32nd Congress of the International Council of the Aeronautical Sciences
September 6-10, 2021
Pudong Shangri-La, Shanghai, China

INTEGRATED ON-BOARD NETWORKS FOR NEXT GENERATION AVIONICS

Elena Suvorova[1], Valentin Olenev[1]

[1]St. Petersburg State University of Aerospace Instrumentation, Russian Federation

Abstract

Future airplanes require new generation avionics concepts and technologies. Supersonic airplane projects are coming back again. Hypersonic passenger planes are in future projects roadmaps. Automated flight technology, software pilot capabilities, unmanned aircraft systems (UAS) promise to dramatically change the face of commercial and passenger planes and their avionics. Next generation aircraft set more demands and strict requirements to its distributed integrated modular avionics (DIMA) - more integration, more strict real-time constraints, more scalability, more fault-tolerance. It requires much higher rate data streams, much more integral processing power. DIMA should implement both basic avionic functions, computational intensive sensor sensors/actuators functions in real time, AI applications, future autonomous piloting functions with extensive data processing. DIMA integrated networking infrastructure should support not only a variety of information flows but distributed high-performance processing also. We consider AFDX, TTE and SpaceFibre networking technologies' features for distributed avionics, propose SpaceFibre based integral interconnection for IMA-NG - Integrated Communications Network (ICN). SpaceFibre-based ICN may form scalable re-configurable distributed platforms for all types of processing capabilities, easily scaled to increase performance. ICN wiring is based both on copper and fiber-optic cabling, depending on particular aircraft and its avionics project scale and requirements. We present the SpaceFibre-based architecture, its protocol stack. The article shows SpaceFibre-based ICN key features – high-rate communications, strict partitioning of communication resources for applications' information flows, low latency guaranteed real-time communications, scalability, reconfigurability, fault-tolerance. SpaceFibre-based ICN is efficient in building DIMA with system level fault-tolerance with operated redundancy. These features are supported both by automated ICN network design and configuration, and by run –time network management.

Keywords: SpaceFibre, DIMA, time synchronization, reconfiguration, fault-tolerance.

1. Introduction

A large number of different applications are operating in modern on-board networks. From the system operation point of view, these applications perform critical and non-critical functions. Isolation of applications is very important for on-board computer networks. The execution of non-critical applications should not interfere with the execution of critical applications in any way. In addition, there should be no influence between various critical applications. Any errors that can potentially occur during the execution of one application (process) should not lead to distortion of the data, which is executed by other applications (processes), or distortion of the source code of other applications. All processes should be provided with computing and / or communication resources in a timely manner, so that they have time to finish execution before the deadline [1, 2, 3, 4, 5, 9].

For quite a long time, on-board computer networks have been developed as federated systems [2, 5, 6, 10] (see Figure 1A). In this case, for each application or a group of applications performing a certain separate function in the system, separate hardware resources are allocated. These resources are a hardware subsystems consisting of a separate computing module or a group of computing modules interconnected by a dedicated communication system. If applications performing different functions should communicate with each other, then a communication network should be organized between them. In such systems, the absence of mutual influence between applications is ensured by the hardware implementation, because all the hardware resources that are needed for the particular application are used by this application only. In this case, resource conflicts are impossible. Access of some applications to the memory of other applications is also impossible. This is a very significant advantage of this design approach. Another advantage is the possibility of completely autonomous development, debugging and certification of each subsystem.

However, on-board computer networks designed in this way have a number of disadvantages. The composition of subsystems is specialized for a single task or a very narrow range of tasks. The on-board computer network as a whole can include several dozen types of subsystems, which makes its maintenance and repair a very time-consuming and expensive task. The loading of equipment in the part of the subsystems can be less than expected, however, it is not possible to use this subsystem for solving other problems, redistribute the load of the subsystem. In case of a failure of one of the subsystems, it is not possible to transfer the tasks performed in it to other subsystems. To manage the failures, the separate spaced redundancy is performed for each of the subsystems. This leads to bigger chip area and high power consumption overheads. [6, 7, 11].

The Integrated Modular Avionics (IMA) concept was developed to address these issues [1, 8, 13] (see Figure 1B). Within the framework of this concept, the same hardware resources are used to perform various tasks, including tasks of varying degrees of criticality. Hardware resources are used by different tasks in a time-sharing mode. Each application or group of applications implementing a separate function runs in a logically separate environment (partition).

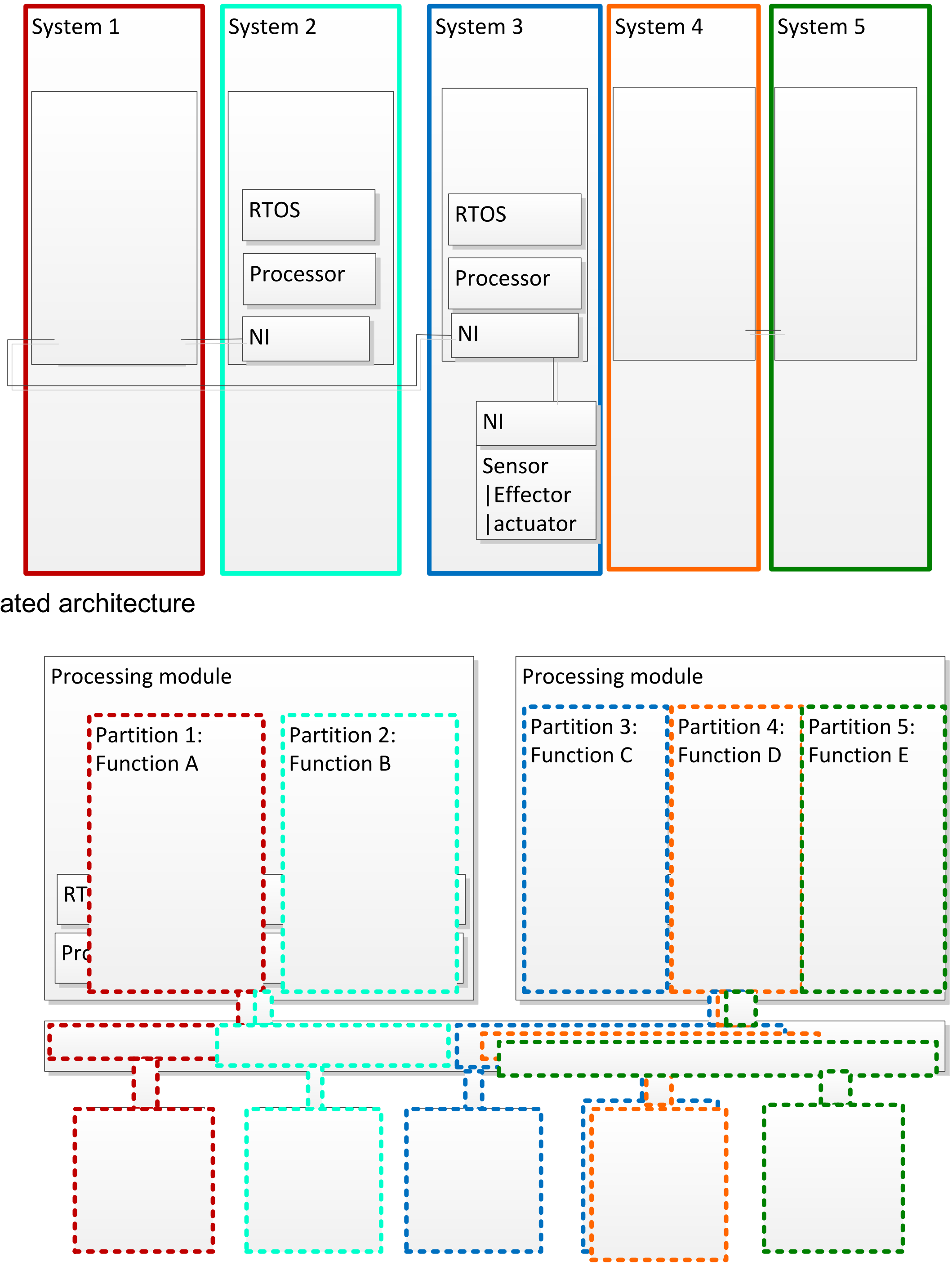
To implement this concept, the ARINC 653 “Avionics Application Software Standard Interface” standard was developed. This standard specifies the partitioning scheme - the division of the resources of the on-board computing system between applications that perform critical and non-critical functions. The operating system for each application implements a separate function, allocates a partition – a separate environment for execution. Not all actions performed within one partition should have no effect on applications running on other partitions. The standard defines a set of mechanisms for organizing of interaction between partitions.

When using this concept at the physical level, the on-board computer network is represented by a set of terminal nodes interconnected by a local network. A set of terminal nodes can include universal computers, specialized computers, sensors (devices that collect information), devices that perform information visualization functions; there can also be separate terminal nodes – memory. All system resources except memory resources can be used by different partitions in a time sharing mode (see Figure 1). Each partition is given a set of resources that it can use. A set of timeslots is specified for each resource, which can be used by a particular partition. Space partitioning is used for memory resources. Each Partition is allocated a set of memory areas (range of address) and the rules for their use (admissibility of read and write access) [2, 11, 13].

This approach prevents the propagation of an error that occurs in one of the partitions beyond its boundaries. The error will not be able to affect the operation of the whole system [2, 7, 11, 14, 15, 16].

The use of this concept and this standard makes it possible to assemble an on-board computer network from universal subsystems at the hardware level. This greatly simplifies and reduces the cost of assembly, debugging and subsequent maintenance of equipment. Due to the use of universal hardware, the time required for the development, validation and verification (V&V) of the developed software is reduced. Such systems support dynamic reconfiguration. Processes can be moved from one hardware subsystem to another in case of a failure or to manage the load of the subsystems. This is achieved by changing the set of hardware resources and the rules for their use, specified for each partition. Thus, the use of this approach significantly simplifies and reduces the cost of development and exploitation of systems. Also it reduces hardware costs in comparison with federated architectures [5, 8, 9, 10, 12].

However, using this approach requires very strong proof that there is no interference between processes running in different partitions. Such proofs should be fulfilled for every newly developed system.



A) Federated architecture

B) IMA architecture

Figure 1 – Examples of a federated (A) and IMA (B) system structures

The communication network between terminal nodes is one of the most important parts of the onboard computer system. In the framework of the IMA concept, it is called a “virtual backplane” of a data communications network. Its characteristics largely determine whether the system will meet the condition that there is no interaction between tasks.

The communication system must ensure the guaranteed delivery time for data packets between the nodes. Each data stream may have specific delivery time requirements. They should be met regardless of the presence or absence of other data streams in the system, of their intensities. Also requirements should be met when errors occur during the operation of the system, as a result of failures in terminal nodes, routers and communication channels. In particular, when the intensity of one of the data streams becomes significantly higher than the expected, as a result of an error like "babbling idiot" in a terminal node or router.

In this case, it should be possible to allocate a certain amount of bandwidth, timeslots for data exchange between individual sources and receivers. To implement this, many systems use mechanisms that ensure the use of the resources of the communication system by different data streams. Most of them are based on assigning different data streams to different virtual channels / virtual networks / virtual connections, each of which is allocated a certain set of timeslots, in which the use of the resources of the communication system is allowed.

The communication system should provide the possibility of dynamic reconfiguration – changing the set of data transmission paths, the resources of the communication system allocated for them. Dynamic reconfiguration can be performed when the operating mode of the system changes and the set of tasks changes; tasks are redistributed to control the load, or due to failures of individual system nodes.

Currently, there are several onboard computer networks developed using the IMA concept [4, 13]. For the implementation of communication systems, they use the AFDX [17, 18] and TTethernet [19, 20, 21] standards. Both of these standards are built on top of the Ethernet standard. Initially, the Ethernet standard does not define the mechanisms necessary to implement the IMA concept. Additional mechanisms are defined at the upper levels of the AFDX and TTethernet standards.

2. SpaceFibre standard

The SpaceFibre standard was developed specifically for on-board networks, taking into account the specifics of this class of networks [22]. Modern on-board networks can include from several tens to hundreds and thousands of terminal nodes and routers. Onboard networks are used to transfer data streams from sensors, to form High Performance Computer (HPC) clusters, etc. Networks based of SpaceFibre standard have an arbitrary topology (graph of connections between terminal nodes and routers).

The SpaceFibre standard specifies the layers of the OSI model protocol stack from physical to network (see Figure 2). A number of transport protocols have been developed for SpaceFibre networks. These protocols RMAP [23], STP-ISS [24], ESDP [25], etc. Specifications of these protocols are implemented as separate standards.

SpaceFibre offers to use copper cables or fiber optics at the physical level. When using optical fiber, the length of communication lines can be several tens of meters. The data transfer rate could range from 1.25Gbps to 6.25Gbps.

The SpaceFibre standard defines a mechanism for virtual channels (at the data link layer), virtual networks (at the network level). Each data link can support from one to 64 virtual channels. The number of virtual channels is determined by the minimum number of virtual channels among the two subscribers participating in this data link. Each VC within one device port is assigned a physical number, starting from 0. One data link can have up to 32 virtual channels. When exchanging data in a data link, data is transferred between virtual channels with the same physical numbers.

Data transfer at the level of virtual channels is carried out in terms of frames (see Figure 3). The data frame size can range from 1 to 64 4-byte words. Frame boundaries and packet boundaries are generally unrelated. For example, the end of one packet and the beginning of the next can be transmitted in the same frame. Several short packets also can be transmitted in one frame.

In the data link, a crediting scheme is used to avoid data loss due to overflow of the receive buffer. The receiving side sends the transmitting side special FCT control code with credits – information on the amount of free space in the buffer. FCTs are sent on a per VC basis. One FCT corresponds to

64 4-byte words. At the beginning of work or after a reset, the receiving side sends the number of FCTs corresponding to the buffer size of each virtual channel. Further, after 64 words of data arrive from the transmitting side via the virtual channel, the receiving side can send another FCT.

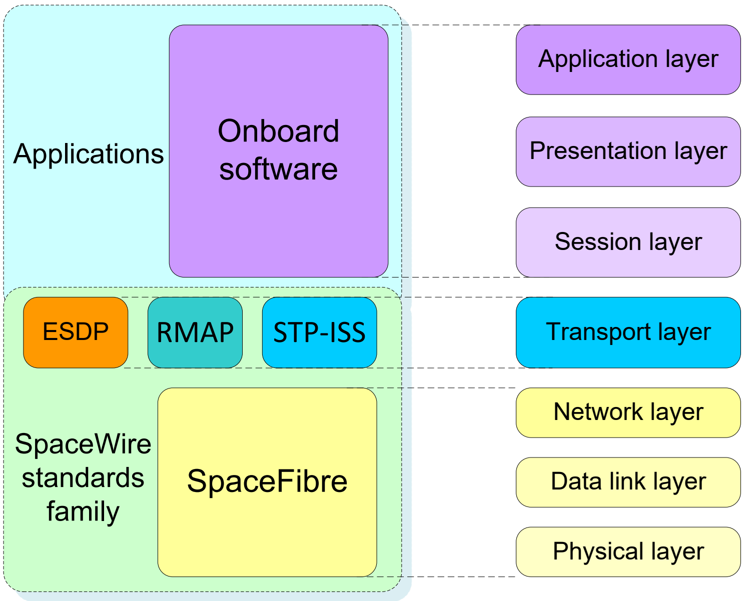


Figure 2 – SpaceFibre and transport protocols

All frames are provided with checksums (CRC) and sequence numbers to detect errors that may occur during data transmission over a physical channel, and lead to distortion or loss of data. In case of mismatch of the checksum or the frame sequence number, the transmitting side is informed about a possible error and the frames are re-transmitted. If the receiveк gets a frame with an incorrect format, it is discarded.

These mechanisms make possible to prevent failures in physical data transmission channels. To avoid failures, regular redundancy of network equipment can be used.

A class of service should be specified for each virtual channel. It is characterized by a priority level, physical channel bandwidth and / or a list of timeslots in which transmission is allowed.

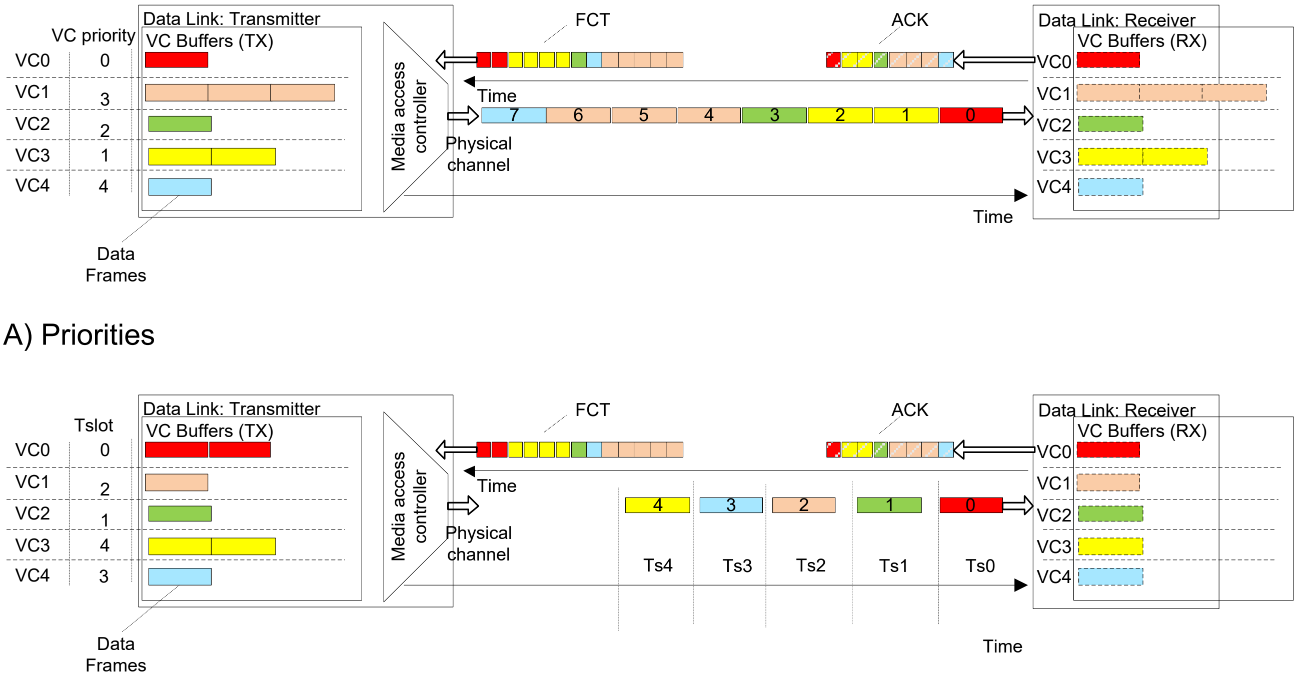
On the transmission side, after the transmission of the next data frame, the Media access controller selects from which virtual channel the next frame will be transmitted. The selection involves those virtual channels in which there is a data and credits for transmission. Only those channels that are allowed to transmit in the current timeslot are considered. If none of the virtual channels are allowed to transmit has data for transmission or credits, then a special IDLE frame is transmitted. If there are several channels that can transmit data in the current timeslot, then the channel with the highest priority sends it. If there are several channels with the same priority, then the one with the largest unused bandwidth is selected.

If the next timeslot ends and at this time the next frame is transmitted, then its transmission is not interrupted, the frame is transmitted to the end.

It should be noted that the settings of the virtual channel in the data link may not be symmetrical. For example, if a data link connects port 1 of router 1 and port 3 of router 2, then in port 1 can be allocated 80% of the bandwidth, and in port 2 can be allocated 20%. This allows you to support asymmetric data streams in different directions.

The total bandwidth allocated to all virtual links on one device port must be less than 100%. Some bandwidth must be reserved for the transmission of service information. As a rule, for the transmission of service information, up to 10% of the bandwidth is reserved.

A virtual network is a set of virtual channels belonging to different links of data transmission (see Figure 4). If data passing through a virtual network should be transmitted through a data link, then this data link should have a virtual channel corresponding to this virtual network. Each port of the SpaceFibre device has a table of correspondence between the numbers of virtual channels and virtual networks.



A) Priorities

b) Scheduling

Figure 3 – Data exchange at the datalink layer

One physical network can have up to 64 virtual networks. The number of virtual networks using one data link is limited by the number of virtual channels in this data link. At the network level (in routers), data belonging to different virtual networks should be transmitted independently. This mechanism allows to share network resources (network devices, physical communication channels) between data streams in a time-sharing mode. The supported quality of service are priorities, guaranteed delivery time (based on scheduling mechanism), guaranteed bandwidth.

The same quality of service type is defined for all data streams transmitted over the same virtual network.

At the network level, the SpaceFibre standard uses wormhole routing. However, it would be more accurate to call it Virtual Cut Through due to the fact that there are buffers in the input and output port controllers.

The SpaceFibre network topology can be different. However, to avoid deadlocks, data paths should not contain loops.

At the network level, it supports three types of addressing - path, logical and regional-logical. A path address is a list of router output port numbers which packet should pass during transmission from source to destination. The network header of a packet consists of one or more segments. The length of each header segment is one byte. The number of segments in the network header of a packet in this case is equal to the number of transit routers, which the packet should pass. Each segment contains the output port number through which the packet should be transmitted in the next router. Each transit router uses a segment at the beginning of the packet header to select an output port and then erases that segment. The use of path addresses does not impose any restrictions on the size of the network, the number of terminal nodes in the network. However, with the growth of the network, the increase in the number of transit routers between the source and destination, the length of the packet header also increases. Because of it path addressing in large networks is used mainly for network discovery and monitoring purposes.

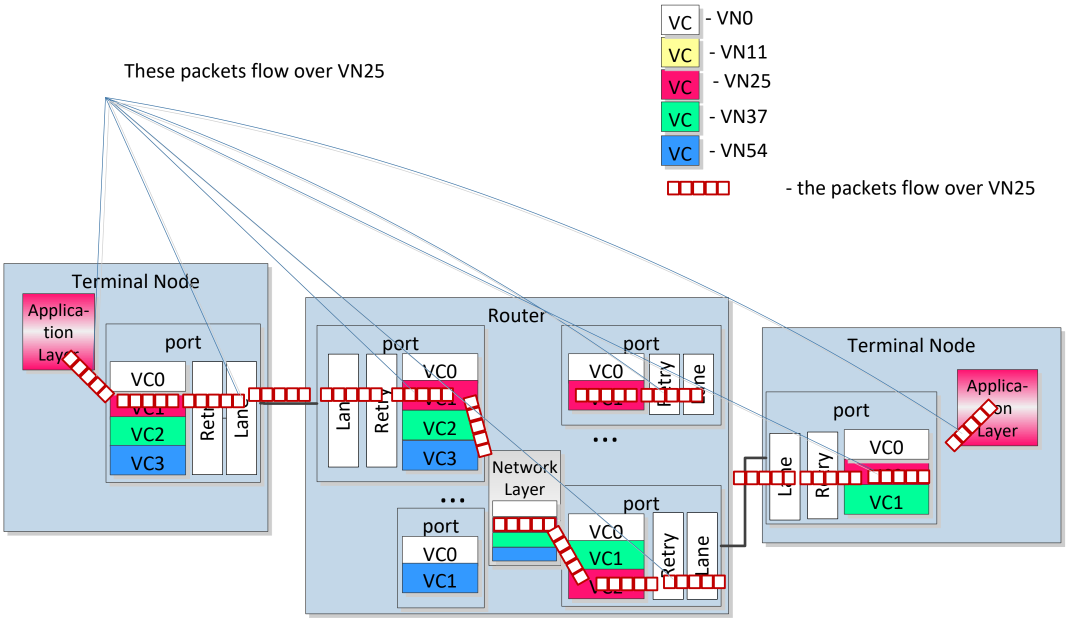


Figure 4 – Example of a virtual network with virtual channels

When using logical addressing, the packet header consists of one segment, which contains the logical address of the packet. To select the outgoing port number in this case, routers use a routing table that sets the correspondence between addresses and outgoing port numbers. When using logical addressing, all transit routers use the same logical address for routing. It is possible to use 224 logical addresses, so the routing table in each router contains 224 lines. This is significantly less than the typical table size for routers supporting standards such as Serial RapidIO and InfiniBand. The small size of the routing table is a great advantage for aerospace applications, since it is very small. It allows to significantly reduce hardware implementation costs.

However, a network using logical addressing can support up to 224 subscribers. This is not enough for many on-board networks. To eliminate this limitation, regional-logical addressing is used. When using this type of addressing, the packet header consists of several segments. Each segment contains a logical address. The entire network is logically divided into separate regions. When transmitting a packet within one region, the address located first in the packet header is used. In routers located at the border of the region, the first byte of the header is deleted. To implement this mechanism, the header separation flag is indicated in the routing tables for the packet header.

Regional-logical addressing allows to create networks of any size. In this case, the length of the packet's network header will be significantly less than when using path addressing.

At the network level, a priority mechanism can be supported. To implement it, each network header in the routing table can be assigned a priority level.

To avoid errors of the network and upper layers a timeout mechanism is supported. This mechanism allows to detect situations when the packet header or the next byte of the packet cannot be transmitted to the output port due to its unavailability. The unsent part of such a packet is discarded. Also, this mechanism allows to detect situations when the next byte of the packet does not come from the input port for too long. In this case, the erroneous end of the packet EEP is sent to the outgoing port. This timeout allows to prevent blocking of virtual network resources by a packet whose transmission has stopped, which is especially critical for wormhole routing networks.

According to the SpaceFibre standard, each network device should support a configuration port through which a remote system administrator can access the device's configuration address space to set the operating mode and to read the configuration. The memory map of the configuration address space is currently not specified.

The RMAP transport protocol is typically used to access the configuration space. Within the framework of this protocol, packet formats are specified. It makes possible to implement commands for reading, writing address memory or FIFO (reading and writing of one or more words at the same address). Atomic calls are also supported, during the execution of which a word is read, the bits of this word specified by the mask are modified, and the modified value is written to memory.

Potentially, other transport protocols can be used to access the configuration space, for example, STP-ISS.

3. Building a DIMA network based on the SpaceFibre standard

As it was shown in the previous section, mechanisms allowing to share physical network resources between data streams with different characteristics are integrated directly into the main part of the SpaceFibre standard and are supported in the data link. This distinguishes this standard from standards such as AFDX, TTEthernet, in which these mechanisms are implemented as an extended feature. Let us see how SpaceFibre mechanisms can be used to transfer mixed critical traffic.

Since SpaceFibre specifies quality of service between virtual networks, traffic flows of different types can be assigned to different virtual networks.

As noted above, virtual network 0 is always used for network configuration and network management. The developer can use the rest of the virtual networks in accordance with the tasks solved in the system. For example, all traffic with no delivery time requirements (not critical for the operation of the system) can be assigned to one virtual network. The traffic that is critical for the operation of the system can be assigned to one or several virtual networks. In particular, traffic with a different degree of criticality, with different requirements for delivery time, and different characteristics of data streams can be assigned to different virtual networks. For example, one of the virtual networks can be used to transmit command traffic, which is the most critical. Another virtual network can be used to transmit real-time video traffic, which also requires guaranteed delivery times.

It should be noted that the basic version of the SpaceFibre standard does not provide the possibility of network resources distributing between data streams belonging to the same virtual network. Regarding to this, if data streams intersect, then it is possible to place them in different virtual networks in order to avoid conflicts of resources (even if they need to provide the same quality service).

A scheduling mechanism implemented at the transport layer and operating within the same virtual network can also be used to avoid resource conflicts in a single virtual network.

In order to enforce the use of physical links in a time-sharing mode between virtual networks, the SpaceFibre scheduling mechanism should be used. When using this mechanism, each virtual network (each virtual channel of this virtual network) is allocated a set of timeslots, in which it is allowed to transmit data. If only one virtual network is allowed to transmit data in one timeslot, then conflicts for network resources are completely excluded. In this way, strictly partitioning support can be provided. In this case, conflicts for the physical resources of the network are completely excluded. The network can provide a guaranteed data transfer time equal to the minimum data transfer time. If an error occurs in any virtual network as a result of a failure in one of the terminal nodes transmitting data in this virtual network or transit routers, then this will not affect the transfer of data for other virtual networks. The transmission of data in each virtual networks is done in strictly allocated timeslots. If there is a deadlock in one of the virtual networks, then this will also not affect the transfer of data through other virtual networks. If a Media access controller in one of the devices fails and the data transmission for different virtual channels does not correspond to the allocated timeslots, then the propagation of this error in the network will be stopped by the next devices in the data transmission path. In these devices, Media access controllers will not allow propagation of data further into inappropriate timeslots. Thus, the correct functioning of the remaining virtual networks will continue. It is illustrated in Figure 5.

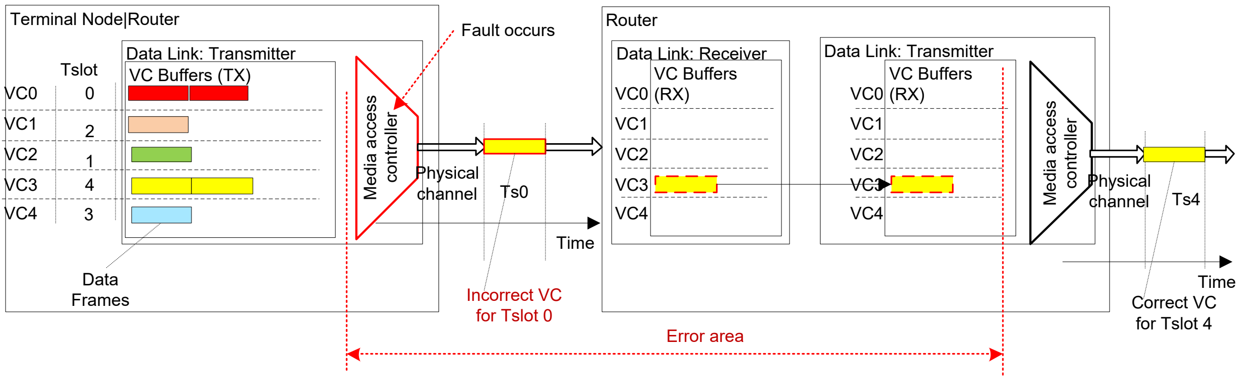


Figure 5 – A mechanism to stop the propagation of an error

Within the framework of the standard, it is possible to use the same timeslot for data transmission of several virtual networks. For example, a virtual network used to carry command traffic can be allowed to transmit data in all timeslots.

If the transmission of data belonging to several virtual networks is allowed in one timeslot, then the choice of the next frame for transmission is carried out in accordance with the priorities of the virtual networks. For example, the virtual network used to carry command traffic can be assigned the highest priority level. This will ensure the minimum waiting time for the transmission of frames over this network if the physical channel is busy.

At different periods of operation of an aircraft in a distributed computer network, different sets of tasks can be performed. They are different modes of operation of the network. In different modes, the data transmission paths over the network can be different; the characteristics of the data streams can significantly vary. Changes to the configuration of data transmission paths can also be performed due to network equipment failures.

The SpaceFibre network has the ability to dynamically change the configuration. Modification of routing tables is possible. String values can be changed / replaced with new ones. It should be noted that if the transmission path of packets with a certain address changes, then the elimination of the possibility of loops in the transmission paths caused by this change should be controlled by the network designer. If such a cycle occurs, there would be no blocking of the virtual network due to the timeout mechanism. However, some of the transmitted data packets will be lost (erased).

Modification of virtual channel settings is possible. Priority levels, bandwidth share, set of timeslots in which transmission is allowed can be changed. To perform these settings, you do not need to close the connection (disconnect) on the physical channel during the reconfiguration. Frames that started transmission before the configuration change will be transmitted according to the old settings. The next frames will be transmitted according to the new settings.

It is also possible to dynamically change the tables of correspondence between the number of the virtual channel and the virtual network. In this case, in order to exclude the situation when some of the frames related to one packet will be transmitted over one virtual network, and some over another virtual network, it is necessary to prohibit the operation of virtual channels with the changed settings in the lookup table. As a result, if the setting was performed at the time of packet transmission, then after the transmitted part of the packet, the erroneous end of the packet (EEP) will be sent, and the tail of the packet will be discarded. If the change in the correspondence between the virtual channel number and the virtual network number is not performed for all virtual channels, then the transmission of the data could be continued over those channels for which the reconfiguration is not performed. Thus, the possibility of dynamic reconfiguration of some virtual networks is supported and the possibility of regular data transfer in other virtual networks is maintained.

In addition to the discussed errors detection possibilities additional mechanisms for traffic filtering in transit routers can be implemented. These mechanisms are mainly designed to detect errors within the same virtual network. Errors of this kind are critical for a separate partition that uses this virtual network (they cannot affect other partitions), except the cases when they concern the data exchange mechanisms between different partitions.

For each virtual network, the maximum allowable packet length can be defined. If a longer packet appears in the virtual network, or even an infinitely long packet, the tail of the packet that exceeds the maximum allowed length will be discarded. As a result, the blocking of this virtual network by this tail of the packet will be excluded.

Due to the fact that the length of the network header of SpaceFibre packets can be different and the standard does not define the end of the network header, generally it is not possible to check the correctness of the network header in transit routers. In addition, it is not possible to check any conditions, associated with the used transport protocols. However, for specific implementations, the allowable lengths and values of all network header bytes in the various routers that are used by a given virtual network may be standardized. This can be used to detect packets with invalid network headers on the network and discard such packets. Network header errors can occur during network debugging, particularly during software debugging. Being able to detect them will simplify the debugging process.

The same scheme can be used to control the transport headers of packets. Using this mechanism can make possible to prevent the propagation of packets related to transport protocols that should not be used in this virtual network, to exclude commands that should not be present in this virtual network.

These mechanisms can be relevant in the process of debugging the network. In the course of normal operation, they can make it possible to eliminate the use of network resources for the transmission of erroneous packets. Such packets use the resources allocated for this virtual network, and can cause the network blocking, which can lead to a violation of the guaranteed delivery time for normally transmitted packets.

As it was mentioned before, guaranteed time data delivery mechanisms of SpaceFibre and transport-layer mechanisms within the virtual network could be implemented by the scheduling mechanisms. In turn, scheduling requires time synchronization methods, which are not specified in the SpaceFibre standard. SpaceFibre specifies the broadcast messages that could be used for the time synchronization [22].

In paper [26] it is shown, that using of time synchronization mechanism offers the synchronization accuracy within one µs for the networks, in which the distance between the synchronization source and other nodes does not exceed 10 transit routers.

4. Extension of SpaceFibre functionality with transport layer protocols

According to Figure 2, there are only a few transport layer protocols are available for SpaceFibre networks. SpaceFibre specifies the RMAP protocol for the remote memory access. However, RMAP functionality is not enough to solve all the tasks of on-board networks. ESDP protocol is specifically intended for transmission of streaming data. STP-ISS is a powerful solution, which offers different quality of service types, configuration flexibility and other features.

Each SpaceFibre virtual channel supports the priority, bandwidth reservation and scheduling qualities of service (QoS). In turn, STP-ISS entities in the nodes support the best effort, guaranteed, priority and scheduling QoS. But STP-ISS and SpaceFibre provide QoS at different levels. STP-ISS provides the end-to-end operation and QoS for the whole network; SpaceFibre describes the datalink layer operation, so the QoS is provided for the point-to-point link.

The joint use of STP-ISS and SpaceFibre protocols has many advantages. Using of STP-ISS best effort QoS gives an ability to use pure SpaceFibre QoS without any additional mechanisms. SpaceFibre guarantees the point-to-point delivery, but it cannot guarantee the end-to-end delivery, because anything can happen inside the router or link may fail, for example. STP-ISS guaranteed QoS could handle such kind of error. If SpaceFibre transmits data in continuous mode, some packets could be deleted from the Virtual Channel buffer. STP-ISS will resend these packets end-to-end.

STP-ISS priority QoS allows applications to send different types of packets via the same virtual channel, in one virtual network. Therefore, on-board network could have virtual networks with several source nodes and different QoS.

STP-ISS schedules end-nodes' entities, and SpaceFibre schedules virtual channels, so STP-ISS scheduling provides QoS partitioning between information flows in a virtual network in case of multiple transmitters. Scheduling in STP-ISS can be used to control data transmission from several end node initiators.

If the hardware resources are limited and we cannot have many virtual networks. Due to the limited hardware resources in real network implementations there are much more data flows than the number of implemented virtual channels. With STP-ISS QoS network designers would be able to place several initiators in one virtual network.

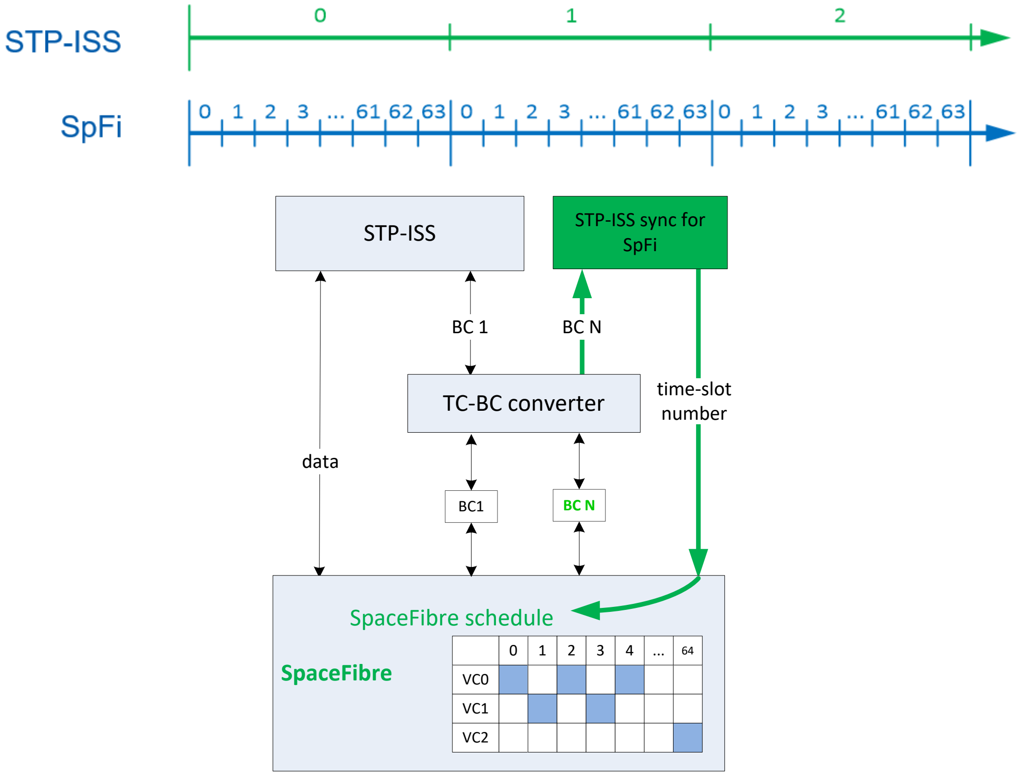
STP-ISS would help to solve the problem of packets being stuck in a SpaceFibre virtual network. STP-ISS discards a packet in case of application does not read data from receive buffer (and it becomes full). Discarding of the packet releases network recourses, so that other packets can be transmitted.

The problem of porting of STP-ISS protocol for SpaceFibre networks is described in [27].

Additional STP-ISS feature that could be used for SpaceFibre is time synchronization. SpaceFibre time synchronization mechanism is not specified, but STP-ISS protocol describes the synchronization mechanism for Scheduling QoS (adjustable number of time-slots in epoch, synchronization once in an epoch). In its turn SpaceFibre provides point-to-point scheduling for virtual channels and the number of time-slots is always 64, no synchronization mechanisms, just time-slot number to determine the beginning of a time-slot.

STP-ISS and SpaceFibre schedules are asynchronous, however it is possible to use STP-ISS synchronization block for SpaceFibre.

In this case, STP-ISS synchronization for SpaceFibre will use another broadcast channel. Synchronization would be done once in an epoch on reception of correspondent broadcast. Upon synchronization, the time-slot number will be issued to SpaceFibre (from 0 to 63). This way all SpaceFibre devices in a network will be in the same time-slot (see Figure 6).



	0	1	2	3	4	...	64
VC0	▨		▨		▨		
VC1		▨		▨			
VC2							▨

Figure 6 – STP-ISS and SpaceFibre time synchronization

If STP-ISS time-slot is equal to one or several SpaceFibre epochs then it is possible to use STP-ISS scheduling for end-nodes operation via the same virtual network.

5. Conclusion

The paper discusses the SpaceFibre standard and the possibility of building DIMA based on this standard. It is shown that high-rate communications, strict partitioning of communication resources for applications' information flows, low latency guaranteed real-time communications, scalability, reconfigurability, fault-tolerance are provided.

Using of SpaceFibre standard for building of modern DIMA would be a huge step forward for the on-board networks. SpaceFibre is a new standard, but it is very powerful and effective using of this standard will reduce the number of the onboard networks failures during the operation, which is a very important parameter for the aircraft.

Also the paper gave a description of the STP-ISS transport protocol and SpaceFibre joint operation. STP-ISS scheduling can resolve collisions in SpaceFibre virtual networks, virtualization of SpaceWire network to SpaceFibre virtual channels and using of several independent flows in one virtual network. In addition, STP-ISS can provide an operating solution for SpaceFibre time synchronization. Different configurations of SpaceFibre and STP-ISS can provide users with variety of different operation scenarios and modes, add the additional configuration flexibility.

6. Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.

References

[1] Alena R., Ossenfort J., Laws K and others. Communications for Integrated Modular Avionics, IEEEAC, 19 pp. 2006.

[2] J. Schmalzel, F. Figueroa, J. Morris, S. Mandayam, R. Polikar, An Architecture for Intelligent Systems Based on Smart Sensors, *IEEE Transactions on Instrumentation and Measurement*, Vol. 54, No. 4, pp. 1612-1616. 2005.

[3] Fleming C., Leveson N. Improving Hazard Analysis and Certification of Integrated Modular Avionics. *Journal of aerospace information systems*. Vol. 11, No. 6, 15 pp. 2014.

[4] Hollow P., McDermid J., Nicholson M. *Approaches to Certification of Reconfigurable IMA Systems*. 8 pp. 2000.

[5] Christopher B. Watkins, Randy Walter. Transitioning from federated avionics architectures to integrated modular avionics. *DASC'07*. *IEEE/AIAA 26th*. 2007.

[6] Driscoll K. Integrated Modular Avionics (IMA) Requirements and Development, *Proceedings of ARTIST2 Workshop on Integrated Modular Avionics*, Rome, 2007.

[7] Loveless A. On TTEthernet for Integrated Fault-Tolerant Spacecraft Networks. *American Institute of Aeronautics and Astronautics*. 23 pp.

[8] Radio Technical Commission for Aeronautics DO-297. *Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations*, Washington, D.C., 2005.

[9] Bieber P., Boniol F., Boyer M., Noulard E., Pagetti C., *New Challenges for Future Avionics Architectures*, 2014.

[10] Fuchs C. The Evolution of Avionics Networks From ARINC 429 to AFDX, *Proc. Seminar Aerospace Networks SS2012*, Munich, 2012.

[11] Newhouse M., Friberg K., Fesq L., Barley B. Fault Management Guiding Principles, *Proc. Infotech Aerospace 2011 Conference: Unleashing Unmanned Systems*, St. Louis, MI, 2011.

[12] Lauer M., Mullins J., Yeddes M. Cost optimization strategy for iterative integration of multi-critical functions in IMA and TTEthernet architecture. *Proceedings of the 2013 IEEE 37th Annual Computer Software and Applications Conference Workshops*. 2013.

[13] Watkins C. Integrated Modular Avionics: Managing the Allocation of Shared Intersystem Resources, *25th Digital Avionics Systems Conference*, IEEE, Piscataway, NJ, pp. 1–12. 2006.

[14] Ganesan D., Lindvall M., McComas D., Bartholomew M., Slegel S., Medina B. Architecture-Based Unit Testing of the Flight Software Product Line, *Proc. 14th International Software Product Line Conference (SPLC)*, Jeju Island, Republic of Korea, 2010.

[15] Bartley G., Lingberg B. Certification Concerns of Integrated Modular Avionics (IMA) Systems, *27th Digital Avionics Systems Conference*, IEEE, Piscataway, NJ, pp. 1.E.1-1–1.E.1-12. 2008.

[16] Rushby J. New Challenges in Certification for Aircraft Software, *Proceedings of the Ninth ACM International Conference on Embedded Software*, ACM, New York, pp. 211–218. 2011.

[17] Airbus France, *AFDX End System Detailed Functional Specification*, 2003.

[18] Aircraft Data Network, Part 7: *Avionics Full Duplex Switched Ethernet (AFDX) Network*. 2009.

[19] SAE Aerospace. *Time-Triggered Ethernet*, SAE AS6802 edition. 2011.

[20] Steiner W., Bauer G., Hall B., Paulitsch M. *Time-Triggered Ethernet, Time-Triggered Communication*. 2015.

[21] Steiner W. Paulitsch M. Time-Triggered Ethernet, *Industrial Communication Technology Handbook*, Boca Raton, FL, 2nd ed., 2015.

[22] ESA-ESTEC. *SpaceFibre - Very high-speed serial link. ECSS-E-ST-50-11C*.Noordwijk, The Netherlands. 233 p. 2019.

[23] ESA-ESTEC. *SpaceWire - Remote memory access protocol*. ECSS-E-ST-50-52C. 2010.

[24] Olenev V., Lavrovskaya I., Sheynin Yu., Korobkov I., Suvorona E. and others. STP-ISS Transport Protocol for SpaceWire On-Board Networks: Development and Evolution. *International Journal of Embedded and Real-Time Communication Systems*, №5(4). Tampere. IGI Global, pp. 45-76. 2014.

[25] Korobkov I., Suvorova E., Sheynin Yu., Olenev V. Streaming Services over SpaceFibre Networks, *Proceedings of 7th International SpaceWire Conference 2016 (ISC2016)*, Yokohama, pp. 151-158. 2016.

[26] Suvorova E. Time synchronization in SpaceFibre Networks. *Proceedings of 28th Conference of Open Innovations Association Finnish-Russian University Cooperation in Telecommunications (FRUCT) Program*. 12 p. 2021.

[27] Sheynin Yu., Olenev V., Dymov D. STP-ISS Transport Protocol Application for SpaceFibre On-Board Networks, *International Conference on Control, Decision and Information Technologies*. pp. 914-919. 2020.