Methods for real-time data delivery with SpaceWire

Dmitry Raszhivin
St.Petersburg State University of Aerospace Instrumentation
Russia Federation
Email: dmitry.raszhivin@guap.ru

Abstract

SpaceWire is a modern communication network, proposed for spacecraft and aircraft applications. There is a need for a spacecraft avionics network technology which combines the key features of SpaceWire with the quality of service requirements of real-time avionics applications. Several approaches were proposed to build real-time predictable network over SpaceWire. Review and comparison of this approaches are presented in this paper.

I. Introduction

SpaceWire [1] is a spacecraft communication network based in part on the IEEE 1355 standard of communications. Within a SpaceWire network the nodes are connected through low-cost, low-latency, full-duplex, point-to-point serial links and packet switching wormhole routing routers. SpaceWire covers two (physical and data-link) of the seven layers of the OSI model for communications.

SpaceWire is an asynchronous network which uses wormhole routing. The leading byte(s) on a SpaceWire packet determines the route through the network using path or logical addressing. When the start of a packet arrives at a router it is switched to the required output port straightaway, provided that the required output port is not already being used to transfer another packet. Storing and forwarding of packets is not used by SpaceWire routers. This reduces the amount of buffer memory required in the routing switches. A disadvantage arises when the output port is already being used to transfer another packet and the packet has to wait for the output port to become free. The packet will be left strung out across the network from the blockage back to the source of the packet. It will prevent any other packet being transferred across the links that the packet is occupying.

SpaceWire is ideal for data-handling applications but does not address avionics and other applications where responsiveness, robustness, determinism and durability are essential requirements. There is a need for a spacecraft avionics network technology which combines the key features of SpaceWire with the quality of service requirements of real-time avionics applications. One critical requirement for avionics applications is deterministic delivery of information.

Several approaches were proposed to build real-time predictable network over SpaceWire. Review and comparison of this approaches are presented in this paper.

II. Time-division multiplexing

Nowadays in Space Technology Centre of University of Dundee under the leadership of Steve Parkes is being developed a family of standards, proposed for deterministic data delivery with SpaceWire. They are based on common idea of separating bandwidth using time codes.

A. SpaceWire-RT

SpaceWire-RT was presented as Initial Protocol Definition Draft A at 2008.[3]

The communications model for SpaceWire-RT is one of virtual point-to-point connections across the SpaceWire network each of which connects a source channel buffer in one node to a destination channel buffer in another node. There are two types of system supported:

1) Asynchronous, where the sending of information over the SpaceWire network is asynchronous and priority is used to provide timeliness of delivery. Information in lower number source channels will be sent before information in higher number channels.
2) Synchronous, where information is sent over the SpaceWire network synchronously with each source channel being assigned one or more time-slots when it is allowed to transmit information.

A user application writes into one of the source channel buffers available. This information is then transferred across the SpaceWire network and become available in the corresponding destination channel buffer.

The SpaceWire-RT protocol includes the following functions:

1) Segmentation Function
   SpaceWire-RT sends information across the SpaceWire network in protocol data units (PDUs) each with a size up to a specific maximum PDU size.
2) End to End Flow Control
   End to end flow control is necessary to make sure that there is room in a buffer at the destination node before a PDU is sent. Flow control is achieved by the destination channel buffer sending a buffer flow control token (BFCT) when it has enough room for another maximum length PDU. To avoid a problem if an BFCT is lost BFCTs are acknowledged.
3) Redundancy Function
   The Redundancy Model adopted by SpaceWire-RT is that of alternative paths from a source node to a destination node across a SpaceWire network.
4) Priority
   A separate source channel buffer and destination channel buffer are used for each level of priority to be supported for each destination. A PDU will be sent from a source channel buffer with a lower channel number before sending a PDU from a source channel buffer with a high channel number.
5) Scheduling
   The network bandwidth is separated using time-division multiplexing into a series of repeating time-slots. A schedule table is used in each source to specify in which source channel buffer(s) are allowed to send information during each time-slot. The schedule tables in every source are devised to avoid conflicts on the network.
   SpaceWire-RT provides four QoS classes, based on different reliability and timeliness guarantees.

B. SpaceWire-D

SpaceWire-RT was presented as Initial Protocol Definition Draft A at 2008[2]. It can be considered as a subset of SpaceWire-RT standard.

Due to SpaceWire-D requirements, information shall be passed between SpaceWire nodes using the RMAP protocol. Network bandwidth is divided into set of time-slots, that are delimited by SpaceWire time-codes.

As well as in SpaceWire-RT, a schedule table is used in each source to specify which node is allowed to send information during each time-slot. The schedule tables in every source are devised to avoid conflicts on the network.

A time-code watchdog timer should be kept in each initiator to check for the correct arrival of each time-code. If regular time code was received early or late, user application would be noticed about this event. The time-code watchdog is illustrated in figure 1.

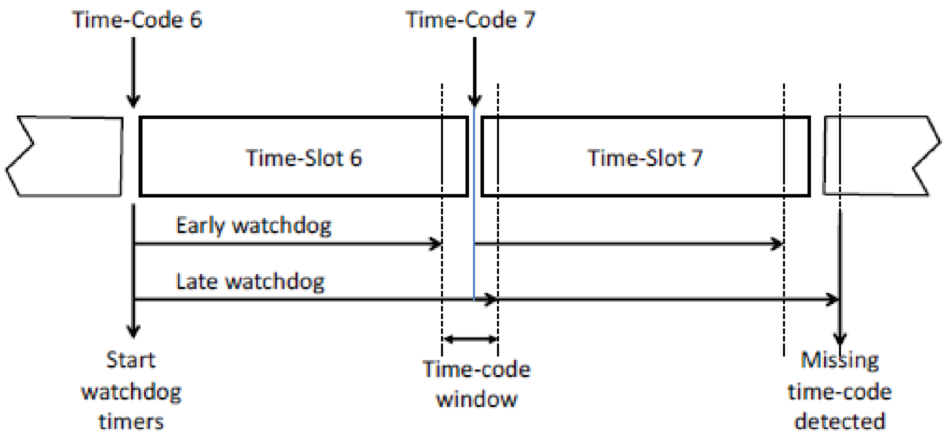

Fig. 1. Time-code watchdog

III. Packet overtaking

Yusuke Murata, Takuma Kogo and Nobuyuki Yamasaki from Keio University propose a packet overtaking scheme to realize pre-emption on a SpaceWire network.[4]

The routing table at SpaceWire nodes consists of a logical destination, a physical output, and a priority. New real-time control code proposed to notify if packet overtaking occurs at switch.

Figure 2 shows a packet overtaking scheme in a SpaceWire router switch. If a packet is overtaken in the midstream of a packet, the control code for the high priority packet is added to the head of the high priority packet. This code switches a virtual channel at next router, so next packet would be received at new buffer. After transferring the high priority packet, the control code for the low priority packet is added to the head of the low priority packet. This code switches virtual channel back, so the low priority communication restarts. The SpaceWire switch keeps the destination address of the pre-empted packet to transfer the packet. Even if a low priority packet is divided by a high priority packet, the low priority packet that consists of discontinuous flits can be sent to the correct destination node.

IV. Comparison

A general comparison of the described methods is presented in table I. Before choosing one or another solution, system architect must decide, what services are required for realizable network and what traffic would be transmitted.

SpaceWire-D and SpaceWire-RT are low adapted to the bursty traffic, so if applications in network generate data unpredictably, it is appropriate to use packet overtaking scheme. Also, packet overtaking is preferable scheme if new nodes would be connected to existing

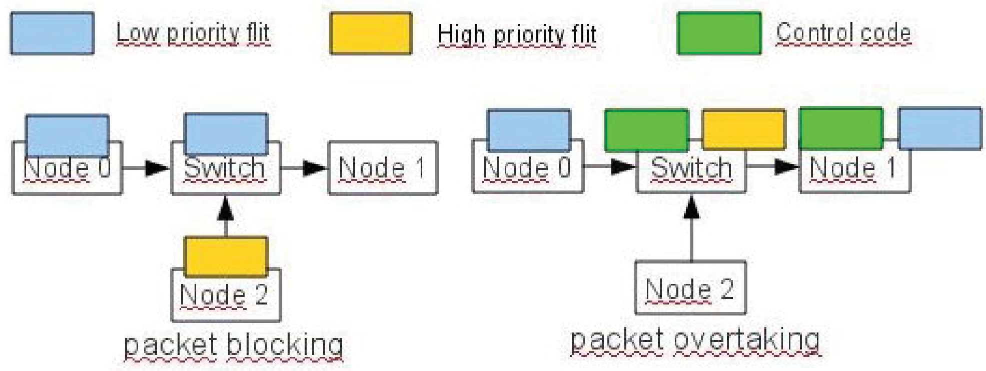

Fig. 2. Packet overtaking scheme

TABLE I
COMPARISON

	SpaceWire-RT	SpaceWire-D	Packet overtaking
Changes to SpW standard are needed	No	No	Yes
Guaranteed data delivery	Yes	No	No
Predictable delivery time	Yes	Yes	Yes
Suitable for bursty traffic	No	No	Yes
Limited packet size	Yes	Yes	No

previously created network. With SpaceWire-RT or SpaceWire-D, when connecting a new node, the schedule table has to be manually reconfigured.

But if network is predictable and it's structure is pre-defined, SpaceWire-D or SpaceWire-RT is a better solution. SpaceWire-D is much simpler SpaceWire-RT, so if network does not require mechanism for guaranteed data delivery, or this functionality is shifted to high-level protocols, SpaceWire-D is preferable.

Development methods of automatic schedule table composing is interesting research area for future work. Different types of schedule tables are described at SpaceWire-D and SpaceWire-RT standards, but there is no algorithm for schedule compiling from network model.

REFERENCES

[1] ECSS, "SpaceWire - Links, nodes, routers and networks", ECSS-E-ST-50-12C, July 2008
[2] S.Parkes, A. Ferrer "SpaceWire-D: Deterministic Data Delivery with SpaceWire", International SpaceWire Conference, St Petersburg, Russia, June 2010.
[3] S.M. Parkes and A. Ferrer-Florit, "SpaceWire-RT Initial Protocol Definition", Draft 1.0, SpaceNet Report No. SpW-RT WP3-200.1, ESA Contract No. 220774-07-NL/LvH, March 2008.
[4] Y. Murata, T. Kogo, N. Yamasaki "A SpaceWire extension for distributed real-time systems", International SpaceWire Conference, St Petersburg, Russia, June 2010.