University of Dundee School of Computing

Space Technology Centre
University of Dundee

SpaceNet - SpaceWire-RT

Requirements

Revision:	Issue 1.0
Date:	11th February 2008
ESA Contract Number	220774-07-NL/LvH
Ref:	SpW-RT WP3-100.1

Space Technology Centre
School of Computing
University of Dundee
Dundee, DD1 4HN
Scotland, UK
spacetech.computing.dundee.ac.uk

Document Authors

Steve Parkes

Document Change Log

Date	Revision No	Comments
11th February 2008	Issue 1.0	First issue

A comprehensive list of the changes made to this document in each revision is provided in section 4.

CONTENTS

I LIST OF FIGURES

II LIST OF TABLES

1 INTRODUCTION

1.1 BACKGROUND

1.1.1 SpaceWire

SpaceWire has emerged as one of the main data-handling networks for spacecraft since the SpaceWire standard was published in January 2003 [AD1]. It is now being used on many ESA, NASA and JAXA spacecraft and by other space agencies, research organisations and space industry across the world. SpaceWire is designed to connect together high data-rate sensors, processing units, memory sub-systems and the down link telemetry sub-system. It provides high-speed (2 Mbits/s to 200 Mbits/s), bi-directional, full-duplex, data links which connect together the SpaceWire enabled equipment. Networks can be built to suit particular applications using point-to-point data links and routing switches. Application information is sent along a SpaceWire link in discrete packets. Control and time information can also be sent along SpaceWire links. One of the main advantages of SpaceWire is its low complexity (low gate count) and the fact that it can be implemented easily in both ASICs and FPGAs. SpaceWire is supported by several radiation tolerant ASICs designed by or for ESA, NASA and JAXA, and extensive test and development equipment is available.

A SpaceWire network comprises links, nodes and routers. A SpaceWire node is some piece of electronic equipment that needs to use the services of the SpaceWire network. One or more SpaceWire interfaces connect the node to SpaceWire links. The other ends of these links are either connected directly to other nodes or are connected to a SpaceWire router which provides an indirect connection to other nodes. A SpaceWire router comprises a set of SpaceWire interfaces and a non-blocking cross-bar switch. Packets of information arriving at a router are forwarded towards their destination by switching them to pass out of the most appropriate link attached to the router. Each SpaceWire packet comprises a packet destination address, the information being carried in the packet (cargo) and an end of packet marker. The destination address is either a unique identity code for the destination node or a description of the path through the SpaceWire network to the destination node. The SpaceWire router uses the destination address to determine which link it should use to forward a packet.

1.1.2 SpaceWire Remote Memory Access Protocol

The remote memory access protocol (RMAP) was designed to provide a means for one SpaceWire node to write to and read from memory inside another SpaceWire node [AD2]. The aim of the RMAP protocol is to standardize the way in which SpaceWire units are configured and to provide a low-level mechanism for the transfer of data between two SpaceWire nodes. For example RMAP may be used to configure a camera or a mass memory device. The camera device can then write image data to allocated areas of memory in the mass memory using RMAP, or the mass memory may read image data from the camera. RMAP has been adopted by ESA, JAXA and NASA for several space missions and is being implemented in several radiation tolerant SpaceWire devices.

1.1.3 SpaceWire and RMAP Quality of Service

RMAP and SpaceWire operate with a best effort quality of service. While error detection, reporting and recovery techniques are defined in both standards, there is no defined means of recovering any data that was lost or that arrived at its destination in error. Also there is no concept of timeliness in either of these standards. For many SpaceWire applications this is not a problem, but for other quality of service is a key issue.

1.1.4 CCSDS SOIS

The CCSDS Spacecraft Onboard Interface Services (SOIS) working group has defined a set of high level services that a spacecraft onboard network or bus should support.

The CCSDS SOIS Sub-Network red books cover five services:

- **Memory access service** which provides a means for a user entity to retrieve or change data located in memory hosted by a node on a data-link/sub-network.
- **Packet service** which transfers information from one end-point on a data-link/sub-network to another end-point on the same data-link/sub-network.
- **Test service** which provides a means for a user entity to test data system functionality and connectivity of the sub-network.
- **Time distribution service** which provides a means for a user entity to maintain knowledge of time which is common to all data systems on the sub-network.
- **Device discovery service** which provides a means for a user entity to receive notification of data systems' presence on the sub-network.

1.1.5 Quality of Service

Extensive work [RD3, RD4, RD5] was done by the former CCSDSD SOIS Time-Critical Onboard Network Services (TCONS) working group on quality of service (QoS) and the concepts developed by this group have been adopted by the SOIS sub-network group and used for the memory access, packet and time distribution services. There are four levels of quality of service defined by SOIS:

Best-Effort Service Class

- Makes a single attempt to deliver data to its destination but cannot ensure that it will be delivered successfully.

- Data is provided in-sequence (within a priority value), without errors and without duplication.
- The order of data packets is not necessarily preserved.
- Priority indicates the relative urgency with which PDUs should be handled by the sub-network. Priority is applied across the best-effort and assured service classes where both classes are provided.

Assured Service Class

- Ensures delivery of data to its destination.
- Should it not be possible to provide the assured service this is indicated to the sending entity
- Data is provided in sequence (within a priority value), complete, without errors and without duplication.
- Priority indicates the relative urgency with which PDUs should be handled by the sub-network. Priority is applied across the best-effort and assured service classes where both classes are provided.

Reserved Service Class

- Makes a single attempt to deliver data to its destination but cannot ensure that it will be delivered successfully.
- Data is provided in sequence (within the channel and within a priority value), without errors and without duplication.
- A Channel defines the resources that are used to transmit the SDU.
- Priority indicates the relative urgency with which PDUs should be handled by the sub-network. This priority defines the priority of a communication within the resource reservation i.e. within a channel. Priority is applied across the reserved and guaranteed service classes where both classes are provided within a channel.

Guaranteed Service Class

- Ensures delivery of data to its destination.
- Should it not be possible to provide the guaranteed service this is indicated to the sending entity i.e. the user is informed if it is not possible to deliver the data.
- Data is provided in sequence (within a channel and within a priority value), complete, without errors and without duplication.
- A Channel defines the resources that are used to transmit the SDU.

- Priority indicates the relative urgency with which PDUs should be handled by the sub-network. This priority defines the priority of a communication within the resource reservation i.e. within a Channel. Priority is applied across the reserved and guaranteed service classes where both classes are provided within a channel.

Common features

- SDUs extracted from PDUs containing errors will not be delivered.
- Individual Service Data Units being sent cannot be larger than the MTU.

There are two points where these quality of service classes differ from those defined by TCONS: data being delivered in-sequence in the best effort and resource-reserved services.

1.1.6 SpaceWire Real-Time

SpaceWire Real-Time (SpaceWire-RT) is intended to provide a consistent quality of service mechanism for SpaceWire and to support the CCSDS sub-network services. Furthermore it is intended to support control applications running over SpaceWire where timely delivery is essential. To achieve this SpaceWire-RT implements SOIS quality of service over SpaceWire.

1.2 Aims and Objectives

The aim of this document is to provide the requirements for SpaceWire-RT. This will be accomplished by:

- Reviewing the SOIS sub-network service documents,
- Developing a comprehensive set of requirements for SpaceWire-RT, and
- providing a set of use cases that cover the principal uses of SpaceWire-RT.

1.3 Guide to Document

Section 2 provides a review of the CCSDS SOIS Sub-Network Services

Section 2.7 contains the functional requirements for SpaceWire-RT.

Section 4 lists the non-functional requirements for SpaceWire-RT.

Section 5 provides the use cases for SpaceWire-RT.

1.4 Acronyms and Abbreviations

AD	Applicable Document

CCSDS	Consultative Committee for Space Data Systems
ECSS	European Cooperation for Space Standardization
ESA	European Space Agency
ESTEC	ESA Space Technology and Research Centre
ID	Identity
I/F	Interface
PC	Personal Computer
PDU	Protocol Data Unit
QoS	Quality of Service
RD	Reference Document
RMW	Read/Modify/Write
SDU	Service Data Unit
SOIS	Spacecraft Onboard Interface Services
SpW	SpaceWire
TCONS	Time-Critical Onboard Network Services
UoD	University of Dundee

1.5 Terms and Definitions

1.5.1 Numbers

In this document hexadecimal numbers are written with the prefix 0x, for example 0x34 and 0xdf15. Binary numbers are written with the prefix 0b, for example 0b01001100 and 0b01.

1.5.2 Requirement Terms

May is indicates a possibility or hint. It is neither mandatory nor desirable and is entirely up to the implementer as to whether this requirement is implemented or not.

Shall indicates a mandatory requirement.

Should indicates a desirable requirement.

1.6 Reference Documents

The documents referenced in this document are listed in Table 1-1.

Table 1-1: Reference Documents

REF	Document Number	Document Title
RD1	UoD-SpaceNet v7, 23rd April 2007	Proposal for SpaceWire Network and Future Onboard Data-Handling, Technical, Management and Administrative Proposal
RD2	TEC-ED/WG/2005.15	SpaceWire Network “SpW-Net” SpaceWire and Future Onboard Data Handling SpaceNet Statement of Work Annex1
RD3	CCSDS 000.0-W-1.1	Time Critical Onboard Network (TCONS) and Onboard Bus LAN (OBL) Architecture April 2006
RD4	CCSDS 000.0-W-1.1	Time Critical Onboard Network (TCONS) Quality of Service September 2005
RD5	CCSDS 000.0-R-1.1	Spacecraft Onboard Interface Services Intra-Network Service April 2006

1.7 Applicable Documents

The documents applicable to this document are listed in Table 1-2.

Table 1-2: Applicable Documents

REF	Document Number	Document Title
AD1	ECSS-E50-12A, January 2003	SpaceWire: Links, nodes, routers and networks
AD2	ECSS-E50-11A Draft 0.5	SpaceWire Protocols Feb 2008
AD3	CCDSD ccc.c-R-1.0 Draft Red Book	Spacecraft Onboard Interface Services Memory Access Service Jan 2007
AD4	CCDSD ccc.c-R-1.0 Draft Red Book	Spacecraft Onboard Interface Services Subnetwork Packet Service Jan 2007
AD5	CCDSD ccc.c-R-1.0 Draft Red Book	Spacecraft Onboard Interface Services Test Service Jan 2007
AD6	CCDSD ccc.c-R-1.0 Draft Red Book	Spacecraft Onboard Interface Services Time Distribution Service Jan 2007
AD7	CCDSD ccc.c-R-1.0 Draft Red Book	Spacecraft Onboard Interface Services Device Discovery Service Jan 2007

2 REVIEW OF CCSDS SOIS SUB-NETWORK SERVICES

In this section the CCSDS SOIS sub-network service documents are reviewed.

2.1 QUALITY OF SERVICE

The quality of service for the SOIS sub-network services has been adopted from the work of the CCSDS TCONS working group [RD3, RD4, RD5] with two exceptions: the best effort and assured services for SOIS require in-sequence delivery of packets while TCONS did not have this requirement.

Ensuring in-sequence delivery for a best effort service means that additional data will be lost is something is not in sequence. Supposing packets 1, 2, 3, 4 and 5 are sent and they arrive in the order 1, 3, 2, 4, 5. Packet 3 is detected as out of sequence and will be discarded. Packet 2 is then retained if the sequence is restarted after an error (which is rather arbitrary). Packet 4 will then be discarded as is it out of sequence. Packets 5, 6 etc will be retained. Because one packet was out of sequence (packet 3) two packets (3 and 4) have been discarded. There is no basis for assuming that discarding packets that are out of sequence will be more detrimental to system operation than not having them delivered at all. The assured service provides a mechanism for delivering packets "without error" and in-sequence and this service should be used when these quality of service characteristics are required.

2.2 MEMORY ACCESS SERVICE

2.2.1 Function

The function of the SOIS Memory Access Service is to provide a means for a user entity to retrieve or change data located in the memory hosted by a node on a data-link/sub-network. This service is provided across a single sub-network. The Memory Access Service provides all four quality of service classes listed in section 1.1.5.

2.2.2 Parameters

The parameters used by the service are summarised below with comments in italics:

Source sub-network service access point (SSNSAP) which identifies a data system attached to a sub-network and a user entity within that data system that wants to use the SOIS service.

Destination address is a logical address (global to a spacecraft or local to the sub-network) which defines the data system where the memory to be accessed is located.

There is a substantial difference between a global spacecraft address and a local sub-network address. Is this choice left up to the implementer, or is there some global (SOIS) addressing scheme *implied. This is unclear. Furthermore, the other SOIS documents do not have this distinction. This is probably left over from the TCONS work where SOIS networking covered data transfer across multiple sub-networks.*

Memory ID which identifies a logical memory space containing the memory locations. It may be used to different banks or types of memory.

Start Memory Address is the start address of the memory to be accessed.

Size is the amount of data to be changed or retrieved.

Mask indicated which data are to be changed in a read-modify-write operation.

Data is the values to be read from or written to memory.

Service Class is one of the four SOIS service classes.

Channel specifies an end to end resource reservation for a network communication.

This should of course be a sub-network communication not a network communication as SOIS services are across a single sub-network only.

Priority indicates the importance of the data to the systems.

This should state "Priority indicates the relative urgency with which PDUs should be handled by the sub-network." Something can be of low priority but of immense importance to the system.

Failure Metadata is information generated by the sub-network memory access service provided to the sending entity to provide information related to a failure of service provision.

The term sending entity is not defined – it should be replaced by source sub-network service access point (or preferably the latter should be replaced by the former).

The main problem with these parameters is that they are abstract and their size and format are not covered by the SOIS service definition. This means that an implementation can implement them in whatever form it likes. While this gives complete freedom to the SpaceWire working group to define appropriate parameters for SpaceWire-RT, it does mean that one of the former aims of CCSDS SOIS is lost: that of software portability from one underlying sub-network to another.

Are these parameters intended to be a complete set or is the implementer allowed to add further parameters, or to implement a reduced set?

2.2.3 Primitives

The Memory Access Service has five primitives:

- Read.Request which requests to retrieve the contents of memory,
- Read.Indication which returns the retrieved contents of memory,
- Write.Request which requests to change the contents of memory,
- Read/Modify/Write.Request which invokes an atomic Read/Modify/Write cycle at the memory,
- Memory Access Failure.Indication which informs a user of the failure of a memory access operation.

The fifth bullet point in section 3.2 of the SOIS memory access service document should read "Memory Access Failure.Indication which informs a user of the failure of a memory access operation.

The primitives provided and their parameters are very close to the RMAP primitives and parameters. RMAP provides full the full protocol description whereas SOIS provides an abstract service interface.

At first thought it may seem appropriate to use RMAP for implementing SOIS memory access service over SpaceWire but there are several problems:

- RMAP does not provide any of the qualities of service required by SOIS memory access service. RMAP is close to being able to implement the Best Effort service class but there is no priority defined in RMAP and in-sequence delivery of data is not ensured. It is recommended that priority is added but in-sequence delivery should not be required for the best effort service.
- RMAP uses an initiator SpW address, initiator logical address and transaction identifier to identify the user entity in the initiator. It is not clear if all three of these are represented in an abstract manner by the Source Sub-Network Service Access Point.
- The atomic read-modify-write is not covered fully in the RMAP standard and could not be implemented by RMAP alone.
- The set of primitives provided by SOIS memory access service are incomplete when compared to the RMAP set of primitives. For example there is no primitive to indicate to the data system at the destination (RMAP target) that data has been written to memory.
- The read.indication primitive provides the complete set of parameters with the implication that it is up to the user entity to sort out where the data came from. RMAP associates a request to a reply using the transaction identifier, simplifying operation and relieving the unnecessary burden on the user entity. It is recommended that a more abstract (transaction identifier) approach is used to associate reply to request as is done in RMAP.

- There is no indication (at either end) that a write has been successful or that it has taken place.
- The reason for the read-modify-write operation is not clear. Also the extent of the “atomic” nature of this command is not clear and if there are any restrictions on data length etc.
- The statement “Receipt of the READ/MODIFY/WRITE.request primitive shall cause the SOIS sub-network service provider to retrieve the memory data and to block modification by other user entities.” implies that once a read-modify-write operation has occurred the memory cannot be modified by any other user entities. This is a “lock” not an atomic operation. It is recommended that the read-modify-write primitives are re-thought: what are they to be used for and why, is there no other way of doing the same job?
- The memory access failure indication is used only for the assured and guaranteed services. Suppose that a best effort request to read from memory was made and a user entity (e.g. software routine) was waiting for this data, if no failure indication is given then the user entity would be stuck waiting for the data. It would then be up to the user entity to implement a time-out timer every time it made such a request.
- There is an implication when compared to the parameters used in the sub-network packet service that the memory resides within the sub-network. This implication arises because the memory access service uses the term destination address whereas the packet service uses the term destination sub-network service access point. The location of the memory within the SOIS architecture needs to be made clear and terms used consistently across the various service documents.

2.3 Packet Service

2.3.1 Function

The SOIS sub-network packet service transfers Service Data Units, which are comprised of variable length, delimited octet strings, from one end-point on a data-link/sub-network to another end-point on the same data-link/sub-network.

A Service Data Unit (SDU) is defined as: an amount of information whose identity is preserved when transferred between peer entities in a given layer and which is not interpreted by the supporting entities in that layer.

So what this means is that the packet service transfers information in packets (variable length, delimited octet strings) from one user entity to another user entity on the sub-network.

Is an end-point a service access point? If so a consistent term should be used, if not the term “end-point” should be defined.

2.3.2 Parameters

The parameters used by the service are summarised below with comments in italics:

Data is the service data unit (SDU) of the SOIS sub-network packet service.

Is the SDU described anywhere? If so a reference to its definition should be included in the SOIS packet service document.

Source sub-network service access point (SSNSAP) which identifies a data system attached to a sub-network and a user entity within that data system that want to send a packet.

Destination sub-network service access point (DSNSAP) which identifies a data system attached to a sub-network and a user entity within that data system that a packet is to be delivered to.

This is a different term to the Destination Address used in the SOIS memory service document. Consistent terminology should be used or the reason for different terminology made clear.

Service Class is one of the four SOIS service classes.

Channel specifies an end to end resource reservation for a network communication.

This should of course be a sub-network communication not a network communication as SOIS services are across a single sub-network only.

Priority indicates the importance of the data to the systems.

This should state "Priority indicates the relative urgency with which PDUs should be handled by the sub-network."

Failure Metadata is information generated by the sub-network memory access service provided to the sending entity to provide information related to a failure of service provision.

2.3.3 Primitives

The Packet Service has three primitives:

- Send.Request which requests to send an SDU,
- Receive.Indication which indicates that a packet has been received and which passes the corresponding SDU to the End System.
- Failure.Indication which indicates a failure to provide an assured or guaranteed service.

There are several functions required to provide these primitives over SpaceWire:

- Address mapping from the SSNSAP and DSNSAP to SpaceWire addresses. As the SSNSAP and DSNSAP formats are not defined this seems to be entirely open to the implementation.

- Some form of priority mechanism although here is no definition of the number of priority levels that are to be supported. Again this seems to be up to the implementation.
- Support for the various quality of service classes. The number of channels provided is not defined.

2.4 TEST SERVICE

2.4.1 Function

The SOIS sub-network Test service provides a means for a user entity to test data system functionality and connectivity of the sub-network.

The test service operates with best effort quality of service only.

This is very open-ended: at various points in the SOIS test service document it is the data system, sub-network connectivity, sub-network interface, or sub-network functionality that is being tested. Exactly what is being tested needs to be clarified and the document made consistent.

2.4.2 Parameters

The parameters used by the service are summarised below with comments in italics:

Source sub-network service access point (SSNSAP) which identifies a data system attached to a sub-network and a user entity within that data system that wants to invoke the test service.

Destination address defines a data system whose status is to be verified. A Destination Address identical to the Source Address indicates that the sub-network interface and sub-network functionality local to the invoking application entity is to be verified.

Source address is not defined.

Why is destination address used instead of DSNSAP?

Test Status indicates the result of the test.

2.4.3 Primitives

The Test Service has two primitives:

- Test.Request which requests that a verification be performed,
- Test.Indication which returns the results of the verification.

It appears that this service is used to invoke some test on a data system attached to the sub-network. The test that is run is entirely up to the implementation of the data system and the format of the response is also implementation dependent. There is no attempt made to provide some standard mechanism or interface.

The sub-network connectivity test is incidental; being done by asking each user entity on the sub-network to run a test will check the connectivity between the initiator of the test and the user entities being tested. There is no testing of the assured, resource-reserved or guaranteed services connectivity at all.

2.5 TIME DISTRIBUTION SERVICE

2.5.1 Function

The SOIS sub-network Time Distribution service provides a means for a user entity to maintain knowledge of time which is common to all data systems on the sub-network

2.5.2 Parameters

Source sub-network service access point (SSNSAP) which identifies a data system attached to a sub-network and a consumer user entity within that data system that want to receive time data.

Destination sub-network service access point (DSNSAP) which identifies a data system attached to a sub-network and a producer user entity within that data system that produces time data.

Time is an estimate of the time at the instance of the time.indication primitive.

2.5.3 Primitives

There are four primitives used by the Time service:

- Time Distribution.Request, by which the time consumer requests time data
- Time Distribution.Indication which informs the time producer of the time distribution request
- Time.Request by which the time producer requests time to be sent to the consumer
- Time.Indication which delivers time data to the consumer.

The document states that the time distribution service “operates with a best effort quality of service”, but it then goes on to state “the time data should be delivered with best effort and with bounded latency” which as the document explains requires “data link specific mechanisms such as resource reservation, system analysis or the use of dedicated timing bus”.

There is no requirement on the performance of the time distribution service: specifically “The quality of the bounded latency will depend on the mechanisms available.”

The name of this service is a misnomer: time is not distributed it is obtained. Time distribution implies that everyone on the sub-network is told what time it is. Obtaining the time implies a request for the time and the requestor being told what the time is.

If this service is provided in on a best effort basis then it can be done with RMAP or something similar by just reading a time register from a master-time device on a SpaceWire network.

2.6 Device Discovery Service

2.6.1 Function

The SOIS sub-network Device Discovery service provides a means for a user entity to receive notification of data systems' presence on the sub-network.

Only the best effort quality of service is used for the device discovery service.

The use of best effort quality of service for device discovery implies that if a device present on the is not discovered on the first attempt it will not be reported as being present i.e. a transient error in a packet could cause a device to not be reported as present. This could be very dangerous in a flight system.

2.6.2 Parameters

The parameters used by the service are summarised below with comments in italics:

Source sub-network service access point (SSNSAP) which identifies a data system attached to a sub-network and a user entity within that data system that is to be the recipient of the device discovery information.

Destination address identifies a data system connected to the sub-network.

Once again the distinction between Destination Address and DSNSAP needs to be made.

2.6.3 Primitives

There are two primitives used by this service:

- Device Discovery.Request which requests that device identities be retrieved from the subnetwork,
- Device Discovery.Indication which returns device identities.

When a device discovery request is made the device discovery service has to report back a list of the devices attached to the network.

At one level this is trivial: a constant list of nodes connected to the network could be returned to the data system that makes the device discovery request. This would be adequate for many spacecraft where the architecture is known a priori or where configuration is controlled from the ground.

At another level where devices are being power-up autonomously or connected asynchronously it is a little more complex. The service implementation then has to find out what devices are currently on the network either from a known map of possible devices, or by exploring the entire network.

2.7 General Comments

Does a device have to implement all the SOIS functions?

Are there any constraints at all on the implementation?

Is in-sequence delivery essential for te best effort and resource-reserved services, if so why?

The terminology across the SOIS documents needs to be consistent.

The four quality of service classes provide a complete set covering most, if not all, spacecraft applications.

The five SOIS services are straightforward. With the modification of the in-sequence requirement for the best effort and resource reserved services, all five services with best effort QoS could be implemented with RMAP.

3 SPACEWIRE-RT FUNCTIONAL REQUIREMENTS

In this section the functional requirements for SpaceWire-RT are covered. This takes into account the review of the SOIS sub-network services and the needs of control applications.

3.1 SENDING INFORMATION

R1-1 Packets

SpW-RT shall send information in SpaceWire packets.

Rationale: SpW-RT is intended to operate over a SpaceWire network.

Risk: Low

R1-2 Packet Length

The SpaceWire packets used to transfer SpW-RT information shall be of variable length up to a defined maximum.

Rationale: A maximum length of packet is necessary to ensure efficient packet multiplexing across the network so that high priority packets are not blocked for a long time while a large low priority packet is being sent.

Risk: Low

R1-3 Packet Address

The SpaceWire packets used to transfer SpW-RT information shall have a header containing the SpaceWire address of the destination which can be composed of SpaceWire path, logical or regional logical address characters.

Rationale: A variety of SpaceWire addressing methods is required to support the broadest range of applications. It is important that there is a common method of translating from a SOIS address to a SpaceWire address.

Risk: Low

R1-4 Primary Route

The SpaceWire address of a destination describes the route through a network to the destination. If there are two or more routes available from a source to a destination one route shall be selected as the primary route which is normally used for sending packets.

Rationale: This is necessary to support autonomous redundancy switching which is important for autonomous systems where Earth based operator intervention is not practical, for example a spacecraft in the process of landing on Mars.

Risk: Low

R1-5 Alternative Routes

Alternative routes from a source to a destination may be defined and used in the event of a fault occurring on the primary route.

Rationale: Alternative routes across a SpaceWire network are essential to provide redundancy.

Risk: Low

R1-6 Channel Identifier

A channel shall identify a collection of network resources associated with a route from a source to a destination which is used to control the flow of information across the network and help to ensure timely delivery.

Rationale: The channel collects together a set of network resources that are necessary to provide communications between a source and destination. The channel provides a convenient way of managing those resources.

Risk: Moderate

R1-7 Channel Time-Division

A channel identifier may identify one or more time slots in a time-division multiplexed system where communication from a specific source to a specific destination is only allowed during a particular time slot.

Rationale: Time-division multiplexing is one means of sharing bandwidth. It is not efficient in the overall use of resources but does provide a simple way of implementing deterministic delivery.

Risk: Moderate

R1-8 Channel Bandwidth Reserved

A channel identifier may identify a bandwidth limit in a bandwidth reserved system where communication from a specific source to a specific destination is controlled by limiting the available bandwidth.

Rationale: Bandwidth reservation is an alternative method of sharing bandwidth to time-division multiplexing. It is more efficient in its use of overall system bandwidth and helps to provide timely delivery, but it is not as deterministic as a time-division multiplexed system.

Risk: Moderate

3.2 Quality of Service

R2-1 Quality of Service

SpW-RT shall four classes of quality of service: best effort, assured, resource-reserved and guaranteed.

Rationale: These four classes provide all combinations of once-only/retry and not-reserved/reserved.

Risk: Moderate

3.2.1 Best Effort Service

R2.1-1 Best Effort Service

SpW-RT shall provide a best effort service.

Rationale: This is the simplest service and is already widely used in SpaceWire systems.

Risk: Low

R2.1-2 Best Effort Single Attempt

The best effort service shall make a single attempt to deliver data to its destination but cannot ensure that it will be delivered successfully.

Rationale: A single attempt at delivering a packet is the simplest form of communication. It does not have the complexity of sending acknowledgements and retries in the event of a failure. In a single attempt it is not possible to ensure that data is delivered successfully.

Risk: Low

R2.1-3 Best Effort Not In-Sequence

The best effort service shall not necessarily deliver data in the order in which it was sent (witin a priority level).

Rationale: Since the packets used to send the data will be going over the same route through the network they will normally arrive in the same order in which they are sent. The order in which packets are sent can be different to the order in which they were registered for sending if they are of different priority: the higher priority packets will be sent before the lower priority ones. Furthermore, if group adaptive routing is used and one route gets blocked, it is possible that the packets will arrive in a different order to the order in which they were sent.

Risk: Low

R2.1-4 Best Effort Without Error

The best effort service shall ensure that data delivered has a low probability of containing errors.

Rationale: Erroneous data is not to be delivered. Using an error detection method like a CRC does not ensure that data does not contain errors, only that there is a low probability of there being an error.

Risk: Low

R2.1-5 Best Effort Discard Errors

The best effort service shall discard any data found to contain errors.

Rationale: Erroneous data is not to be delivered as it could cause unwanted or unexpected system behaviour.

Risk: Low

R2.1-6 Best Effort No Duplication

The best effort service shall not deliver duplicate data.

Rationale: To keep the best effort service simple it does not have to cope with duplicate packets being received. This ought not happen as the best effort sender will only send a particular packet once even if it does not get delivered or is delivered with an error.

Risk: Low

R2.1-7 Best Effort Priority

The best effort service shall send packets of high priority before sending packets of lower priority.

Rationale: In a source where there are several packets waiting to be sent it is important that urgent packets are sent before ones that are not urgent. This can be achieved by assigning priority levels to the packets reflecting the level of urgency with which they should be sent.

Risk: Low

R2.1-8 Best Effort Assured Priority

The best effort service shall share a priority domain with the assured service i.e. a high-priority best effort packet shall be sent prior to a lower-priority assured packet and vice versa.

Rationale: The best effort and assured traffic does not use reserved resources and so is sent at any time. A best effort packet and an assured packet could be competing to be sent. In this case a mechanism is needed to decide which one should go first. A common set of priority levels is a more flexible way of doing this than sending assured traffic before best effort traffic.

Risk: Low

3.2.2 Assured Service

R2.2-1 Assured Service

SpW-RT shall provide an assured service.

Rationale: The assured service adds reliability to the best effort service. This is necessary for sending packets asynchronously where it is essential that the packets are delivered.

Risk: Low

R2.2-2 Assured Delivery

The assured service shall ensure that data is delivered to its destination.

Rationale: This is a principal role of the assured service.

Risk: Low

R2.2-3 Assured Non-Delivery Notification

If for whatever reason it is not possible to deliver data to its destination the assured service shall notify the sending entity.

Rationale: When there is an error the assured service will resend the packet that failed to be delivered. If after several retries the packet has still not been delivered then the sender needs to be informed so that it can take appropriate action.

Risk: Low

R2.2-4 Assured In-Sequence

The assured service shall deliver data in-sequence (within a priority value).

Rationale: Since the assured service ensures that all packets arrive without error, it is useful to make sure that all the packets are delivered to the user entity in the destination in the order in which they were sent.

Risk: Moderate

R2.2-5 Assured Complete

The assured service shall deliver all the data that it is requested to send.

Rationale: Missing packets are not acceptable in an assured service.

Risk: Low

R2.2-6 Assured Without Error

The assured service shall ensure that data delivered has a low probability of containing errors.

Rationale: While it is not possible to ensure that the data delivered has no error, the probability of the data having an error has to be very low. Erroneous data can lead to anomalous system behaviour.

Risk: Low

R2.2-7 Assured Discard Errors

The assured service shall discard any packet found to contain errors.

Rationale: If there is an error in a packet then it is not of use.

Risk: Low

R2.2-8 Assured Retry

If a packet goes missing or arrives in error at the destination the assured service shall send the packet again to make sure that all the data arrives at the destination.

Rationale: Resending the packet is necessary in the event of a packet being corrupted or not arriving at its intended destination.

Risk: Low

R2.2-9 Assured No Duplication

The assured service shall not deliver duplicate data.

Rationale: A packet could be delivered successfully to its destination but the acknowledgment gets corrupted so the packet is sent and delivered successfully a second time. This means that a duplicate packet has been delivered and must be discarded to avoid the data sequence being incorrect.

Risk: Low

R2.2-10Assured Priority

The assured service shall send packets of high priority before sending packets of lower priority.

Rationale: In a source where there are several packets waiting to be sent it is important that urgent packets are sent before ones that are not urgent. This can be achieved by assigning priority levels to the packets reflecting the level of urgency with which they should be sent.

Risk: Low

3.2.3 Resource-Reserved Service

R2.3-1 Resource-Reserved Service

SpW-RT shall provide a resource-reserved service.

Rationale: The resource-reserved service adds timeliness to the set of protocols supported. It provides a best effort type of service but with timeliness of delivery ensured.

Risk: Moderate

R2.3-2 Resource-Reserved Single Attempt

The resource-reserved service shall make a single attempt to deliver a data to its destination but cannot ensure that it will be delivered successfully.

Rationale: The resource-reserved service provides a timely best effort service with a single attempt at delivery. The guaranteed service provides a timely, reliable service.

Risk: Low

R2.3-3 Resource-Reserved Not In-Sequence

The resource-reserved service shall not necessarily deliver data in the order in which it was sent (within a channel and within a priority value).

Rationale: Packets going over the same route will normally arrive in the sequence in which they were sent. If group adaptive routing is used and one route gets blocked, it is possible that the packets will arrive in a different order to the order in which they were sent.

Risk: Low

R2.3-4 Resource-Reserved Without Error

The resource-reserved service shall ensure that data delivered has a low probability of containing errors.

Rationale: Erroneous data is not to be delivered. Using an error detection method like a CRC does not ensure that data does not contain errors, only that there is a low probability of there being an error.

Risk: Low

R2.3-5 Resource-Reserved Discard Errors

The resource-reserved service shall discard any data found to contain errors.

Rationale: Erroneous data is not to be delivered as it could cause unwanted or unexpected system behaviour.

Risk: Low

R2.3-6 Resource-Reserved No Duplication

The resource-reserved service shall not deliver duplicate data.

Rationale: To keep the resource-reserved service simple it does not have to cope with duplicate packets being received. This ought not happen as the resource-reserved sender will only send a particular packet once even if it does not get delivered or is delivered with an error.

Risk: Low

R2.3-7

Resource-Reserved Priority

The resource-reserved service shall send packets of high priority before sending packets of lower priority.

Rationale: In a source where there are several packets waiting to be sent it is important that urgent packets are sent before ones that are not urgent. This can be achieved by assigning priority levels to the packets reflecting the level of urgency with which they should be sent.

Risk: Low

R2.3-8 Resource-Reserved Guaranteed Priority

The resource-reserved service shall share a priority domain with the guaranteed service i.e. a high-priority resource-reserved packet shall be sent prior to a lower-priority guaranteed packet in the same channel and vice versa.

Rationale: Where channels are being used for managing resources the channels are to handle both reliable data transfer and single attempt data transfer. Reliability is not a feature of the channel but a separate dimension of quality of service. Hence resource-reserved and guaranteed services use the same channels and the same priority levels within each channel.

Risk: Low

R2.3-9 Resource-Reserved Timely

The resource-reserved service shall provide timely delivery of packets to the destination.

Rationale: The entire reason for the resource-reserved service is to reserve communication resources so that packets can be delivered in a timely, more deterministic manner than with an entirely asynchronous communications system.

Risk: High

3.2.4 Guaranteed Service

R2.4-1 Guaranteed Service

SpW-RT shall provide a guaranteed service.

Rationale: The guaranteed service combines timeliness by reserving resources with reliability by retrying failed communication attempts.

Risk: High

R2.4-2 Guaranteed Delivery

The guaranteed service shall ensure that data is delivered to its destination.

Rationale: This is one of the principal roles of the guaranteed service.

Risk: Low

R2.4-3 Guaranteed Non-Delivery Notification

If for whatever reason it is not possible to deliver data to its destination the guaranteed service shall notify the sending entity.

Rationale: When there is an error the guaranteed service will resend the packet that failed to be delivered. If after several retries the packet has still not been delivered then the sender needs to be informed so that it can take appropriate action.

Risk: Low

R2.4-4 Guaranteed In-Sequence

The guaranteed service shall deliver data in-sequence (within a channel and within a priority value).

Rationale: Since the guaranteed service ensures that all packets arrive without error, it is useful to make sure that all the packets are delivered to the user entity in the destination in the order in which they were sent.

Risk: Low

R2.4-5 Guaranteed Complete

The guaranteed service shall deliver all the data that it is requested to send.

Rationale: Missing packets are not acceptable in a guaranteed service.

Risk: Low

R2.4-6 Guaranteed Without Error

The guaranteed service shall ensure that data delivered has a low probability of containing errors.

Rationale: While it is not possible to ensure that the data delivered has no error, the probability of the data having an error has to be very low. Erroneous data can lead to anomalous system behaviour.

Risk: Low

R2.4-7 Guaranteed Discard Errors

The guaranteed service shall discard any packet found to contain errors.

Rationale: If there is an error in a packet then it is not of use.

Risk: Low

R2.4-8 Guaranteed Retry

If a packet goes missing or arrives in error at the destination the guaranteed service shall send the packet again to make sure that all the data arrives at the destination.

Rationale: Resending the packet is necessary in the event of a packet being corrupted or not arriving at its intended destination. The retry has to remain within the resource limitations of the guaranteed service.

Risk: Low

R2.4-9 Guaranteed No Duplication

The guaranteed service shall not deliver duplicate data.

Rationale: A packet could be delivered successfully to its destination but the acknowledgment gets corrupted so the packet is sent and delivered successfully a second time. This means that a duplicate packet has been delivered and must be discarded to avoid the data sequence being incorrect.

Risk: Low

R2.4-10 Guaranteed Priority

The guaranteed service shall send packets of high priority before sending packets of lower priority.

Rationale: In a source where there are several packets waiting to be sent it is important that urgent packets are sent before ones that are not urgent. This can be achieved by assigning priority levels to the packets reflecting the level of urgency with which they should be sent.

Risk: Low

R2.4-11 Guaranteed Timely

The guaranteed service shall provide timely delivery of packets to the destination.

Rationale: A principal role of the guaranteed service is to reserve communication resources so that packets can be delivered in a timely, more deterministic manner than with an entirely asynchronous communications system.

Risk: High

3.3 Address Resolution

R3-1 Local Addresses

A local address shall be used to represent user entities attached to the SpaceWire sub-network.

Rationale: The local address is something used by higher layer entities (like SOIS) to address particular user entities attached to the sub-network.

Risk: Low

R3-2 Local Address Number

There shall be a maximum of 65536 local addresses on a single SpaceWire sub-network.

Rationale: There is a choice here between using the SpaceWire logical address as the sub-network address and limiting the number of addresses to 223, or to use a larger address range. The larger number of addresses can be accessed by a combination of path, regional logical and logical addressing. 65536 seems like a reasonable maximum number of addresses on a single SpaceWire network, but it is important to note that there can potentially be many local addresses within a single SpaceWire node. The drawback with a 16-bit address is that it increases the sizes of the PDUs the addresses will have to be transferred from source to destination.

Risk: Low

R3-3 Address Resolution

SpW-RT shall convert from the local address to a SpaceWire address in order to route a packet to its intended destination.

Rationale: SpaceWire will not recognise a local address directly: it has to be translated into something that a SpaceWire network can understand e.g. a path address.

Risk: Low

3.4 Retry and Redundancy Strategies

R4-1 Retry and Redundancy Strategies

SpW-RT shall support the following retry and redundancy strategies:

- Simultaneous retry
- Resend
- Resend Redundant
- Report and Reconfigure

Rationale: The type of retry and redundancy strategy employed is very much dependent upon the user application. These four strategies cover most (if not all) possible approaches.

Risk: Moderate

R4-2 Simultaneous Retry

When simultaneous retry is being used two identical packets shall be sent over different routes through the sub-network to the intended destination.

Rationale: Rather than send one packet, wait for a failure indication and then send the packet again over a different path, it is much simpler and quicker to send two packets in the first place. This approach is only applicable when the primary and alternative routes are both active (i.e. powered up).

Risk: Moderate

R4-3 Resend

If a packet is not known to be delivered without error to its destination, an identical packet shall be resent over the same route for a specified maximum number of retry attempts.

Rationale: The reason that the packet has not reached its destination can be a transient failure and if the packet is resent over the same route it will be delivered correctly. When alternative routes are not active (i.e. the redundant part of the network is powered down) then it is worth resending on the primary path first before activating the redundant network components. The number of attempts made on the primary route can be a management parameter so that N attempts on the primary route can be made before signalling an error to the user entity or before switching automatically to the alternative path.

Risk: Moderate

R4-4 Resend Alternative

If after the resending of a packet for the specified number of attempts through the primary route the packet still has not been delivered successfully, SpW-RT shall use an alternative route through the SpaceWire network a specified maximum number of retry attempts, assuming that an alternative route is available.

Rationale: If the primary route fails continually then it is necessary to try an alternative route.

Risk: Moderate

R4-5 Retry Multiple Alternatives

If after the resending of a packet for the specified number of attempts through the alternative route the packet still has not been delivered successfully, SpW-RT shall use other alternative routes in turn a specified maximum number of retry attempts each, assuming that other alternative routes are available.

Rationale: Where several alternative paths exist they can all be tried.

Risk: Moderate

R4-6 Report Failure

If after the resending of a packet through all the possible alternative routes to the destination it is still not delivered correctly, then the user entity requesting the packet to be sent shall be informed of the failure.

Rationale: When all available routes have been tried (which could be just the primary route) then the user entity needs to be informed.

Risk: Low

R4-7 Report and Reconfigure

SpW-RT shall support the reconfiguration of a network that has failed consistently to deliver a packet to a destination.

Rationale: In the event of the autonomous approaches to error recovery either all failing or not being enabled in the first place, the error needs to be reported to a network manager (or human operator) in some way. The network can then be reconfigured to cope with the problem e.g. redundant parts of the network can be activated.

Risk: Low

R4-8 Retry Priority

The guaranteed service shall treat retry packets as any other packet as far as priority is concerned i.e. a retry packet is not sent before another packet unless the retry packet has the higher priority.

Rationale: The packet that is resent is an identical copy of the original.

Risk: Low

4 NON-FUNCTIONAL REQUIREMENTS

In this section the non-functional requirements for SpaceWire-RT are provided.

N-1 Small footprint

The SpW-RT protocols shall be efficient to implement.

Rationale: SpaceWire is a very lightweight communications protocol which is important for space applications. SpW-RT should also be lightweight with a small implementation footprint.

Risk: Moderate

N-2 Hardware or Software

SpW-RT should be implementable in hardware or software or a combination of the two.

Rationale: Hardware implementation is necessary for nodes that do not have an embedded computer. Software implementation is possible when there is an embedded computer available which is not fully loaded. Both options need to be supported as far as possible by SpW-RT.

Risk: High

N-3 Timeliness

For a moderate size SpaceWire network with three routers to be traversed the resource reserved and guaranteed service shall be able to deliver a packet within 1 millisecond

Rationale: This level of determinism is likely to support many control applications.

Risk: High

5 USE CASES

5.1 USE CASES

There are six principal use cases for SpW-RT:

- Best effort service with priority
- Assured service with primary/alternative retry
- Resource-reserved service using time-division multiplexing
- Resource-reserved service using bandwidth reservation
- Guaranteed service using time-division multiplexing and retry
- Guaranteed service using bandwidth reservation and retry

Note that time-division multiplexing and bandwidth reservations are alternative possible implementations for the resource-reserved and guaranteed services which are to be traded off during the study.

In addition the different approaches to retry are covered by the following use cases.

- Assured service with simultaneous retry
- Guaranteed service using bandwidth reservation and simultaneous retry.

This results in a total of eight use cases which are described in detail below. Note that the sequence diagrams provided to illustrate each use case show the interface to the user entity. The interaction shown between source and destination is indicative only and does not form part of the SpW-RT requirements. A service data unit (SDU) is the item of information passed between SpW-RT and the user entity. A protocol data unit (PDU) is the information that flows across the SpaceWire network carrying the SDU.

5.1.1 Best effort service with priority

A sequence diagram illustrating the best effort service with priority is illustrated in Figure 5-1.

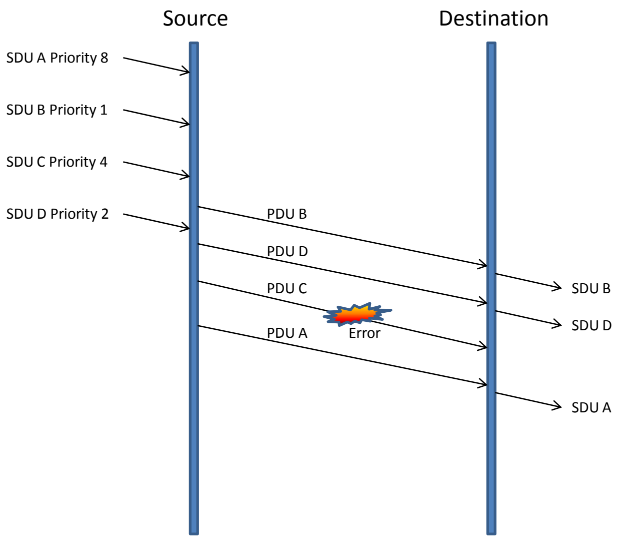

Figure 5-1 Best effort service with priority

The following sequence of events takes place:

1. Several service data units (SDUs) are passed to the source for sending to the destination, each with a different priority level.
2. The highest priority SDU (SDU B) is selected for sending and is encapsulated in a protocol data unit and sent across the SpaceWire network to the destination.
3. When the PDU arrives at the destination the information inside it is extracted and the SDU passed to the destination user entity.
4. Service data unit SDU D is now the highest priority SDU so this is transferred to the destination user entity.
5. The next SDU to be sent is SDU C, but an error occurs during sending the corresponding PDU across the network, so the information inside (SDU C) is lost.
6. Service data unit SDU A is now the highest priority SDU and is transferred across the SpaceWire network to the destination user entity.

The key characteristics of SpW-RT that are covered by this use case are:

- Best effort service with no acknowledgement of packets and no retry in the event of an error.
- Priority with the higher priority data being sent before lower priority data.

5.1.2 Assured service with primary/alternative retry

A sequence diagram illustrating the assured service with primary/alternative retry is illustrated in Figure 5-2.

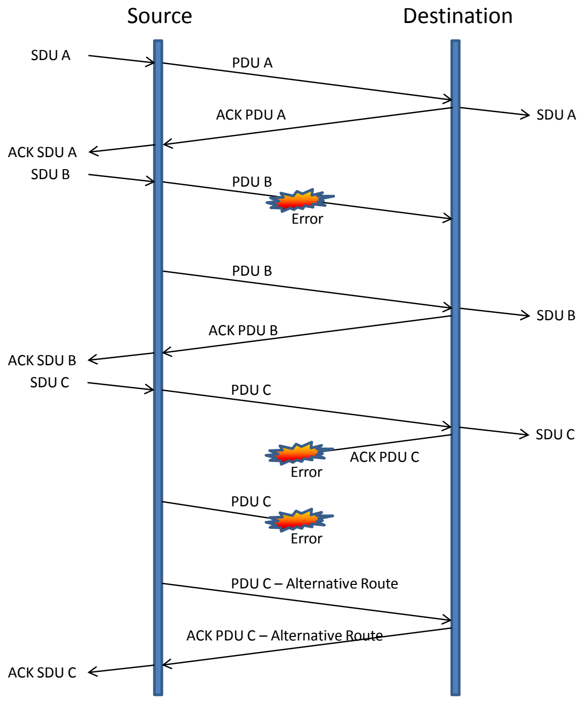

Figure 5-2 Assured service with primary/alternative retry

The following sequence of events takes place:

1. A service data unit (SDU A) is passed to the source for sending to the destination.
2. It is encapsulated in a PDU and sent across the SpaceWire network to the destination.
3. When it arrives at the destination the SDU is extracted from the PDU and passed to the destination user entity.
4. An acknowledgment (ACK PDU A) is returned to the source.
5. When the acknowledgement arrives at the source an indication is passed to the source user entity confirming that the SDU was delivered successfully.
6. The next SDU (SDU B) is passed to the source for sending, encapsulated in a PDU and sent across the SpaceWire network towards the required destination.
7. An error occurs while transferring PDU B to its destination and it arrives there containing an error.
8. The PDU (PDU B) is discarded because it contains an error and no acknowledgement is sent back to the source.
9. The source waits for an acknowledgement to arrive but does not receive one so it resends the protocol data unit (PDU B).

10. This time PDU B reaches the destination without error and the SDU is extracted and passed to the destination user entity.
11. An acknowledgement is sent back to the source.
12. When the acknowledgement arrives at the source an indication is passed to the source user entity confirming that the SDU was delivered successfully.
13. A third SDU is then passed to the source for sending (SDU C).
14. SDU C is encapsulated into a PDU and sent across the SpaceWire network arriving safely at the destination.
15. The SDU is passed to the destination user entity and an acknowledgement is sent back to the source.
16. The acknowledgement is corrupted and lost during its passage across the SpaceWire network.
17. Since no acknowledgement is received at the source it resends the SDU (SDU C) over the primary route through the network.
18. The resent SDU is also lost (possibly because the primary path through the network has a permanent fault).
19. Since no acknowledgement is received at the source and since a retry has already been sent over the primary route, SDU C is resent over the alternative route.
20. There is no problem with the alternative route so the PDU arrives safely, however this is a duplicate PDU so it is discarded.
21. An acknowledgement is sent back to the source using the alternative route.
22. The acknowledgement arrives at the source and the source user entity is informed that SDU C was delivered correctly.

The key characteristics of SpW-RT that are covered by this use case are:

- Assured service with acknowledgement of packets and retry in the event of an error.
- Retry over the primary route and then retry over the alternative route.

5.1.3 Resource-reserved service using time-division multiplexing

A sequence diagram illustrating the resource-reserved service using time-division multiplexing is illustrated in Figure 5-3.

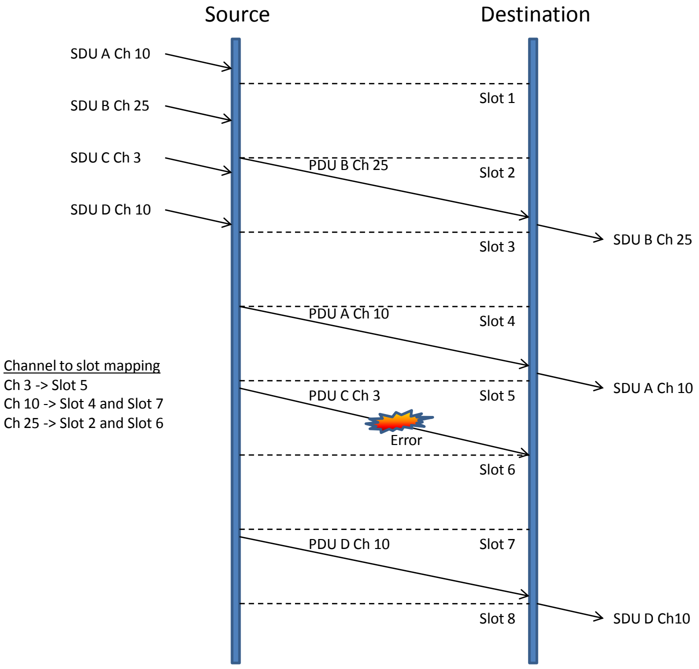

Figure 5-3 Resource-reserved service using time-division multiplexing

The following sequence of events takes place:

1. A service data unit (SDU A) is passed to the source for sending to the destination. This SDU is to be transferred over communication channel 10. The channel number identifies a set of resources needed for the communication. In this case since time-division multiplexing is being used the channel number identifies one or more time-slots during which the SDU can be transferred.

2. Several other SDUs are also submitted for sending by the source each with a channel number. These include SDU B which is assigned to channel 25 which in turn is assigned to time-slots 2 and 6.

3. When slot 2 arrives SDU B is encapsulated in PDU B and transferred across the network to the destination. During time-slot 2 no other users of the network are scheduled to use any of the network resources (SpaceWire links) that channel 25 is using.

4. PDU B arrives at its destination and the corresponding SDU is extracted and passed up to the destination user entity or channel 25.

5. Slot 4 is the next time-slot that is allocated to any of the channels form the source. In this time-slot SDUs in channel 10 can be transferred.

6. SDU A is the next SDU waiting in channel 10. It is encapsulated and transferred across the network to the destination.
7. PDU A arrives at its destination and the corresponding SDU is extracted and passed up to the destination user entity or channel 10.
8. In time-slot 5 an SDU in channel 3 can be sent. SDU C is in channel 3 and is encapsulated in PDU C and sent towards its destination.
9. PDU C is corrupted on its passage across the network, so SDU C is lost as there is no retry.
10. In time-slot 7 SDUs in channel 10 can be sent. SDU D is therefore encapsulated and transferred across the network to its destination.

The key characteristics of SpW-RT that are covered by this use case are:

- Resource-reserved service using time-division multiplexing to manage the use of communication resources.
- The effect of errors on the information transfer.

5.1.4 Resource-reserved service using bandwidth reservation

A sequence diagram illustrating the resource-reserved service using bandwidth reservation is illustrated in Figure 5-4.

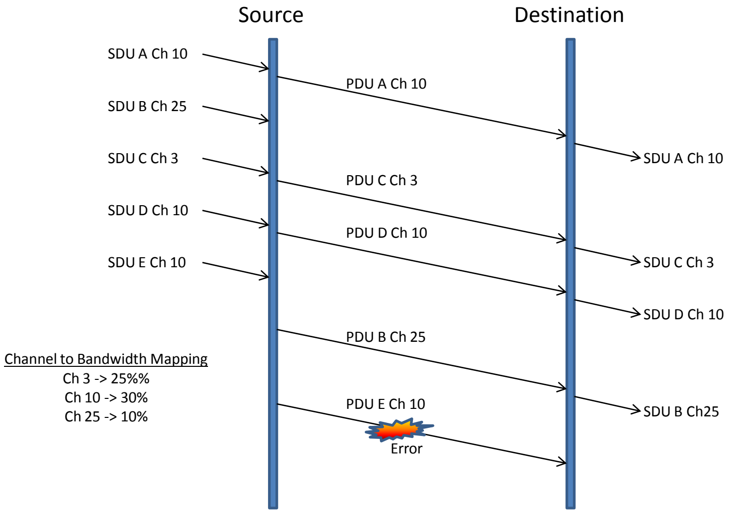

Figure 5-4 Resource-reserved service using bandwidth reservation

The following sequence of events takes place:

1. A service data unit (SDU A) is passed to the source for sending to the destination. This SDU is to be transferred over communication channel 10. The channel number identifies a set of

resources needed for the communication. In this case since bandwidth reservation is being used a bandwidth limit is associated with the channel number along with a bandwidth utilisation measurement which reflects the amount of traffic that has been sent through that channel.

2. Since the bandwidth utilisation measurement is substantially less than the bandwidth limit there is capacity in channel 10 to send SDU A straightaway. It is encapsulated in PDU A and sent across the network.
3. When PDU A reaches its destination, SDU A is unpacked and passed on to the destination user entity for channel 10.
4. SDU B is submitted for sending over channel 25 but the bandwidth utilisation measurement indicates that this channel has had its fair share of bandwidth so the SDU cannot be sent just yet.
5. SDU C is submitted and sent over channel 3.
6. SDU D is submitted and sent over channel 10. This brings the bandwidth utilisation measurement close to the limit for channel 10.
7. Another SDU (SDU E) is submitted for sending over channel 10, but channel 10 is now close to its bandwidth limitation so cannot be sent.
8. In the meantime since no PDUs have been sent over channel 25 its bandwidth utilisation has fallen and now SDU B can be sent.
9. Sometime later the bandwidth utilisation of channel 10 has fallen enough so that SDU E can be transferred.
10. The PDU corresponding to SDU E is corrupted during transfer so this SDU is not passed to the destination user entity for channel 10.

The key characteristics of SpW-RT that are covered by this use case are:

- Resource-reserved service using bandwidth reservation to manage the use of communication resources.
- The effect of errors on the information transfer.

5.1.5 Guaranteed service using time-division multiplexing and retry

A sequence diagram illustrating the guaranteed service using time-division multiplexing and retry is illustrated in Figure 5-5.

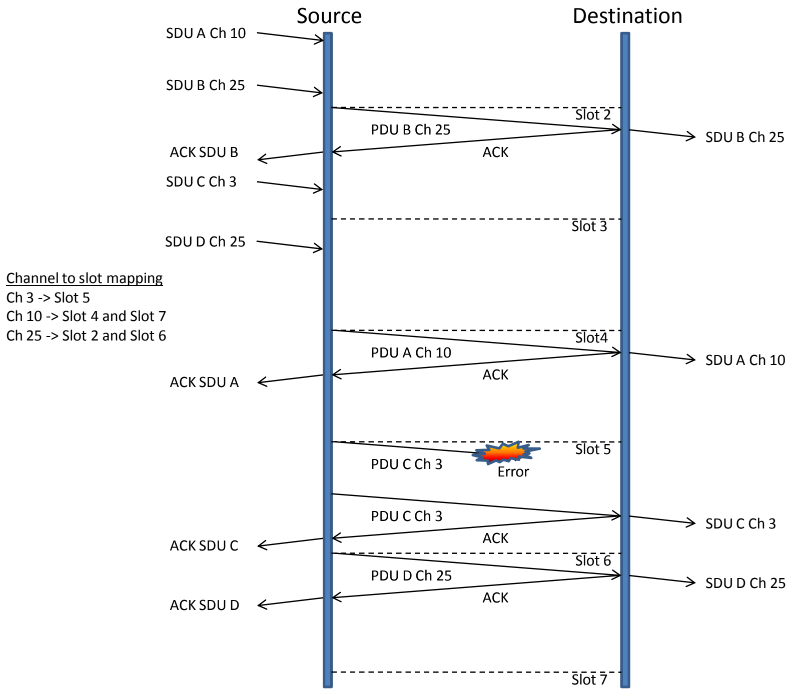

Figure 5-5 Guaranteed service using time-division multiplexing and retry

The following sequence of events takes place:

1. A service data unit (SDU A) is passed to the source for sending to the destination. This SDU is to be transferred over communication channel 10. The channel number identifies a set of resources needed for the communication. In this case since time-division multiplexing is being used the channel number identifies one or more time-slots during which the SDU can be transferred.
2. Several other SDUs are also submitted for sending by the source each with a channel number. These include SDU B which is assigned to channel 25 which in turn is assigned to time-slots 2 and 6.
3. When slot 2 arrives SDU B is encapsulated in PDU B and transferred across the network to the destination. During time-slot 2 no other users of the network are scheduled to use any of the network resources (SpaceWire links) that channel 25 is using.
4. PDU B arrives at its destination and the corresponding SDU is extracted and passed up to the destination user entity or channel 25.
5. An acknowledgement to PDU B is send from the destination to the source of the PDU.

6. When the acknowledgement arrives the safe arrival of the PDU B at its destination is reported to the user entity of channel 25. Note that the complete transaction occurs within the limits of the time-slot.
7. Slot 4 is the next time-slot that is allocated to any of the channels form the source. In this time-slot SDUs in channel 10 can be transferred.
8. SDU A is the next SDU waiting in channel 10. It is encapsulated and transferred across the network to the destination.
9. PDU A arrives at its destination and the corresponding SDU is extracted and passed up to the destination user entity or channel 10.
10. An acknowledgement is send and the safe arrival of PDU A at its destination is reported to the user entity of channel 10.
11. In time-slot 5 and SDU in channel 3 can be sent. SDU C is in channel 3 and is encapsulated in PDU C and sent towards its destination.
12. PDU C is corrupted on its passage across the network.
13. Since PDU C does not arrive at its destination no acknowledgement is sent.
14. The source realises that no acknowledgement has been received for PDU C so it resends the PDU.
15. The PDU is delivered successfully and SDU C is extracted and passed to the user entity of channel 3.
16. An acknowledgement is returned to the source of PDU C and when it arrives the user entity of channel 3 is informed that SDU C was delivered successfully.
17. In time-slot 6 channel 25 is allowed to send traffic so PDU D is transferred to its destination and acknowledged.

The key characteristics of SpW-RT that are covered by this use case are:

- Guaranteed service using time-division multiplexing to manage the use of communication resources.
- The effect of errors on the information transfer.

Note: since the entire transaction including acknowledgement and any retry has to be included in the time-slot, it may be more efficient to send the PDU wait a little and then send the PDU a second time (over the same route) without waiting for an acknowledgement or a time-out.

5.1.6 Guaranteed service using bandwidth reservation and retry

A sequence diagram illustrating the guaranteed service using bandwidth reservation and retry is illustrated in Figure 5-6.

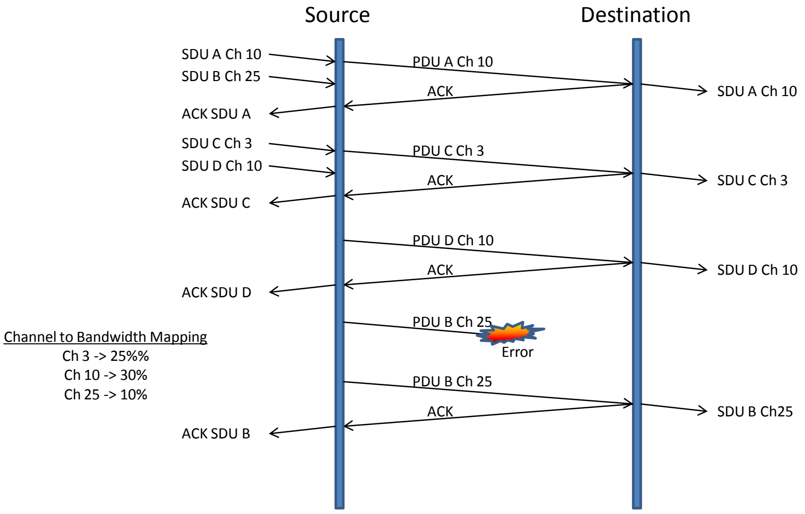

Figure 5-6 Guaranteed service using bandwidth reservation and retry

1. A service data unit (SDU A) is passed to the source for sending to the destination. This SDU is to be transferred over communication channel 10. The channel number identifies a set of resources needed for the communication. In this case since bandwidth reservation is being used a bandwidth limit is associated with the channel number along with a bandwidth utilisation measurement which reflects the amount of traffic that has been sent through that channel.

2. Since the bandwidth utilisation measurement is substantially less than the bandwidth limit there is capacity in channel 10 to send SDU A straightaway. It is encapsulated in PDU A and sent across the network.

3. When PDU A reaches its destination, SDU A is unpacked and passed on to the destination user entity for channel 10.

4. An acknowledgement is send back to the source of PDU A

5. When the acknowledgement arrives at the source the user entity of channel 10 is informed that SDU A was delivered successfully.

6. SDU B is submitted for sending over channel 25 but the bandwidth utilisation measurement indicates that this channel has had its fair share of bandwidth so the SDU cannot be sent just yet.

7. SDU C is submitted and sent over channel 3.

8. PDU C arrives successfully at the destination, SDU C is extracted from the PDU and passed to the destination user entity of channel 3.
9. When the acknowledgement to PDU C arrives at the source the user entity of channel 10 is informed that SDU A was delivered successfully.
10. SDU D is submitted and sent over channel 10. This brings the bandwidth utilisation measurement close to the limit for channel 10.
11. In the meantime since no PDUs have been sent over channel 25 its bandwidth utilisation has fallen and now SDU B can be sent.
12. PDU B corresponding to SDU B is corrupted during transfer so the SDU is not passed to the destination user entity for channel 25.
13. Since PDU B was corrupted no acknowledgement arrives back at its source.
14. The source realises that no acknowledgement has been received for PDU B and resends it.
15. This time PDU B successfully reaches its destination, the SDU is extracted and passed to the destination user entity of channel 25.
16. An acknowledgement is sent to the source of PDU B.
17. When this acknowledgement is received the safe arrival of SDU B at its destination is reported to the user entity of channel 25.

The key characteristics of SpW-RT that are covered by this use case are:

- Guaranteed service using bandwidth reservation to manage the use of communication resources.
- The effect of errors on the information transfer.

5.1.7 Assured service with simultaneous retry

A sequence diagram illustrating the assured service with simultaneous retry is illustrated in Figure 5-7.

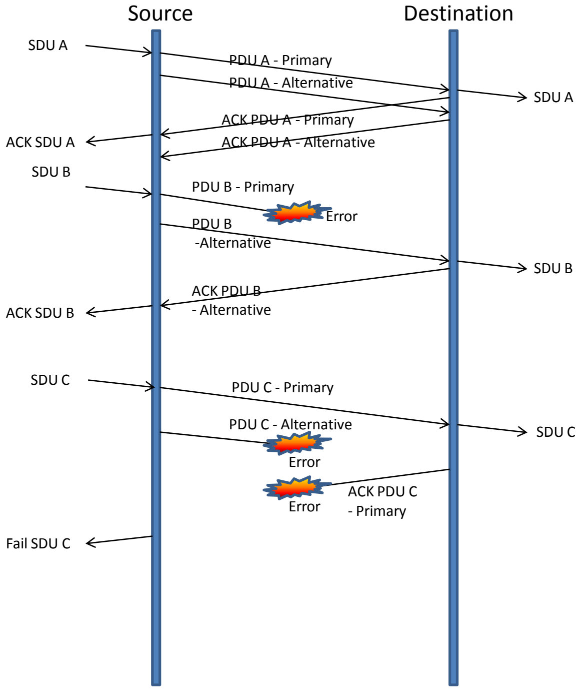

Figure 5-7 Assured service with simultaneous retry

The following sequence of events takes place:

1. A service data unit (SDU A) is passed to the source for sending to the destination.
2. It is encapsulated twice in two PDUs one of which is sent across the SpaceWire network to the destination using the primary router and the other using an alternative route.
3. The primary PDU A arrives at the destination and SDU A is extracted from the PDU and passed to the destination user entity.
4. An acknowledgment (ACK PDU A) is returned to the source using the primary route.
5. In the meantime the alternative PDU A arrives at the destination. Since it is a duplicate of the primary PDU A no SDU is passed to the destination user entity. An acknowledgement is, however, returned to the source using the alternative route.
6. When the primary acknowledgement arrives at the source an indication is passed to the source user entity confirming that the SDU was delivered successfully.
7. When the secondary acknowledgement arrives at the source it is discarded because successful delivery of SDU A has already been reported.
8. The next SDU (SDU B) is passed to the source for sending. It is encapsulated twice in two PDUs which are sent across the SpaceWire network to the required destination using the primary and alternative routes.

9. An error occurs while transferring the primary PDU B to its destination and it arrives at the destination containing an error.
10. The PDU (PDU B) is discarded because it contains an error and no acknowledgement is send back to the source.
11. The alternative PDU B arrives at the destination without error and SDU B is extracted and passed to the destination user entity.
12. An acknowledgment (ACK PDU B) is returned to the source using the alternative route.
13. When the alternative acknowledgement arrives at the source an indication is passed to the source user entity confirming that SDU B was delivered successfully.
14. SDU C is now passed to the source for sending. It is encapsulated twice in two PDUs which are sent across the SpaceWire network to the required destination using the primary and alternative routes.
15. The primary PDU C arrives at the destination without error and SDU C is extracted and passed to the destination user entity.
16. An acknowledgment (ACK PDU C) is returned to the source using the alternative route.
17. This acknowledgement is corrupted and lost
18. An error also occurs while transferring the alternative PDU C to its destination and it arrives at the destination containing an error.
19. The alternative PDU C is discarded because it contains an error and no acknowledgement is send back to the source.
20. Since no acknowledgement was sent for the alternative PDU C and since the acknowledgement for the primary PDU C was corrupted, no acknowledgement reaches the source of PDU C.
21. The source detects that no acknowledgement has been received and informs the user entity that it failed to deliver SDU C. In fact SDU C has been delivered.

The key characteristics of SpW-RT that are covered by this use case are:

- Assured service with simultaneous retry.
- Reporting of a failure when the simultaneous retry is unsuccessful.

5.1.8 Guaranteed service using bandwidth reservation and simultaneous retry

A sequence diagram illustrating the guaranteed service using bandwidth reservation and simultaneous retry is illustrated in Figure 5-8.

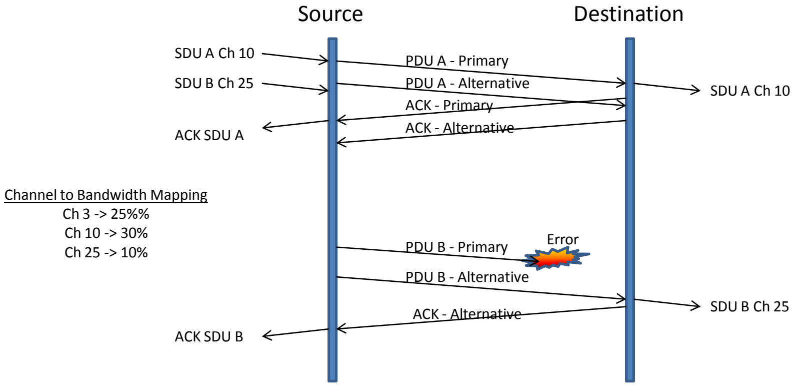

Figure 5-8 Guaranteed service using bandwidth reservation and simultaneous retry

1. A service data unit (SDU A) is passed to the source for sending to the destination. This SDU is to be transferred over communication channel 10. The channel number identifies a set of resources needed for the communication. In this case since bandwidth reservation is being used a bandwidth limit is associated with the channel number along with a bandwidth utilisation measurement which reflects the amount of traffic that has been sent through that channel.
2. Since the bandwidth utilisation measurement is substantially less than the bandwidth limit, there is capacity in channel 10 to send SDU A straightaway. It is encapsulated twice into two PDUs and sent across the network using the primary and alternative routes.
3. When primary PDU A reaches its destination, SDU A is unpacked and passed on to the destination user entity for channel 10.
4. An acknowledgement is sent back to the source of PDU A using the primary route.
5. When the acknowledgement arrives at the source the user entity of channel 10 is informed that SDU A was delivered successfully.
6. When alternative PDU A reaches its destination it is discarded as it is a duplicate PDU.
7. An acknowledgement is send back to the source of PDU A using the alternative route.
8. When the acknowledgement arrives at the source it is discarded because PDU A has already been acknowledged.
9. SDU B is submitted for sending over channel 25 but the bandwidth utilisation measurement indicates that this channel has had its fair share of bandwidth so the SDU cannot be sent just yet.
10. Sometime later since no PDUs have been sent over channel 25 its bandwidth utilisation has fallen and now SDU B can be sent.

11. SDU B is encapsulated twice into two PDUs which are sent over the primary and alternative routes to the destination.
12. The primary PDU B is corrupted on its way to the destination and is not received at the destination.
13. The alternative PDU B successfully reaches its destination so the SDU is extracted and passed to the destination user entity of channel 25.
14. An acknowledgement is sent to the source of PDU B over the alternative route.
15. When this acknowledgement is received the safe arrival of SDU B at its destination is reported to the user entity of channel 25.

The key characteristics of SpW-RT that are covered by this use case are:

- Simultaneous retry operation with the guaranteed service.
- The effect of errors on the information transfer.

6 DOCUMENT CHANGES

The changes made this document are listed in this section.

Table 6-1: Changes to Document (Issue 1.0 to Issue 1.0)

Section/Reference	Change