SpaceWire Standard Evolution

Towards next SpaceWire standard release

Yuriy Sheynin
St. Petersburg State University of Aerospace Instrumentation
Institute of High-Performance Computer and Network Technologies
190 000 St. Petersburg
Bolshaya Morskaya st., No 67
E-mail: sheynin@online.ru

Main topics

General

- Limits for the SpaceWire standard evolution.
  What to do and what not to do in the SpaceWire standard evolution.
- SpaceWire outside Space applications.

Physical level

- Cables, cable assembly, distances

Signal level

- Bit rates vs distances. Link start rate. Adaptable link rate

Physical level/Signal level

- Conductivity isolation problems with SpaceWire links

Character level

- Distributed Interrupts

Exchange level

- State machine specification: error correction and Simplex mode modification

Network level

- Broadcast/multicast. Nodes with multiple links. Configuration space

General
Limits for the SpaceWire standard evolution

What to do and what not to do in the SpaceWire standard evolution

- Not to loose the SpaceWire simplicity and compactness of implementations.
  Not to design another Infiniband or RapidIO
- Where to stop in upgrading the SpaceWire protocol stack?
  - At Transport Level?
  - At Application Adaptation Layer (AAL)?
  - Add some Network management services ?

General
SpaceWire outside Space applications

- Feasibility of any modern technology is governed by its volume of production and application
- Space applications – low volume production
- Necessary to move SpaceWire technology to wider market
  - Avionics is a natural candidate
  - Industrial and telecommunication applications
- Not so fine, but more feasible cabling, connectors, etc. should be defined in the SpaceWire standard also

Physical level

Variety of cables

- Eliminate particular cables type specification as normative; make it informative.
  Substitute it by requirements to the cable characteristics
- Set a section with requirements for non-space environment (e.g. for EGSE) and military (e.g. for airborne applications, for railroad systems), industrial, commercial environments

Variety of distances

- Longer distances (with corresponding bit rates scaling) allowed

Cable assembly

- Add requirements specification for cable assemblies with cable/cable connectors inside

Variety of connectors

- Connector requirements in the normative part of standard, connector types – in the informative part
- New connector types to be recommended in addition to the 9-pin D-type connector

Signal level

Bit rates vs distances

- Higher bit rates for limited distances
  600 – 800 Mb/s @ 1-2 meters could be reasonable on a PCB or inside a block
- Longer distances with lower upper rates limits
  e.g. 150 Mb/s @ 20 m, 80-100 Mb/s @ 25-30 m, etc.
  can make a real difference for some SpaceWire applications

Link start rate.

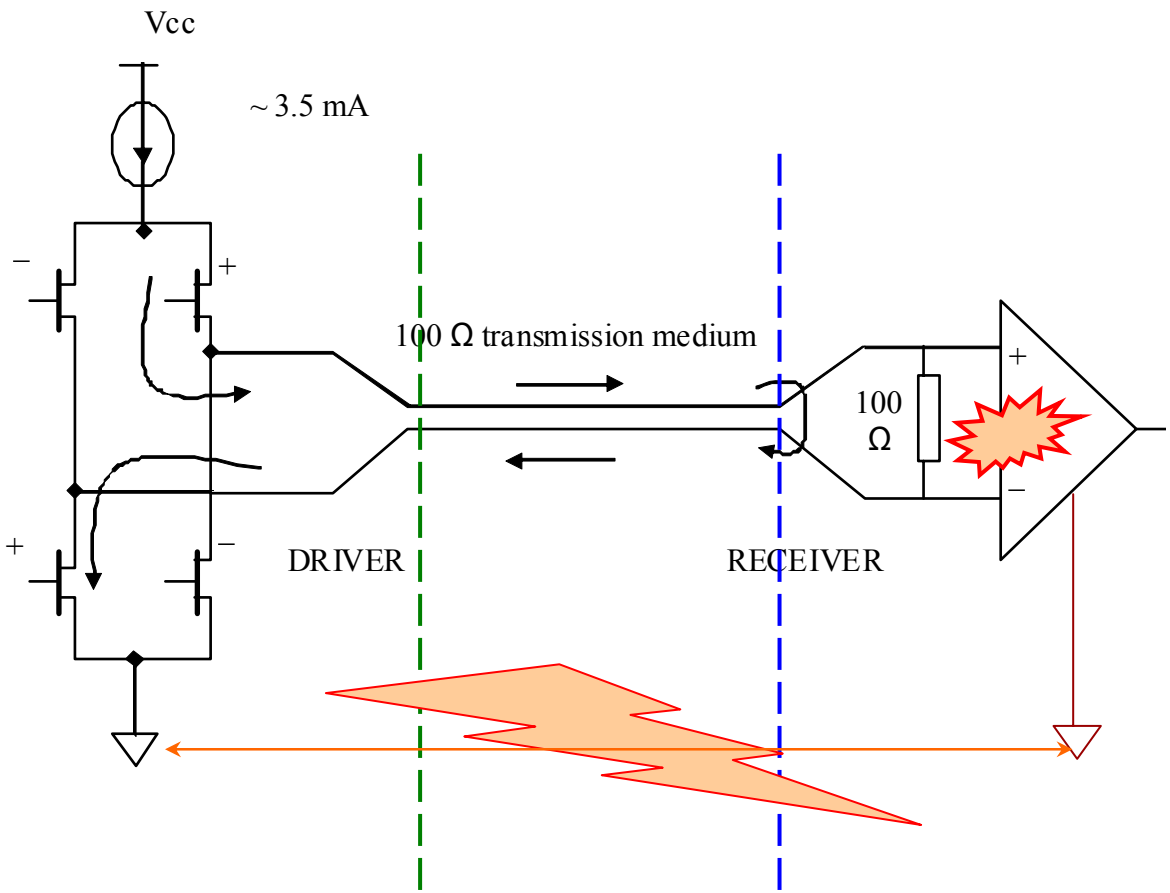
- Lower link start rate ?
- Higher link start rate
  - When the link is running at regular rate of hundreds of Mb/s,
    to restart the link starting at 10 Mb/s after every detected error and then moving to the regular for this link rate causes unreasonable delays, gaps in information flow.
    We restart a link at its *regular rate* at once.
- **Adaptable link rate**
  - duplex link rate matching procedure by negotiation and/or by sequence of attempts is required.
    SpaceWire is a standard with smooth, continuous rates scale
    and lack of a two-side procedure to agree on rates looks as a flaw in the standard

Physical level/Signal level

Galvanic isolation problems with SpaceWire links

- How to use a SpaceWire link to connect galvanic isolated blocks ?
- How to ensure conductivity isolation for a SpaceWire link ?

SpaceWire link to connect galvanic isolated blocks



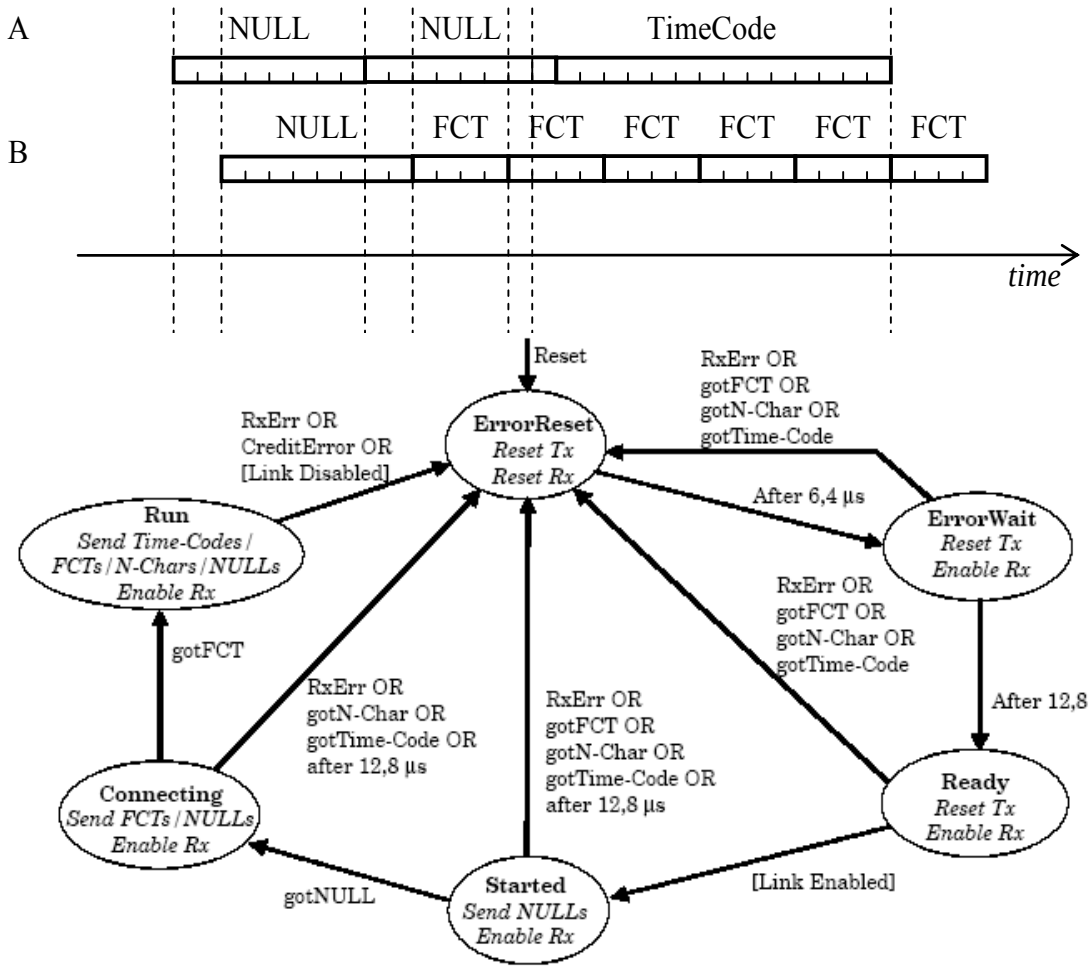
Character level Distributed Interrupts

- Proposed at the 2nd SpaceWire WG meeting in October, 2004
- Since December 2006 the specification draft for Distributed Interrupts is pending for comments at the SpaceWire WG web site
- Additional control codes, processing and routing procedures for Distributed Interrupts
- Implemented in a variety of chips

Exchange level
State machine correction

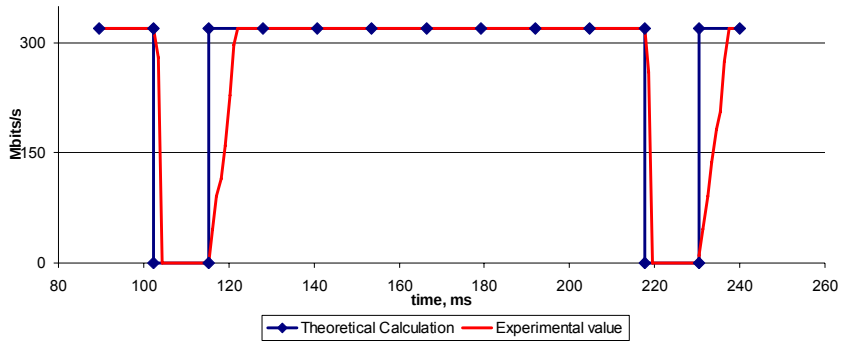
An only sending node can never set a connection

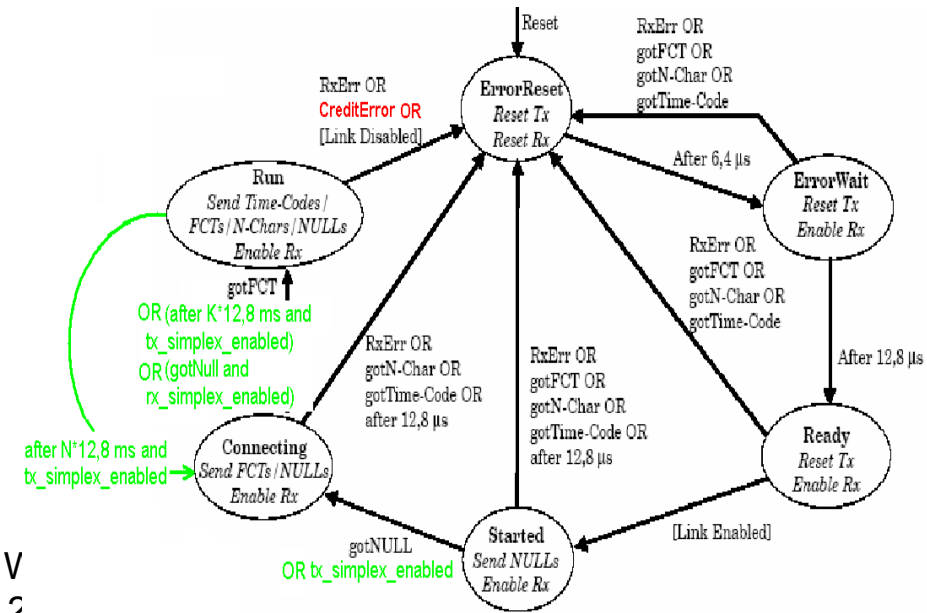
Add a requirement always to send FCT before going to the RUN state



Exchange level
Link Simplex mode

- Using two new signals – *tx_simplex_enabled* and *rx_simplex_enabled*
- two types of the simplex mode link operation – transmitting simplex or receiving simplex.
- Transmitting: transmitter sends data for $N*12{,}8$ microseconds.
- Reconnecting: transmitter goes to Connecting State and sends only NULL symbols on the frequency 10MHz for $12{,}8*K$ microseconds.



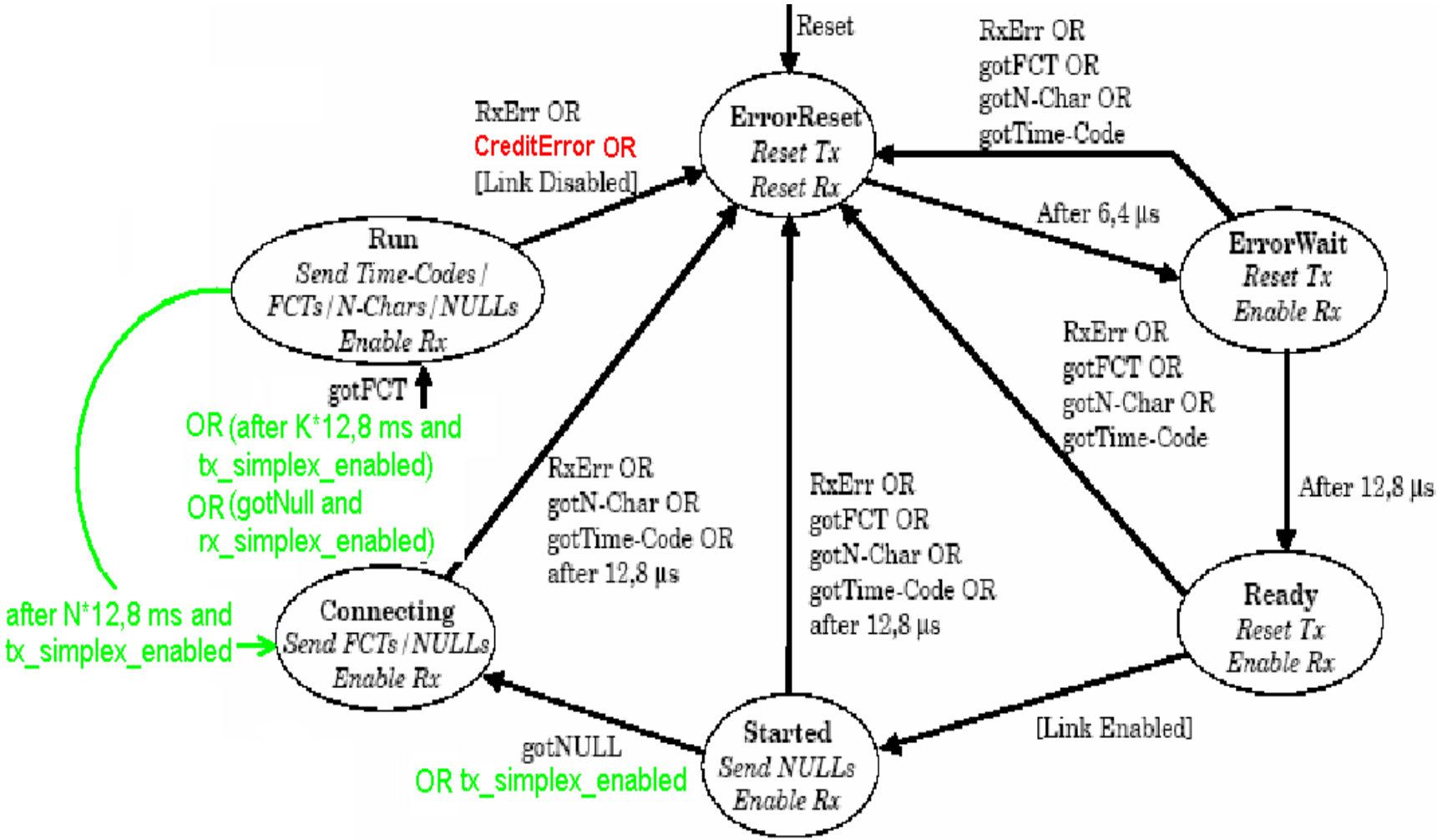


Network level

- Broadcast/multicast modes in SpaceWire interconnections
  In the standard it is limited to router → node.
  It can be extended for router → router for some interconnection topologies, (e.g. tree) and accurate routing tables writing
- SpaceWire nodes with multiple links (not a router!)
  Many SpW nodes implementations have more than one link (for fault-tolerance, for throughput improvement, etc.).
  It isn't covered in the standard, how the links and the node should operate (same/different LA, common/separate time-code register(s), etc.)
- Configuration space (basic) specification
  for a router and for a node (with regard to SpaceWire links)

Back up

State machine with Simplex mode



8.5.2.6 Connecting

a. The *Connecting* state shall be entered from the *Started* state after a NULL is received (gotNULL condition set).

b. On entering the *Connecting* state a 12,8 µs timeout timer shall be started.

c. In the *Connecting* state the receiver shall be enabled and the transmitter shall be enabled to send FCTs and NULLs.

d. If an FCT is received (gotFCT condition true) the state machine shall move to the *Run* state.

e. If, while in the *Connecting* state, a disconnect error, parity error or escape error is detected, or if any character other than NULL or FCT is received, then the state machine shall move to the *ErrorReset* state.

f. If the 12,8 µs timeout referred to in point b. above occurs then the state machine shall move to the *ErrorReset* state.

NOTE The *Connecting* state is entered when the link interface (end A) receives a NULL, waiting then for the reception of an FCT indicating that the other end of the link (end B) has also received a NULL. When the link interface receives a NULL and an FCT it means that communication is established in both directions. If an FCT fails to arrive within 12,8 µs then something is wrong with the link connection and so the link interface is reset once more (*ErrorReset* state) and connection is attempted once again.