PCIe COMPLIANCE CASE STUDY

Bluehatsoft
Embedded firmware consulting

From Failed Link-Up to PCI-SIG Compliance: Gen5 AI Accelerator Card

6 weeks
to compliance

9 of 10
server platforms (Intel, AMD, ARM64)

Pass
lane margining: PCI-SIG and Linux LMT

Client

A Silicon Valley SoC startup building an SoC with multi-core Arm application processors and a proprietary power-management processor, delivering high-efficiency AI inference in a low-power envelope. Its add-in card (AIC) is a PCIe Gen5 x4 (32 GT/s) endpoint in CEM form factor.

The challenge

During PCI-SIG compliance testing, the AIC:

- Failed Configuration Verification (CV) tests
- Failed to link up with several servers in interoperability testing
- Failed to enumerate when connected to some hosts
- Failed to sustain Gen5 throughput, with a large number of AER-reported errors
- Failed lane margining while passing protocol and electrical tests

Results

- Compliance achieved in **6 weeks**, against the client team's own estimate of 4 months
- 9 of 10 server platforms (Intel, AMD, ARM64) now interoperate. The remaining platform, an Intel-based server released in 2010, failed because of an outdated host BIOS
- Lane margining passes with both the PCI-SIG test and the Linux Lane Margining Tool
- The AIC is now on the PCI-SIG Integrators List
- Access to Enterprise and Industrial markets, where PCI-SIG compliance is a purchasing prerequisite
- No surprises at the compliance lab: Bluehatsoft's in-lab mock tests (interoperability, configuration verification, lane margining) predicted the official results

Why Bluehatsoft

Hands-on experience with PCIe IP from Synopsys, Rambus/PLDA, Cadence, and AMD/Xilinx

Firmware for devices on I/O buses like PCIe, CXL, USB3 (XHCI and endpoints), SPI, I2C, I3C, and GPIO

Linux/Windows drivers for your software stack

Is your PCIe, CXL, or USB3 product ready for compliance?

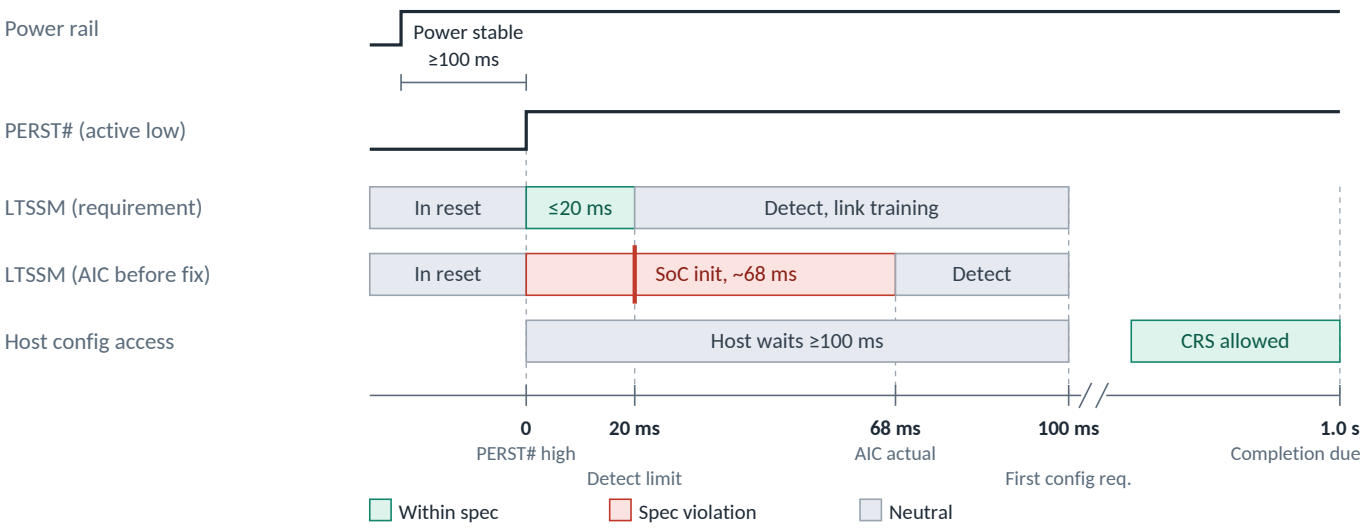
Ask about a pre-compliance assessment.

www.bluehatsoft.com | contact@bluehatsoft.com | 408-475-5169

Root causes we found and fixed

Bluehatsoft
Embedded firmware consulting

Reset timing: PERST#, LTSSM Detect, and the first configuration completion



The card's LTSSM entered Detect about 68 ms after PERST# deasserted, against a 20 ms limit. Time axis not to scale; it breaks between 100 ms and 1.0 s. For links above 5.0 GT/s, the host's 100 ms wait starts after link training completes.

Symptom	Root cause	Fix
CV test failure.	Incorrect Next Capability Offsets in the PCIe extended capability list.	Corrected Next Capability Offsets.
Link never trained; host system could not detect the PCIe Endpoint (EP).	PERST# tied to board reset. The PCIe Base Specification requires the EP PHY's LTSSM to enter Detect within 20 ms of fundamental reset deassertion. When board reset was tied to PERST#, system initialization routines took about 68 ms before the EP PHY's LTSSM entered Detect state, failing to meet the timing requirement.	Decoupled PERST# from board reset; redesigned the firmware PCIe configuration path to handle asynchronous PERST#.
Some host systems failed to enumerate the EP during a warm reboot.	EP never returned configuration read completions or Configuration Request Retry Status (CRS) to the host within 1 s of the end of reset. The host root complex hit a completion timeout while enumerating the device.	EP firmware was modified to enable CRS in the controller so the endpoint returns CRS to configuration requests until the SoC is ready to handle configuration transactions.
Large number of correctable errors reported via AER (Bad TLP, Bad DLLP), limiting throughput.	The AIC used its local oscillator by default, with spread-spectrum clocking disabled. The frequency mismatch between local clock and host-supplied refclk caused frequent elastic buffer overruns.	Reprogrammed the AIC's clock MUX to source its reference clock from the host (100 MHz refclk via the card-edge connector).
Host OS failed to hibernate / suspend to memory.	WAKE# terminated to ground on the AIC. A ground termination holds the open-drain WAKE# signal permanently asserted.	Removed the pull-down resistor so WAKE# is driven only by the endpoint's open-drain output.
Receiver lane margining failed.	EP configuration used fixed equalization coefficients that were suboptimal for the failed platforms.	Implemented firmware-controlled LTSSM retraining to request alternative equalization coefficients from the downstream port.