

Hermes API

User's Manual

Python Library for PCIe Gen6 EDSFF Redriver Control

Version 0.1.0

Serial Cables, LLC
www.serialcables.com

1. Introduction

The Hermes API is a Python library that provides programmatic control of Serial Cables PCIe Gen6 EDSFF Redriver cards. It enables test and validation engineers to automate signal conditioning configuration, device diagnostics, and EDSFF sideband management through a simple, intuitive interface.

This library is designed for integration into automated test frameworks, allowing repeatable configuration and validation of PCIe Gen6 signal integrity across varying hardware configurations.

1.1 Supported Hardware

The Hermes API supports the following Serial Cables PCIe Gen6 EDSFF Redriver cards:

Part Number	Description
PCI6-AD-x8EDSFF-E3-H-SC	PCIe Gen6 x8 EDSFF Redriver, Standard
PCI6-AD-x8EDSFF-E3-H-QA	PCIe Gen6 x8 EDSFF Redriver, Quarch-Compatible
PCI6-AD-x16EDSFF-E3-H-SC	PCIe Gen6 x16 EDSFF Redriver, Standard
PCI6-AD-x16EDSFF-E3-H-QA	PCIe Gen6 x16 EDSFF Redriver, Quarch-Compatible

1.2 Key Features

- Signal tuning: Equalization (EQ), Flat Gain (FG), and Output Swing (SW) control
- EDSFF sideband control: PERST#, dual-port mode, power disable, host LED
- Real-time diagnostics: Temperature, fan speed, current monitoring
- I²C/SMBus access for downstream device communication
- Configuration persistence: Save optimized settings to flash
- Automatic card type detection (x8 vs x16)
- Command-line interface (CLI) with JSON output support

2. Installation

2.1 Requirements

1. Python 3.9 or later
2. pyserial >= 3.5
3. USB connection to Hermes card via CN3 (USB Type-C) connector

2.2 Install from PyPI

Install the package using pip:

```
pip install serialcables-hermes
```

2.3 Verify Installation

Verify the installation by checking the CLI:

```
hermes --help
```

3. Hardware Connection

The Hermes API communicates with the redriver card's MCU via USB serial connection.

3.1 Connection Setup

1. Connect a USB Type-C cable from your host system to the CN3 connector on the Hermes card
2. The card will enumerate as a USB serial device (e.g., /dev/ttyUSB0 on Linux, COM3 on Windows)
3. Ensure the user has read/write permissions on the serial port

3.2 Serial Port Permissions (Linux)

On Linux, add your user to the dialout group for serial port access:

```
sudo usermod -a -G dialout $USER
```

Log out and back in for the group change to take effect.

4. Quick Start

The following example demonstrates basic usage of the Hermes API:

```
from serialcables_hermes import Hermes

# Connect to the Hermes card
with Hermes("/dev/ttyUSB0") as card:
    # Get device information
    version = card.get_version()
    print(f"Firmware: {version.firmware_version}")

    # Set equalization to 8.2 dB on all chips
    card.set_eq("all", 5)

    # Send PERST# to reset the link
    card.send_perst()

    # Check card status
    status = card.get_status()
    print(f"Temperature: {status.temperature_c}°C")

    # Save configuration to flash
    card.save_configuration()
```

5. API Reference

5.1 Connection

Hermes(port, baudrate=115200, timeout=2.0)

Creates a connection to the Hermes redriver card. Supports context manager protocol for automatic resource cleanup.

Parameter	Type	Description
port	str	Serial port path (e.g., /dev/ttyUSB0, COM3)
baudrate	int	Baud rate (default: 115200)
timeout	float	Command timeout in seconds (default: 2.0)

5.2 Information Methods

get_version()

Returns firmware version, serial number, and card identification.

Returns: VersionInfo dataclass with firmware_version, serial_number, card_type fields

get_status()

Returns current card status including temperature, fan speed, and power metrics.

Returns: CardStatus dataclass with temperature_c, fan_rpm, current_ma, power_watts fields

run_bist()

Runs built-in self-test on all PS7161 redriver chips.

Returns: BISTResult with pass/fail status for each chip

detect_edsff()

Detects presence of EDSFF devices on Channel A and Channel B.

Returns: EDSFFDetection with channel_a and channel_b presence status

5.3 Signal Tuning Methods

The Hermes card uses PS7161 redriver chips to condition PCIe Gen6 signals. Each chip provides three adjustable parameters: Equalization (EQ), Flat Gain (FG), and Output Swing (SW).

Equalization (EQ) Values

EQ compensates for high-frequency signal loss. Higher values provide more boost for longer traces.

Step	dB	Step	dB	Step	dB
0	2.3	6	9.5	12	16.5
1	3.2	7	10.3	13	17.9
2	4.3	8	11.6	14	19.3
3	5.4	9	12.6	15	20.8
4	7.2	10	14.0		
5	8.2	11	15.2		

Flat Gain (FG) Values

FG provides broadband gain/attenuation across all frequencies.

Step	dB
0	-2.8
1	-2.0
2	-0.8
3	-0.2

Output Swing (SW) Values

SW controls the output voltage swing amplitude.

Step	mV
0	1050
1	1150
2	1270
3	1380
4	1420
5	1490
6	1600
7	1630

set_eq(chip, value)

Sets equalization value for specified chip(s).

Parameter	Type	Description
chip	int str	Chip number (0-3 for x8, 0-7 for x16) or "all"
value	int	EQ step value (0-15)

set_fg(chip, value)

Sets flat gain value for specified chip(s). Parameters same as set_eq(), value range 0-3.

set_sw(chip, value)

Sets output swing value for specified chip(s). Parameters same as set_eq(), value range 0-7.

get_eq_settings() / get_fg_settings() / get_sw_settings()

Returns current settings for all chips as structured dataclass with per-chip values.

5.4 Configuration Management

load_configuration(mode)

Loads a factory preset configuration optimized for specific trace lengths.

Mode	Description
LoadMode.SHORT	Optimized for traces < 9 inches
LoadMode.MEDIUM	Optimized for traces 6-9 inches (default)
LoadMode.LONG	Optimized for traces 10-12 inches

save_configuration()

Saves current EQ/FG/SW settings to flash memory. Settings persist across power cycles.

5.5 EDSFF Sideband Control

send_perst(channel=None)

Asserts PERST# (PCIe Reset) signal to trigger link retraining.

Parameter	Type	Description
channel	Channel None	Channel.A, Channel.B, or None for both channels

set_dual_port(state)

Enables or disables dual-port mode for EDSFF drives that support it.

Parameter	Type	Description
state	DualPortState	DualPortState.ON or DualPortState.OFF

set_power_disable(level)

Controls the PWRDIS sideband signal level.

set_host_led(state)

Controls the host LED indicator state (LEDState.ON or LEDState.OFF).

5.6 I²C/SMBus Access

i2c_read(address, num_bytes, register=None)

Reads data from an I²C device on the EDSFF sideband bus.

i2c_write(address, data)

Writes data to an I²C device on the EDSFF sideband bus.

6. Command Line Interface

The Hermes package includes a command-line interface for quick operations and scripting.

6.1 Basic Usage

```
hermes -p /dev/ttyUSB0 <command> [options]
```

6.2 Commands

Command	Description
list	List available serial ports
version	Display firmware version and card info
status	Show temperature, fan speed, current
bist	Run built-in self-test
detect	Detect EDSFF devices
eq [chip] [value]	Get or set EQ (no args = show current)
fg [chip] [value]	Get or set FG
sw [chip] [value]	Get or set SW
load <mode>	Load preset (short/medium/long)
save	Save current settings to flash
perst [channel]	Send PERST# (optional: a or b)
reset	Reset the MCU

6.3 CLI Examples

```
# Get card status
hermes -p /dev/ttyUSB0 status

# Set EQ to 8.2 dB on all chips
hermes -p /dev/ttyUSB0 eq all 5

# Output as JSON
hermes -p /dev/ttyUSB0 -j status

# Show EQ value help
hermes -p /dev/ttyUSB0 eq --help-values
```

7. Troubleshooting

7.1 Connection Issues

"Serial port not found" — Verify USB cable is connected to CN3. Check device enumeration with `dmesg | tail` (Linux) or Device Manager (Windows).

"Permission denied" — Add user to dialout group (Linux) or run as Administrator (Windows).

"Command timeout" — Increase timeout parameter. Check MCU is responsive with `hermes -p <port> version`.

7.2 Signal Issues

Link not training to Gen6 — Try increasing EQ value. Longer traces typically need higher EQ (10-15). Verify EDSFF device supports Gen6.

Link unstable / frequent retraining — Adjust SW (output swing) up or down. Check thermal conditions with `get_status()`.

BIST failures — Contact Serial Cables support. May indicate hardware issue with redriver chip.

8. Support

For technical support, documentation, and firmware updates:

- Website: www.serialcables.com
- Email: support@serialcables.com
- PyPI Package: `serialcables-hermes`

© 2025 Serial Cables, LLC. All rights reserved.