CEAS Space Journal (2023) 15:451–465
https://doi.org/10.1007/s12567-022-00454-z

ORIGINAL PAPER

SpaceWire communications in power over ethernet environment

Jack Martin[1] · Yaseen Zaidi[1]

Received: 15 September 2021 / Revised: 7 May 2022 / Accepted: 12 May 2022 / Published online: 21 June 2022

Abstract

This is a high-level study into the validation of a unified electric power and communications architecture for modern spacecraft systems. We leverage from blending the European Cooperation for Space Standardisation (ECSS) SpaceWire and the IEEE 802.3 Power over Ethernet (PoE) standards for a common onboard power and communications interface. The focus of the work has been communication performance. The requirements from the protocols of both standards were analysed and combined to create a full protocol stack for SpaceWire PoE. The stack is implemented on an embedded platform represented as a proof-of-concept SpaceWire PoE system. The results of functional testing demonstrate SpaceWire PoE as a valid solution that meets the requirements of both protocols. For the 100 Mb/s links, the SpaceWire PoE throughput is on a par (97.39%) with Ethernet (97.52%) and SpaceWire (99.2%). Shared communication and power architecture is intended for decentralised operation achieving greater autonomy of the integrated subsystems. This potentially may lead to standardisation of electrical, mechanical and communication interfaces across spacecraft subsystem manufacturers, easing harness complexity and routing during the assembly, integration and test (AIT) phase of the spacecraft as well as saving mass and launch cost.

Keywords Ethernet · On-board communication · Power over Ethernet · Protocol · Spacecraft · SpaceWire

1 Introduction

To get more lift, the mass and power savings in commercial aircraft design and compactness demand in electric and hybrid aircraft have led the aircraft industry to transition from federated to integrated module architecture [1]. This is an ongoing activity expected to last many years. The space industry has also been on the outlook for new concepts of system architectures that are based on interoperability and reconfigurability for performance, efficiency, and cost savings. The operational capability and sustainability of current spacecraft are constrained by their inherent physical system architecture, highly customised monolithic design, and the limited or no availability of servicing and maintenance.

Modular Spacecraft Assembly and Reconfiguration (MOSAR) [2] is a European Union (EU) funded initiative made by an international consortium that aims to raise the degree of modularity of space systems by an order of magnitude to the current practice. The MOSAR concept is an in-orbit reconfigurable spacecraft that can benefit from the plug and play and common interfaces whether in hardware, software, communication, or computation. MOSAR complements the modular and efficiency-focused culture within the space industry. An area of opportunity identified in MOSAR is a common architecture that could provide two functionalities: intercommunications and the distribution of power throughout the spacecraft. This would result in spacecraft requiring less material for construction and being reusable due to the ability to reconfigure the system architecture for a variety of mission operations as well as easing the hardware and software integration.

Current spacecraft systems lend to bespoke and/or proprietary methods of deploying communication networks such as MIL-STD 1553B and Controller Area Network (CAN 2.0) [3]. A full protocol stack could provide the underpinning intercommunications between future spacecraft subsystems while running over a common physical medium that distributes electric power.

The space industry appreciates the use of modular components to reduce mission costs and development time. Plug and play avionics allow devices to be added to the

✉ Yaseen Zaidi
yaseen.zaidi@uwe.ac.uk

Jack Martin
Jack_M_94@hotmail.co.uk

[1] School of Engineering, University of the West of England, Bristol, UK

spacecraft, replaced, or upgraded later in the manufacturing process. Thales Alenia Space has developed a SpaceWire and SpaceFibre [4] network interface that offers full galvanic isolation between modular devices. The project goal is to develop a space-suitable power delivery solution and an extension to the plug and play avionics such that power negotiation and management can take place within the modular network. Moreover, power delivery to a remote endpoint over Ethernet is required to be consistent with the existing spacecraft power standard ECSS-E-ST-20-20C. The present project builds on this work by adding Power over Ethernet (PoE) capabilities to communication links. However, before the delivery of power is made, the communication proof-of-concept is ensured. Therefore, the focus of this paper is the development of a handshaking protocol implemented for an embedded microprocessor of an onboard command and data handling (C&DH) system or on-board computer (OBC).

2 SpaceWire and IEEE 802.3 power over ethernet (PoE)

SpaceWire was introduced in 2003 and since then it has been implemented in several missions of major space agencies. The SpaceWire protocol defines data handling for both low-rate on-board communications such as housekeeping and high-rate science data interfaces such as mass memory, processor busses and interfaces to high-speed download antennas. The protocol specifies network routing and switches for increased onboard connectivity. The SpaceWire physical layer defines the electrical characteristics of the signal transmission such as differential signalling, skew, losses, voltage levels and termination impedances between the transmitter and the receiver. This includes wires, cables, assemblies and even printed circuit traces to efficiently carry the high-speed data achieving good signal integrity.

On the other hand, Ethernet an invention of Xerox Corp. is the de facto standard for terrestrial short distance networks since the 1970s. Ethernet has gone through a tremendous evolution considering the need for speed. Right about when SpaceWire was debuted, the IEEE 802.3af PoE interface came into existence for concurrent Ethernet data communications and delivery of up to 13 W of power to any device at 13 m. For meeting green aviation and sustainability goals, PoE [5] has been suggested for delivering power over fast Ethernet in linear or star network topologies. In commercial aviation, powering of non-avionics such as cabin lamps, flight entertainment systems and routine telemetry collection have been proposed over Ethernet [6]. Microsemi develops an array of modules and integrated circuits for powering non-critical flight systems over Ethernet.

SpaceWire and PoE are intended for two very different applications. However, the commonality is a layered approach of a network. This empowers SpaceWire to ride on the Ethernet stack which in turn has the capability of transmission over a shared physical medium for simultaneous Ethernet communication and power transfer. Such technology could leverage the high interoperability by replacing units in a spacecraft when late mission decisions are made. For instance, a different attitude determination and control unit may be plugged in during the assembly, integration and test (AIT) phase of the spacecraft that has power and data interfaces for a different sensors and actuators configuration than the one conceived originally. Similarly, a high data rate and high-power radiofrequency system may be substituted. Smaller AIT cycles, high reliability and mass, area, volume and cost savings would be added advantages. Naturally, the development of such technology is a significant undertaking given the complexity of diverse technologies involved and the severity of the space environment. The present work is a feasibility study of validating SpaceWire communications over Ethernet. This is done with a technology demonstrator constructed with Commercial Off The Shelf (COTS) hardware. The physical medium is not the standard Ethernet but the PoE. However, we do not show live powering ON/OFF of the devices on the network. Such a study is beyond the scope of this write-up.

2.1 An overview of the protocols

The ECSS SpaceWire specification is defined within the ECSS-E-ST-50-12C Rev. 1 Standard [7]. The rationale behind SpaceWire was to standardise the proprietary and ad hoc approach adopted by space equipment manufacturers for onboard data handling and communications. SpaceWire mandates the use of layers of the Open Systems Interconnection (OSI) concept, notably the physical, datalink, network, and application layers. A SpaceWire network facilitates telemetry, telecommand and payload/science data transfers to mass memory and provides interfaces to units on standard interfaces such as CAN [8].

The IEEE 802.3 Standard [9] is an international standard for local and metropolitan area networks, employing Carrier Sense Multiple Access/Collision Detection (CSMA/CD) and the Ethernet protocol for data communication. The Ethernet protocol implements the physical and datalink layer for data communications and the sublayers Media Access Control (MAC) and Logical Link Controller (LLC) make up the datalink layer of the OSI reference model.

The IEEE 802.3 standard defines the distribution of power over an Ethernet-based network, in addition to data transmission within a network. The latest PoE capability is stipulated through IEEE 802.3bt [10] utilising the 4-pair twisted pair cabling that has been qualified to be used within the standard. A PoE system comprises three elements, these are power supply defined as the Power Sourcing Equipment

(PSE), the powered load defined as the Powered Device (PD) and twisted-pair cabling connecting the two devices. This functionality is illustrated in Fig. 1. There are up to 8 classes of power configurations. The PD may request any class of power from the PSE ranging from 4 to 90 W.

2.2 Related research

Modularity has been sought after for a long time in spacecraft systems [11]. The majority of schemes have remained restricted and proprietary to the spacecraft bus manufacturers who attempt to achieve reconfigurable and reusable platforms accommodating a variety of payloads. US Air Force's plug and play satellite PnPSat-1 [12] was the most recent notable effort. Concerning the communication aspect of modularity in the commercial and interplanetary missions, SpaceWire has flown on BepiColombo [13], Astro-H [14], Geostationary Operational Satellite R series [15] and several other missions. Ethernet has been considered for on-board communications [16] as well. For terrestrial applications, PoE is an established concept. However, in the space sector, this idea is still relatively new and therefore significant literature is lacking. The SpaceWire concept is well-accepted though. Within the community of interest for SpaceWire applications, there have been studies on how it can be more efficient. These studies include broader quality of services such as:

i. Distribution of SpaceWire time codes for spacecraft synchronisation [17].
ii. Synchronisation of SpaceWire interrupt codes delivery in onboard networks [18].
iii. SpaceWire network management and discovery [19].

The implementation of transmitting SpaceWire packets over Ethernet has been investigated in literature [20, 21] identifying the methods, limitations, and issues that arise. The overarching issue is the limitation of the format of the Ethernet frame containing a fixed-sized header, payload, and error checking sections that in some scenarios are inefficient. SpaceWire packets, in principle, can have infinitely long payload sizes whereas Ethernet can only transmit a fixed length of between 46 and 1500 bytes. Therefore, it can be recognised that for SpaceWire payloads larger than 1500 bytes further processing within the network layer is required to re-build segmented data payloads.

The constraint of Ethernet frames also requires further processing to determine the SpaceWire datatype that is contained within the Ethernet payload, including N-Char data and broadcast codes which consist of time codes and interrupt codes. An interpretation of how this can be implemented is given in [22]. According to that illustration, the necessary artificial headers are inserted to allow the identification and size of the SpaceWire data being sent, and subsequently how this is handled within the receiving element via further processing.

The requirement of further SpaceWire headers to be inserted within the Ethernet frame resultantly has a detrimental impact on the efficiency of the communications link. Rozanov and Yablokov [21] conducted studies and depicted graphically the relationship between the number of bytes sent for the payload for both the Ethernet and SpaceWire protocols. They showed that the viability of the Ethernet protocol is only applicable where payloads of more than 30–40 bytes are transmitted per packet. Therefore, for the implementation of the combined SpaceWire and Ethernet protocol, this characteristic must be considered for the intended use case of spacecraft intercommunication.

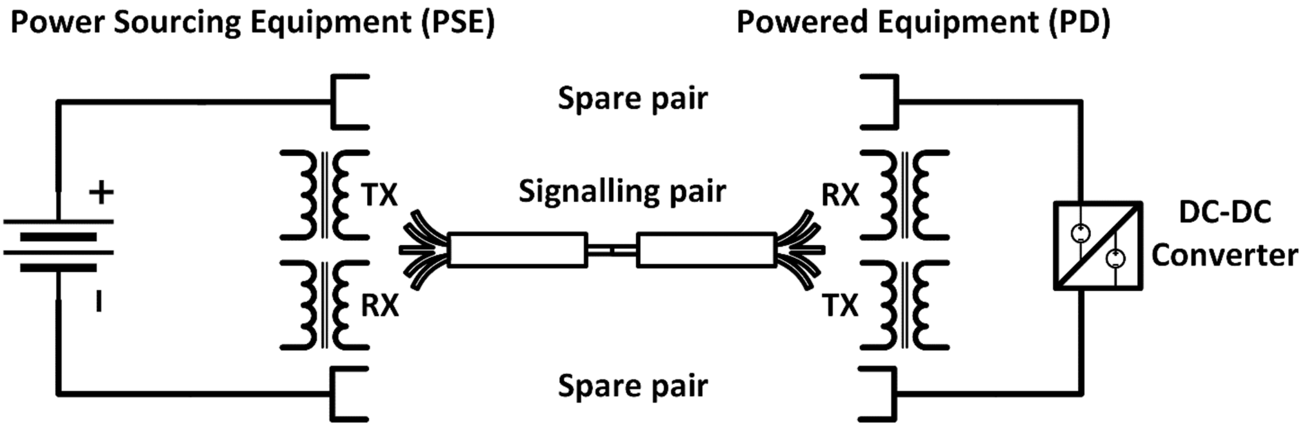

Fig. 1 IEEE Ethernet PoE concept for powering a device from a source over twisted pair

3 SpaceWire PoE protocol stack

3.1 Full protocol stack proposal

The implementation of an integrated SpaceWire PoE protocol stack was identified by reviewing the existing layers of the Ethernet and SpaceWire protocols. Considering the OSI model, the full protocol stack can be described at a high level as shown in Table 1; the datalink and physical layer would be achieved by utilising the Ethernet protocol, and the network layer of the SpaceWire protocol. Additional layers of the OSI model, namely the application, presentation, transport and session layers are not required for the operation of the SpaceWire PoE protocol. The application and presentation layers were utilised for the development and functional testing of the SpaceWire PoE concept demonstrator only.

3.2 Network layer implementation

The network layer of the SpaceWire PoE protocol stack is responsible for the formatting of the individual Ethernet frames. It was identified that further processing is required for the transmission of SpaceWire packets within Ethernet frames. Therefore, an artificial header one byte in size has been added within the payload of the Ethernet frame to address the type of data between an N-Char and control code as required in the SpaceWire protocol. Artificial headers and their respective values and meaning for the proposed SpaceWire PoE protocol stack are shown in Table 2. The table details the SpaceWire protocol value of how the packet is identified, the SpaceWire PoE adopted byte value and the functional description of each artificial header byte.

Figure 2 diagrammatically shows the proposed packet format for the SpaceWire PoE protocol. The coloured elements show the origin of the header requirements for the packet from each protocol. The proposed packet format is for the use case of transmitting N-Char data, where the minimum amount of data that can be sent is 0 bytes (46 bytes with padding) and a maximum of 1498 bytes per packet to meet the requirements of the Ethernet frame.

Sending one SpaceWire packet per Ethernet frame is not the most efficient method of transmission, therefore, multiple SpaceWire packets with the appropriate header byte and EOP marker can be inserted into the Ethernet payload section. Further processing is required at the network level to separate the received frame into the individual SpaceWire packets transmitted.

Table 1 High-level functional description of the SpaceWire PoE protocol stack

OSI layer	Layer implementation	Description/function
Application	User GUI/embedded platform	Shall act as the User Interface for manual requests of data and control codes to be distributed to SpaceWire nodes
Presentation	Embedded platform	Shall take the data from the network layer (packets) and encode them into the appropriate format depending on the purpose. ASCII characters to be used for readability
Session	Not used	
Transport	Not used	
Network	SpaceWire	Shall format the data and control codes received from the User Interface or other SpaceWire Nodes through the datalink layer into the SpaceWire packet format
Datalink	IEEE 802.3 Ethernet	Shall establish and maintain a link with other Ethernet nodes on the network Shall encode the SpaceWire packets received from the network layer into frames using Manchester encoding (10 Mb/s) or 4B/5B NRZ-I encoding (100 Mb/s) Shall decode the bitstream received from the physical layer into frames to be passed up to the network layer to receive the SpaceWire packet Shall direct the frames to the required destination address using the MAC protocol
Physical	IEEE 802.3 Ethernet	Shall drive and receive the bitstream data signals through the twisted pair cabling using differential signalling

Table 2 Artificial header bytes for SpaceWire PoE protocol

Artificial header byte name	SpaceWire	SpaceWire PoE Value	Function description
EOP_HEADER	N/A	0b101010 (ASCII char *)	The payload contains N-Char packet
EOP	0bX101 (X = Parity bit)	0b100010 (ASCII char ‘)	SpaceWire End of Packet marker
CCODE_HEADER	NULL Time code Interrupt code	0b100011 (ASCII char #)	The payload contains a control code

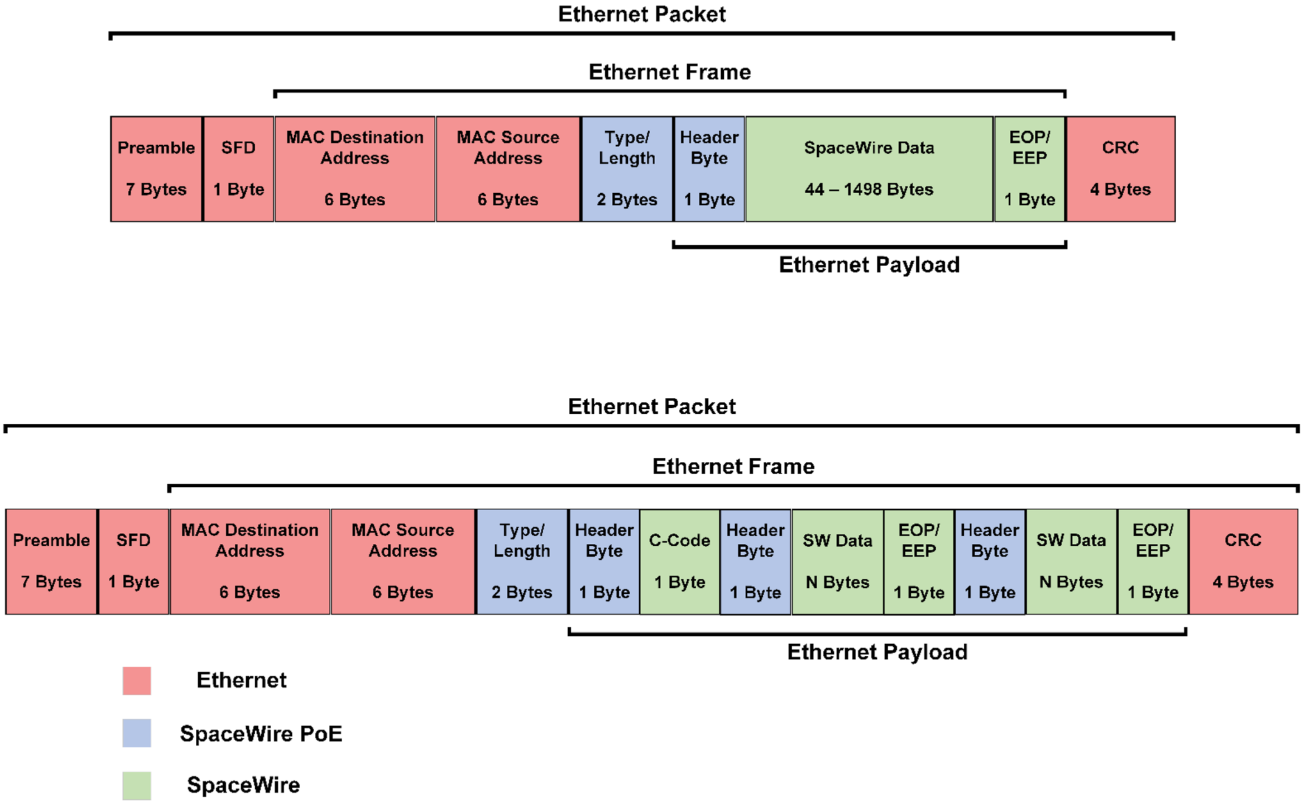

Fig. 2 SpaceWire PoE implemented packet format

3.3 Datalink and physical layer implementation

The datalink of the SpaceWire PoE protocol is responsible for the communications link, encoding of SpaceWire packets from the network layer, decoding of incoming bitstream from the physical layer and directing Ethernet frames to the MAC destination address. The physical layer of the SpaceWire PoE protocol is responsible for the driving and receiving of bitstream data signals over the physical medium. The datalink and physical layers can be implemented by a system deploying an Ethernet MAC interface and PHY driver compliant with IEEE 802.3 2008 [9].

4 SpaceWire PoE concept demonstrator system

4.1 System definition and use case

To verify the proposed implementation of the SpaceWire PoE protocol, the concept demonstrator system was developed in accordance with the following use case, which was defined in collaboration with space industry subject matter expert Thales Alenia Space UK, who are contributing towards project MOSAR.

> *"The use of a SpaceWire PoE common-architecture to switch ON and OFF nodes of the mission system, negotiate the demand upon the request of power if the node can supply then transfer the maximum power. The SpaceWire PoE common architecture will also be used to distribute SpaceWire packets within the mission system between nodes."*

To enable a rapid development lifecycle and due to the pandemic restrictions at the time this work was conducted, the technical scope of the concept demonstrator system was defined as below.

i. Functional testing shall prove the capability only through qualitative criteria of incorporating the SpaceWire network layer into the Ethernet PoE protocol.
ii. Functional testing shall be conducted with available COTS.
iii. SpaceWire PoE network shall only consist of an end-to-end link (no routing switches).

4.2 System requirements

From the proposed SpaceWire PoE full protocol stack defined in Table 1 and the given use case, the requirements of the concept demonstrator system were elicited into a system requirements document (SRD) (Supplementary Annex A). The approach adopted was for the SpaceWire protocol requirements to be extracted for the network, datalink and physical layers; thereafter an analysis was conducted to assess the level of compliance that the proposed implementation of the SpaceWire PoE protocol stack would achieve in accordance with the SpaceWire protocol. Analysing the requirements from this perspective allowed for it to be evident where the Ethernet layers of the SpaceWire PoE protocol did not satisfy SpaceWire and what issues would be present if introduced into a SpaceWire network. The requirements of the SRD were defined as either threshold or objective. The threshold requirements were considered for the minimum level of compliance with the SpaceWire protocol. The objective requirements were defined as optional in the SpaceWire protocol and do not need to be implemented for the operation of SpaceWire PoE.

4.3 System hardware development

The datalink and physical layers of the SpaceWire PoE protocol for the concept demonstrator system were achieved by the integration of suitable hardware. To best replicate the typical topology of a spacecraft mission system that exploits a SpaceWire network and how it is utilised as a method of communication between mission subsystems, the system architecture of the concept demonstrator was designed as shown in Fig. 3. The SRD (Supplementary Annex A) defined the hardware requirements of each subsystem of the concept demonstrator system to achieve the given use case. Subsequently, the appropriate COTS equipment is described below.

i. Master and slave node: Espressif Systems ESP32-Ethernet-Kit-V1.1.

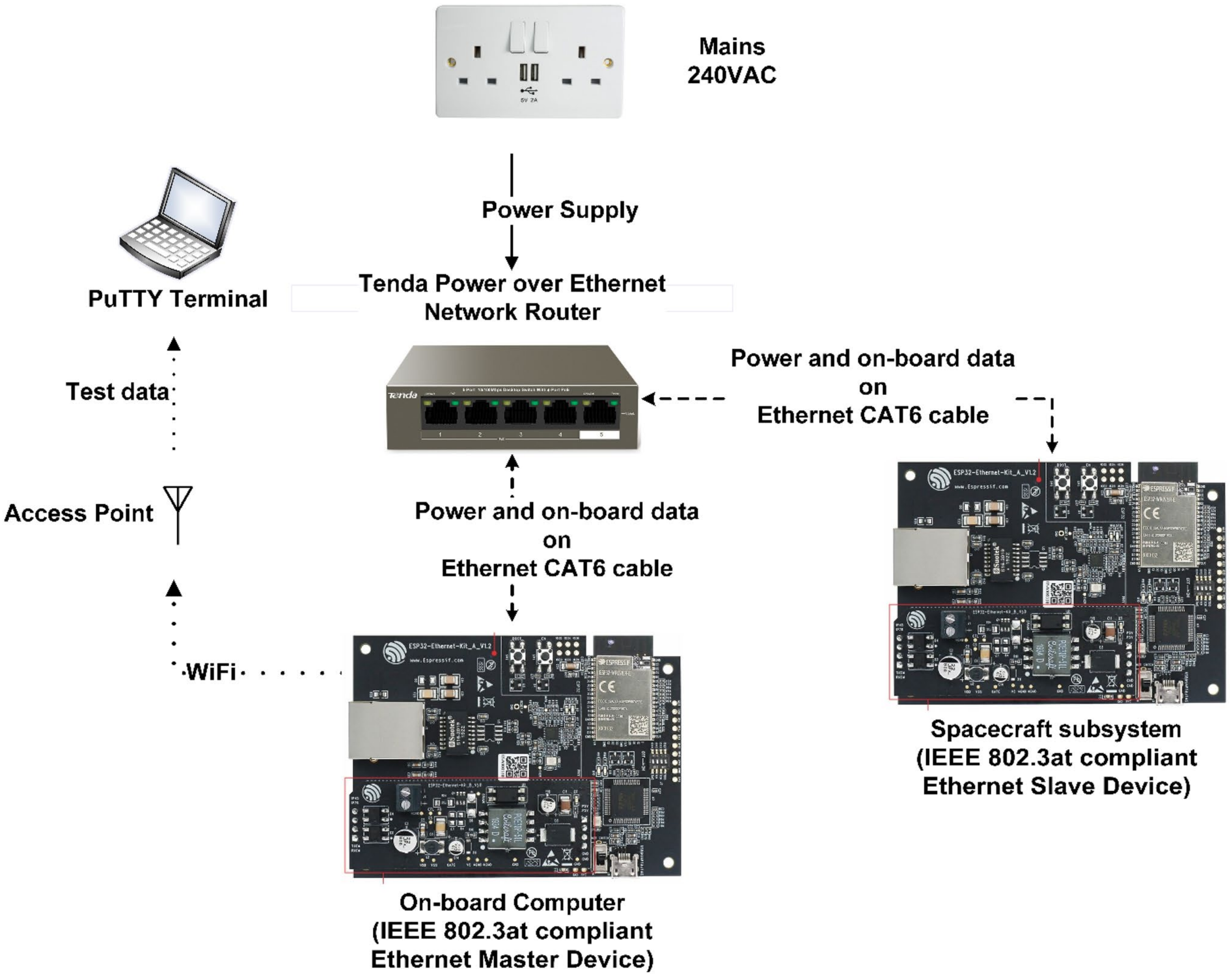

Fig. 3 Concept demonstrator physical system architecture

ii. PoE network router: Tenda desktop switch with 4-port PoE.
iii. Cable: IEEE 802.3 CAT6 Ethernet.

4.4 System software development

The network layer of the SpaceWire PoE protocol was achieved by the implementation of bespoke software in the master and slave Ethernet nodes. The use of unified modelling language (UML) diagrams (Supplementary Annex B) supported the design and verification of the concept demonstrator software. These diagrams enabled the high-level software architecture to achieve the required functionality. The UML use case diagram of the concept demonstrator is shown in Fig. 4, identifying the primary actors in the system and required functionalities.

The scope of the sequence and state UML diagram only considers the transmission and receiving of SpaceWire packets between the master and slave Ethernet nodes. This is due to the fact that the PoE functionality is handled passively by the COTS hardware. The sequence diagram illustrates the software functions to be invoked for sending (red box in Fig. 5) and sending and receiving (yellow box in Fig. 5) of SpaceWire packets between nodes. The state diagram was generated to depict the different states of the master and slave nodes for the given use case and what event-driven actions would occur in software during the receiving and transmission of SpaceWire packets.

4.5 Hardware and software integration

The hardware and software components of the concept demonstrator were integrated using built-in drivers and libraries within the ESP-IDF for the ESP32 development board, which could be flashed with the bespoke source code to enable SpaceWire PoE protocol functionality. The Ethernet network switch is IEEE 802.3at compliant, therefore, no additional firmware/software was required for its integration into the system. The whole realisation of the concept demonstrator system is shown in Fig. 5.

4.6 Concept demonstrator functional testing and analysis

Functional testing was conducted to validate the concept demonstrator system requirements had been met as specified within the SRD (Supplementary Annex A). Test cases were generated considering the SRD. The test cases were only generated for the network layer of the SpaceWire PoE protocol stack as compliance to the datalink and physical layers were achieved by the already IEEE 802.3 PoE standard compliant hardware.

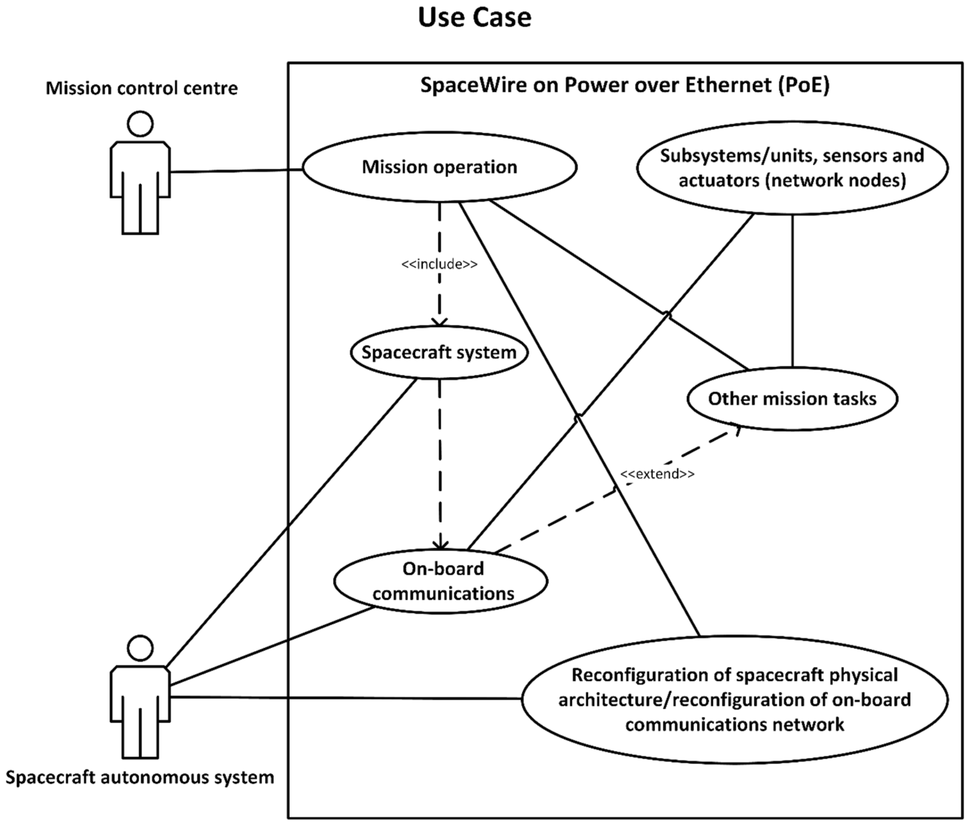
Use Case
Mission control centre
SpaceWire on Power over Ethernet (PoE)
Mission operation
Subsystems/units, sensors and actuators (network nodes)
<<include>>
Spacecraft system
Other mission tasks
<<extend>>
On-board communications
Reconfiguration of spacecraft physical architecture/reconfiguration of on-board communications network
Spacecraft autonomous system

Fig. 4 SpaceWire PoE UML use case diagram

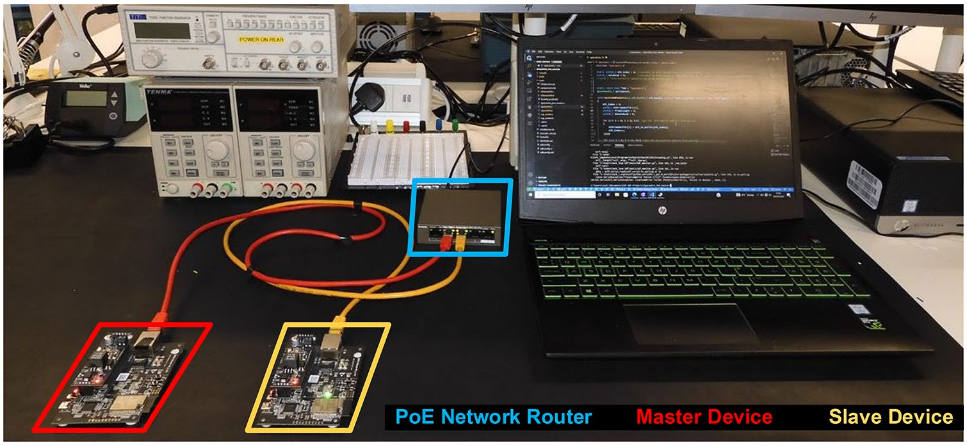

Fig. 5 SpaceWire PoE concept demonstrator system

The system was configured as shown in Fig. 3 for each test case. The arrows represent the connections between the components of the system, solid arrows being physical, and dashed wireless. The purpose of introducing the wireless element was to demonstrate that the system could function in isolation from other external power sources and to assure confidence that the only power being supplied to the master and Ethernet nodes was by the PoE. The WiFi client/server terminal window (PuTTY) allowed for the test data to be extracted during testing.

The system passed all 4 of the test cases, although full compliance with the system requirements was not achieved. The partial compliances were due to the lack of SpaceWire time code registers implemented in the slave and master nodes to hold the time value at a system level. Further development is needed for full compliance. The achieved compliance against the system requirements is documented in the SRD (Supplementary Annex A). The functional testing conducted demonstrates the full functionality of the concept demonstrator; a system with an implemented SpaceWire PoE full protocol stack that provides electrical power and communications.

5 SpaceWire PoE performance analysis

Theoretical assessments have been conducted on the SpaceWire PoE system to quantitively measure the performance between the SpaceWire, Ethernet and SpaceWire PoE protocols.

5.1 Transmission efficiency

Figure 6 shows the efficiency of the SpaceWire PoE system and related protocols in terms of the ratio of payload (SpaceWire) data per frame/packet in the scenario of sending one SpaceWire packet with a payload containing N-Char data in the range of 1–1498 bytes. The efficiency shown for each protocol is in the context of the packets/frames being implemented for an end-to-end link (no logical/path addressing for SpaceWire required).

It can be observed that the SpaceWire protocol is efficient for the transmission of smaller data payloads. The Ethernet protocol requires a minimum of 64 bytes to be transmitted even for smaller payloads and the inflection point on the graph signifies this change, therefore being less efficient in the range of sending between 0 and 46 bytes. The SpaceWire PoE protocol is only minimally less efficient than the Ethernet protocol in the range of 40–400 bytes due to the total header size of the frame being larger by 2 bytes for the header-byte and EOP marker, which eventually converges with the Ethernet protocol over 400 bytes where the difference in efficiency can be deemed negligible. The graph reinforces the assessment made within the SpaceWire community of interest, stating the efficiency losses when utilising Ethernet for transmission of SpaceWire data.

The maximum and minimum efficiencies of transmission are calculated using Eqs. 1 and 2. Equation 1 is used for SpaceWire packet lengths under or equal to 46 bytes, and Eq. 2 for SpaceWire packet lengths over 46 bytes.

$$\eta = 100\left(\frac{L_{SpW} - L_{SH}}{L_{eth}}\right) \tag{1}$$

$$\eta = 100\left(\frac{L_{SpW} - L_{SH}}{L_{EH} + L_{SpW}}\right) \tag{2}$$

η is the efficiency (%), L_{SpW} is the length of SpaceWire packet (bytes), L_{eth} is the minimum length of Ethernet frame (bytes), L_{EH} is the length of Ethernet frame headers (bytes), L_{SH} is the length of SpaceWire PoE headers (bytes).

Table 3 shows the values used to derive the theoretical maximum and minimum efficiencies that the SpaceWire PoE protocol could achieve. Depending on the payload size performance can be $1.5 \leq \eta \leq 98.7$.

For more efficient use of the Ethernet frames implemented by the SpaceWire PoE protocol, multiple SpaceWire packets can be inserted per Ethernet frame. In the instance of multiple SpaceWire packets inserted into the Ethernet frame of varying length and type (N-Char or C-Code), the efficiency cannot be assumed to increase in a linear behaviour similar to that demonstrated in Fig. 6. To simulate the volatility of transmission efficiency for SpaceWire packets sent within an Ethernet frame, we randomised the length of the Ethernet frame payload and subsequently the length of each SpaceWire packet to be inserted within the frame.

The results shown in Fig. 7 are split into two datasets for comparison, that being the size of the SpaceWire packets being inserted into the Ethernet frames are between the ranges of 1–1498 bytes for the top plot and 1–250 bytes for the bottom plot. This was to demonstrate the disparity in efficiency depending on the use case for the SpaceWire PoE protocol in relation to sending larger packets for science data or sending smaller packets for telemetry data.

For both plots, it can be observed that for SpaceWire data under or equal to 44 bytes the variance in efficiency is significant with no correlation with the number of packets sent. The use case of sending larger packets of SpaceWire data (1–1498 bytes) results in an efficiency nominally between 90 and 100%. This is due to the requirement of fewer header bytes and EOP markers within the frame, increasing the ratio of SpaceWire data to the total frame length. Achieving an equivalent efficiency of nominally between 90 and 100% for smaller packet sizes is achieved when ten or more are inserted into the Ethernet frame. This analysis helps assess the suitability of the SpaceWire PoE protocol for the individual use cases performing at a baseline efficiency.

The random nature of the data being generated for the plots cannot be visualised at the specific points of interest for the change of rate of efficiency, particularly the inflection point shown in Fig. 6.

5.2 Bandwidth

The maximum bandwidth that can be achieved in the link of each protocol is derived using Eqs. 3 and 4. The link speed was benchmarked at 100 Mb/s for all protocols. It is to be noted that SpaceWire is capable of link speeds of between 2 and 200 Mb/s and up to 400 + in specific cases.

$$P_n = \frac{L_s}{P_s B} \tag{3}$$

$$BW = P_n P_S B \tag{4}$$

P_n is the No. of packets per second (Ns^{-1}), L_s is the Link speed (Mb/s), P_s is the Size of packet (Bytes), B is the Size of byte (Bits), BW is the Bandwidth (Mb/s).

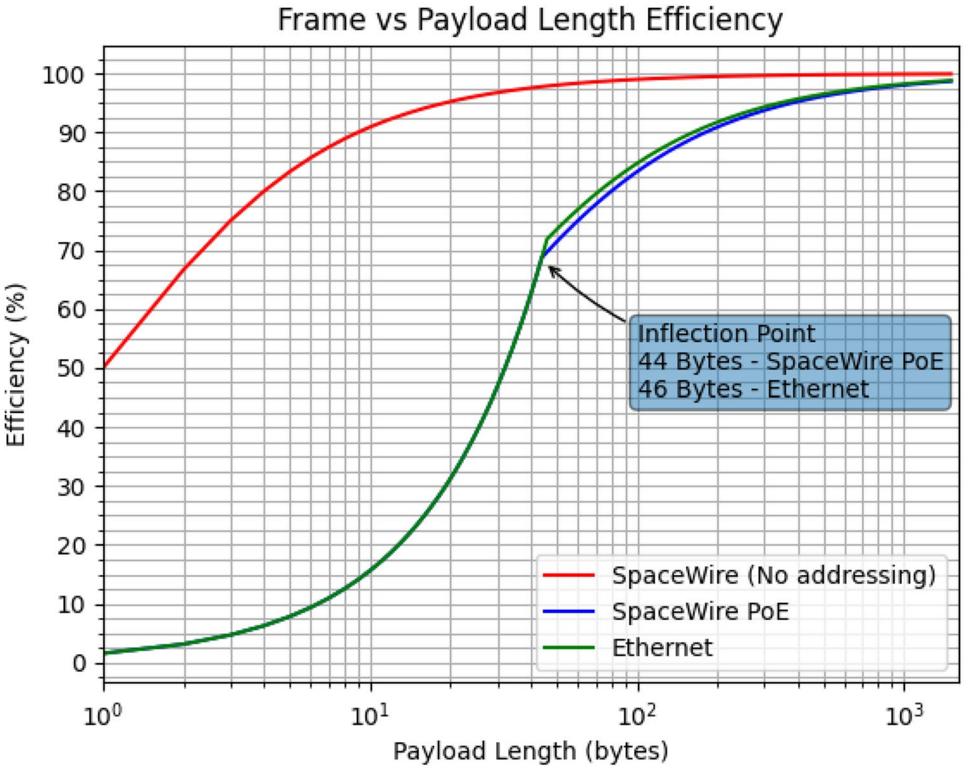

Fig. 6 SpaceWire PoE, SpaceWire, Ethernet frame and payload comparison

Table 3 SpaceWire PoE maximum and minimum transmission efficiencies

Efficiency level	L_{SpW}	L_{eth}	L_{EH}	L_{SH}	$\eta(\%)$
Maximum	1500	N/A	18	2	98.68
Minimum	3	64	N/A	2	1.56

The maximum data rate achieved by each protocol is shown in Tables 4 and 5. In theory, SpaceWire packets (payload size) can be of infinite length. Therefore, to allow appropriate comparison with other protocols the payload size, P_s, for SpaceWire was defined to match the throughput of Ethernet and SpaceWire PoE at 100 Mb/s. In the context of the maximum utilisation of the bandwidth, SpaceWire is faster and subsequently more efficient by a margin of 1.81% in comparison to standard Ethernet. The margin would increase linearly as the speed of the SpaceWire link is increased.

5.3 Power consumption

The total amount of power to transmit and receive SpaceWire data via the SpaceWire PoE protocol can be estimated from the use of the measured energy consumption determined in [22] which states the estimated power or energy consumption of the operation of an Ethernet network and the transmission of Ethernet packets over a network. The energy values used for further analysis of power consumption are shown in Table 6.

The total amount of power to process an Ethernet packet can be approximated using Eq. 5 and P_n being 8127. The value of 1.6 mW can be an underestimate due to this calculation considering the least number of packets sent per second, therefore, for high traffic consisting of small payloads this value could increase. The determined energy value of per-packet processing energy (Table 6) includes the processing of IP headers which does not apply to SpaceWire PoE protocol, although SpaceWire PoE protocol contains more bytes as headers for processing the data, especially in the instance of multiple SpaceWire packets within an Ethernet frame. Therefore, the mentioned value is an acceptable approximation for these theoretical calculations.

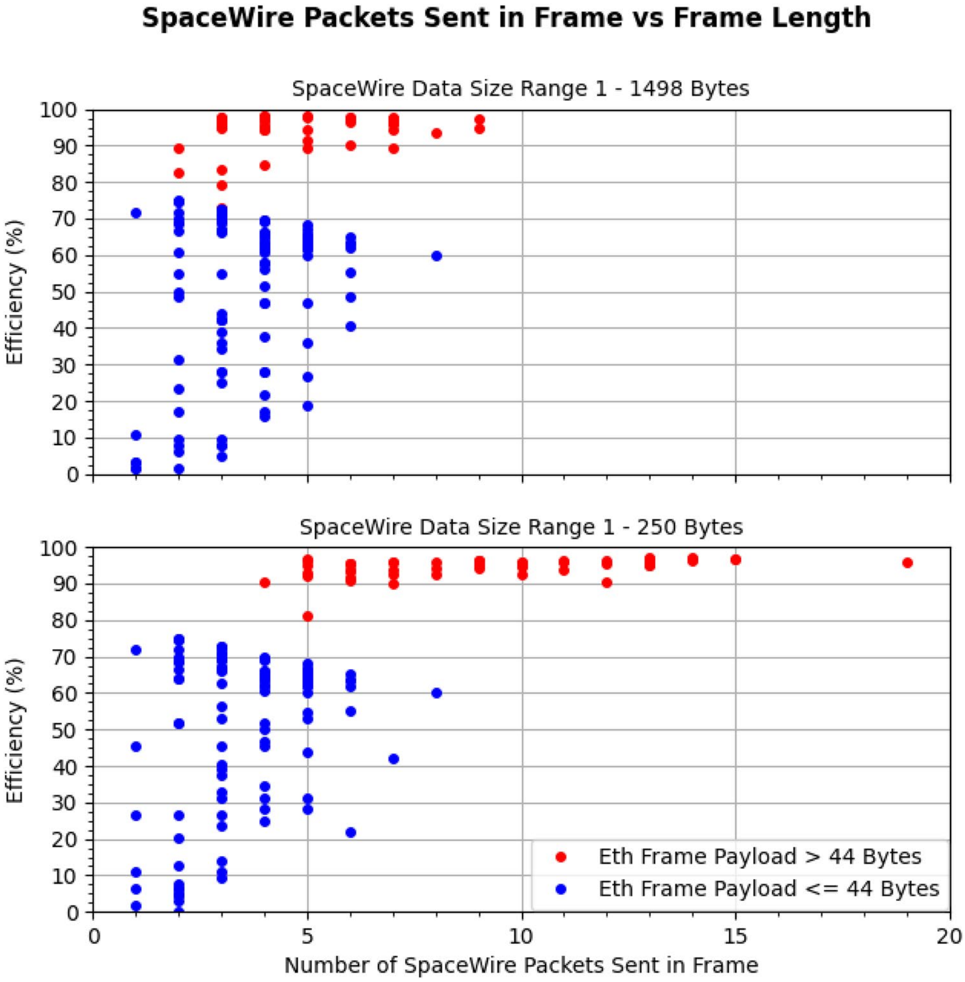

Fig. 7 Ethernet frame transmission efficiency for carrying multiple SpaceWire packets

Table 4 Packets and frames rates of Ethernet, SpaceWire and SpaceWire PoE

Protocol	Link speed, L_s (Mb/s)	Packet size,P_s	Size of bytes, B (bits)	No. of packets per second,P_N
Ethernet	100	1538	8	8127
SpaceWire	100	100,000	8	124
SpaceWire PoE	100	1538	8	8127

Table 5 Bandwidth comparison of Ethernet, SpaceWire and SpaceWire PoE

Protocol	No. of packets per second, P_N	Payload size, P_s (bytes)	Size of bytes, B (bits)	Bandwidth, BW (Mb/s)
Ethernet	8127	1500	8	97.52
SpaceWire	124	100,000	8	99.2
SpaceWire PoE	8127	1498	8	97.39

$$P_p \approx \frac{E_p P_n}{t} \approx \frac{197.2 \times 8127}{1} \approx 1.6mW \tag{5}$$

P_p is the total power to process packets (mW), E_p is the per packet processing energy (nJ), P_n is the total no. of packets per second (Ns^{-1}), t is the total time (s).

The amount of energy required for the entire lifecycle of an Ethernet byte is shown in Eq. 6. The definitions of the constants are given in Table 6.

$$E_b = \left(E_{rx} + E_{rs}\right) + \left(E_{tx} + E_{ts}\right) \tag{6}$$

To derive the estimated maximum total power for the transmission of Ethernet bytes, Eq. 7 was used with P_n being 8127, to convert the given energy value to power consumption for the maximum number of bytes per second for a link speed of 100 Mb/s.

$$P_b \approx \frac{E_b P_s P_n}{t} \approx \frac{3.4 \times 10^{-9} \times 1500 \times 8127}{1} \approx 41.45mW \tag{7}$$

P_b is the total power of bytes (mW), E_b is the energy per byte (nJ), P_s is the packet payload Size, P_n is the packets per second (Ns^{-1}), t is the total time (s).

From the derived power values for the processing and transmission of Ethernet data, the total maximum power consumption for a SpaceWire PoE enabled system can be expressed with Eq. 8.

$$P_T \approx P_C + NP_E + N(P_p + P_b) \tag{8}$$

P_T is the total power consumption (W), P_C is the power consumption of power sourcing equipment (W), P_p is the total power to process packets (W), P_E is the power consumed by Ethernet port (W), P_b is the total power of bytes (W), N is the no. of Ethernet nodes in the system.

Table 6 shows the approximated total maximum power consumption of the implemented concept demonstrator system for varying data rates. The value of P_C used is an estimate from the use of the electrical specifications of the power supply for the Ethernet network switch. The values of P_p and P_b are adjusted by a factor of 10 to consider the change in energy consumption due to the increase or decrease in time to transmit and receive a byte or process a packet for the data rates of 10 Mb/s and 1000 Mb/s. The number of nodes, N, reflects that the PSE shall be providing power to two nodes (master and slave). The power consumption of the embedded system has not been considered in this calculation, although it is assumed to be of negligible impact for these approximations.

The results from Table 7 are graphically presented in Fig. 8. It can be observed that the power consumption to establish and maintain the Ethernet link is responsible for much of the power consumption and the transmission of SpaceWire data is only minimal in comparison (shown on top of bar). The variance of data rates complements the findings in the supporting study [22] that the higher the data rate the higher the total power consumption.

Table 6 Estimated power and energy consumption of Ethernet

Energy component and description	Energy component label	Energy component estimated value (nJ)
Per-packet processing energy	E_p	197.2
Per-byte energy	E_b	3
Per-byte ingress storage energy	E_{rs}	0.8
Per-byte egress storage energy	E_{ts}	1.6
Per-byte transmit/receive energy	E_{tx}/E_{rx}	0.5

6 Implementation issues

The following key issues have been identified that require to be addressed to further the technology readiness level of the SpaceWire PoE protocol.

Table 7 Total estimated power consumption for SpaceWire PoE protocol

Link speed (Mb/s)	Power consumption of PSE, P_C (W)	Power consumption of ethernet port, P_E (W)	Total power to process packets, P_p (W)	Total power for bytes, P_b (W)	No. of nodes, N	Total power consumption, P_T (W)
10	4	1.102	0.00016	0.004145	2	6.21
100	4	1.102	0.0016	0.04145	2	6.29
1000	4	1.102	0.016	0.4145	2	7.065

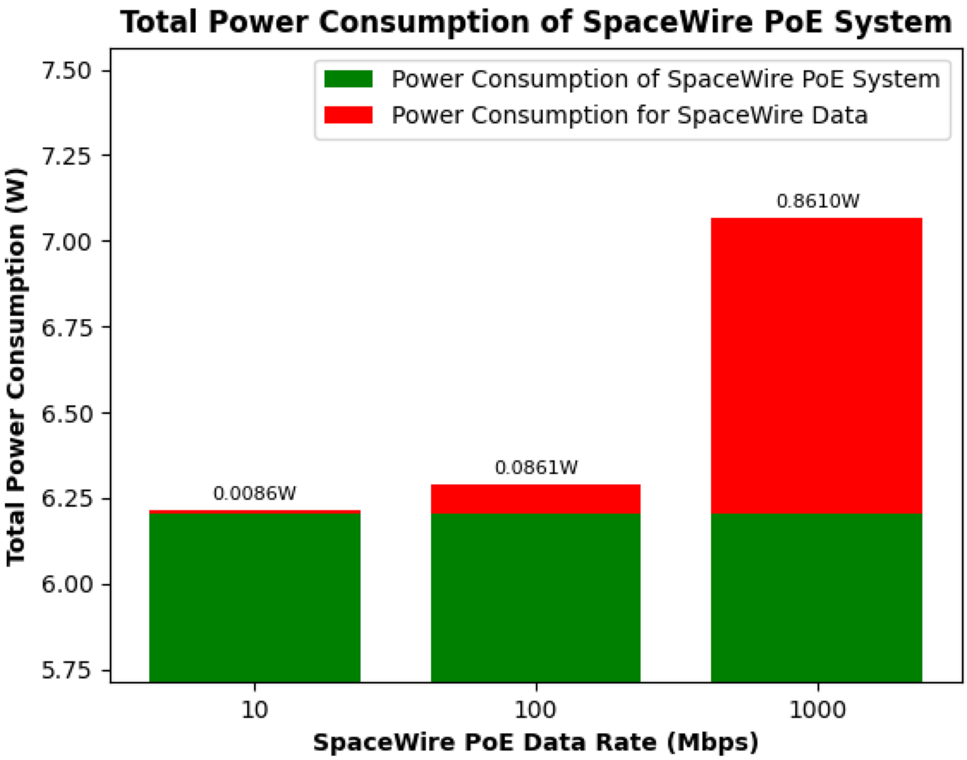

Fig. 8 Estimated power consumption of concept demonstrator

6.1 Efficiency volatility

The volatility in efficiency demonstrated in Fig. 8 is due to the insertion of artificial header bytes to distinguish the type of data being sent (N-Char or C-Code). The SpaceWire PoE system is not optimal for space system architectures that only require the transmission of SpaceWire data that consists of nominally smaller-sized packets or control codes that do not utilise the full length of an Ethernet frame. The SpaceWire PoE protocol is best utilised for sending larger-sized packets, that make full use of the Ethernet frame size or use multiple Ethernet frames to send packets that are larger than 1498 bytes. This functionality can be achieved with further development of the proposed SpaceWire on PoE protocol to include additional header bytes and flags for segmentation and building packets larger than 1498 bytes.

The SpaceWire PoE protocol for smaller packets can still be viable if the overall efficiency of the mission system is improved. For example, the adoption of a SpaceWire PoE mission system architecture loses efficiency in data transmission, although weight savings from reduced cabling eliminate the efficiency offsets.

6.2 IEEE 802.3 PoE functionality

During the development of the concept demonstrator, it was recognised there is an inability to access the PoE functionalities of COTS products to be optimised to the space industry use case. The nature of how PoE is currently applied in terrestrial applications is that it is introduced as a passive element where the power identification and control management are achieved through application specific integrated chips (ASIC) in COTS solutions. Therefore, for the implementation of a SpaceWire PoE system further development would be required to understand and implement how the PoE function can be optimised to perform as desired in accordance with the given use case to turn ON and OFF Ethernet nodes when required.

6.3 Detection and feedback of errors

The handling of error detection is processed in the data link layer of the SpaceWire PoE protocol. If an error is to occur, i.e. the cyclic redundancy check (CRC) is wrong, currently there is no capability to isolate whether the error occurred in the link or data. Therefore, no EEP marker (header-byte) is placed in the SpaceWire packet. For SpaceWire data that is larger than an Ethernet frame in size, this will need to be investigated as to how packets are recovered in the event of errors, although it could be determined that the CRC check is sufficient.

7 Future work

To progress the technology readiness level and validity for the deployment of a SpaceWire PoE system architecture in future spacecraft systems, the following areas of future work have been identified.

7.1 IEEE 802.3 PoE bespoke implementation

Further development is required by the concept demonstrator to evidence a bespoke implementation of the PoE functionality. This is primarily to enable and disable Ethernet nodes when required for the transmission of SpaceWire data as described by the use case.

7.2 Network discovery and configuration

To support the development of modular spacecraft design embodied by project MOSAR, further research and testing are to be conducted to determine how a SpaceWire PoE protocol enabled system performs under a changing network topology. This could be the detection, link, and transmission of data to a changing network consisting of Ethernet nodes and how this is managed by the system. Research in this area would directly support the current literature that is being generated by the community of interest, which contributes towards the synchronisation and harmonisation of modular spacecraft mission subsystems.

7.3 Interoperability to legacy SpaceWire and new SpaceFibre equipment

For terrestrial applications, PoE has shown promising interoperability [23]. For space applications, it is to be considered that the adoption of the new SpaceWire over PoE communications protocol within a spacecraft architecture may be interoperable with legacy systems that have adopted solely the SpaceWire communication and not for power transfer. The SpaceWire state machine is required to handle equipment that was developed for a previous standard of SpaceWire. If the receiver node is disabled, the ErrorReset state is entered. The plug and play MOSAR spacecraft units may encounter any connection problem which will be handled in an appropriate state such as disable/enable, link start and autostart. The link initialisation behaviour is time-dependent on the clock frequency. The advanced network behaviour aspects such as link error handling and recovery, timeouts and resetting will be implemented in the future as receiver nodes are added to the network. A further aspect is an implementation of fault detection, isolation and recovery routine which is critical to the plug and play MOSAR spacecraft. A legacy routing switch may encounter addressing problems as the network grows since the paths will require adaptive routing. Non-deterministic path addressing could lead to nodes being undiscoverable, inaccessible, stuck in an error state, inaccurate routing map or network unreliability. Beyond this, power demand/supply negotiation will take place for one of the eight classes.

A study is needed that ascertains if and how a system that has adopted the SpaceWire protocol can operate with the proposed SpaceWire PoE protocol. This will require analysis of the physical, datalink, and network layers similar to that conducted in the present work, except working in the reverse order. Such work could look to mitigate the potential interoperability issues.

The SpaceFibre standard [24] is backwards compatible with SpaceWire. Any equipment designed for SpaceWire over PoE would be able to handle data to/from SpaceFibre equipment. The SpaceFibre equipment must have its powering source. The other limitation of SpaceFibre is the 5 m electrical wire length which is kept short for high throughput data interfaces such as for imaging or scientific instruments.

7.4 Network layer refinement

The implementation of the network layer can be improved. The current implementation uses artificial headers that are one byte in size to distinguish different types of packets and the required SpaceWire markers such as the EOP and EPP. Further work can seek to refine the currently proposed protocol to reduce the overhead and/or increase the usefulness of the bytes sent within the Ethernet frame. Artificial header bytes for the SpaceWire PoE protocol could include single and/or multi-bit signifiers for type, packet number, length for more error resistant and efficient transmissions.

8 Conclusion and recommendations

Radiation-tolerant SpaceWire hardware is commercially available [25] and mechanical load, thermal and gamma dose tests have been reported for SpaceFibre [26]. However, the current readiness level of the SpaceWire over PoE technology is low for practical use in a mission especially considering the required ruggedness for the space environment. The present work aims to hash out the functionality of the SpaceWire communication on the Ethernet stack COTS equipment. The present findings are not exhaustive by any measure. A fully grown communications network remains to be tested after which device powering and power demand negotiation will be carried out. Once achieved a proper road map would require developing custom hardware and software in FPGA or ASIC and subjecting it to space qualification.

A SpaceWire PoE protocol concept demonstrator was developed and evaluated by functional testing and supported by a theoretical assessment of the performance of the proposed SpaceWire PoE protocol. The functional testing confirmed that a combined SpaceWire and Ethernet protocol stack can be implemented on COTS PoE hardware to achieve communications and power delivery between nodes on a shared physical medium. Theoretical analysis has quantified the degradation of efficiency in communications and estimated power consumption of a SpaceWire PoE enabled system. Implementation issues and future work were identified that will better inform the validity of SpaceWire PoE protocol deployed on spacecraft, for subsystems intercommunications and power distribution achieved through a singular architecture.

From this study, some recommendations can be made to further assess the validity of the SpaceWire over Ethernet in a PoE environment. The implementation of the SpaceWire PoE protocol is dependable on the wider sustainability and savings that are made on the intended spacecraft or mission system. Suitable engineering judgement should be made on the overall gain of implementing a singular system architecture for the transmission of data and distribution of power, where the potential savings of mass in cables, reusable resources and reduction in maintenance outweigh that of the loss in efficiency of not solely using SpaceWire network architecture. Various network topologies also need to be investigated for SpaceWire over PoE protocol as the network extends when the nodes come online or go away or are not detectable for example due to a corrupt routing table or addressing problem.

Once the communication matters are settled and the network performance is acceptable, the next step would be to implement the powering sequence feature to switch spacecraft units ON/OFF by querying the power demand of different units, negotiating, approving or denying the power or switching the units OFF upon malfunction.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s12567-022-00454-z.

Acknowledgements The authors are thankful to the Electronics Engineering group of Thales Alenia Space UK for providing the technical support for this work. Thanks also go to the anonymous reviewers who thoroughly reviewed the article to bring it to the present form.

References

1. Watkins C.B., Walter, R.: Transitioning from federated avionics architectures to integrated modular avionics. In: IEEE/AIAA 26th Digital Avionics Systems Conference, Dallas (2007)
2. MOdular Spacecraft Assembly and Reconfiguration (MOSAR).: https://www.h2020-mosar.eu
3. Khurram, M., Zaidi, S.M.Y.: CAN as a spacecraft communication bus in leo satellite mission. In: Proceedings of 2nd International Conference on Recent Advances in Space Technologies. RAST 2005, pp. 432–437 (2005). https://doi.org/10.1109/RAST.2005.1512606
4. Dellandrea, B., Gouin, B., Parkes, S., Jameux, D.: MOST: modeling of spacewire & spacefibre traffic applications and operations: on-board segment, data systems. In: Aerospace, DASIA, Warsaw, European Space Agency, (Special Publication) ESA SP, SP 725 (2014)
5. Heller, C., Heidinger, E., Schneele, S., Fischer, W., Klose, P.: Power-over-Ethernet for avionic networks. In: 29th Digital Avionics Systems Conference, pp. 5.A.2-1–5.A.2-11 (2010). https://doi.org/10.1109/DASC.2010.5655327
6. Canaday, H.: War on wiring. In: Aerospace America, pp. 24–27 (2017)
7. European Cooperation for Space Standardisation—ESA Requirements and Standards Division, ECSS-E-ST-50-12C Rev.1, SpaceWire—Links, nodes, routers and networks (2019)
8. Parkes, S.: SpaceWire User's Guide. STAR-Dundee Ltd. (2021)
9. IEEE Computer Society.: IEEE Standard for Ethernet 802.3-2018 (2018)
10. Man, L.A.N., Committee, S., Computer, I.: IEEE Standard for Ethernet Amendment 2: Physical Layer and Management Parameters for Power over Ethernet over 4 pairs. 2019.

11. Bartlett, R.O.: NASA standard multimission modular spacecraft for future space exploration. In: American Astronautical Society and Deutsche Gesellschaft fuer Luft- und Raumfahrt, Goddard Memorial Symposium, 16th, Washington, D.C., Mar. 8–10 (1978)
12. Lyke, J.C.: U.S. air force's plug-and-play satellites. IEEE Spectrum (2012)
13. Windsor, J., Gasti, W., Clerigo, I.: Bepicolombo—building a robust data management subsystem utilising spacewire networks: missions and applications, long paper. International SpaceWire Conference (SpaceWire) **2016**, 1–8 (2016). https://doi.org/10.1109/SpaceWire.2016.7771615
14. Ozaki, M., Takahashi, T., Kokubun, M., Takashima, T., Odaka, H.: Spacewire driven architecture for the astro-h satellite. In: 2010 International SpaceWire Conference Proceedings 2010, 22nd–24th June, St. Petersburg
15. Anderson, W., Krimchansky, A., Birmingham, M., Lombardi, M.: The geostationary operational satellite r series Spacewire based data system. International SpaceWire Conference (SpaceWire) **2016**, 1–6 (2016). https://doi.org/10.1109/SpaceWire.2016.7771620
16. Webb, E.: Ethernet for space flight applications. In: Proceedings of IEEE Aerospace Conference, pp. 4–4 (2002). https://doi.org/10.1109/AERO.2002.1036905
17. Clancy, S.C., Shihabi, M.M., Angkasa, K.S.: Using spacewire time codes for spacecraft time synchronization: spacewire missions and applications, short paper. International SpaceWire Conference (SpaceWire) **2016**, 1–5 (2016). https://doi.org/10.1109/SpaceWire.2016.7771596
18. Koblyakova, L., Suvorova, E., Sheynin, Y.: Synchronization of distributed interrupts delivery in aerospace onboard networks: network and protocols, short paper. International SpaceWire Conference (SpaceWire) **2016**, 1–4 (2016). https://doi.org/10.1109/SpaceWire.2016.7771633
19. Romanowski, K., et al.: Spacewire network management using network discovery and configuration protocol: spacewire networks and protocols, short paper. International SpaceWire Conference (SpaceWire) **2016**, 1–7 (2016). https://doi.org/10.1109/SpaceWire.2016.7771602
20. Vinogradov, A., Yablokov, E., Yachnaya, V.: Upgrade of Ethernet-spacewire protocol. In: 24th Conference of Finnish-Russian University Cooperation in Telecommunications (FRUCT) Association (2019)
21. Rozanov, V., Yablokov, E.: Problems of developing spacewire—ethernet bridge and transferring spacewire packages over Ethernet. In: Proceeding of the 16th Conference of Finnish-Russian University Cooperation in Telecommunications (FRUCT) Association (2014)
22. Sivaraman, V., Vishwanath, A., Zhao, Z., Russell, C.: Profiling per-packet and per-byte energy consumption in the NetFPGA Gigabit router. In: IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS (2011)
23. Maniktala, S.: Power over Ethernet interoperability. McGraw-Hill Companies, Inc. (2013)
24. European Cooperation for Space Standardisation—ESA Requirements and Standards Division, ECSS-E-ST-50-11C. SpaceFibre—Very high-speed serial link (2019)
25. Ekergaren, J.: GR718 Radiation-Tolerant 18x SpaceWire Router in DASIA 2015—DAta Systems in Aerospace, vol. 732 (2015)
26. Karppinen, M., et al.: Radiation tolerant optical transceivers for spacefibre data links. In: Proceedings of the 8th International SpaceWire Conference, Los Angeles (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.