SpaceWire Router

Data Sheet

Ref:	UoD_SpW-10X_DataSheet
Document Revision:	Issue 2.0
Date:	18th August 2006

Prepared by -	Chris McClements, University of Dundee Steve Parkes, University of Dundee
Checked by -	Steve Parkes, University of Dundee
ESA Manager -	Pierre Fabry, ESTEC

Document Change log

Date	Issue	Comments	Author
19th-March-2004	Issue 1.0	Initial draft version	Chris McClements
26th-August-2004	Issue 1.2	Timing for FPGA model added	Chris McClements
16 November 2004	Issue 1.3	Register Definitions Updated	Chris McClements
27 April 2004	Issue 1.4	Latency and Jitter Specifications added	Chris McClements
2 May 2005	Issue 1.5	Footer indicates "Preliminary." Section 8.6 subject to change notice added to front page.	Steve Parkes
21 December 2005	Issue 1.6	RMAP section added	Chris McClements
19th July 2006	Issue 1.7	Corrections and clarifications	Chris McClements
18th August 2006	Issue 2.0	Editorial changes and clarifications	Steve Parkes

CONTENTS

CONTENTS 3

I LIST OF FIGURES 7

II LIST OF TABLES 8

1. INTRODUCTION 10
1.1 TERMS AND ACRONYMS 10
1.2 DOCUMENTS 10

2. APPLICATIONS 12
2.1 STAND-ALONE ROUTER 12
2.2 NODE INTERFACE 12
2.3 EMBEDDED ROUTER 12
2.4 EXPANDING THE NUMBER OF ROUTER PORTS 12

3. FUNCTIONAL OVERVIEW 13
3.1 SPACEWIRE PORTS 14
3.2 EXTERNAL PORTS 14
3.3 CONFIGURATION PORT 15
3.4 ROUTING TABLE 15
3.5 ROUTING CONTROL LOGIC AND CROSSBAR 15
3.6 TIME-CODE PROCESSING 16
3.7 CONTROL/STATUS REGISTERS 16

4. DEVICE INTERFACE 17
4.1 GLOBAL SIGNALS 17
4.2 SPACEWIRE SIGNALS 18
4.3 EXTERNAL PORT DATA SIGNALS 20
4.4 TIME-CODE SIGNALS 22
4.5 LINK ERROR INDICATION SIGNALS 23
4.6 RESET CONFIGURATION SIGNALS 24

4.7 POWER AND GROUND SIGNALS 27

5. INTERFACE OPERATIONS 28

5.1 EXTERNAL PORT INTERFACE OPERATION 28

5.2 TIME-CODE INTERFACE OPERATION 29

5.3 ERROR/STATUS ERROR INDICATION INTERFACE OPERATION 30

5.4 RESET CONFIGURATION INTERFACE OPERATION 32

6. SWITCHING CHARACTERISTICS 33

6.1 CLOCK AND RESET TIMING PARAMETERS 33

6.2 SERIAL SIGNALS TIMING PARAMETERS 33

6.3 EXTERNAL PORT TIMING PARAMETERS 34

6.4 TIME-CODE INTERFACE TIMING PARAMETERS 35

6.5 ERROR/STATUS INTERFACE TIMING PARAMETERS 36

6.6 LATENCY AND JITTER 37
6.6.1 Clock Periods 37
6.6.2 Switching Latency 37
6.6.3 Router Latency 37
6.6.4 Time-code Latency 38
6.6.5 Time-code Jitter 39
6.6.6 200M bits/s Input and Output Bit Rate Example 39

7. ELECTRICAL CHARACTERISTICS 40

7.1 ABSOLUTE MAXIMUM RATINGS 40

7.2 DC CHARACTERISTICS 40

7.3 OPERATING TEMPERATURE RANGE 40

8. SPACEWIRE ROUTER PACKET TYPES 41

8.1 PACKET ADDRESSES 41

8.2 PACKET PRIORITY 41

8.3 PACKET HEADER DELETION 42

8.4 INVALID ADDRESSES 43

8.5 DATA PACKETS 43

8.6 COMMAND PACKETS 44
8.6.1 Supported Commands 44
8.6.2 Read Command 45

8.6.3 Read Incrementing Command 49
8.6.4 Read Modify Write Command 54
8.6.5 Write Command 59
8.6.6 Command Error Response 63
8.6.7 Command Packet Cyclic Redundancy Check 65
8.6.8 Local Source Path Address....65
8.6.9 Source Path Address Field 66
8.6.10 Command Packet Fill Bytes....67

9. CONTROL LOGIC AND OPERATIONAL MODES....68

9.1 SPACEWIRE LINK CONTROL....68
9.1.1 Default operating mode....68
9.1.2 Auto-Start 68
9.1.3 Link-Start....68
9.1.4 Link-Disable 69
9.1.5 Automatic tri-state enable mode 69
9.1.6 Setting the data-rate....69
9.1.7 Setting the default 10Mbits/s data-rate 70
9.1.8 Data rate comparison table....71

9.2 GLOBAL SPACEWIRE LINK CONTROL 71
9.2.1 Automatic Link-Start mode....71
9.2.2 Automatic Link-Disable mode 72

9.3 CONTROL LOGIC AND ROUTING....73
9.3.1 Packet address error....73
9.3.2 Arbitration....73
9.3.2.1 Arbitration of packets with matching priority (1)....74
9.3.2.2 Arbitration of packets with matching priority (2)....75
9.3.2.3 Arbitration of packets with different priority (1)....76
9.3.2.4 Arbitration of packets with different priority (2)....77
9.3.3 Group Adaptive Routing....78
9.3.3.1 Normal Group adaptive routing....78
9.3.3.2 Group adaptive routing when busy 78
9.3.3.3 Group adaptive routing when ports not ready....79
9.3.4 Loop-back routing 80
9.3.5 Packet Blocking....81
9.3.5.1 Blocking Allowed 82
9.3.5.2 Blocking Watchdog Timer 82
9.3.5.3 Scenario 1 Destination Node Blocked 82
9.3.5.4 Scenario 2 Source Node Stalled....84
9.3.5.5 Scenario 3 Destination node slow, Source node waiting....86
9.3.5.6 Scenario 4 Destination port not ready....88

10. REGISTER DEFINITIONS 91

10.1 INTERNAL MEMORY MAP 91

10.2 REGISTER ADDRESSES SUMMARY 92

10.3 GROUP ADAPTIVE ROUTING TABLE REGISTERS 93

10.4 PORT CONTROL/STATUS REGISTERS 95
10.4.1 Generic port control/status register fields. 96
10.4.2 Configuration port control/status register fields. 96
10.4.3 SpaceWire link interface port control/status register bits. 99
10.4.4 External port control/status register bits. 102

10.5 ROUTER CONTROL/STATUS REGISTERS 102
10.5.1 Network Discovery Register 102
10.5.2 Router Identity Register 103
10.5.3 Router Control Register 104
10.5.4 Error active Register 106
10.5.5 Time-Code Register 107
10.5.6 Device Manufacturer and Chip ID Register 108
10.5.7 General Purpose Register 109
10.5.8 Time-Code Enable Register 109
10.5.9 Transmit Clock Control Register 110
10.5.10 Destination Key Register 113
10.5.11 Unused Registers and Register Bits 113
10.5.12 Empty packets 113

10.6 WRITING TO A READ-ONLY REGISTER 113

11. DOCUMENT CHANGES 114

11.1 ISSUE 1.7 TO ISSUE 2.0 114

11.2 ISSUE 1.6 TO ISSUE 1.7 114

11.3 ISSUE 1.5 TO ISSUE 1.6 114

11.4 ISSUE 1.4 TO ISSUE 1.5 114

11.5 ISSUE 1.3 TO ISSUE 1.4 114

11.6 ISSUE 1.2 TO ISSUE 1.3 114

11.7 ISSUE 1.1 TO ISSUE 1.2 114

11.8 ISSUE 1.0 TO ISSUE 1.1 115

I LIST OF FIGURES

Figure 3-1 SpaceWire router block diagram ... 14
Figure 4-1 Configuration interface timing specification ... 26
Figure 5-1 External port write timing specification ... 28
Figure 5-2 External port read timing specification ... 28
Figure 5-3 Time-Code Input Interface ... 29
Figure 5-4 Time-Code Output Interface ... 29
Figure 5-5 Time-code reset interface ... 30
Figure 5-6 Status Multiplexer output interface ... 30
Figure 5-7 Reset configuration interface timing specification ... 32
Figure 6-1 DS minimum consecutive edge separation ... 33
Figure 6-2 External port input FIFO timing parameters ... 34
Figure 6-3 External port output FIFO timing parameters ... 34
Figure 6-4 Time-Code Input Interface ... 35
Figure 6-5 Time-Code Output Interface ... 35
Figure 6-6 Time-code TIME_CTR_RST interface ... 35
Figure 8-1 Normal router data packets ... 44
Figure 8-2 Command Packet Format ... 44
Figure 8-3 Read Single Address Command Format ... 46
Figure 8-4 Read Single Address Reply Packet Format ... 48
Figure 8-5 Read Incrementing Address Command Format ... 51
Figure 8-6 Read Incrementing Address Reply Packet Format ... 53
Figure 8-7 Read-Modify-Write Command Packet Format ... 55
Figure 8-8 Read-Modify-Write example operation ... 57
Figure 8-9 Read-Modify-Write Reply Packet Format ... 58
Figure 8-10 Write Single Address Command Packet ... 60
Figure 8-11 Write Single Address Reply Packet ... 62
Figure 8-12 Source Path Address field decoding ... 67
Figure 8-13 Source Path Addresses in Reply Packet ... 67
Figure 8-14 Normal Configuration Packet Header Structure ... 67
Figure 8-15 Fill Bytes Configuration Header Structure ... 67
Figure 9-1 Tri-state enable operating mode ... 69
Figure 9-2 Tx10MBitDIV equation ... 70
Figure 9-3 Automatic Link-Start mode ... 72
Figure 9-4 Automatic Link-Disable mode ... 72
Figure 9-5 Arbitration of two packets with matching priority. ... 74
Figure 9-6 Arbitration of three packets with matching priority ... 75
Figure 9-7 Arbitration of two packets with different priority ... 76
Figure 9-8 ... 77

FIGURE 9-9 NORMAL GROUP ADAPTIVE ROUTING 78
FIGURE 9-10 GROUP ADAPTIVE ROUTING WHEN OTHER PORTS BUSY 79
FIGURE 9-11 GROUP ADAPTIVE ROUTING WHEN PORTS NOT READY 79
FIGURE 9-12 PACKET ADDRESS SELF MODE 80
FIGURE 9-13 SCENARIO 1 DESTINATION BLOCKED (BLOCKING ALLOWED MODE) 83
FIGURE 9-14 SCENARIO 1 DESTINATION BLOCKED (WATCHDOG TIMER MODE) 84
FIGURE 9-15 SCENARIO 2 SOURCE NODE STALLED (BLOCKING ALLOWED MODE) 85
FIGURE 9-16 SCENARIO 2 SOURCE NODE STALLED (WATCHDOG TIMER MODE) 86
FIGURE 9-17 SCENARIO 3 DESTINATION SLOW (BLOCKING ALLOWED MODE) 87
FIGURE 9-18 SCENARIO 3 DESTINATION SLOW (WATCHDOG TIMER MODE) 88
FIGURE 9-19 SCENARIO 4 DESTINATION PORT NOT READY (BLOCKING ALLOWED MODE) 89
FIGURE 9-20 SCENARIO 4 DESTINATION PORT NOT READY (BLOCKING WATCHDOG TIMER) 90
FIGURE 10-1 ROUTER INTERNAL MEMORY MAP 91
FIGURE 10-2 GAR REGISTER FIELDS 93
FIGURE 10-3 SPACEWIRE PORT CONTROL/STATUS REGISTER FIELDS 99
FIGURE 10-4 NETWORK DISCOVERY REGISTER FIELDS 103
FIGURE 10-5 ROUTER CONTROL REGISTER FIELDS 104
FIGURE 10-6 ERROR ACTIVE REGISTER FIELDS 107
FIGURE 10-7 TIME-CODE REGISTER FIELDS 108
FIGURE 10-8 DEVICE MANUFACTURER AND CHIP ID REGISTER FIELDS 108
FIGURE 10-9 TIME-CODE ENABLE REGISTER FIELDS 110
FIGURE 10-10 TRANSMIT CLOCK CONTROL REGISTER 111
FIGURE 10-11 TX10MBITDIV EQUATION 113

II LIST OF TABLES

TABLE 1-1 APPLICABLE DOCUMENTS 11
TABLE 4-1 GLOBAL SIGNALS 17
TABLE 4-2 DATA AND STROBE SPACEWIRE SIGNALS 18
TABLE 4-3 EXTERNAL PORT INTERFACE SIGNALS 20
TABLE 4-4 TIME-CODE SIGNALS 22
TABLE 4-5 LINK ERROR INDICATION SIGNALS 23
TABLE 4-6 RESET CONFIGURATION SIGNALS 25
TABLE 4-7 RESET CONFIGURATION SIGNALS 27
TABLE 5-1 MULTIPLEXED STATUS PINS BIT ASSIGNMENT 31
TABLE 6-1 CLOCK AND RESET TIMING PARAMETERS 33
TABLE 6-2 SERIAL SIGNAL TIMING PARAMETERS 33
TABLE 6-3 EXTERNAL PORT TIMING PARAMETERS 34
TABLE 6-4 TIME-CODE INTERFACE TIMING PARAMETERS 35
TABLE 6-5 STATUS MULTIPLEXER TIMING PARAMETERS 36
TABLE 6-6 SPACEWIRE ROUTER LATENCY AND JITTER MEASUREMENTS (BIT RATE = 200MBITS/S 39

Table 8-1 Packet Address Mapping ... 41
Table 8-2 Packet Priority Mapping ... 42
Table 8-3 Packet Header Deletion Mapping ... 43
Table 8-4 Supported RMAP Command Codes ... 45
Table 8-5 Read Single Address Characteristics ... 46
Table 8-6 Read Single Address Command Packet Fields ... 47
Table 8-7 Read Single Address Reply Packet Fields ... 48
Table 8-8 Read Incrementing Address Characteristics ... 50
Table 8-9 Read Incrementing Address Command Packet Fields ... 51
Table 8-10 Read Incrementing Address Reply Packet Fields ... 53
Table 8-11 Read-Modify-Write Command Characteristics ... 54
Table 8-12 Read-Modify-Write Command Packet Fields ... 56
Table 8-13 Read-Modify-Write Reply Packet Fields ... 58
Table 8-14 Write Command Characteristics ... 59
Table 8-15 Write Single Address Command Packet Fields ... 60
Table 8-16 Write Single Address Reply Packet Fields ... 62
Table 8-17 Configuration Port Errors Summary ... 63
Table 8-18 Source Path Address Reference Table ... 66
Table 9-1 Port ready definitions ... 81
Table 9-2 Packet blocking ... 81
Table 10-1 Types of Register within Configuration Port ... 92
Table 10-2 Configuration Register Addresses ... 93
Table 10-3 GAR Table Register Description ... 95
Table 10-4 Configuration Port Control/Status Register Fields ... 96
Table 10-5 Configuration Port Control/Status Register Fields ... 97
Table 10-6 SpaceWire Port Control/Status Register Fields. ... 100
Table 10-7 External Port Contol/Status Fields ... 102
Table 10-8 Network Discovery Register Fields ... 103
Table 10-9 Router Identity Register Field ... 104
Table 10-10 Router Control Register Fields ... 105
Table 10-11 Error Active Register Fields ... 107
Table 10-12 Time-Code Register Fields ... 108
Table 10-13 Device Manufacturer and Chip ID Register Fields ... 109
Table 10-14 Time-Code Enable Register Fields ... 110
Table 10-15 Transmit Clock Control Register Bits ... 112
Table 10-16 Destination Key Register ... 113

1. INTRODUCTION

This document defines a technical reference for the implementation and operation of the SpaceWire Router device, SpW-10X. Detailed timing information for the ASIC implementation is not yet available.

1.1 TERMS AND ACRONYMS

3V3	3.3 volt interface levels.
ACK	Acknowledge
AD	Applicable Document
CLK	Input clock to the SpaceWire router
CRC	Cyclic Redundancy Check
DC	Direct Current
EEP	Error end of packet, used to denote an error occurred during packet transfer.
EOP	End of packet used to denote a normal end of packet in SpaceWire.
FIFO	First in - First out buffer used to transfer data between logic.
FPGA	Field Programmable Gate Array.
LVDS	Low voltage differential signalling.
NACK	Negative acknowledge (error acknowledge)
RD	Read
RMAP	Remote Memory Access Protocol
RST	Aysnchronous reset to the SpaceWire router.
UoD	University of Dundee.
WR	Write

1.2 DOCUMENTS

In this section the documents referenced in this document are listed.

Table 1-1 Applicable Documents		
REF	**Document Number**	**Document Title**
AD1	ECSS-E5O-12A	SpaceWire - links, nodes, routers and networks.
AD2	ECSS-E50-11 Draft E	SpaceWire Remote Memory Access Protocol

2. APPLICATIONS

The SpaceWire router applications are defined in the following sections

2.1 STAND-ALONE ROUTER

The SpaceWire Router may be used as a stand-alone router with up to eight SpaceWire links connected to it. Configuration of the routing tables etc. may be done by sending SpaceWire packets containing configuration commands to the router.

2.2 NODE INTERFACE

The SpaceWire Router has two external ports which enable the device to be used as a node interface. The equipment to be connected to the SpaceWire network is attached to one or both external ports. One or more SpaceWire ports are used to provide the connection into the SpaceWire network. Unused SpaceWire ports may be disabled and their outputs tri-stated to save power. In this arrangement configuration of the routing tables and other parameters may be done by sending configuration packets from the local host via an external port or from a remote network manager via a SpaceWire port.

2.3 EMBEDDED ROUTER

The SpaceWire Router device can also be used to provide a node with an embedded router. In this case the external ports are used to provide the local connections to the node and the SpaceWire ports are used to make connections to other ports in the network. The difference between this configuration and that of section 2.2 is just a conceptual one with the Node interface configuration normally using fewer SpaceWire ports than the Embedded Router configuration.

2.4 EXPANDING THE NUMBER OF ROUTER PORTS

If a routing switch with a larger number of SpaceWire (or external) ports is required then this can be accomplished by joining together two or more routers using some of the SpaceWire links. For example using two SpaceWire links to join together two router devices would create an effective router with twelve SpaceWire ports and four external ports. Note, however, that an extra path addressing byte is needed to route packets between the two routers and that there is additional routing delay.

3. FUNCTIONAL OVERVIEW

A SpaceWire routing switch is able to connect together many nodes, providing a means of routing packets between the nodes connected to it. It comprises a number of SpaceWire link interfaces and a routing matrix. The routing matrix enables packets arriving at one link interface to be transferred to and sent out of another link interface on the routing switch

The SpaceWire router comprises the following functional logic blocks:

- Eight SpaceWire bi-directional serial ports.
- Two external parallel input/output ports each comprising an input FIFO and an output FIFO.
- A non-blocking crossbar switch connecting any input port to any output port.
- An internal configuration port accessible via the crossbar switch from the external parallel input/output port or the SpaceWire input/output ports.
- A routing table accessible via the configuration port which holds the logical address to output port mapping.
- Control logic to control the operation of the switch, performing arbitration and group adaptive routing.
- Control registers than can be written and read by the configuration port and which hold control information e.g. link operating speed.
- An external time-code interface comprising tick_in, tick_out and current tick count value
- Internal status/error registers accessible via the configuration port
- External status/error signals

A block diagram of the routing switch is given in Figure 3-1.

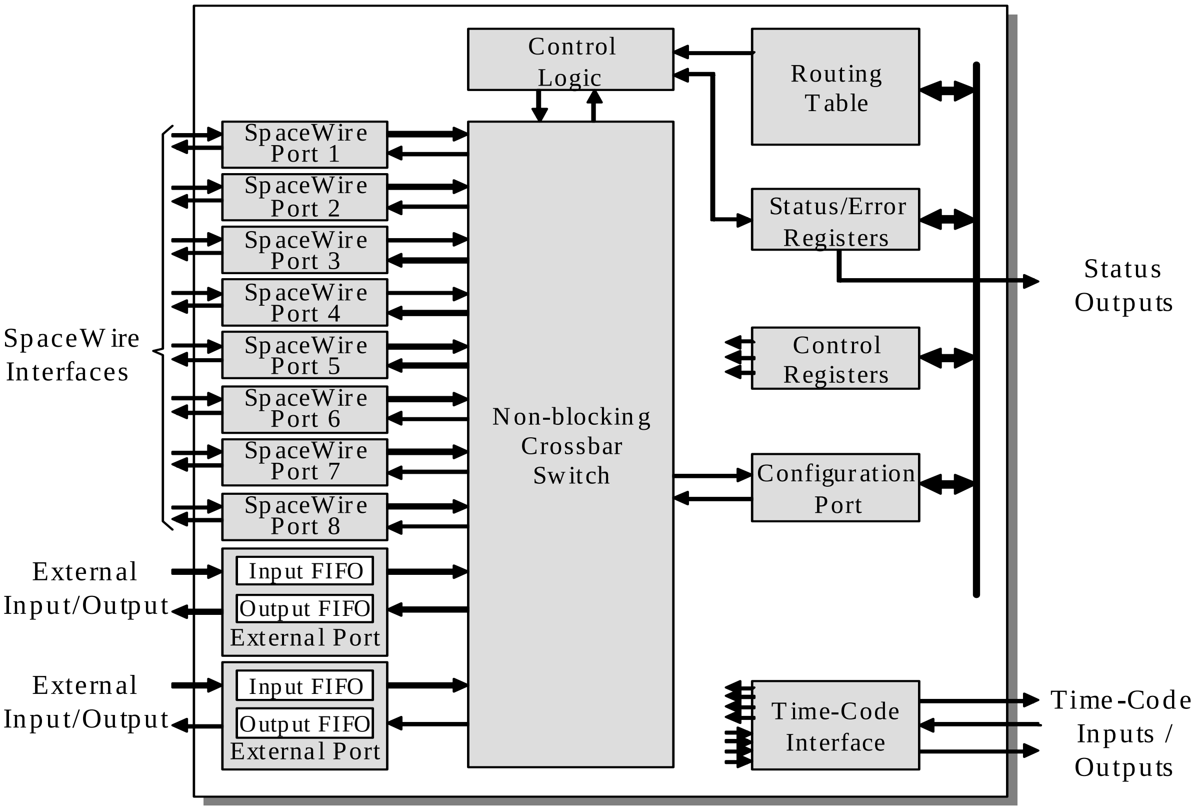


Figure 3-1 SpaceWire router block diagram

The following paragraphs define the SpaceWire router functional logic blocks in more detail.

3.1 SPACEWIRE PORTS

The SpaceWire router has eight bi-directional SpaceWire links each conformant with the SpaceWire standard. Each SpaceWire link is controlled by an associated link register and routing control logic. Network level error recovery is performed when an error is detected on the SpaceWire link as defined in the SpaceWire standard. Packets received on SpaceWire links are routed by the routing control logic to the configuration port, other SpaceWire link ports or the external FIFO ports. Packets with invalid addresses are discarded by the SpaceWire router dependent on the packet address. The SpaceWire link status is recorded in the associated link register and error status is held by the router until cleared by a configuration command.

3.2 EXTERNAL PORTS

The SpaceWire router has two bi-directional parallel FIFO interfaces to an external host system. Each FIFO is written to or read from synchronously to the 30MHz system clock. An eight-bit data interface and an extra control bit for end of packet markers are provided by each external port FIFO. Packets received by the external port are routed by the routing control logic to the configuration port, SpaceWire link ports or the other external port dependent on the packet address. Packets with invalid addresses are discarded by the SpaceWire router.

3.3 CONFIGURATION PORT

The SpaceWire router has one configuration port which performs read and write operations to internal router registers. Packets are routed to the configuration port when a packet with a leading address byte of zero is received. The command packet format used in the SpaceWire Router is the Remote Memory Access Protocol (RMAP) [AD2]. A detailed description of the format of the RMAP command packets is defined in section 8.6. If an invalid command packet is received then the error is flagged to the associated status register and the packet is discarded. The internal router registers are described in section 0.

3.4 ROUTING TABLE

The SpaceWire router routing table is set by the router command packets to assign logical addresses to physical destination ports on the router. A group of destination ports can be set, in each routing table location, to enable group adaptive routing. In group adaptive routing a packet can be routed to its destination through one of a set of output ports dependent on which ports in the set are free to use. When a packet is received with a logical address the routing table is checked by the routing control logic and the packet is routed to the destination port when the port is ready.

Routing table locations are set to invalid at power on or at reset. An invalid routing address will cause the packet to be spilled by the control logic. The routing table logical addresses can also be set to support high priority and header deletion. High priority packets are routed before low priority packets and header deletion of logical addresses can be used to support regional logical addressing (see AD1).

3.5 ROUTING CONTROL LOGIC AND CROSSBAR

The routing control logic is responsible for arbitration of output ports, group adaptive routing and the crossbar switching. Arbitration is performed when two or more source ports are requesting to use the same destination port. A priority based arbitration scheme with two priority levels, high and low, is used where high priority packets are routed before low priority packets. Fair arbitration is performed on packets which have the same priority levels to ensure each packet gets equal access to the output port.

Group adaptive routing control selects one of a number of output ports for sending out the source packet.

3.6 TIME-CODE PROCESSING

An internal time-code register is used in the router to allow the router to be a time-code master or a time-code slave.

In master mode the time-code interface is used to provide a tick-in to the SpaceWire routing causing time-codes to be propagated through the network. Two modes of time master operation are supported, an automatic mode where a time-code is propagated on each external tick-in and a normal mode where the time-code is propagated dependent on the external time-in signal.

In time-code slave mode a valid received time-code, one plus the value of the router time-code register, causes a tick-out to be sent to the SpaceWire links and the external time-code interface. The time-code is propagated to all time-code ports except the port on which the time-code was received. If the time-code received is not one plus the value of the time-code register then the time-code register is updated but the tick-out is not performed. In this way circular network paths do not cause a constant stream of time-codes to be sent in a loop.

3.7 CONTROL/STATUS REGISTERS

The control and status registers in the SpaceWire router provide the means to control the operation of the router, set the router configuration and parameters or monitor the status of the device. The registers are accessed using RMAP [AD2] commands packets received by the configuration port.

4. DEVICE INTERFACE

The device pins used by each interface are described in this section. There is a table for each type of interface listing the signals in that interface. These tables have the following fields:

Pin No:	The device pin number
Signal:	The name of the signal
Dir:	The direction of the signal; in, out or in/out
Description:	An explanation of what the signal does.
Type:	The type of signal

The sections below define the pin out of the SpaceWire router. Its interfaces are split into several types, separated by headings for clarity:

- Global signals: clock and reset
- SpaceWire interface signals
- External port signals
- Time-code interface signals
- Configuration signals
- Reset configuration signals
- Power and Ground

The following signal types are used in the SpaceWire Router:

CMOS3V3	3.3 Volt CMOS logic
LVDS	Low Voltage Differential Signal
3V3	3.3 Volt power
GND	0 Volt ground

4.1 GLOBAL SIGNALS

The global system clock and reset signals are listed in Table 4-1.

Table 4-1 Global Signals

PinNo	Signal	Dir	Description	Type
	CLK	In	System clock. Provides the reference clock for all modules except the interface receivers.	CMOS3V3
	RST_N	In	Asynchronous system reset (active low)	CMOS3V3

	FEEDBDIV(2:0)	In	Set the output clock rate of the internal PLL as follows: **"000" → 100MHz** **"001" → 120MHz** **"010" → 140MHz** **"011" → 160MHz** **"100" → 180MHz** **"101" → 200MHz** **"110" → 200MHz** **"111" → 200MHz** See section 9.1.6 for setting the transmit rate.	CMOS3V3

4.2 SPACEWIRE SIGNALS

The SpaceWire interface signals are listed in Table 4-2. For further details about SpaceWire see the SpaceWire standard [AD1].

Table 4-2 Data and Strobe SpaceWire Signals

PinNo	Signal	Dir	Description	Type
	DOUT+(1) **DOUT-(1)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 1.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(2) **DOUT-(2)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 2.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(3) **DOUT-(3)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 3.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(4) **DOUT-(4)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 4.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(5) **DOUT-(5)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 5.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(6) **DOUT-(6)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 6.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(7) **DOUT-(7)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 7.	LVDS+ (P Side) LVDS - (N Side)
	DOUT+(8) **DOUT-(8)**	Out	Differential output pair, data part of Data-Strobe SpaceWire link 8.	LVDS+ (P Side) LVDS - (N Side)

	SOUT+(1) **SOUT-(1)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 1.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(2) **SOUT-(2)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 2.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(3) **SOUT-(3)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 3.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(4) **SOUT-(4)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 4.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(5) **SOUT-(5)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 5.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(6) **SOUT-(6)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 6.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(7) **SOUT-(7)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 7.	LVDS+ (P Side) LVDS - (N Side)
	SOUT+(8) **SOUT-(8)**	Out	Differential output pair, strobe part of Data-Strobe SpaceWire link 8.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(1) **DIN-(1)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 1.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(2) **DIN-(2)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 2.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(3) **DIN-(3)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 3.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(4) **DIN-(4)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 4.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(5) **DIN-(5)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 5.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(6) **DIN-(6)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 6.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(7) **DIN-(7)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 7.	LVDS+ (P Side) LVDS - (N Side)
	DIN+(8) **DIN-(8)**	In	Differential input pair, data part of Data-Strobe SpaceWire link 8.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(1) **SIN-(1)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 1.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(2) **SIN-(2)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 2.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(3) **SIN-(3)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 3.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(4)	In	Differential input pair, strobe part of Data-Strobe	LVDS+ (P Side)

	SIN-(4)		SpaceWire link 4.	LVDS - (N Side)
	SIN+(5) **SIN-(5)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 5.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(6) **SIN-(6)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 6.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(7) **SIN-(7)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 7.	LVDS+ (P Side) LVDS - (N Side)
	SIN+(8) **SIN-(8)**	In	Differential input pair, strobe part of Data-Strobe SpaceWire link 8.	LVDS+ (P Side) LVDS - (N Side)

4.3 EXTERNAL PORT DATA SIGNALS

The External port signals are listed in Table 4-3. The timing of these signals is shown in Figure 5-1 External port write timing specification and Figure 5-2 External port read timing specification.

Table 4-3 External Port Interface Signals

PinNo	Signal	Dir	Description	Type

	EXT_OUT_DATA0(8)
	EXT_OUT_DATA0(7)
	EXT_OUT_DATA0(6)
	EXT_OUT_DATA0(5)
	EXT_OUT_DATA0(4)
	EXT_OUT_DATA0(3)
	EXT_OUT_DATA0(2)
	EXT_OUT_DATA0(1)
	EXT_OUT_DATA0(0)
		Out	Output data from external port zero FIFO. Bit eight determines the type - data, EOP or EEP. The encodings are defined as:
			(8)(7......0) – **Bits**
			(0)(dddddddd) - Data byte
			(1)(XXXXXXX0) - EOP.
			(1)(XXXXXXX1) - EEP.
			Bit 7 is the most significant bit of the data byte.
				CMOS3V3

	EXT_OUT_EMPTY_N0	Out	FIFO ready signal for external output port zero. When high the FIFO has data. When low the FIFO is empty.	CMOS3V3
	EXT_OUT_READ_N0	In	Asserted (low) to read from the external output port zero FIFO.	CMOS3V3

	EXT_IN_DATA0(8)
	EXT_IN_DATA0(7)
	EXT_IN_DATA0(6)
		In	Input data to external port zero FIFO. Bit eight determines the type - data, eop or eep. The encodings are defined as:	CMOS3V3

	EXT_IN_DATA0(5)
	EXT_IN_DATA0(4)
	EXT_IN_DATA0(3)
	EXT_IN_DATA0(2)
	EXT_IN_DATA0(1)
	EXT_IN_DATA0(0)
			(8)(7......0) – **Bits**
			(0)(dddddddd) - Data byte
			(1)(XXXXXXX0) - EOP.
			(1)(XXXXXXX1) - EEP.
			Bit 7 is the most significant bit of the data byte.	

	EXT_IN_FULL_N0	Out	FIFO ready signal for external input port zero. When high there is space in the FIFO so it can be written to. When low the FIFO is full.	CMOS3V3
	EXT_IN_WRITE_N0	In	Asserted (low) to write to the external input port zero FIFO.	CMOS3V3

	EXT_OUT_DATA1(8)
	EXT_OUT_DATA1(7)
	EXT_OUT_DATA1(6)
	EXT_OUT_DATA1(5)
	EXT_OUT_DATA1(4)
	EXT_OUT_DATA1(3)
	EXT_OUT_DATA1(2)
	EXT_OUT_DATA1(1)
	EXT_OUT_DATA1(0)
		Out	Output data from external port one FIFO . Bit eight determines the type - data, eop or eep. The encodings are defined as:
			(8)(7......0) – **Bits**
			(0)(dddddddd) - Data byte
			(1)(XXXXXXX0) - EOP.
			(1)(XXXXXXX1) - EEP.
			Bit 7 is the most significant bit of the data byte.
				CMOS3V3

	EXT_OUT_EMPTY_N1	Out	FIFO ready signal for external output port one. When high the FIFO has data. When low the FIFO is empty.	CMOS3V3
	EXT_OUT_READ_N1	In	Asserted (low) to read from the external output port one FIFO.	CMOS3V3

	EXT_IN_DATA1(8)
	EXT_IN_DATA1(7)
	EXT_IN_DATA1(6)
	EXT_IN_DATA1(5)
	EXT_IN_DATA1(4)
	EXT_IN_DATA1(3)
	EXT_IN_DATA1(2)
	EXT_IN_DATA1(1)
		In	Input data to external port one FIFO. Bit eight determines the type - data, eop or eep. The encodings are defined as:
			(8)(7......0) – **Bits**
			(0)(dddddddd) - Data byte
			(1)(XXXXXXX0) - EOP.
			(1)(XXXXXXX1) - EEP.
			Bit 7 is the most significant bit of the data byte.
				CMOS3V3

	EXT_IN_DATA1(0)			
	EXT_IN_FULL_N1	Out	FIFO ready signal for external input port one. When high there is space in the FIFO so it can be written to. When low the FIFO is full.	CMOS3V3
	EXT_IN_WRITE_N1	In	Asserted (low) to write to the external input port one FIFO.	CMOS3V3

4.4 TIME-CODE SIGNALS

The time-code interface signals are listed in Table 4-4. The timing of this interface is shown in Figure 5-3 and Figure 5-4.

Table 4-4 Time-Code Signals

PinNo	Signal	Dir	Description	Type
	EXT_TICK_IN	In	The rising edge of the **EXT_TICK_IN** signal is used to indicate when a time-code is to be sent. On the rising edge of the **EXT_TICK_IN** signal the **SEL_EXT_TIME** signal is sampled to determine if the time-code value is to be provided by the internal time-counter or by the external time input **EXT_TIME_IN(7:0)**. The **SEL_EXT_TIME** and the **EXT_TIME_IN(7:0)** signals must be set up prior to the rising edge of **EXT_TICK_IN** and must be held static sometime afterwards.	CMOS3V3

	EXT_TIME_IN(7)
	EXT_TIME_IN(6)
	EXT_TIME_IN(5)
	EXT_TIME_IN(4)
	EXT_TIME_IN(3)
	EXT_TIME_IN(2)
	EXT_TIME_IN(1)
	EXT_TIME_IN(0)
		In	**EXT_TIME_IN(7:0)** provides the value of the time-code to be distributed by the router when an external time-code source is selected i.e. when **SEL_EXT_TIME** is high on the rising edge of **EXT_TICK_IN.**
			When **SEL_EXT_TIME** is high on the rising edge of **EXT_TICK_IN** the value of the time-code counter is used for bits 5:0 of the time-code and bits 7:6 of the **EXT_TIME_IN(7:0)** are used for the two control signals, bits 7:6 of the time-code.
				CMOS3V3

	SEL_EXT_TIME	In	If **SEL_EXT_TIME** is high on the rising edge of **EXT_TICK_IN** the value on **EXT_TIME_IN(7:0)** is loaded into the internal time-code register and	CMOS3V3

			propagated by the router. If **SEL_EXT_TIME** is low on the rising edge of **EXT_TICK_IN** the value to be sent in the time-code will be taken from the internal time-code counter in the router. The two control-bits (bits 7:6) of the time-code will come from bits 7:6 of the **EXT_TIME_IN(7:0)**input.	
	TIME_CTR_RST	In	This signal causes the internal time-code counter to be reset to zero. The timing parameters used for **EXT_TICK_IN** also apply to the time-code counter reset signal **(TIME_CTR_RST).**	CMOS3V3
	EXT_TICK_OUT	Out	The falling edge of **EXT_TICK_OUT** is used to indicated the reception of a time-code. The value of this time-code is place on the **EXT_TIME_OUT(7:0)**outputs and is valid on the rising edge of **EXT_TICK_OUT**.	CMOS3V3

	EXT_TIME_OUT(7)
	EXT_TIME_OUT(6)
	EXT_TIME_OUT(5)
	EXT_TIME_OUT(4)
	EXT_TIME_OUT(3)
	EXT_TIME_OUT(2)
	EXT_TIME_OUT(1)
	EXT_TIME_OUT(0)
		Out	Received time-code value which is valid when **EXT_TICK_OUT** is asserted.
			The value of a received time-code is output on the falling edge of **EXT_TICK_OUT**. The **EXT_TIME_OUT(7:0)** value is held until the next time-code is output.
				CMOS3V3

4.5 LINK ERROR INDICATION SIGNALS

The link error indication signals are listed in Table 4-5.

Table 4-5 Link error indication Signals

PinNo	Signal	Dir	Description	Signal Type

	STAT_MUX_ADDR(3)
	STAT_MUX_ADDR(2)
	STAT_MUX_ADDR(1)
		in	Select the error indication status signals to be output on **STAT_MUX_OUT** as defined in Table 5-1.	CMOS3V3

	STAT_MUX_ADDR(0)			

	STAT_MUX_OUT(7)
	STAT_MUX_OUT(6)
	STAT_MUX_OUT(5)
	STAT_MUX_OUT(4)
	STAT_MUX_OUT(3)
	STAT_MUX_OUT(2)
	STAT_MUX_OUT(1)
	STAT_MUX_OUT(0)
		inout	Multi function pin.
			Power on Configuration
			After reset the **STAT_MUX_OUT** pins are inputs which define the power on configuration status of the router. The pin mappings are listed in section 5.1.6.
			Normal Operation
			After the power on reset configuration of the router has been read from STAT_MUX_OUT the pins are driven as outputs by the router. The function of these output pins is defined by **STAT_MUX_ADDR(3:0)**.
				CMOS3V3

4.6 RESET CONFIGURATION SIGNALS

The Reset Configuration signals are listed in Table 4-6. These signals are input on **STAT_MUX_OUT** after reset to initialise the router. They are not used at any other time except immediately after reset. The Reset Configuration signals set relevant bits in the configuration registers (see section 0). Following reset the values of these signals are synchronously loaded into the router. The timing of the Reset Configuration signals is illustrated in Figure 5-7.

Table 4-6 Reset Configuration Signals			
Signal	**Dir**	**Description**	**Signal Type**
STAT_MUX_OUT(2:0) [maps to -> POR_TX_RATE(2:0)]	In	Sets the transmitter maximum data rate after reset. The data rate can subsequently be changed during normal operation using port configuration commands. The values are listed below. “111” – Full data rate after link start-up. “110” – 1/2 data rate after link start-up. “101” – 1/3 data rate after link start-up. “100” – 1/4 data rate after link start-up. “011” – 1/5 data rate after link start-up. “010” – 1/6 data rate after link start-up. “001” – 1/7 data rate after link start-up. “000” – 1/8 data rate after link start-up. Note: **POR_TX_RATE** affects all SpaceWire ports in the router. Note: The data rate is dependent on **FEEDBDIV** at reset	CMOS3V3
STAT_MUX_OUT(3) [maps to -> POR_ADDR_SELF_N	In	If asserted (low) after reset allows a router port to address itself and therefore cause an input packet to be returned through the same input port. This mode may only be suitable for debug and test operations. This signal is active low.	CMOS3V3
STAT_MUX_OUT(4) [maps to -> POR_TIMEOUT_EN_N]	In	Power on reset signal which determines if output port timeouts are enabled at start-up. When asserted (low) the port timeouts are enabled. When de-asserted (high) they are disabled. This signal is active low.	CMOS3V3
STAT_MUX_OUT(5) [maps to -> POR_SEL_TIMEOUT0_N]	In	Power on reset value which determines the initial timeout value. The following values determine which timeout is selected at power up. ‘1’ => Timeout period is ~ 100 us. ‘0’ => Timeout period is ~ 1.3 s.	CMOS3V3

		Timeout Period is: '1' => 200x(2^2)x(10 MHz clk period) '0' => 200x(2^16)x(10 MHz clk period)	
STAT_MUX_OUT(6) [maps to -> POR_START_ON_REQ_N]	In	Power on reset signal which determines if the output ports automatically start up when they are the destination address of a packet. When asserted (low) the output port will automatically start on request. This signal is active low.	CMOS3V3
STAT_MUX_OUT(7) [maps to -> POR_DSBLE_ON_SILENCE_N]	In	Power on reset signal which determines if the output ports are disabled when no activity is detected on an output port for the current timeout period. When asserted (low) an output port is disabled when it has not sent any information for longer than the current timeout period. This signal is active low.	CMOS3V3

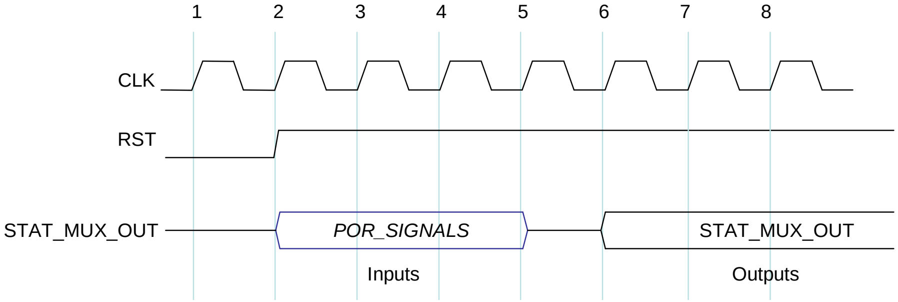


Figure 4-1 Configuration interface timing specification

The POR configuration signals (POR_SIGNALS) listed in Table 4-6 are loaded into the appropriate internal configuration registers of the router after **RST** is de-asserted. To make sure that the POR configuration signal values are loaded properly they should be held stable for at least three **CLK** cycles following **RST** being de-asserted. The status output **STAT_MUX_OUT** is driven on the fourth **CLK** cycle after **RST** is de-asserted.

Note: The recommended method for setting the POR signals is to use external pull up/down resistors in which case the timing of the POR signals is not critical.

4.7 POWER AND GROUND SIGNALS

The Power and Ground connections are listed in Table 4-6.

Table 4-7 Reset Configuration Signals

Signal	Dir	Description	Signal Type
Power	-	3.3 V power for the device	3V3
Ground	-	Ground connection for the device	GND

5. INTERFACE OPERATIONS

5.1 EXTERNAL PORT INTERFACE OPERATION

In this section the external port interface operation is defined.

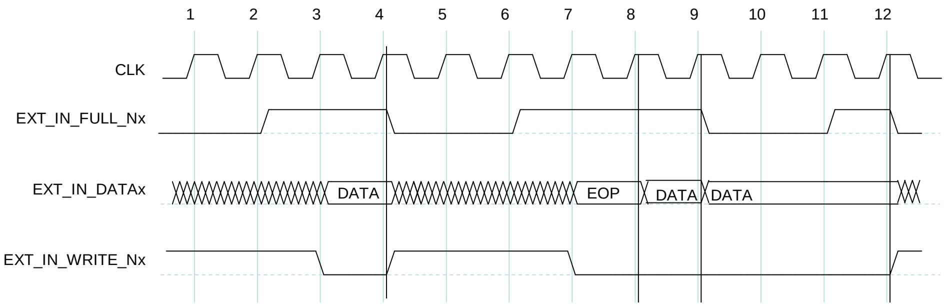


Figure 5-1 External port write timing specification

The operation of the External port during write operations starts with the **EXT_IN_FULL_N** signals being de-asserted by the router (at clock cycle 2 in Figure 5-1) to indicate to the external system that the router has room for more data and is ready to receive it through the External port. The External system then puts data onto the **EXT_IN_DATA** data lines and asserts **EXT_IN_WRITE_N** to transfer data into the External port on the next rising edge of **SYSCLK**. As long as there is room for new data (**EXT_IN_FULL_N** is inactive) the writer access is performed as long as **EXT_IN_WRITE_N** is active. If no room is available the write access is ignored (cycle 9 and 10 in Figure 5-1) and will be performed when room has become available if **EXT_IN_WRITE_N** is still active. Therefore the data (**EXT_IN_DATA**) must be valid at that time.

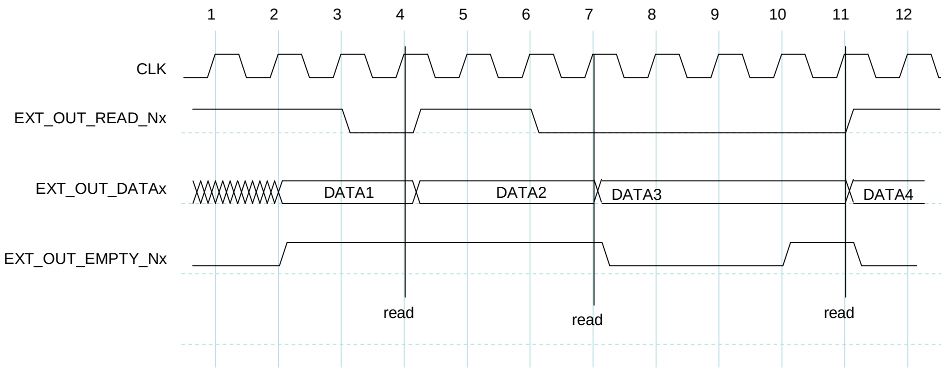


Figure 5-2 External port read timing specification

Reading of the External port is illustrated in Figure 5-2. When data is available in the External port FIFO then it is placed on the **EXT_OUT_DATA** bus and the **EXT_OUT_EMPTY_N** signal is asserted to signal to the external system that data is available. This is done synchronously to the **SYSCLK** signal (e.g. clock cycle 2 in Figure 5-2). When it is ready the external system asserts the

EXT_OUT_READ_N signal synchronously with the **SYSCLK** signal (e.g. clock cycle 3) and the data is then read out of the external port on the next rising edge of the **SYSCLK** (e.g. start of clock cycle 4). If there is no more data available in the FIFO then the **EXT_OUT_EMPTY_N** is de-asserted once the data has been read. If the FIFO contains more data to transfer then the **EXT_OUT_EMPTY_N** remains asserted, the new data is placed on the **EXT_OUT_DATA** bus and the external system can read it as soon as it is ready. The read access is ignored if there is no data available (**EXT_OUT_EMPTY_N** is active).

5.2 TIME-CODE INTERFACE OPERATION

In this section the time-code interface operation is defined.

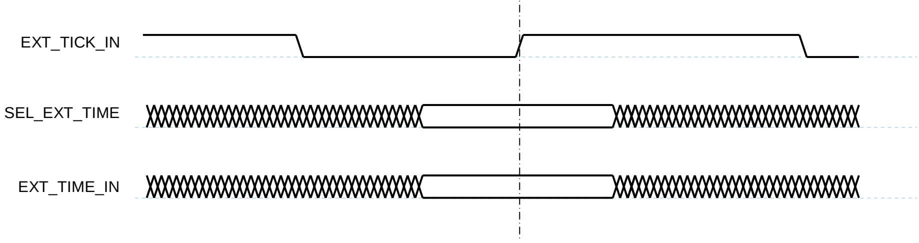


Figure 5-3 Time-Code Input Interface

Time-codes can be generated by the router on request of the external system to which it is attached. A time-code is generated whenever the router detects a rising edge on the. The value of the time-code to be transmitted is either taken from the inputs or from the time-code counter inside the router. The time-code source used depends on the value of the **SEL_EXT_TIME** signal when **EXT_TICK_IN** signal has a rising edge. If **SEL_EXT_TIME** is 1 then the **EXT_TIME_IN(7:0)** inputs are used to provide the contents of the time-code. If **SEL_EXT_TIME** is 0 then the internal time-code counter provides the least-significant 6-bits of the time-code and the **EXT_TIME_IN(7:6)** inputs provide the most-significant 2-bits. When using the **EXT_TIME_IN(7:0)** inputs to provide the complete time-code, the time-code is only broadcast if it is a valid time-code i.e. is one more than the internal time register of the router (see SpaceWire standard [AD1]). Note that only one router or node in a SpaceWire network should operate as a time master generating time codes (see SpaceWire standard [AD1]).

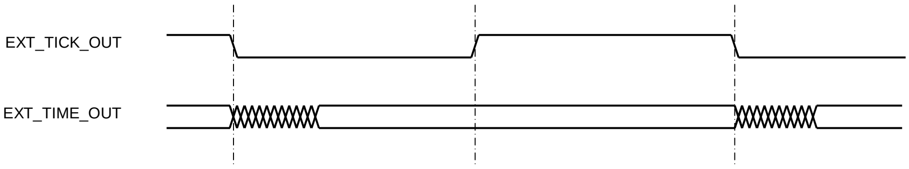


Figure 5-4 Time-Code Output Interface

When a valid time-code is received by the router the value of this time-code (flags plus time value) will be placed on the **EXT_TIME_OUT** outputs and the **EXT_TICK_OUT** signal will be set to zero. The **EXT_TICK_OUT** signal is set to one a short time later, once the **EXT_TIME_OUT** outputs have stabilised, to indicate that these outputs are valid. They then remain valid until the next time-code is received and the **EXT_TICK_OUT** signal will be set to zero.

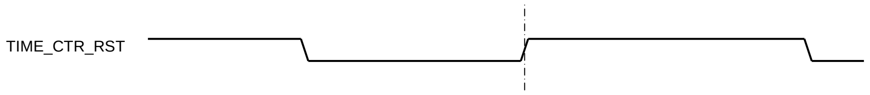


Figure 5-5 Time-code reset interface

When a rising edge is detected on **TIME_CTR_RST** then the time-code register is reset to zero.

5.3 ERROR/STATUS ERROR INDICATION INTERFACE OPERATION

The **STAT_MUX_ADDR** signal determines the output status on **STAT_MUX_OUT** as shown in Figure 5-6 and in Table 5-1.

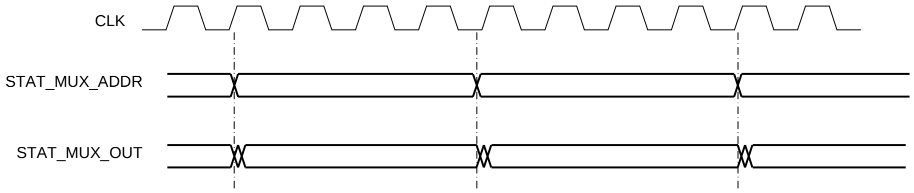


Figure 5-6 Status Multiplexer output interface

When **STAT_MUX_ADDR** is stable **STAT_MUX_OUT** is output from after each clock edge.

Table 5-1 Multiplexed Status Pins Bit Assignment				
Mux Address	Status Register	Status Signal	Status Register Bits	Status Output Bits
0	Configuration Port	Packet return address error Output port timeout error Checksum error Packet too short error Packet too long error Packet EEP termination Protocol byte error Invalid address/data error	1 2 3 4 5 6 7 8	0 1 2 3 4 5 6 7
1 - 8	SpaceWire Ports 1 - 8 respectively	Packet Address Error Output Port Timeout Disconnect Error Parity Error Escape Error State A State B State C	1 2 3 4 5 8 9 10	0 1 2 3 4 5 6 7
9 - 10	External Ports 0 - 1 respectively	Error Active Packet Address Error Output Port Timeout Input Buffer Empty Input Buffer Full Output Buffer Empty Output Buffer Full	0 1 2 3 4 5 6	0 1 2 3 4 5 6
11	Network Discovery Router Identity	Return port Least-significant 4-bits	7:4 3:0	7:4 3:0
12	Router Control	Timeout Enable	0	0

		Timeout Selection Enable Disable-on-Silence Enable Start-on-Request Enable Self-Addressing	3:1 4 5 6	3:1 4 5 6
13	Error Active	Configuration Port Error SpaceWire Ports 1-5 Error External Ports 1,2 Error	0 5 :1 10 :9	0 5 :1 7 :6
14	Time-code	Time-code	7:0	7:0
15	General Purpose	Least Significant 8-bits	7:0	7:0

5.4 RESET CONFIGURATION INTERFACE OPERATION

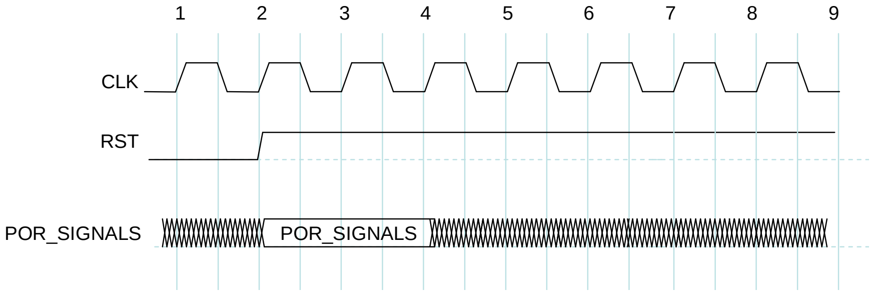


Figure 5-7 Reset configuration interface timing specification

The POR configuration signals (POR_SIGNALS) listed above are loaded into the appropriate internal configuration registers of the router on the first rising edge of the system clock, **SYSCLK**, after **RST** is de-asserted. To make sure that the POR configuration signal values are loaded properly they should be held stable for at least two **CLK** cycles following **RST** being de-asserted.

6. SWITCHING CHARACTERISTICS

6.1 CLOCK AND RESET TIMING PARAMETERS

The global clock and asynchronous reset timing parameters are listed below.

Table 6-1 Clock and reset timing parameters

Description	Symbol	Value	Units
Clock period minimum value	T_{CL}		ns, min
Clock period maximum value	T_{CH}		ns, max
Clock input jitter	$T_{CJITTER}$	+/- 300	ps, max
Reset minimum period	T_{ARST}		ns, min

6.2 SERIAL SIGNALS TIMING PARAMETERS

The data strobe minimum consecutive edge separation timing parameter is defined as shown in the figure below.

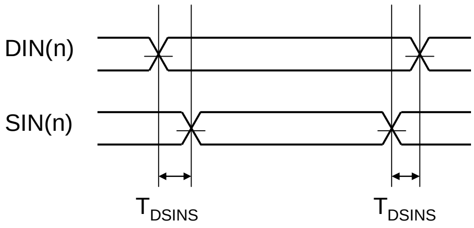


Figure 6-1 DS minimum consecutive edge separation

The serial signal timing parameters are defined in the table below.

Table 6-2 Serial signal timing parameters

Description	Symbol	Value	Units
DS maximum input bit rate	T_{DS}	200	Mbits/s, max
DS minimum consecutive edge separation	T_{DSINS}	1.5	ns, min
Data Strobe output skew	$T_{DSOSKEW}$	0.5	ns, max

6.3 EXTERNAL PORT TIMING PARAMETERS

The external port input timing parameters can be viewed below

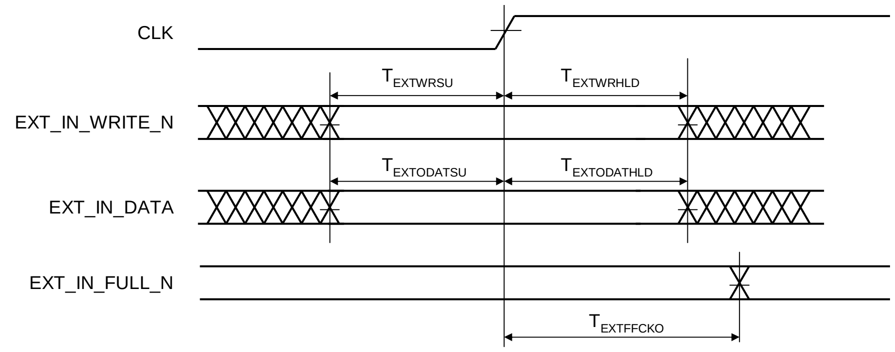


Figure 6-2 External port input FIFO timing parameters

The external port input timing parameters can be viewed below

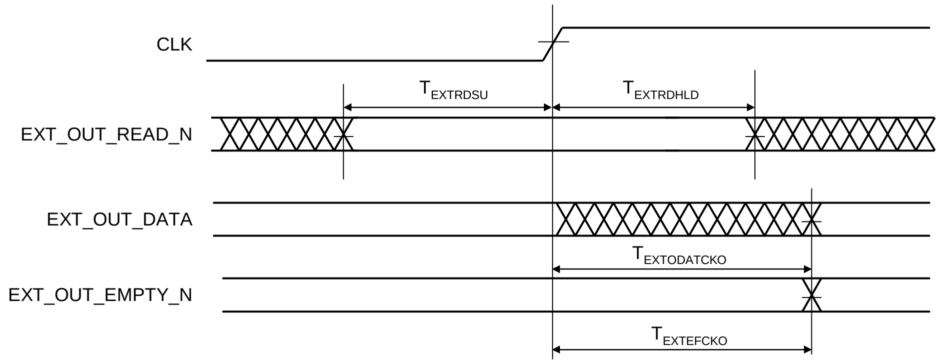


Figure 6-3 External port output FIFO timing parameters

Table 6-3 External port timing parameters

Description	Symbol	Value	Units
Write enable setup time to CLK rising edge	$T_{EXTWRSU}$	3.6	ns, min
Write enable hold time after CLK rising edge	$T_{EXTWRHLD}$	0	ns, min
Write data setup time to CLK rising edge	$T_{EXTIDATSU}$	3	ns, min
Write data hold time after CLK rising edge	$T_{EXTIDATHLD}$	1.5	ns, min
CLK rising edge to full flag output	$T_{EXTFFCKO}$	11	ns, max
Read enable setup time to CLK rising edge	$T_{EXTRDSU}$	3	ns, min
Read enable hold time after CLK rising edge	$T_{EXTRDHLD}$	0.8	ns, min
CLK rising edge to read data output	$T_{EXTODATCKO}$	8.7	ns, max
CLK rising edge to empty flag output	$T_{EXTEFCKO}$	11.2	ns, max

6.4 TIME-CODE INTERFACE TIMING PARAMETERS

The following diagrams define the timing parameters for the time-code input and output

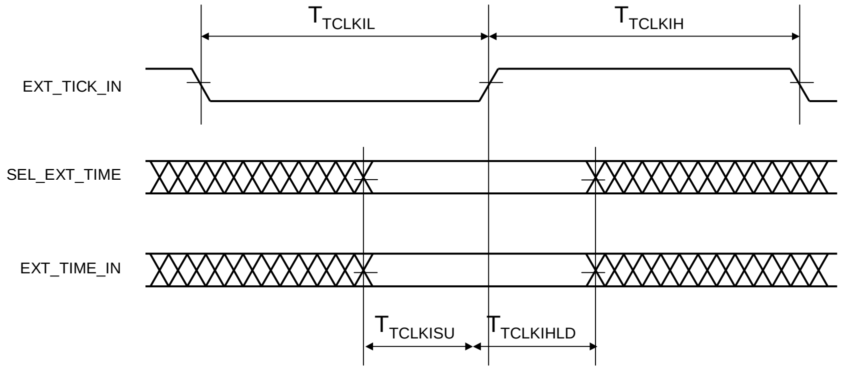


Figure 6-4 Time-Code Input Interface

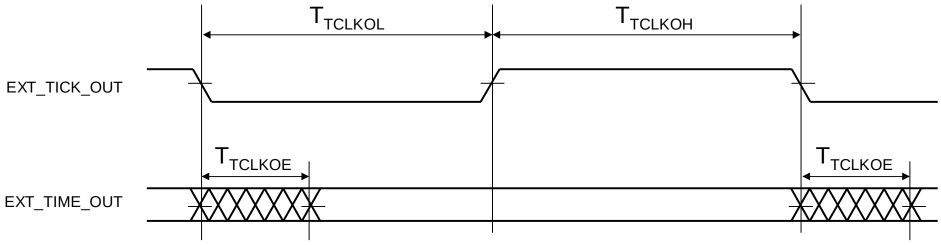


Figure 6-5 Time-Code Output Interface

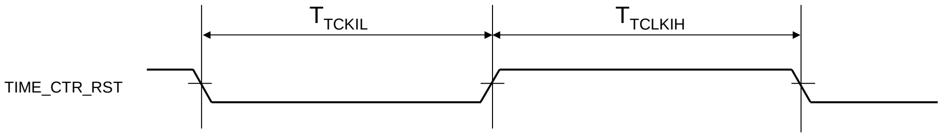


Figure 6-6 Time-code TIME_CTR_RST interface

The Time-code timing parameters are shown below.

Table 6-4 Time-code interface timing parameters

Description	Symbol	Value	Units
Tick-in and time reset low time	$\mathbf{T_{TCLKIL}}$	500	ns, min
Tick-in and time reset high time	$\mathbf{T_{TCLKIH}}$	500	ns, min
Select external time and Time-code in set-up time	$\mathbf{T_{TCLKISU}}$	2.7	ns, max
Select external time and Time-code in hold time	$\mathbf{T_{TCLKIHLD}}$	0.1	ns, max
Tick-out low time	$\mathbf{T_{TCLKOL}}$	500	ns, min
Tick-out high time	$\mathbf{T_{TCLKOH}}$	500	ns, min

Time-code output valid delay time	T_{TCLKOE}	1	ns, max

6.5 ERROR/STATUS INTERFACE TIMING PARAMETERS

The timing parameters for the status multiplexer port are show below.

Table 6-5 Status Multiplexer timing parameters

Description	Symbol	Value	Units
Status address change to status output change	T_{STMUX}	20.6	ns, max

6.6 LATENCY AND JITTER

The timing parameters for the data and time-code latency and the time-code jitter are derived from the receive clock, transmit clock and system clock period. The worst case number of clock cycles required is used in each equation.

In the SpaceWire router the system clock is a known frequency and the transmitter and receiver frequency are derived from the input and output bit rates. The clock frequencies are defined as follows.

Note: All figures are worst case. Due to the uncertainty of synchronisation between clock domains the measured time may be less than indicated.

In the following sections the clock periods are defined and the latency and jitter timing parameter definitions are detailed.

6.6.1 Clock Periods

System Clock Period

$T_{SYSPERIOD}$ = 33.333 ns (Clock Frequency = 30 MHz)

Transmit Clock Period

$T_{TXPERIOD}$ = Transmit bit rate period * 2 (Where Transmit bit rate period is the output bit rate selected by the user configuration)

Receive Clock Period

$T_{RXPERIOD}$ = Receive bit rate period * 2 (Where Receive bit rate period is the period of the input bit rate)

6.6.2 Switching Latency

Switching latency is the time it takes the router to connect a waiting input port to an output port that has just finished sending a packet. It includes any time for group adaptive routing and arbitration of two or more input ports competing for the same output port.

Switching latency for the router is defined as follows

$$T_{SWITCH} = 4 \times T_{SYSPERIOD}$$

6.6.3 Router Latency

Router latency is the time taken for a character in a packet to pass through the router assuming that the packet has already been switched to an output port and that there is no blocking of the output port. Router latency for the SpaceWire router is defined for port to port data transfer operations as follows:

SpaceWire port to SpaceWire port

Last bit of data into receiver to last bit of data out of transmitter (Worst case where transmitter is sending a time-code and FCT character before data)

$$T_{SSDATA} = (5 \times T_{RXPERIOD}) + (8 \times T_{SYSPERIOD}) + (23 \times T_{TXPERIOD})$$

SpaceWire port to External port

Last bit of data into receiver to external port not empty flag

$$T_{SEDATA} = (5 \times T_{RXPERIOD}) + (8 \times T_{SYSPERIOD})$$

External port to SpaceWire port

External port write enable to last bit of data out of transmitter (Worst case where transmitter is sending a time-code and FCT character before data)

$$T_{ESDATA} = (4 \times T_{SYSPERIOD}) + (23 \times T_{TXPERIOD})$$

External port to External port

External port write enable to external port not empty flag.

$$T_{EEDATA} = (5 \times T_{SYSPERIOD})$$

6.6.4 Time-code Latency

The maximum time taken to propagate a time code through a routing switch

SpaceWire port to SpaceWire port

Last bit of time-code into receiver to last bit of time-code out of transmitter (worst case where transmitter has started sending a before time-code data character)

$$T_{SSTC} = (5 \times T_{RXPERIOD}) + (6 \times T_{SYSPERIOD}) + (16 \times T_{TXPERIOD})$$

SpaceWire port to External port

Last bit of time-code into receiver to external port EXT_TICK_OUT rising edge

$$T_{SETC} = (5 \times T_{RXPERIOD}) + (8 \times T_{SYSPERIOD})$$

External port to SpaceWire port

EXT_TICK_IN rising edge to last bit of time-code out of transmitter (worst case where transmitter has started sending a before time-code data character)

$$T_{ESTC} = (6 \times T_{SYSPERIOD}) + (16 \times T_{TXPERIOD})$$

6.6.5 Time-code Jitter

The variation in time to propagate a time-code through a routing switch.

Time-code jitter occurs in the synchronisation handshaking circuits and the transmitter where the maximum delay time the time-code has to wait to be transmitted is one data character. The jitter is measured as

$$T_{TCJIT} = \left(2 \times T_{SYSPERIOD}\right) + \left(5 \times T_{TXPERIOD}\right)$$

6.6.6 200M bits/s Input and Output Bit Rate Example

The following table defines the latency and jitter measurements when the transmit bit rate and receive bit rate are 200M bits/s.

Table 6-6 SpaceWire Router Latency and Jitter Measurements (Bit rate = 200Mbits/s

Description	Symbol	Value	Units
Switching Latency	T_{SWITCH}	133.3	ns, max
Router Latency – SpaceWire to SpaceWire port	T_{SSDATA}	546.6	ns, max
Router Latency – SpaceWire to External port	T_{SEDATA}	316.6	ns, max
Router Latency – External to SpaceWire port	T_{ESDATA}	363.3	ns, max
Router Latency – External to External port	T_{EEDATA}	166.6	ns, max
Time-code Latency – SpaceWire to SpaceWire port	T_{SSTC}	409.3	ns, max
Time-code Latency – SpaceWire to External port	T_{SETC}	316.6	ns, max
Time-code Latency – External to SpaceWire port	T_{ESTC}	359.9	ns, max
Time-code Jitter	T_{TCJIT}	116.6	ns, max

[1] Note all figures are worst case

7. ELECTRICAL CHARACTERISTICS

The electrical characteristics for the SpaceWire router are defined in this section

7.1 ABSOLUTE MAXIMUM RATINGS

TBD

7.2 DC CHARACTERISTICS

TBD

7.3 OPERATING TEMPERATURE RANGE

TBD

8. SPACEWIRE ROUTER PACKET TYPES

This section defines how the routing control logic interprets packets.

8.1 PACKET ADDRESSES

The routing control logic interprets the first byte of each received packet as the packet address. The packet address defines the physical ports through which the routing control logic will use to route the packet towards its destination.

Packets which have a path address (0-31) as the first byte are always routed to the corresponding physical port number on the router. Packets which have a logical address (32-255) are routed to physical ports dependent on the contents of the routing table. The internal SpaceWire router routing table can be set up to assign logical addresses to the physical ports, except the configuration port (port 0) which can only be accessed by path addressing

The physical port addresses for the SpaceWire router and the expected packet type is defined in the table below. The packet types can be viewed in section 5.

Table 8-1 Packet Address Mapping

Packet Address	Expected Packet Type	Physical Port type
0	Command packet	Configuration port
1	Any type	SpaceWire link port 1
2	Any type	SpaceWire link port 2
3	Any type	SpaceWire link port 3
4	Any type	SpaceWire link port 4
5	Any type	SpaceWire link port 5
6	Any type	SpaceWire link port 6
7	Any type	SpaceWire link port 7
8	Any type	SpaceWire link port 8
9	Any type	External FIFO port 1
10	Any type	External FIFO port 2
11-31	N/A	Invalid addresses
32-255	Any type	Logical addresses

Note that logical address 255 is reserved in the SpaceWire standard [AD1].

8.2 PACKET PRIORITY

Each packet which is input to the router has an associated priority level, either as a result of the packet address or the internal routing table. Two priority levels HIGH and LOW are supported.

The table below defines the priority levels for packet addresses

Table 8-2 Packet Priority Mapping

Packet Address	Packet Priority	Physical Port type
0	HIGH	Configuration port
1	HIGH	SpaceWire link port 1
2	HIGH	SpaceWire link port 2
3	HIGH	SpaceWire link port 3
4	HIGH	SpaceWire link port 4
5	HIGH	SpaceWire link port 5
6	HIGH	SpaceWire link port 6
7	HIGH	SpaceWire link port 7
8	HIGH	SpaceWire link port 8
9	HIGH	External FIFO port 1
10	HIGH	External FIFO port 2
11-31	N/A	Invalid addresses
32-255	Dependent on routing table - Default LOW - May be configured HIGH (see section 10.3)	Logical addresses

8.3 PACKET HEADER DELETION

Header deletion is performed on packets dependent on the packet address. Packets which have path addresses or logical addresses which have the header deletion bit set in the routing table have the header address byte removed before the packet is routed to the destination.

The table below defines the header deletion settings for each address.

Table 8-3 Packet Header Deletion Mapping

Packet Address	Header Deletion	Physical Port type
0	Enabled	Configuration port
1	Enabled	SpaceWire link port 1
2	Enabled	SpaceWire link port 2
3	Enabled	SpaceWire link port 3
4	Enabled	SpaceWire link port 4
5	Enabled	SpaceWire link port 5
6	Enabled	SpaceWire link port 6
7	Enabled	SpaceWire link port 7
8	Enabled	SpaceWire link port 8
9	Enabled	External FIFO port 1
10	Enabled	External FIFO port 2
11-31	N/A	Invalid addresses
32-255	Dependent on routing table (default not enabled)	Logical addresses

Note that header deletion is always enabled for path addresses and cannot be changed by configuration. Header deletion for logical addresses can be enabled or disabled via a configuration register (see section 10.3).

8.4 INVALID ADDRESSES

Packets which have invalid addresses are discarded by the routing control logic. Path addresses which are in the range 11-31, logical addresses which are set as invalid in the routing table and empty packets (packets with no address or cargo) input to the external port are flagged as invalid packet addresses.

A packet address error is also generated when a packet address causes the packet to be routed back through the port on which the packet was received, i.e. a loop-back, and the router control register bit address self is not enabled.

When an invalid address packet is received by the router then the routing control logic flags the error to the corresponding port status register, spills the packet address, data and end of packet marker and waits for the next packet.

8.5 DATA PACKETS

Packets which have addresses in the range 1 to 255 are routed to the SpaceWire ports and the external ports dependent on the packet address. Data packets have an address header byte a cargo field and an end of packet marker. The normal packet structure is show below.

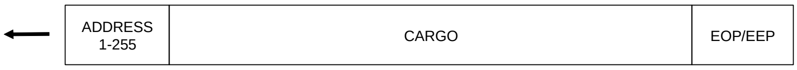


Figure 8-1 Normal router data packets

8.6 COMMAND PACKETS

Command packets are routed to the internal configuration port when the packet address is zero. Command packets perform write and read operations to registers in the SpaceWire router. Command packets accepted by the SpaceWire router are in the form shown in Figure 8-2.

Configuration read packets are in the form:

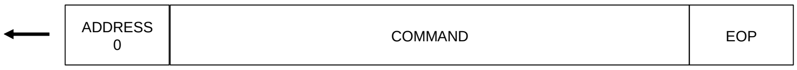


Figure 8-2 Command Packet Format

The SpaceWire router supports the Remote Memory Access Protocol (RMAP) [AD2] for configuration of the internal router control registers and monitoring of the router status.

The following sections define the RMAP commands which are supported and the format of the RMAP commands used by the SpaceWire Router.

8.6.1 Supported Commands

The RMAP Command set is listed in Table 8-4 and the supported RMAP commands are defined. The commands which are not used are depicted with a grey background.

Table 8-4 Supported RMAP Command Codes

RMAP Command Code	Description	Supported in SpaceWire Router
“0000”	*Not used*	-
“0001”	*Not used*	-
“0010”	Read single address	Yes
“0011”	Read incrementing address	Yes
“0100”	*Not used*	-
“0101”	*Not used*	-
“0110”	*Not used*	-
“0111”	Read-modify-write incrementing address	Yes
“1000”	Write single address, no verify, no acknowledge	No
“1001”	Write incrementing address, no verify, no acknowledge.	No
“1010”	Write single address, no verify, send acknowledge	No
“1011”	Write incrementing address, no verify, send acknowledge.	No
“1100”	Write single address, verify data, no acknowledge	No
“1101”	Write incrementing address, verify data, no acknowledge.	No
“1110”	Write single address, verify data, send acknowledge	Yes
“1111”	Write incrementing address, verify data, send acknowledge.	No

8.6.2 Read Command

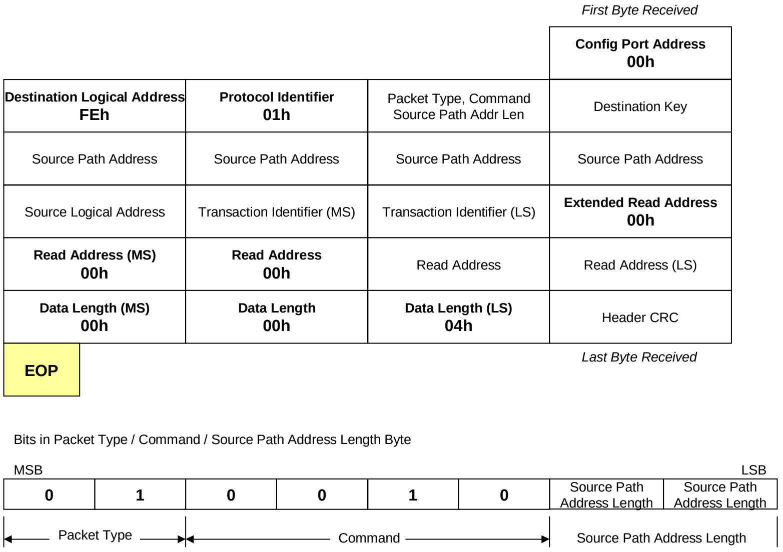
The read single address characteristics of the SpaceWire router are defined in Table 8-5.

Table 8-5 Read Single Address Characteristics

Action	Supported/ Not Supported	Maximum number of bytes	Non-aligned access accepted
8-bit read	NS	-	-
16-bit read	NS	-	-
32-bit read	S	4	No
64-bit read	NS	-	-
Word or byte address	32-bit aligned		
Accepted Logical Addresses	0xFE		
Accepted destination keys	0x20 at power on.		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 0109		
Address Incrementation	No		

The RMAP read single address command is supported in the SpaceWire router. The single address command is used to read a single 32 bit register location from the router registers.

In Figure 8-3 the format of a read single address command is illustrated. The first byte received by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional.



First Byte Received

			Config Port Address 00h
Destination Logical Address FEh	**Protocol Identifier 01h**	Packet Type, Command Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	**Extended Read Address 00h**
Read Address (MS) 00h	**Read Address 00h**	Read Address	Read Address (LS)
Data Length (MS) 00h	**Data Length 00h**	**Data Length (LS) 04h**	Header CRC
EOP			

Last Byte Received

Bits in Packet Type / Command / Source Path Address Length Byte

MSB							LSB
0	**1**	**0**	**0**	**1**	**0**	Source Path Address Length	Source Path Address Length
Packet Type		Command				Source Path Address Length	

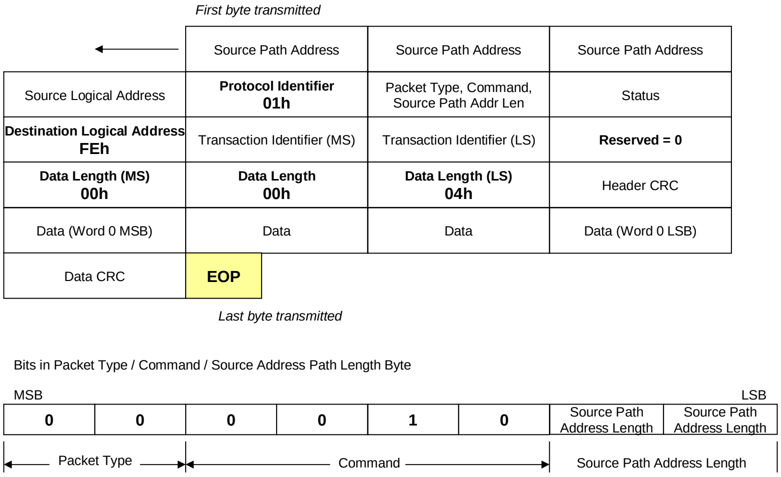
Figure 8-3 Read Single Address Command Format

Table 8-6 Read Single Address Command Packet Fields

Field	Description	Bytes
Config Port Address	The configuration port address field routes the packet to the configuration port of the router. The configuration port address (00h) is always present when configuring the SpaceWire Router.	1
Destination Logical Address	The destination logical address field is not used in the SpaceWire Router. The SpaceWire router accepts packets which have the default destination logical address of 254h (FEh).	1
Protocol Identifier	The RMAP protocol identifier is 01h.	1
Command Byte	The command byte indicates a read single address packet. The Source path address length fields are set to the number of source path addresses required as defined in section 8.6.9.	1
Destination Key	The destination key identifier must match the contents of the destination key register, see section 10.5.10. The default (power-on) destination key is 20h.	1
Source Path Address	The source path address field is used to add source path addresses to the head of the reply packet. The expected number of source path addresses is specified in the command byte. See section 8.6.9 for source path address decoding.	0,4,8,12
Source Logical Address	The source logical address should be set to the logical address of the node which sent the command.	1
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number.	2
Extended Read Address	The extended read address is not used in the SpaceWire router and shall always be set to zero.	1
Read Address	The read address identifies the register address to read from. The valid read addresses are defined in section 0.	4
Data Length	The data length of a read single address command shall be set to 4 to read one 32 bit register location.	3

Header CRC	The header CRC is the eight bit CRC code used to detect errors in the command packet. The CRC code is checked before the command is executed	1

In Figure 8-4 the format of the reply to a read single address command is illustrated. The first byte sent by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional. Note that the reply is always sent out of the same port as the command was received on. The Source Path Address should not include the output port of the router being commanded as the reply will be automatically sent out of the same port that the command arrived on. See section 8.6.8.


First byte transmitted
Last byte transmitted
Bits in Packet Type / Command / Source Address Path Length Byte
MSB
LSB
Packet Type
Command
Source Path Address Length

	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	**Protocol Identifier** **01h**	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address **FEh**	Transaction Identifier (MS)	Transaction Identifier (LS)	**Reserved = 0**
Data Length (MS) **00h**	**Data Length** **00h**	**Data Length (LS)** **04h**	Header CRC
Data (Word 0 MSB)	Data	Data	Data (Word 0 LSB)
Data CRC	**EOP**		

0	**0**	**0**	**0**	**1**	**0**	Source Path Address Length	Source Path Address Length

Figure 8-4 Read Single Address Reply Packet Format

Table 8-7 Read Single Address Reply Packet Fields		
Field	**Description**	**Bytes**
Source Path Address	Optional source path addresses specified in the command packet. If no source path addresses are specified then the first byte will be the source logical address.	>=0
Source Logical Address	The source logical address specified in the command packet. If source path addresses are not used then the source logical address is the address of the return packet.	1
Protocol Identifier	The RMAP protocol identifier value 01h.	1

Command Byte	Read single address reply command byte. The packet type bits in the command byte indicate this packet is a response packet.	1
Status	The command status is returned in this field. The command status can be command successful or an RMAP error code as defined in section 8.6.6.	1
Destination logical address	The destination logical address is set to the default value FEh as the SpaceWire router does not have a logical address.	1
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number. The transaction identifier in the reply packet is copied from the command packet and returned in this field, so that the command and the corresponding reply have the same transaction identifier value.	2
Data Length	The data length field is set to 4 bytes as this is a single read command.	3
Header CRC	The header CRC used to detect errors in the header part of the command packet. See section 8.6.7 for CRC generation.	1
Data	The data read from the registers in the device.	4
Data CRC	The data CRC used to detect errors in the data part of the reply packet. See section 8.6.7 for CRC generation.	1

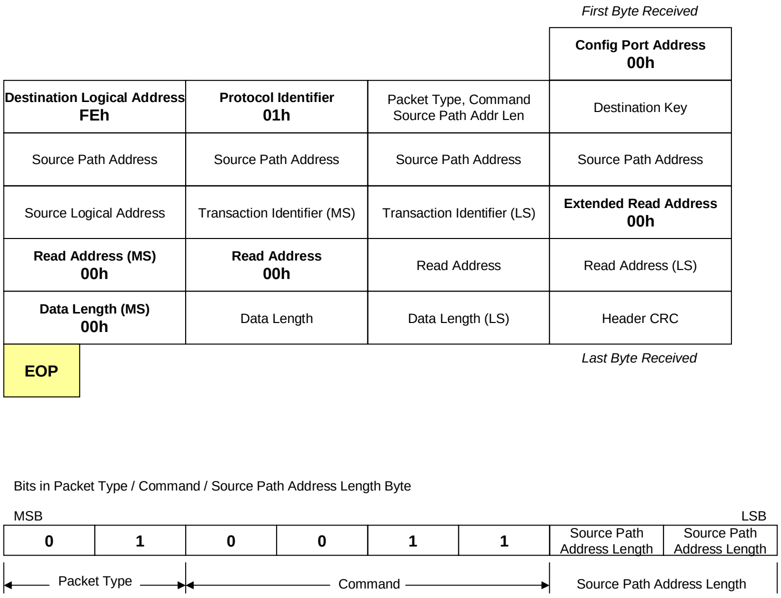
8.6.3 Read Incrementing Command

The read incrementing address characteristics of the SpaceWire router are defined in Table 8-8.

Table 8-8 Read Incrementing Address Characteristics			
Action	Supported/ Not Supported	Maximum number of bytes	Non-aligned access accepted
8-bit read	NS	-	-
16-bit read	NS	-	-
32-bit read	S	1064	No
64-bit read	NS	-	-
Word or byte address	32-bit aligned		
Accepted Logical Addresses	0xFE		
Accepted destination keys	0x20 at power on		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 0109		
Incrementing address	Incrementing address only		

The RMAP read incrementing address command is supported in the SpaceWire router. The read incrementing address is used to read a continuous block of registers from the SpaceWire router, e.g. the complete group adaptive routing table can be read in one command or all the status registers for the SpaceWire links can be read in one command.

In Figure 8-5 the first byte received by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional.



First Byte Received

			Config Port Address **00h**
Destination Logical Address **FEh**	**Protocol Identifier** **01h**	Packet Type, Command Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	**Extended Read Address** **00h**
Read Address (MS) **00h**	**Read Address** **00h**	Read Address	Read Address (LS)
Data Length (MS) **00h**	Data Length	Data Length (LS)	Header CRC
EOP			

Last Byte Received

Bits in Packet Type / Command / Source Path Address Length Byte

MSB							LSB
0	**1**	**0**	**0**	**1**	**1**	Source Path Address Length	Source Path Address Length
Packet Type		Command				Source Path Address Length	

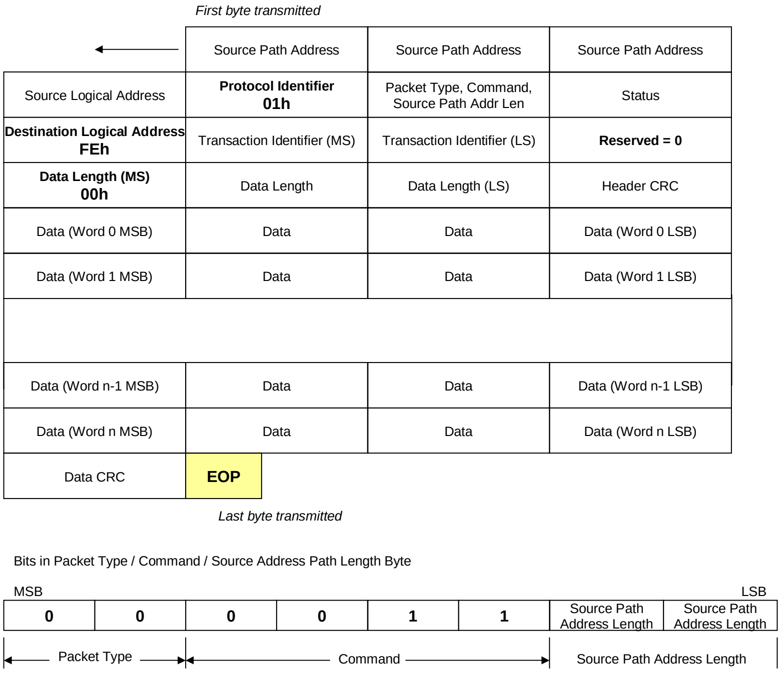
Figure 8-5 Read Incrementing Address Command Format

Table 8-9 Read Incrementing Address Command Packet Fields

Field	**Description**	**Bytes**
Config Port Address	The configuration port address field routes the packet to the configuration port of the router. The configuration port address is always present when configuring the SpaceWire Router.	1
Destination Logical Address	The destination logical address is not used in the SpaceWire Router. The SpaceWire router accepts packets which have the default destination logical address of 254h (FEh).	1
Protocol Identifier	The RMAP protocol identifier is 01h.	1
Command Byte	The command byte indicates a read incrementing packet. The Source path address length fields are set to the number of source path addresses required as defined in section 8.6.9.	1

Destination Key	The destination key identifier must match the contents of the destination key register, see section 10.5.10.	1
Source Path Address	The source path address field is used to add source path addresses to the head of the reply packet. The expected number of source path addresses is specified in the command byte. See section 8.6.9 for source path address decoding.	0,4,8,12
Source Logical Address	The source logical address should be set to the logical address of the node which sent the command or it should be set to the default value of FEh.	1
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number.	2
Extended Read Address	The extended read address is not used in the SpaceWire router and shall always be set to zero.	1
Read Address	The read address identifies the start address for the read incrementing command. The valid starting read addresses are defined in section 0.	4
Data Length	The data length defines the number of bytes to read from the router. Valid data lengths are in the range 4-1064. 1064 allows the all the router registers to be read in one command. If the data length field is not a multiple of four bytes then the command is rejected by the SpaceWire router.	3
Header CRC	The header CRC is the eight bit CRC code used to detect errors in the command packet. The CRC code is checked before the command is executed.	1

In Figure 8-6 the format of the reply to a read incrementing address command is illustrated. The first byte sent by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional. Note that the reply is always sent out of the same port as the command was received on. The Source Path Address should not include the output port of the router being commanded as the reply will be automatically sent out of the same port that the command arrived on. See section 8.6.8.



First byte transmitted

←	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	**Protocol Identifier 01h**	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address FEh	Transaction Identifier (MS)	Transaction Identifier (LS)	**Reserved = 0**
Data Length (MS) 00h	Data Length	Data Length (LS)	Header CRC
Data (Word 0 MSB)	Data	Data	Data (Word 0 LSB)
Data (Word 1 MSB)	Data	Data	Data (Word 1 LSB)
Data (Word n-1 MSB)	Data	Data	Data (Word n-1 LSB)
Data (Word n MSB)	Data	Data	Data (Word n LSB)
Data CRC	**EOP**		

Last byte transmitted

Bits in Packet Type / Command / Source Address Path Length Byte

MSB							LSB
0	0	0	0	1	1	Source Path Address Length	Source Path Address Length
Packet Type		Command				Source Path Address Length	

Figure 8-6 Read Incrementing Address Reply Packet Format

Table 8-10 Read Incrementing Address Reply Packet Fields

Field	Description	Bytes
Source Path Address	Optional source path addresses specified in the command packet. If no source path addresses are specified then the first byte will be the source logical address.	0-12
Source Logical Address	The source logical address specified in the command packet. If source path addresses are not used then the source logical address is the address of the return packet.	1
Protocol Identifier	The RMAP protocol identifier value 01h.	1

Command Byte	Read incrementing address reply command byte. The packet type bits in the command byte indicate this packet is a reply packet.	1
Status	The command status is returned in this field. The command status can be command successful or an RMAP error code as defined in section 8.6.6.	1
Destination logical address	The destination logical address is set to the default value FEh as the SpaceWire router does not have a logical address.	1
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number. The transaction identifier in the reply packet is copied from the command packet and returned in this field, so that the command and the corresponding reply have the same transaction identifier value.	2
Data Length	The data length field is the number of bytes read from the router as specified in the data length field of the command packet.	3
Header CRC	The header CRC used to detect errors in the header part of the command packet. See section 8.6.7 for CRC generation.	1
Data	The data read from the registers in the device. The data is returned in 32 bit words starting from the address specified in read address in the command packet.	>=4
Data CRC	The data CRC used to detect errors in the data part of the reply packet. See section 8.6.7 for CRC generation.	1

8.6.4 Read Modify Write Command

The read-modify-write command characteristics are defined in Table 8-8.

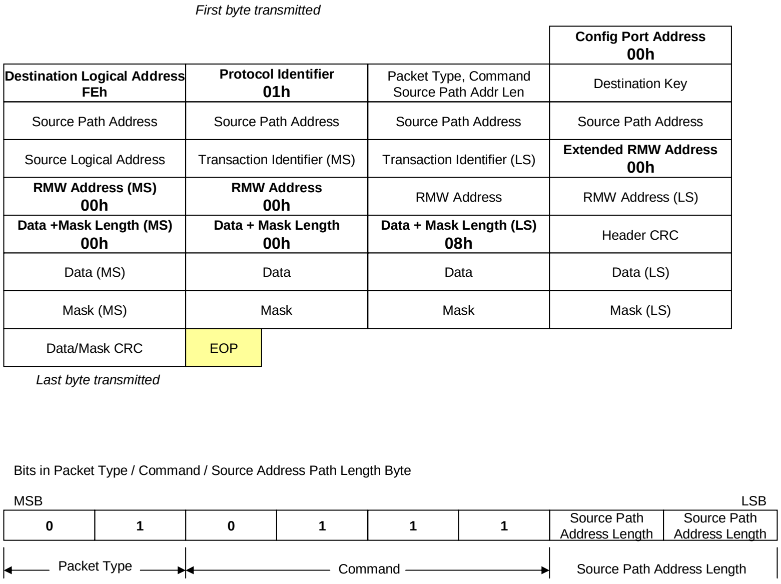
Table 8-11 Read-Modify-Write Command Characteristics

Action	Supported/ Not Supported	Maximum number of bytes	Non-aligned access accepted
8-bit read-modify-write	NS	-	-
16-bit read-modify-write	NS	-	-
32-bit read-modify-write	S	4	No

64-bit read-modify-write	NS	-	-
Word or byte address	32-bit aligned		
Accepted Logical Addresses	0xFE		
Accepted destination keys	0x20 at power on		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 0109		
Incrementing address	No		

The RMAP read-modify-write command is supported by the SpaceWire router. The read modify write command is used to set or reset a single or number of bits in a router register. The Read-Modify-Write command is useful when it is desirable to set a link register setting without upsetting the other settings in one command, i.e. set the start bit without modifying the data rate.

In Figure 8-7 the first byte received by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional.



First byte transmitted

			Config Port Address **00h**
Destination Logical Address **FEh**	**Protocol Identifier** **01h**	Packet Type, Command Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	**Extended RMW Address** **00h**
RMW Address (MS) **00h**	**RMW Address** **00h**	RMW Address	RMW Address (LS)
Data +Mask Length (MS) **00h**	**Data + Mask Length** **00h**	**Data + Mask Length (LS)** **08h**	Header CRC
Data (MS)	Data	Data	Data (LS)
Mask (MS)	Mask	Mask	Mask (LS)
Data/Mask CRC	EOP		

Last byte transmitted

Bits in Packet Type / Command / Source Address Path Length Byte

MSB							LSB
0	1	0	1	1	1	Source Path Address Length	Source Path Address Length
Packet Type		Command				Source Path Address Length	

Figure 8-7 Read-Modify-Write Command Packet Format

Table 8-12 Read-Modify-Write Command Packet Fields

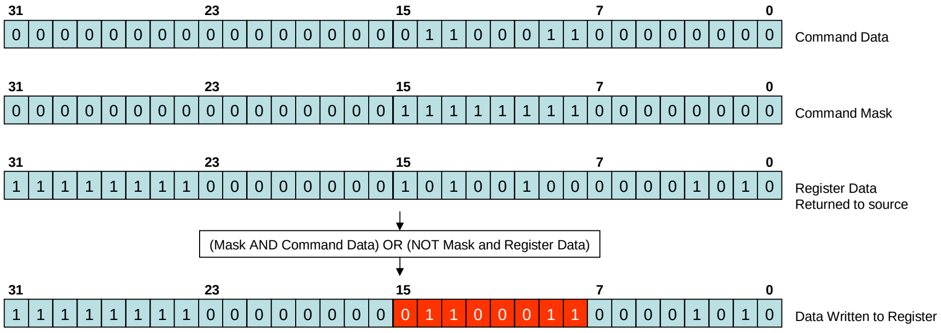
Packet Field	Description	Bytes
Config Port Address	The configuration port address field routes the packet to the configuration port of the router. The configuration port address is always present when configuring the SpaceWire Router.	1
Destination Logical Address	The destination logical address is not used in the SpaceWire Router. The SpaceWire router accepts packets which have the default destination logical address of 254h (FEh).	1
Protocol Identifier	The RMAP protocol identifier is 01h.	1
Command Byte	The command byte indicates a read-modify-write command. The Source path address length fields are set to the number of source path addresses required as defined in section 8.6.9.	1
Destination Key	The destination key identifier must match the contents of the destination key register, see section 10.5.10.	1
Source Path Address	The source path address field is used to add source path addresses to the head of the reply packet. The expected number of source path addresses is specified in the command byte. See section 8.6.9 for source path address decoding.	0,4,8,16
Source Logical Address	The source logical address should be set to the logical address of the node which sent the command.	1
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number.	2
Extended RMW Address	The extended read address is not used in the SpaceWire router and shall always be set to zero.	1
RMW Address	The read-modify-write address identifies the SpaceWire router register address to modify. Valid RMW addresses are defined in section 0.	4
Data + Mask	The data length of the read-modify-write command is 8, 4 bytes for data and	3

Length	4 bytes for the mask to modify a 32-bit register.	
Header CRC	The header CRC used to detect errors in the header part of the command packet.	1
Data and Mask	The data and mask values to write to the SpaceWire router. The data is written dependent on the mask as shown in Figure 8-8.	8
Data and Mask CRC	The data and mask CRC used to detect errors in the data part of the command packet.	1

A Read-Modify-Write command modifies the bits of a SpaceWire router register dependent on the contents of the register (Register Data), the command data (Command Data) and the command mask value (Mask) as follows:

Register Value = (Mask AND Command Data) OR (NOT Mask AND Register Data)

An example is shown below, the highlighted bits are set or reset by the command.



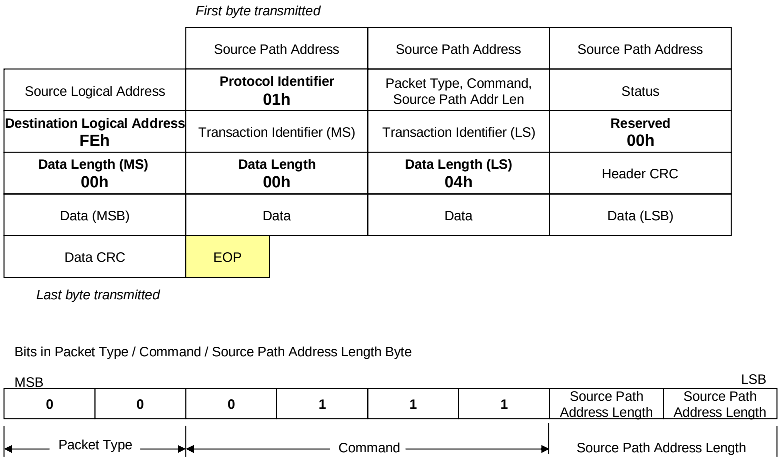
31								23								15								7							0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	Command Data
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	Command Mask
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	1	0	1	0	Register Data Returned to source

(Mask AND Command Data) OR (NOT Mask and Register Data)

31								23								15								7							0	
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	0	0	0	0	1	0	1	0	Data Written to Register

Figure 8-8 Read-Modify-Write example operation

In Figure 8-9 the format of the reply to a Read-Modify-Write command is illustrated. The first byte received by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional. Note that the reply is always sent out of the same port as the command was received on. The Source Path Address should not include the output port of the router being commanded as the reply will be automatically sent out of the same port that the command arrived on. See section 8.6.8.


First byte transmitted
Last byte transmitted
Bits in Packet Type / Command / Source Path Address Length Byte
MSB
LSB
Packet Type
Command
Source Path Address Length

	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	**Protocol Identifier** **01h**	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address **FEh**	Transaction Identifier (MS)	Transaction Identifier (LS)	**Reserved** **00h**
Data Length (MS) **00h**	**Data Length** **00h**	**Data Length (LS)** **04h**	Header CRC
Data (MSB)	Data	Data	Data (LSB)
Data CRC	EOP		

0	**0**	**0**	**1**	**1**	**1**	Source Path Address Length	Source Path Address Length

Figure 8-9 Read-Modify-Write Reply Packet Format

Table 8-13 Read-Modify-Write Reply Packet Fields		
Field	**Description**	**Bytes**
Source Path Address	Optional source path addresses specified in the command packet. If no source path addresses are specified then the first byte will be the source logical address.	0-12
Source Logical Address	The source logical address specified in the command packet. If source path addresses are not used then the source logical address is the address of the return packet.	1
Protocol Identifier	The RMAP protocol identifier value 01h.	1
Command Byte	Read-Modify-Write reply command byte. The packet type bits in the command byte indicate this packet is a response packet.	1
Status	The command status is returned in this field. The command status can be command successful or an RMAP error code as defined in section 8.6.6.	1
Destination	The destination logical address is set to the default value FEh as the	1

logical address	SpaceWire router does not have a logical address.	
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number. The transaction identifier in the reply packet is copied from the command packet and returned in this field, so that the command and the corresponding reply have the same transaction identifier value.	2
Data Length	The data length field is set to 4 bytes as 4 bytes are returned in the Read-Modify-Write command.	3
Header CRC	The header CRC used to detect errors in the header part of the command packet. See section 8.6.7 for CRC generation.	1
Data	The data read from the SpaceWire router registers before the modify operation is performed.	4
Data CRC	The data CRC used to detect errors in the data part of the reply packet. See section 8.6.7 for CRC generation.	1

8.6.5 Write Command

The write command characteristics of the SpaceWire router are defined in Table 8-14.

Table 8-14 Write Command Characteristics

Action	Supported/ Not Supported	Maximum number of bytes	Non-aligned access accepted
8-bit write	NS	-	-
16-bit write	NS	-	-
32-bit write	S	4	No
64-bit write	NS	-	-
Word or byte address	32-bit aligned		
Accepted Logical Addresses	0xFE		
Accepted destination keys	0x20 at power on		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 0109		
Incrementing address	No		

The RMAP write single address, with data verify and acknowledgement command is supported in the SpaceWire router. The RMAP write command is used to write a 32 bit value into one of the SpaceWire router registers.

In Figure 8-10 the first byte received by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional.

			Config Port Address **00h**
Destination Logical Address **FEh**	**Protocol Identifier** **01h**	Packet Type, Command, Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	**Extended Write Address** **00h**
Write Address (MS) **00h**	**Write Address** **00h**	Write Address	Write Address (LS)
Data Length (MS) **00h**	**Data Length** **00h**	**Data Length (LS)** **04h**	Header CRC
Data (MSB)	Data	Data	Data (LSB)
Data CRC	**EOP**		

Bits in Packet Type / Command / Source Path Address Length Byte

MSB							LSB
0	1	1	1	1	0	Source Path Address Length	Source Path Address Length
Packet Type		Command				Source Path Address Length	

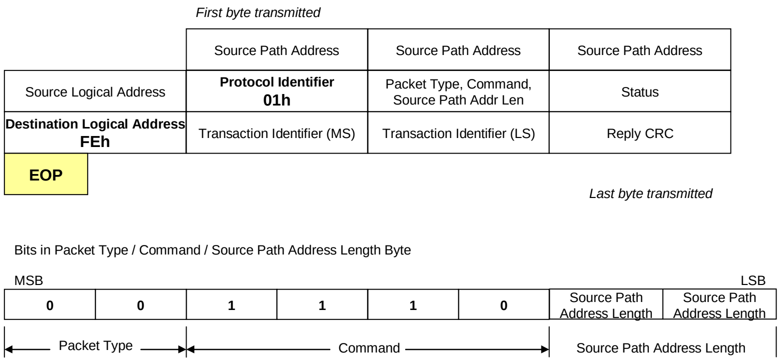
Figure 8-10 Write Single Address Command Packet

Table 8-15 Write Single Address Command Packet Fields

Field	Description	Bytes
Config Port Address	The configuration port address field routes the packet to the configuration port of the router. The configuration port address is always present when configuring the SpaceWire Router.	1
Destination Logical Address	The destination logical address is not used in the SpaceWire Router. The SpaceWire router accepts packets which have the default destination logical address of 254h (FEh).	1
Protocol Identifier	The RMAP protocol identifier is 01h.	1
Command Byte	The command byte indicates a write single address, with verification and acknowledgement packet. The Source path address length fields are set to	1

	the number of source path addresses required as defined in section 8.6.9.	
Destination Key	The destination key identifier must match the contents of the destination key register, see section 10.5.10.	1
Source Path Address	The source path address field is used to add source path addresses to the head of the reply packet. The expected number of source path addresses is specified in the command byte. See section 8.6.9 for source path address decoding.	0,4,8,12
Source Logical Address	The source logical address should be set to the logical address of the node which sent the command.	1
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number.	2
Extended Write Address	The extended write address is not used in the SpaceWire router and is always expected to be zero.	1
Write Address	The write address identifies the register to write the RMAP data. The valid write addresses are defined in section 0.	4
Data Length	The data length of a write single address command is expected to be 4 bytes, to write to a 32 bit register location	3
Header CRC	The header CRC is the eight bit CRC code used to detect errors in the command packet. The CRC code is checked before the command is executed	1
Data	The 32 bit data value to write to the SpaceWire router register.	4
Data CRC	The data CRC used to detect errors in the data part of the command packet.	1

In Figure 8-11 the format of the reply to a write command is illustrated. The first byte sent by the SpaceWire router configuration logic is the port address followed by the destination logical address. Fields which are depicted in bold text are expected values. Fields which are shaded are optional. Note that the reply is always sent out of the same port as the command was received on. The Source Path Address should not include the output port of the router being commanded as the reply will be automatically sent out of the same port that the command arrived on. See section 8.6.8.



First byte transmitted

	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	**Protocol Identifier 01h**	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address FEh	Transaction Identifier (MS)	Transaction Identifier (LS)	Reply CRC
EOP			

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB							LSB
0	0	1	1	1	0	Source Path Address Length	Source Path Address Length
Packet Type		Command				Source Path Address Length	

Figure 8-11 Write Single Address Reply Packet

Table 8-16 Write Single Address Reply Packet Fields		
Field	**Description**	**Bytes**
Source Path Address	Optional source path addresses specified in the command packet. If no source path addresses are specified then the first byte will be the source logical address.	0-12
Source Logical Address	The source logical address specified in the command packet. If source path addresses are not used then the source logical address is the address of the return packet.	1
Protocol Identifier	The RMAP protocol identifier value 01h.	1
Command Byte	Write single address reply command byte. The packet type bits in the command byte indicate this packet is a reply packet.	1
Status	The command status is returned in this field. The command status can be command successful or an RMAP error code as defined in section 8.6.6.	1
Destination logical	The destination logical address is set to the default value FEh as the SpaceWire router does not have a logical address.	1

address		
Transaction Identifier	The transaction identifier identifies the command packet and reply packet with a unique number. The transaction identifier in the reply packet is copied from the command packet and returned in this field, so that the command and the corresponding reply have the same transaction identifier value.	2
Header CRC	The header CRC used to detect errors in the header part of the command packet. See section 8.6.7 for CRC generation.	1

8.6.6 Command Error Response

A summary of the error conditions and the action taken is given in Table 8-17. The error conditions are recorded in the configuration port status register.

Table 8-17 Configuration Port Errors Summary

Register Bits	Description	Reply Packet	Returned As	Returned RMAP Status Code
Invalid Header CRC	The header CRC was invalid therefore the header is corrupted	No	*No Reply Packet.*	-
Unsupported Protocol Error	The protocol byte is not the RMAP protocol identifier	No	*No Reply Packet.*	-
Source Logical Address Error	The source logical address is invalid (outside the range 20h-FFh)	No	*No Reply Packet.*	-
Source Path Address Sequence Error	The source path address sequence is invalid as specified in section 8.6.9	No	*No Reply Packet.*	-
Unused RMAP command or packet type	The command code is an unused command code or the packet type is invalid.	Yes	Unused RMAP command or packet type	2
Invalid Destination Key	The destination key in the command packet is invalid.	Yes	Invalid Destination Key	3
Invalid Data CRC	The Data CRC is invalid therefore the data part of the	Yes	Invalid Data CRC	4

	packet is corrupted			
Early EOP	The command packet was terminated early with an EOP. A reply packet is sent if the early EOP error occurs on the data part of the packet	Yes	Early EOP	5
Cargo too Large	The expected amount of SpaceWire cargo has been received without receiving an EOP marker	Yes	Cargo too large	6
Early EEP	The command packet was terminated early with an EEP. A reply packet is sent if the early EEP error occurs on the data part of the packet	Yes	Early EEP	7
Verify Buffer Overrun Error	The data length field is invalid when performing a verified write command. The valid length is 4 bytes of data.	Yes	Verify Buffer Overrun	9
Command not implemented	A command code was received which is not supported by the SpaceWire router. Supported command codes are listed in F1-18.	Yes	RMAP Command not implemented or not authorised	10
Invalid Data Length	The data length field is invalid. A data length error is recorded when: 1. The data length is not a multiple of 4. 2. The data length is zero 3. The data length is outside the range 4-1064 when performing an incrementing read	Yes	RMAP Command not implemented or not authorised (ote 1: a Verify Buffer Overrun error shall be returned when the data length is not 4 in a verified write command. Note 2: a Read Modify Write Data Length error shall be returned when the data length is not 8.	10

	4. The data length is not 4 in a verified write command. 5. The data length is not 8 in a read modify write command.			
Invalid Register Address	The address field is addressing an unknown register for a read command or a read only register in a write command.	Yes	RMAP Command not implemented or not authorised	10
Read Modify Write Data Length Error	The read modify write data length is not 8	Yes	RMW Data Length Error	11
Invalid Destination Logical Address	The destination logical address is invalid. The destination logical address is expected to be the default 254 value	Yes	Invalid Destination Logical Address	12

8.6.7 Command Packet Cyclic Redundancy Check

The header and data part of an RMAP packet are protected from errors by the use of an 8 bit CRC code. The header and data CRC is formed using the CRC-8 code used in ATM (Asynchronous Transfer Mechanism). CRC-8 has the polynomial: $X^{8} + X^{2} + X^{1} + 1$ wait a starting value of 00h.

Command packets received by the SpaceWire router which have an invalid header CRC are discarded and the Invalid Header CRC bit is set in the configuration port register.

8.6.8 Local Source Path Address

The configuration reply packet shall be routed out of the router port the packet arrived on. For example, if SpaceWire port 1 passed a configuration command to the configuration port then the reply packet is returned to port 1.

8.6.9 Source Path Address Field

The RMAP command field “source path address length” indicates the number of source path addresses which are expected in the packet. Up to 12 source path addresses can be accepted by the router configuration port. The source path addresses shall be decoded by the SpaceWire router as follows.

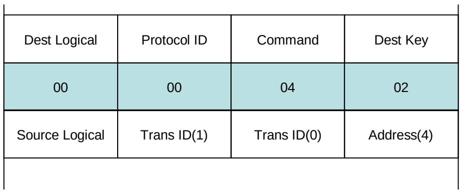
- Leading zero source path address bytes are not returned in the RMAP reply packet.
- If the source path address contains only zero bytes the Source Path Address Sequence error is reported, see section 8.6.6.
- After the first non zero byte in the packet any following zeros shall be treated as an error. A source path address sequence is reported, see section 8.6.6.

The table below shows how to set the source path address length and packet address fields for the required path addresses

Table 8-18 Source Path Address Reference Table

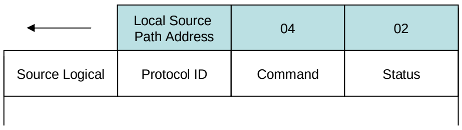
Source Path Address Length	RMAP Source Path Address fields (First→Last Transmitted)	Reply Path Address (First→Last Reply)
0	None	None
1	[00 00 00 20]	20
1	[00 02 08 09]	02 08 09
1	[01 02 03 04]	01 02 03 04
2	[00 00 00 00] [00 00 00 02]	02
2	[00 00 00 00] [01 02 03 02]	01 02 03 02
2	[00 00 12 01] [02 B2 03 05]	12 01 02 B2 03 05
2	[00 32 01 02] [07 02 05 08]	32 01 02 07 02 05 08
1	[00 00 00 00]	Invalid
2	[00 00 00 00] [00 00 00 00]	Invalid
1	[00 02 00 01]	Invalid
1	[00 A3 00 00]	Invalid
2	[00 02 03 00] [01 00 00 00]	Invalid
2	[00 00 00 02] [00 00 01 00]	Invalid
2	[00 00 00 00] [02 03 00 01]	Invalid

Figure 8-12 and Figure 8-13 illustrate how source path addresses are returned in relation to the RMAP packet description.



Dest Logical	Protocol ID	Command	Dest Key
00	00	04	02
Source Logical	Trans ID(1)	Trans ID(0)	Address(4)

Figure 8-12 Source Path Address field decoding

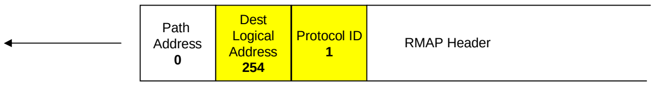


←	Local Source Path Address	04	02
Source Logical	Protocol ID	Command	Status

Figure 8-13 Source Path Addresses in Reply Packet

8.6.10 Command Packet Fill Bytes

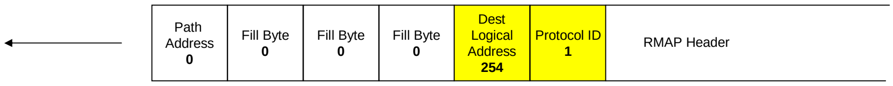
The Configuration port accepts packets which are addressed to port 0. In the RMAP command the next byte after the destination address 0 is the destination logical address byte (which in the router is expected to be the default 254 value). The format is shown in Figure 8-14.



←	Path Address **0**	Dest Logical Address **254**	Protocol ID **1**	RMAP Header

Figure 8-14 Normal Configuration Packet Header Structure

To allow source nodes which have a 16, 24 or 32 bit access port then the configuration port accepts up to three null bytes at the start of the packet. The null bytes must be zero otherwise they will be treated as the destination logical address and an invalid destination logical address shall be recorded if the byte is not 254. The header with fill bytes is shown in the Figure 8-15.



←	Path Address **0**	Fill Byte **0**	Fill Byte **0**	Fill Byte **0**	Dest Logical Address **254**	Protocol ID **1**	RMAP Header

Figure 8-15 Fill Bytes Configuration Header Structure

9. CONTROL LOGIC AND OPERATIONAL MODES

In this section the SpaceWire router control logic and operational modes are defined. The router control logic determines how the SpaceWire link ports operate, how received packets are routed to their destination and how the timeout mechanism detects packet blockages in the router.

9.1 SPACEWIRE LINK CONTROL

Each of the eight SpaceWire links has an associated SpaceWire control register. The register records status information from each link including link error information, link state and run status (see section 10.4.3).

The SpaceWire link control bits determine how the SpaceWire link operates. The link control bits are Auto-start (default), Link-Start, Link-Disable and Tri-State. The SpaceWire link data rate divider can also be set in the link control register.

The following paragraphs define each of the link control functions

9.1.1 Default operating mode

The default operating mode is Auto-Start. This is the mode setting for each link after power on or reset.

9.1.2 Auto-Start

In auto-start mode the SpaceWire port will remain inactive until a connection attempt is made by the SpaceWire device at the other end of the SpaceWire link. The port will then start-up and make the connection

The Auto-Start mode in conjunction with the automatic Link-Start and disable modes can help reduce power consumption by only activating SpaceWire links when packet data is transferred. See section 9.1.7.

9.1.3 Link-Start

The link-start control bit commands the SpaceWire port to try to make a connection with a SpaceWire device at the other end of the link. Assuming a SpaceWire device is connected to the other end of the link the SpaceWire port will move to state Run. Data transfer can take place when the link is started and the Link state is Run.

9.1.4 Link-Disable

The SpaceWire port can be disabled therefore rendering the link unusable. When a SpaceWire link which is running is disabled it will disconnect from the far end and refuse connection attempts by the far end of the link. Caution should be used when using this command for a stand alone router as disabling all the links will leave the router unusable, except through a reset operation.

WARNING: If the link that is being used to configure the router is disabled then it will not be possible to configure the router, unless there is another not disabled connection that can be used.

9.1.5 Automatic tri-state enable mode

The SpaceWire link tri-state bit can be set to cause the data and strobe outputs for the link to be disabled when the link is inactive. The tri-state mode takes effect dependent on the state of the Auto-start or the Link-Disabled control bits.

When Auto-start is enabled and the tri-state bit is set then the data and strobe LVDS drivers are tri-stated until the interface receives a connection attempt by an external SpaceWire device. The drivers are then enabled until the external device disconnects the link or the SpaceWire link control bit setup is changed.

When Link-disable is asserted and the tri-state bit is set then the data and strobe LVDS drivers are always tri-stated as the link is not in use.

The figure below shows the effect of the tri-state bit when the link setting is Auto-start only.

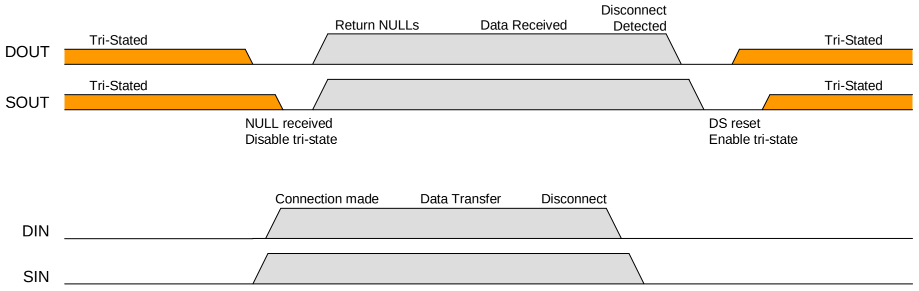


Figure 9-1 Tri-state enable operating mode

Note the DOUT tri-state enable/disable is performed one CLK cycle before the SOUT tri-state, avoiding any simultaneous transitions on Data or Strobe.

9.1.6 Setting the data-rate

The SpaceWire link data-rate is dependent on the input signal FEEDBDIV (See section 4.1), the PLL output clock divider value TXDIV (See section 10.5.9) and the data-rate divider value in each

SpaceWire link control register (See section 10.4.3). The resultant data rate is determined by the function.

$$DataRate = \left(\frac{\left(\frac{(100MHz + (20MHz * FEEDBDIV))}{2^{TXDIV+1}} \right)}{TXRATE+1} \right) * 2$$

Where **FEEDBDIV** is in the range 0-5.

Note: If (**TXRATE**+1) is an even number then the duty cycle will not be 50/50 on the output bit-stream. The worst case is when **TXRATE =1** (i.e. **TXRATE**+1=2) then the duty cycle will be 75/25.

Setting the Transmit Data-Rate:

1. Set the FEEDBDIV pins to give the required PLL clock rate.

2. Set the base transmit data-rate using the TXDIV divider. Do this once only.

3. Set the required transmit data-rate of each link individually using the TXRATE divider. The resulting transmit data-rate must be equal to or higher than 2 Mbits/s.

WARNING: If a data-rate of less than 2 Mbits/s is set then the link will repeatedly connect and disconnect.

9.1.7 Setting the default 10Mbits/s data-rate

The input signal **FEEDBDIV** and the transmit control register value **TxDiv** select the input clock frequency to the link transmitters. At link start-up and initialisation the default data-rate is set by the transmit clock control register value **TX10MBitDiv** where TX10MbitDIV should be set such that**:**

$$Tx10MbitDIV = \left(\frac{\left(\frac{100MHz + (20MHz * FEEDBDIV)}{2^{TXDIV+1}} \right)}{5MHZ} \right) - 1$$

Figure 9-2 Tx10MBitDIV equation

Where FEEDBDIV is in the range 0-5.

Note: If Tx10MbitDIV is an odd number then the duty cycle on DOUT and SOUT will not be 50/50. In the worst case when Tx10MbitDIV=1 the duty cycle will be 75/25.

9.1.8 Data rate comparison table

			TXDIV					
			0	0	1	1	2	2
		PLLFreq	Min	Max	Min	Max	Min	Max
FEEDBDIV	0	100	0.78	100	0.39	50	0.20	25
	1	120	0.94	120	0.47	60	0.23	30
	2	140	1.09	140	0.55	70	0.27	35
	3	160	1.25	160	0.63	80	0.31	40
	4	180	1.41	180	0.70	90	0.35	45
	5	200	1.56	200	0.78	100	0.39	50

All Numbers in Mbits/s

The maximum numbers are the data rates that are achieved when the TXRATE divider is 0. The minimum values are those achieved when TXRATE divider is set to 7. See section 10.4.3.

9.2 GLOBAL SPACEWIRE LINK CONTROL

The following modes are global to all SpaceWire links. The modes can be set in the SpaceWire router control register. (see section 10.5).

9.2.1 Automatic Link-Start mode

The automatic link start mode is set in the router control register bit setting **CFG_START_ON_REQ**. The input signal **POR_START_ON_REQ_N** determines the power on or reset state of the register bit.

The SpaceWire router automatic Link-Start mode is used to instruct a link to automatically attempt startup when a packet is received whose destination is a SpaceWire link which is not ready (i.e. not Running).

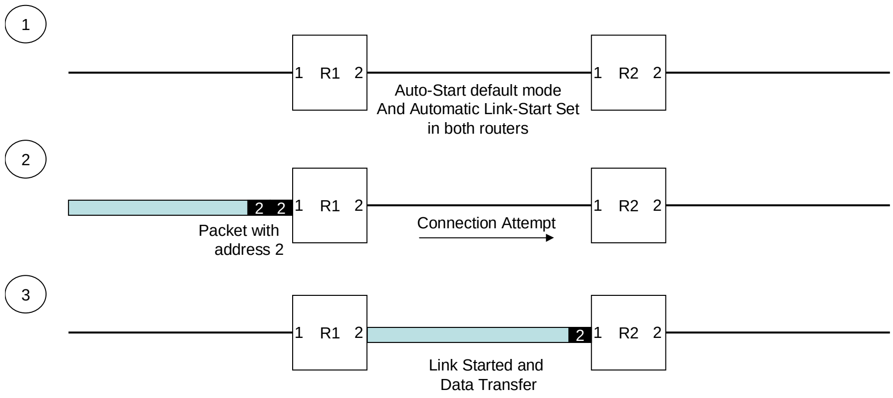


Figure 9-3 Automatic Link-Start mode

9.2.2 Automatic Link-Disable mode

The automatic Link-Disable mode is set in the router control register bit setting **CFG_DISABLE_ON_SILENCE**. The input signal **POR_DISABLE_ON_SILENCE_N** determines the power on or reset state of the register bit.

The figure below defines the usage of the automatic link-disable mode.

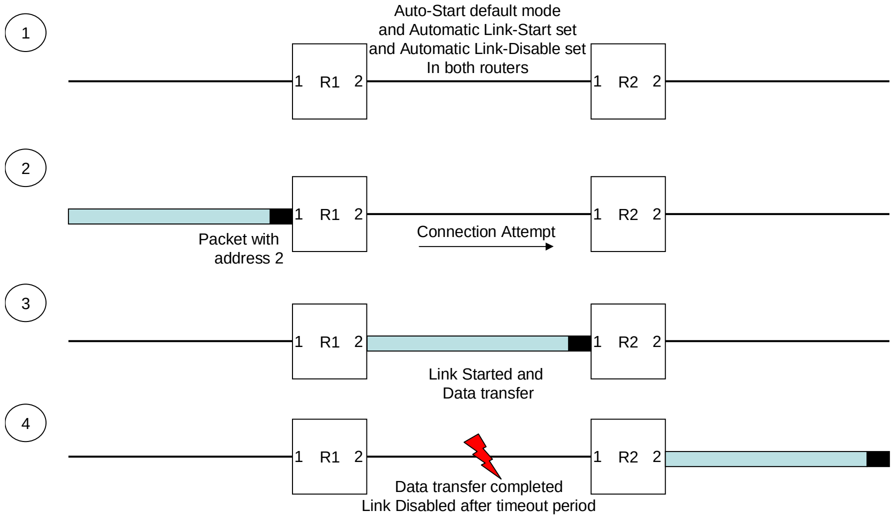


Figure 9-4 Automatic Link-Disable mode

The SpaceWire router automatic Link-Disable mode is used to disable a link after it has been used to send packet data. The Automatic Link-Disable mode is enabled only when the router timeout setting has been enabled. The link is disabled if no packet activity has been detected for the timeout period set in the router control register.

The SpaceWire router will only disable the SpaceWire link when the SpaceWire router is the source of the data transfer. If an external device starts the SpaceWire link or sends packet data to the router through the link then the link will not be disabled.

9.3 CONTROL LOGIC AND ROUTING

This section defines the function of the SpaceWire routing logic and how packets are handled dependent on the router control modes of operation. The modes of operation which affect the router are the router control register settings, timeout enable, address self, and start on request.

9.3.1 Packet address error

A packet address error is detected when a packet with an invalid address, see section 8.4, is received. The packet is discarded by the router independent of the current mode settings and the router is set up to receive the next packet after the invalid address packet has been spilt. The next input port to transfer data to an output port is the next highest port number (modulo number of ports) that has data to send. For example, if port 2 is sending data out of port 1 and ports 5 and 7 are waiting to send data through port 1, then port 5 will selected next.

9.3.2 Arbitration

Arbitration is performed by the SpaceWire router when two or more packets are to be routed through the same destination port.

Packets which have high priority are always selected before low priority packets in the router. If two or more packets of the same priority level are attempting to use the same destination port then the packets are arbitrated in a fair manner.

The router chooses the next packet to be routed to the destination port dependent on the previous port which had access to the router. Therefore the port which previously had access to the output port will be selected last by the router control logic.

Note that configuration packets, both commands and replies, are treated the same as any other packet as far as arbitration is concerned. The arbitration scheme is equally applicable to all types of packets with no exceptions. When sending long packets, arbitration can cause substantial delays in transferring information.

The following sections illustrate the scenarios described above.

9.3.2.1 Arbitration of packets with matching priority (1)

In the figure below only router ports 1-5 are shown for clarity.

At stage one ports 1 and 3 have packets to be routed to port 5. The previous port to use port 5 was port 3 therefore the next port to be selected by port 5 will be port 1 (assuming ports 6, 7, 8, 9, 10 and 0 are not requesting to use the port).

At stage two the router selects the packet arriving at port 1 and the packet is routed through port 5. Port 3 waits until all of the packet from port 1 has been transferred.

At stage three all port 1's packet data has been transferred, therefore port 3 is selected and the packet data is routed through port 5.

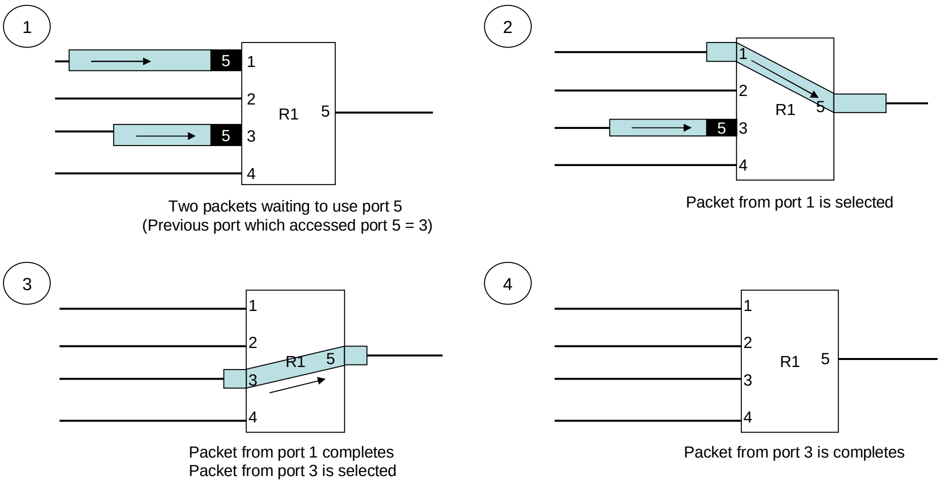

1
5
1
2
R1
5
5
3
4
Two packets waiting to use port 5
(Previous port which accessed port 5 = 3)
2
1
2
R1
5
5
3
4
Packet from port 1 is selected
3
1
2
R1
5
3
4
Packet from port 1 completes
Packet from port 3 is selected
4
1
2
R1
5
3
4
Packet from port 3 is completes

Figure 9-5 Arbitration of two packets with matching priority.

9.3.2.2 Arbitration of packets with matching priority (2)

In the figure below only router ports 1-5 are shown for clarity.

At stage one ports 1 and 3 have packets to be routed to port 5. The previous port to use port 5 was port 3 therefore the next port to be selected by port 5 will be port 1 (assuming ports 6, 7, 8, 9, 10 and 0 are not requesting to use the port).

At stage two the router selects the packet at port one and the packet is routed through port 5. Port number 3 waits until all of that packet has been transferred. While the packet from port 1 is routed to port 5 another packet arrives at port 2 to be routed to port 5.

At stage three the packet from port 1 has been forwarded and the packet from port 2 is selected by the router to be routed through port 5. Port number two is selected before port 3 as it is the next port to be considered by the routing control logic after port 1.

At stage four port 2's packet data has been transferred to port 5 therefore port 3 is selected and its packet data is routed through port 5.

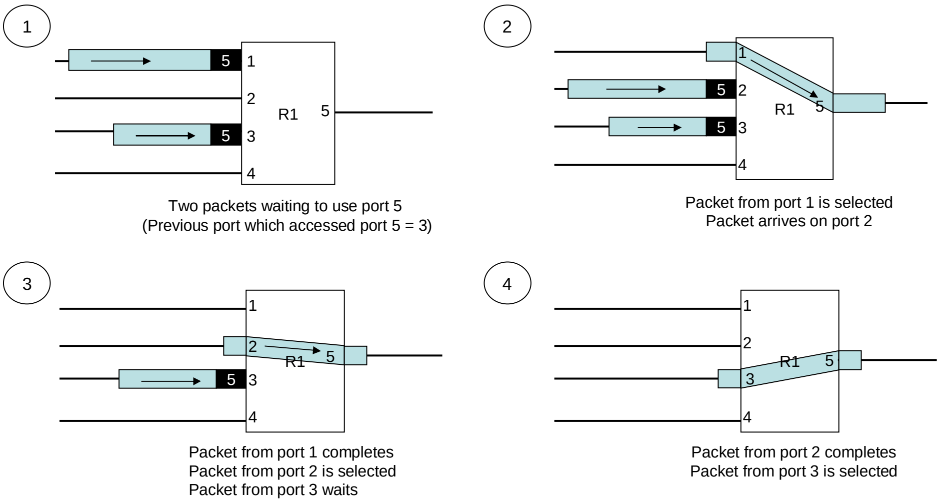

1
5
1
2
R1
5
5
3
4
Two packets waiting to use port 5
(Previous port which accessed port 5 = 3)
2
1
5
2
R1
5
5
3
4
Packet from port 1 is selected
Packet arrives on port 2
3
1
2
R1
5
5
3
4
Packet from port 1 completes
Packet from port 2 is selected
Packet from port 3 waits
4
1
2
R1
5
3
4
Packet from port 2 completes
Packet from port 3 is selected

Figure 9-6 Arbitration of three packets with matching priority

9.3.2.3 Arbitration of packets with different priority (1)

In the figure below only router ports 1-5 are shown for clarity.

At stage one ports 1 and 3 have packets with logical addresses 80 and 52 which are both to be routed to port 5. The previous port selected by port 5 was port 2 but logical address 80 has HIGH priority therefore will be selected before address 52 which has LOW priority.

At stage two the router selects the packet arriving at port 1 and the packet is routed through port 5.

At stage three the packet from port 1 has been transferred and the packet from port 3 is selected by the router to be routed through port 5.

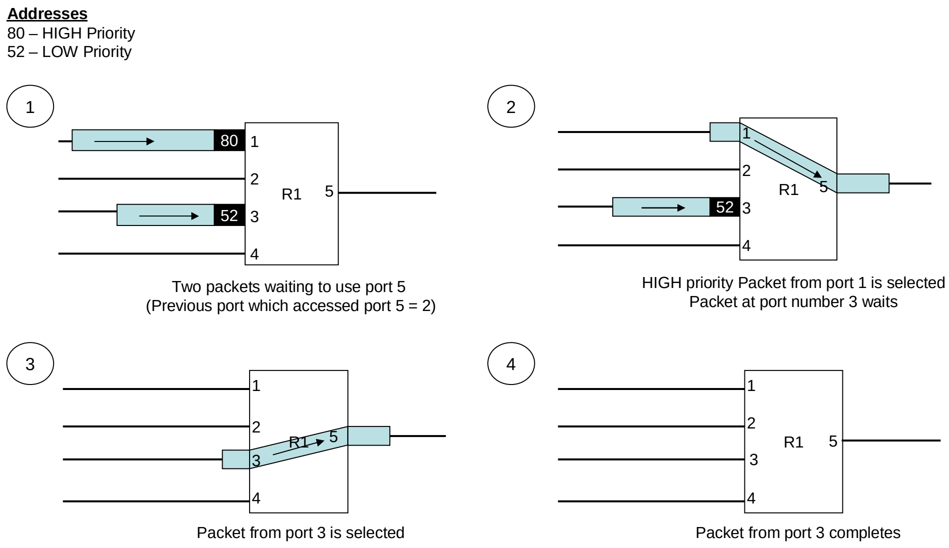

Addresses
80 – HIGH Priority
52 – LOW Priority
1
80
52
1
2
3
4
R1
5
Two packets waiting to use port 5
(Previous port which accessed port 5 = 2)
2
52
1
2
3
4
R1
5
HIGH priority Packet from port 1 is selected
Packet at port number 3 waits
3
1
2
3
4
R1
5
Packet from port 3 is selected
4
1
2
3
4
R1
5
Packet from port 3 completes

Figure 9-7 Arbitration of two packets with different priority

9.3.2.4 Arbitration of packets with different priority (2)

In the figure below only router ports 1-5 are shown for clarity.

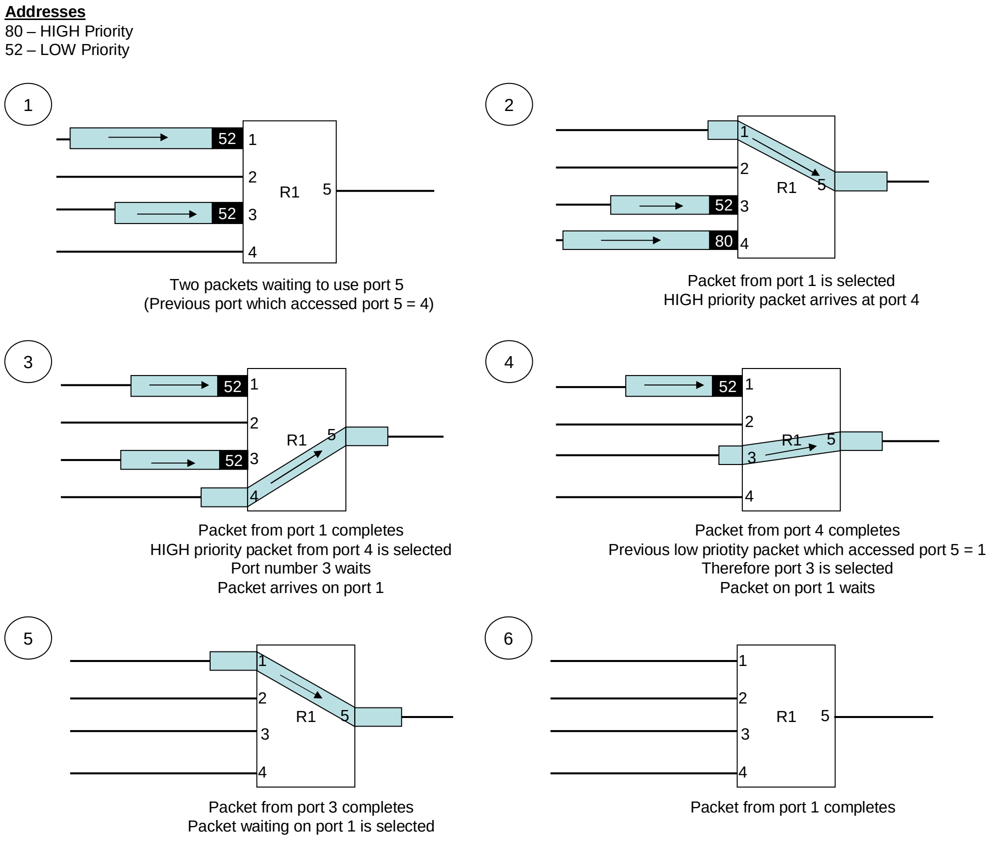
At stage one ports 1 and 3 have packets with logical addresses 80 and 52 which are both to be routed to port 5. The previous port selected by port 5 was port 4 therefore the packet waiting at port 1 is routed, but logical address 80 has HIGH priority therefore will be selected before address 52 which has LOW priority.

At stage two the router selects the packet arriving at port 1 and this packet is routed through port 5. In the meantime a packet with HIGH priority logical address 80 arrives at port 4.

At stage three the packet from port 1 has been forwarded and the packet with HIGH priority at port 4 is selected by the routing control logic as the next packet to be routed to port 5.

At stage number four the high priority packet has been forwarded and the routing control logic must arbitrate again for port 5. The previous low priority packet that was routed through port 5 is 1 therefore the next packet selected by the routing control logic is port 3.

At stage number five the packet from port 3 has completed and the packet waiting at port 1 gets its chance to access port 5 and is forwarded.



Two packets waiting to use port 5
(Previous port which accessed port 5 = 4)

Packet from port 1 is selected
HIGH priority packet arrives at port 4

Packet from port 1 completes
HIGH priority packet from port 4 is selected
Port number 3 waits
Packet arrives on port 1

Packet from port 4 completes
Previous low priotity packet which accessed port 5 = 1
Therefore port 3 is selected
Packet on port 1 waits

Packet from port 3 completes
Packet waiting on port 1 is selected

Packet from port 1 completes

Figure 9-8

9.3.3 Group Adaptive Routing

The SpaceWire router routing table can be set up to support group adaptive routing of packets. Setting the routing table contents is described in section 10.3.

In group adaptive routing a set of output ports can be assigned to a logical address. When a packet arrives with the logical address the routing table is checked for the set of output ports which the packet can use. The routing control logic then checks the possible router ports to determine if they are free and ready to use. When one of the possible ports associated with the logical address of the packet is free and ready to use then the packet is routed through the port.

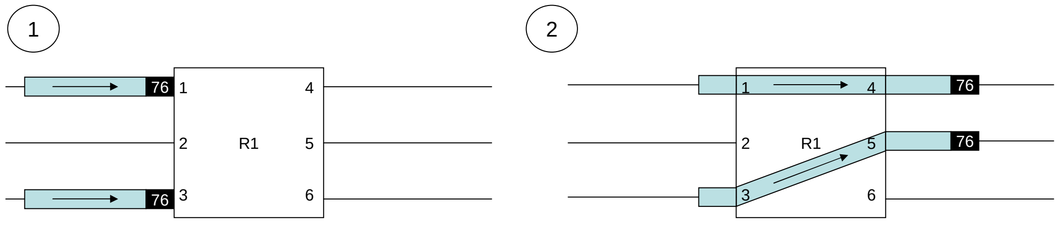
If all the set of ports which the logical address packet can use are free then the router chooses the lowest numerical destination port number to route the packet. The figures below detail the group adaptive routing scenarios defined above.

Arbitration is performed on group adaptive routing packets as defined in section 9.3.2.

9.3.3.1 Normal Group adaptive routing

In normal group adaptive routing then the lowest numerical output port is assigned to the packet.

Address 76 – Routing table entry
Header Deletion disabled
Port 4
Port 5
Port 6



Group adaptive routing packet with address 76 arrives at port 1
Group adaptive routing packet with address 76 arrives at port 3

Routing logic assigns port 4 to port 1
And port 5 to port 3

Figure 9-9 Normal group adaptive routing

9.3.3.2 Group adaptive routing when busy

In this scenario, two of the ports which address 76 can use to route towards the destination are busy. The packet is assigned to the free port 6 and the packet is routed out of port number 6.

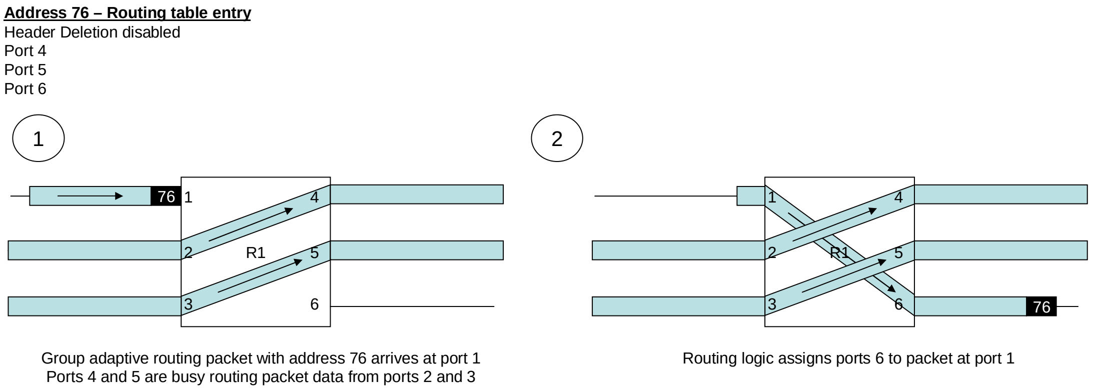

Address 76 – Routing table entry
Header Deletion disabled
Port 4
Port 5
Port 6
1
76
1
2
3
4
5
6
R1
Group adaptive routing packet with address 76 arrives at port 1
Ports 4 and 5 are busy routing packet data from ports 2 and 3
2
1
2
3
4
5
6
R1
76
Routing logic assigns ports 6 to packet at port 1

Figure 9-10 Group adaptive routing when other ports busy

9.3.3.3 Group adaptive routing when ports not ready

In this scenario, two of the ports which address 76 can use to route towards the destination are not ready for use (i.e. the links are not running). The packet is assigned to link number 6 by the routing control logic as it can be used immediately to route the packet to the destination.

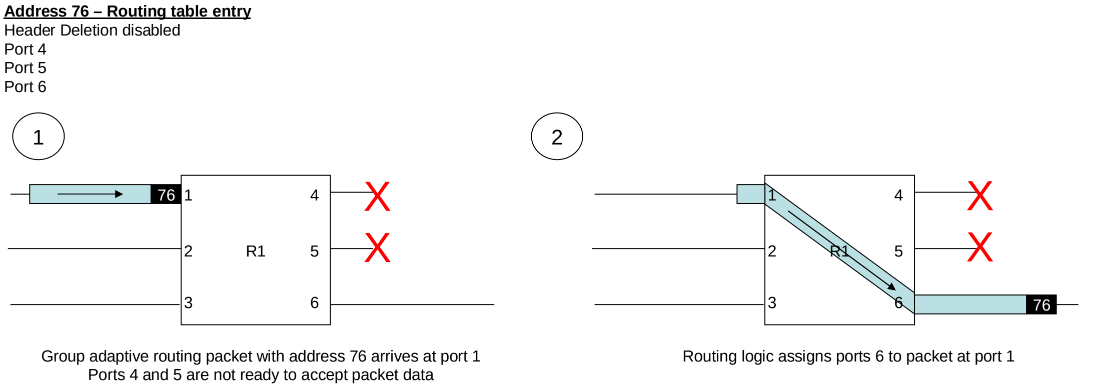

Address 76 – Routing table entry
Header Deletion disabled
Port 4
Port 5
Port 6
1
76
1
2
3
4
5
6
R1
X
X
Group adaptive routing packet with address 76 arrives at port 1
Ports 4 and 5 are not ready to accept packet data
2
1
2
3
4
5
6
R1
X
X
76
Routing logic assigns ports 6 to packet at port 1

Figure 9-11 Group adaptive routing when ports not ready

Note if the automatic link-start mode is enabled then the ports which are not ready will attempt to make a connection. In this way the group adaptive routing packet will be routed to the port which is ready first.

9.3.4 Loop-back routing

The router control register "address-self" bit determines if the router supports loop-back connections. Loop-back connections can be useful for debugging or ping operations where a packet is "bounced" of the router and returned to the source. If the address self bit is disabled then the packet is discarded and a packet address error is recorded.

Command reply packets which are returned through the same port they arrived on are not affect by the value of the "address-self" register bit.

The diagram below shows the "address-self" mode when enabled and when disabled.

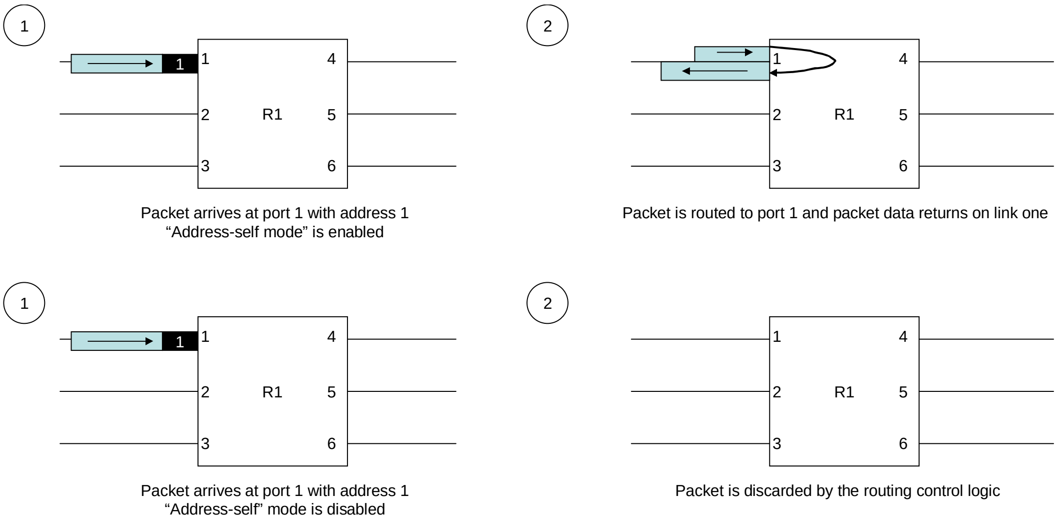

1
1
1
2
3
R1
4
5
6
Packet arrives at port 1 with address 1
"Address-self mode" is enabled
2
1
2
3
R1
4
5
6
Packet is routed to port 1 and packet data returns on link one
1
1
1
2
3
R1
4
5
6
Packet arrives at port 1 with address 1
"Address-self" mode is disabled
2
1
2
3
R1
4
5
6
Packet is discarded by the routing control logic

Figure 9-12 Packet address self mode

9.3.5 Packet Blocking

As described in the arbitration and group adaptive routing sections above, sections 9.3.2 and 9.3.3, a packet is routed to the destination port dependent on the packet address and the state of the router. A packet is only routed to the destination port when the destination port is ready. The definition of ready for each type of port is listed in the table below.

Table 9-1 Port ready definitions

Physical Port type	Ready when
Configuration port	Port data buffer is not full.
SpaceWire link port	Port data buffer is not full **and** SpaceWire link is in the run state.
External FIFO port	Port data buffer is not full.

When a router port is ready, a packet can be routed towards its destination and the routed resources are freed. When a router port is not ready then a packet is blocked in the router as the packet cannot be routed towards the destination.

The scenarios which can cause packet blocking are defined below.

Table 9-2 Packet blocking

Types of blocking	Effect of blocking
Packet has been routed to destination but packet destination stalls and stops accepting data before the complete packet has passed through router.	Packet is blocked in router and destination port will not become free until packet destination starts accepting data again.
Packet has been routed to destination port but the source of the packet stalls and the complete packet has not passed through the router.	Destination port is blocked waiting for end of packet. Destination port will not become free until packet source supplies the end of packet.
Packet destination port is a SpaceWire port which is not connected to an external device.	Packet cannot be routed by router as physical link connection does not exist.
Packet destination port is a SpaceWire port which is not started	Packet cannot be routed by router unless automatic Link-Start is enabled

For the packet blocking types listed above the router may become blocked for an indefinite period of time and a reset of the router would be required to free the blockage. Packets which are blocked may also cause blockages through an entire network as the packet data buffering on the router is small therefore the tail of a packet may be spread over a number of routers or the entire network. The packet timeout mechanism defined below provides a method of ensuring that prolonged packet blocking does not occur in the router.

Two modes of blocking are be supported in the SpaceWire router, “Blocking allowed” and “Blocking Watchdog Timer”.

9.3.5.1 Blocking Allowed

In the blocking allowed mode packets wait indefinitely on other packets to complete. When a packet of infinite length is being routed then no other packets can access that channel and the packets will block for an infinite time. The disadvantage of this method is that if a destination node stops reading a packet then the packet will block the network indefinitely.

An exception to this occurs when the router output port to be used is a SpaceWire port and the port is not started. In this case the packet waits as long as the timeout period and is then discarded. If group adaptive routing is used and at least one of the destination ports is running then the packet waits indefinitely.

9.3.5.2 Blocking Watchdog Timer

In the watchdog timer mode a watchdog timer is used to clear packets from the network when they become blocked. The watchdog blocking timer is restarted on each byte of a packet that is transferred. The complete duration of the packet is not checked but instead the timer checks if a packet has blocked (i.e. no data transfers).

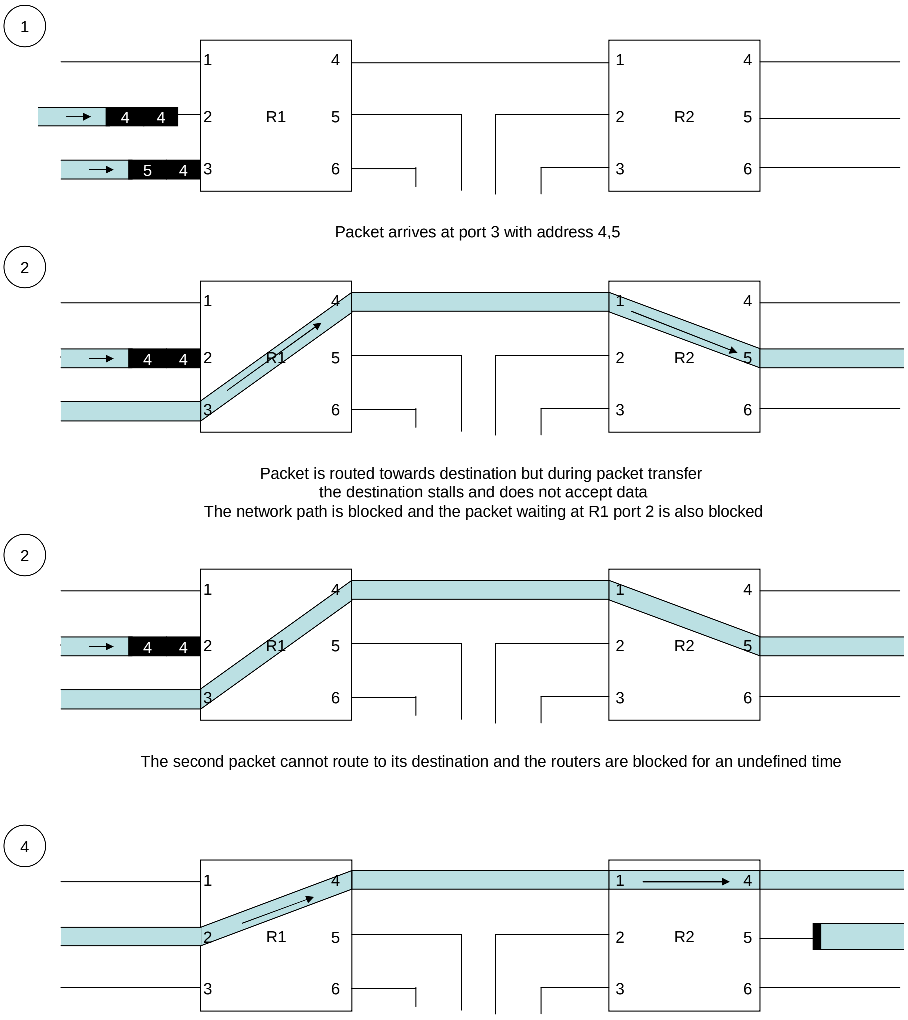
The following sections consider various routing scenarios where blocking occurs. Blocking allowed and watchdog timer modes are examined for each scenario.

9.3.5.3 Scenario 1 Destination Node Blocked

In scenario 1 the destination node is blocked therefore packet data builds up through the routers. In this case the tail of the packet is distributed across multiple routers and other network paths become blocked waiting on the blocked packet to complete.

- In blocking allowed mode the network path is blocked until the destination node starts to accept data again. Packets waiting to use the network path will wait indefinitely.
- In watchdog timer mode the routers will timeout and the network path will be cleared so other packets can use the path. Packets waiting to use the network path will wait for as long as the timeout period.

The two figures below illustrate the actions of the router in the blocking allowed mode and the blocking watchdog timer mode.



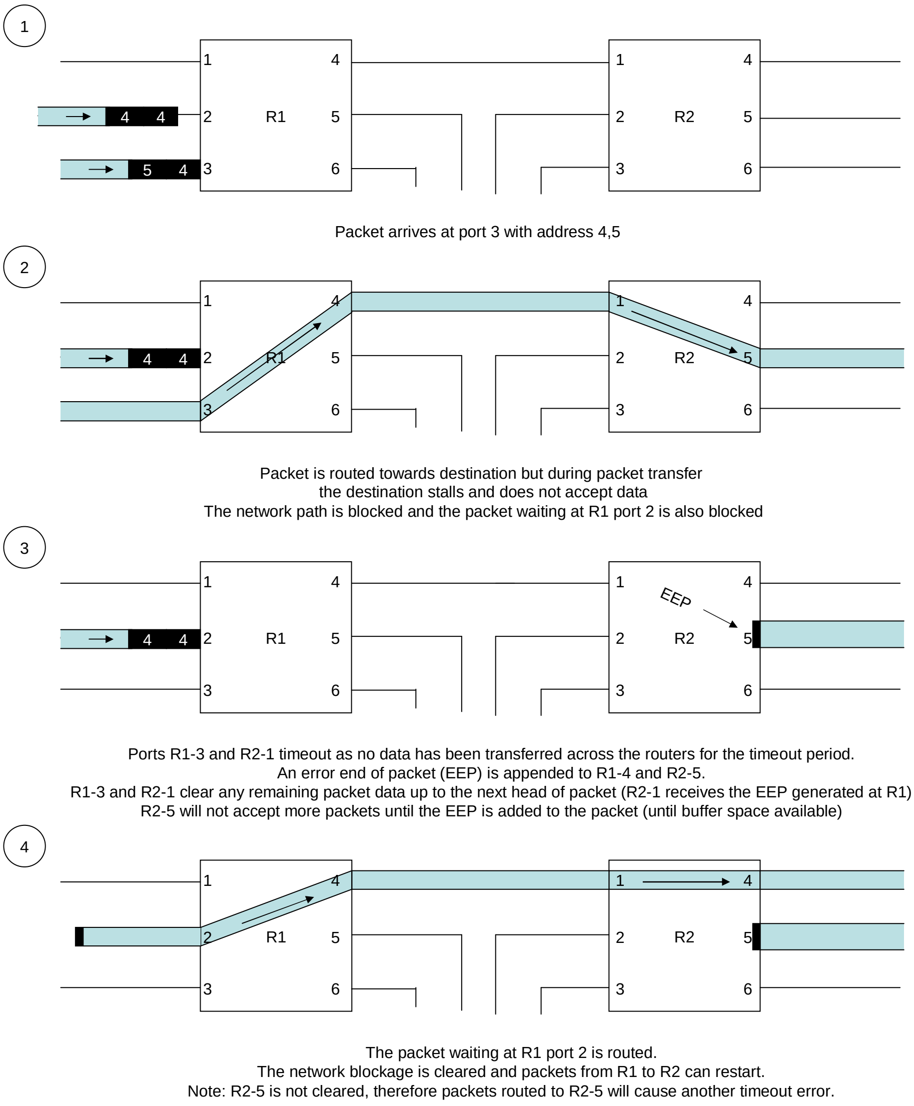
Packet arrives at port 3 with address 4,5

Packet is routed towards destination but during packet transfer
the destination stalls and does not accept data
The network path is blocked and the packet waiting at R1 port 2 is also blocked

The second packet cannot route to its destination and the routers are blocked for an undefined time

After an undefined time the destination node connected to R2-5 accepts data
The packet from R1-2 can route to its destination

Figure 9-13 Scenario 1 Destination blocked (Blocking allowed mode)



Packet arrives at port 3 with address 4,5

Packet is routed towards destination but during packet transfer
the destination stalls and does not accept data
The network path is blocked and the packet waiting at R1 port 2 is also blocked

Ports R1-3 and R2-1 timeout as no data has been transferred across the routers for the timeout period.
An error end of packet (EEP) is appended to R1-4 and R2-5.
R1-3 and R2-1 clear any remaining packet data up to the next head of packet (R2-1 receives the EEP generated at R1)
R2-5 will not accept more packets until the EEP is added to the packet (until buffer space available)

The packet waiting at R1 port 2 is routed.
The network blockage is cleared and packets from R1 to R2 can restart.
Note: R2-5 is not cleared, therefore packets routed to R2-5 will cause another timeout error.

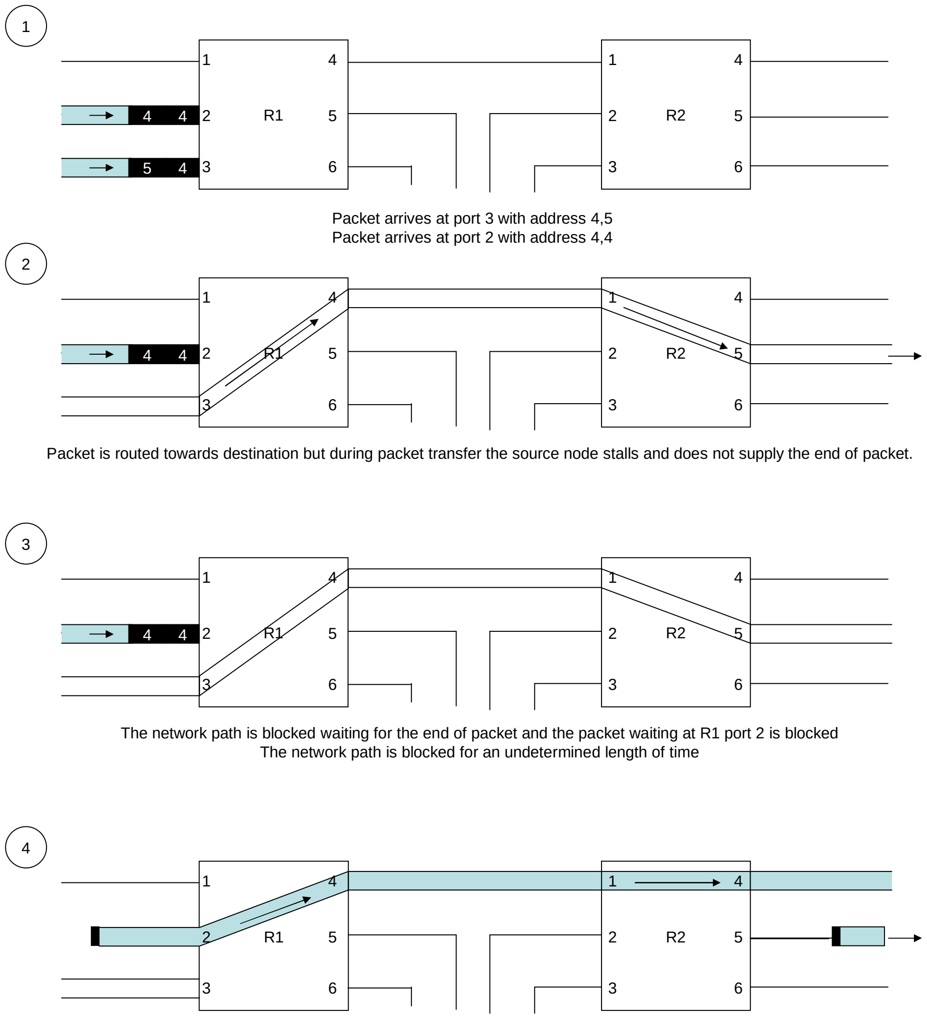
Figure 9-14 Scenario 1 Destination blocked (Watchdog Timer Mode)

9.3.5.4 Scenario 2 Source Node Stalled

In scenario 2 the source node stalls after the head of packet is routed to the destination node. The network path is blocked as the routers are waiting for the end of packet to be supplied by the source node.

- In blocking allowed mode the network path is blocked until the source node supplies the end of packet. Other packets waiting to use the network path will wait indefinitely.
- In watchdog timer mode the routers will timeout and the network path will be cleared so other packets can use the path. Other packets waiting to use the network path will wait as long as the timeout period.

The two figures below illustrate the actions of the router in the blocking allowed mode and the blocking watchdog timer mode.



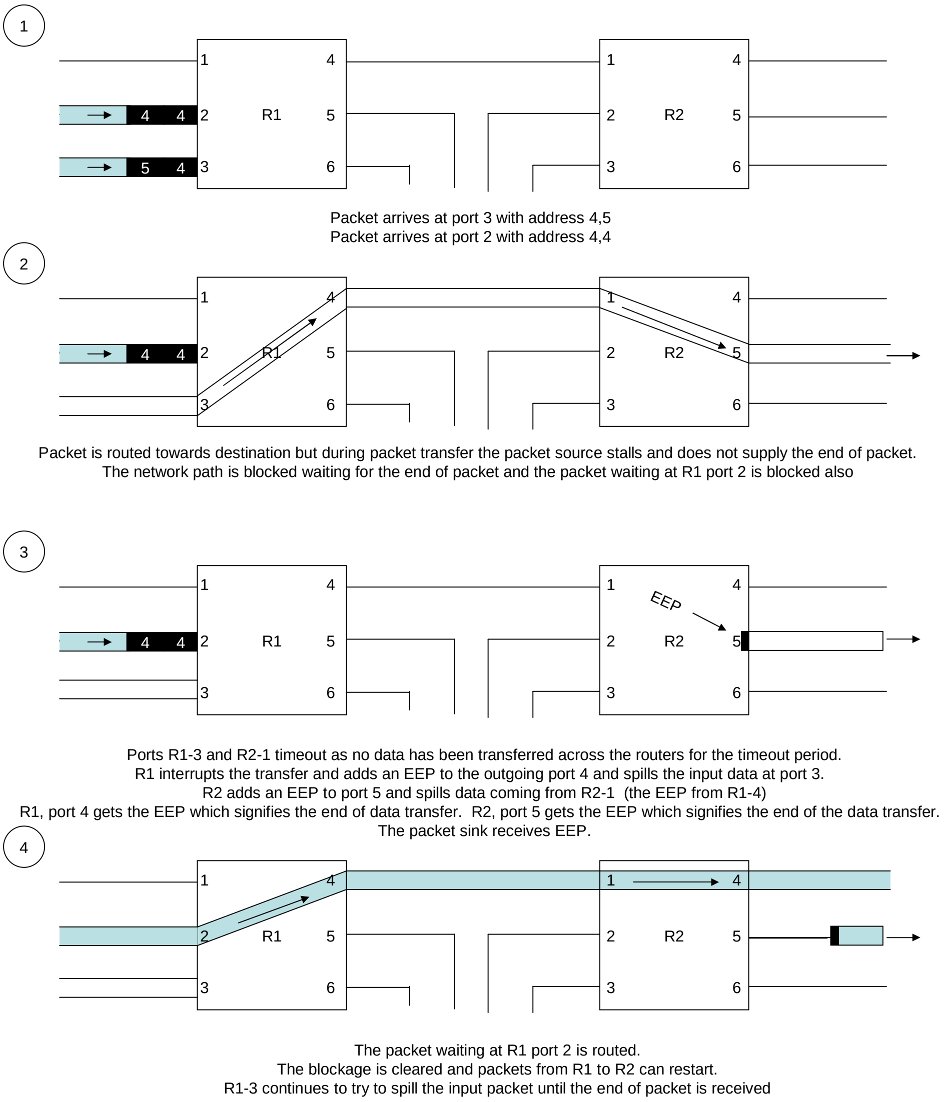
Packet arrives at port 3 with address 4,5
Packet arrives at port 2 with address 4,4

Packet is routed towards destination but during packet transfer the source node stalls and does not supply the end of packet.

The network path is blocked waiting for the end of packet and the packet waiting at R1 port 2 is blocked
The network path is blocked for an undetermined length of time

After an undetermined time the source node supplies the end of packet and the packet waiting at R1-2 can be routed

Figure 9-15 Scenario 2 Source node stalled (Blocking allowed mode)



Packet arrives at port 3 with address 4,5
Packet arrives at port 2 with address 4,4

Packet is routed towards destination but during packet transfer the packet source stalls and does not supply the end of packet.
The network path is blocked waiting for the end of packet and the packet waiting at R1 port 2 is blocked also

Ports R1-3 and R2-1 timeout as no data has been transferred across the routers for the timeout period.
R1 interrupts the transfer and adds an EEP to the outgoing port 4 and spills the input data at port 3.
R2 adds an EEP to port 5 and spills data coming from R2-1 (the EEP from R1-4)
R1, port 4 gets the EEP which signifies the end of data transfer. R2, port 5 gets the EEP which signifies the end of the data transfer.
The packet sink receives EEP.

The packet waiting at R1 port 2 is routed.
The blockage is cleared and packets from R1 to R2 can restart.
R1-3 continues to try to spill the input packet until the end of packet is received

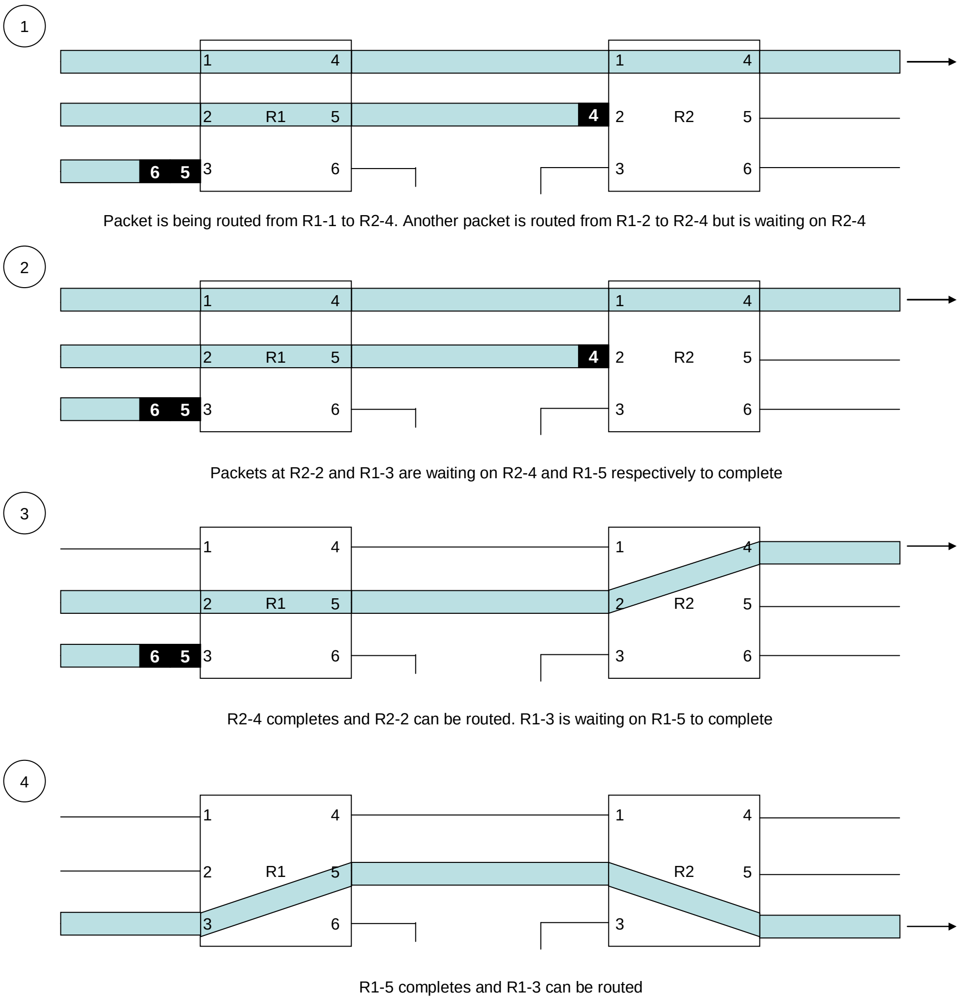
Figure 9-16 Scenario 2 Source node stalled (Watchdog timer mode)

9.3.5.5 Scenario 3 Destination node slow, Source node waiting

In this scenario the destination node slowly accepts data from the source node through the network.

- In blocking allowed mode, other packets waiting to use the network path will wait indefinitely for the slow packet to complete.
- In watchdog timer mode, packets which are waiting to use the network path will wait as long as the timeout period as they cannot be routed to the destination node. Other packets waiting to use the network path will wait as long as the timeout period.

The two figures below illustrate the actions of the router in the blocking allowed mode and the blocking watchdog timer mode.



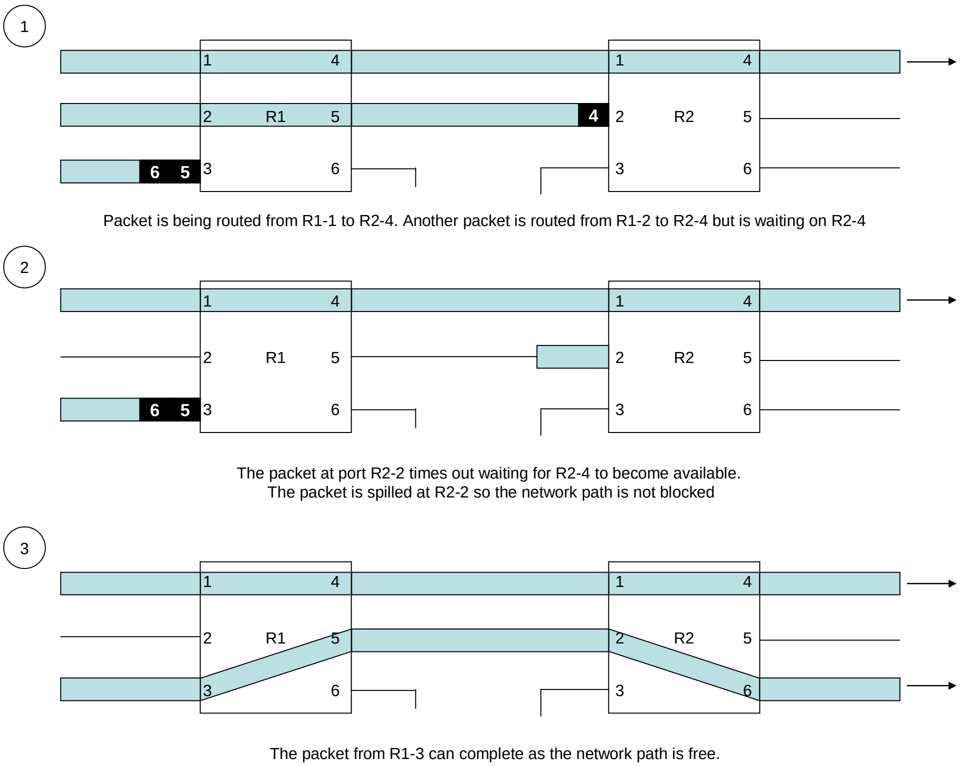
Packet is being routed from R1-1 to R2-4. Another packet is routed from R1-2 to R2-4 but is waiting on R2-4

Packets at R2-2 and R1-3 are waiting on R2-4 and R1-5 respectively to complete

R2-4 completes and R2-2 can be routed. R1-3 is waiting on R1-5 to complete

R1-5 completes and R1-3 can be routed

Figure 9-17 Scenario 3 Destination slow (Blocking Allowed Mode)



Packet is being routed from R1-1 to R2-4. Another packet is routed from R1-2 to R2-4 but is waiting on R2-4

The packet at port R2-2 times out waiting for R2-4 to become available.
The packet is spilled at R2-2 so the network path is not blocked

The packet from R1-3 can complete as the network path is free.

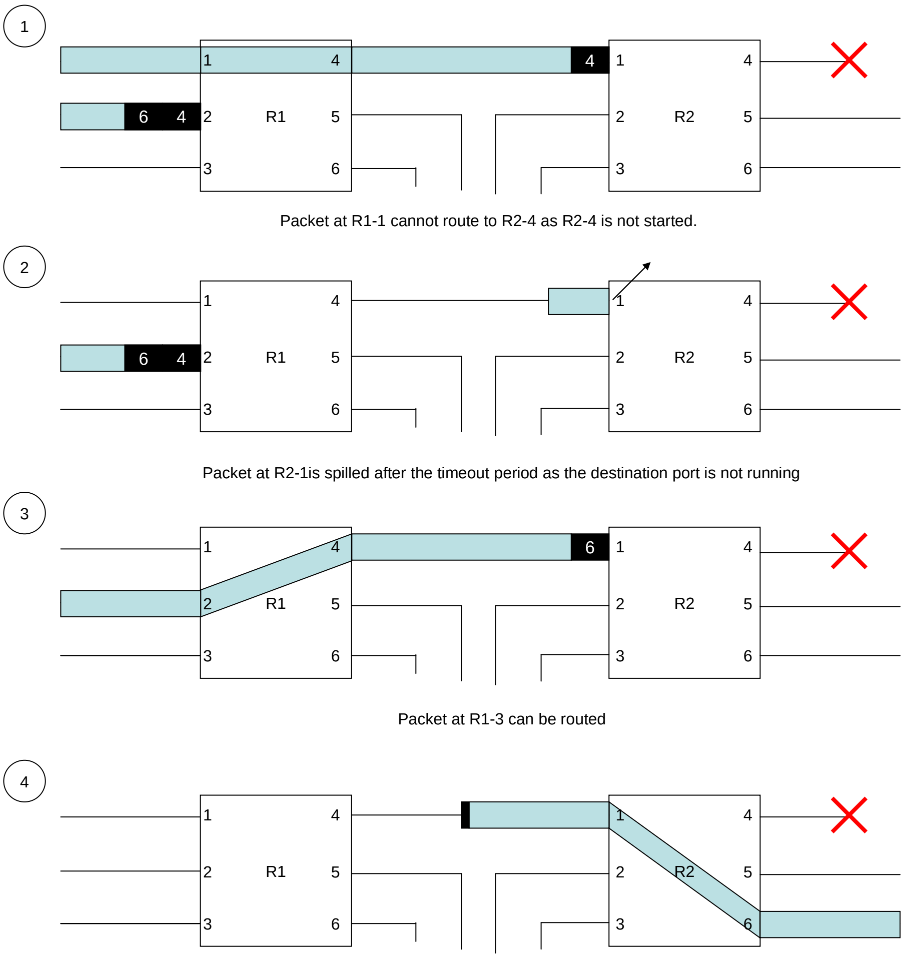
Figure 9-18 Scenario 3 Destination slow (Watchdog Timer Mode)

9.3.5.6 Scenario 4 Destination port not ready

In this scenario the destination port in the router is not ready to accept a new packet. If the destination port is an external port or the configuration port then it is not ready when the destination FIFO is full (cannot accept mode data). If the destination port is a SpaceWire port then the destination port is not ready when the destination FIFO is full (transmit FIFO) or the SpaceWire link is not started.

- In blocking allowed mode the input packet will wait indefinitely until the destination port is ready. An exception to this is when the router output port is a SpaceWire port and the port is not started, in this case the packet waits as long as the timeout period and then discards the packet. If group adaptive routing is used and at least one of the destination ports is running then the packet waits indefinitely.
- In watchdog timer mode the input packet will wait as long as the timeout period to be routed to the destination port.

The two modes of operation are illustrated in the figures below (note the identical operation when a port is not started).



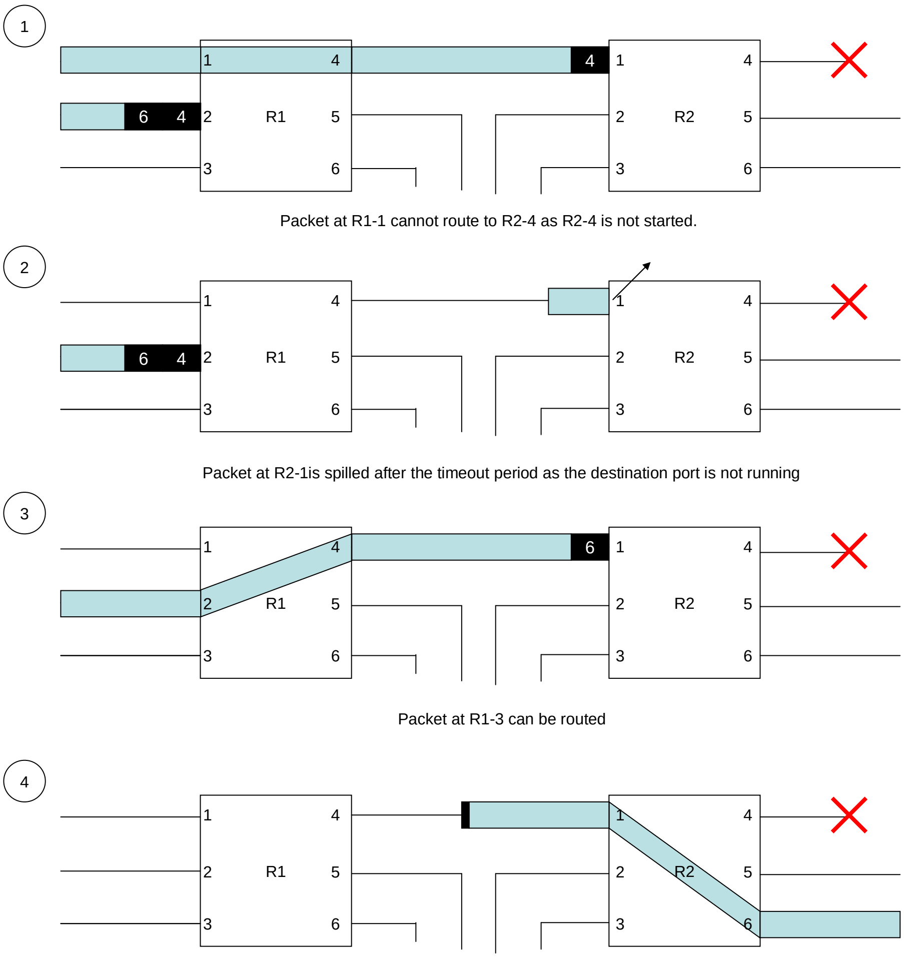
Packet at R1-1 cannot route to R2-4 as R2-4 is not started.

Packet at R2-1is spilled after the timeout period as the destination port is not running

Packet at R1-3 can be routed

Packet completes

Figure 9-19 scenario 4 Destination Port Not Ready (Blocking allowed mode)



Packet at R1-1 cannot route to R2-4 as R2-4 is not started.

Packet at R2-1is spilled after the timeout period as the destination port is not running

Packet at R1-3 can be routed

Packet completes

Figure 9-20 scenario 4 Destination Port Not Ready (Blocking watchdog timer)

10. REGISTER DEFINITIONS

This section describes the internal registers and the internal memory map of the SpaceWire Router.

The internal register size is 32 bits unless otherwise specified in the register summary. There are 263 registers in the configuration port addresses. The necessary 9-bit register address is provided by the one byte address field (A7-A0) in the command packet structure together with bit-2 of the command field (A8).

The following subsections contain register bit description tables which hold the following information:

- The bit numbers of each field
- A descriptive name for each field
- The reset value for each field
- A description of what the each field in the register is used for
- An indication of whether the field is readable and/or writeable by a configuration command.

Registers that are shorter than 32-bits or that have unused fields will return zero in all the unused bit positions when read. The unused bit positions are ignored during writing but should in any case be set to zero.

10.1 INTERNAL MEMORY MAP

The memory map for the SpaceWire Router is shown in Figure 10-1.

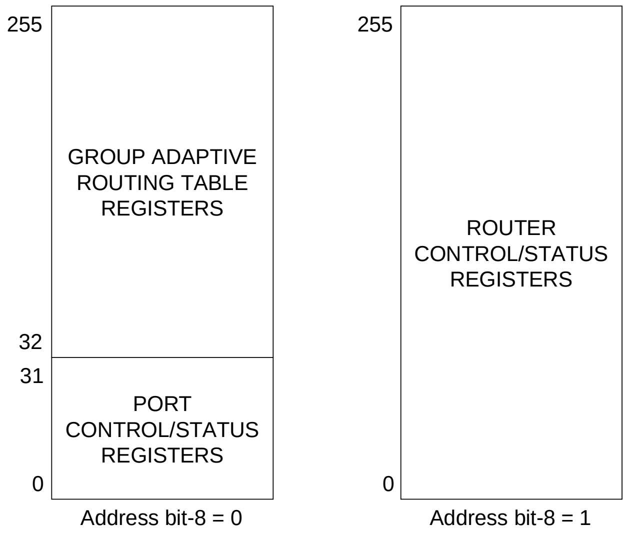

255
GROUP ADAPTIVE ROUTING TABLE REGISTERS
32
31
PORT CONTROL/STATUS REGISTERS
0
Address bit-8 = 0
255
ROUTER CONTROL/STATUS REGISTERS
0
Address bit-8 = 1

Figure 10-1 Router Internal Memory Map

The Group Adaptive Routing (GAR) registers map SpaceWire logical addresses 32-255 to the physical ports – SpaceWire ports or External ports. The link control/status registers allow the internal physical ports to report status information to a network manager and allow control bits to be set to determine the state of the physical ports. The router control/status registers allow the router management control and status information to be accessed by a network manager using configuration commands.

Table 10-1 provides an overview of each of the different type of register within the configuration port. Each register type is then described in detail in the following subsections.

Table 10-1 Types of Register within Configuration Port

Register Name	**Description**	**Address**
Group Adaptive Routing Table Registers	Allows the setting of group adaptive routing logical addresses by assigning the output ports which should be accessed when a packet is received with logical address.	32-255 (0x20 – 0xFF)
Port Control/Status Registers	Controls the Configuration, SpaceWire and External Ports. Provides the status of the ports.	0 – 31 (0x00 – 0x1F)
Router Control/Status Registers	Controls the overall operation of the router. Reports the router status.	256-265 (0x100 – 0x109)

10.2 REGISTER ADDRESSES SUMMARY

Table 10-2 lists all the registers in the configuration memory space.

Table 10-2 Configuration Register Addresses

Address	Register
31–0 (*1Fh–00h*)	Port control/status registers
255–32 (*FFh –20h*)	Group adaptive routing table registers
256 (*100h*)	Network discovery register
257 (*101h*)	Router identity register
258 (*102h*)	Router control register
259 (*103h)*	Error active register
260 (*104h)*	Time-code register
261 (*105h*)	Device Manufacturer and Chip ID register
262 (*106h*)	General Purpose register
263 (*107h*)	Time-Code Enable register
264 (*108h*)	Clock Control register
265 (109h)	Destination Key Register

10.3 GROUP ADAPTIVE ROUTING TABLE REGISTERS

The Group Adaptive Routing (GAR) table is accessed through configuration memory addresses 32-255 (20h-FFh).

The fields in the GAR registers are illustrated in Figure 10-2.

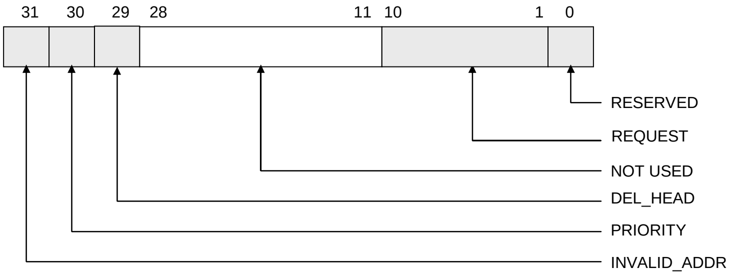


Figure 10-2 GAR Register Fields

The GAR table holds the routing table information that maps logical addresses to one or more port addresses. There is one entry (register) in the GAR table for each possible logical address. The configuration memory address range of the GAR table is 32-255 (20h – FFh). The configuration memory address corresponds to the logical address; hence the GAR table entry at address 39 corresponds to logical address 39.

The logical address to port mapping is held in the REQUEST field. Each bit in this field represents a physical output port; thus up to 28 possible output ports can be specified using the GAR register. When a bit is set in the REQUEST field packets may be routed to the corresponding output ports. The port number corresponds to the bit position in the GAR register. For example, if configuration memory address 39 has bit 2 set then a packet with logical address 39 may be routed out of port 2. If bits 2 and 4 are both set then the packet may be routed out of either port 2 or 4. Port 0, the configuration port, is a special port which can only be accessed using address 0 so this bit position in the GAR table registers is reserved and will always be set to zero.

Note that the SpaceWire ports are port numbers 1 to 8 and the External ports are ports 9 and 10.

The DEL_HEAD bit, when set, causes the leading byte (header) of a packer to be deleted.

The PRIORITY bit determines the priority of the logical address when packets waiting at two input ports wish to use the same output port (1 = high priority, 0 = low priority).

The INVALID_ADDR bit is set to indicate that the corresponding logical address is not valid.

Table 10-3 describes each field in the GAR register.

Table 10-3 GAR Table Register Description				
Address Range: 32-255 (20h – FFh)				
Bits	**Name**	**Reset Value**	**Description**	**Read/Write**
0	RESERVED	'0'	Reserved bit – always set to zero.	R
10:1	REQUEST	All bits set to zero.	The request bits determine which output ports the logical address will arbitrate for. When bit one is set then SpaceWire port one will be requested. When bit two is set then SpaceWire port two will be accessed and so on. By setting more than one bit then group adaptive routing can be used by allowing the input packet to arbitrate for more than one output port. If a write is performed and bits 10:1 are set to zero then the INVALID_ADDR bit will be set and all other bits will be set to zero Note: The configuration port (port 0) is not accessible through logical addresses.	R/W
28:11	NOT USED	-	-	-
29	DEL_HEAD	'0'	Delete header: when set the leading header byte of the input packet will be removed before it is transferred to the output port.	R/W
30	PRIORITY	'0'	When a packet addresses a logical address with the Priority bit set then the packet will arbitrate for output ports with higher priority than packets with Priority bit set to zero.	R/W
31	INVALID_ADDR	'1'	When the Invalid Address bit is set it indicates that the corresponding logical address is invalid. In this case, any packets arriving at the router with an invalid address are spilt and an address error is reported in the port status register.	R/W

10.4 PORT CONTROL/STATUS REGISTERS

The port control/status registers address range is 31-0 (*1Fh – 0*)

The port control/status registers provide the means to configure and control the ports of the router and for reading the status of each port. There is a port control/status register for each SpaceWire port, each External port and for the configuration port. The address in configuration memory space of a port control/status register reflects the physical address of the port. For example, the register for port 0, the configuration port, is at address 0, and the register for a SpaceWire port number 3 is at address 3. Each port control/status register is a 32-bit register.

The fields within the port control/status register depend on the type of port that it is attached to. All port control/status registers have fields for port type and current port connection. These generic fields will be described first followed by the specific fields for the configuration port, SpaceWire ports and External ports.

Port control/status register bits 31:24 are generic to all ports. Register bits 23:0 are specific to the type of port to which the register is attached.

10.4.1 Generic port control/status register fields.

The configuration port control/status fields are described in Table 10-4.

Table 10-4 Configuration Port Control/Status Register Fields

Bits	Name	Reset Value	Description	Read/Write
28:24	Current port connection	All bits set to one.	The current port connection bits indicate the input port which this output port is currently connected to. Port number 31 (bits 28:24 set to 11111) means that there is no port currently connected to the input port. This is the reset condition.	R
31:29	Port Type	All bits set to zero.	Indicates the port type which corresponds the port number of this port register. The port types are listed below. “000” – Configuration port. “001” – SpaceWire port. “010” – External port.	R

10.4.2 Configuration port control/status register fields.

The configuration port control/status fields specific to the configuration port are described in Table 10-5. Any errors occurring in the configuration port are reported via status bits in this register and the configuration command that caused the error is replied to with a NACK (i.e. the status byte in the reply contains an error code so is non-zero).

Note: these bits are cleared by writing to the Error Active register, see section 10.5.4.

Table 10-5 Configuration Port Control/Status Register Fields

Bits	Name	Reset Value	Description	Read/Write
0	Error active	‘0’	The error active bit is set when one of the error bits is active	R
1	Port timeout error	‘0’	The port timeout error bit is set when a timeout event is detected by the configuration port routing logic. A timeout event can occur as specified in F4-4.	R
2	Invalid Header CRC	‘0’	The Invalid header CRC bit is set when the header CRC is invalid.	R
3	Invalid Data CRC	‘0’	The invalid data CRC is set when the data part of the packet is corrupted and the CRC does not match the internally generated CRC.	R
4	Invalid Destination Key	‘0’	The invalid destination key bit is set when the destination key in the command packet is invalid.	R
5	Command not implemented	‘0’	The command not implemented bit is set when the command code is a valid RMAP code but the command is not supported by the SpaceWire Router.	R
6	Invalid Data Length	‘0’	The invalid data length bit is set when a data length error is detected	R
7	Invalid RMW Data Length	‘0’	The read modify write command data length is invalid. When a read modify write is performed the expected data length is 8.	R
8	Invalid Destination Logical Address	‘0’	The invalid destination logical address bit is set when the destination logical address in the command packet is not the default value of 254.	R
9	Early EOP	‘0’	The early EOP bit is set when the command packet is terminated before the end of packet with an EOP	R
10	Late EOP	‘0’	The late EOP bit is set when the command packet is not terminated correctly and trailing bytes are detected before the end of packet.	R
11	Early EEP	‘0’	The early EEP bit is set when the command packet is terminated before the end of packet with an EEP	R
12	Late EEP	‘0’	The late EEP bit is set when the command packet is not terminated correctly and trailing bytes are detected before the end of packet.	R
13	Verify Buffer Overrun Error	‘0’	The verify buffer overrun error bit is set when a verified write command is performed and the data length is not 4.	R

14	Invalid Register Address	‘0’	The invalid register address bit is set when an unknown register address is given in the command packet or a write is attempted to a read only register	R
15	Unsupported protocol error	‘0’	The unsupported protocol error bit is set when a command packet is received with a protocol identifier which is not the RMAP protocol identifier of 01h.	R
16	Source logical address error	‘0’	The source logical address error bit is set when an invalid source logical address is received.	R
17	Source Path Address Error	‘0’	The source path address sequence is invalid.	R
18	Cargo too large	‘0’	The RMAP command packet is too large	R
19	Unused RMAP command or packet type	‘0’	The command code is an unused command code or the packet type is invalid as specified in F1-8.	R
23:20	Not Used	-	-	-

10.4.3 SpaceWire link interface port control/status register bits.

The port control/status fields specific to SpaceWire ports are shown in Figure 10-3 and Table 10-6.

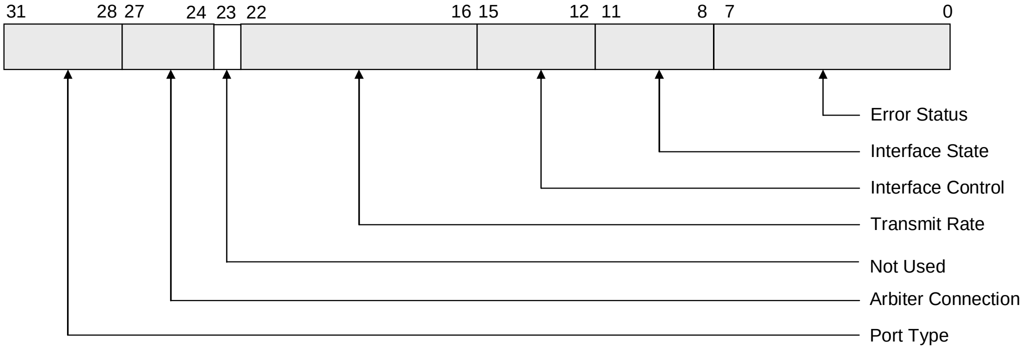


Figure 10-3 SpaceWire Port Control/Status Register Fields

Note: these bits are cleared by writing to the Error Active register, see section 10.5.4.

Table 10-6 SpaceWire Port Control/Status Register Fields.				
Bits	**Name**	**Reset Value**	**Description**	**Read/Write**
0	Error active	‘0’	The error active bit is set when one of the error bits are set.	R
1	Packet address error	‘0’	The packet address error bit is set when a packet is received with an incorrect address.	R
2	Output port timeout error	‘0’	The output timeout error bit set when the output port has become blocked for a period of time.	R
3	Disconnect error	‘0’	The disconnect error bit is set when a disconnect error occurs on the SpaceWire link	R
4	Parity error	‘0’	The parity error bit is set when a parity error occurs on the SpaceWire link	R
5	Escape error	‘0’	The escape error bit is set when an escape error occurs on the SpaceWire link	R
6	Credit error	‘0’	The credit error bit is set when a credit error occurs on the SpaceWire link	R
7	Character sequence error	‘0’	The character sequence error bit is set when a character sequence error occurs on the SpaceWire link	R
10:8	Interface state	“000”	The interface state bits indicate the state of the interface state machine in the SpaceWire link. “000” = Error Reset “001” = Error Wait “010” = Ready “011” = Started “100” = Connecting “101” = Run	R
11	Running	‘0’	The running bit is set when the SpaceWire interface state machine is in the Run state.	R
12	AutoStart	‘1’	When set the SpaceWire link will autostart as defined in the SpaceWire standard [AD1]: the SpaceWire port will wait until the other end of the link tries to make	R/W

			a connection and will then automatically start.	
13	Start	'0'	When set then the SpaceWire link will initiate start-up as defined in the SpaceWire standard [AD1]: the SpaceWire port will try to make a connection with the other end of the link.	R/W
14	Disable	'0'	When set then the SpaceWire link will be disabled as defined in the SpaceWire standard [AD1]: the SpaceWire port will not start and will not respond to any attempt to make a connection by the other end of the link.	R/W
15	Tri-state	'0'	When set the DOUT and SOUT serial SpaceWire signals will be reset dependent on the state of the SpaceWire link interface. The DOUT and SOUT tri-state output state mappings are listed below: ErrorReset → Tri-state ErrorWait → Tri-state Ready → Tri-state Started → Enabled Connecting → Enabled Run → Enabled Note: DOUT is tri-stated one system clock cycle before SOUT to ensure simultaneous edges due to tris-state do not occur.	R/W
22:16	Transmitter data signalling rate (TXRATE)	Bits 18 to 16 are set according to the POR_TX_RATE(2:0) pins. Bits 22:18 are set to zero.	Allows the SpaceWire link data signalling rate to be set. The transmit rate is $\frac{TxClkFreq * 2}{TXRATE + 1}$ where TxClkFreq is the output frequency of the transmit clock PLL. Note: bits 22:16 must not be set to all ones. Bits 22:16 must not be set to give a transmit rate of less than 2 Mbits/s.	R/W
23	Not Used	-	-	-

10.4.4 External port control/status register bits.

The port control/status fields specific to the External port are described in Table 10-7.

Table 10-7 External Port Contol/Status Fields

Bits	Name	Reset Value	Description	Read/Write
0	Error Active	'0'	This bit is set to one when any of the error bits are set.	R
1	Packet Address Error	'0'	The packet address error bit is set when a packet is received with an incorrect address. A packet address error is also generated when an empty packet is input to the external port.	R
2	Output port timeout error	'0'	The output timeout error bit is set when the output port has become blocked for a period of time.	R
3	Input Buffer Empty	'0'	The external port input buffer is empty. Note: The input buffer writes data to the SpaceWire router.	R
4	Input Buffer Full	'0'	The external port input buffer is full.	R
5	Output Buffer Empty	'0'	The external output port buffer is empty. Note: The output buffer writes data to the external device connected to the external port.	R
6	Output Buffer Full	'0'	The external output port buffer is full.	R
23:5	Not used	-	-	-

Note: these bits are cleared by writing to the Error Active register, see section 10.5.4.

10.5 ROUTER CONTROL/STATUS REGISTERS

The router control/status registers are described below.

10.5.1 Network Discovery Register

The network discovery register address is 256 (*100h*).

The network discovery register allows a network manager to determine the layout of the network by reading the contents of the register. Its fields are shown in Figure 10-4 and described in Table 10-8.

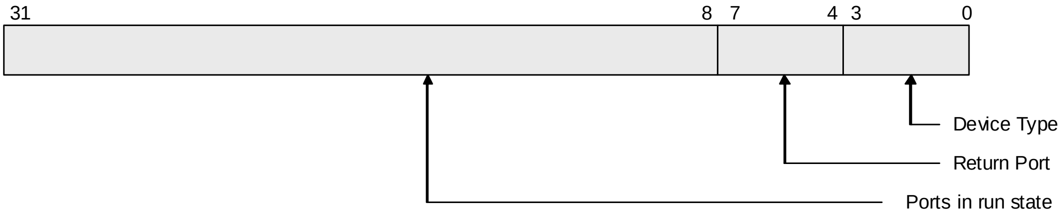


Figure 10-4 Network Discovery Register Fields

Table 10-8 Network Discovery Register Fields

Bits	Description	Reset Value	Usage	Read/Write
3:0	Device Type	“0001”	The device type field indicates the type of device which is associated with this network discovery register. At present there is only one device type defined, the Router, along with the unknown device type. “0000” – Unknown Device “0001” – Router	R
7:4	Return Port	All bits set to zero	Indicates the input port number which accessed this network discovery register.	R
31:8	Ports in Run state	All bits set to zero	Indicates the SpaceWire ports which are in the run state. Bit 8 corresponds to SpaceWire port 1. The external ports are the highest numbered port and the corresponding register bits are set to one. In this way a network manager can determine the number of ports and the active ports in one register read.	R

10.5.2 Router Identity Register

The router identity register address is 257 (*101h*).

The router identity register allows a network manager to assign a 32 bit ID to a SpaceWire router device by writing to this register. It may also be used for other purposes. The router identity register is described in Table 10-9.

Table 10-9 Router Identity Register Field

Bits	Description	Reset Value	Usage	Read/Write
31:0	Router Identity	All bits set to zero	A 32-bit read/write register which may be used to hold a unique router identity code for each router in a network.	R/W

10.5.3 Router Control Register

The router control register address is 258 (*102h*).

The router control register sets various control bits in the SpaceWire router. Router functions which can be controlled are:

- Request an output port to initiate start-up when an input packet addresses the port and the output port is not ready to receive data.
- Disable an input/output port when no activity is detected on the port for the timeout period duration.
- Enable output port timeouts which request the output port to flush when a packet becomes blocked for a timeout period.
- Enable the a router ports to address themselves i.e. provide a loop-back capability.

The router control register fields are shown in Figure 10-5 and described in Table 10-10.

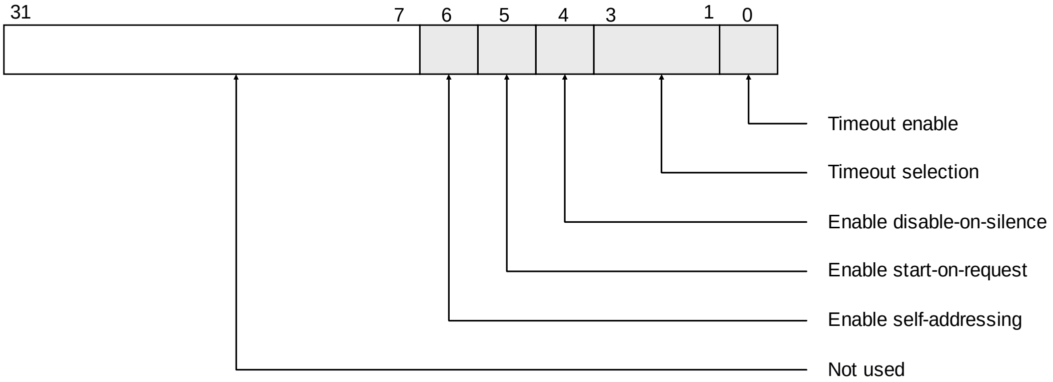


Figure 10-5 Router Control Register Fields

Table 10-10 Router Control Register Fields				
Bits	**Name**	**Reset Value**	**Description**	**Read/ Write**
0	Watchdog Timer Mode / Blocking Allowed Mode	Set by the input signal **POR_TIMEOUT_EN_N**. When '0' – Watchdog Timer Mode When '1' – Blocking Allowed Mode	When set to '1' then the watchdog timer mode is enabled. When set to '0' then the blocking allowed mode is enabled.	R/W
3:1	Timeout Selection	Set by the input signal **POR_SEL_TIMEOUT0_N**. When 0, timeout period is 100 us. When 1, timeout period is 1 ms.	Selects the blockage timeout period. Values as below: "000" => 60-100 us (N = 2) "001" => ~1.3 ms (N = 6) "010" => ~10 ms (N = 9) "011" => ~82 ms (N = 12) "100" => ~1.3 s (N = 16) "101" => ~1.3 s (N = 16) "110" => ~1.3 s (N = 16) "111" => ~1.3 s (N = 16) The actual value of the timeout period is given by: 200 x (2^N) x TCLK +/- 200 x TCLK where TCLK is the period of the 10 MHz clock signal.	R/W
4	Enable disable on silence	Set by the input signal **POR_DISABLE_ON_SILENCE_N**.	When set the corresponding port will be disabled if no activity is detected over the timeout period. The port will only be disabled if the port was initially started by the router using the start on request bit. If an external device starts the link then the port will not be disabled. Events which cause the disable on silence timeout to be reset are • Input port data read.	R/W

			- Output port data write.	
5	Enable start on request	Set by the input signal **POR_START_ON_REQ_N**.	When set the arbiter will request the output port to start-up if an input port receives a packet which is destined for the output port. Note: if the output port link disable bit is set then the link will not start.	R/W
6	Enable Self Addressing	Set by the input signal **POR_ADDR_SELF_N**	When set input ports are permitted to address themselves. If this bit is not set and a packet is to be routed through the same port then an address error is reported and the packet is discarded. When this bit is not set and a group adaptive routing packet is received (a packet which can be routed through two or more ports, dependent on the group adaptive routing table contents) which can be routed through the port it arrived on then the packet is routed through one of the other ports and not the port on which the packet arrived on. An address error is not reported.	R/W
31:7	Not Used	All bits set to zero	-	R

10.5.4 Error active Register

The error active register address is 259 (*103h*).

The error active register indicates the Error Active bit of the port control/status registers. By reading from this register a network manager can determine which ports currently have errors. This register is also used to clear the error bits of the port control/status registers. To do this a write command is sent to the error active register with bits set for those errors that are to be cleared.

The error active register fields are shown in

Figure 10-6 and described in Table 10-11.

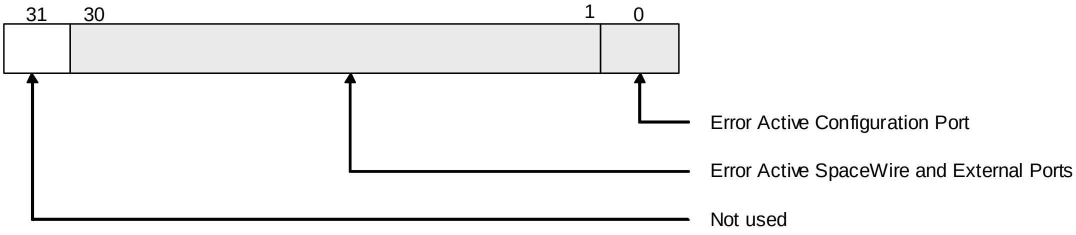

31
30
1
0
Error Active Configuration Port
Error Active SpaceWire and External Ports
Not used

Figure 10-6 Error Active Register Fields

Table 10-11 Error Active Register Fields

Bits	Name	Reset Value	Description	Read/Write
0	Configuration Port Error Active	‘0’	Indicates that the Error Active bit in the configuration port is asserted. A write to this register with bit 0 set will clear all the error flags in the configuration port control/status register.	R/W
30:1	SpaceWire and External Port Error Active	All bits set to zero	Indicates that the Error Active bit in the corresponding number SpaceWire or External port is asserted. A write to this register with one or more of these bits set will clear all the error flags in the corresponding SpaceWire and External port control/status registers. When writing to this register with all bits set all error flags in all port control/status registers are cleared.	R/W
31	Not used	All bits set to zero	Not used because there are a maximum of 30 SpaceWire / External ports supported by the router design.	R

10.5.5 Time-Code Register

The time-code register address is 260 (*104h*).

The time-code register contains the current value of the internal time-code register. Its fields are shown in Figure 10-7and described in Table 10-12.

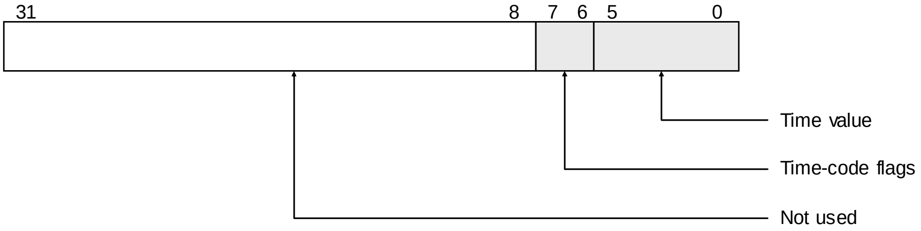


Figure 10-7 Time-Code Register Fields

Table 10-12 Time-Code Register Fields				
Bits	**Name**	**Reset Value**	**Description**	**Read/Write**
5:0	Time Value	All bits set to zero	6-bit time-code value	R
7:6	Time-Code Flags	“00”	Two time-code flags	R
31:8	Not used	All bits set to zero		R

10.5.6 Device Manufacturer and Chip ID Register

The device manufacturer and chip ID register address is 261 (*105h*).

This register contains three eight-bit fields which hold a device manufacturer identity, chip identity and version number. The fields of the device manufacturer and chip ID register are shown in Figure 10-8 and described in Table 10-13.

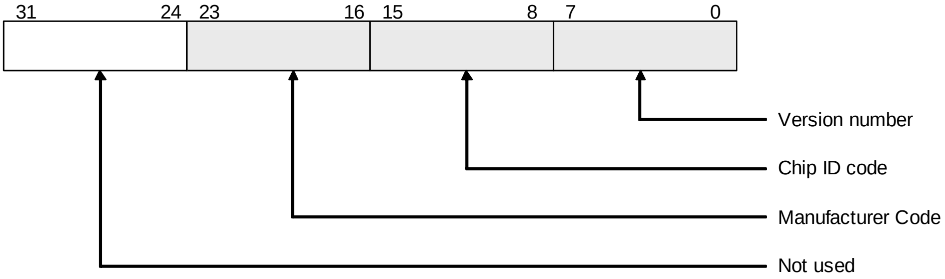


Figure 10-8 Device Manufacturer and Chip ID Register Fields

Table 10-13 Device Manufacturer and Chip ID Register Fields

Bits	Name	Reset Value	Description	Read/Write
7:0	Version Number	Version Number of chip design	Version number of the chip design	R
15:8	Chip ID Code	Chip type	Identity code for the SpaceWire chip from the particular manufacturer	R
23:16	Manufacturer Code	Manufacturer identity code	Manufacturer identity code “00000000” = Unknown Manufacturer “00000001” = University of Dundee	R
31:24	Not used	All bits set to zero		R

10.5.7 General Purpose Register

The general purpose register address is 262 (*106h*).

The general purpose register contains 32-bits and may be set by a configuration write command to a user defined value as required. It may also be read with a configuration read command. The general purpose register has no effect on the operation of the router.

The least-significant 8-bits of the general purpose register are available on the multiplexed status pins, see section 5.3.

10.5.8 Time-Code Enable Register

The time-code enable register address is 263 (*107h*).

The time-code enable register enables the passing of time-codes out of individual ports on the router. Bits 1 to 8 of the time-code enable register are used to enable time-code distribution through SpaceWire ports 1 to 8 respectively. If one of these bits is set to 1 then the corresponding SpaceWire port is enable for time-code distribution and will send out a time-code when one is received by the router. For example, if bit-1 in the enable register is set to 1, time-codes are passed to SpaceWire port 1, whereas if bit-1 is set to 0, time-codes are not passed to SpaceWire port 1.

Bit-9 of the time-code enable register controls time-code distribution to the external time-code interface in a similar manner. Note that there is only one external time-code interface although there are two External ports.

The fields of the time-code enable register are shown in Figure 10-9 and described in Table 10-14.

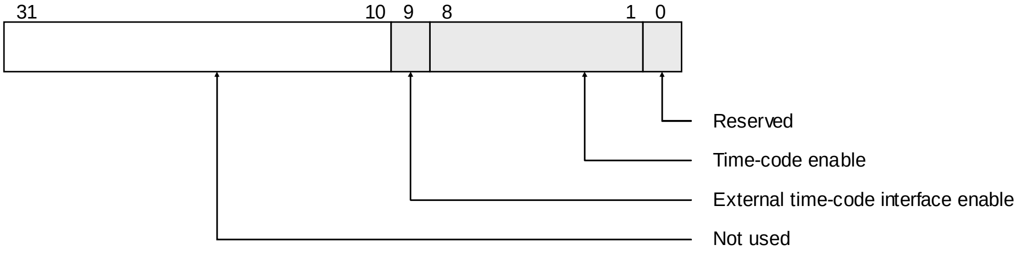


Figure 10-9 Time-Code Enable Register Fields

Table 10-14 Time-Code Enable Register Fields

Bits	Name	Reset Value	Description	Read/Write
0	Reserved	0	Reserved bit	R
8:1	SpaceWire Time-Code Enable	0	Time-code distribution enable bits for SpaceWire ports 8 to 1 respectively	R/W
9	External Time-Code Interface Enable	1	Time-code distribution enable for External time-code port	R/W
31:10	Not used	All bits set to zero		R

10.5.9 Transmit Clock Control Register

The transmit clock control register address is 264 (*108h*).

The transmit clock control register is shown below. Bits 1 to 0 are used to determine the output divide ratio for the transmit clock internal PLL. Bits 15 to 8 are used to stop the transmitter clocks of SpaceWire interfaces that are not being used to save power. Bits 20 to 16 are used to set the default 10Mbits/s transmit data rate.

Note: The transmit clock should not be disabled when an output port is sending data or when the interface is in the run state. The port control/status registers can be used to determine if an output port is currently connected to an input port and therefore transferring data.

WARNING: If the link that is being used to configure the router is has its transmit clock turned off then it will not be possible to configure the router, unless there is another connection with an active clock and which is not disabled that can be used.

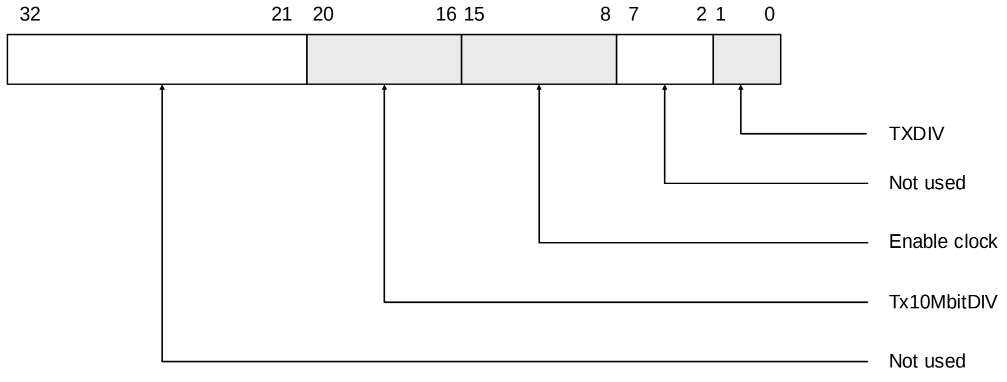


Figure 10-10 Transmit clock control register

The transmit clock control register

Table 10-15 Transmit Clock Control Register Bits				
Bits	**Name**	**Reset Value**	**Description**	**Read/Write**
1:0	TXDIV	“01”	Transmit clock internal PLL output divider. Selects the divided output from the transmit clock as follows “00” → divide by 2 “01” → divide by 4 “10” → divide by 8 “11” → divide by 8 **Example**: If the PLL output frequency is 200MHz (set by FEEDBDIV, see section 4.1) and TXDIV = 01 then the transmit clock frequency will be 50MHz and the transmit data rate will be 100Mbit/s	R/W
7:2	Not used	All bits set to zero	-	R
15:8	Enable clock	All bits set to 1	Enable the transmit clock trees. Setting a bit to zero disables the transmit clock for the corresponding interface, i.e. setting bit zero to zero causes the transmit clock for SpaceWire link one to be stopped.	R/W
20:16	Tx10MbitDIV	Dependent on **FEEDBDIV** at reset **FEEDBDIV=**“000” → “00100” **FEEDBDIV=**“001” → “00101” **FEEDBDIV=**“010” → “00110” **FEEDBDIV=**“011” → “00111” **FEEDBDIV=**“100” → “01000” **FEEDBDIV=**“101” → “01001” **FEEDBDIV=**“110” → “01001” **FEEDBDIV=**“111” → “01001”	Set the default 10Mbit/s data rate divider value. The Tx10MbitDIV value should be set as shown in Figure 10-11.	R/W
31:21	Not used	All bits set to zero	-	R

$$Tx10MbitDIV = \left(\frac{\left(\frac{PLLFreq}{2^{TXDIV+1}} \right)}{5MHZ} \right) - 1$$

Figure 10-11 Tx10MBitDIV equation

Note: If Tx10MbitDIV is an odd number then the duty cycle on DOUT and SOUT will not be 50/50.

10.5.10 Destination Key Register

The Destination Key register address is 265 (*109h*).

The destination key register fields are listed in the table below.

Table 10-16 Destination Key Register

Bits	Name	Reset Value	Description	Read/Write
7:0	DESTKEY	20h	The destination key is checked when a security key is required for RMAP configuration packets to access the router registers.	R/W
31:8	Not used	All bits set to zero	-	

10.5.11 Unused Registers and Register Bits

If an unused register address is referenced in a configuration command then the command will not be acted upon and a NACK will be sent in the reply to the command.

All unused bits in valid configuration registers will return ‘0’ when read.

10.5.12 Empty packets

An empty packet received at the configuration port are discarded by the configuration port and no reply packet is sent. An empty packet has no address or cargo bytes and consists only of an EOP.

10.6 WRITING TO A READ-ONLY REGISTER

If a write command is sent with a register address that corresponds to a register whose entire contents is read only, then an error will be generated. See section 10.4.2.

11. DOCUMENT CHANGES

11.1 ISSUE 1.7 TO ISSUE 2.0

Section	Ref	Change
All		Final updates and editorial corrections before release

11.2 ISSUE 1.6 TO ISSUE 1.7

Section	Ref	Change
All		Corrections added following validation

11.3 ISSUE 1.5 TO ISSUE 1.6

Section	Ref	Change
8		RMAP section added

11.4 ISSUE 1.4 TO ISSUE 1.5

Section	Ref	Change
All		Footer changed to indicate preliminary and a note added on the front page to indicate that section 8.6 is subject to change

11.5 ISSUE 1.3 TO ISSUE 1.4

Section	Ref	Change
7		Latency and jitter specifications added

11.6 ISSUE 1.2 TO ISSUE 1.3

Section	Ref	Change
8		Section on fill bytes added
10		Registers updated to 3.3 specification document

11.7 ISSUE 1.1 TO ISSUE 1.2

Section	Ref	Change
6.3, 6.4, 6.5	Table 6-3, Table 6-4, Table 6-5	FPGA timing data added

11.8 ISSUE 1.0 TO ISSUE 1.1

Section	Ref	Change
4.1	Table 4-1	FEEDBDIV PLL clock settings section added
4.5	Table 4-5	**STAT_MUX_OUT** changed to multi function pin
4.6	Table 4-6	Power on reset signals mapped to **STAT_MUX_OUT** pins
9.1.6		Setting the data rate takes account of **FEEDBDIV** and transmit clock control register setting **TXDIV**
9.3.5		Packet blocking correction does not cause disconnection of links.