St. Petersburg State University of Aerospace Instrumentation

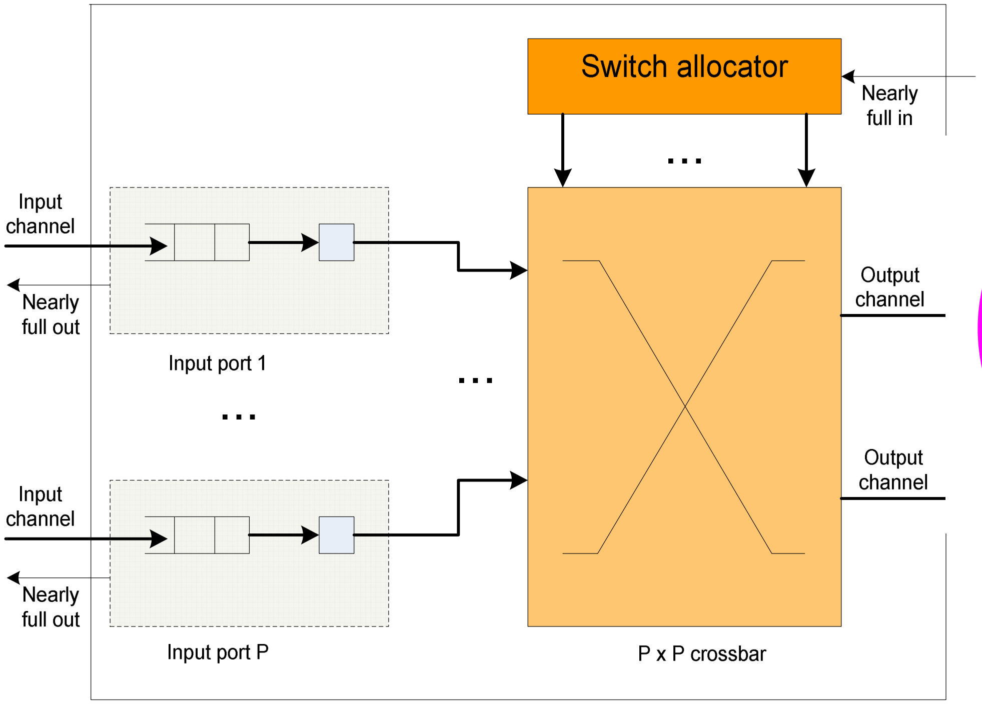
Institute of High-Performance Computer and Network Technologies

Low Power Protocols Development and Implementation

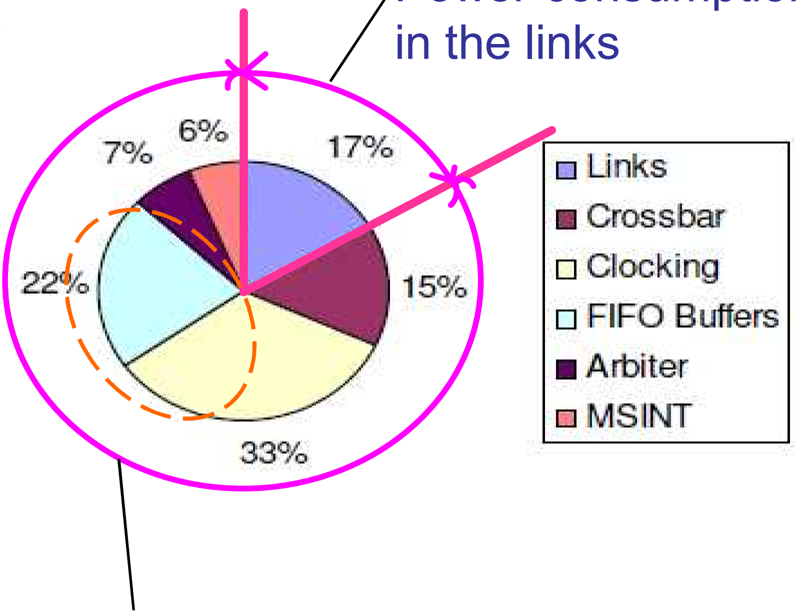
F.Shutenko, E.Suvorova, E.Yablokov,
felixshutenko@yandex.ru, suvorova@aanet.ru, kabal@yandex.ru
SUAI, B.Morskaya 67,
190000, Saint-Petersburg, Russia

Michel Gillet
Nokia Research Center
Helsinki , Finland
Michel.Gillet@Nokia.com

Typical ratio between power consumption of different components in a routing switch based networks-on-chip with packet switching



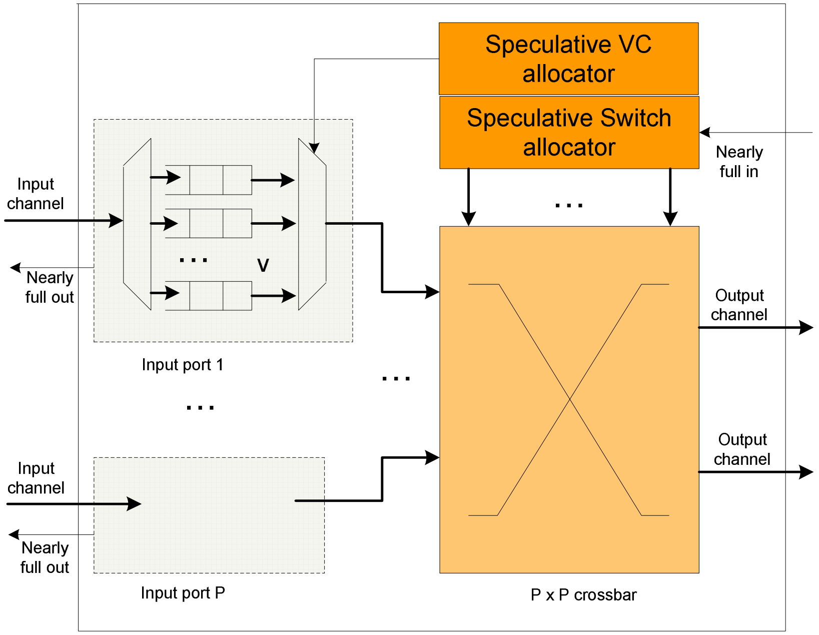
Typical structure of a routing switch



Power consumption in the links

Power consumption in a routing switch

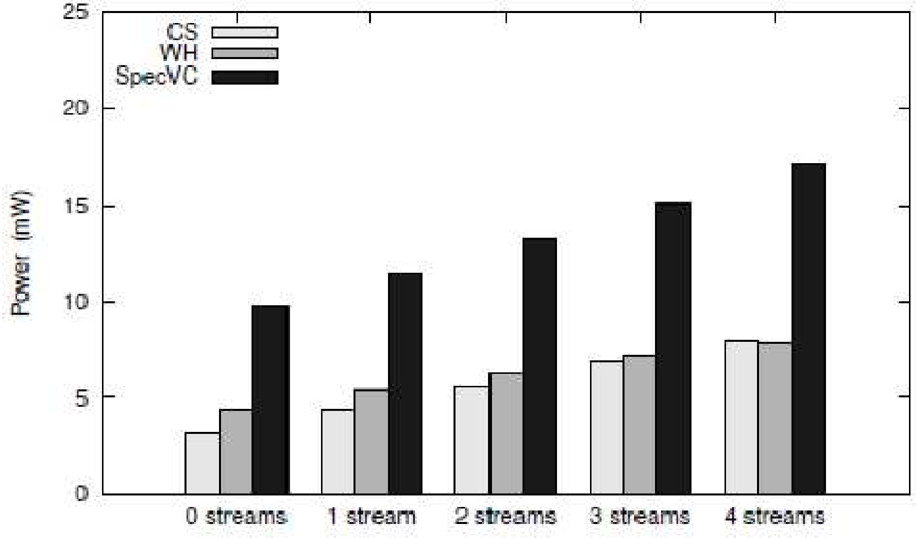
Routing switch power consumption comparison



Typical structure of routing switch with virtual channels

CS -without packet buffers (switching on-the-fly),

WH - with buffers that also work in on-the-fly mode,
SpecVC -with virtual channels (4 VCs for every port)



(b) Router power

0 Streams corresponds to the routing switch power consumption when no data are transferred
1 Stream – 4 Streams corresponds to the number of data flow sources (data flow of all sources are equal, with full throughput of one routing switch port each)

Dependency of routing switches power consumption on protocols

On example of power consumption of **SpaceWire** and **RapidIO** routing switches

SpaceWire features:

The using of buffers in input ports is determined by the data flow control in SpaceWire protocol.
Data flow control uses a credit scheme, to avoid data symbols lost when receiver has not enough place in buffer.

For the SpaceWire standard every node/switch input port should include a buffer with size not less than 8 data symbols (Nchars).

The main mode is switching-on-the-fly, but switching with buffering is allowed also.

The full packet buffering is not required and not possible because the maximal packet length is not specified.

SpaceWire routing switch features:

The transmission speed from 2 to 400Mbit/s.

Different ports of the switch can work with different speed.
Full buffering could be used for packet flows from low speed ports to high speed port multiplexing (and demultiplexing in oposite direction)

The the packet header consists of a sequence of addresses, one byte length each. The length of address sequence could be arbitrary.

Physical (address is equal to output port number), logical (uses the routing table) and regionally-logical (routing table and with first byte of header deletion).

The routing table with size of 256 strings is recommended for routing.

Number of addressed devices in a SpaceWire network is not constrained.

RapidIO features:

The main mode is switching with full packet buffering.

Switching without buffering is also allowed,
but in this case some packets could be lost because buffers in the receiver could be full.

Receiver can send to the sender the information about its buffers current state. But RapidIO standard does not include methods for synchronization between this notation flow and data flow from sender to receiver.

Packets Buffering in the transmitter is the main mechanism for packet loss avoiding

The maximum packet length (276 symbols) is defined in RapidIO standard for providing full packet buffering possibility

RapidIO features:

Four priority levels are defined for packets.

Four buffers with minimum size equal to one packet for every port of the switch is recommended for priority support in arbitration.

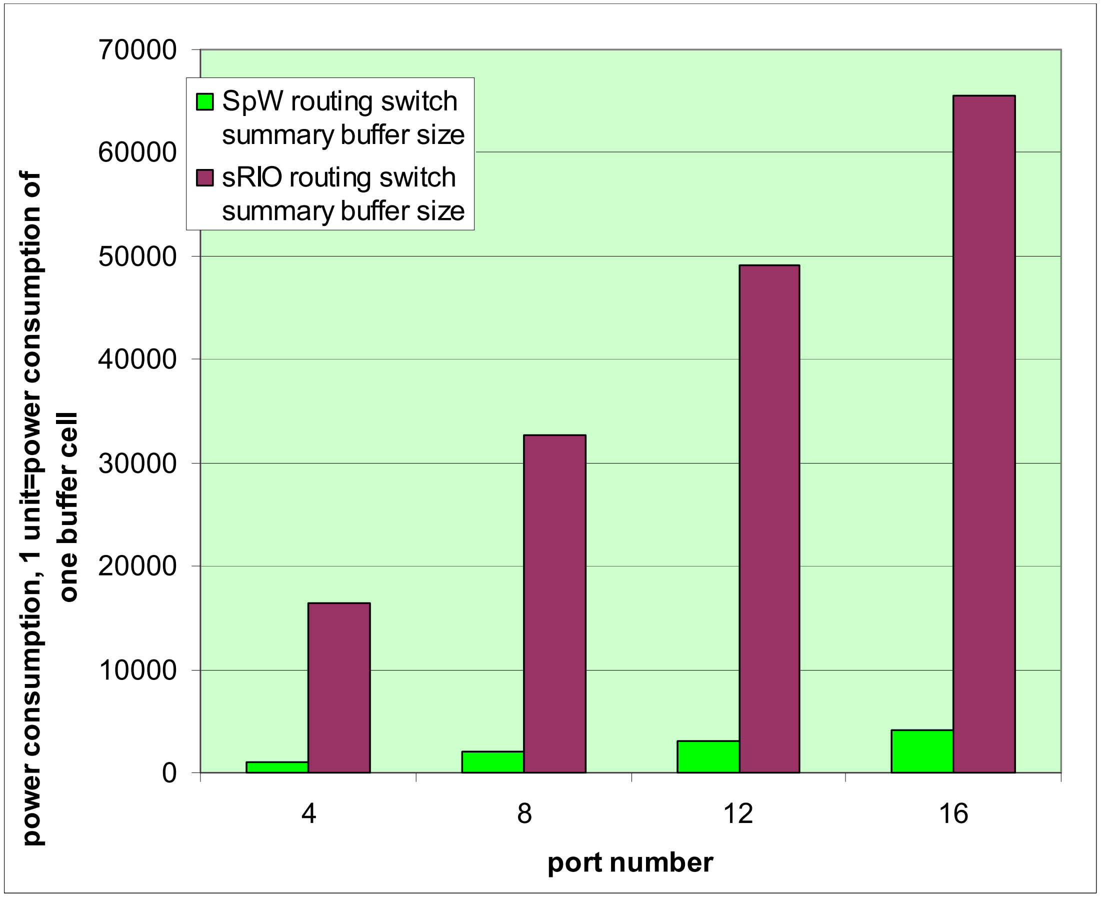
It allows forwarding of packets with biggest priority before packets with lower priority in the router.

In RIO protocol packet header includes Destination Address which length is 1 or 2 Bytes. Also up to 2^{16} logical addresses could exist in RIO network.

Only logical addressing is obviously defined in frame of RIO standard.

But separate routing table for every port of switch is allowed. This possibility could be used as functional analog of regional-logical addressing that is defined in SpW standard

Buffering scheme power consumption

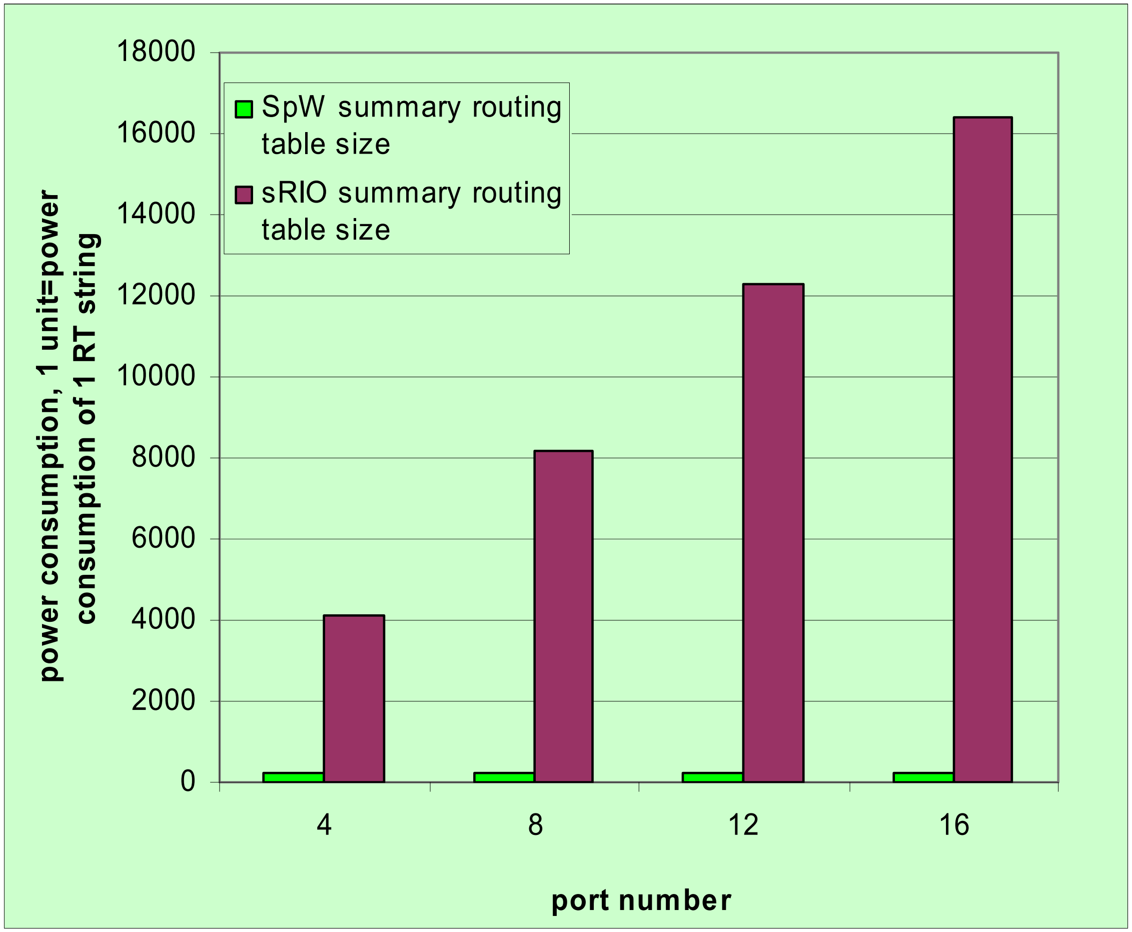


SpW: one buffer (256 NChars) for every port

RIO: a set of buffers
(min. four, 276 symbols each)
for every port

Power consumption is significantly larger when data flow control at packet level is used than when data flow control at symbol level is used

Routing tables power consumption



SpW: one routing table for the router (256 rows)

serial RIO:
one routing table for every router port (1024 rows each)

Power consumption in implementation of protocols that use switched routing with regional-logical addressing on packet headers is essentially less than in implementation of protocols that do not include this type of addressing

Power estimation by simulation

The idea:

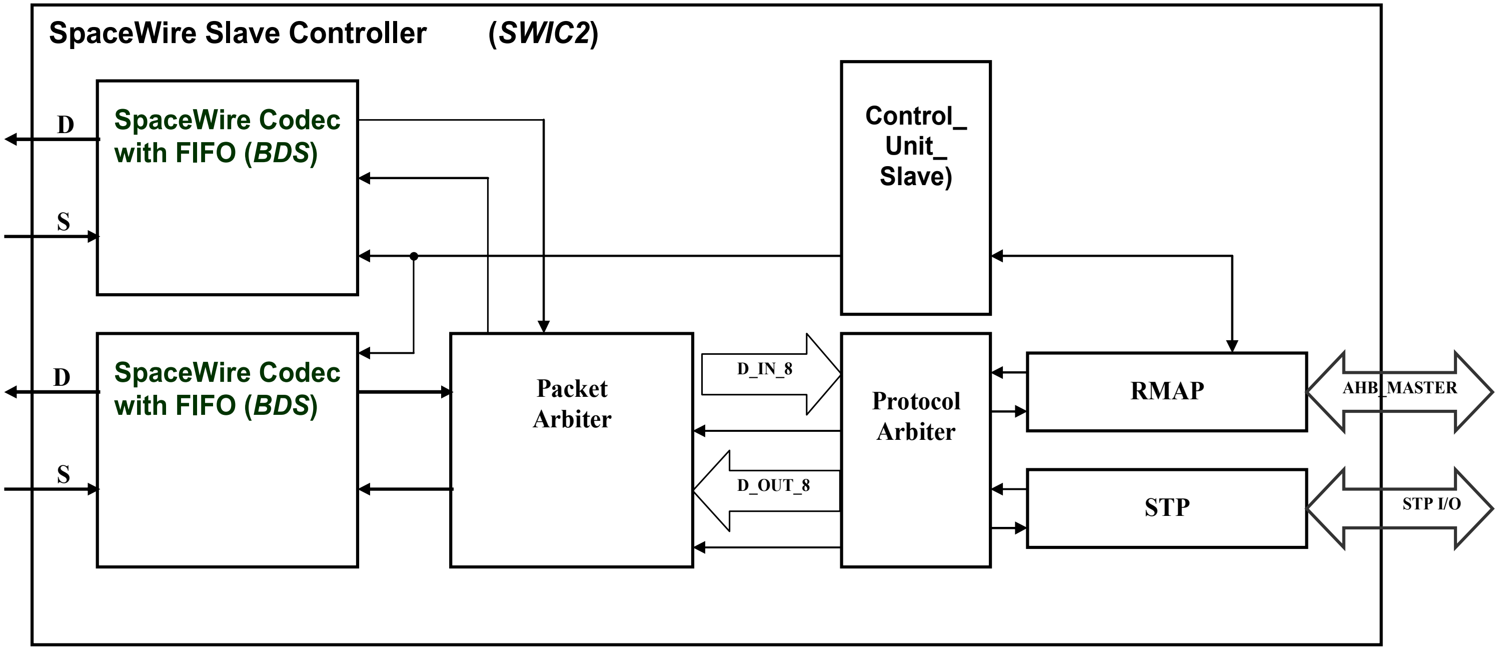
- To see dependence between parameters and features of the protocols (switching/routing modes, packet size, buffer size, etc.) and power consumption of two different data transfer protocols (RMAP and STP)
  on the post-implementation models

Simulation Features:

- using the same HW-implementation
  (RMAP and STP as two components),
- transmission the same set of data,
- fixed simulation time,
- fixed packet size,
- fixed packet send rate
- fixed channel speed,
- different data transfer protocols (RMAP and STP)

SWIC(SpaceWire)-controller

- Includes 2 SpW Codecs for interfacing an embedded SpaceWire network
- RMAP component compatible with ECSS-E-50-11 Draft F (for remote memory access from embedded application of other nodes
- STP component (Streaming Transport Protocol) for coherent regular data flows
- IRQ interface (Control Unit)
- Internal Control Unit (AHB slave, optional, 32-Bit bus).
  It gives possibility to configure the core itself via RMAP commands


SpaceWire Slave Controller (*SWIC2*)
D
S
SpaceWire Codec with FIFO (*BDS*)
D
S
SpaceWire Codec with FIFO (*BDS*)
Control_ Unit_ Slave)
Packet Arbiter
D_IN_8
D_OUT_8
Protocol Arbiter
RMAP
STP
AHB_MASTER
STP I/O

The timeline of RMAP host and slave system interface (example)

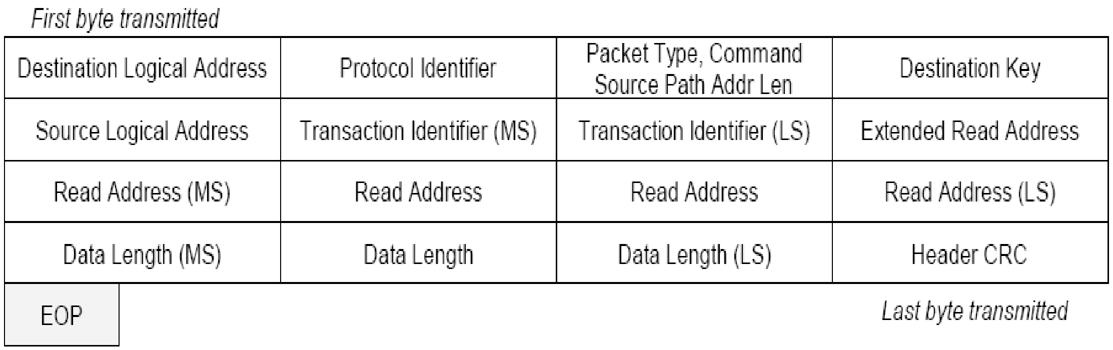
1) First, address 0x87 is applied to send data to channel 0
2) Second, address 0x64 will by applied to RMAP unit. RMAP unit, thus, have logical address of 0x64
3) Then, both ports 0x64 and 0x87 must by configured to transmit packets in specified manner. Port 0x64 will transfer packet to RMAP core without extracting Source Path Byte.
4) Port 0x87 will transfer packet to SpaceWire port0 with extracting the Source Path Byte.
5) Host system of RMAP protocol sends Read command to slave in order to get new portion of data from remote unit.
6) Host system starts counter to wait delay before reading next portion of data.
7) After counter reaches control value the algorithm repeats steps 5-7.

The timeline of STP host and slave system interface

1. Address 0x87 is applied to send data to channel 0
2. Address 0x23 will by applied to STP unit (its logical address is 0x23). Both ports 0x23 and 0x87 must by configured to transmit packets in specified manner.
   Port 0x23 will transfer packet to STP core without extracting Source Path Byte.
3. Port 0x87 will transfer packet to SpaceWire port0 with extracting the Source Path Byte.
4. The Host of STP protocol opens new connection with the slave.
5. The Host receives open-connection-acknowledge reply from the slave unit
6. STP Host system of the protocol runs data transfer from the slave by sending “infinite” credit to the slave.
7. Time by time, the host system receives packets with next portion of data from the STP slave unit
8. After specified time, host system closes the connection to stop the data transfer.

RMAP Packet Format

- Based on the packet transfer and packet ACK concept
- AHB bus memory access

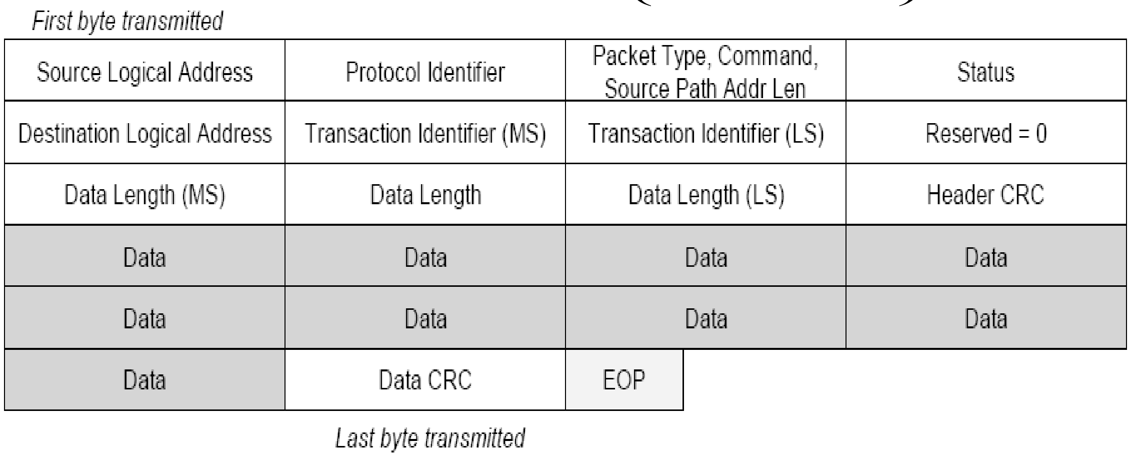


First byte transmitted

Destination Logical Address	Protocol Identifier	Packet Type, Command Source Path Addr Len	Destination Key
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Extended Read Address
Read Address (MS)	Read Address	Read Address	Read Address (LS)
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
EOP			

Last byte transmitted

RMAP Read command format (from HOST to SLAVE)



First byte transmitted

Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Reserved = 0
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
Data	Data	Data	Data
Data	Data	Data	Data
Data	Data CRC	EOP	

Last byte transmitted

RMAP Read Reply command (data from SLAVE to HOST)

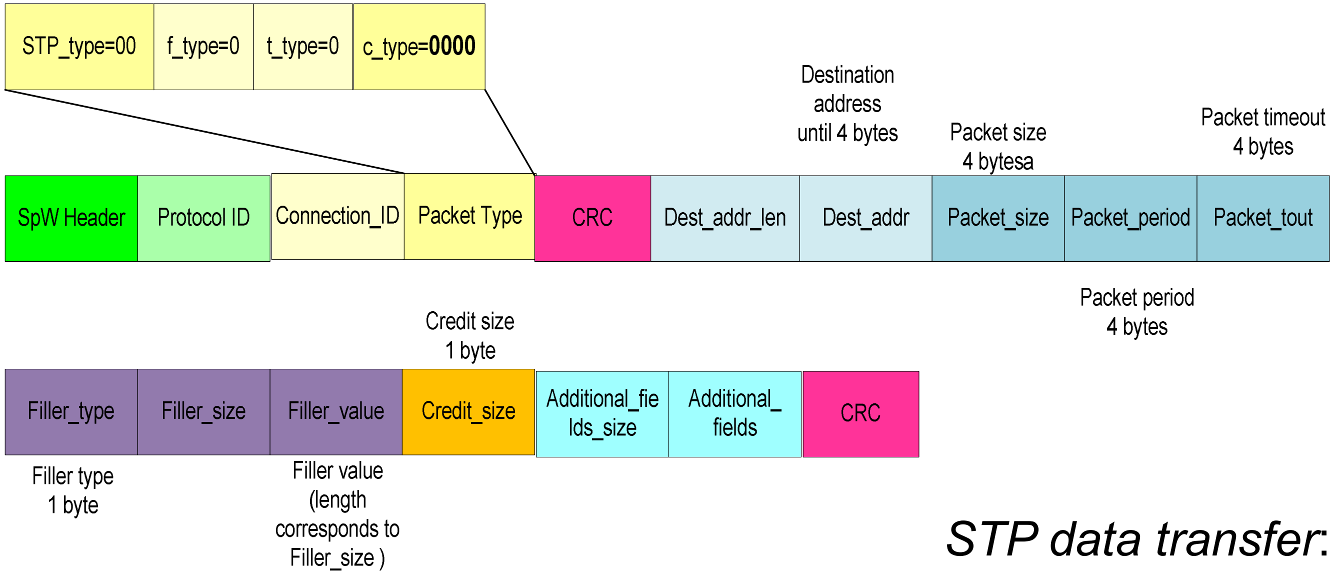
STP Packet Format

Based on concept of regular dataflow transfer

Examples:

- ADC data transfer
- FIFO data access

Set STP connection command:



STP_type=00 | f_type=0 | t_type=0 | c_type=**0000**

Destination address until 4 bytes

Packet size 4 bytesa

Packet timeout 4 bytes

SpW Header | Protocol ID | Connection_ID | Packet Type | CRC | Dest_addr_len | Dest_addr | Packet_size | Packet_period | Packet_tout

Packet period 4 bytes

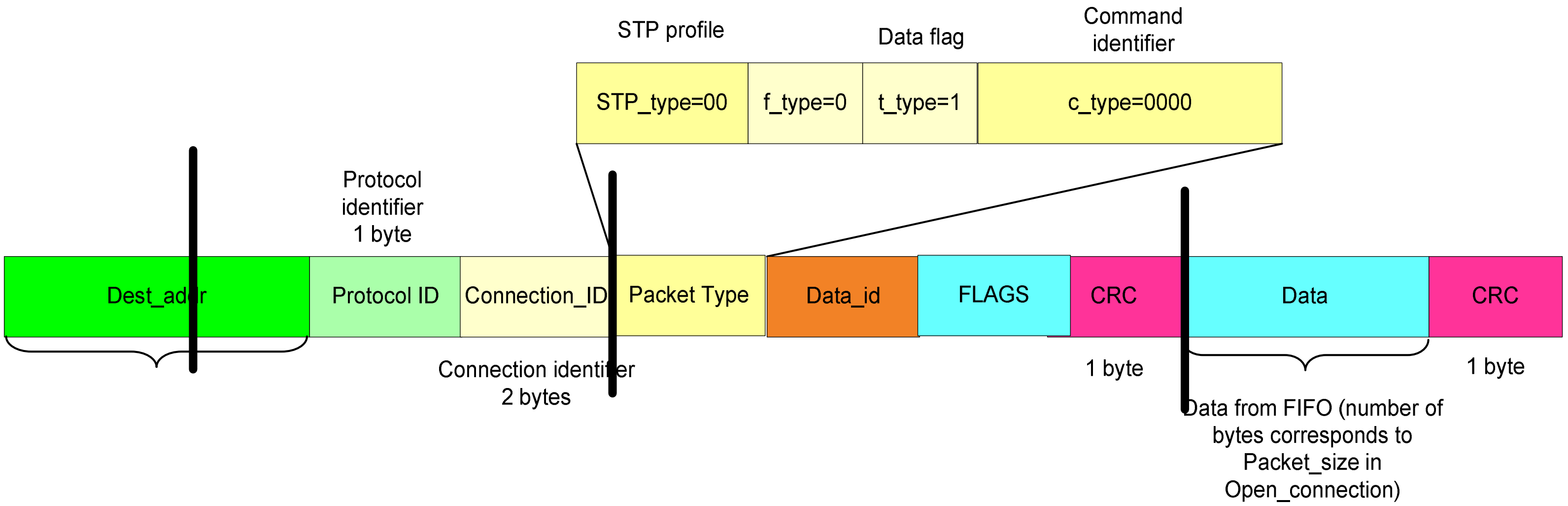
Credit size 1 byte

Filler_type | Filler_size | Filler_value | Credit_size | Additional_fields_size | Additional_ fields | CRC

Filler type 1 byte

Filler value (length corresponds to Filler_size)

STP data transfer:



STP profile

Data flag

Command identifier

STP_type=00 | f_type=0 | t_type=1 | c_type=0000

Protocol identifier 1 byte

Dest_addr | Protocol ID | Connection_ID | Packet Type | Data_id | FLAGS | CRC | Data | CRC

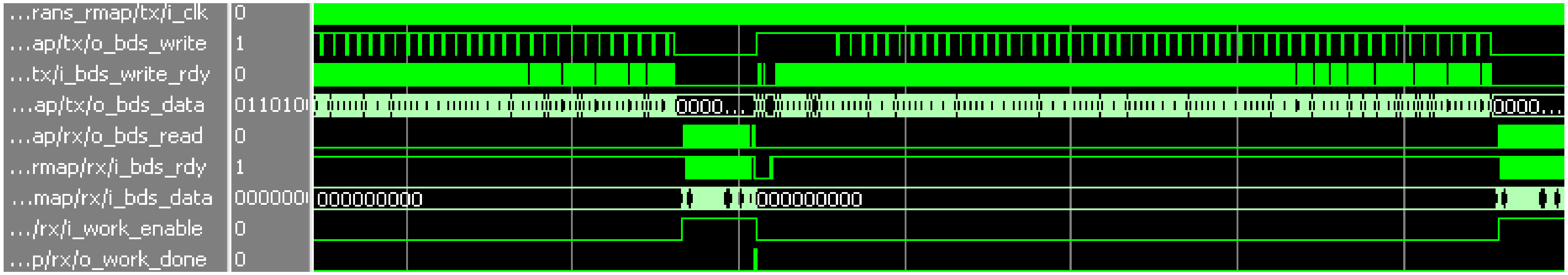
Connection identifier 2 bytes

1 byte

1 byte

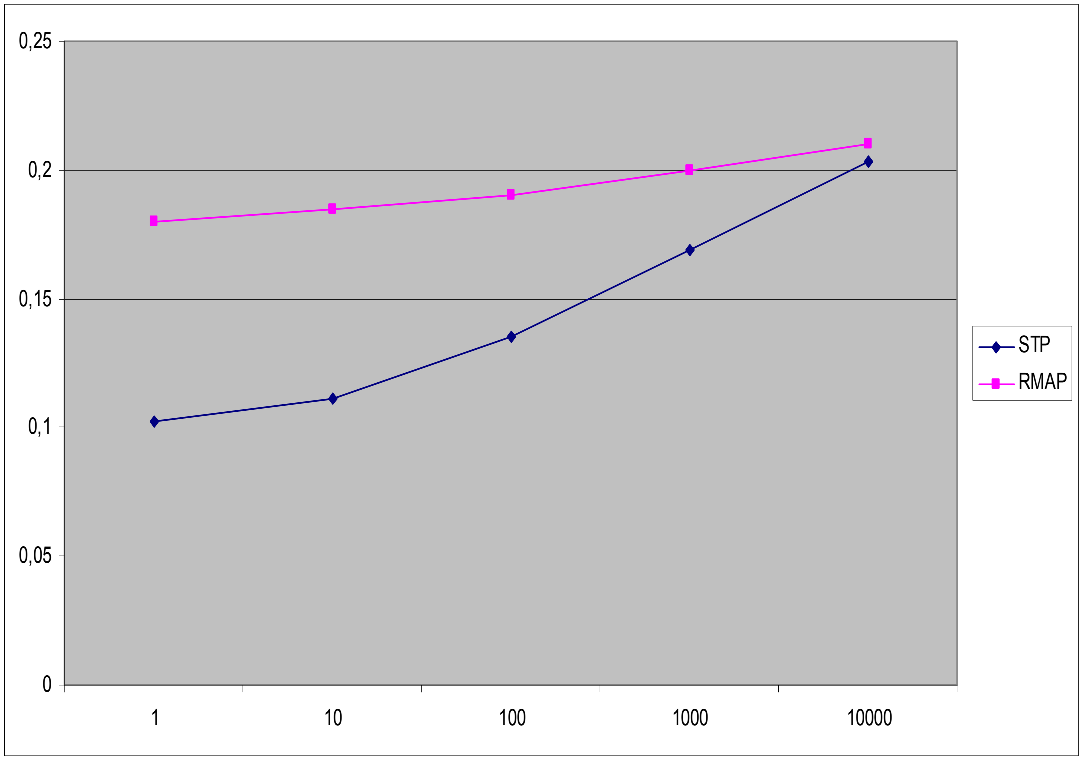
Data from FIFO (number of bytes corresponds to Packet_size in Open_connection)

Simulation Results



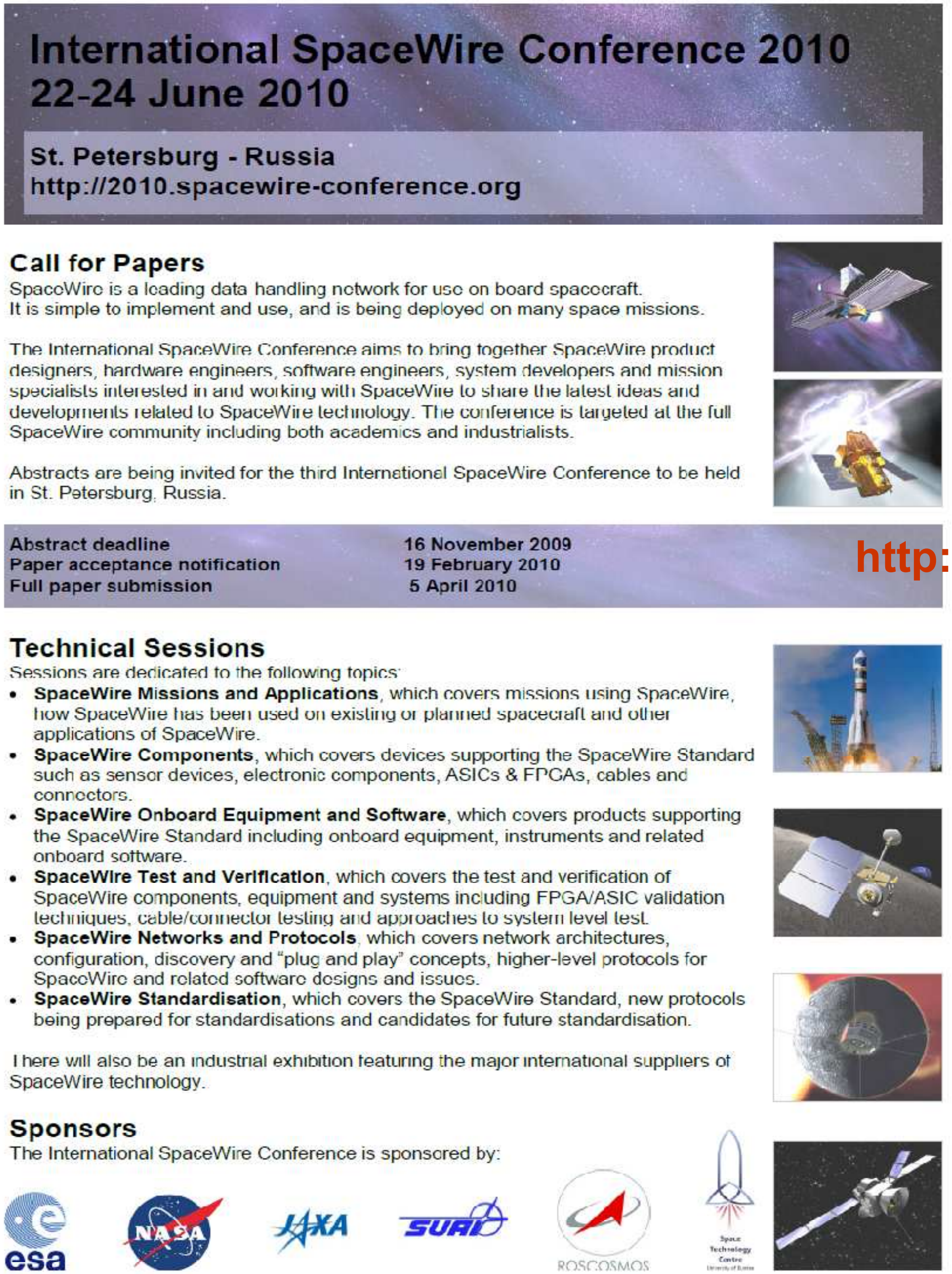
…rans_rmap/tx/i_clk	0
…ap/tx/o_bds_write	1
…tx/i_bds_write_rdy	0
…ap/tx/o_bds_data	0110100
…ap/rx/o_bds_read	0
…rmap/rx/i_bds_rdy	1
…map/rx/i_bds_data	0000000
…/rx/i_work_enable	0
…p/rx/o_work_done	0

- RMAP power is higher because of per-packet request and acknowledge RMAP mechanism
- Difference between RMAP and STP power is obvious on short packets



Conclusion

- We have presented an approach to power estimation for the network and transport layer protocols.
- At the network layer protocols it is done by estimating buffering/routing power consumption at the high level of routing switch conceptual model.
- At the transport layer protocols' power/energy characteristics comparison is done by the model of RMAP/STP protocols hardware implementation model.
- Further investigations require reasonable power models of architecture components implementations.
They could be developed in tight correlation with
SoC implementation technologies and tools



International SpaceWire Conference 2010
22-24 June 2010

St. Petersburg - Russia
http://2010.spacewire-conference.org

Call for Papers

SpaceWire is a leading data handling network for use on board spacecraft.
It is simple to implement and use, and is being deployed on many space missions.

The International SpaceWire Conference aims to bring together SpaceWire product designers, hardware engineers, software engineers, system developers and mission specialists interested in and working with SpaceWire to share the latest ideas and developments related to SpaceWire technology. The conference is targeted at the full SpaceWire community including both academics and industrialists.

Abstracts are being invited for the third International SpaceWire Conference to be held in St. Petersburg, Russia.

Abstract deadline	**16 November 2009**
Paper acceptance notification	**19 February 2010**
Full paper submission	**5 April 2010**

Technical Sessions

Sessions are dedicated to the following topics:

- **SpaceWire Missions and Applications**, which covers missions using SpaceWire, how SpaceWire has been used on existing or planned spacecraft and other applications of SpaceWire.
- **SpaceWire Components**, which covers devices supporting the SpaceWire Standard such as sensor devices, electronic components, ASICs & FPGAs, cables and connectors.
- **SpaceWire Onboard Equipment and Software**, which covers products supporting the SpaceWire Standard including onboard equipment, instruments and related onboard software.
- **SpaceWire Test and Verification**, which covers the test and verification of SpaceWire components, equipment and systems including FPGA/ASIC validation techniques, cable/connector testing and approaches to system level test.
- **SpaceWire Networks and Protocols**, which covers network architectures, configuration, discovery and "plug and play" concepts, higher-level protocols for SpaceWire and related software designs and issues.
- **SpaceWire Standardisation**, which covers the SpaceWire Standard, new protocols being prepared for standardisations and candidates for future standardisation.

There will also be an industrial exhibition featuring the major international suppliers of SpaceWire technology.

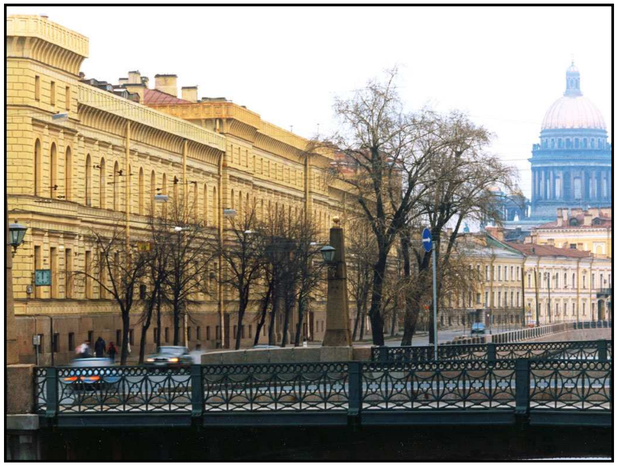
Sponsors

The International SpaceWire Conference is sponsored by:

http://2010.spacewire-conference.org

SUAI

Thank you !



Back up