SpaceWire

Implementation of a TLM 2.0 model

Dr. Nikos Mouratidis
Qualtek Sprl.
36 Avenue Gabriel Emile Lebon
B-1160, Brussels, Belgium
01/04/2010

Project Overview

- Activity objectives:
  - High-level modeling of the ESA SpaceWire using SystemC TLM 2.0
    - SystemC ⇒ Simulation speed
    - TLM 2.0 ⇒ Interoperability
  - Functional validation and timing accuracy analysis of said model
    - Model functions identical to RTL IP core
    - Reported timing is useful for system performance evaluation
- Intended uses of activity outputs:
  - Allow ESA to develop new VPs based on the available SystemC IP Models and distribute them to contractors without being subject to any fee or restriction, allowing software development before SoC hardware is ready
  - ESA will license the SystemC IP Models without being subject to any fee or restriction and allow contractors to develop new VPs using the guidelines in the DF and perform design space exploration for future SoCs to be implemented either as ASICs or on FPGA

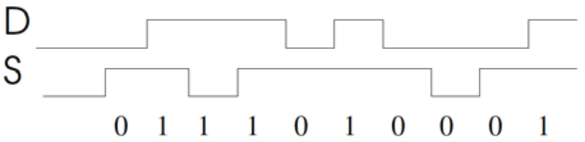
- Point-to-point data connection
  - Lightweight protocols (w.r.t to alternatives)
  - Low-latency data transfer (minimal buffering)
  - Flow controlled (data loss avoidance)
- low-complexity technology for building scalable, fault tolerant networks
  - using routing switches connected by point-to-point links
- Aims at equipment compatibility and reuse
- High speed data link
  - Minimum speed is ~2Mb/s
- Widely adopted

SpaceWire standard*

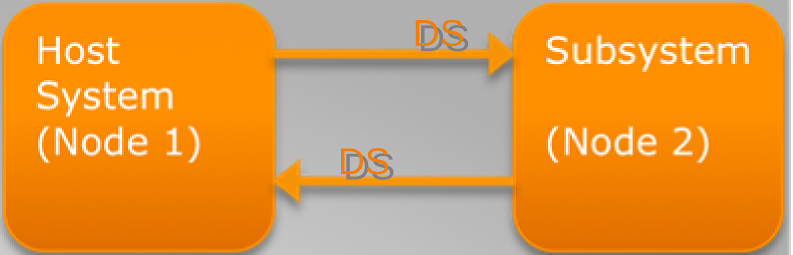
*Available at http://www.ecss.nl

SpaceWire CODEC

- SpaceWire-b (SpW-b) CODEC
  - compliant with SpaceWire standard
  - serial transmitter/receiver
  - full-duplex, bidirectional, point-to-point data link
- Data Tx via 2 (pairs of) wires using LVDS
  - The clock can be recovered as (D xor S)
    - Phase-locked-loops are not needed
    - Accurate control of clock frequencies not needed
  - Link failure detected by loss of the derived clock signal



- Operation:
  - a SpaceWire node sends Tokens to the node on the other end of the link
  - Each Token that a node sends indicates to the receiving node that the sender has 8 bytes of available buffer space
    - Three tokens would indicate there are 24 bytes of available space in the Host System buffer
  - Both nodes are senders and receivers of Tokens and data etc.



Host System (Node 1)

DS

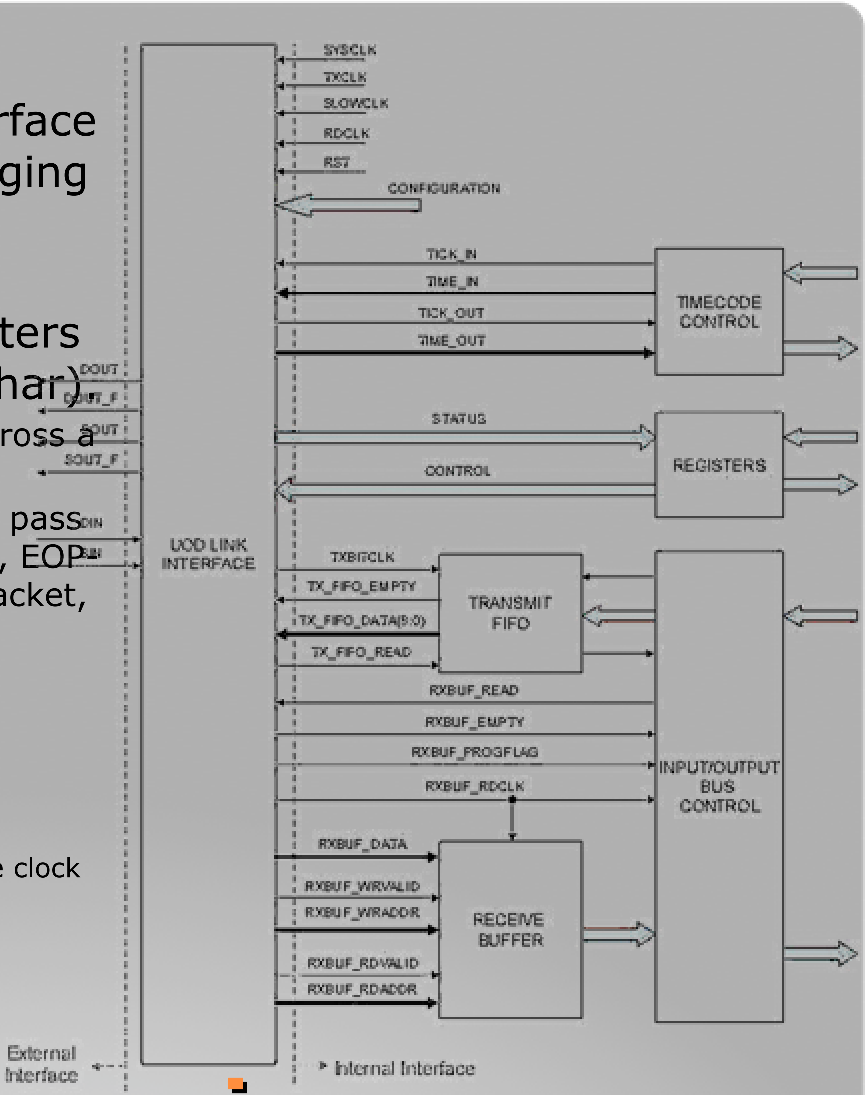
DS

Subsystem

(Node 2)

SpaceWire IP overview

- CODEC responsible for making connection with the SpaceWire interface at the other end of a link and managing the flow of data across the link
- Interface transmits and receives characters which can be link characters (L-Char) or normal characters (N-Char).
  - L-Chars used to manage the flow of data across a link (NULL & FCT - Flow Control Token)
  - N-Chars are the characters that are used to pass information across the link (data characters, EOP-Normal End of Packet, EEP – Error End of Packet, and time-codes)
- Configurable
  - Pipelined / non pipelined
  - DDR or SDR outputs
  - Transmission clock configuration options
    - Allow independent Tx clock and default reference clock
  - Internal variable data rate generation
  - Configurable receive buffer size; FCT credit counter operations are handled internally



- Why is the model of an existing IP necessary?
  - Royalty-free access
    - Development of VPs prior to commitment / licensing
    - Actual IP licensed only after grounds for commitment
    - Licensing process does not stall development
  - Simulation speed
    - x10s to x1000s improvement over RTL simulation
    - Simulation kernel constitutes part of model
      - Linked library
  - No specialized simulation tools necessary
  - Interoperability
    - Compliant models utilise standardised interface
      - TLM 2.0 implements interoperability layer
    - Model may be directly used in compliant platforms
      - No interface adaptations necessary
  - Different use cases demand different level of detail
    - Software development – LT (b_transport)
    - Design space exploration – AT (nb_transport)

Motivation

Qualtek
Thinking Forward

Implementation objectives

- Dual coding style targets
  - Interfaces at two different accuracy levels to be implemented
- Timing accuracy
  - Max Divergence from RTL: 20%
- Functional compliance (Spec. ⇔ RTL)
  - Same input ⇒ same output
- Simulation Performance
  - Assessed against RTL IP
- CAD tool independence
- Scalability
  - Memory footprint / CPU load

Implementation constraints

- Timing report target examples (±20%)
  - Tx 1024 bytes@200mbits, 1024 byte packet: 51.22 us (159.937 Mbps effective rate)
  - Tx 1024 bytes@200mbits, 1 byte packet: 103 ns (114.286 Mbps effective rate)
  - Delay between ErrorReset and ErrorWait: 6.4 us
  - Delay between ErrorWait and Ready: 12.8 us
- Several times faster on same machine
- Internal accuracy
  - Character level
- Interface accuracy
  - LT: transfers at the packet level ⇒ no flow control
  - AT: transfers at the character level ⇒ all control in place

Implementation decisions - I

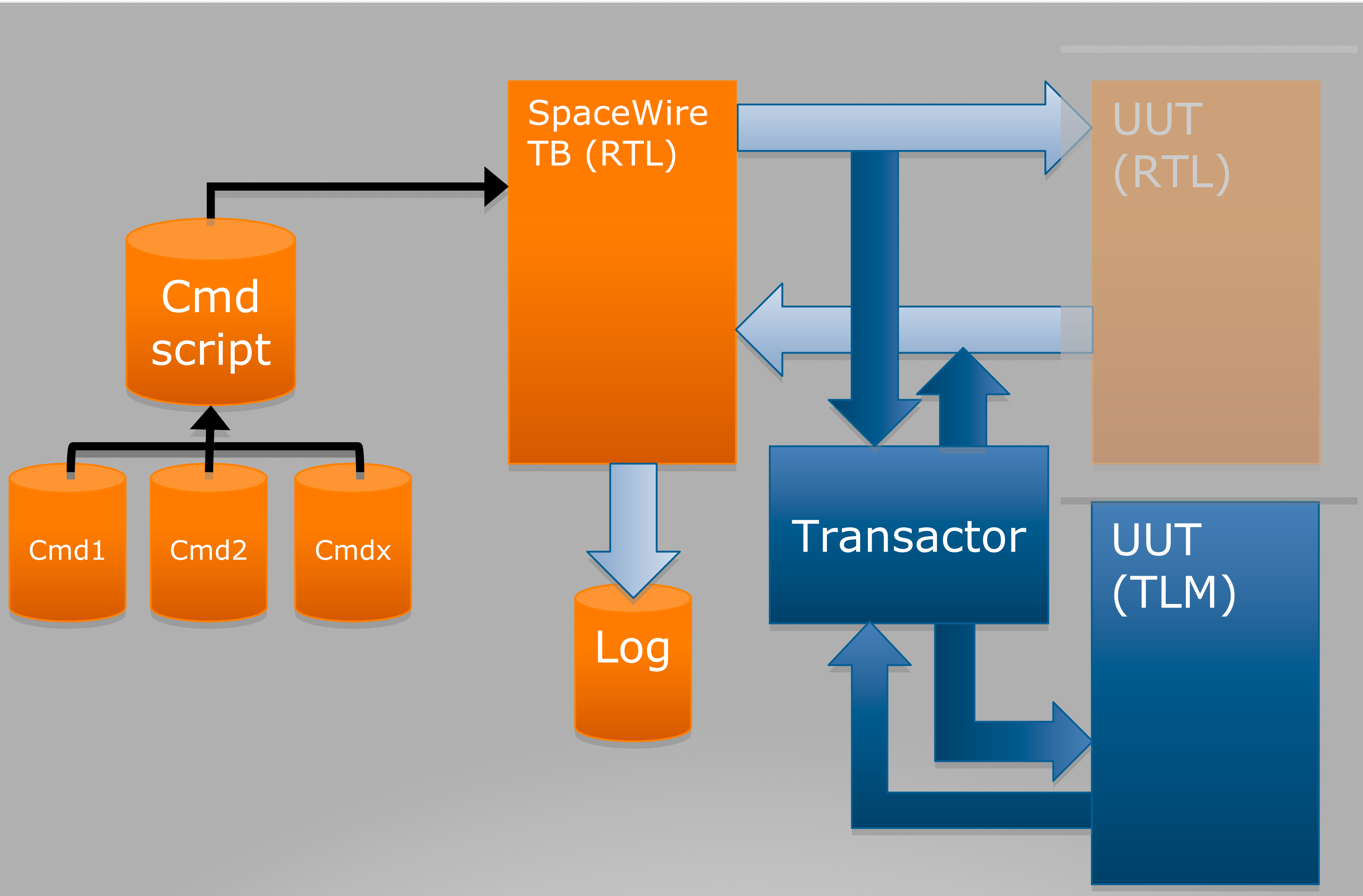
- Temporal decoupling
  - Individual SystemC processes are permitted to run ahead without actually advancing simulation time until they reach the point when they need to synchronize with the rest of the system
  - SystemC processes run ahead of simulation time for an amount of time known as the time quantum
- DMI – Direct Memory Interface
  - Enables an initiator to bypass the usual path through the interconnect components used by the transport interface (i.e. b_transport or nb_transport calls from initiator through interconnect components to target)
  - Intended to accelerate regular memory transactions in a loosely-timed simulation
- Timing annotations instead of explicit timing
  - Communicating processes exchange timing related information instead of actually delaying simulation
  - WAIT statements are aggregated, thus avoiding multiple context switches

Implementation decisions - II

- LT/AT switching
  - Style switching desirable during model execution
  - Loosely Timed – helps in advancing simulation faster during sections not important for particular use case
  - Approximately Timed – provides increased timing detail and data accuracy, but slows simulation down
    - LT ➔ AT
      - Switching has to delay for the length of the maximum 'wait' within all outstanding b_transport calls
    - AT ➔ LT
      - If outstanding nb_transport calls exist, issuing of b_transport calls is stalled until every single nb_transport completes
- Simulation control
  - Details like transaction tracing and value recording are done through SystemC coding
    - Unlike vendor specific simulation tools that use script languages and GUI commands to control simulation details

Verification challenges

- RTL core provides meticulous testbench
  - Re-use methodology desirable
    - Interface (e.g. transactors)
    - Timing – satisfy assertions of testbench
- Verification coverage
  - ≥90% of code



Testbench – re-use method

Qualtek
Thinking Forward

Verification strategy

- TBV - Transactor Based Verification
  - Original testbench used
  - TLM model interfaced through transactors
  - Automated verification mechanisms from RTL maintained to great extent
  - Possibility to adopt incremental block replacement using some transactors internally
  - Variable and Transaction Recording
    - Values of a variable across time recorded using value-change callback functions
    - Recording timing information and attribute information associated with transactions

Tools

- Build platform
  - SystemC 2.2.0
  - TLM 2.0.1
  - SCV 1.0e
  - GCC version ≥4.3.4
  - Cmake version ≥2.8.1
- Waveform viewer
  - GTKWave

Thank you