SPACEFIBRE

Session: SpaceWire Standardisation

Long Paper

Steve Parkes[1], Chris McClements[1], Martin Suess[2]

[1]School of Computing, University of Dundee, Dundee, DD1 4HN, Scotland, U.K.

E-mail: sparkes@computing.dundee.ac.uk

[2]Martin Suess, ESTEC, Noordwijk, The Netherlands

ABSTRACT

SpaceFibre is a proposed very high speed serial data link intended to complement the existing SpaceWire high-speed data link standard. SpaceWire operates at speeds up to 200 Mbits/s in radiation tolerant technology. SpaceFibre will be able to operate over fibre optic and copper cable and support data rates of 2.5 Gbit/s and possibly higher.

This paper describes the work done by University of Dundee in developing SpaceFibre. It starts by looking at lessons learnt from the development of SpaceWire. The principal requirements for SpaceFibre are then listed and the baseline architectural design presented. It concludes with an overview of the SpaceFibre demonstration system developed by University of Dundee.

LESSONS LEARNT FROM SPACEWIRE

SpaceWire [1] [2] is an effective onboard communications link which is being used on a range of ESA, NASA, JAXA and other space missions. It does however have a number of issues that should be addressed in the development of a new onboard communication standard.

1. **Cable Mass:** SpaceWire uses a data-strobe signalling technique [1]. The data and strobe signals are differentially encoded using LVDS so that two twisted pairs are required for communications in each direction, giving a total of eight wires. This approach was adopted in preference to a self-clocking data stream because it removed the need for a phase-locked loop in the receiver and made implementation of the SpaceWire CODEC possible in any ASIC or FPGA technology. It does mean, however that the cable contains more wires and is thus heavier (~87g/m) than would otherwise be necessary. SpaceWire cable is specially manufactured for flight applications. It should be noted that the data-strobe technique gives better skew tolerance than a simple data-clock method.
2. **Data Rate:** The speed of transmission is limited by two primary factors: the attenuation in the cable at high frequencies and the skew between data and strobe signals. Attenuation reduces the amplitude of the SpaceWire signals as they

propagate down the cable making them more susceptible to noise. Skew between data and strobe pairs affects the signal timing reducing the maximum data rate that can be supported. The longer the cables the worse these effects become. SpaceWire will work at data rates of 200 Mbits/s over distances of 10 m. Shorted cables allow higher data rates while longer cables can only support lower data rates.

3. **Character Sizes:** There are two types of character sent over a SpaceWire link: control and data characters. In addition there are two control code sequences made up from control and data characters (NULL and time-code) [1]. This results in four different length codes sent over the SpaceWire link: control 4-bit, NULL 8-bit, data 10-bit and time-code 14-bit). Handling these different sized characters complicates the transmitter and receiver circuitry. The original intention (I believe) with having control characters shorter than data characters was to save link bandwidth when FCTs were sent and also, less importantly, when EOP markers were sent. In SpaceWire an FCT is exchanged for eight data-characters, which results in a 4-bit overhead to send 80-bits data (encoded). Note that the overhead is actually in the opposite direction of the link to the direction that the data is travelling. This is a 5% overhead. The same level of overhead could be achieved by exchanging one 10-bit FCT for every 200 bits data i.e. 20 data characters. This would increase the minimum amount of buffering needed, which was a possible issue in the original IEEE1355-1995 standard [3] from which SpaceWire was derived, but is not an issue today. It would allow the control characters to be the same size as the data characters simplifying transmit and receive circuitry.

4. **Parity Coverage:** Parity coverage in SpaceWire is rather peculiar. The parity bit covers the data/control field from the previous character and the data/control flag from the next character. This approach complicates both the transmitter and receiver because two characters have to be considered together to determine the parity value. The reason that this was done in IEEE1355-1995 is not known.

5. **Transmitted DC Component:** SpaceWire characters use all possible bit patterns of the 10-bit data and 4-bit control characters. This means that depending on the data pattern sent there will be a DC component to the transmitted signal. For example, if a stream of data characters of value 0xff is sent, there will be a positive DC bias. This degrades the transmission characteristics of the SpaceWire signals making them broadband and prevents AC coupling from being used.

6. **Galvanic Isolation:** SpaceWire does not provide a method of galvanic isolation. A technique using capacitive coupling and bus-hold circuits has been proposed for SpaceWire [4] as used in IEEE1394 [5] but has not been prototyped. A galvanic isolation unit has been designed at Dundee to support ground based isolation at speeds up to 100 Mbits/s.

7. **Matched Impedance Connectors:** The 9-pin micro-miniature d-type connectors used in SpaceWire are not controlled impedance. This can cause problems at

higher speeds when there will be reflections from the impedance mismatch adding noise and jitter to the SpaceWire signals. A matched impedance connector is essential at higher speeds than 200 Mbits/s. The 9-pin micro-miniature d-type connector was used in SpaceWire because it was readily available in space qualified form and was able to operate with the data rates that SpaceWire was then intended to work at.

8. **Initialisation Protocol:** The initialisation protocol in SpaceWire is based on part handshake and part timing. This can lead to false initialisation caused by noise and data characters being sent due to noise, before a parity error or other error is detected and the link is terminated. This approach was taken to allow backwards compatibility with IEEE1355-1995 and the SMCS devices. A full handshake scheme which did not use flow control tokens (FCTs) would be better and prevent the flow of data characters before a proper link connection has been made. Note that the use of bias resistors on the LVDS receiver inputs can provide a noise threshold which will eliminate the false start due to noise problem in most systems.
9. **Link Speed Negotiation:** The initial link speed is fixed in SpaceWire to 10 Mbits/s. This link speed is always used for link initialisation. Once a connection has been made either or both ends of the link can switch to different speeds. There is no link speed negotiation protocol in SpaceWire, so that if a low speed device is connected to one operating at higher speed the two devices will start communication at 10 Mbits/s, make a connection, then switch to their operating speeds. At this point the slower device will fail to receive the characters from the high speed device correctly resulting in an error and disconnection of the link. A link negotiation protocol would allow a more controlled setting of the link operating speed so that when a high speed device is connected to a lower speed device, they operate at mutually acceptable data rates.
10. **Initialisation Speed:** Because the initialisation protocol has a fixed timing and because the initial link speed is fixed, the time taken to initialise a SpaceWire connection is around 20 us. If data is being transferred at 20 Mbytes/s corresponding to 200 Mbits/s data signalling rate, then the 20 us initialisation time is equivalent to 400 bytes of data. If a link level retry mechanism is being implemented then each router or node will need to store at least this amount of data in the event of a temporary fault (e.g. parity error) occurring in a continuous data stream. In practice the buffer storage could be significantly higher than this. If the link is to always operate at high speed (e.g. 200 Mbits/s) then a faster disconnect and re-initialisation interval would result in less buffer space being needed for a retry mechanism. This approach is being considered by GSFC for JWST.
11. **Transport Layer:** SpaceWire lacks a transport layer so there is no consistent way of managing a connection in SpaceWire: buffer management and end-to-end flow control, fault recovery and packet retransmission are all missing.

While addressing these issues the important features of SpaceWire should not be lost:

- Simplicity
- Low implementation cost (gate count)
- Bi-directional, full-duplex communication
- Time-code distribution
- Group adaptive routing

Requirements

SpaceFibre aims to extend the capabilities of SpaceWire: improving the data rate by a factor of 10, reducing the cable mass by a factor of four and providing galvanic isolation. It will also address the other issues with SpaceWire.

The principal requirements for SpaceFibre are listed below:

- Provide symmetrical, bi-directional, point-to-point link connection
- Operate at high speed (1-10 Gbits/s) with a target of at least 2.5 Gbits/s
- Operate over fibre cable lengths of up to 100 m
- Also operate over copper cable over shorter length of 5 m
- Have a fibre optic cable mass of less than 20 g/m for a full-duplex link
- Provide galvanic isolation
- Support arbitrary network architectures
- Support mixed SpaceWire-SpaceFibre networks using a SpaceWire-SpaceFibre router
- Be able to multiplex a scalable number of SpaceWire links over a SpaceFibre link

Architectural Design

The baseline concept for the SpaceFibre interface is illustrated in Figure 1.

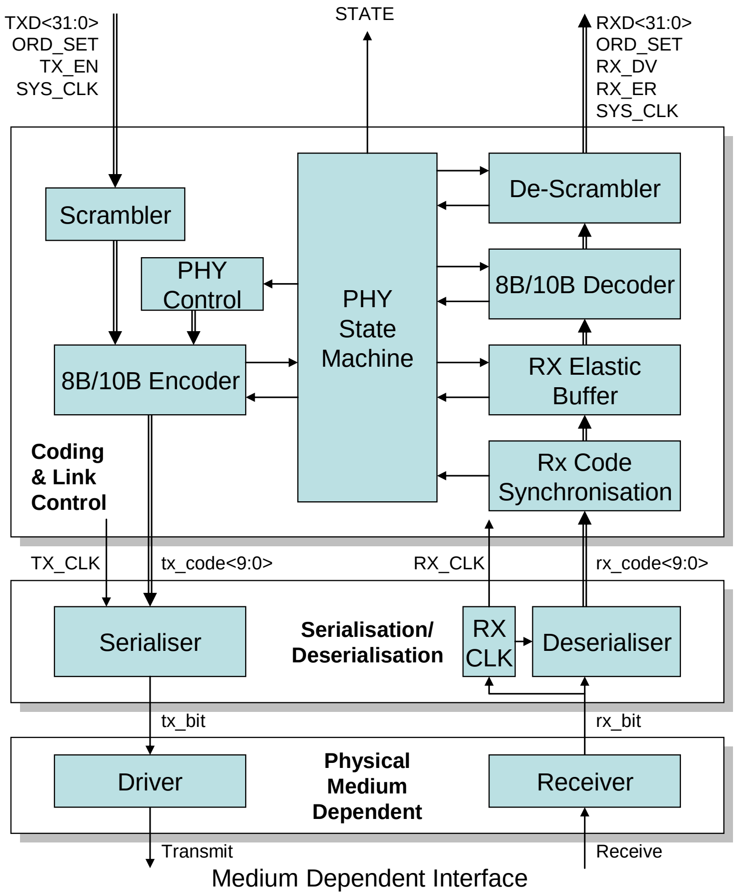

Figure 1 Baseline SpaceFibre Interface

This block diagram shows the possible interfaces that may be exposed.

Data words (32-bits) or ordered sets to be transmitted are passed though the transmit side of the port interface. They are first scrambled by a data scrambler which is re-seeded at the start of a frame. The scrambled data is then encoded byte by byte by the 8B/10B encoder [6] and passed on to the Serialiser. The bit stream from the Serialiser is driven onto the transmission medium by an appropriate medium dependent driver.

Whenever there is no data or ordered sets to transmit, idle ordered sets are automatically added to the data stream.

The bit stream is recovered from the medium by first converting the medium dependent signal into a digital signal. A receive clock is generated from this digital signal using a phase-locked loop and is used to recover the transmitted bit stream which is converted to 10-bit words using the de-serialiser. Using the 8B/10B "comma" contained in the idle ordered sequences the receive code synchronizer synchronizes to 10-bit word boundaries and passes a correctly aligned 10-bit word to the receive elastic buffer. The elastic receive buffer copes with any differences in system clock and receive clock by adding or deleting Idle ordered sets in the elastic buffer. This means that the interface to the system is synchronous to the system clock. The output from the elastic receive buffer is passed to the 8B/10B decoder. This decodes the 10-bit word to an 8-bit wide data stream plus comma codes. The decoded data stream is fed to a descrambler which recovers the data originally sent.

A link control state machine is responsible for link initialization, link-level flow control and for responding to errors.

SpaceWire time-codes may be transmitted using a subset of the ordered sets.

SPACEFIBRE PHYSICAL LAYER

Patria New Technologies Oy, VTT, INO, Fibre Pulse and Gore have been working on the physical fibre optic components for SpaceFibre with some encouraging results. INO selected and tested a fibre optic cable core which is radiation tolerant. Fibre Pulse recommended suitable robust fibre optic connectors. Gore assembled cables and connectors into rugged cable assemblies. VTT developed a fibre optic transceiver that can operate at well over 2.5 Gbits/s and is thought to be radiation tolerant. These physical components have been tested by Patria New Technologies Oy. These fibre optic components can be seen on the right hand side of Figure 2.

Figure 2 SpaceFibre Demonstration Unit

SPACEWIRE-SPACEFIBRE ROUTER

The prototype SpaceFibre CODEC has been designed in a SpaceWire-SpaceFibre router which sends SpaceWire packets over SpaceFibre. This prototype system was implemented on a Xilinx Virtex 4 device using an ML405 prototyping board. The SpaceFibre interface is implemented using the RocketIO facilities of the Xilinx FPGA device. In this demonstration system four SpaceWire ports can be multiplexed over a single SpaceFibre link which operates at 2 Gbits/s. The reduced speed compared to the target speed of 2.5 Gbits/s is due to a problem with the ML405 prototyping board. A block diagram of the design in the Xilinx FPGA is illustrated in Figure 3.

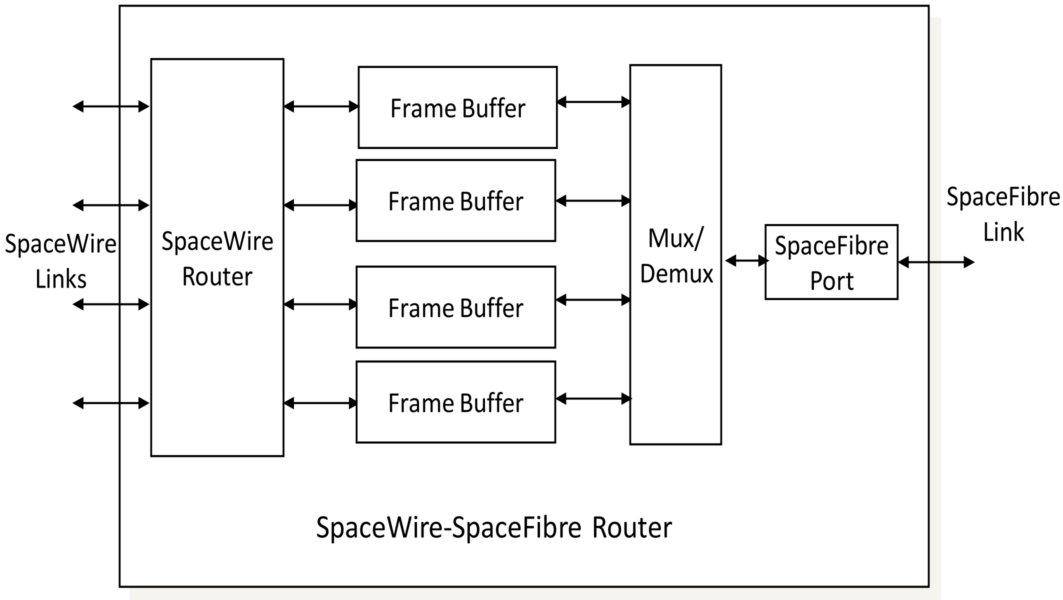

Figure 3 SpaceWire-SpaceFibre Router

The four SpaceWire links feed into an eight-port SpaceWire router which has four SpaceWire ports and four internal ports connected to the SpaceFibre interface. The internal ports are fed into frame buffers, which collect SpaceWire data-characters, EOPs, EEPs and time-codes and prepare them for sending over SpaceFibre. The prepared frames are multiplexed by the multiplexing/de-multiplexing block and fed into the SpaceFibre interface for transmission over SpaceFibre.

Frames arriving over SpaceFibre are de-multiplexed, and passed into frame buffers going the other way. Information is read out of these frame buffers and passed into the SpaceWire router and out of the appropriate SpaceWire port.

The ML405 board and SpaceWire connectors can be seen on the left hand side of Figure 2.

CONCLUSIONS

SpaceWire has been very successful for spacecraft onboard data handling and is now being widely used. There is a need now for a higher speed data link running at well over 1 Gbit/s. The development of SpaceFibre offers an opportunity to benefit from the widespread use of SpaceWire and also to overcome some of its deficiencies. The University of Dundee has developed an appropriate SpaceFibre CODEC. The next step is to provide a draft SpaceFibre standard document for wider review.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support of ESA for the work done on the SpaceFibre at the University of Dundee, which was done under ESA Contract Number 17938/03/NL/LvH.

REFERENCES

[1] European Cooperation for Space Standardization, Standard ECSS-E-50-12A, “SpaceWire, Links, Nodes, Routers and Networks”, Issue 1, European Cooperation for Space Data Standardization, January **2003**.
[2] European Space Agency, “SpaceWire Web Page”, European Space Agency, http://spacewire.esa.int/.
[3] IEEE Computer Society, “IEEE Standard for Heterogeneous Interconnect (HIC) (Low-Cost, Low-Latency Scalable Serial Interconnect for Parallel System Construction)”, IEEE Standard 1355-1995, IEEE, June 1996.
[4] Parkes SM et al, “Review of Standard and Status”, Digital Interface Circuit Evaluation Study WP1000 Technical Report, Document No. UoD-DICE-TN-1000, ESA Contract No. 12693/97/NL/FM, University of Dundee, July 1998.
[5] IEEE Computer Society, “IEEE Standard for a High Performance Serial Bus”, IEEE Standard 1394-1995, IEEE, August 1996.
[6] Kembel, R.W., “Comprehensive Fibre Channel”, Solution Technology, Northwest Learning Associates Inc, **2002.**