SpaceFibre: Multiple Gbit/s Network Technology with QoS, FDIR and SpaceWire Packet Transfer Capabilities

SpaceWire Standardisation, Long Paper

Steve Parkes, Chris McClements,
Space Technology Centre, University of Dundee,
Dundee, DD1 4EE, UK
sparkes@computing.dundee.ac.uk

Albert Ferrer, Alberto Gonzalez,
STAR-Dundee
STAR House, 166 Nethergate, Dundee, DD1 4EE, UK

***Abstract*— SpaceFibre is a very high-speed serial link designed specifically for use onboard spacecraft. It carries SpaceWire packets over virtual channels and provides a broadcast capability similar to SpaceWire time-codes but offering much more capability. SpaceFibre operates at 10 times the data-rate of SpaceWire, can run over fibre optic or electrical media, provides galvanic isolation, includes coherence Quality of Service (QoS) and Fault Detection Isolation and Recovery (FDIR) support, and provides low-latency signalling. SpaceFibre can run over distances of 5m with copper cable and 100 m or more with fibre optic cable.**

SpaceFibre is compatible with the packet level of the SpaceWire standard (ECSS-E-ST-50-12) and is therefore able to run the SpaceWire protocols defined in ECSS-E-ST-50-51C, 52C and 53C. This means that applications developed for SpaceWire can be readily transferred to SpaceFibre.

The aim of SpaceFibre is to provide point-to-point and networked interconnections for very high data-rate instruments, mass-memory units, processors and other equipment, on board a spacecraft.

This paper introduces SpaceFibre, describes the SpaceFibre QoS, FDIR and network level operation of SpaceFibre.

Index Terms*—*SpaceWire, SpaceFibre, networks, spacecraft onboard processing

I. *INTRODUCTION*

SpaceFibre [1] [2] [3] [4] is a very high-speed serial data-link being developed by the University of Dundee for ESA which is intended for use in data-handling networks for high data-rate payloads. SpaceFibre is able to operate over fibre-optic and electrical cable and support data rates of 2 Gbit/s in the near future and up to 5 Gbit/s long-term. It aims to complement the capabilities of the widely used SpaceWire onboard networking standard [5]: improving the data rate by a factor of 10, reducing the cable mass by a factor of four and providing galvanic isolation. Multi-laning improves the data-rate further to well over 20 Gbits/s.

SpaceFibre provides a coherent quality of service mechanism able to support best effort, bandwidth reserved, scheduled and priority based qualities of service. It substantially improves the fault detection, isolation and recovery (FDIR) capability compared to SpaceWire.

SpaceFibre aims to support high data-rate payloads, for example synthetic aperture radar and hyper-spectral optical instruments. It provides robust, long distance communications for launcher applications and supports avionics applications with deterministic delivery constraints through the use of virtual channels. SpaceFibre enables a common onboard network technology to be used across many different mission applications resulting in cost reduction and design reusability. SpaceFibre uses a packet format which is the same as SpaceWire enabling simple connection between existing SpaceWire equipment and high-speed SpaceFibre links and networks.

The SpaceFibre interface is designed to be implemented efficiently, requiring substantially fewer logic gates than a RapidIO interface. It is currently being prototyped in a range of onboard processing, mass memories and other spacecraft applications. Interoperability tests between independent Japanese and European implementations were carried out successfully in December 2012 and April 2013.

II. SPACEFIBRE PROTOCOL STACK

The SpaceFibre protocol stack is illustrated in Figure 1.

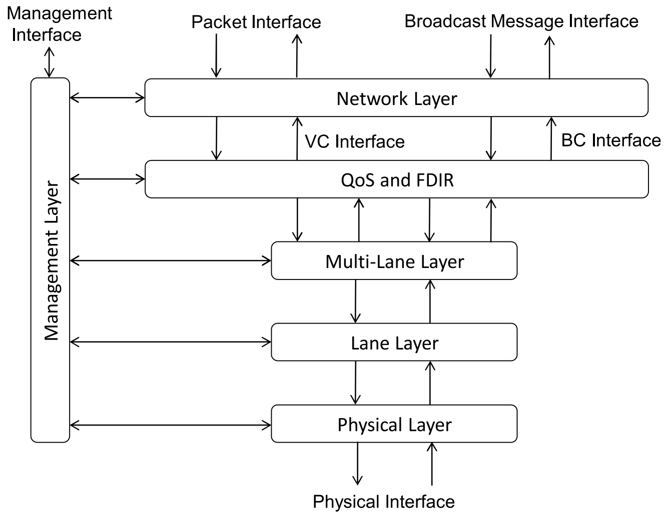

Figure 1 SpaceFibre Protocol Stack

The network layer protocol is responsible for the transfer of application information over a SpaceFibre network. It provides two services: Packet Transfer Service and Broadcast Message Service. The Packet Transfer Service transfers SpaceFibre packets over the SpaceFibre network, using the same packet format and routing concepts as SpaceWire uses. SpaceFibre supports both path and logical addressing. The broadcast message service is responsible for broadcasting short messages (8 bytes) to all nodes on the network. These messages can carry time and synchronisation signals and be used to signal the occurrence of various events on the network.

The management layer is responsible for configuring, controlling and monitoring the status of all the layers in the SpaceFibre protocol stack. For example it can configure the QoS settings of the virtual channels in the QoS and FDIR layer.

The QoS and FDIR layer is responsible for providing quality of service and managing the flow of information over a SpaceFibre link. It frames the information to be sent over the link to support QoS and scrambles the packet data to reduce electromagnetic emissions. The QoS and FDIR layer also provides a retry capability, detecting any frames or control codes that go missing or arrive containing errors and resending them. With this inbuilt retry mechanism SpaceFibre is very resilient to transient errors.

The Multi-Lane layer is responsible for operating several SpaceFibre lanes in parallel to provide higher data throughput. In the event of a lane failing the Multi-Lane layer provides support for graceful degradation, automatically spreading the traffic over the remaining working links.

The Lane layer is responsible for lane initialisation and error detection. In the event of an error the lane is automatically re-initialised. The Lane layer encodes data into symbols for transmission using 8B/10B encoding and decodes these symbols in the receiver. 8B/10B codes are DC balanced supporting AC coupling of SpaceFibre interfaces.

The Physical layer is responsible for serialising the 8B/10B symbols and for sending them over the physical medium. In the receiver the Physical layer recovers the clock and data from the serial bit stream, determines the symbol boundaries and recovers the 8B/10B symbols. Both electrical cables and fibre-optic cables are supported by SpaceFibre.

III. SpaceFibre Quality of Service

A SpaceFibre interface includes a number of virtual channels. Each provides a FIFO type interface like a SpaceWire link. When data from a SpaceWire packet is placed in a SpaceFibre virtual channel it is transferred over the SpaceFibre link and placed in the same numbered virtual channel at the other end of the link. Data from the several virtual channels are interleaved over the physical SpaceFibre connection. To support the interleaving, data is sent in short frames of up to 256 SpaceWire N-chars each. A virtual channel can be assigned a quality of service which determines the precedence with which that virtual channel will compete with other virtual channels for sending data over the SpaceFibre link. Priority, bandwidth reservation, and scheduled qualities of service can be supported all operating together using a simple precedence mechanism.

In this section the SpaceFibre quality of service mechanism is described.

A. Frames and Virtual Channels

To provide quality of service, it is necessary to be able to interleave different data flows over a data link or network. If a large packet is being sent with low priority and a higher priority one requests to be sent, it must be possible to suspend sending the low priority one and start sending the higher priority packet. To facilitate this SpaceWire packets are chopped up into smaller data units called frames. When the high priority packet requests to be sent, the current frame of the low priority packet is allowed to complete transmission, and then the frames of the high priority packet are sent. When all the frames of the high priority packet have been sent, the remaining frames of the low priority packet can be sent.

Each frame has to be identified as belonging to a particular data flow so that the stream of packets can be reconstructed at the other end of the link. Low priority packets belong to one data stream and high priority packets belong to another data stream.

Each independent data stream allowed to flow over a data link is referred to as a virtual channel (VC). Virtual channels are unidirectional and have a QoS attribute, e.g. priority. At each end of a virtual channel is a virtual channel buffer (VCB), which buffers the data from and to the application. An output VCB takes data from the application and buffers it prior to sending it across the data link. An input VCB receives data from the data link and buffers it prior to passing it to the receiving application.

There can be several output virtual channels connected to a single data link, which compete for sending information over the link. A medium access controller determines which output virtual channel is allowed to send the next data frame. When an output VCB has a frame of data ready to send and the corresponding input VCB at the other end of the link has room for a full data frame, the output VCB requests the medium access controller to send a frame. The medium access controller arbitrates between all the output VCBs requesting to send a frame. It uses the QoS attribute of each of the requesting VCBs to determine which one will be allowed to send the next data frame.

Priority is one example of a QoS attribute. Other types of QoS are considered in the subsequent sections.

B. Precedence

For the medium access controller to be able to compare QoS attributes from different output VCBs, it is essential that they are all using a common measure that can be compared. The name given to this measure is precedence. The competing output VCB with the highest precedence will be allowed to send the next frame.

C. Bandwidth Reservation

When connecting an instrument via a network to a mass memory, what the systems engineer needs to know is "how much bandwidth do I have to transfer data from the instrument to the mass memory?" Once the network bandwidth allocated to a particular instrument has been specified, it should not be possible for another instrument to impose on the bandwidth allocated to that instrument. A priority mechanism is not suitable for this application. If an instrument with high priority has data to send it will hog the network until all its data has been sent. What is needed is a mechanism that allows bandwidth to be reserved for a particular instrument.

Bandwidth reservation calculates the bandwidth used by a particular virtual channel, and compares this to the bandwidth reserved for that virtual channel to calculate the precedence for that virtual channel. If the virtual channel has not used much reserved bandwidth recently, it will have a high precedence. When a data frame is sent by this virtual channel, its precedence will drop. Its precedence will increase again over a period of time. If a virtual channel has used more than its reserved bandwidth recently, it will have a low precedence.

A virtual channel specifies a portion of overall Link Bandwidth that it wishes to reserve and expects to use, i.e. its Expected Bandwidth.

When a frame of data is send by any virtual channel, each virtual channel computes the amount of bandwidth that it would have been permitted to send in the time interval that the last frame was sent. This is known as the Bandwidth Allocation. Bandwidth Allowance is calculated as follows:

$$BandwidthAllowance = Expected \times LastFrameBandwidth$$

Where Expected or Expected Bandwidth Percentage is the portion of overall link bandwidth that a virtual channel wishes to use, and Last Frame Bandwidth is the amount of data sent in the last data frame.

Each virtual channel can use this to determine its Bandwidth Credit, which is effectively the amount of data it can send and still remain within its Expected Bandwidth. Bandwidth Credit is the Bandwidth Allowance less the Bandwidth Used accumulated over time.

Bandwidth Credit is calculated for each virtual channel as follows:

$$BandwidthCredit = \sum_{Frames} \frac{BandwidthAllowance - UsedBandwidth}{Expected}$$

Where Used Bandwidth is the amount of data sent by a particular virtual channel in the last data frame, which is zero except for all virtual channels except for the one that sent the last frame.

The Bandwidth Credit is updated every time a data frame for any virtual channel has been sent. A Bandwidth Credit value close to zero indicates nominal use of bandwidth by the virtual channel. A negative value indicates that the virtual channel is using more than its expected amount of link bandwidth. A positive value indicates that the virtual channel is using less than its expected amount of link bandwidth.

To simplify the hardware required to calculate the Bandwidth Credit it is allowed to saturate at plus or minus a Bandwidth Credit Limit, i.e. if the Bandwidth Credit reaches a Bandwidth Credit Limit it is set to the value of the Bandwidth Credit Limit.

When the Bandwidth Credit for a virtual channel reaches the negative Bandwidth Credit Limit it indicates that the virtual channel is using more bandwidth than expected. This may be recorded in a status register and used to indicate a possible error condition. A network management application is able to use this information to check correct utilisation of link bandwidth by its various virtual channels.

For a virtual channel supporting bandwidth reserved QoS, the value of the bandwidth counter provides the precedence value for that virtual channel.

The operation of a bandwidth credit counter is illustrated in Figure 2.

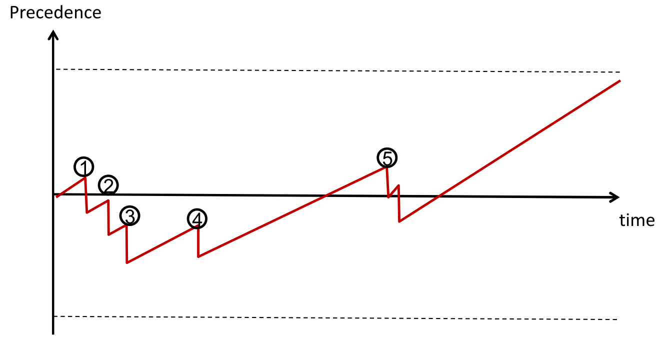

Figure 2 Bandwidth Credit Counter

The bandwidth credit for a particular VC increments gradually. At point (1) a frame is sent from by this VC, resulting in a sudden drop in credit. The size of the drop is amount of data sent in the frame divided by the percentage bandwidth reserved for the VC. This means that the smaller the percentage bandwidth the larger the drop, and hence the longer it takes to regain bandwidth credit.

After the drop at point (1) the bandwidth credit gradually increments until point (2) when another frame is sent by the VC. Further frames are sent at points (3), (4), (5) etc. If the frames sent are full frames then the drop in bandwidth credit every time a frame is sent, will be the same size.

The bandwidth credit counter for another VC is illustrated in Figure 3. This VC has about half the bandwidth of the VC in Figure 2 allocated to it. This means that the drops in bandwidth credit when frames are sent by this VC are about twice the size, as can be seen Figure 3 at points (1), (2) and (3).

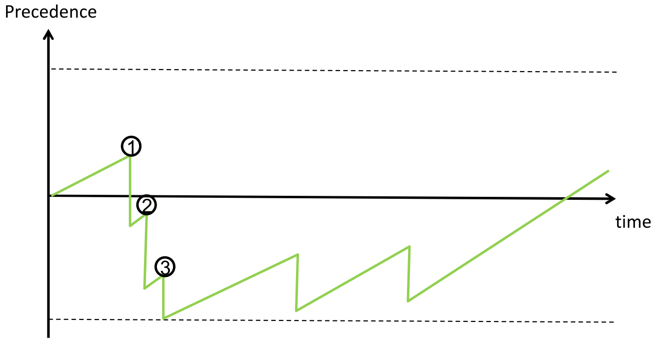

Figure 3 Bandwidth Credit Counter with Smaller Reserved Bandwidth

The bandwidth credit counter of another VC is shown in Figure 4. In this case the bandwidth credit slowly increments and although some frames are sent at points (1), (2) and (3), the bandwidth credit eventually saturates, reaching its maximum permitted value at point (4). Although more bandwidth should be accumulated after point (4) this is effectively ignored since the maximum possible bandwidth credit has been reached. At point (5) a frame is sent once more, resulting in a drop from the maximum bandwidth credit value.

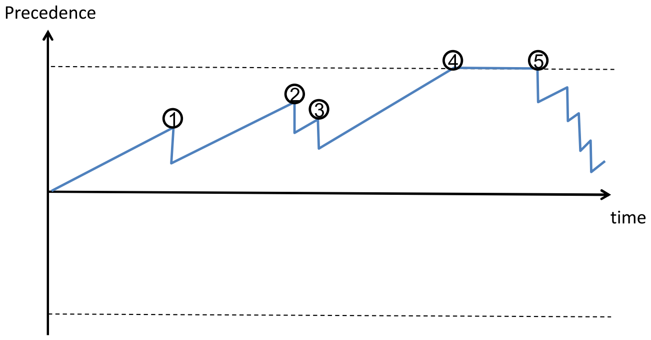

Figure 4 Bandwidth Credit Counter Reaching Saturation

All three VCs are shown together in Figure 5. When a VC has a data frame ready to send and room for a full data frame at the other end of the link, it competes with any other VCs in a similar state, the one with the highest bandwidth credit being allowed to send the next data frame. At points (1), (2) and (3) the red VC has data to send and sends frames. At points (4), (5) and (6) the green VC has data to send and sends a data frame. At point (7) both the blue and the red VCs have data to send. The blue VC wins since it has the highest bandwidth credit count. After this the red VC is allowed to send a further data frame at point (8).

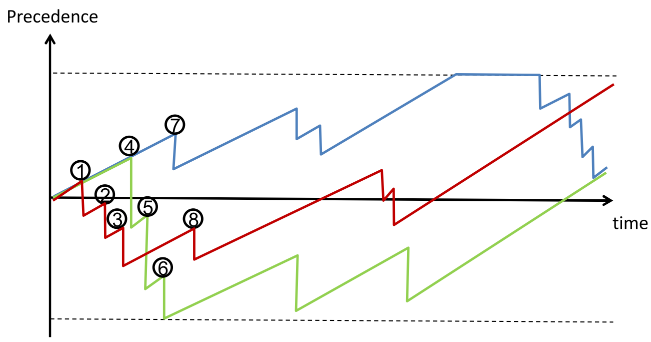

Figure 5 Bandwidth Credit of Competing VCs

If the bandwidth credit counter reaches the minimum possible bandwidth credit value, it indicates that it is using more bandwidth than expected and a possible error may be flagged. This condition may be used to stop the VC sending any more data until it recovers some bandwidth credit, to help with “babbling idiot” protection.

Similarly if the bandwidth credit counter stays at the maximum possible bandwidth credit value for a relatively long period of time, the VC is using less bandwidth than expected and this condition can be flagged to indicate a possible error.

The bandwidth credit value is the precedence used by the medium access controller to determine which VC is permitted to send the next data frame.

D. Priority

The second type of QoS provided by VCs is priority. Each VC is assigned a priority value and the VC with the highest priority (lowest priority number) is allowed to send the next data frame as soon as it is ready. Figure 6 shows three priority levels. SpaceFibre has 16 priority levels.

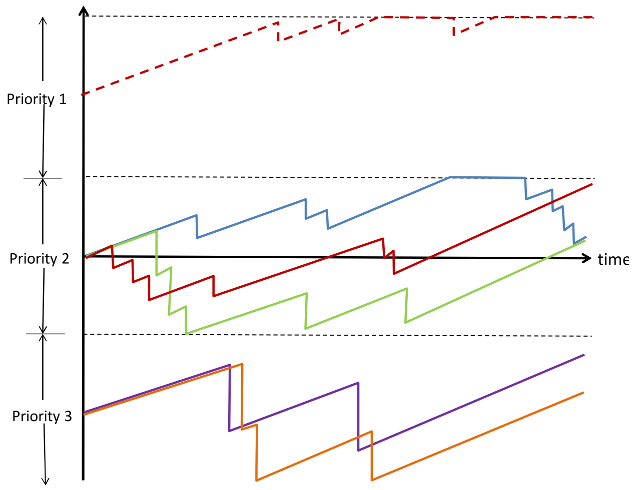

Figure 6 Multi-Layered Precedence Priority QoS

Within any level there can be any number of VCs which compete amongst themselves based on their bandwidth credit. A higher priority VC will always have precedence over a lower priority VC unless its Bandwidth Credit has reached the minimum credit limit in which case it is no longer allowed to send any more data frames. This prevents a high priority VC from consuming all the link bandwidth if it fails and starts babbling. More than one VC can be set to the same priority level in which case those VC's will compete for medium access using bandwidth reservation.

6.5 Scheduled

To provide fully deterministic data delivery it is necessary for the QoS mechanism to ensure that data from specific virtual channels can be sent (and delivered) at particular times. This can be done by chopping time into a series of time-slots, during which a particular VC is permitted to send data frames. This is illustrated in Figure 7.

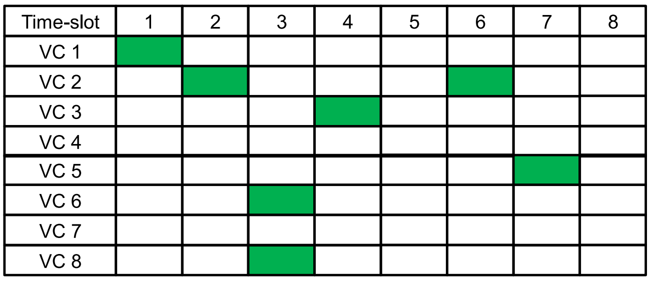

Figure 7 Scheduled Quality of Service

Each VC is allocated one or more time-slots in which it is permitted to send data frames. VC1 is scheduled to send in time-slot 1 and VC2 is scheduled to send in time-slots 2 and 3. The time-slot duration is a system level parameter, typically 100 µs, and there are 256 time-slots.

During a time-slot, if the VC is scheduled to send in that time-slot, it will compete with other VCs also scheduled to send in that time-slot based on precedence (priority and bandwidth credit). A fully deterministic system would have one VC allowed to send in a time-slot.

The schedule is always operating. If a user does not want to use scheduling the schedule table is simply filled completely, allowing any VC to send in any time-slot, competing with precedence.

Scheduling can waste bandwidth if only one VC is allowed to send in a time-slot and that VC is not ready. To avoid this situation, the critical VC can be allocated a time-slot and given high priority. Another VC can be allocated the same time-slot with lower priority. In this way when that time-slot arrives the high priority VC will be allowed to send its data, but if it is not ready the VC with lower priority can send some data. This configuration is illustrated in Figure 7 time-slot 3 and VCs 6 and 8.

Time-slots can be defined using broadcast messages to send start of time-slot signals or to send time information and having a local time counter which determines the start and end of each time-slot. The SpaceFibre broadcast message mechanism support both synchronisation and time distribution.

The SpaceFibre QoS mechanism is simple and efficient to implement and it provides bandwidth reservation, priority and scheduling integrated together, not as separate options. Furthermore SpaceFibre QoS provides a means for detecting "babbling idiots" and for detecting nodes that have ceased sending data when they are expected to be sending information.

IV. SpaceFibre Fault Detection, Isolation and Recovery

SpaceFibre provides automatic fault detection, isolation and recovery. When a fault occurs on a SpaceFibre link, it is detected and the erroneous or missing information resent. SpaceFibre recovers from intermittent faults very rapidly, detecting faults, recovering and resending data faster than SpaceWire disconnects and reconnects a link. The retry mechanism does not depend on time-outs, naturally adapting to different cable delays.

Fault detection is provided by checking each 8B/10B symbol for disparity errors and invalid 8B/10B codes. SpaceFibre has selected the 8B/10B K-codes it uses to have enhanced Hamming distance from data-codes. This means that a single bit error occurring in a data-code cannot result in a valid K-code used by SpaceFibre. In addition each data frame, broadcast frame, FCT, ACK and NACK are protected by a CRC.

Fault isolation is provided at various levels in SpaceFibre. AC coupling is used in the physical layer to prevent damage from faults that cause DC voltages exceeding the maximum permitted to appear on the transmitter outputs or receiver inputs. This feature also enables galvanic isolation to be implemented readily. At the Quality level SpaceFibre provides time containment, containing errors in the data frame in which they occur, and bandwidth containment, containing errors to the virtual channel in which they occur; an error in one VC does not affect data flowing in another VC. Babbling idiots are contained using the QoS mechanism described above.

Fault recovery is provided at the link level using a retry mechanism that resends data frames, broadcast frames and FCTs. The retry is very fast, uses a minimum amount of buffer memory, and adapts automatically to different link lengths. In addition to the retry mechanism the multi-lane functionality includes graceful degradation on lane failure. If a lane fails permanently, so that a retry or re-initialisation does not recover lane operation, a multi-lane system will continue using the remaining lanes available. This reduces the bandwidth available but does not stop the link operating. For critical operations an extra lane can be included and the graceful degradation will then provide automatic replacement of a faulty lane. The bit error rate (BER) of a lane is monitored and a lane reported as faulty if the (BER) is above a level which results in the effective link bandwidth being unusable. This feature allows lanes that can re-initialise successfully but which will not run for very long before having to re-initialise again, to be detected, isolated and replace by a fully functional lane.

V. SpaceFibre Networks

A SpaceFibre network uses similar packet formats, packet addressing and routing concepts to SpaceWire. The main difference is that SpaceFibre includes virtual channels.

A SpaceFibre router is illustrated in Figure 8.

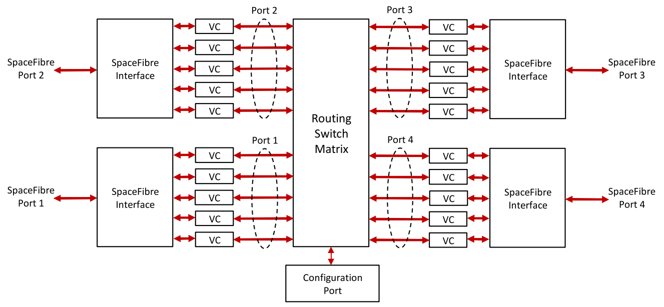

Figure 8 SpaceFibre Router

The SpaceFibre router comprises a number of SpaceFibre interfaces and a routing switch matrix. Each SpaceFibre interface has several virtual channels. The VC number for each virtual channel can be configured, except for VC0 which is a virtual channel used for configuration, control and monitoring of the SpaceFibre network. When a packet arrives on a SpaceFibre interface it is placed in the appropriate virtual channel, i.e. the one with the same VC number as it was transmitted on. The leading data character of the packet determines which port of the routing switch the packet is to be forwarded through using either path or logical addressing. The port that it is to be switched to must have a VC configured with the same number as the VC that the packet arrived on. The packet is then passed through the routing switch matrix and placed frame by frame in the VC of the output port. The packet is then transferred across the SpaceFibre link, competing with other VCs in that port for access to the link medium according to their precedence.

If a packet arrives and the output port that the packet is to be switched to does not have a VC with the same number as that on which it arrived, the packet is spilt and an error recorded.

Virtual channels can be used to construct virtual networks where a single VC number is used for connecting to all or several of the nodes attached to the network. This is illustrated in Figure 9 where VC6 (blue) is used to connect all the nodes on the network. Using VC6 the Control Processor can send commands to Instrument 1 or 2 or the Mass Memory unit, setting their operating mode or reading housekeeping information, etc. This virtual network acts like a SpaceWire network.

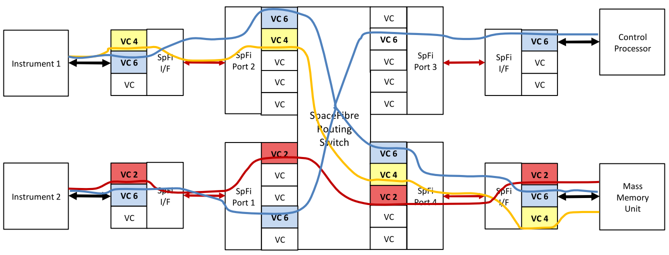

Figure 9 A Simple SpaceFibre Network

Virtual channels can also be used to construct virtual point to point links from one node to another. VC2 and VC4, in Figure 9, are providing virtual point to point links. VC2 provides a virtual point to point link between Instrument 2 and the Mass Memory Unit and VC4 between Instrument 1 and the Mass Memory. These virtual channels can be each allocated the bandwidth they need to send their data to the Mass Memory Unit. Once this bandwidth is allocated other virtual channels or virtual networks will not interfere with their operation.

Figure 10 shows a more realistic onboard network using SpaceFibre which includes a SpaceWire to SpaceFibre Bridge. Two high data-rate instruments (Instruments 1 and 2) have SpaceFibre connections. Four less demanding instruments have SpaceWire connections to the SpaceWire to SpaceFibre Bridge. Each instrument has a virtual point to point connection to the Mass Memory Unit and there is a virtual point to point connection between the Mass Memory and the Downlink Telemetry Unit. The Control Processor has a virtual network for configuring and controlling all devices on the network.

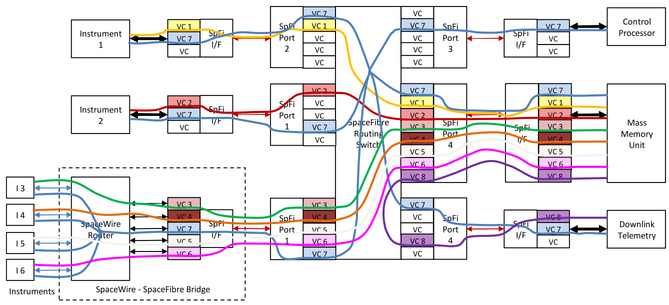

Figure 10 Realistic SpaceFibre Network

Figure 10 is solving a complex communication task with many separate, isolated virtual channels providing point to point links, and a virtual network being used to control the entire system. Figure 11 shows this same network with the virtual channels removed, revealing the simplicity of implementation of a complex communication task when using SpaceFibre.

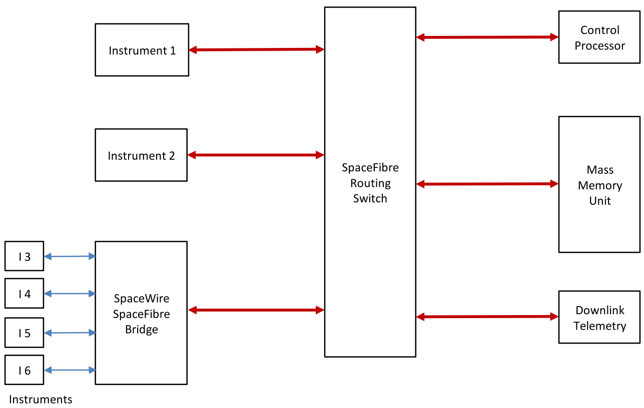

Figure 11 Simple System Architecture with SpaceFibre

VI. SpaceFibre Implementations

The SpaceFibre specification has been written by the University of Dundee for ESA, and has been widely reviewed by the international spacecraft engineering community. It has also been simulated and implemented in several forms. While work remains to be done on the specification the existing draft specification is close to maturity. In this section the current state of SpaceFibre is explored.

A SpaceFibre interface has been designed by University of Dundee and STAR-Dundee for ESA. This VHDL IP core has been used at all stages of the draft specification to validate and prove the concepts being explored. As a consequence the VHDL IP core has gone through as many iterations as the SpaceFibre specification. At present the VHDL IP core implements all layers of the SpaceFibre specification with the exception of the Multi-Lane layer.

To support the testing of SpaceFibre a suitable test platform was required, so STAR-Dundee developed the STAR Fire unit, which has two SpaceFibre interfaces and includes a link diagnostic capability for analysing traffic on a SpaceFibre link. Two STAR Fire units are being used in Figure 12 to help with the testing of radiation tolerant Fibre Optic transceivers for SpaceFibre operating over 100 m of Fibre Optic cable.

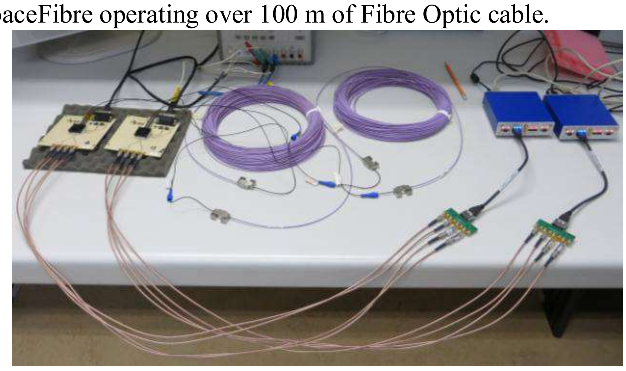

Figure 12 STAR Fire Testing 100m Fibre Optic Cable

A radiation tolerant SpaceFibre interface device (VHiSSI) is being developed by University of Dundee and several partners within a European Union (EU) Framework 7 project [6].

NEC and Melco are both developing SpaceFibre interface devices to the specification produced by the University of Dundee. This work is providing valuable feedback on the specification and implementation of SpaceFibre. Interoperability testing in December 2013 and April 2013 has been successful with various levels of the SpaceFibre protocol stack being implemented and tested.

Research carried out during the SpaceWire-RT EU Framework 7 project resulted in the Quality layer for SpaceFibre being developed by University of Dundee. Within this same project St. Petersburg University of Aerospace Instrumentation (SUAI) modelled and simulated the various layers of SpaceFibre and ELVEES assessed the feasibility of ASIC implementation using a custom designed SerDes.

Several ESA projects are using the Dundee SpaceFibre IP core under a Beta evaluation programme including:

- High Performance COTS Based Computer, Astrium and CGS.
- Leon with Fast Fourier Transform Co-processor, SSBV.
- FPGA Based Generic Module and Dynamic Reconfigurator, Bielefeld University.
- Next Generation Mass Memory, Astrium, IDA and University of Dundee.
- 1 x High Processing Power DSP, Astrium and STAR-Dundee.

Work on the formal European Cooperation for Space Standardization (ECSS) standard for SpaceFibre is schedule to start in early 2014, once the technical specification is complete.

VII. Conclusions

SpaceFibre is a multi-gigabit/s networking technology designed specifically for spaceflight applications. It incorporates a comprehensive quality of service capability providing integrated bandwidth reservation, priority and scheduling. Efficient, effective and rapid fault detection, isolation and recovery mechanisms are included in the SpaceFibre interface, enabling rapid detection and recovery from link level errors.

SpaceFibre is designed to support very high data-rate missions like multi-spectral imagers and synthetic aperture radar. It reduces development time and costs, because of its integrated QoS and FDIR capabilities and because it simplifies previously complex onboard data-handling architectures. SpaceFibre is designed to use the same packet format as SpaceWire enabling straightforward upgrading of spacecraft networks to include the improved QoS, FDIR and bandwidth of SpaceFibre while being able to operate with existing SpaceWire equipment. SpaceWire units can be readily integrated with SpaceFibre using a SpaceWire to SpaceFibre Bridge.

Acknowledgment

The research leading to these results has received funding the European Space Agency under ESA contract numbers 4000102641 and from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 263148 and 284389. We would also like to thank Martin Suess the ESA project manager for the SpaceFibre related activities for his help, advice and guidance.

References

[1] S. Parkes, A. Ferrer, A. Gonzalez, & C. McClements, “SpaceFibre Standard Draft E1”, University of Dundee, 28th September 2012.

[2] S. Parkes, A. Ferrer, A. Gonzalez, & C. McClements, “SpaceFibre: Multiple Gbits/s Network Technology with QoS, FDIR and SpaceWire Packet Transfer Capabilities”, International SpaceWire Conference, Gothenburg, June 2013.

[3] S. Parkes, “Never Mind the Quality, Feel the Bandwidth: Quality of Service Drivers for Future Onboard Communication Networks”, paper no. IAC-10.B2.6.6, 61st International Astronautical Congress, Prague 2010.

[4] S. Parkes, C. McClements and M. Suess, “SpaceFibre”, International SpaceWire Conference, St Petersburg, Russia, 2010, ISBN 978-0-9557196-2-2, pp 41-45.

[5] ECSS Standard ECSS-E-ST-50-12C, “SpaceWire, Links, Nodes, Routers and Networks”, Issue 1, European Cooperation for Space Data Standardization, July 2008.

[6] S. Parkes, A. Ferrer, A. Gonzalez, C. McClements, R. Ginosar, T Liran, G Sokolov, G Burdo, N Blatt, P Rastetter, M Krstic, A Crescenzio, “A Radiation Tolerant SpaceFibre Interface Device”, International SpaceWire Conference, Gothenburg, 2013.