



### Verification Component Overview (eVC)

e Verification Components are reusable, configurable, and plug-and-play into verification environments. They model popular protocols, interfaces, processors, and embody the full Specman Elite® methodology. eVCs yield the fastest and most complete verification approach for each target protocol by delivering directed-random generation of transactions, full assertion checking, and functional coverage analysis of interesting transactions, scenarios, and events. In addition, since eVCs are plug-and-play components, you can intermix all eVCs that comply with the e Reuse Methodology (eRM) within your verification environment. eRM is a complete reuse methodology that codifies the best practices for eVC development.

e Verification Components provide numerous advantages including a major increase in productivity and higher quality products. With eVCs, verification environments are created in days instead of weeks or months. You can begin writing tests much earlier and complete verification faster.

e Verification Component reusability provides yet another major productivity advantage. For example, after performing module-level testing, the same eVC can then be used for top-level verification, enabling you to retain your training investment from project to project.

### SAS eVC Overview

The eVC by IntelliProp is configurable, easy to install, and reusable. It meets current SAS Revision 3b specifications released by the T10 committee.

### Verification Environment

For system level verification, the SAS eVC will test features of the phy, link, port and transport layers. It can be programmed to send SCSI commands, and is capable of testing the various data and identify Frames available in SAS. SSP and SMP protocol checks are built-in and there are hooks for the STP protocol. 8b10b Encoder/Decoder, Scrambling, and CRC checks are part of the eVC.

### Deliverables

- Verified eVC code
- Documentation – user guide/release notes
- Test suite

### Contact

For more information or to purchase please contact IntelliProp in the following manner:

Email: [amits@intelliprop.com](mailto:amits@intelliprop.com)

Address:

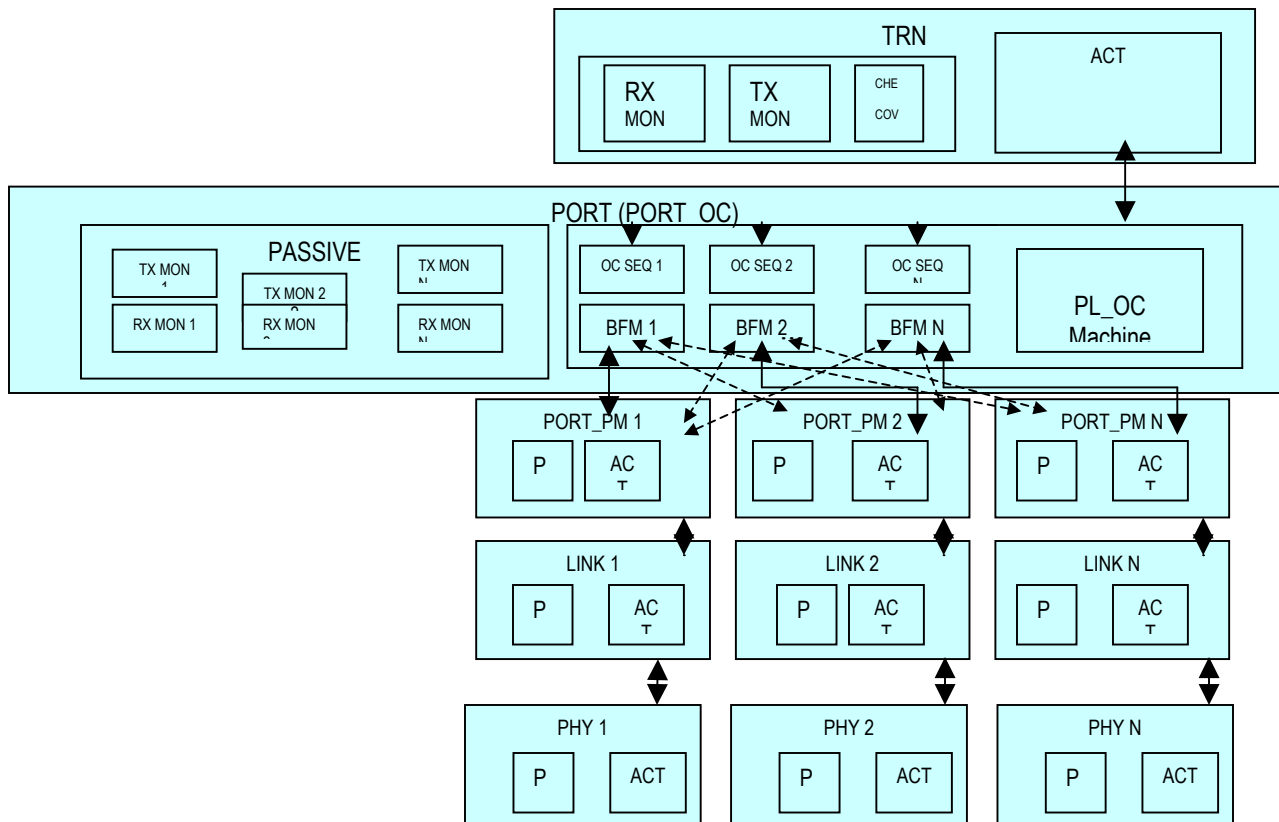
IntelliProp Inc.

4202 Fern Ave

Broomfield CO 80020

### Functional Description

Component	Description
PHY Layer Agent	The PHY layer sequence sets up the connections and checks for timing when connecting to the DUT PHY and SERDES. This agent also includes Bus Functional Model to drive the HDL interface signals. The PHY is responsible for all OOB signals and resets.
LINK Layer Agent	The LINK layer sequence transmits, receives scrambles and unscrambles frames after adding SOF, EOF and CRC to the PORT request. The LINK layer also generates all primitives.
PORT Layer Agent	The PORT layer is in control of PHY connections. The PORT layer also controls the counter and timers for connections per PHY.
TRANSPORT Layer Agent	The TRANSPORT layer sequence constructs SSP and SMP frames for transmission. The TRANSPORT layer also controls the closing of connections.



Features:

- Programmable as SAS Initiator model
- Programmable as SAS Target model
- SSP and SMP protocol checking.
- Easy integration with IntelliProp’s SATA eVC model
- Supports and checks for proper timing and misalignment errors in PHY.
- Supports primitive sequence generation and transmission at all layers.
- Configurable pattern generation for random, directed or erroneous patterns
- Configurable setup for scoreboard integration.
- Checkers will verify protocol timing checks and functional accuracy at each layer.
- Supports layered transmitter and receiver states
- Includes encoder/decoder, scrambler, and CRC checking.
- Built with eRM guidelines (scalable to test multi-port SAS hosts)
- Includes protocol coverage to show what protocol test have been produced.

SAS eVC Architecture (Figure Above):

The figure shows the architecture of the SAS eVC configured as a SAS Host. A similar configuration is possible for the Device.

The connections to/from DUT are done via a bus functional model in the PHY layer TX agent that drives the HDL signals. There is an optional HDL interface at the link layer that can be used if the DUT has a link layer interface. Each agent in the TX side has sequence, sequence driver and a monitor (for coverage and protocol checks). Each agent in the RX side consists of a monitor that collects coverage and performs protocol checks. The lowest layer monitors on the TX and RX side listen to the traffic on the HDL interface to collect the transmitted or received information. The higher layer monitors derive the information from lower layer monitors. The RX monitors also communicate with the respective peer agent on the TX side if the sequence being sent has to respond to the information received on the RX side.

