

SpaceWire Test, Verification and Certification Requirements and Approaches

Yuriy Sheynin

St. Petersburg State University of Aerospace Instrumentation

St.Petersburg, RUSSIA

sheynin@aanet.ru



What to Test, Verify and Certify (TVC)

SpaceWire Test and Verification:

- Devices that implement SpaceWire
 - SpaceWire links, link controller chips
 - SpaceWire nodes
 - SpaceWire routers
- IP-blocks that implement SpaceWire
 - RTL-models
 - Post-synthesis models
 - System-level models

Whose responsibility?

- Standardization board, SpaceWire WG ?
- Designer ?
- Manufacturer ?
- User ?

The responsibility is divided between them

Many interconnection standardization boards have TVC as part of their activity, part of the standardization (RapidIO, Infiniband, MIPI, etc.)

Testing and Verification classes

- Device Interoperability testing (DI);
demonstrates interoperability with a limited set of reference devices.

- Specification compliance testing (SI);
demonstrates that a standard implementation complies the standard itself (no testing equipment specifics are taken into account)

Device Interoperability testing (DI)

(RapidIO as an example)

- DIL-1: Device Initialization, Identification, basic transaction sending/receiving
- DIL-2: Detailed Device testing for all the protocol functions
- DIL-3: Testing Device operation and reaction to external errors

SpaceWire standard(s)

- We have the basic SpaceWire standard
- We (almost) have its extension by Transport layer protocol
 - Transport layer packet format with PID
 - RMAP,
 - Future TPs
- We plan to have next SpaceWire standard release this year

What to support by TVC

1. The basic SpaceWire standard
2. Next release of the basic SpaceWire standard
3. Transport layer protocols, as they will evolve
 - Any new protocol, or a feature into an existing one, should have a support for TVC

What we have

- ❑ Some devices for TVC that have been developed by different companies
- ❑ Proprietary tests, TVC procedures
- ❑ No specified as a document methodology for TVC
- ❑ No agreed and specified test sets, no test sets available for a designer or tester
- ❑ No available instrument set for these non-specified yet methodology

What should be provided along with the standard by the SpaceWire WG

- ❑ Methodology for Test and Verification
(informative)
- ❑ Methodology for Certification
(normative)
- ❑ Test sets
- ❑ Instruments for Test, Verification and Certification
(support in development,
recommended list)

What should be provided by the Designer

- ❑ Testing environment for his design
- ❑ Complement to the standard test set by specific for particular UUT settings (e.g. setting configuration, operation modes, routing tables for a router, etc.)
- ❑ Analysis and interpretation of log files

Problems: open issues in the SpaceWire standard

- Configuration space of nodes and routers is not specified in the SpaceWire, not unified.
 - A node and a router operation depends upon configuration settings,
(e.g. a Router and its routing tables, adaptive routing registers, etc.)
 - We have to deal with the problem somehow...
- [Testing and Verification -- Another reason for Configuration space standardization in SpaceWire]
- Current SpaceWire standard doesn't specify detailed format of packets and reaction of a node to a particular packet

Instruments for device TVC

- Obvious:
 - Test to the UUT generation
 - Running TVC procedures
 - Analyzing results
 - Logging, creating log-files with detailed data in the course of testing
- Generating external faults for a SpaceWire UUT
 - Logical faults
 - Protocol faults
 - Physical level, signal level faults (?)
(e.g. jitter generation?)

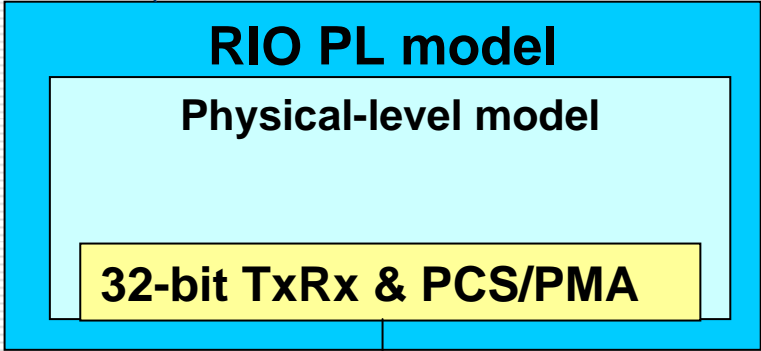
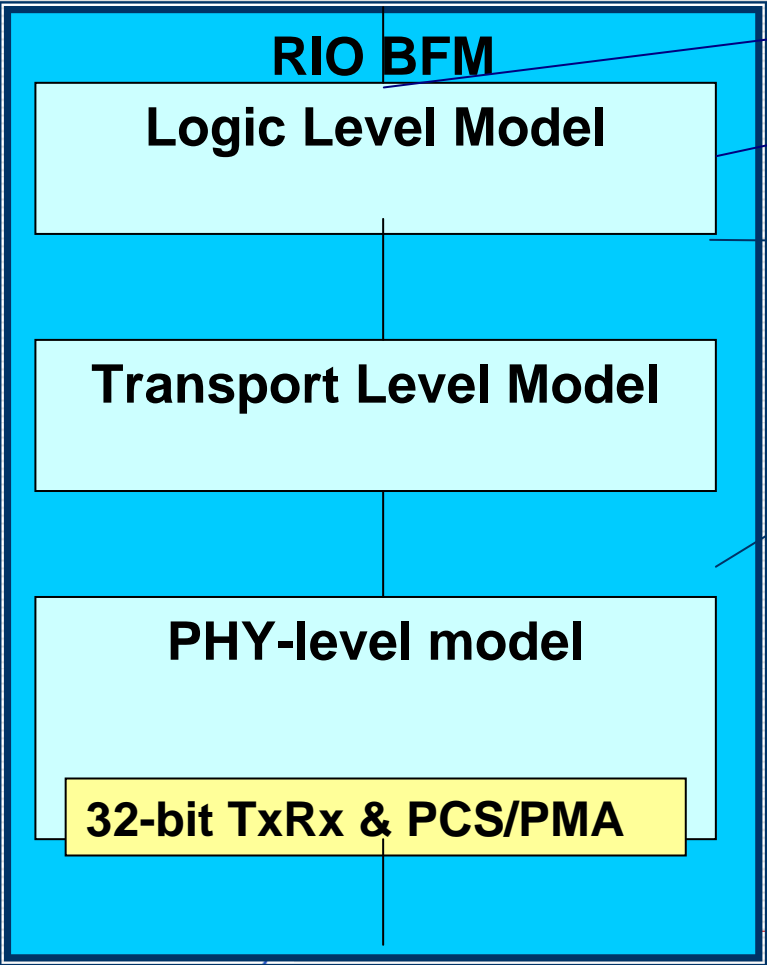
Certification

- ❑ Normative requirements to Certification of products for compliance with the SpaceWire Standard:
procedures, tests, instruments, reports
- ❑ Certification Centres ?

Support for IP-blocks testing and verification

- BFM (Base Formal Model)
 - hierarchical model of a protocol port
 - hierarchical model of a basic router
- BFM levels correspond to the standard's protocol stack layers.
- BFM are programmed in a high-level languages (e.g. **SystemC** for RapidIO BFM, VHDL, Verilog)

BFM Structure for RapidIO



«local device» interface
 «memory directory» interface
 «local memory» interface
 PHY-level interface

8/16 LP-LVDS or 1x/4x LP-Serial

8/16 LP-LVDS or 1x/4x LP-Serial

Testing with BFM

- ❑ IP-block (or chip) designer includes BFM modules in his testing environment
- ❑ Complement the standard test set by specific for particular UUT settings
- ❑ Runs the testing environment with his design for a sequence of standardized TVC procedures
- ❑ Similar procedures, with the ready-made component models, test sequences and prescribed testing procedures could be done by a user of and IP-block

Next steps

- ❑ Coordinate our efforts in TVC
- ❑ Analysis and development of TVC methodology document as a chapter in the SpaceWire standard
- ❑ Specification and development of test sets
- ❑ Designing a set of BFMs for SpaceWire, various levels

Thank you !



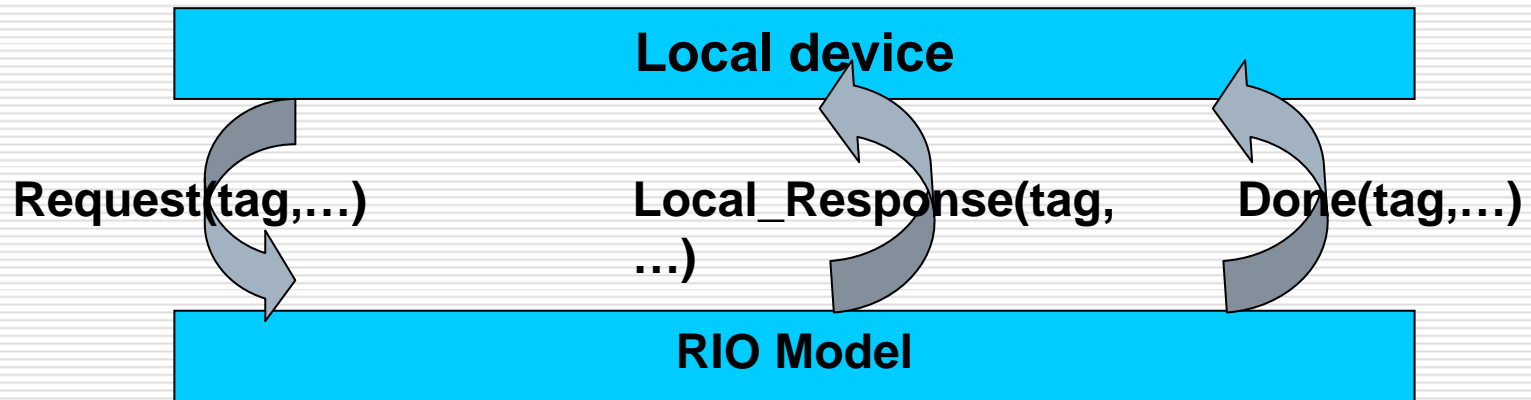
Back up



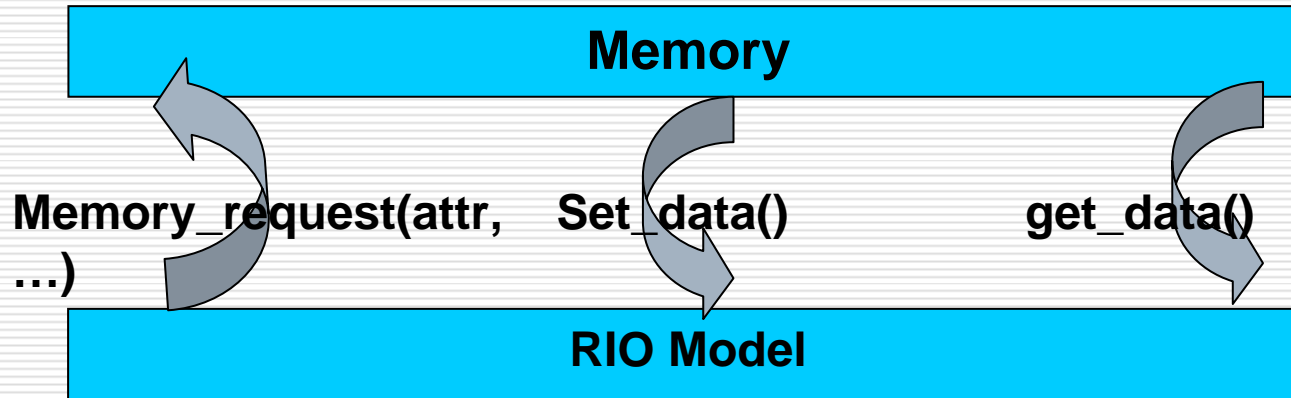
SUAI is ready to participate in:

- ❑ BFM development for the SpaceWire (SystemC, VHDL)
- ❑ Review and development of standard's sections on test, verification and certification methodology, procedures
- ❑ Development of standardized tests, test sequences

Interface «local device»



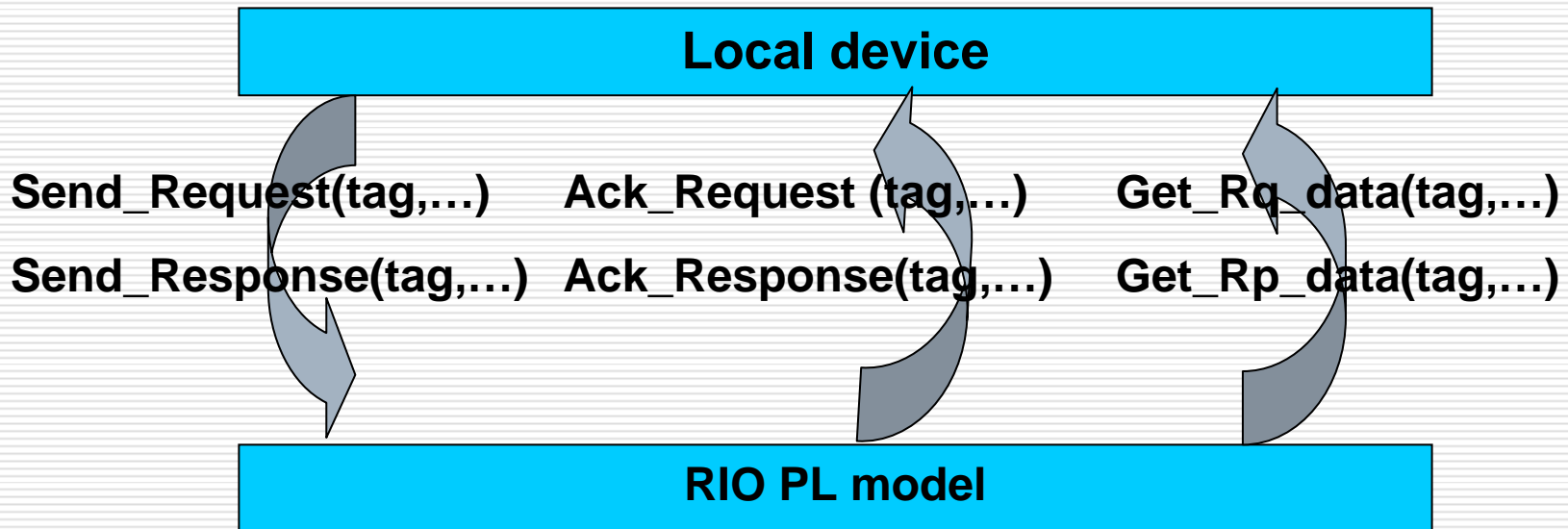
«memory directory» Interface



Data exchange with memory:

1. RIO model (SystemC) calls the `memory_request` function (data transfer direction address words number as parameters)
2. When the memory device is ready for data interchange it calls functions `get_data` or `set_data`

PHY-level interface



The Local device calls function **Send_***** to send a packet.

If the RIO PL model is ready for such type of request, it calls function **Ack_*****; after it, **Get_data** to get data that should constitute the packet.

Instruments for testing

- Protocol analyzer
(e.g. IBCTracer 4X for the Infiniband)
- What levels of the SpaceWire protocol to cover?

Tests and test traffic

- Traffic generator
- Test sets for SpaceWire protocol stack layers
 - packet traffic test sets
 - Signals test sets, test vectors sequences