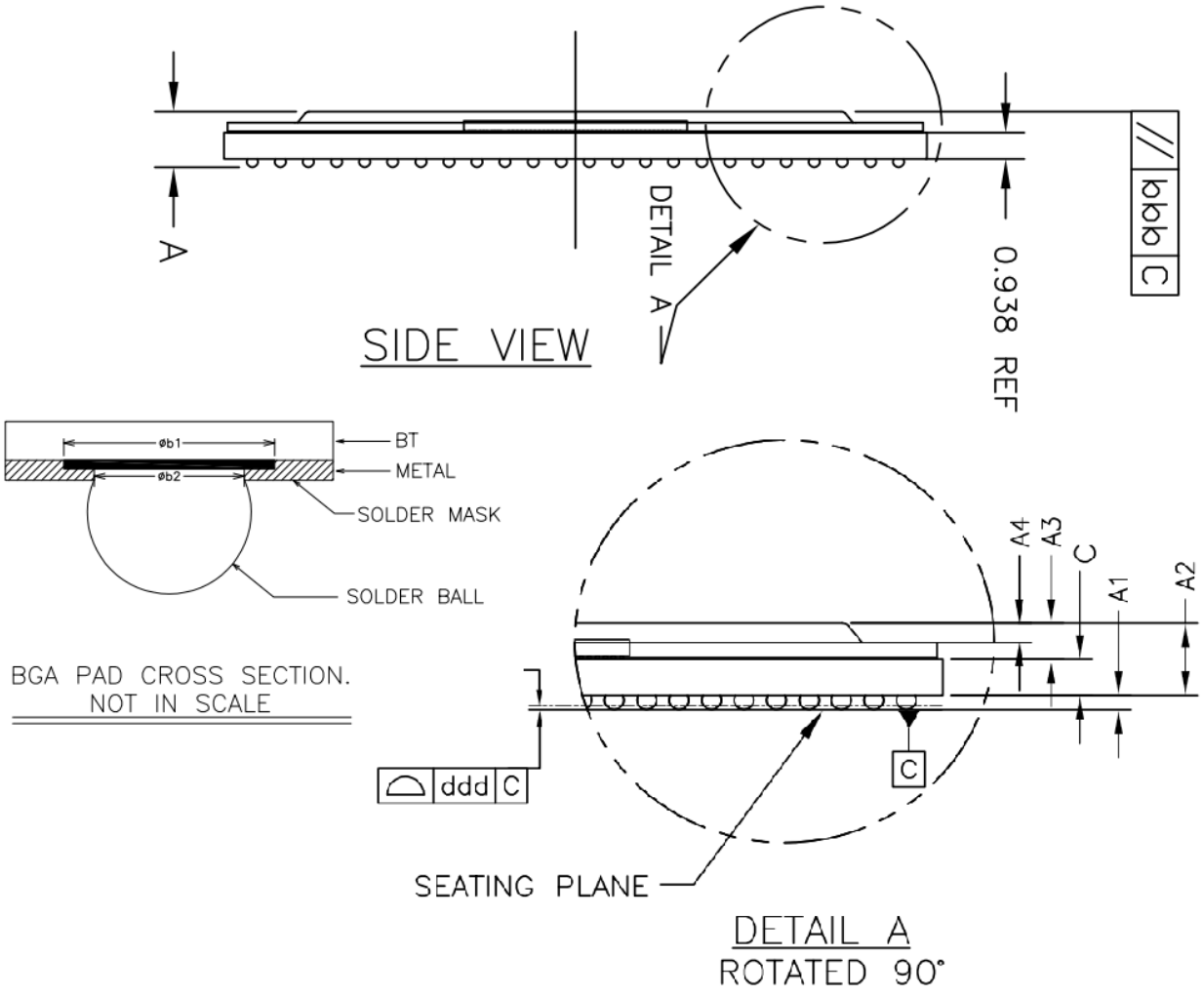


Figure 50: A2000 Target Package Drawing, Bottom View

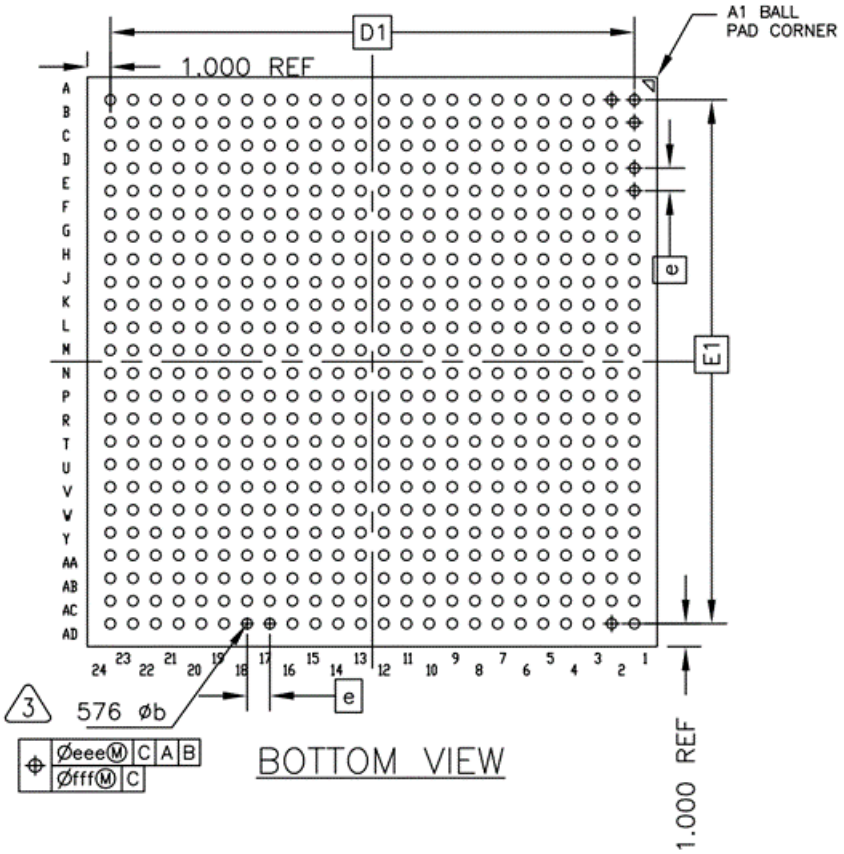


Figure 51: A2000 Target Package Drawing, Notes

Symbol	Dimension in mm		
	MIN	NOM	MAX
A	1.965	2.068	2.171
A1	0.28	0.33	0.38
A2	1.648	1.738	1.828
c	0.838	0.938	1.038
A3	0.76	0.8	0.84
A4	0.41	0.45	0.49
b	0.43	0.48	0.53
b1	0.45	0.5	0.55
b2	0.40	0.425	0.45
D	24.9	25.00	25.1
E	24.9	25.00	25.1
D1	---	23.00	---
E1	---	23.00	---
e	---	1.00 BSC	---
aaa	0.20		
bbb	0.35		
ddd	0.20		
eee	0.25		
fff	0.10		

NOTES:
1. All DIMENSIONS AND TOLERANCE CONFORM TO ASME Y14.5M–1994.
2. TERMINAL POSITIONS DESIGNATION PER JESD 95–1, SPP–010.
3. DIMENSION "b" IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER PARALLEL TO PRIMARY DATUM C.
4. COMPLIANT TO JEDEC DESIGN GUIDE 4.22, NO EXACT VARIATION AND WITH EXCEPTION TO DIM 'A'.
5. RAW SOLDER BALL SIZE DURING ASSEMBLY IS ø0.45MM.

12.2 A2000 Package Thermal Characteristics

The following characteristics apply for a PCB 10 layer 0.5 Oz Cu PCB, 167x 68 x 1.55 cu.mm, PCIE CEM 1.0 Low Profile, Half Length form factor.

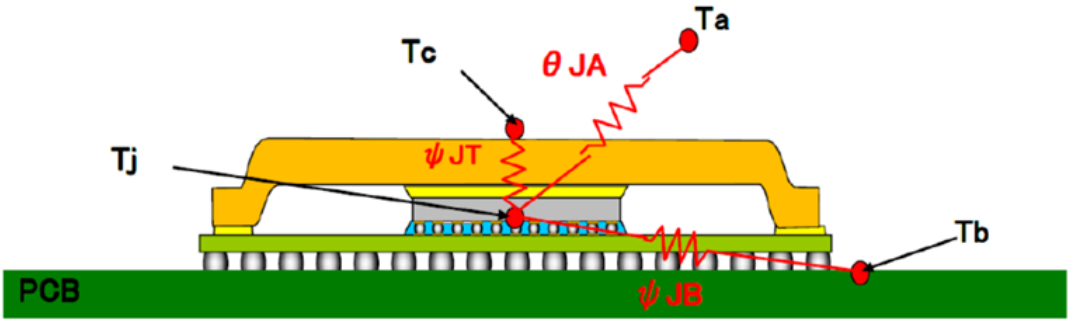
Name		Value	Units
Theta (JA)	0 m/s	11.5	°C/W
	2.5 m/s	8.85	°C/W
Psi (JT)		0.27	°C/W
Theta (JB)		5.11	°C/W
Theta (JC)		0.49	°C/W

$Theta\ JA = \frac{T_j - T_{amb}}{P};$

$Psi\ JT = \frac{T_j - T_{case}}{P}$ *at Theta JA boundary conditions*

$Theta\ JB = \frac{T_j - T_{Board}}{P}$

$Theta\ JC = \frac{T_j - T_{case}}{P}$



Ψjt = (Tj-Tc)/P

P : Power Consumption
Tj : Junction Temperature
Tc : Package surface temperature

Legal and Regulatory

This chapter provides legal and regulatory information for the A2000 Target.

In This Chapter:

13.1 Environmental

Table 243: Environmental Compliance

Targeted Region	Compliance Specification(s)
EU	EU RoHS 2 Directive (2011/65/EU)
	EU Reach Regulation (EC) No 1907/2006
	EU Waste Electrical and Electronic Equipement (WEEE)

13.2 Third Party Licenses

This product may include or use open source software subject to open source licenses. If required by the applicable open source license, Western Digital may provide the open source code to you on request either electronically or on a physical storage medium for a charge covering the cost of performing such distribution, which may include the cost of media, shipping, and handling.

For open source licensing information, please visit https://www.westerndigital.com/company/innovation/open-source/product-compliance.