PHOENIX-SPACEWIRE

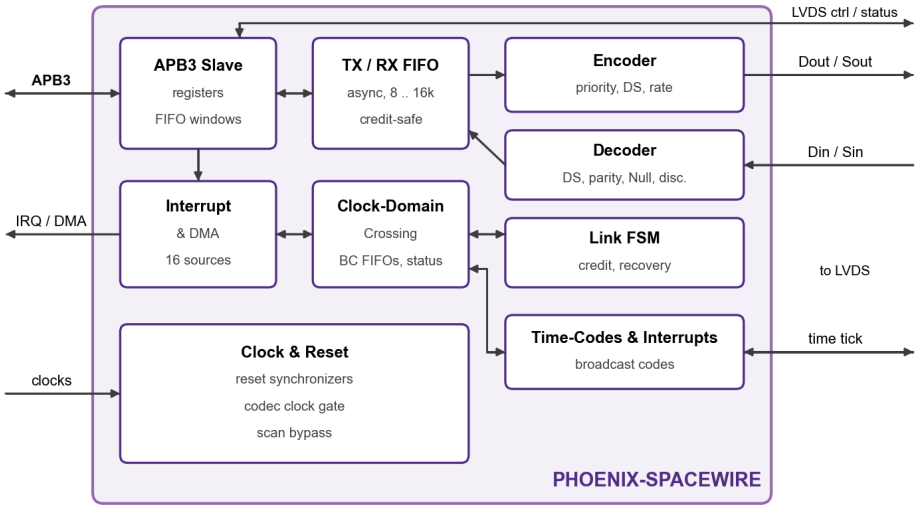
SpaceWire Node Codec (ECSS-E-ST-50-12C Rev.1) with APB3 Host Interface

PHOENIX-SPACEWIRE is a SpaceWire node codec IP core compliant with ECSS-E-ST-50-12C Rev.1. It implements the encoding layer, the data link layer and the node functions of the network layer for one SpaceWire port: Data-Strobe encoding, link initialisation, flow control, link error recovery, packets, time-codes and distributed interrupts. External LVDS cells connect it to the cable.

Software controls the core through an AMBA 3 APB slave. Packets move through configurable TX and RX FIFOs by CPU or DMA; receive credit is granted only for free FIFO space, so no data is lost when the host is slow. Time-codes are sent from a register or a hardware tick input and delivered on a tick output; distributed interrupt and acknowledgement codes are sent and received as 32-bit bitmaps. Sixteen interrupt sources are available both as discrete lines and as one combined line.

The codec runs on its own clock, asynchronous to the APB clock. The transmit rate is a programmable division of it and the receiver oversamples Data and Strobe, so no clock recovery is needed: 200 Mbit/s with a 400 MHz codec clock. All clock-domain crossings sit in one module, and a clock gate stops the codec in a clean idle state for low standby power.

Block Diagram



Implementation Results

Default configuration: TX / RX FIFO depth 64 (flip-flop storage), codec clock 400 MHz (ASIC) / 200 MHz (FPGA); timing met.

Configuration	Technology	Area / Resources
FIFO 64 / 64	TSMC 28nm HPC+	7,120 µm² · 18.8k gates
FIFO 64 / 64	AMD Virtex UltraScale	1,120 LUTs · 980 FFs

ASIC: Design Compiler, ssg 0.81 V 125 °C, pclk 200 MHz, 1 gate = ND2D1 (0.378 µm²); the two FIFOs are about half of the area. FPGA: Vivado, xcvu440-2, placed and routed. Contact SILIPRISM for results on your target technology.

Verification

Verified in a UVM environment with an APB3 master and a SpaceWire link-partner model written from the standard, 29 tests including error injection in every link state, rate sweeps with ppm offset and jitter, and two cores back to back. Regressions pass in three configurations (400 MHz / FIFO 64, 100 MHz / 16, 125 MHz / 256) through LVDS cell models, with a directed testbench (44,567 checks) and C driver co-simulation (83 checks). Coverage: 99.7% functional, 98.7% line, 100% FSM, 97.4% branch. No unwaived lint, CDC or RDC violation; netlist equivalence proven (2,290 comparison points).

Deliverables

- Synthesizable Verilog-2001 RTL source
- SDC constraints; Design Compiler and Vivado synthesis scripts
- SpyGlass lint, CDC and RDC setup with reviewed waivers
- UVM and directed testbenches, APB3 / SpaceWire VIP, LVDS cell model
- C99 bare-metal driver with examples, host and co-simulation tests
- Specification, programming guide and verification guide

FEATURES

ECSS-E-ST-50-12C Rev.1

- Data-Strobe encoding, odd parity, first Null and disconnect detection (850 ns)
- Link state machine with Rev.1 start conditions; optional legacy LinkDisabled behaviour
- Flow control with credit errors, sending priority, link error recovery
- 10 Mbit/s start rate, run rate = f_codec / (N + 1), changed on the fly
- Up to 200 Mbit/s with a 400 MHz codec clock
- Codec clock 45 MHz or more for the 10 ± 1 Mbit/s start rate (some bands from 20 MHz)

Network Layer (Node)

- Packets of any length, EOP / EEP
- Time-code master (register or tick input) and slave (tick output)
- Distributed interrupts and acknowledgements, 32 IDs, standard priority
- Reserved broadcast codes counted

Easy Integration

- AMBA 3 APB slave, zero wait states; 4-byte write window
- TX / RX FIFO depth 8 to 16384 N-Chars
- 16 interrupt sources on discrete and combined lines; DMA requests
- PortReset: flush FIFOs and restart the link by software
- Asynchronous APB and codec clocks; codec clock gate for low standby power
- Active-low resets with internal synchronizers; scan test mode
- LVDS line driver / receiver enables and 22 vendor configuration outputs, 8 status inputs

Software Support

- C99 bare-metal driver with a HAL-style API, no C library needed
- Reference examples and a programming guide

Applications

Spacecraft on-board data handling: instrument, mass-memory and processor interfaces, payload data links and time distribution in satellite and launcher avionics.

Contents subject to change without notice.
Trademarks are the property of their respective owners.