

USB3.0 Super-Speed to SpaceWire Bridge

Part Number: BCAI-USSS

Description

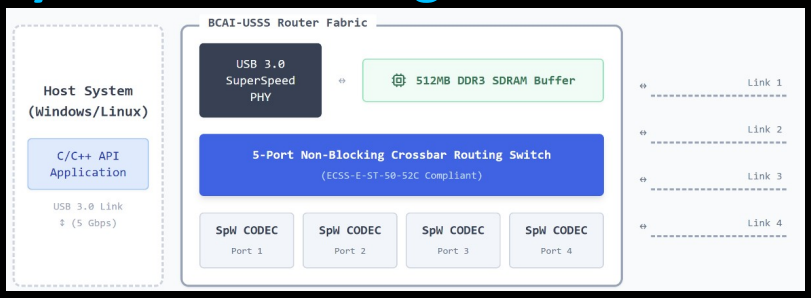
Introducing the **BCAI-USSS** (USB 3.0 SuperSpeed to SpaceWire Router & Bridge), a high-performance piece of Electrical Ground Support Equipment (EGSE) engineered for validation, simulation, and troubleshooting in complex SpaceWire networks. This unit seamlessly interfaces a standard USB 3.0 SuperSpeed host link with a fully embedded, ECSS-E-ST-50-12C compliant and demonstrated 100% conforming to the standard by ESA/ESTEC, providing 4-port SpaceWire Routing Switch.

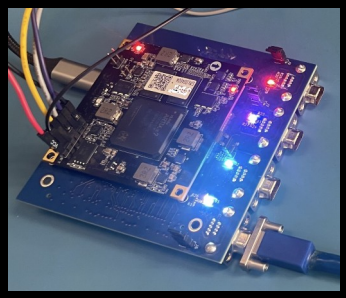
Designed for maximum testbench flexibility, the BCAI-USSS operates in two software-selectable modes:

- **Autonomous 5-Port Non-Blocking Crossbar Switch (ECSS-E-ST-50-52C):** Functions as a central wormhole routing hub (4 physical SpaceWire ports + 1 Host PC port) supporting simultaneous, deterministic packet switching via Path, Logical, and Regional addressing.
- **Direct 4-Channel Multi-Node EGSE Mode:** Exposes the four ports as independent point-to-point SpaceWire interfaces to the Host PC for direct Hardware-in-the-Loop (HIL) node emulation and multi-agent satellite testing.

The architecture natively supports network-wide **ECSS Time-Code** generation, reception, and deterministic forwarding for sub-microsecond spacecraft time synchronization, alongside **Distributed Broadcast Interrupts** (with 32-identifier hardware tracking and acknowledgment handshakes) for low-latency fault and event signaling. Each port features software-configurable transmit baud rates scaling from **1 Mbps up to 200 Mbps** (with automatic 10 Mbps ECSS link initialization). Fortified with an onboard **512 MB DDR3 SDRAM** burst buffer to eliminate host-PC bottlenecks during simultaneous multi-port traffic, the BCAI-USSS delivers up to **3.2 Gbps** of sustained aggregate throughput over its USB 3.0 SuperSpeed backend.

System Block Diagram





Specifications

Dimensions	85mm × 85mm × 25 mm
Power Supply & Data transfer	Via type-C USB connector for data and 2nd type-C for power supply (provided with 2x Type-A male to Type-C male short cable)
Application Programming Interface	Lightweight DLL driver and documented C library: - bc_listdevices(); - bc_open(); - bc_close(); - bc_read(); - bc_write(); - bc_settime(); - bc_gettime(); - bc_settxspeed();
Software	Ready-to-use Application Example, incl. built-in self-test (internal loopback mode) and GUI application.
SpaceWire Port	4x bidirectional link (Fully compliant with the ECSS-E-ST-50-12C standard) with 9-contact micro-miniature D-type **female receptacle** Embedded 5-Port Non-Blocking Routing switch (4 Physical + 1 Host). Supports Path, Logical and Regional addressing. RMAP optional capability
SpaceWire Maximum Speed	200MHz On-board oscillator with configurable speed rates (SDR)
Achievable throughput	Up to 3.2GBps
On-board storage	512MB DDR3 SDRAM, which can be optionally used for time-triggered event or data storage.
Supported Operating System	Windows 11 Contact us for Linux Ubuntu or other OS support

ECSS conformance test report

ECSS Group	Test Name	Measured Hardware Result & ECSS Limit Abaque	Status
Bit Level	Determine Link State	UUT is link enabled; valid D/S transitions observed	SUCCESS
Bit Level	Link Initialisation Test	Link initialisation sequence correct (UUT_delta = 2 µs)	SUCCESS
Bit Level	Start Up Link Speed	Bit-to-bit: 10.00 ± 0.13 Mbps; Rise/Fall: 10.00 ± 0.06 Mbps (Limit: 10 ± 1 Mbps, 130 bits / 13.00 µs)	SUCCESS
Bit Level	Start Up Waveform	Valid start-up waveform 011101000 (D: 0111010001000100..., S: 1101111011101110...)	SUCCESS
Bit Level	Link Shutdown Analysis	Valid shutdown time: 100.00 ± 1.25 ns (ECSS Max Limit: < 555 ns)	SUCCESS
Bit Level	Disconnect Timeout	880 ns ± 10 ns within ECSS valid range [727 ns, 1000 ns] (450–1280 ns, 2000 iter/step)	SUCCESS
Bit Level	Simultaneous D/S Transition	0 simultaneous D/S transitions detected over 10.00 seconds continuous capture	SUCCESS
Exchange	Validate ErrorWait	Moved to ErrorReset on Parity, ESC-EOP, ESC-EEP, ESC-ESC, FCT, N-Char (EOP+EOP), Time-Code 42	SUCCESS
Exchange	Validate Ready / Started	Link configured in active LinkStart=1 mode (immediate Ready→Started transition; all error traps verified)	LINK_START
Exchange	Validate Connecting	Moved to ErrorReset on Parity, ESC-EOP/EEP/ESC, N-Char, Time-Code; accepted FCT → Run	SUCCESS
Exchange	Validate Run	Moved to ErrorReset on Parity, ESC-EOP/EEP/ESC; stayed in Run on FCT, N-Char, and Time-Code 42	SUCCESS
Other	Started 12.8 µs Timeout	Measured: 12.700 µs within ECSS valid range [11.640 µs, 14.330 µs]	SUCCESS
Other	Connecting 12.8 µs Timeout	Measured: 13.101 µs within ECSS valid range [11.640 µs, 14.330 µs]	SUCCESS
Other	Maximum Bit Period	Measured: between 860 and 870 ns (± 10 ns) within ECSS valid range [727 ns, 1000 ns]	SUCCESS
Other	Error Recovery Time	Measured: 20.72 µs (± 0.1 µs) within ECSS valid range [18.46 µs, 22.55 µs] (10,000 iterations)	SUCCESS
Other	Continuous NULLs	Sustained continuous NULL stream reception and transmission without error	SUCCESS

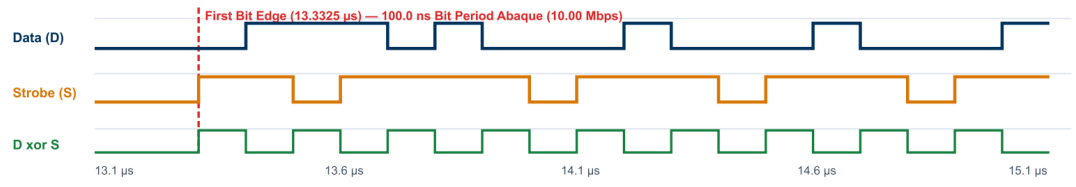


Figure 1: Captured Start-Up Waveform (130 bits / 13.00 µs) — Valid 011101000 NULL & FCT Sequence at 10.00 ± 0.06 Mbps.

The precision charts below plot the hardware verification datasets recorded during the 28-Oct-2026 18:18:19 qualification run directly against the **red ECSS-E-ST-50-12C limit abaques** for Disconnect Timeout, Error Recovery Time, and NULL Arrival Time.

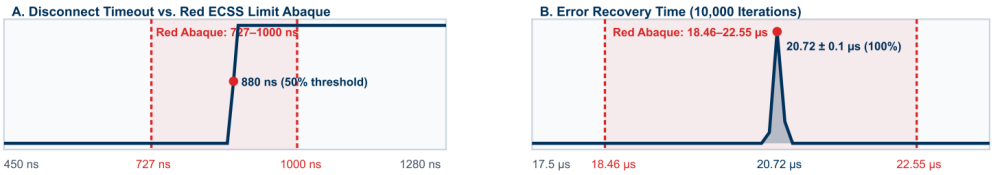


Figure 2: (A) Disconnect Timeout Curve (0% at ≤870 ns, 48.2% at 880 ns, 100% at ≥890 ns; Red ECSS Limit Abaque: 727 to 1000 ns). (B) Error Recovery Time Distribution (100% of all 10,000 test iterations recover sharply within the 20.6–20.8 µs bin centered at 20.72 µs; Red ECSS Limit Abaque: 18.46 to 22.55 µs).

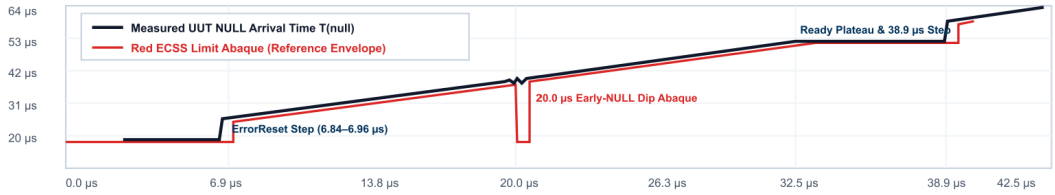


Figure 3: High-Resolution Vector Plot of NULL Arrival Time vs. Red ECSS Limit Abaque — The measured UUT trajectory (black) tracks cleanly above the red ECSS minimum abaque across ErrorReset (6.84–6.96 µs step from 20.72 to 27.64 µs), ErrorWait (20.0 µs dip boundary), Ready plateau (31.5–38.8 µs), and the second step at 38.84–38.96 µs (52.84 to 59.64 µs).

Statistical Verification of Physical Timing Curves: Across all 813 time steps in the NULL arrival dataset (error injection time from 2.52 µs to 35.00 µs, extended to 42.5 µs), the measured NULL arrival time transitions from a constant 20.72 µs baseline during ErrorReset (up to 6.80 µs), steps by 6.92 µs across 6.84–6.96 µs, and follows a linear slope of 1.00 across ErrorWait and Ready to a plateau at 31.48 µs, followed by the second step at 38.84–38.96 µs. In the disconnect timeout test (84 steps, 2,000 iterations per step), the disconnect probability transitions sharply between 870 ns (0.0%) and 890 ns (100.0%), centered at 880.0 ns.

Category	Test Name	Measured Result (`results_4.txt` — 28-Oct-2026 18:18:19)	Status
EOP/EEP	Empty Packet (EOP / EEP)	8 empty EOP packets & 8 empty EEP packets sent; UUT did not disconnect (correct)	SUCCESS
EOP/EEP	Empty Packet Loopback (EOP/EEP)	8 empty EOP & 8 empty EEP packets tested in loopback; UUT did not disconnect (correct)	SUCCESS
EOP/EEP	Send Packet With EEP	Successfully transmitted ` <EEP>` packet (`UUT_recovery = 64 µs`)	SUCCESS
Time-Code	Investigate Time-Code Support	UUT supports time-codes cleanly without disconnecting or corrupting N-Chars	SUCCESS
Time-Code	Time-Code / N-Char Confusion	Single-byte packet `255` + Time-Code `42` separated properly (never merged as `<255 42>`)	SUCCESS
Time-Code	Valid & Invalid Time-Codes	UUT receives valid Time-Code `42` (`tc_reg + 1`) and ignores out-of-sequence Time-Code `21`	SUCCESS
Time-Code	Measure Time-Code Frequency	**637 ticks received (637 valid, 0 invalid); Mean = 64.00 Hz, Min = 64.00 Hz, Max = 64.00 Hz**	SUCCESS
Credit	FCT Overflow Check	1–7 FCTs: stayed in Run (≤56 credits); 8–15 FCTs: moved to ErrorReset (>56 credits)	SUCCESS
Credit	N-Char (EOP / EEP) Overflow	UUT disconnected on credit overflow at 59.99 / 60.00 Mbps UUT link rate	SUCCESS
Credit	Empty Packet Credit Check	UUT correctly credit-counts empty packets and consumes all packets sent (`64 µs` recovery)	SUCCESS
Credit	UUT Credit Error Check	No false credit errors detected during normal credit flow	SUCCESS
Packet (1)	UUT is Data Source	Received 3 packets: `01 <EOP>` [len 1], `02 <EOP>` [len 1], `03 <EOP>` [len 1]	SUCCESS
Packet (1)	UUT is Data Sink	Packet `58 41 20 0f 0d <EOP>` transmitted and consumed by UUT	SUCCESS
Packet (2)	UUT is Command Receiver	Sent `58 41 20 0f 0d <EOP>` → UUT returned expected reply `01 <EOP>`	SUCCESS
Packet (2)	UUT is Command Sender	UUT sent expected command `02 <EOP>` and received reply `58 41 20 0f 0d <EOP>`	SUCCESS



Supporting Your Innovation

Targeting Demanding Aerospace Technologies and Edge-AI Applications