SpaceWire Network Performance Evaluation During the Streaming STP Data Transmition

Ilya Korobkov
State University of Aerospace Instrumentation
Saint-Petersburg, Russia
ilya.korobkov.l@gmail.com

Abstract

Streaming data transmission is one of the most important questions for the spacecraft developers. The research of such kind of SpaceWire network with streaming STP data transfer could give an important information for the Network Administrator, which would make the process of the network configuration much easier. The number of successfully transmitted STP data packets and the number of slave nodes with various characteristics of the network are the interesting parameters for the research.

This paper gives an overview of estimation of STP transport protocol simulation over the SpaceWire SystemC model activity and shares the given results.

Index Terms: Performance evaluation, simulation, queueing theory, data streaming protocol, STP, SpaceWire Network.

I. Introduction

During the development process of the network equipment operating with a communication protocol it's crucial to perform a mathematical evaluation of suitability of the protocol.

This paper represents a theoretical calculation and performance of network model based on spacecraft embedded SpaceWire [2] Standard. To check mathematical results network and application models have been developed.

The overview of STP simulation results is presented in this paper.

II. STP Modelling Project Overview

Our team had a task [1] to implement a model for the simultaneous operation of multiple space applications over the SpaceWire Network for streaming data transfer by means of the STP [3] protocol. For this purposes we developed a model of application *STP_App* and a model of network *STP_Network* in SystemC modeling language.

A. *STP_App SystemC application model*

STP_App is the model of host- and slave-application. It provides specified control of each connection during the simulation time. *STP_App* allows simulating of the network load by generating data according to the opened connection parameters:

– total number of data to transfer per connection;
– size of data;

With the financial support of the Ministry of Education and Science of the Russian Federation

– frequency of data generation.

The structure of *STP_App* model is shown at the Fig. 1.

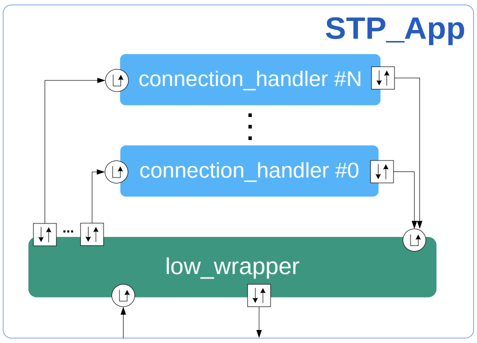

Fig. 1. STP Application SystemC model architectural diagram

The model of *STP_App* application consists of several blocks:

- *stp_app* – the main block which is in order for creation, initialization and binding of all subblocks;
- *connection_handler* controls the logical operation of each specific STP connection during all time of modeling. There are two operation modes: host and slave. In host mode it sends control requests to *STP* model: open/close connection, start/stop data transfer, send a credit. In slave mode it sends requests to confirm or not to confirm establishment of connection and it is responsible for generation of data according to the parameters specified during the connection establishment phase;
- *low_wrapper* passes the arriving requests/notification messages from *STP* model to *connection_handler*.

B. STP_Network SystemC network model

The model of network *STP_Network* is developed in SystemC language for the network topology simulation. *STP_Network* represents the black box with a number of inputs and outputs. The model intends to simulate a transmission of STP packets through the network, time delay of packet transmission, channel bit error rate.

The structure of the *STP_Network* model is demonstrated at the Fig. 2.

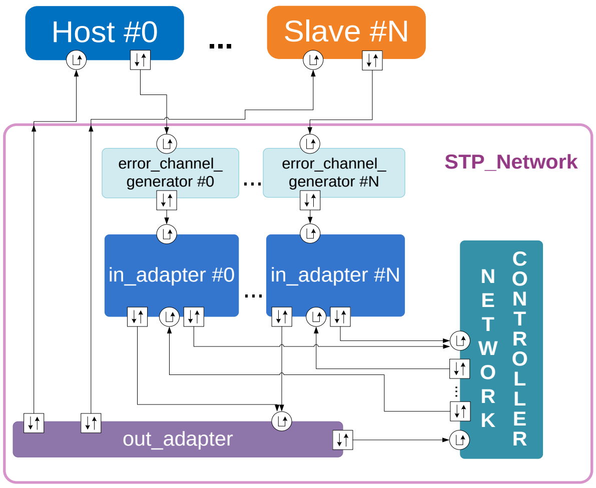

Fig. 2. STP Network SystemC model architectural diagram

The *STP_network* model consists of following blocks:

- *stp_network* - the main block, creates, initializes and binds all subblocks;
- *error_channel_generator* –responsible for bit error rate;
- *in_adapter* - stores the incoming STP packet;
- *out_adapter* - performs packet switching to the appropriate output port of the network model;
- *network_controller* – logical management of *in_adapter* and *out_adapter*, routing and time delay of STP packets.

III. Performance evaluation of SpaceWire network

Developers of the SpaceWire Network, which operates streaming data transfer by means of STP protocol, are interested in the information about packet loss probability and a number of slave nodes with following various network characteristics:

1. Data transmission speed (speed of SpaceWire channel: 2-400 Mbit/s);
2. Network load (e.g: from 50% to 100%);
3. Bit error rate (SpaceWire standard specifies BER = 10^{-12});
4. Number of slave-applications in the network (STP protocol specifies the maximum number of slave-applications for a single host-application: 65534, it is provided when the one connection is established with each slave application);
5. The size of data payload (STP protocol: 4 - 2^{32} bytes);
6. The size of the input buffers in network switching equipment (depends on implementation)

A. The Calculation of packet loss probability

The simulator of STP protocol for the SpaceWire Network presents the queue system with a elementary input stream and a limited amount of buffer. The structural diagram is shown at the Fig. 3.

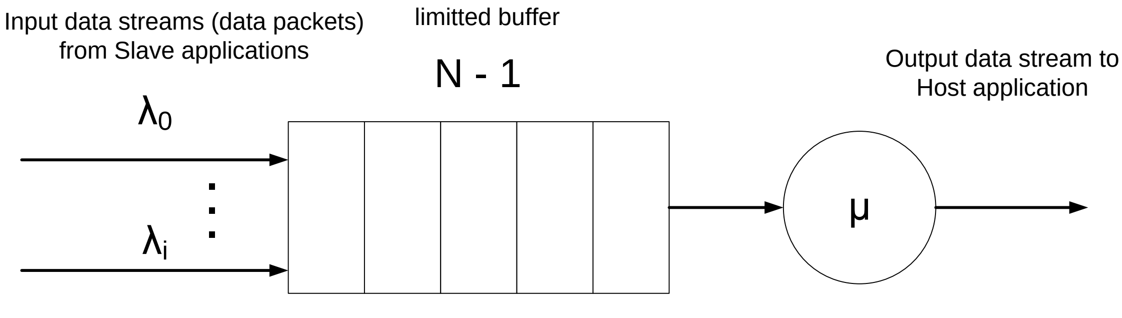

Fig. 3. STP Queue system structural diagram

Let the maximum number of data packets from Slave applications be equal N (including the one that is inside the *STP_network*). The $N+1$ incoming packet is considered as lost. Also, we assume that the time intervals between arrivals of data packets are distributed according to the Poisson law with parameter λ. The time service is represented by μ parameter of exponential law. Then by means of the Erlang formula for systems with a limited length of queue we can calculate the probability of K packets being in the queue system [7]:

$$p_{k=n+s} = \frac{\frac{Z^n}{n!}\left(\frac{Z}{n}\right)^s}{\sum_{j=0}^{n}\frac{Z^j}{j!} + \frac{Z^n}{n!}\sum_{j=1}^{N-1}\left(\frac{Z}{n}\right)^j}\ , \qquad (1)$$

where $Z = \frac{\lambda}{\mu}$; n – number of process units; s – amount of data packets in a buffer.

We have just one host-application (n = 1), so the formula looks like this:

$$p_{k=1+s} = \frac{Z^k}{1 + Z + Z\sum_{j=1}^{N-1} Z^j} = \frac{Z^k}{1 + \sum_{j=1}^{N} Z^j} = \frac{Z^k}{\left(\frac{1 - Z^{N+1}}{1 - Z}\right)} = Z^k \frac{1 - Z}{1 - Z^{N+1}},$$

$$0 \le k \le N \qquad (2)$$

$p_k = 0$, when $k > N$

The formula (2) allows calculating the probability of packet loss: the packet will be dropped if the two events occur: the system got N packets and N+1 packet comes to the input. Probability of the former is:

$$p_{k=N} = \frac{Z^N(1 - Z)}{1 - Z^{N+1}} \qquad (3)$$

And the latter's probability is:

$$p^{(N+1)} = \lim_{S\to\infty} \frac{S - N}{S} = 1\ , \qquad (4)$$

So the probability of packet loss is calculated:

$$p_B = p_N p^{(N+1)} = \frac{Z^N(1 - Z)}{1 - Z^{N+1}}\ , \qquad (5)$$

Thus the formula (5) permits to calculate the probability of packet loss. Moreover it's possible to get quantity of dropped packets per time unit by:

$$p_B = \frac{R}{\lambda} => R = p_B\lambda\ , \qquad (6)$$

where *R* is the intensity of dropped packets per time unit.

C. The given calculated results

For performance evaluation of the SpaceWire Network the following parameters of network have been used:

- Channel speed of data transmission = 40 Mbit/s;
- Network loading = 60 %;
- Size of data payload = 1014 byte;
- Total time of data_packet transfer = 60 ms
- Buffer's size varies from 1 to 128 data packets. Full data packet size is 1024 Bytes.

Below the given results are presented at the Fig. 4.

The Fig.4a shows the exponential dependence of probability of packet loss on quantity of slave-applications.

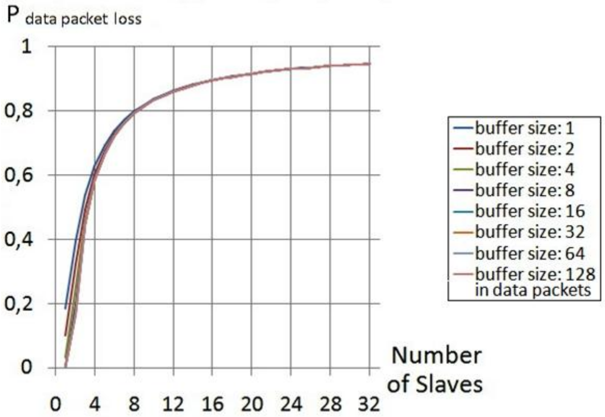

Fig. 4. a) Slave speed channel = 40 Mbps, Host speed channel = 40 Mbps, network load = 60%

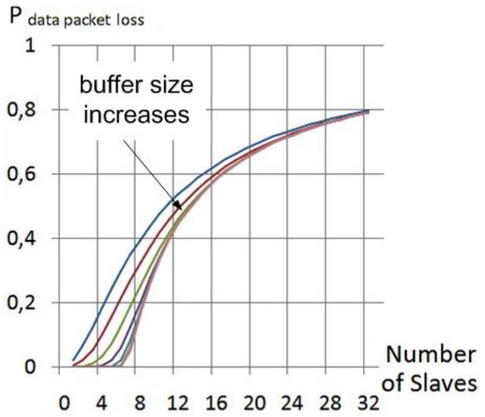

Fig. 4. b) Slave speed channel = 40 Mbps, Host speed channel = 160 Mbps, network load = 60%

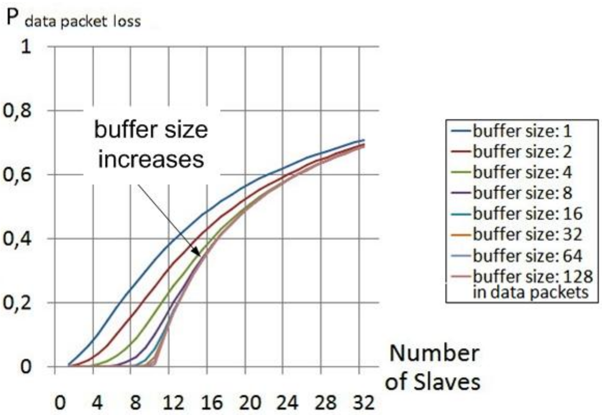

Fig. 4. c) Slave speed channel = 40 Mbps, Host speed channel = 240 Mbps, network load = 60%

Fig. 4. d) Slave speed channel = 40 Mbps, Host speed channel = 400 (SpW max) Mbps, network load = 60%

As it is seen from the Fig. 4b, 4c and 4d the increasing speed of host channel doesn't notable improve the probability of packet loss. The maximum bottom boundary of value is 0.48 with size of buffer = 8 data packets. Since full size of data packet with payload is 1024 Bytes. Then buffer's size will be 8192 Bytes = 8 KBytes. It corresponds with one of the technical parameters of the SpaceWire routing switch MSK-2. And we have just 52% of successful data transfer. Furthermore, the increasing of buffer's size from 8 to 128 data packets doesn't make the situation better. Hence employment of buffer with size more than 8KBytes is undesirable.

For video-voice streaming applications and others the percentage of packet loss should be generally not more than 1% [6].

To get this quality we've decreased the network load from 60% to 20%. The results presented below at the Fig. 5.

From the Fig. 5d it's clear that the probability of packet loss is close to 1%. It's got due to decreased network load. The relevant size of buffer = 8 data packets (8 KBytes). It looks more promising.

Evaluation of various parameters and its combinations gather required information of the network. This kind of information is essential in the development process of the network equipment.

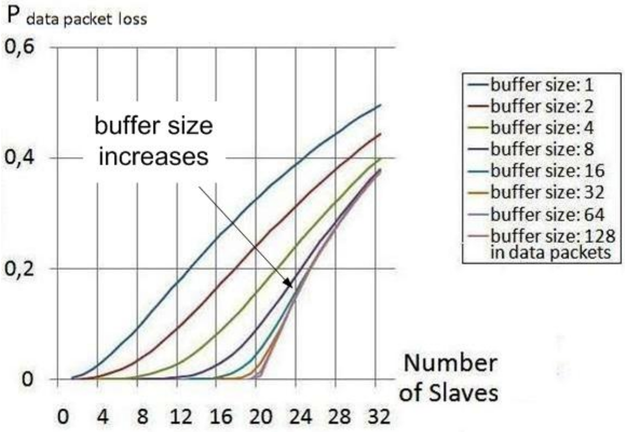

Fig. 5. a) Slave speed channel = 40 Mbps, Host speed channel = 400 (SpW max) Mbps, network load = 50%

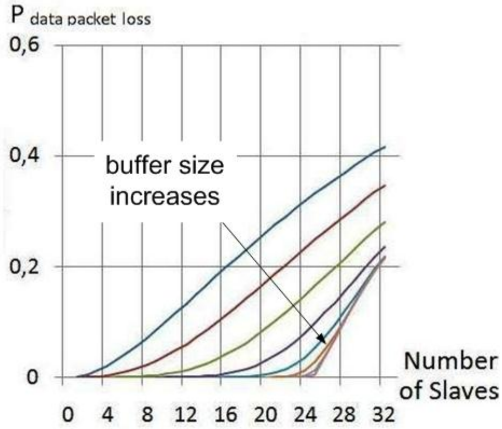

Fig. 5. b) Slave speed channel = 40 Mbps, Host speed channel = 400 (SpW max) Mbps, network load = 40%

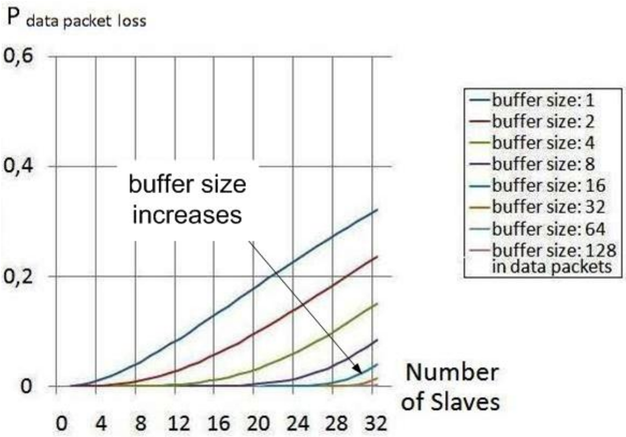

Fig. 5. c) Slave speed channel = 40 Mbps, Host speed channel = 400 (SpW max) Mbps, network load = 30%

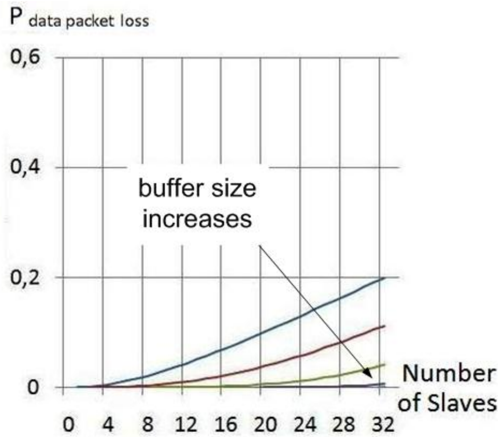

Fig. 5. d) Slave speed channel = 40 Mbps, Host speed channel = 400 (SpW max) Mbps, network load = 20%

III. Conclusion

The research of the SpaceWire Network with streaming STP data transfer could give a significant information for the Network Administrator, which would make the process of the network configuration much easier. The number of successfully transmitted STP data packets and the number of slave nodes with various characteristics of the network are the interesting parameters for the research.

References

[1] I. Korobkov "Embedded systems' transport protocol choosing for modelling over the SpaceWire model", *Proceedings of 9th Seminar of Finnish-Russian University Cooperation in Telecommunications (FRUCT) Program (pp. 51)*, Petrozavodsk State University (PetrSU) , Petrozavodsk, 2011.

[2] ESA, standard ECSS-E-ST-50-52C, "SpaceWire - Remote memory access protocol", *Publications Division ESTEC*, Noordwijk, The Netherlands, February 5, 2010.

[3] Y. Sheynin, E. Suvorova, F. Schutenko, V. Goussev, "Streaming Transport Protocols for SpaceWire Networks", *International SpaceWire Conference 2010*, Saint-Petersburg University of Aerospace Instrumentation (SUAI), Saint-Petersburg, 2010.

[4] V. Olenev, I. Korobkov, N. Martynov, A. Shadursky "RMAP and STP protocols modelling over the SpaceWire SystemC model", *Proceedings of 8th Seminar of Finnish-Russian University Cooperation in Telecommunications (FRUCT) Program (p. 111)*, Lappeenranta Univesity of Technology (LUT), Saint-Petersburg, 2010.

[5] Oren Ben-Kiki, Clark Evans, Ingy döt Net YAML specification final draft version 1.1, "YAML Ain't Markup Language", January 18, 2005.
[6] OpenNet internet resource, Cisco Routers and Switches, Quality of Service, http://www.opennet.ru/docs/RUS/qos_oper/.
[7] Vasilyev K.K. Mathematical simulation of communication systems: manual / K.K.Vasilyev, M.N.Sluzhivy. – Ulyanovsk: UlGTU, 2008. – 170 pages.