Low Power Protocols Development and Implementation

F.Shutenko, E.Suvorova, E.Yablokov

SUAI
190000, B.Morskaya 67,
Saint_Petersburg, Russia
felixshutenko@yandex.ru, suvorova@aanet.ru,
kabal@yandex.ru

Michel Gillet

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

Abstract

Power consumption of network equipment essentially depends on data transfer protocols structure. The protocol determines network equipment (routing switches, interface controllers of terminal nodes) complexity and data transfer overhead. Also these factors affect to power consumption.

In this article we consider effect of some aspects Spacewire and RapidIO protocols to power consumption. We consider data flow control mechanisms in these protocols and evaluate effect of typical implementations of these mechanisms in routing switches to power consumption. We compare addressing schemes in SpaceWire and RapidIO and evaluate power consumption of its typical implementations

Also protocol structure affect to data transfer overhead, correspondingly it affect to power of data transfer. Correspondingly it affect to power that need for translation same size of data block with using of different protocols.

In this article we prepare comparison of two transport layers protocols – RMAP and STP – by describing our estimation model of HW implemented unit of slave SpaceWire controller which supports both RMAP and STP protocols. RMAP is protocol without set transport layer connection, and STP is protocol with transport connection. Also data transfer with using of STP protocol is accompanied with less overhead.

Index Terms: Power consumption, Interconnection protocols, routing, SpaceWire, RMAP, STP, RapidIO.

I. INTRODUCTION

The power consumption of network equipment essentially depends on data transfer protocol (protocols). Also power consumption could be dependent on special implementation features. Therefore in this article we consider mainly typical implementations of network devices.

Typical structure of routing switch is represented on Figure 1 [1].

This diagram shows that power consumption in interconnection lines practically in 5 times less than power consumption of routing switches.

For other types of interconnection networks this ratio could be changed but the difference is not essential. Also in this article we consider power consumption in routing switches and interface controllers of terminal nodes.

Power consumption of routing switches distributed in the following way. Biggest power consumption corresponds to Clocking group. This group includes power consumption of all sequential elements of network-on-chip components.

The typical ratio between power consumption of different components of typical routing switch based networks-on-chip with packet switching is represented on Figure 22 [2].

Next is power consumption of buffers and then power consumption of crossbar switch.

The power consumption of buffers strongly depends on its size and usage conditions (in this case the buffer of average statistical size for most networks-on-chip – 4-flits). The power consumption of

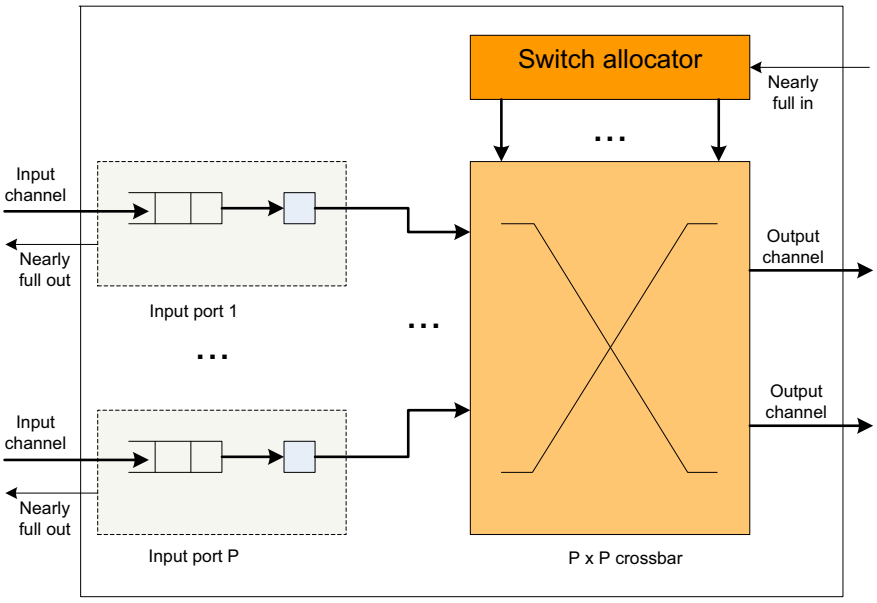


Figure 1 – Typical structure of routing switch

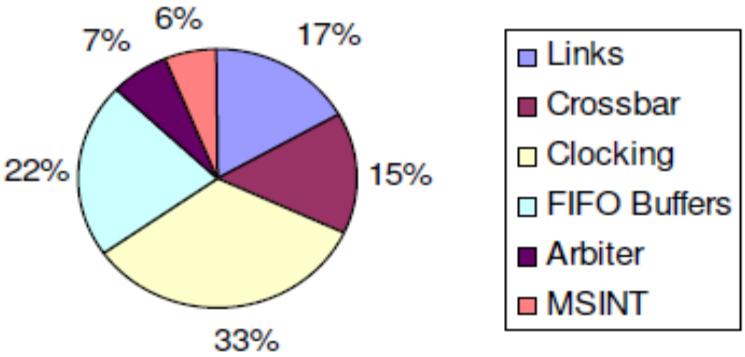


Figure 2 – ratio between power consumption on different components of network-on-chip

bigger buffers (that could be need, for example, for full buffering or for virtual channels implementation) could be essentially bigger.

Buffer size could be differing dependently on routing type (routing with buffering or routing "on-the-fly"). If routing with buffering is used then buffer size could be differ dependently on packet sizes and buffering schemes. In some cases buffers could not be used.

In some cases output ports also could include buffers. Necessity of this feature is determined by data flow control organization on transport layer and also these buffers are used for data transmission speed coordination if protocol allowed different transmission speeds.

Typical structure of routing switch with virtual channels is represented on Figure 3 [2].

Let's consider dependency between power consumption and availability of buffers and virtual channels.

On Figure 44 represented dependency of power consumption from routing type (the four-ports router is considered) [1,2,3].

On this figure CS corresponds to routing switch without buffers (switching on-the-fly), WH corresponds to routing switch with buffers that also worked in on-the-fly regime, SpecVC corresponds to routing switch with virtual channels (four virtual channels to every port)

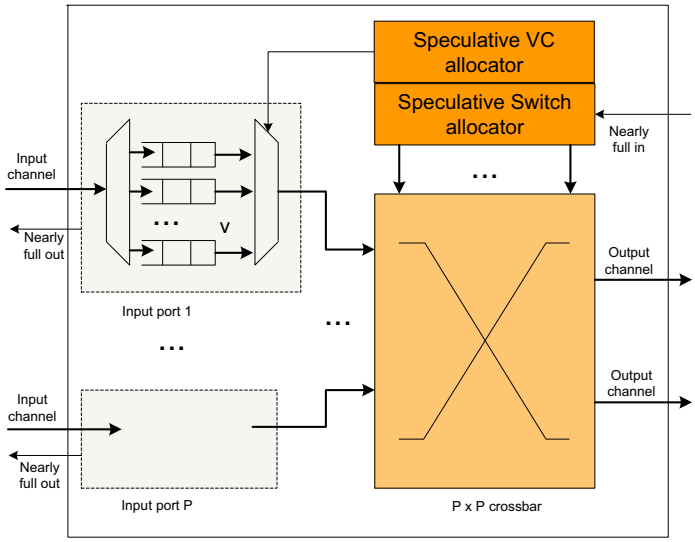
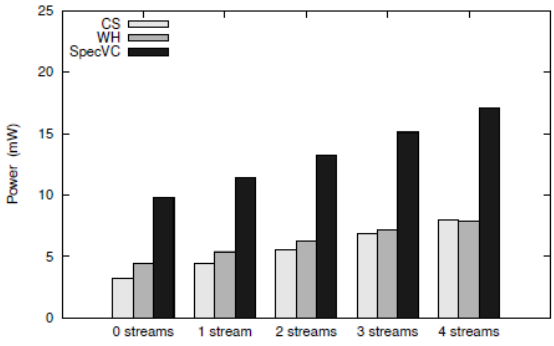


Figure 3 - Typical structure of routing switch with virtual channels

0 Streams corresponds to power consumption of routing switch when data not transferred

1 Streams – 4 Streams corresponds to the number of data flow sources (data flow of all sources is equal and corresponds the throughput of one port of the routing switch)

The maximal power consumption corresponds to routing switch with virtual channels. It practically not dependent on summary data flow from all sources because need for implementation of virtual channels we need essentially more memory than for implementation switching on the fly.



(b) Router power

Figure 4 - Router power consumption

Power consumption of routing switch without buffers and routing switch with buffers (the relatively small buffers are used). When the of summary data packet flow growing the power

consumption of these types of routing switches will be practically equal because change of ratio of memory block's and combination schemes power consumption.

The power consumption of arbiter is small relatively to others components. But in considered research supposed that the destination address includes obvious number of output port and power consumption for routing not exists.

Two arbitration methods those are most popular for networks-on-chip are considered – matrix based arbitration (with static priorities) and arbitration with cyclic dynamic priorities. The power consumption for more complex arbitration methods will be grow not essentially in comparison with the power consumption of other router components.

If switch is routing switch (the number of output port is defined as function of destination address of packet and, maybe, current state of network if adaptive routing is used), then routing affect to power consumption. Power consumption dependent on routing method (table based routing or functional based routing), on header structure (number of operations that need to selection of destination address from header). Routing could be implemented on base of routing tables. In this case power consumption depends on memory block size (of number of registers) that need for routing table implementation.

Routing could be implemented with using of combinational scheme that implements routing function. Typically in this case power consumption of routing on tens times less than when routing table is used. But this method usually could be used only for networks with regular or practically topology. Also the routing function complexity and possibility/impossibility of this function development strongly dependent on address methods defined on concrete standard and possible number of devices in network.

II. DEPENDENCY OF ROUTING SWITCHES POWER CONSUMPTION ON PROTOCOLS

Let's consider power consumption of SpaceWire and RapidIO routing switches. Both protocols obviously not includes virtual channels mechanisms.

The using of buffers in input ports determines by data flow control in SpaceWire protocol.

For data flow control is used credit scheme correspondingly that receiver send to transmitter one credit symbol when in its buffer exists the place for 8 data symbols (one data symbol corresponds to 1 byte). The sender could send as many symbols as was credited (not more than 56 symbols could be credited). This crediting scheme is oriented to data symbols lost avoiding when receiver has not enough place in buffer.

Correspondingly SpW standard every input port could be included buffer that size is not less than 8 data symbols (nchars). The main regime is switching on the fly but switching with buffering is allowed. The full buffering is not possible because the maximal packet length is not defined in SpW standard.

The transmission speed from 2 to 400Mbit/s is allowable is frame of SpW standard. Also if different ports of switch could be work with different speed then full buffering could be used for packet flows for low speeds ports to high speed port multiplexing (and demultiplexing in back direction) if for concrete application the maximal allowable size of packet is defined.

In SpW protocol the header consists of the sequence of addresses, each of them is one Byte length. The length of address sequence could be arbitrary. In frame of standard physical (address is equal to output port number), logical and regionally-logical (with deletion of first byte of header).

The number of addressed devices in SpW network is not constrained if physical or combination of logical and regionally-logical addressing is used. The routing table with size 256 strings is recommended for routing [1].

The main regime in protocol RIO is switching with full packet buffering. Switching without buffering is also allowed, but in this case some packets could be lost because buffers in receiver are full. Receiver could send to sender the information about its buffers current state. But RIO standard not includes methods for synchronization between this notation flow and data flow from sender to receiver.

Also the main mechanism for avoiding of packet losing because of not enough place in receiver buffer is full buffering of packets in transmitter. The packet is kept in transmitter's buffer until receiver confirm that it is successfully received. The maximal packet length (that is 276 symbols) is defined in RIO standard for providing of full packet buffering possibility.

The four priority levels are defined for packets in RIO standard. Also as minimum four buffers with size equal to one packet for every port of switch is recommended for priority support on arbitration level. This allow translation 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 until 2^{16} logical addresses could be exists 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 [1].

For implementation of buffering scheme that is used correspondingly RIO standard in general case needs more memory than for SpW buffering scheme implementation. Also power consumption of RIO buffering scheme is significantly large than SpW buffering. Thus we can drawn a conclussion that power consumption is significantly large when data flow control on packet level is used than when data flow control on symbol level is used.

Selection of output port number with using of routing table is recommended in frame of both standards. In SpaceWire routing switch one routing table with size 256 strings is used.

One routing table per port is recommended in frame of RIO standard for routing switch. The size of routing table could be until 2^{16} strings, but recommended size is 1024 strings because of using separate routing table for every port of routing switch.

Also power consumption of RIO switch's routing tables dependently on port's number and real routing table's size could be in tens times more than power consumption of SpW switch's routing table.

Also we drawn a conclusion that power consumption of implementation of protocols that includes regional-logical addressing on packet headers essentially less than of implementation of protocols that not includes this type of addressing obviously.

If protocol not includes this type of addressing obviously and need provide networks with big number of devices then for its implementation needs very big routing tables in routing switches.

On a higher levels of OSI protocol the specific features of transferring data can be used to design specific protocols with adaptation on the data conception. This can have influence on power consumption.

We are planning to make an estimation for two different protocols of transport layer.

III. DEPENDENCY OF POWER CONSUMPTION ON SPECIFIC FEATURES OF TRANSPORT PROTOCOLS

(aspects of power estimation model)

The idea of power estimation on transport layer is to transmit the same set of data using different data transfer protocols on the same HW-implementation of low levels and to see dependence between common parameters of the protocols (like packet size, buffer size, etc) and power consumption of the overall system. We use protocols RMAP and STP to show the results of the power estimations. Both protocols are designed for different strategies of data transmission via the network. The idea of RMAP is based on conception of request data transfer. This means that host system must send read or write command to remote slave component in order to get access to required data. Thus, to get data periodically after some delay between every portion of data, host system must send read-command to the slave once and once again. This is different from the idea of STP protocol. Streaming Protocol (STP) is developed to send periodical data consequently time by time after specified delay. When using STP, host system must just open new connection and run the transfer. Slave system itself determines the moments of time to produce next portion of data to the host.

Thus, transferring the same set of data in a different manner (using RMAP and STP algorithms), we can compare power consuming by implementations of protocols which are designed for different applications.

The scheme of boths algorithms can be seen on Figure 5. The timeline of dataflow between STP host and slave system is following.

1. First, both STP host systems configure slave unit to work with host command. In the designed IP-Core initialization can be done only using RMAP protocol. First, address 0x87 is applied to send data to channel 0

2. Second, address 0x23 will by applied to STP unit. STP unit, thus, have logical address of 0x23

3. Then, 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.

4. Port 0x87 will transfer packet to SpaceWire port0 with extracting the Source Path Byte.

5. Host system of STP protocol opens new connection with slave.

6. Host system receives open-connection-acknowledge reply from slave unit

7. Host system of STP protocol runs data transfer from slave by sending “infinite” credit to the slave.

8. Time by time, host system receives packets with next portion of data from STP slave unit

After specified time, host syatem closes the connection to stop the data transfer.

The timeline of dataflow between RMAP host and slave system is following.

1. First, both RMAP host systems configure slave unit to work with host command. First, address 0x87 is applied to send data to channel 0

Second, address 0x64 will by applied to RMAP unit. RMAP unit, thus, have logical address of 0x64

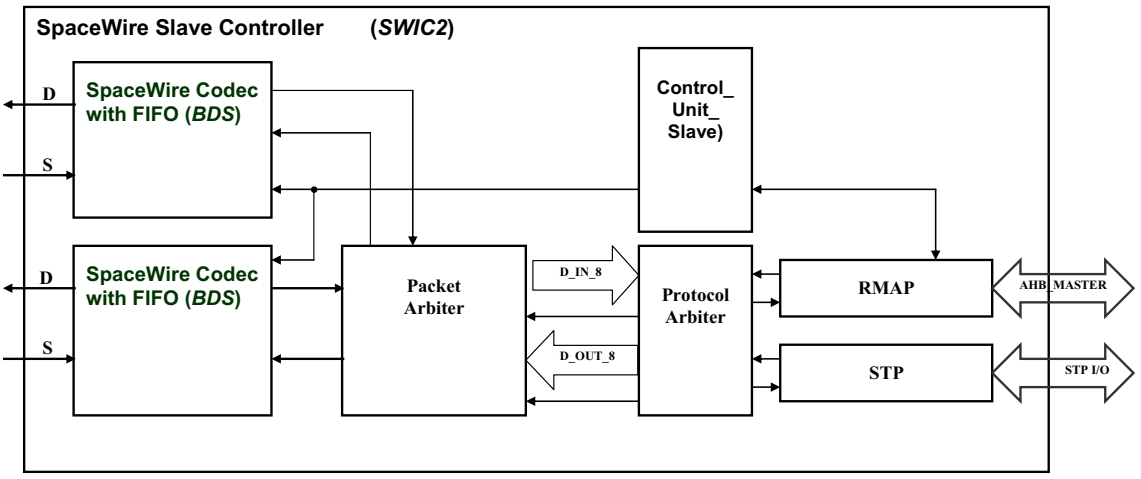


Figure 5 – data flows in the RMAP/STP system

2. 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.

3. Port 0x87 will transfer packet to SpaceWire port0 with extracting the Source Path Byte.

4. Host system of RMAP protocol send read command to slave in order to get new portion of data from remote unit.

5. Host system starts counter to wait delay before reading next portion of data.

6. After counter reaches control value the algorithm repeats steps 5-7.

As can be seen from the Figure 5, we use the same system to estimate power of RMAP and STP protocols. This means that the same implementation of SpaceWire components is used. Thus, the power consumption of data exchange depends on specific features of STP and RMAP protocols and on its implementation in the chip only. The same data will be transferred through the pipeline between remote memory dump (on left side in the figure, not shown) and host system. Thus, power consumption depends on packet format also.

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

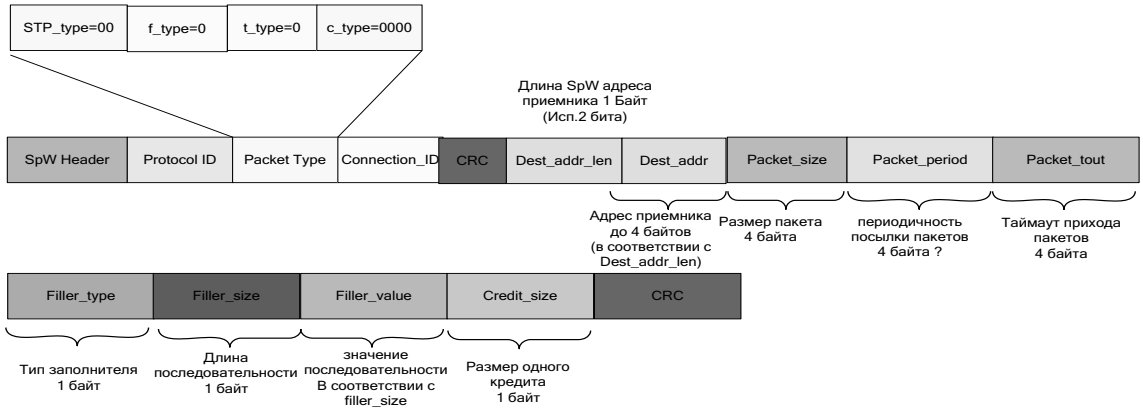
RMAP Read Reply command (data from SLAVE to HOST)

Figure 6 - Read command format and read reply format of RMAP protocol

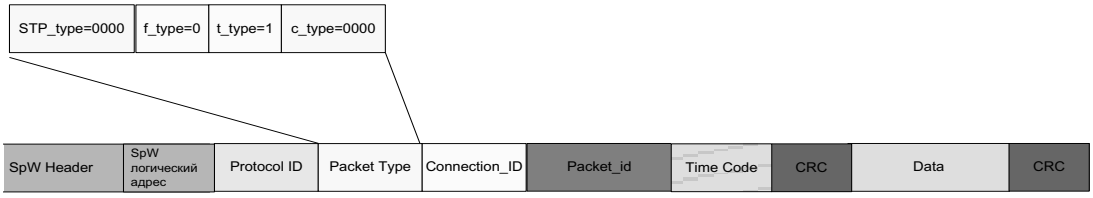
The Figures 6 and 7 shows formats of RMAP and STP packets which are used in the estimated model.

To estimate power of the chosen protocols we use post synthesis model of The SWIC slave controller, supporting both RMAP and STP protocols. The SWIC(SpaceWire)-controller is an example of HW-implemented core of Switch controller (node of embedded network system). The controller

- Include 2 SpW Codecs, to enable insertion the core into an embedded SpaceWire network
- RMAP component compatible with ECSS-E-50-11 Draft F (do not support Read-Modify-Write command), to enable remote memory access from embedded application of other nodes
- STP component (Streaming Transport Protocol)
- IRQ interface (Control Unit)
- Internal Control Unit (AHB slave, optional, 32-Bit bus). This gives possibility to configure the core via RMAP commands



STP connection setup command (STP-host)



Connection data transfer packet (STP-slave)

Figure 7 – Connection Setup and Data Transfer commands format of STP protocol

RMAP component of the node supports read and write commands only, and it responsible to send wrire-reply and read-reply packets only. Thus, RMAP component is slave unit.

The HW module can be implemented as FPGE IP-Core (Spartan or Virtex family) and it has the following characteristics

- 1300 slices (2400 LUTs), thus taking 26% of 3s500evq100-4 Spartan-3 device
- 65 Mhz local clock of 3s500evq100-4 Spartan-3 device

We use Xilinx XPower application to estimate power of the system.

The work on estimation model is in progress.

VI. CONCLUSION

In this article we consider affect of data flow control schemes in SapceWire and RapidIO protocols to power consumption of routing switches. We show that power consumption of RapidIO data flow control scheme implementation is essentially bigger, because full packet buffering need for it.

We consider devices addressing schemes in SpaceWire and RapidIO protocols. We show that typical implementation of RapidIO addressing scheme has essentially bigger power consumption than implementation of SpaceWire addressing scheme.

We propose our approach to estimation power on transport layer by presenting the model of RMAP/STP protocols power estimation model. It bases on implementing of two different protocols on the hardware implementation model with the common lower protocol layers. We are planning to show the differences (in terms of power consumption) between specific features of transport layer implementation of these protocols.

REFERENCES

[1] W. J. Dally, B. Towles. Principles and practices of interconnection networks. Elsevier 2004, 550p.

[2] Andrew Kahng, Bin Li, Li-Shiuan Peh and Kambiz Samadi ORION 2.0: A Fast and Accurate NoC Power and Area Model for Early-Stage Design Space Exploration CS2008-0929
September 30, 2008

[3] A. Banerjee, R. Mullins and S. Moore, “A Power and Energy Exploration of Network-on-Chip Architectures”, *Proc. NoCs*, 2007, pp. 163-172

[4] N. Banerjee, P. Vellanki, and K. S. Chatha. A power and performance model for network-on-chip architectures. In DATE ’04: Proceedings of the conference on Design, automation and test in Europe, page 21250, Washington, DC, USA, 2004. IEEE Computer Society

[5] A. Mello, L. Tedesco, N. Calazans, and F. Moraes. Virtual channels in networks on chip: Implementation and evaluation on hermes NoC. In SBCCI ’05: Proceedings of the 18th annual symposium on Integrated circuits and system design, pages 178–183, New York, NY, USA, 2005. ACM Press

[6] ECSS E50 12A, “SpaceWire. Links, nodes, routers and networks”, 31 July 2008

[7] RapidIO™ Interconnect Specification Part 6: 1x/4x LP-Serial Physical Layer Specification, Rev. 1.3. - June 2005

[8] ESA, specification RMAP, ECSS-E-50-11 Draft F 4th December SpaceWire Remote Memory Access Protocol, Steve Parkes and Chris McClements, University of Dundee, Applied Computing, Dundee, DD1 4HN, Scotland, UK.