SpeedChip XRS7000 and XRS3000 Series

User Manual

The XRS7004, XRS7003 and XRS3003 are HSR and PRP (IEC 62439-3 Clause 5 & 4) enabled single-chip gigabit Ethernet switches. XRS7003 and XRS3003 can be employed in HSR and PRP End-nodes and XRS7004 in both End-Nodes and HSR and PRP RedBoxes. A QuadBox can be built using two XRS7004 devices.

Features

- 144-pin Plastic Enhanced Quad Flat Pack (EQFP) (22mm x 22mm, 0.5 mm pitch) or 256-pin Fine Ball Grid Array (FBGA) Package (17mm x 17mm, 1.0 mm ball pitch)
- Industrial Temperature Range -40°C to +100°C
- Two (XRS7003, XRS3003) or three (XRS7004) 10/100/1000 Mbit/s Full-Duplex Ethernet interfaces (RGMII)
- 10/100 Mbit CPU port (RMII)
- Gigabit wire speed forwarding capacity, non-blocking
- I2C and MDIO interfaces for register access (only MDIO in XRS3003)
- Cut-through and Store-and-Forward operation
- Quality of Services (QoS) with four priority queues per port
- Per port packet filtering
- VLAN tagging (not in XRS3003)
- Priority tagging (not in XRS3003)
- IEEE 1588 Precision Time Protocol (PTP) support with internal Real-Time Clock (RTC)
- HW counters for implementing Remote Network MONitoring (RMON) SNMP MIB (not in XRS3003)
- Support for MAC address based authentication methods
- Support for Spanning Tree Protocol (STP) and Rapid Spanning Tree Protocol (RSTP) implementations
- PPS (Pulse per Second) input and output (only output in XRS3003)

High-availability Seamless Redundancy and Parallel Redundancy Protocol

HSR and PRP protocols are used in applications that require short reaction time and high availability. Typical applications include smart grid electrical substation automation and other critical networking applications such as industrial automation, motion control, vehicle and military communication. HSR and PRP provide a network that has no single point of failure and zero recovery time in case of a failure: Single network faults will not result in any frame loss. The network is fully operational even during maintenance as any network device can be disconnected and replaced without breaking the network connectivity.

Table 1. Device Features

Feature	XRS7004E	XRS7004F	XRS7003E	XRS7003F	XRS3003F
10/100/1000 Mbit/s RGMII ports	3	3	2	2	2
10/100 Mbit/s RMII ports	1	1	1	1	1
High-Availability Seamless Redundancy (HSR)	Yes	Yes	Yes	Yes	Yes
Parallel Redundancy Protocol (PRP)	Yes	Yes	Yes	Yes	Yes
Precision Time Protocol (PTP)	Yes	Yes	Yes	Yes	Yes
Register Access	MDIO, I^2C	MDIO, I^2C	MDIO, I^2C	MDIO, I^2C	MDIO only
Queues per port	4	4	4	4	4
Maximum number of VLANs	4096	4096	4096	4096	No VLAN support
Recommended HSR network size	Up to 512 hops	Up to 512 hops	Up to 512 hops	Up to 512 hops	Up to 512 hops
HSR proxy node table size	512 MAC Addresses	512 MAC Addresses	64 MAC Addresses	64 MAC Addresses	1 MAC Address
Package	EQFP144	FBGA256	EQFP144	FBGA256	FBGA256
Operating Junction Temperature range	-40°C to +100°C	-40°C to +100°C	-40°C to +100°C	-40°C to +100°C	-40°C to +100°C

Contents

Figures

Tables

Revision History

Rev	Date	Comments
1.0	20.11.2015	First release
1.1	15.4.2016	XRS3003 and FBGA256 package added. JTAG interface removed.
1.2	11.5.2016	Interface timing changes: - I2C input data setup time min changed from 0 ns to 50 ns - RMII output delay max changed from 12 ns to 13 ns - Power supply ramp time max changed from 50 ms to 3 ms RGMII port indexes changed from 0…2 to 1…3.
1.3	6.11.2017	Added register bit for enabling/disabling support for independent VLANs. (Bit 7 in GENERAL register)

1. CONVENTIONS USED IN THIS DOCUMENT

Signal names are written in this document with `SignalName` style. Block names are written with Capital first letter.

Register descriptions in this document follow these rules: Unless otherwise stated, all the bits that activate or enable something are active when their value is 1 and inactive when their value is 0. The explanation of register bit types is the following:

- RO = Read Capable Only. The bits marked with RO can be read. Writing to these bits is allowed unless otherwise stated. If writing is allowed, it does not affect the value of the bit.
- R/W = Read and Write capable. The bits can be read and written. Writing 1 to the bit makes its value 1. Writing 0 to the bit makes its value 0.
- R/C = Read and Clear capable. The bits can be read and cleared. Writing 0 to the bit makes its value 0. Writing 1 does nothing.
- R/SC = Read, Write and Self Clear. The bits can be read and written. Writing 0 to the bit does nothing. Writing 1 to the bit makes its value 1 for a while, but after that the value automatically returns back to 0.

The bits marked as *Reserved* should not be written anything but 0, even if they are marked as read capable only, because their function may change in future versions.

2. TYPICAL APPLICATIONS

Typical applications for XRS7004, XRS7003 and XRS3003 are presented in Figure 1, Figure 2, Figure 3 and Figure 4.

Figure 1. End-Node Application

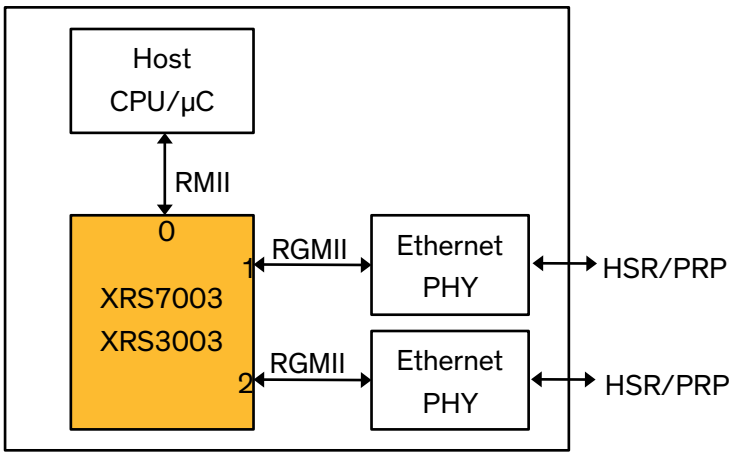


Figure 2. RedBox Application

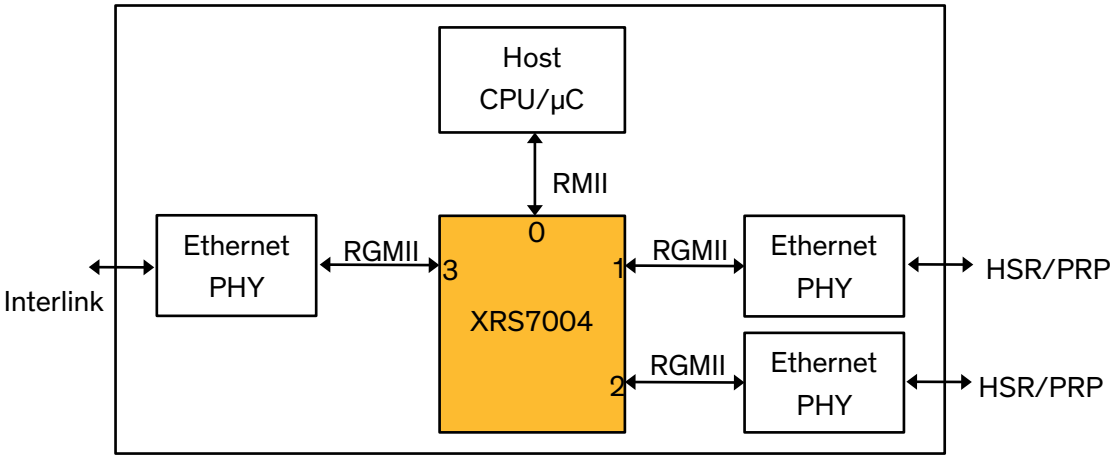


Figure 3. Multiport RedBox Application

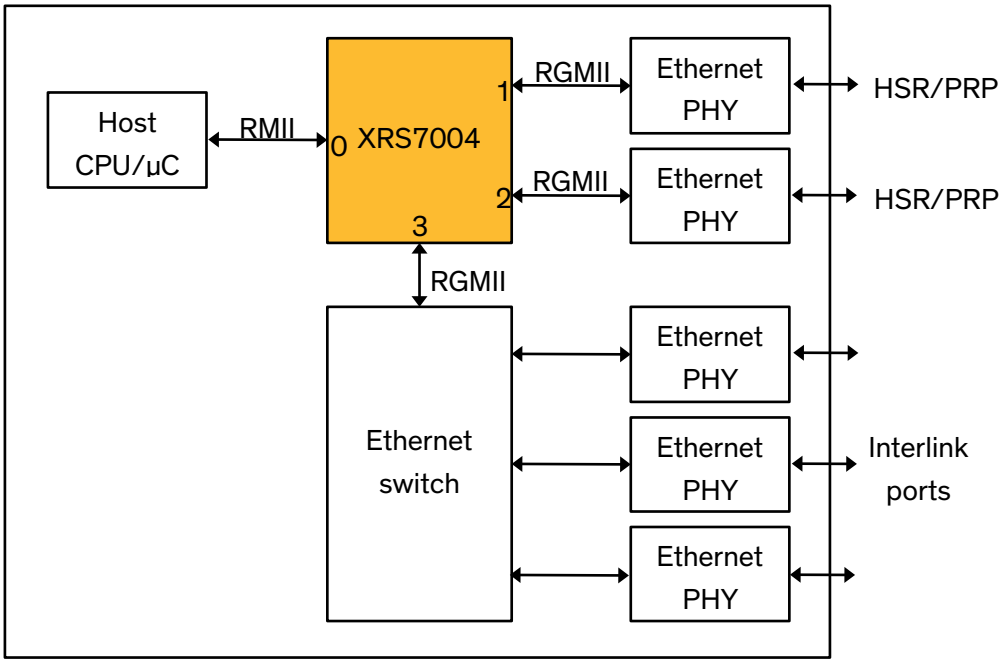
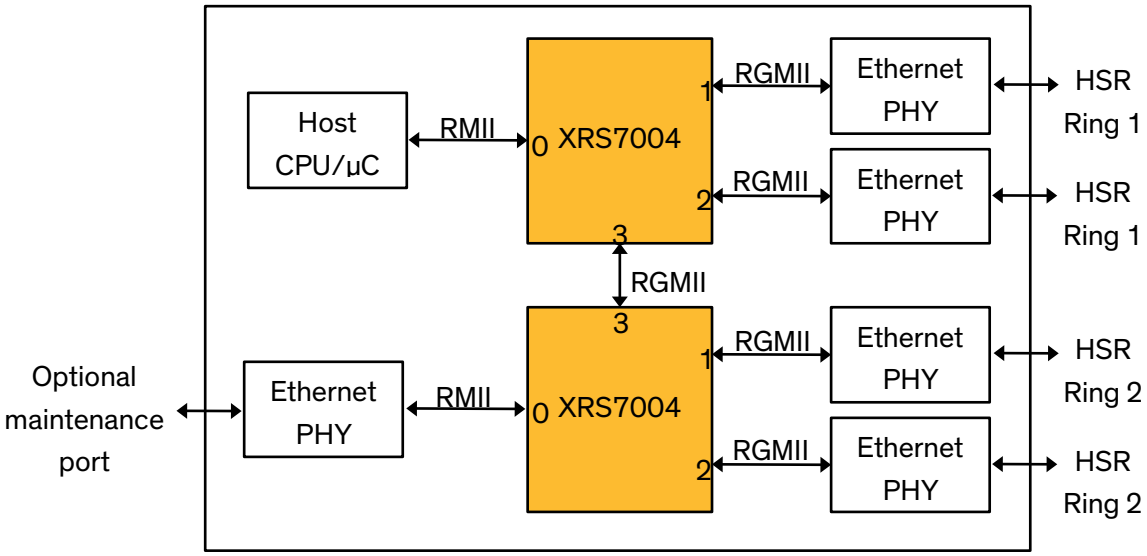


Figure 4. QuadBox Application

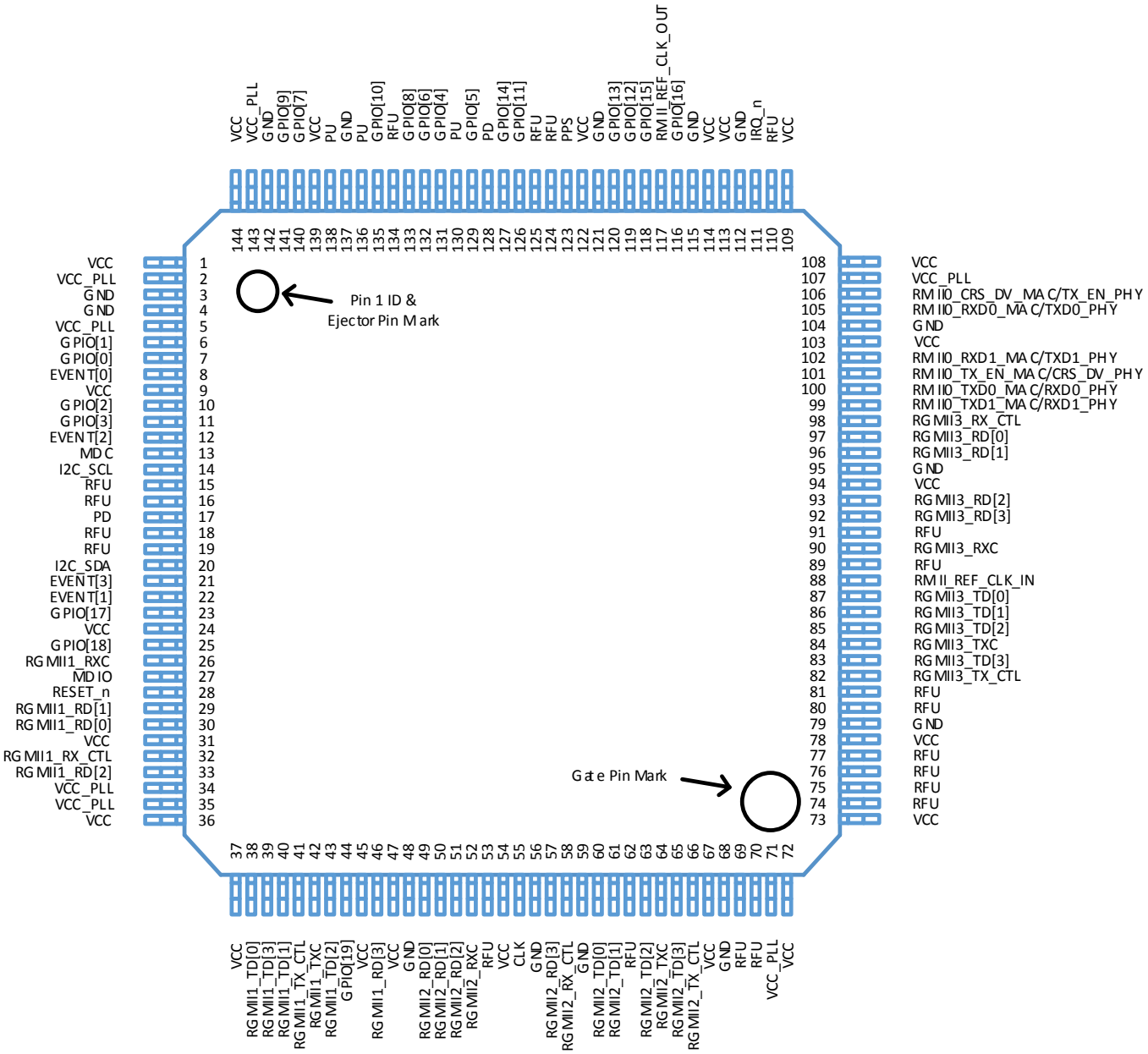


3. PIN DESCRIPTION

Devices are available in two different packages; XRS7004E and XRS7003E have EQFP144 package (Chapter 3.1) and XRS7004F, XRS7003F and XRS3003F have FBGA256 package (Chapter 3.2).

3.1 EQFP144 Package

Figure 5. XRS7004E Device Pinout (Top View)



Pin type definitions are listed in Table 2.

Table 2. Pin Types

Pin type	Definition
I/O	Input and output
I	Input only
O	Output only
Power	Operating power for the device
Ground	Ground connection for the device

The pin definitions of XRS7004E (Figure 5) and XRS7003E are presented in the following tables (Table 3 to Table 9). Note that the only difference between the pinouts of XRS7004E and XRS7003E is that XRS7003E has two RGMII interfaces (RGMII1 and RGMII2) while XRS7004E has three RGMII interfaces (RGMII1, RGMII2 and RGMII3). At XRS7003E the RGMII3 input pins have internal pull-ups and RGMII3 output pins are driven low. At XRS7003 it is allowed to leave the RGMII3 input pins floating, drive them low or high or connect them to an RGMII interface of another chip.

Generally it is recommended to pull or tie all the unused input pins up or down unless otherwise stated in the Pin Description.

Table 3. EQFP144 Package RGMII Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
26 52 90	RGMII1_RXC RGMII2_RXC RGMII3_RXC	I	Receive Clock. The RX clock is 2.5 MHz for 10 Mbit/s, 25 MHz for 100 Mbit/s and 125 MHz for 1000 Mbit/s. Note that the **PCB is required to add 1.5 ns to 2.0 ns more delay** to the clock line than the other lines, unless the other end (PHY) has configurable RX clock delay. See RGMII timing in Chapter 10.5.3.
32 58 98	RGMII1_RX_CTL RGMII2_RX_CTL RGMII3_RX_CTL	I	Combined RXDV and RXER.
30 29 33 46 49 50 51 57 97 96 93 92	RGMII1_RD[0] RGMII1_RD[1] RGMII1_RD[2] RGMII1_RD[3] RGMII2_RD[0] RGMII2_RD[1] RGMII2_RD[2] RGMII2_RD[3] RGMII3_RD[0] RGMII3_RD[1] RGMII3_RD[2] RGMII3_RD[3]	I	Receive Data. Double data rate at speed of 1000 Mbit/s and single data rate at speeds of 10/100 Mbit/s.

Pin Number	Pin Name	Pin Type	Pin Description
42 64 84	RGMII1_TXC RGMII2_TXC RGMII3_TXC	O	Transmit Clock. The TX clock is 2.5 MHz for 10 Mbit/s, 25 MHz for 100 Mbit/s and 125 MHz for 1000 Mbit/s. Note that the **PCB is required to add 1.5 ns to 2.0 ns more delay** to the clock line than the other lines, unless the other end (PHY) has configurable TX clock delay. See RGMII timing in Chapter 10.5.3. Note that the IO voltage is 3.3V and therefore the other end has to be 3.3V tolerant.
41 66 82	RGMII1_TX_CTL RGMII2_TX_CTL RGMII3_TX_CTL	O	Combined TXEN and TXER. Note that the IO voltage is 3.3V and therefore the other end has to be 3.3V tolerant.
38 40 43 39 60 61 63 65 87 86 85 83	RGMII1_TD[0] RGMII1_TD[1] RGMII1_TD[2] RGMII1_TD[3] RGMII2_TD[0] RGMII2_TD[1] RGMII2_TD[2] RGMII2_TD[3] RGMII3_TD[0] RGMII3_TD[1] RGMII3_TD[2] RGMII3_TD[3]	O	Transmit Data. Double data rate at speed of 1000 Mbit/s and single data rate at speeds of 10/100 Mbit/s. Note that the IO voltage is 3.3V and therefore the other end has to be 3.3V tolerant.

Table 4. EQFP144 Package RMII Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
105 102	RMII0_RXD0_MAC/TXD0_PHY RMII0_RXD1_MAC/TXD1_PHY	I	RMII input data. In PHY mode this is transmit data. In MAC mode this is receive data.
106	RMII0_CRS_DV_MAC/TX_EN_PHY	I	RMII input data valid. In PHY mode this is TX_EN. In MAC mode this is CRS_DV.
100 99	RMII0_TXD0_MAC/RXD0_PHY RMII0_TXD1_MAC/RXD1_PHY	O	RMII output data. In PHY mode this is receive data. In MAC mode this is transmit data.
101	RMII0_TX_EN_MAC/CRS_DV_PHY	O	RMII output data valid. In PHY mode this is CRS_DV. In MAC mode this is TX_EN.
88	RMII_REF_CLK_IN	I	RMII 50 MHz reference clock input. The same reference clock must be used for the both ends.
117	RMII_REF_CLK_OUT	O	50 MHz reference clock output. Can be connected to RMII_REF_CLK_IN or left unconnected.

Table 5. EQFP144 Package Register Access and Interrupt Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
13	MDC	I	Management Data Clock If the interface is not used, the signal must be pulled or tied up or down.
27	MDIO	I/O	Management Data I/O. Must be externally pulled up. 10 kΩ pull-up resistor recommended.
14	I2C_SCL	I	I2C clock If the interface is not used, the signal must be pulled or tied up or down.
20	I2C_SDA	I/O	I2C data. Open drain. Must be externally pulled up. 10 kΩ pull-up resistor recommended.
111	IRQ_n	O	Interrupt Request Output. Active low, open drain. 10 kΩ pull-up resistor recommended.

Table 6. EQFP144 Package Time Synchronization Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
8 22 12 21	EVENT[0] EVENT[1] EVENT[2] EVENT[3]	I	Event Input. Time Stamper (TS, Chapter 8) can be used to time stamp the rising edges of the signal. Signals with frequency of 0 Hz to 25 MHz are supported. Pull or tie unused inputs up or down.
123	PPS	O	Pulse Per Second output. A pulse with length of 20 µs occurring once a second. The rising edge of the pulse is when nanoseconds value of the RTC wraps around and the seconds value is incremented by one second.

Table 7. EQFP144 Package Clock and Reset Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
55	CLK	I	25 MHz reference clock input.
28	RESET_n	I	Active low hardware reset signal.

Table 8. EQFP144 Package Power and Ground Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
9 24 31 54 45 67 78 94 103 113 139 122 73 72 47 37 36 144 114 109 108 1	VCC	Power	3.3V operating power for the device.

Pin Number	Pin Name	Pin Type	Pin Description
35 34 5 107 143 71 2	VCC_PLL	Power	3.3V power for the internal PLL blocks of the device. Use separate power islands for VCC_PLL and VCC. VCC_PLL must be isolated from other power planes by using a ferrite bead or equivalent method. The ferrite bead should have low DC resistance and high impedance at 100 MHz.
3 4 95 79 68 59 56 48 142 137 121 115 112 104 Exposed pad	GND	Ground	Ground connection for the device. Note that the Exposed pad at the bottom of the device has to be connected to ground as well.

Table 9. EQFP144 Package Other Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
7 6 10 11 131 129 132 140 133 141 135 126 119 120 127 118 116 23 25 44	GPIO[0] GPIO[1] GPIO[2] GPIO[3] GPIO[4] GPIO[5] GPIO[6] GPIO[7] GPIO[8] GPIO[9] GPIO[10] GPIO[11] GPIO[12] GPIO[13] GPIO[14] GPIO[15] GPIO[16] GPIO[17] GPIO[18] GPIO[19]	I/O	General Purpose Input/Output. These pins can be controlled by the user, see GPIO registers in Chapter 9.1. It is recommended to tie the unused GPIO pins to GND. GPIO[0:1] are multi-function pins that are used to configure the I2C and MDIO addresses of the device during startup, see Chapter 5.4.
17 128	PD	I	Pull Down. These pins must be pulled down to GND. 1 kΩ pull-down resistors recommended.
130 136 138	PU	I/O	Pull Up. These pins must be pulled up to 3.3V VCC. 1 kΩ pull-up resistors recommended. Use individual pull-up resistor for each pin. Do not connect these pins to each other.

Pin Number	Pin Name	Pin Type	Pin Description
76 77 124 125 134 15 16 18 19 53 62 69 70 74 75 80 81 89 91 110	RFU	-	Reserved for Future Use. Do Not Connect these pins anywhere as the purpose of these pins may change in future versions.

3.2 FBGA256 Package

Figure 6. XRS7004F Device Pinout (Top View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
A	GND	GPIO[0]	RFU	GPIO[1]	RFU	GPIO[3]	GPIO[5]	GPIO[7]	RFU	GPIO[9]	GPIO[10]	GPIO[12]	RFU	RFU	RFU	GND	A
B	MDC	EVENT[0]	RFU	RFU	GPIO[2]	GPIO[4]	GPIO[6]	RFU	GPIO[8]	RFU	GPIO[11]	GPIO[13]	RFU	GND	RFU	RFU	B
C	MDIO	I2C_SDA	I2C_SCL	RFU	RFU	RFU	VCCIO33	GND	RFU	RFU	GND	RFU	RFU	RFU	RFU	RFU	C
D	IRQ_n	GND	GND	VCC_PLL	VCCA	GND	VCCIO33	VCCIO33	RFU	VCCIO33	VCCIO33	RFU	VCC_PLL	RFU	RFU	RFU	D
E	EVENT[1]	GND	GPIO[14]	VCCA	VCCA	GND	PU	PU	RFU	RFU	RFU	VCCA	GND	RFU	RFU	RMII0_CRS_DV_MAC	E
F	EVENT[2]	RESET_n	GND	GPIO[15]	PPS	VCC	PU	PD	CLK	GND	GND	RFU	VCCIO33	RFU	GND	RMII0_RXD1_MAC	F
G	RFU	GPIO[16]	GND	VCCIO33	GPIO[17]	GND	VCC	GND	VCC	GND	GND	RFU	VCCIO33	RFU	RMII0_RXD0_MAC	RMII0_TXD0_MAC	G
H	RFU	RFU	PD	VCCIO33	RFU	GND	GND	VCC	VCC	VCC	GND	RFU	VCCIO33	GND	RMII0_TX_EN_MAC	RMII0_REF_CLK_OUT	H
J	GPIO[18]	EVENT[3]	RFU	VCCIO33	GPIO[19]	GND	VCC	VCC	VCC	GND	GND	RFU	VCCIO33	RFU	RMII_REF_CLK_IN	RFU	J
K	GND	RFU	GND	VCCIO25	RFU	GND	GND	VCC	GND	VCC	GND	RFU	VCCIO25	RFU	RMII0_TXD1_MAC	GND	K
L	RGMII1_RD[1]	RFU	RGMII1_RXC	VCCIO25	VCCA	GND	GND	GND	GND	GND	GND	RFU	VCCIO25	GND	RFU	RFU	L
M	RFU	RGMII1_RD[2]	RFU	GND	VCC_PLL	RFU	RFU	RFU	RGMII3_RXC	RFU	RFU	VCCA	VCCIO25	RFU	RFU	RFU	M
N	RGMII1_RX_CTL	RGMII1_RD[0]	RFU	RFU	RFU	VCCIO25	VCCIO25	VCCIO25	GND	VCCIO25	VCCIO25	GND	VCC_PLL	RFU	GND	RGMII3_TD[1]	N
P	RGMII1_RD[3]	RFU	GND	RFU	RFU	RFU	GND	RGMII2_RXC	RFU	RGMII3_RX_CTL	RGMII3_RD[1]	RFU	RFU	RFU	RFU	RGMII3_TD[2]	P
R	RGMII1_TD[0]	RGMII1_TD[2]	RGMII1_TXC	RGMII2_RX_CTL	RGMII2_RD[2]	RGMII2_RD[3]	RGMII2_TD[0]	RGMII2_TD[1]	RGMII2_TD[2]	RGMII2_TD[3]	RGMII3_RD[3]	RGMII3_RD[0]	GND	RGMII3_TXC	RGMII3_TD[3]	RFU	R
T	GND	RGMII1_TX_CTL	RGMII1_TD[1]	RGMII1_TD[3]	RGMII2_RD[1]	RGMII2_RD[0]	RFU	RGMII2_TXC	RGMII2_TX_CTL	GND	RFU	RGMII3_RD[2]	RFU	RGMII3_TD[0]	RGMII3_TX_CTL	GND	T
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

Pin type definitions are listed in Table 10.

Table 10. Pin Types

Pin type	Definition
I/O	Input and output
I	Input only
O	Output only
Power	Operating power for the device
Ground	Ground connection for the device

The pin definitions of XRS7004F (Figure 5), XRS7003F and XRS3003F are presented in in the following tables (Table 11 to Table 17). Note that the only difference between the pinouts of XRS7004F and XRS7003F is that XRS7003F has two RGMII interfaces (RGMII1 and RGMII2) while XRS7004F has three RGMII interfaces (RGMII1, RGMII2 and RGMII3). At XRS7003F and XRS3003F the RGMII3 input pins have internal pull-ups and RGMII3 output pins are driven low. At XRS7003F and XRS3003F it is allowed to leave the RGMII3 input pins floating, drive them low or high or connect them to an RGMII interface of another chip.

The pinout of XRS3003F differs from the pinout of 7003F in the following ways:

- EVENT[0…3] pins are reserved in XRS3003F. Connect like it was XRS7003/XRS7004, or leave the pins floating. XRS3003F has weak internal pull-up in EVENT signals.
- I2C pins (I2C_SCL, I2C_SDA) are reserved in XRS3003F. Connect like it was XRS7003/XRS7004, or leave the pins floating. XRS3003F has weak internal pull-up in I2C pins.

Otherwise the pinouts are similar, and the devices are interchangeable.

Table 11. FBGA256 Package RGMII Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
L3 P8 M9	RGMII1_RXC RGMII2_RXC RGMII3_RXC	I	Receive Clock. The RX clock is 2.5 MHz for 10 Mbit/s, 25 MHz for 100 Mbit/s and 125 MHz for 1000 Mbit/s. Note that the **PCB is required to add 1.5 ns to 2.0 ns more delay** to the clock line than the other lines, unless the other end (PHY) has configurable RX clock delay. See RGMII timing in Chapter 10.5.3.
N1 R4 P10	RGMII1_RX_CTL RGMII2_RX_CTL RGMII3_RX_CTL	I	Combined RXDV and RXER.
N2 L1 M2 P1 T6 T5 R5	RGMII1_RD[0] RGMII1_RD[1] RGMII1_RD[2] RGMII1_RD[3] RGMII2_RD[0] RGMII2_RD[1] RGMII2_RD[2]	I	Receive Data. Double data rate at speed of 1000 Mbit/s and single data rate at speeds of 10/100 Mbit/s.

R6 R12 P11 T12 R11	RGMII2_RD[3] RGMII3_RD[0] RGMII3_RD[1] RGMII3_RD[2] RGMII3_RD[3]		
R3 T8 R14	RGMII1_TXC RGMII2_TXC RGMII3_TXC	O	Transmit Clock. The TX clock is 2.5 MHz for 10 Mbit/s, 25 MHz for 100 Mbit/s and 125 MHz for 1000 Mbit/s. Note that the **PCB is required to add 1.5 ns to 2.0 ns more delay** to the clock line than the other lines, unless the other end (PHY) has configurable TX clock delay. See RGMII timing in Chapter 10.5.3.
T2 T9 T15	RGMII1_TX_CTL RGMII2_TX_CTL RGMII3_TX_CTL	O	Combined TXEN and TXER.
R1 T3 R2 T4 R7 R8 R9 R10 T14 N16 P16 R15	RGMII1_TD[0] RGMII1_TD[1] RGMII1_TD[2] RGMII1_TD[3] RGMII2_TD[0] RGMII2_TD[1] RGMII2_TD[2] RGMII2_TD[3] RGMII3_TD[0] RGMII3_TD[1] RGMII3_TD[2] RGMII3_TD[3]	O	Transmit Data. Double data rate at speed of 1000 Mbit/s and single data rate at speeds of 10/100 Mbit/s.

Table 12. FBGA256 Package RMII Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
G15 F16	RMII0_RXD0_MAC/TXD0_PHY RMII0_RXD1_MAC/TXD1_PHY	I	RMII input data. In PHY mode this is transmit data. In MAC mode this is receive data.
E16	RMII0_CRS_DV_MAC/TX_EN_PHY	I	RMII input data valid. In PHY mode this is TX_EN. In MAC mode this is CRS_DV.
G16 K15	RMII0_TXD0_MAC/RXD0_PHY RMII0_TXD1_MAC/RXD1_PHY	O	RMII output data. In PHY mode this is receive data. In MAC mode this is transmit data.
H15	RMII0_TX_EN_MAC/CRS_DV_PHY	O	RMII output data valid. In PHY mode this is CRS_DV. In MAC mode this is TX_EN.
J15	RMII_REF_CLK_IN	I	RMII 50 MHz reference clock input. The same reference clock must be used for the both ends.

H16	RMII_REF_CLK_OUT	O	50 MHz reference clock output. Can be connected to RMII_REF_CLK_IN or left unconnected.

Table 13. FBGA256 Package Register Access and Interrupt Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
B1	MDC	I	Management Data Clock If the interface is not used, the signal must be pulled or tied up or down.
C1	MDIO	I/O	Management Data I/O. Must be externally pulled up. 10 kΩ pull-up resistor recommended.
C3	I2C_SCL	I	I2C clock If the interface is not used, the signal must be pulled or tied up or down.
C2	I2C_SDA	I/O	I2C data. Open drain. Must be externally pulled up. 10 kΩ pull-up resistor recommended.
D1	IRQ_n	O	Interrupt Request Output. Active low, open drain. 10 kΩ pull-up resistor recommended.

Table 14. FBGA256 Package Time Synchronization Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
B2 E1 F1 J2	EVENT[0] EVENT[1] EVENT[2] EVENT[3]	I	Event Input. Time Stamper (TS, Chapter 8) can be used to time stamp the rising edges of the signal. Signals with frequency of 0 Hz to 25 MHz are supported. Pull or tie unused inputs up or down.
F5	PPS	O	Pulse Per Second output. A pulse with length of 20 µs occurring once a second. The rising edge of the pulse is when nanoseconds value of the RTC wraps around and the seconds value is incremented by one second.

Table 15. FBGA256 Package Clock and Reset Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
F9	CLK	I	25 MHz reference clock input.
F2	RESET_n	I	Active low hardware reset signal.

Table 16. FBGA256 Package Power and Ground Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
K8 K10 J9 J8 J7 H9 H8 H10 G9 G7 F6	VCC	Power	1.2V operating power for the device.
M5 D13 D4 N13	VCC_PLL	Power	1.2V operating power for the device internal PLL blocks. Use separate power island for VCC_PLL. VCC_PLL must be isolated from other 1.2V power planes by using a ferrite bead or equivalent method. The ferrite bead should have low DC resistance and high impedance at 100 MHz.
L5 E12 E4 D5 E5 M12	VCCA	Power	2.5V operating power for the device internal PLL blocks. Use separate power island for VCCA. VCCA must be isolated from other 2.5V power planes by using a ferrite bead or equivalent method. The ferrite bead should have low DC resistance and high impedance at 100 MHz.
H4 G4 J4 J13 H13 G13 F13 D11 D10 D8 D7 C7	VCCIO33	Power	3.3V IO voltage. The IO voltage for all the interfaces except RGMII.
L4 K4 N8 N7 N6	VCCIO25	Power	2.5V IO voltage for RGMII interfaces.

N11 N10 M13 L13 K13			
D2 F3 E2 T16 T10 T1 R13 P7 P3 N9 N15 N12 M4 L14 K9 K7 K3 K16 K1 J10 H7 H14 G8 G3 G10 F15 E6 E13 D6 D3 C8 C11 B14 A16 A1 F10 F11 G6 G11 H6 H11 J6	GND	Ground	Ground connection for the device.

J11 K6 K11 L6 L7 L8 L9 L10 L11			

Table 17. FBGA256 Package Other Pin Definitions

Pin Number	Pin Name	Pin Type	Pin Description
A2 A4 B5 A6 B6 A7 B7 A8 B9 A10 A11 B11 A12 B12 E3 F4 G2 G5 J1 J5	GPIO[0] GPIO[1] GPIO[2] GPIO[3] GPIO[4] GPIO[5] GPIO[6] GPIO[7] GPIO[8] GPIO[9] GPIO[10] GPIO[11] GPIO[12] GPIO[13] GPIO[14] GPIO[15] GPIO[16] GPIO[17] GPIO[18] GPIO[19]	I/O	General Purpose Input/Output. These pins can be controlled by the user, see GPIO registers in Chapter 9.1. It is recommended to tie the unused GPIO pins to GND. GPIO[0:1] are multi-function pins that are used to configure the I2C and MDIO addresses of the device during startup, see Chapter 5.4.
F8 H3	PD	I	Pull Down. These pins must be pulled down to GND. 1 kΩ pull-down resistors recommended.
E7 E8 F7	PU	I/O	Pull Up. These pins must be pulled up to 3.3V VCCIO. 1 kΩ pull-up resistors recommended. Use individual pull-up resistor for each pin. Do not connect these pins to each other's.
A3 A5 A9 A13 A14	RFU	-	Reserved for Future Use. Do Not Connect these pins anywhere as the purpose of these pins may change in future versions.

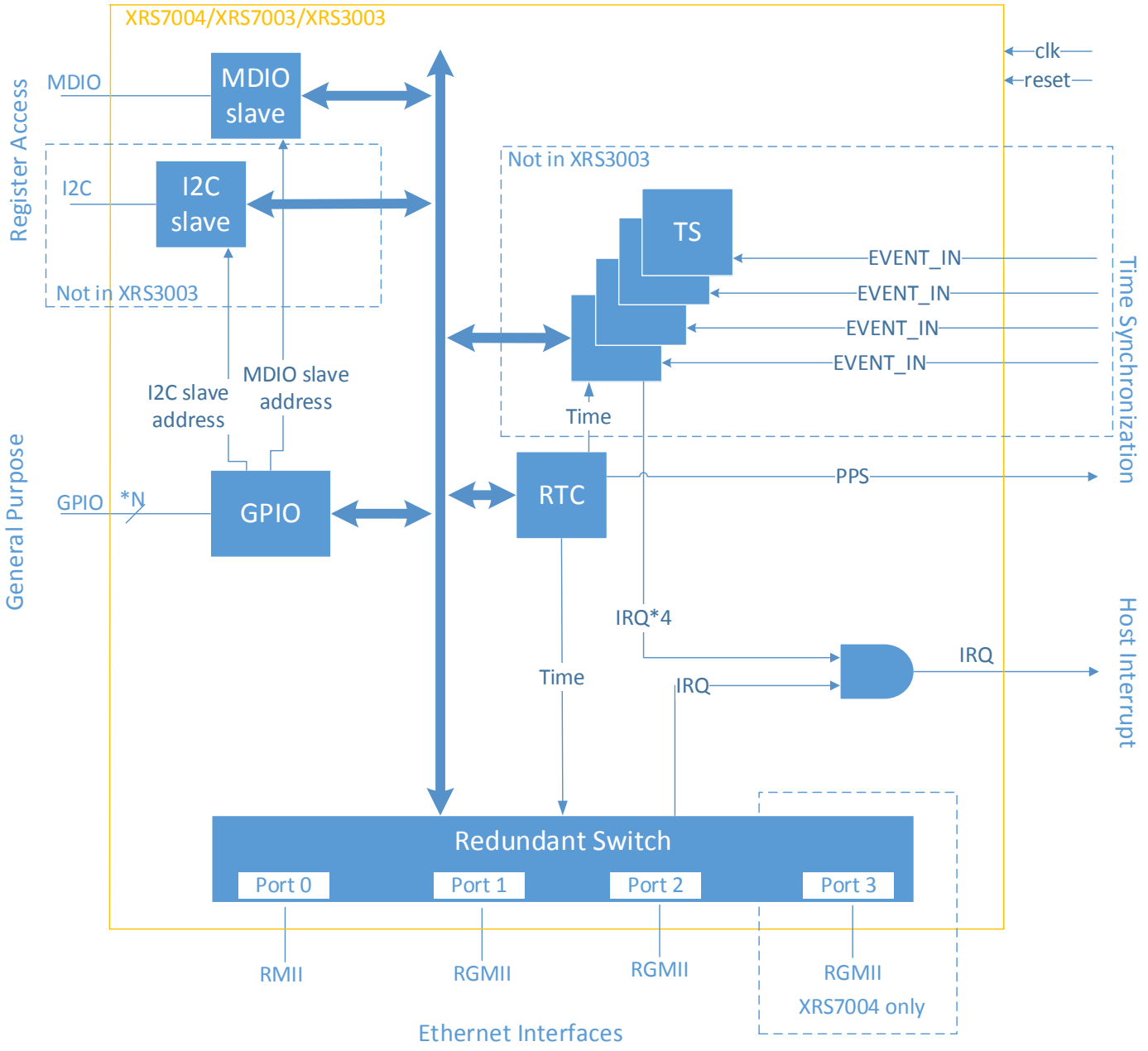
A15 B3 B4 B8 B10 B13 B15 B16 C4 C5 C6 C9 C10 C12 C13 C14 C15 C16 D9 D12 D14 D15 D16 E9 E10 E11 E14 E15 F12 F14 G1 G12 G14 H1 H2 H5 H12 J3 J12 J14 J16 K2 K5 K12 K14 L2 L12			

L15 L16 M1 M3 M6 M7 M8 M10 M11 M14 M15 M16 N3 N4 N5 N14 P2 P4 P5 P6 P9 P12 P13 P14 P15 R16 T7 T11 T13			

4. FUNCTIONAL OVERVIEW

Top level block diagram is presented in Figure 7. The most important block is the Redundant Switch (RS) that forwards Ethernet frames between Ethernet (RMII and RGMII) interfaces. For PTP (Precision Time Protocol) functionality device internal time is kept in RTC (Real-Time Clock). The internal time can be transferred outside the device or the device can be synchronized to external time with help of PPS (Pulse Per Second) output or Event Inputs. The functionality of the device is controlled by the host system using either MDIO or I2C protocol to access device internal registers.

Figure 7. Top Level Block Diagram



5. REGISTER INTERFACE

The functionality of XRS7004/XRS7003/XRS3003 is controlled through registers. The internal register map is presented in Table 18. The registers are accessible using MDIO (Chapter 5.2) or I2C (Chapter 5.3) interface. XRS7004 and XRS7003 support both MDIO and I2C, XRS3003 supports only MDIO.

Table 18. Internal Register Map

Address Offset	Register Set	Section	Table
0x0000 0000	Device Identification	5.1	Table 19
0x0001 0000	GPIO (General Purpose I/O)	9.1	Table 39
0x0020 0000	Port 0 Configuration Registers	6.10	Table 30
0x0021 0000	Port 1 Configuration Registers	6.10	Table 30
0x0022 0000	Port 2 Configuration Registers	6.10	Table 30
0x0023 0000	Port 3 Configuration Registers (only in XRS7004)	6.10	Table 30
0x0028 0000	RTC (Real-Time Clock)	7.1	Table 37
0x0029 0000	TS0 (Time Stamper 0) (not in XRS3003)	8.1	Table 38
0x0029 8000	TS1 (Time Stamper 1) (not in XRS3003)	8.1	Table 38
0x002A 0000	TS2 (Time Stamper 2) (not in XRS3003)	8.1	Table 38
0x002A 8000	TS3 (Time Stamper 3) (not in XRS3003)	8.1	Table 38
0x0030 0000	Switch Configuration Registers	6.9	Table 26

5.1 Device Identification Registers

For identifying the device XRS7004, XRS7003 and XRS3003 have identification registers in the beginning of the register address space. The identification registers are presented in Table 19.

Table 19. Identification Registers

Address	Register	Description
0x0000	DEV_ID0	Reset: 0 x 01 00 (XRS7003E), 0 x 02 00 (XRS7004E), 0 x 01 01 (XRS7003F), 0 x 02 01 (XRS7004F), 0 x 03 01 (XRS3003F) Device Identification register 0. Bits | 15-0: | RO | Static value for identifying device type
0x0002	DEV_ID1	Reset: 0 x 00 40 Device Identification register 1. Bits | 15-0: | RO | Static value for identifying device type
0x0004	INT_ID0	Reset: 0 x XX XX Manufacturer Internal Revision Identification register 0. Only for manufacturer's own use. Bits | 15-0: | RO | Revision ID, bits 15-0.

Address	Register	Description
0x0006	INT_ID1	Reset: 0 x XX XX Manufacturer Internal Revision Identification register 1. Only for manufacturer's own use. Bits 15-0: RO Revision ID, bits 31-16.
0x0008	REV_ID	Reset: 0 x XX XX Revision Identification register. Bits 7-0: RO Minor Number Bits 15-8: RO Major Number

5.2 MDIO Slave

The MDIO Slave block (see Figure 7) connects the external MDIO bus to device internal bus so that external devices (typically CPU) can access the registers of the device internal blocks to control their functionality.

The MDIO registers are presented in Table 20.

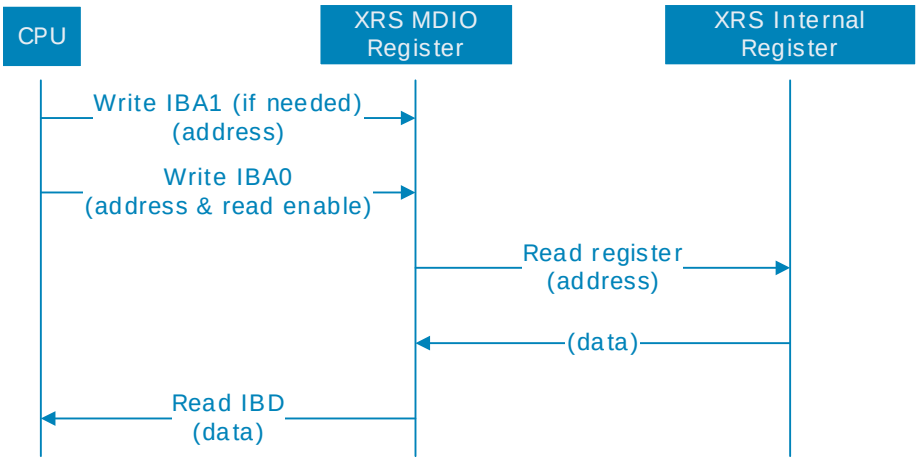
Table 20. XRS MDIO Registers

MDIO Register Address	Register	Description
0x00…0x0F	*Reserved*	*Reserved*
0x10	IBA0	Reset: 0 x 00 00 Internal bus address0. Address for the internal bus access, lowest bits. The requested action (Read/Write) is performed immediately after writing this register. Bit 0: R/W Read/Write 0 = Read. Data is available on next read from IBD register (see address 0x14). 1 = Write. Data in IBD register is written to internal bus. Bits 15-1: R/W Address Bits 15-1 of the address on the internal bus to where data is written or from where data is read. Address bit 0 is always 0 (because of 16 bit registers).
0x11	IBA1	Reset: 0 x 00 00 Internal bus address1. Address for the internal bus access, highest bits. Bits 15-0: R/W Address Bits 31:16 of the address on the internal bus to where data is written or from where data is read.
0x12…0x13	*Reserved*	*Reserved*

MDIO Register Address	Register	Description
0x14	IBD	Reset: 0 x 00 00 Internal bus data. Data for the internal bus access. Bits 15-0: R/W Data If write command is given to IBA0 register, contents of this register are used in write access to internal bus. If read command is given (bit 0 in register IBA0) the read data is copied into this register.
0x15…0x1F	*Reserved*	*Reserved*

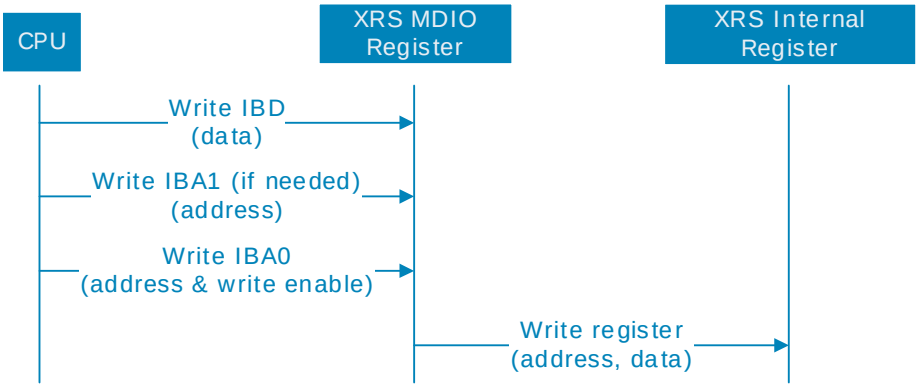
Read access to internal bus using the MDIO registers is presented in Figure 8.

Figure 8. Read Access using MDIO



Write access to internal bus using the MDIO registers is presented in Figure 9.

Figure 9. Write Access using MDIO



Note that if accesses are made to the same memory area so that the highest address bits of the device internal bus address are the same in consecutive read or write accesses, it is not necessary to write the register IBA1. Leaving

out these unnecessary MDIO register writes speeds up the register access. It is also not necessary to write register IBD if consecutive write accesses are made using the same data.

The MDIO registers are in conformance with IEEE std. 802.3. The reserved registers return 0 when read. Writing is not allowed to reserved registers.

The XRS7004/7003/3003 MDIO device address is configurable as presented in Chapter 5.4. The XRS7004/7003/3003 MDIO slave does not react to MDIO accesses with other MDIO device addresses.

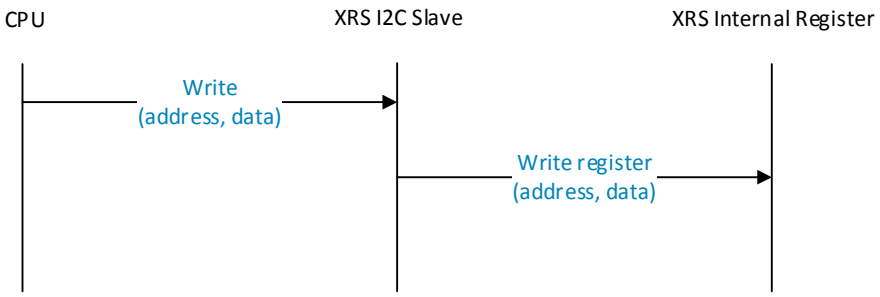
It is prohibited to use MDIO and I2C to access the device internal registers at the same time.

5.3 I2C Slave

The I2C Slave block (see Figure 7) connects the external I2C bus to device internal bus so that external devices (typically CPU) can access the registers of the device internal blocks to control their functionality. The I2C slave (and the device internal bus) can be accessed by using the I2C address that is configurable as presented in Chapter 5.4. I2C Slave is not present in XRS3003 devices.

I2C accesses are 8-bit oriented. When writing to device internal bus from I2C bus, one write to device internal bus consists of one I2C write access, see Figure 10.

Figure 10. Write Access using I2C

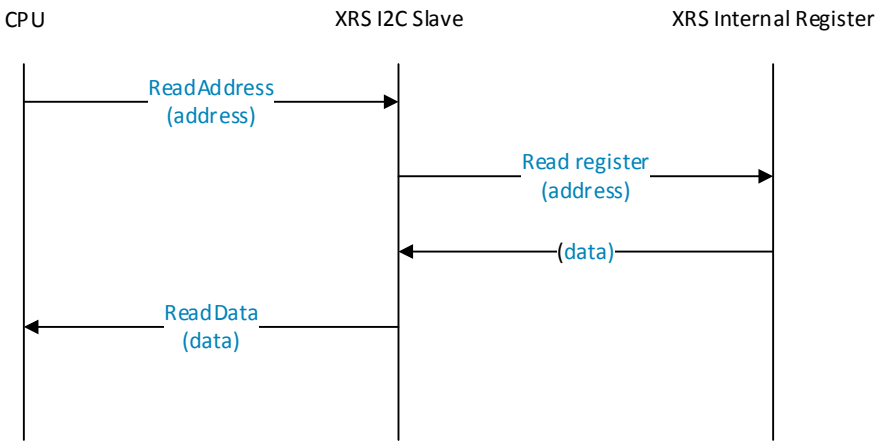


One read from device internal bus consist of two I2C accesses:

1. I2C write access that informs the device internal address to be read.
2. I2C read access that reads the data fetched from the device internal address.

See Figure 11.

Figure 11. Read Access using I2C



The three types of I2C access (Write, ReadAddress, ReadData) are presented in Figure 12, Figure 13 and Figure 14.

Figure 12. I2C Write

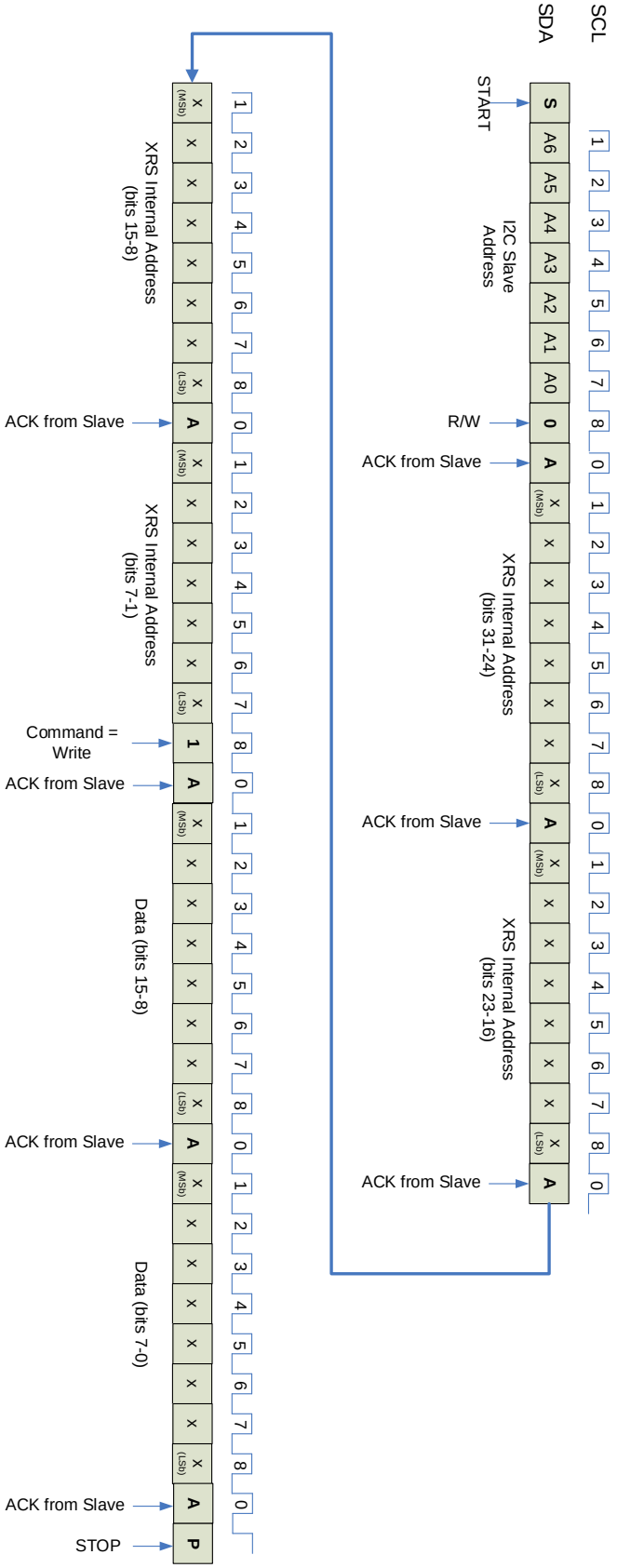


Figure 13. I2C ReadAddress

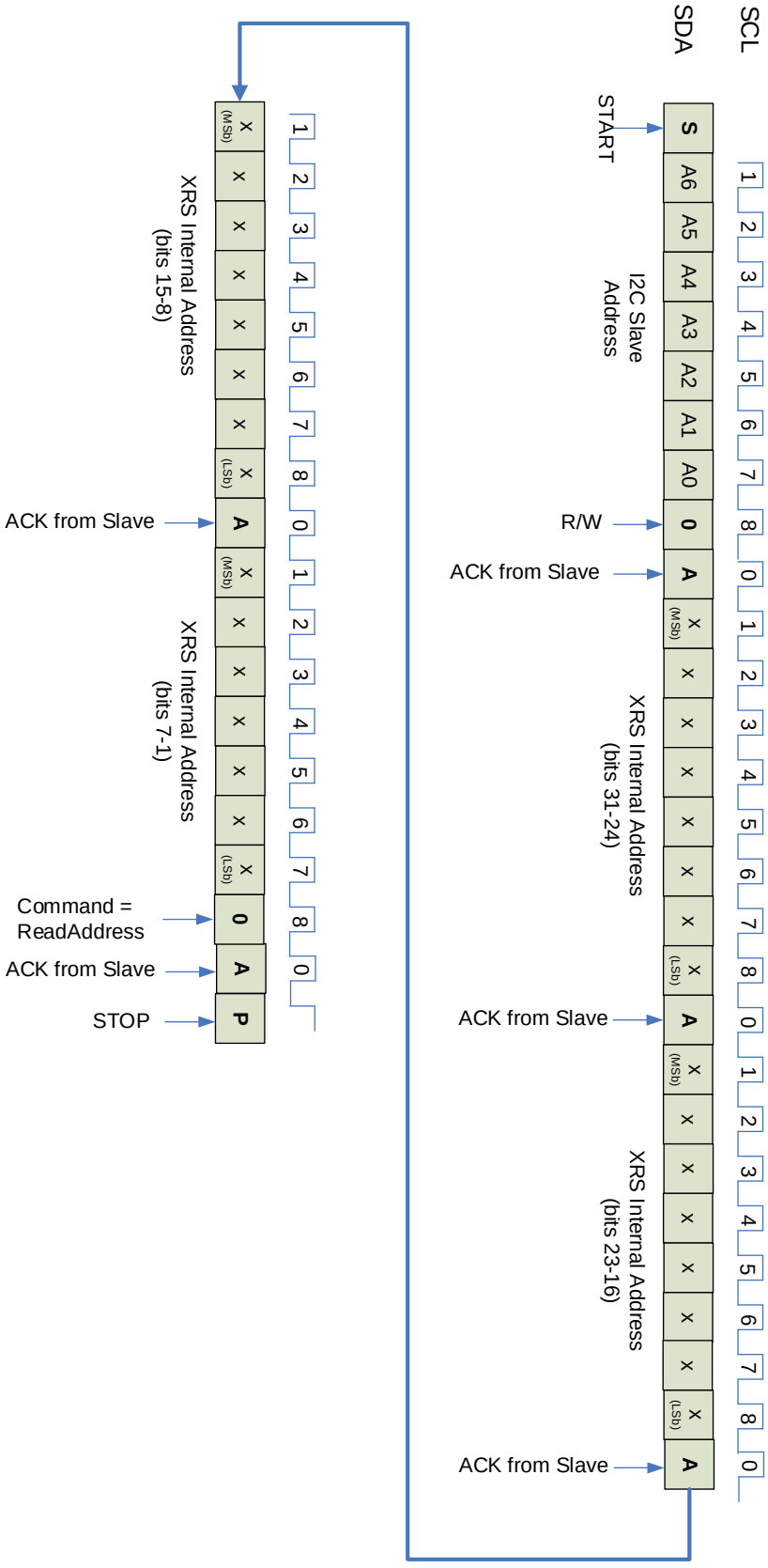
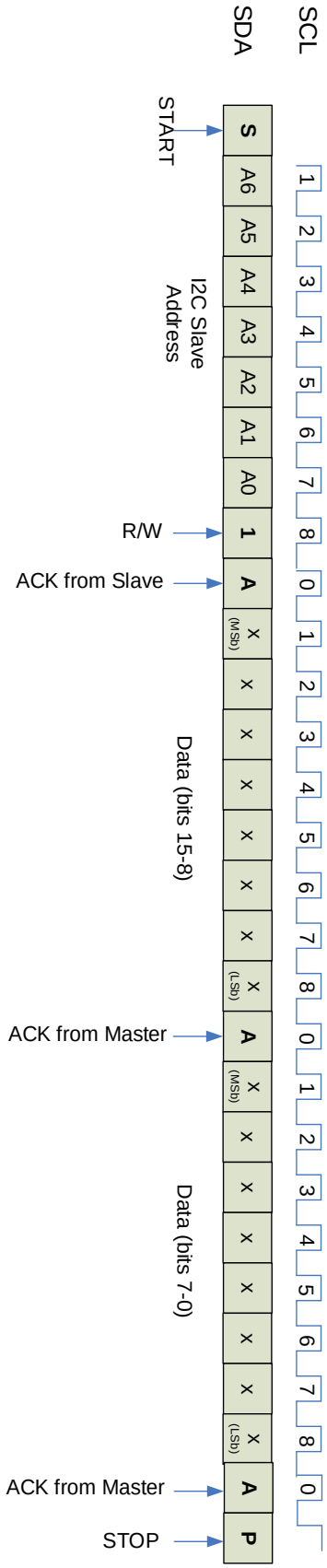


Figure 14. I2C ReadData



The I2C messages can be combined so that the master can send a new start bit and omit the preceding stop bit.

The I2C interface supports speeds up to 400-kHz Fast-mode (Fm).

5.4 MDIO and I2C Slave Address Configuration

The MDIO and I2C addresses of XRS7004/7003/3003 are configured with the external GPIO signals by pulling the GPIO pin 0 and the GPIO pin 1 up or down in startup. After the reset signal is de-asserted the GPIO pin 0 and the GPIO pin 1 have to be kept steady for at least 10 clock cycles of the *clk* clock. During this period XRS7004/7003/3003 reads the status of the GPIO pins 0 and 1 and configures the MDIO and I2C slave addresses as presented in Table 21 and Table 22. After the period of 10 clock cycles the GPIO pins 0 and 1 can be used for normal I/O.

Table 21. MDIO Address

	MDIO Address				
GPIO_1, GPIO_0	**A4 (MSB)**	**A3**	**A2**	**A1**	**A0 (LSB)**
0, 0	0	1	0	0	0
0, 1	0	1	0	0	1
1, 0	1	1	0	0	0
1, 1	1	1	0	0	1

Table 22. I2C Address

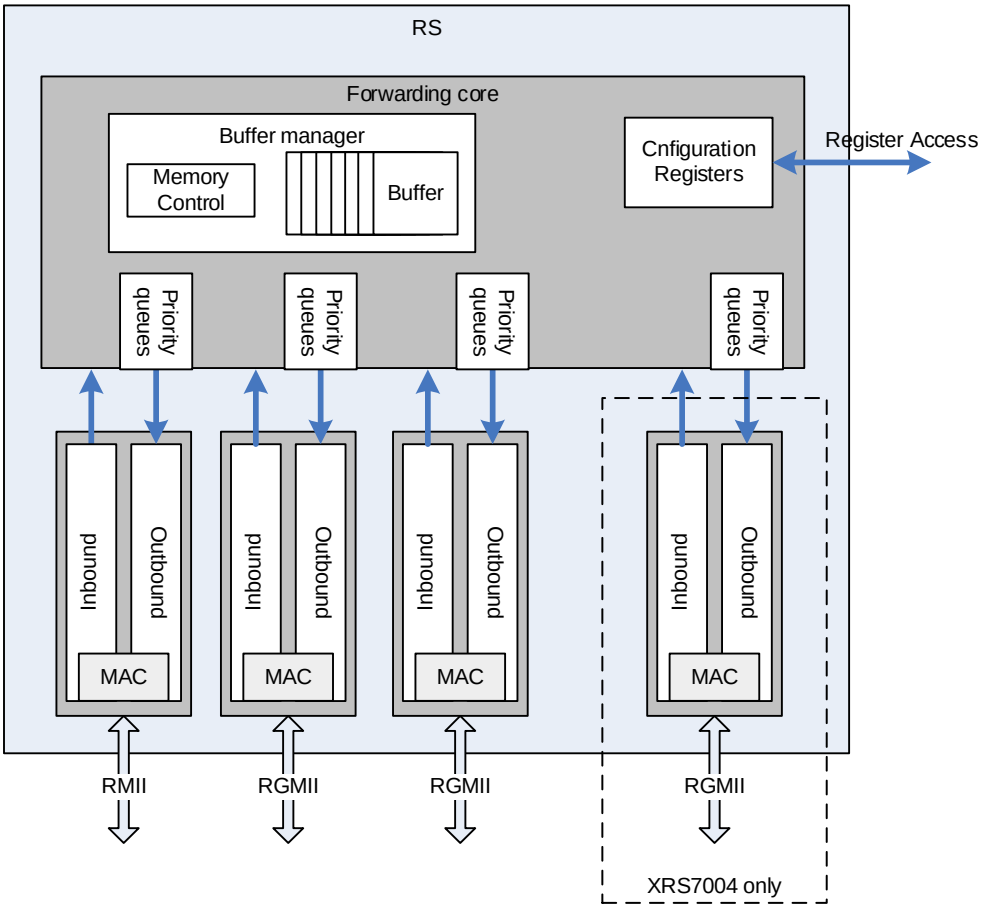
	I2C Address						
GPIO_1, GPIO_0	**A6 (MSB)**	**A5**	**A4**	**A3**	**A2**	**A1**	**A0 (LSB)**
0, 0	0	1	0	0	1	0	0
0, 1	0	1	1	0	1	0	0
1, 0	1	1	0	0	1	0	0
1, 1	1	1	1	0	1	0	0

6. REDUNDANT SWITCH (RS)

The Redundant Switch (RS) block handles the forwarding of Ethernet frames, HSR/PRP duplicate generation and removal, IEEE 1588 time stamping and all the normal Ethernet switch functionality in XRS7004/7003/3003.

The functional blocks of RS are presented in Figure 15. The functionality of RS can be controlled via configuration registers defined in Chapter 6.9 and 6.10.

Figure 15. RS Block Diagram



RS consists of three main blocks: Forwarding Core and Inbound and Outbound processing. The inbound and outbound processing include Ethernet Media Access Control (MAC).

The Forwarding core is responsible for managing the frames inside the switch. The forwarding core is common for all the ports and it does the actual forwarding of frames between ports. As frames may need to spend time inside the switch, they are stored in into a buffer memory.

Every port has its own MAC and inbound and outbound processing. The inbound and outbound processing of a port is independent from the other ports. The only exception to this is that the inbound processing entities share the same MAC address table.

Figure 16. Forwarding Path

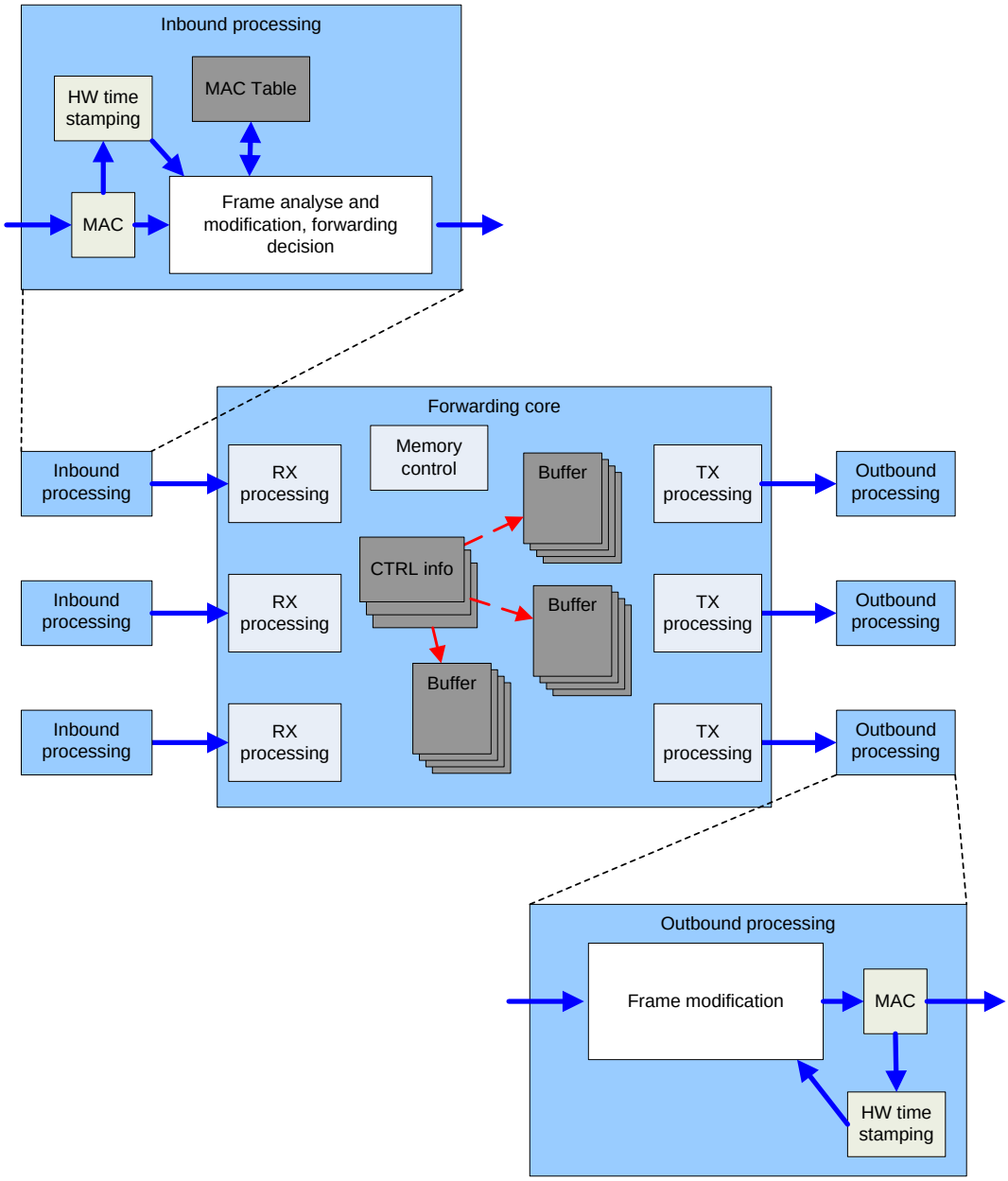


Figure 16 presents the processing path of a frame inside RS.

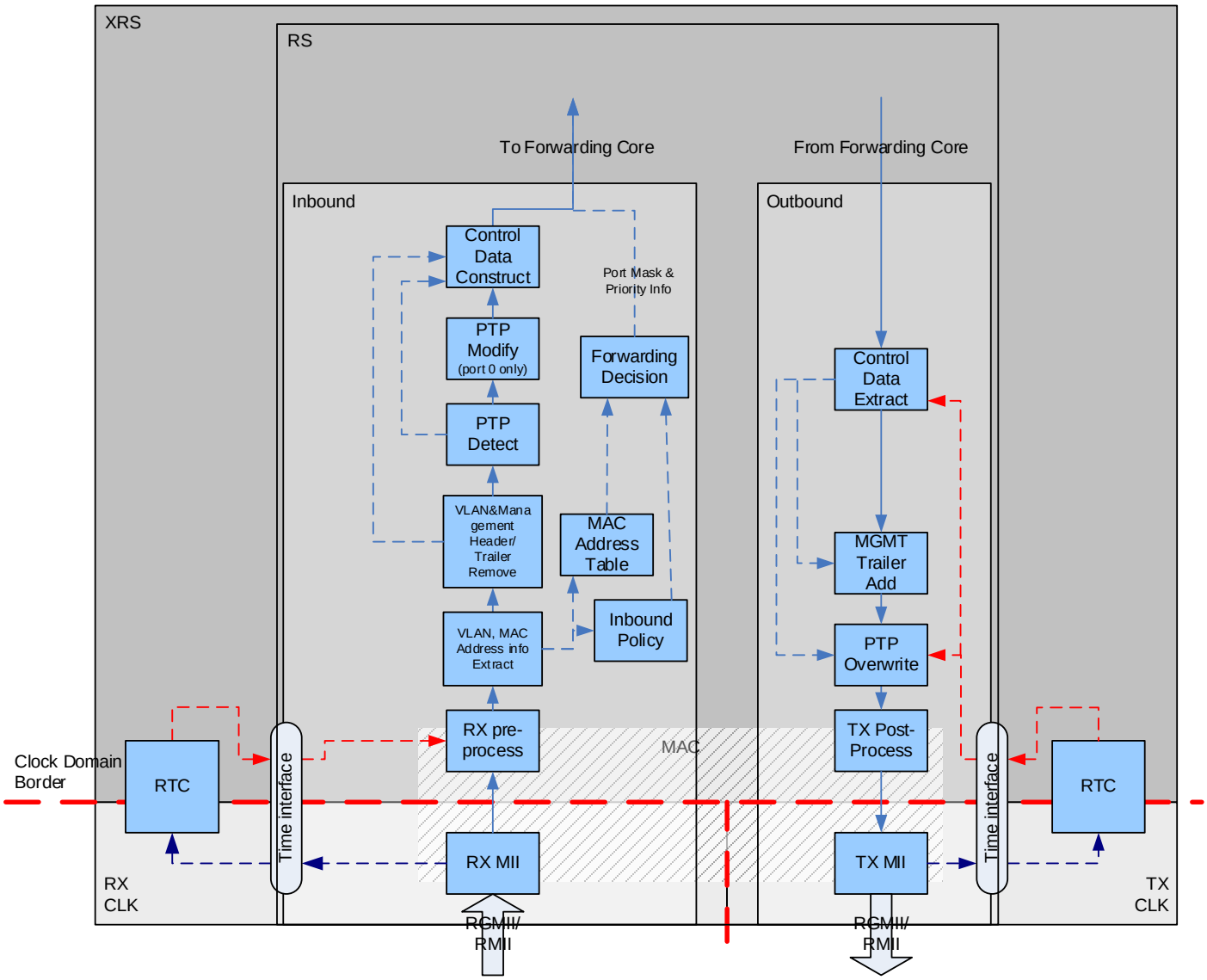
Inbound processing contains Ethernet MAC and data processing blocks that are able to analyze and modify the frame.

The Forwarding core contains RX and TX processing blocks for controlling Inbound and Outbound processing blocks. Memory controller block manages the memory used for storing frames and their control information. Storing of frames is needed because there can be more frames forwarded to an output port than what its capacity is. Frames are stored into the buffer memory in chunks of 512 Bytes. This means that every stored frame consumes N * 512 B of buffer memory, where N=1…3. When frames are stored into the buffer memory waiting to

be transmitted to an output port, they are in an output priority queue of the output port. The output priority queues contain pointers to the frames; no actual frame data is moved from place to another when queuing the frames.

The main function of the Outbound processing is to send frames from buffer memory to Ethernet. It contains functionality for retrieving data from the buffer memory, time stamping functions for PTP use and Ethernet MAC functionality for sending data to the medium. Every port has its own individual Outbound processing entity.

Figure 17. RS Inbound and Outbound Processing Blocks



RS Inbound and Outbound processing paths are depicted in Figure 17. The Inbound processing and Outbound processing functionality are completely independent of each other.

6.1 Inbound Processing

Inbound processing receives frames from Ethernet and transfers them to the buffer memory (see Figure 17). The functionality of inbound processing blocks are described in sub-paragraphs in this chapter.

During reception, Inbound processing does:

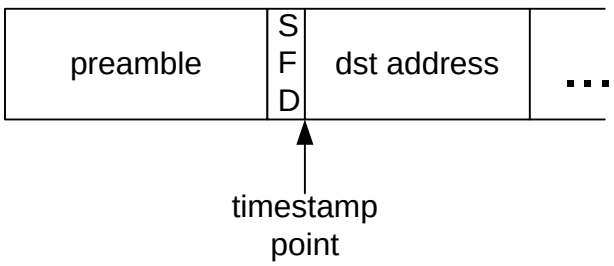
- Detect frame errors
- Timestamp frames
- Filter and recognize frames
- Determine the destination port(s) for every frame
- Perform MAC address learning
- Modify frames

6.1.1 *RX MII*

The RX MII receives frames from the Ethernet PHY. When RX MII is operational and frames arrive from the network, it writes the frames received from the network to the RX pre-process block. While the frame is being received, RX MII calculates CRC over the frame. After the reception of the frame is completed it indicates the status of the CRC calculation.

RX MII block indicates the start of the frame to Timestamp block via Time interface. The timestamp point for a frame is defined in Figure 18.

Figure 18. Ethernet Frame Timestamp Point



There are four kinds of errors that can occur while RX MII is receiving a frame. These are: Size Error, CRC Error, Octet Error and Line Error.

Size Error indicates that the received frame is over 1532 bytes long (without preamble, SFD and CRC). In that case the frame is truncated to 1532 bytes, and Size Error is generated.

CRC Error signals that the CRC checksum in the received frame was not the same as the one that was calculated while receiving the frame. This is a result of an error in the data of the received frame and an indication that the frame should be discarded. Size Error, Octet Error and Line Error usually cause also a CRC error to the received frame.

Octet Error occurs when the received frame contains an uneven number of half bytes (nibbles). This kind of a frame is not valid.

Line Error indicates that while receiving the Frame the PHY reported RX MII of an error.

All the frames received with an error are dropped by the Forwarding Core and the corresponding error counters are incremented (see Table 35). Also frames whose size is less than 64 bytes are discarded.

6.1.2 *Timestamp*

The RX MII block gives start of frame indication to RTC (Chapter 7) to determine the exact value of the reception time. The reception time of the frame is then given to the RX pre-process block and to PTP Modify block (port 0 only).

6.1.3 *RX Pre-process*

The RX pre-process block provides the received frames to the rest of the Inbound processing blocks. The Inbound processing blocks are chained in a row between the RX pre-process and the Forwarding core.

6.1.4 *Forwarding Decision*

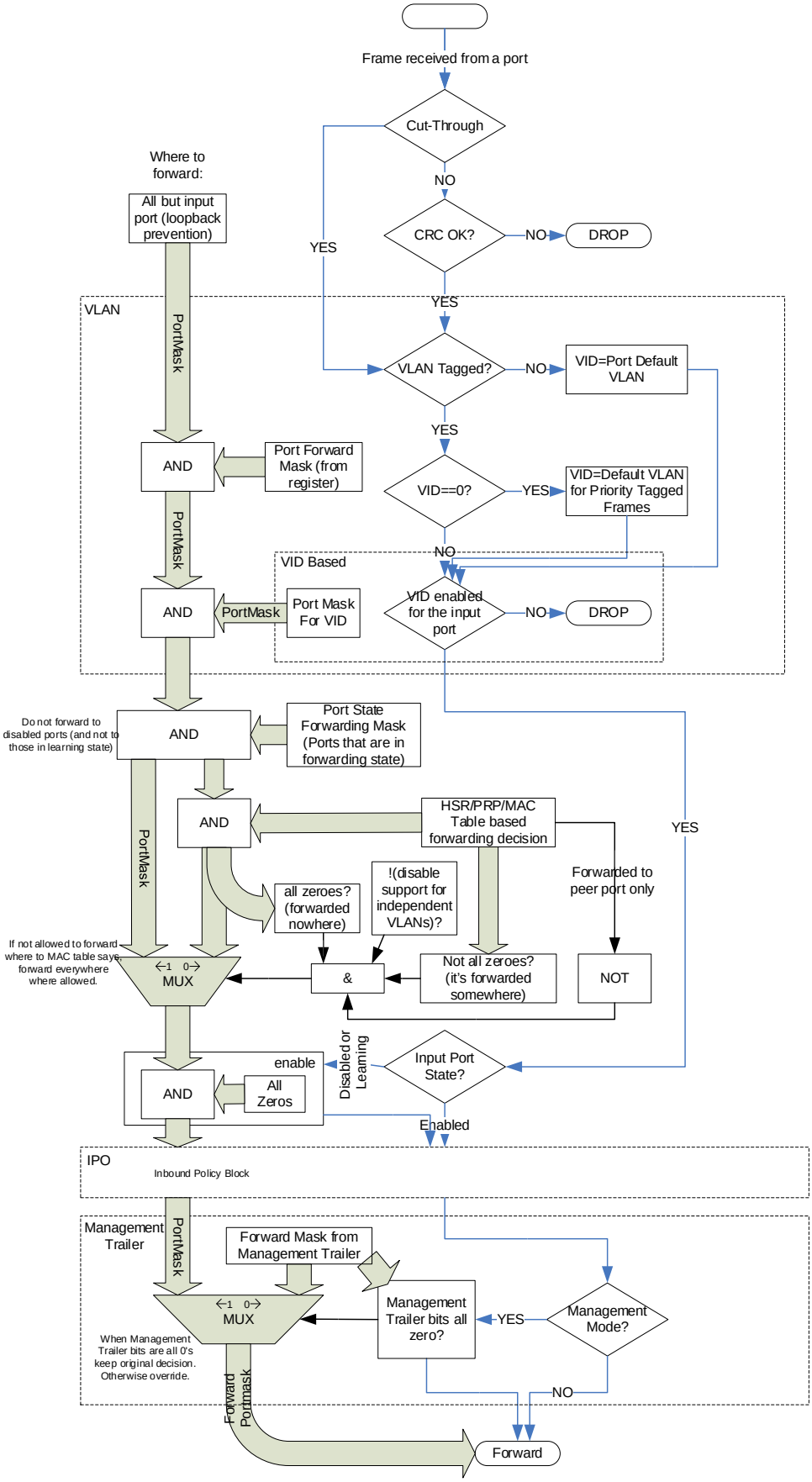
A Forwarding decision is made based on information from the following sources:

- MAC address table (Chapter 6.1.9.4)
- Management Trailer (Chapter 6.1.8)
- Inbound Policy (Chapter 6.1.5)
- VLAN configuration and VLAN ID (Chapter 6.1.10)
- Port state (Port State Register, Table 31)
- HSR tag (Chapter 6.4)
- PRP trailer (Chapter 0)

Regardless of a frame being dropped or not, it is always received to the buffer memory of the forwarding core, which means that it goes through the whole inbound processing chain. If the frame is to be dropped the memory resources allocated by the frame are freed right after the reception.

The forwarding decision is presented in Figure 19. Note that also frames coming into a disabled port are received to the buffer memory, but because their forwarding decision is not to forward them to any port, they are dropped. This behavior however can be changed, and frames can be forwarded from disabled ports to other ports by using Inbound Policy (see Chapter 6.1.5).

Figure 19. Forwarding Decision



6.1.5 *Inbound Policy*

Inbound Policy checks the source and the destination MAC addresses of all the received frames. The user can configure through the register interface what kind of a treatment should frames with certain source or destination MAC addresses get. Many protocols use protocol specific multicast MAC addresses and the destination MAC address can therefore be used for forwarding those frames to CPU port and not to other ports. MAC address based authentication methods can use the Inbound Policy to enable communication from certain MAC addresses and to not forward frames coming from other MAC addresses.

The alternatives for certain source or destination MAC addresses are the following:

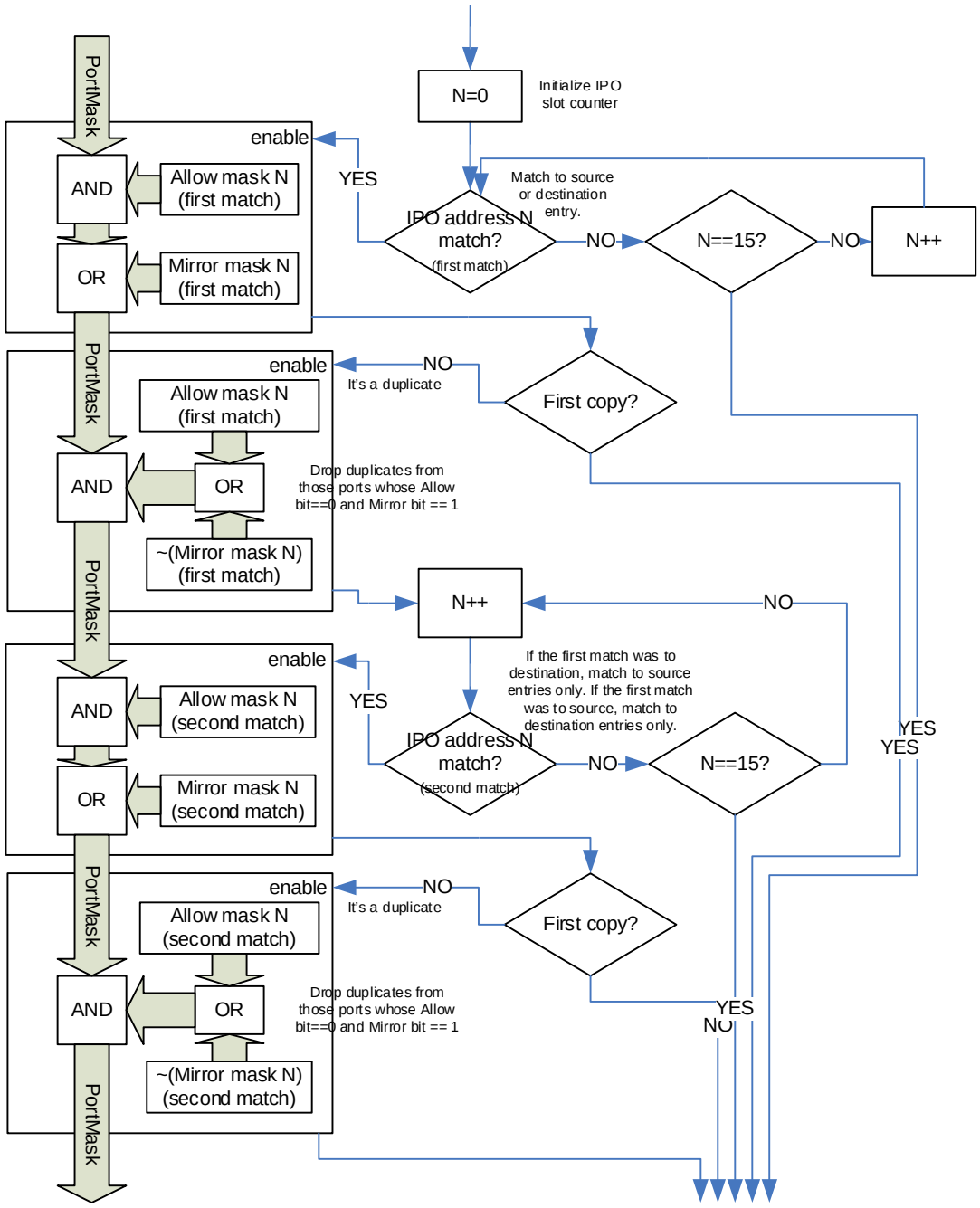
- Drop
- Allow forwarding only to certain ports
- Forced forwarding (mirroring) to certain ports
- Forward without adding HSR tag or PRP trailer

It is also possible to enable or disable:

- All unicast frames
- All multicast frames
- All broadcast frames

See Figure 20 how Inbound Policy affects the forwarding decision.

Figure 20. Inbound Policy



The MAC addresses for the inbound policy are configured using Inbound Policy (IPO) registers presented in Table 36. The inbound policy goes through the MAC addresses configured in the IPO registers in ascending order for every incoming frame. It finds the first matching rule (if any) for both source and destination MAC address in the frame and applies the both rules to the forwarding decision (in the order the matches were found). The effect of possible IPO matches to the forwarding decision can be seen in Figure 19. The other settings for the matching frame (priority, whether to send without HSR/PRP tag) are taken from the latest match (if any).

6.1.6 *Priority-setting*

When an incoming frame is VLAN tagged in case of XRS7004/7003 its priority is defined by the VLAN PCP (Priority Code Point) bits in the VLAN tag and the configured priority for the PCP (see register PORT_VLAN_PRIO). If the incoming frame had no VLAN tag, its priority is defined by port default PCP (in PORT_VLAN register) and by the above mentioned configured priority for the PCP. XRS3003 has no VLAN support, XRS3003 accepts frames with VLAN tags, but does not use the information in the tag.

Inbound policy can override the priorities for the frames according to the priority setting in Inbound Policy Configuration Register (Table 36). The priority is used by the Forwarding core to place the frames into correct transmit priority queues.

VLAN PCP (Priority Code Point) for outgoing frames is the same the frame had when it came in. For outgoing frames that came in untagged, the PCP is the default PCP of the input port.

6.1.7 *Precision Time Protocol*

RS supports PTP message transportation directly over Ethernet (IEEE 1588-2008 Annex F) and over User Datagram Protocol (UDP) over Internet Protocol version 4 (IEEE 1588-2008 Annex D). PTP Mode setting in General Register (see Table 27) selects which one of the two modes is selected. XRS3003 supports only PTP message transportation directly over Ethernet.

Inbound processing always timestamps every incoming Ethernet frame. PTP Detect block recognizes PTP version 2 event messages, determines the type of the event message (Sync, Delay_Req, Pdelay_Req and Pdelay_Resp) and calculates the offsets of the message fields in the frame, see Figure 17.

6.1.7.1 End-to-end Transparent Clock Functionality

RS implements PTP version 2 end-to-end transparent clock functionality in one-step mode with pure hardware. In Inbound processing chain PTP Detect block recognizes IEEE 1588 PTP version 2 event messages that need to have special processing inside Ethernet switches providing PTP transparent clock functionality. In practice what is done to the recognized frames is that RS adds the frame residence time inside the switch to the PTPv2 event message correctionField. The correctionField is modified in Outbound processing, in PTP Overwrite block. The modified message types include Sync, Delay_Req, Pdelay_Req and Pdelay_Resp messages.

6.1.7.2 Peer-to-Peer Transparent Clock Support

From RS point of view Peer-to-peer transparent clock support differs from End-to-end transparent clock support by only a little: In Peer-to-peer transparent clock also the line delay associated with the ingress path is added to the correction field of Sync messages. For this purpose there are registers (PTP_DELAY_NS_LOW, PTP_DELAY_NS_HIGH) for the link delay (see Chapter 6.10.3). For End-to-end transparent clock value zero is written into these registers. For Peer-to-peer transparent clock there has to be software that determines the link delay and writes it into these registers, after which RS is able to make the corrections automatically.

6.1.7.3 Ordinary and Boundary Clock Support

IEEE 1588 PTP Ordinary and Boundary clock implementations are able to achieve significantly better performance if there are hardware features to assist in timestamping and modifying of the PTP frames. Typically such features are located in the Ethernet Controller or in the Ethernet PHY. For systems that do not have such supporting features in the controller or PHY, RS port 0 has frame timestamp modification feature built-in.

6.1.7.3.1 Time Stamp Recording

At inbound and outbound of port 0 timestamps of PTP event messages (Sync, Delay_Req, Pdelay_Req and Pdelay_Resp) are written into a register. In addition to the timestamp, also part of the data in the frame is written into registers to be able to recognize the frame each timestamp corresponds to.

RS is able to store timestamps for eight frames at a time; for four frames in inbound direction and for four frames in outbound direction. The software has to acknowledge the recorded timestamps before RS is able to record more of them (Transfer bit in Transmit Timestamp Control register, see Table 28).

6.1.7.3.2 Frame Modification

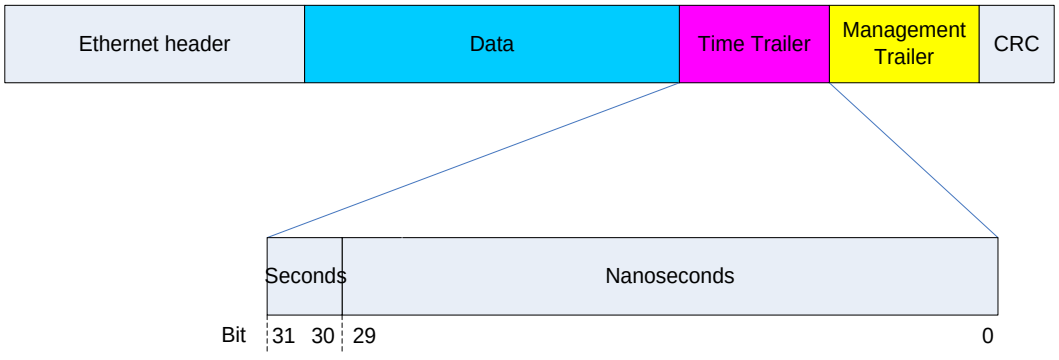
At inbound of port 0 PTP Sync messages are modified when PTP frame modification feature is enabled (Modify Sync Frames bit in General register, see Table 27). The Sync messages are modified so that the exact receive time of the frame is written to the originTimestamp Field (offset of 34 octets from the start of PTP header).

6.1.7.3.3 Time Trailer

At outbound of port 0 timestamps of PTP event messages (Sync, Delay_Req, Pdelay_Req and Pdelay_Resp) can be added to the frames themselves. When the feature is enabled (see General Register, Table 27), a Time Trailer (see Figure 21) is added between the Ethernet frame payload data and Management Trailer (6.1.8). If Management Trailer is not enabled, the Time Trailer is added between the Ethernet frame payload data and the CRC. Time Trailer contains the exact time the frame was sent out of port 0. The time presentation in the trailer includes 30 bits for nanoseconds and two bits for seconds (presenting two lowest bits of the seconds). Time Trailer is never added to other frames than PTP event messages.

Note that the nanoseconds value can at some cases be more than 999 999 999. The software using the timestamp in the trailer can handle the situation by subtracting 1 000 000 000 from the nanoseconds value and adding 1 to seconds.

Figure 21. Ethernet Frame with Time Trailer



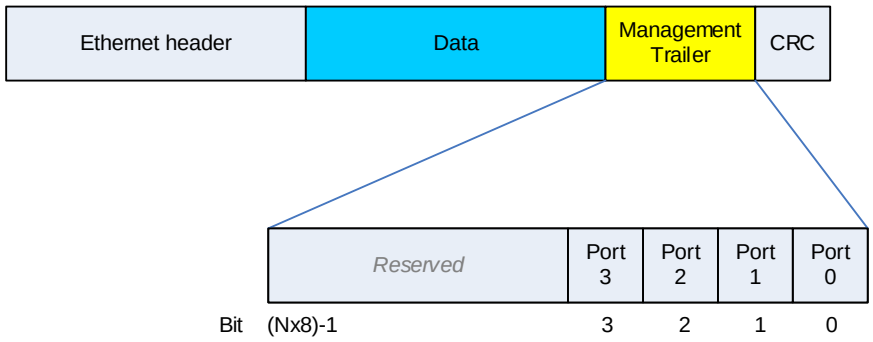
6.1.8 *Management Trailer*

The ports (only port 0 in XRS3003) can be configured to a special mode called management mode. In management mode a port supports the following feature:

- From the management port, it is possible to forward Ethernet frames to any other port independent from other configurations (MAC table, Virtual LAN configuration, Inbound Policy, Disabled ports, and so on).

When in management mode, every frame sent and received to/from the port is equipped with a management trailer. RS adds a management trailer to every frame it sends out of the port and RS expects every frame received by the port to have a management trailer. A management trailer contains information about the input/output port: from which port has the frame been received or to which port(s) it is to be sent. An Ethernet frame containing the management trailer is depicted in Figure 22.

Figure 22. Ethernet Frame with Management Trailer



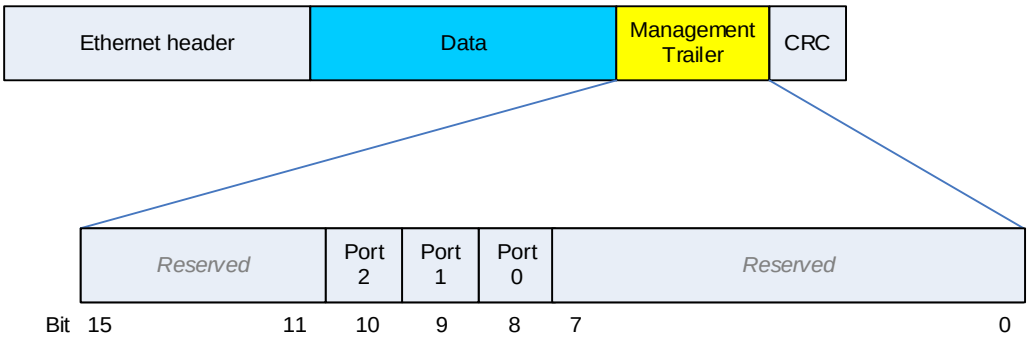
The length of the Management Trailer is one or two octets depending on the Management Trailer Length setting in GENERAL register (Table 27). For up to eight ports, the length of the management trailer can be 8 bits (one octet, in Figure 22 N=1) or 16 bits (two octets, in Figure 22 N=2). For nine to twelve port RS the length of the management trailer is 16 bits (two octets, in Figure 22 N=2). Every bit in the trailer corresponds to a certain port, starting from port number 0 in the least significant bit. Unused trailer bits are ignored by RS and if RS forwards a frame from a management port to another management port RS does not alter them.

When the host CPU wants to send for example a frame from port number 0 to port number 1 (CPU attached to port 0, port 0 in management mode), it adds a management trailer to the frame with the bit corresponding to port 1 set to one and the other bits in the management trailer set to zero. The host CPU can send a frame to multiple RS ports by setting multiple bits in the management trailer. By setting all management trailer bits (unused bits are ignored) to zero, the host CPU lets RS make the forwarding decision. Note that the host CPU sending to management port has to take care that the frame minimum length requirement of 64 bytes is met also after the Management Trailer is removed by RS.

When RS sends a frame out of a port that is in management mode, RS adds a management trailer to the frame always with one of the Port bits set. The bit corresponds to the port from which the frame was received by RS. When RS forwards a frame from a management port to another management port, it does not alter the unused (reserved) management trailer bits. There is also a special feature called Management Trailer Offset (see GENERAL register in Table 27) which allows different XRS devices to use different bits (8-bit offset) in the trailer (see example in Figure 23). This feature makes it possible to send frames for example from a CPU to certain port of

an XRS device through another XRS device. This can be useful when two XRS devices are used in a QuadBox design.

Figure 23. Management Trailer with 3-port RS and 8-bit offset



Usage of management trailer is required for example to be able to support IEEE1588 Ordinary and Boundary clock, Spanning Tree protocols (STP/RSTP), MAC address based authentication protocols, HSR/PRP supervision protocol etc. on attached CPU.

6.1.9 *MAC Address Table*

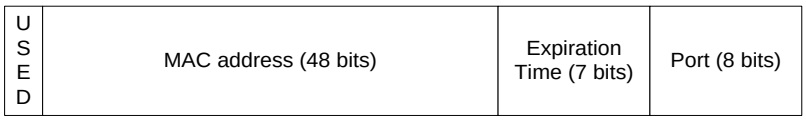
Forwarding decisions of RS are typically made by an address search to the MAC Address Table, although Inbound policy, Management trailer or VLAN settings may modify or override this decision.

In XRS3003 there is no MAC Address Table; the device has memory for only one single automatically learned MAC address. There is no MAC address aging either and the user cannot read the stored MAC address.

6.1.9.1 Address Table Entry

The structure of an entry in MAC Address Table is depicted in Figure 24.

Figure 24. MAC Address Entry



The MAC address entry contains a USED bit, the actual MAC Address, Expiration time and a port number. USED bit set to 1 indicates that the entry is currently in use. The port where the frame was received from is stored into the Port field.

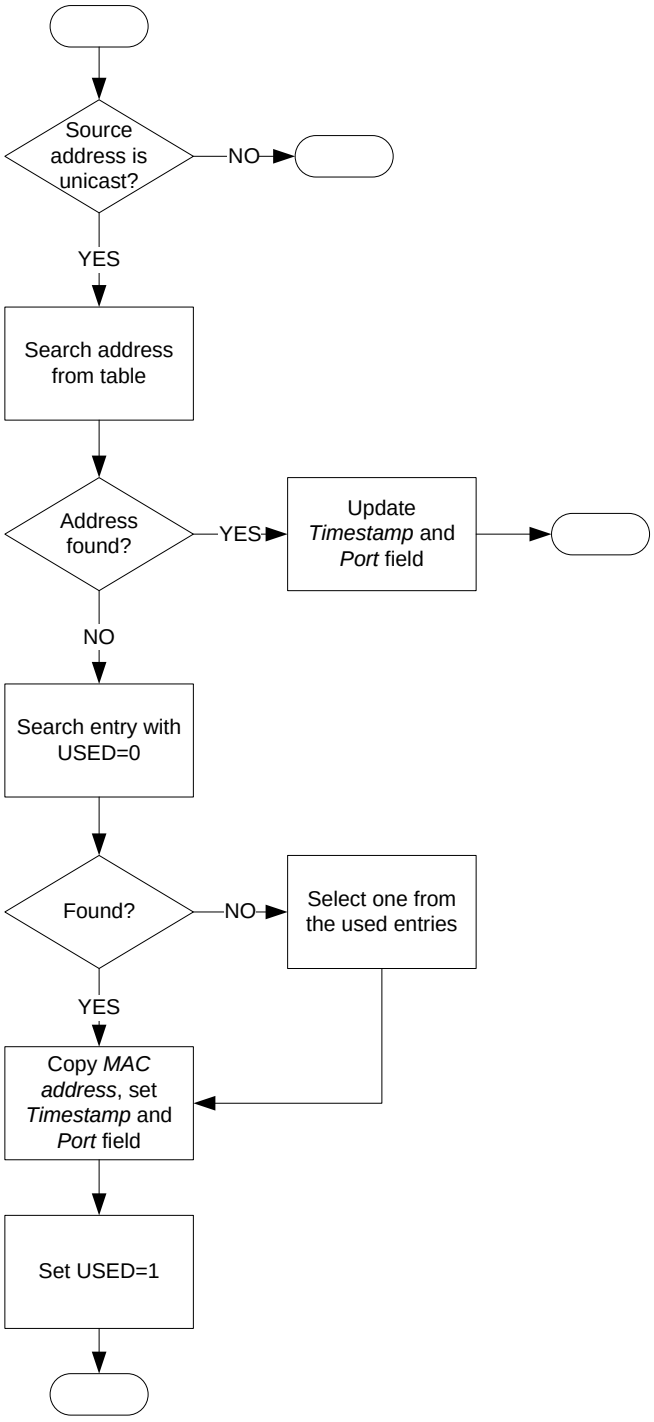
The MAC address aging time is set to Expiration time field. There is an internal counter that keeps track of the current time, and when the expiration time is updated a value of current time + address lifetime is written to the Expiration time field. The address lifetime is user configurable in the register map (see Table 27). The Expiration time value is stored in multiples of 16 seconds.

6.1.9.2 Address Learning

RS updates the MAC address table automatically according to the source MAC address information in the received Ethernet frames. The Address learning process updates the MAC Address Table when the receive port of the frame is in Forwarding or Learning state (See port state in Table 31). After learning the source MAC address from a frame received from a port that is in Learning state, the frame is dropped by the Forwarding core. If a port is in Disabled state, the MAC Address Table is not updated.

The address learning process for the source address of a received Ethernet frame is depicted in Figure 25.

Figure 25. Address Learning



The Address learning process is the following:

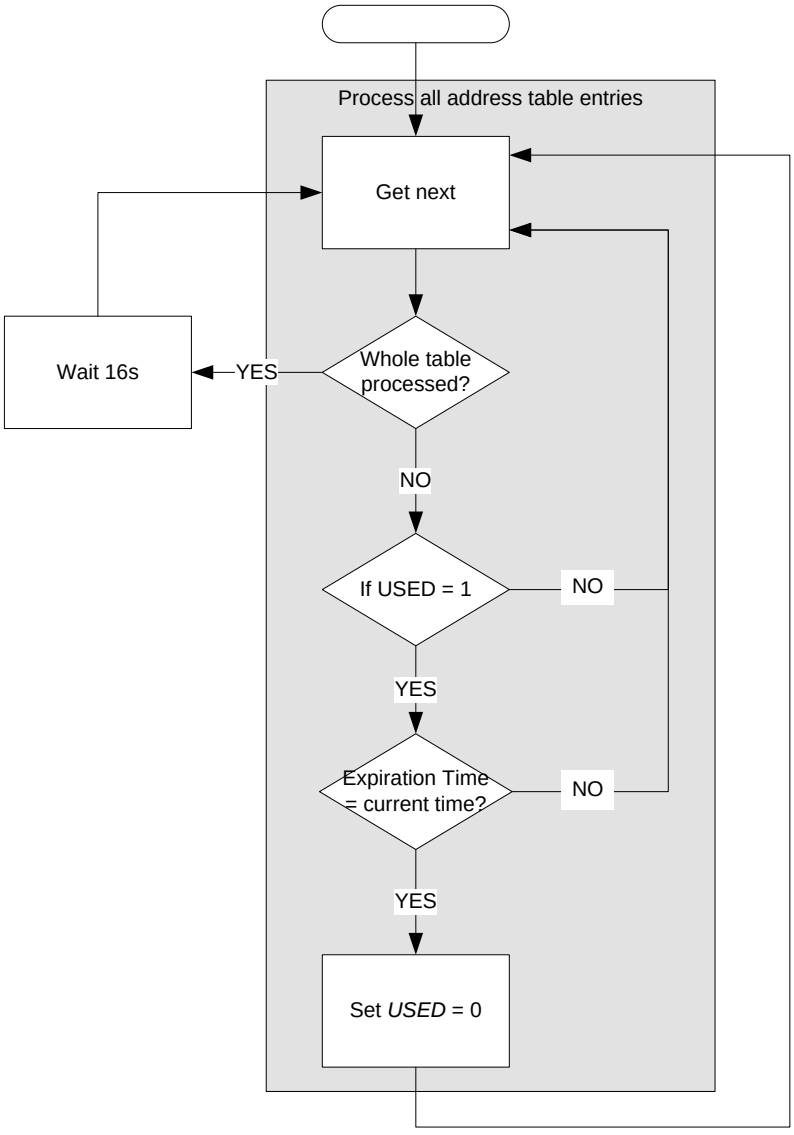
1. Check that the source address is a unicast Ethernet address.
2. Search for the address in the table
3. If the address is found, update *Expiration time* and *port* fields with expiration time and source port and exit.
4. If the address is not found, search for an address entry with *USED* bit set to 0.
5. If unused entry is not found, select one of the already used entries.

6. Copy MAC address and *Port* fields to the selected entry. Set *Expiration time* to expiration time field and set *USED* bit to 1.

6.1.9.3 Address Aging

Address aging processes for the MAC address table removes entries that are found to be expired. The Address aging process is depicted in Figure 26.

Figure 26. Address Aging

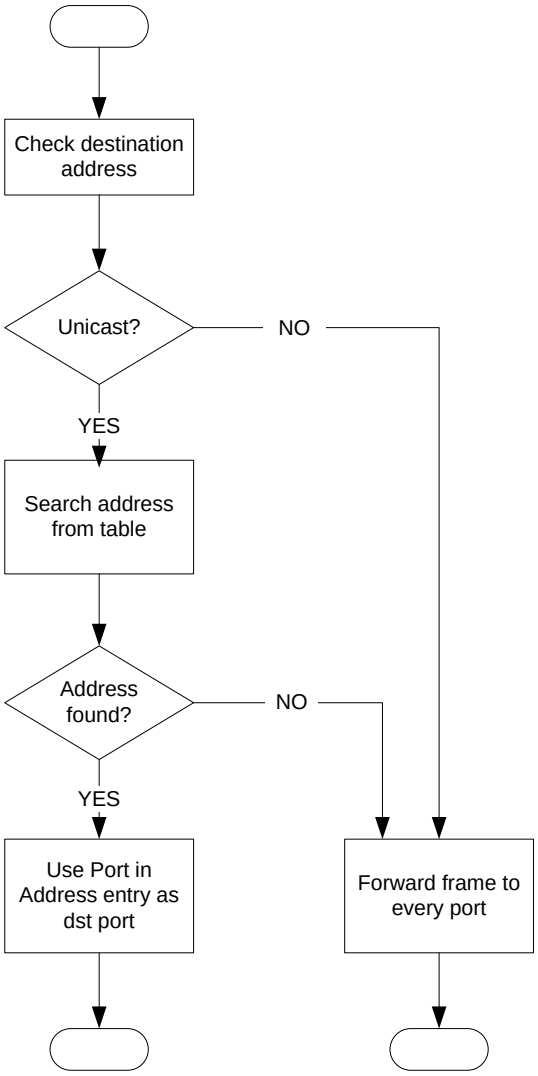


6.1.9.4 MAC Address and Forward Decision

The Address Search process searches MAC addresses from the MAC Address table. Destination MAC address is extracted from every frame received and a search is made to the MAC address table with the address. If the address is found in the MAC Address table, it means that it is known behind which port the destination node is, and the frame can be forwarded to that port. Note that Inbound policy, Management trailer or Virtual LAN settings may still override the forwarding decision made by the MAC Address Table search algorithm.

The MAC Address based forwarding decision is depicted in Figure 27. The whole forwarding decision process is presented in Figure 19.

Figure 27. MAC Address Search & Forwarding Decision



6.1.10 *Virtual LANs (VLANs)*

By using VLANs the switch can be divided into two or more virtual switches; frames are not forwarded to ports that are not configured to be members of the same VLAN. VLANs for the ports can be configured using port configuration registers (see Table 31). See Figure 19 on how VLAN configuration affects the forwarding decision.

XRS3003 does not support VLANs; XRS3003 does not use the information in VLAN tags, it forwards VLAN tagged frames like frames without VLAN tag.

6.1.11 *Forward Portmask*

Forward portmask is another way to control how frames can be forwarded between ports.

Forward portmask configuration is made using port configuration registers (see Table 31). With Forward portmask the user defines to which ports it is possible to forward frames from a port. Note that the forwarding rule can be configured separately for each direction. The Forward portmask can be useful for example in systems where one of the ports is connected to a CPU. Forward portmask can be used to force the forwarding of frames from the other ports only to the CPU port.

See Figure 19 on how Forward portmask affects the forwarding decision.

6.2 Outbound Processing

Outbound processing block transfers frames from forwarding core (buffer memory) to Ethernet medium (refer to Figure 17). Every port has its own individual Outbound processing entity. During transmitting the Outbound processing does the following:

- Timestamp frames
- Modify frames

6.2.1 *TX Post-process*

The Outbound processing blocks are chained in a row between the Forwarding core and the TX post-process. TX MII controls the rate at which the data flows through the Outbound processing path and the Forwarding core feeds the data at that rate. TX post-process transfers data to TX MII block for sending out to the medium.

6.2.2 *TX MII*

TX MII block automatically calculates and inserts correct CRC checksums to transmitted frames. Also the preamble and the Start Frame Delimiter (SFD) are automatically inserted. TX MII block indicates the start of the frame to RTC (see Figure 17). The timestamp point of a frame is defined in Figure 18.

6.2.3 *Timestamp*

The RTC uses the start-of-frame indication from TX MII block to determine the exact value of the transmit time. The transmit time is then given to the PTP overwrite block.

6.2.4 *PTP Overwrite*

Outbound processing timestamps every Ethernet frame and the difference between the outbound and the inbound time stamps (the time the frame spent inside the switch) is added to the correctionField of PTP event message headers. For Sync messages also the link delay of the ingress port is added to the correctionField. Event messages are recognized already at Inbound processing as specified in Chapter 6.1.7.

The time the frame spent inside the switch is calculated by subtracting the RX timestamp of the frame from the TX timestamp. The accuracy of the calculation is 2^-16 nanoseconds.

6.3 Forwarding Core

The Forwarding core is responsible for forwarding frames between ports; that is from the inbound processing path of a port to the outbound processing path of another. The forwarding core is common to all ports. The Forwarding core includes memory management, four transmit priority queues per port and management of the inbound and the outbound processing paths. The Forwarding core also drops frames during high load situations, when running out of buffer memory space.

6.3.1 *Memory Controller*

Memory controller block (see Figure 15) is responsible for the memory management of the buffer memory used for buffering the frames. The buffer memory is common to all the ports and there is no fixed buffer space reserved per port. Instead, a port is able to buffer more frames when other ports have shorter queues.

The Ethernet frames to be forwarded are stored into the buffer memory in one to three fragments depending on the length of the frame. The size of each fragment is 512 Bytes. This is also the size of the unit in which Memory controller manages the buffer memory. Every output port has enough bandwidth to the buffer memory to achieve wire-speed operation.

Memory controller provides an indication to the Frame Early Drop algorithm (see Chapter 6.3.3) in situations where the buffer memory is so crowded that some of the already stored frames have to be dropped to make space for new ones.

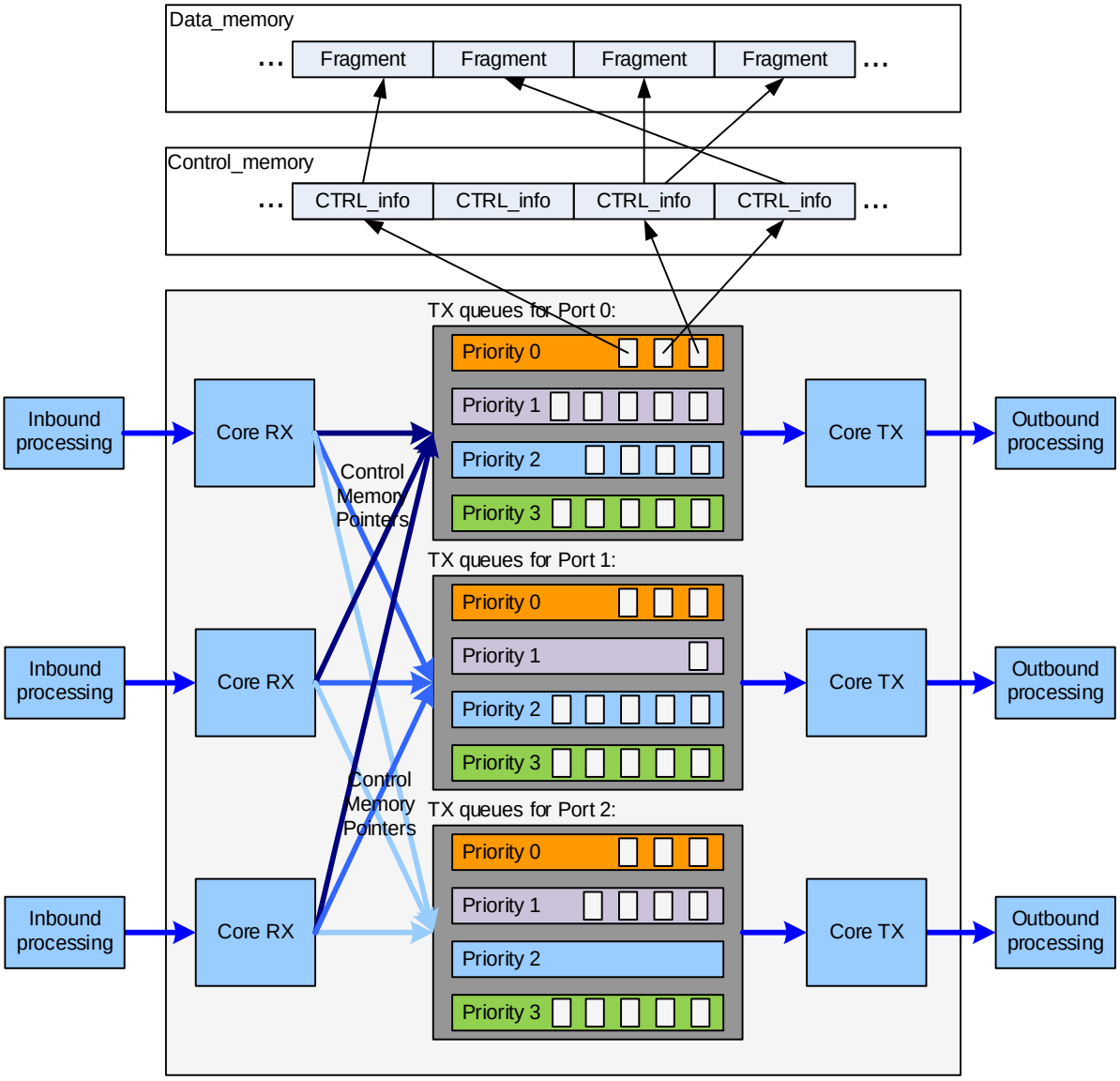
6.3.2 *Priority Queues*

There are four priority queues for every output interface. When a frame is received, the priority of the frame and the destination port(s) are determined during Inbound processing. When the Inbound processing passes the frame for the Forwarding core, the Forwarding core places the frame into the priority queue(s) that matches its priority. The priority queues are FIFO type and they actually contain only a pointer to the data of the frame (see Figure 28). The data of the frame is not moved, when the pointer moves forward in the queue.

The queues are emptied in priority order so that frames from higher priority queues are sent before any frames from any of the lower priority queues of the port.

All the priority queues have fixed length of 32 frames. If the destination priority queue is full, the frame is dropped. In case of a frame that is forwarded to multiple ports, the frame is dropped only from queue(s) that are full. When a frame is dropped because the priority queue is full, the corresponding error counter is incremented.

Figure 28. Priority Queues (three ports shown)



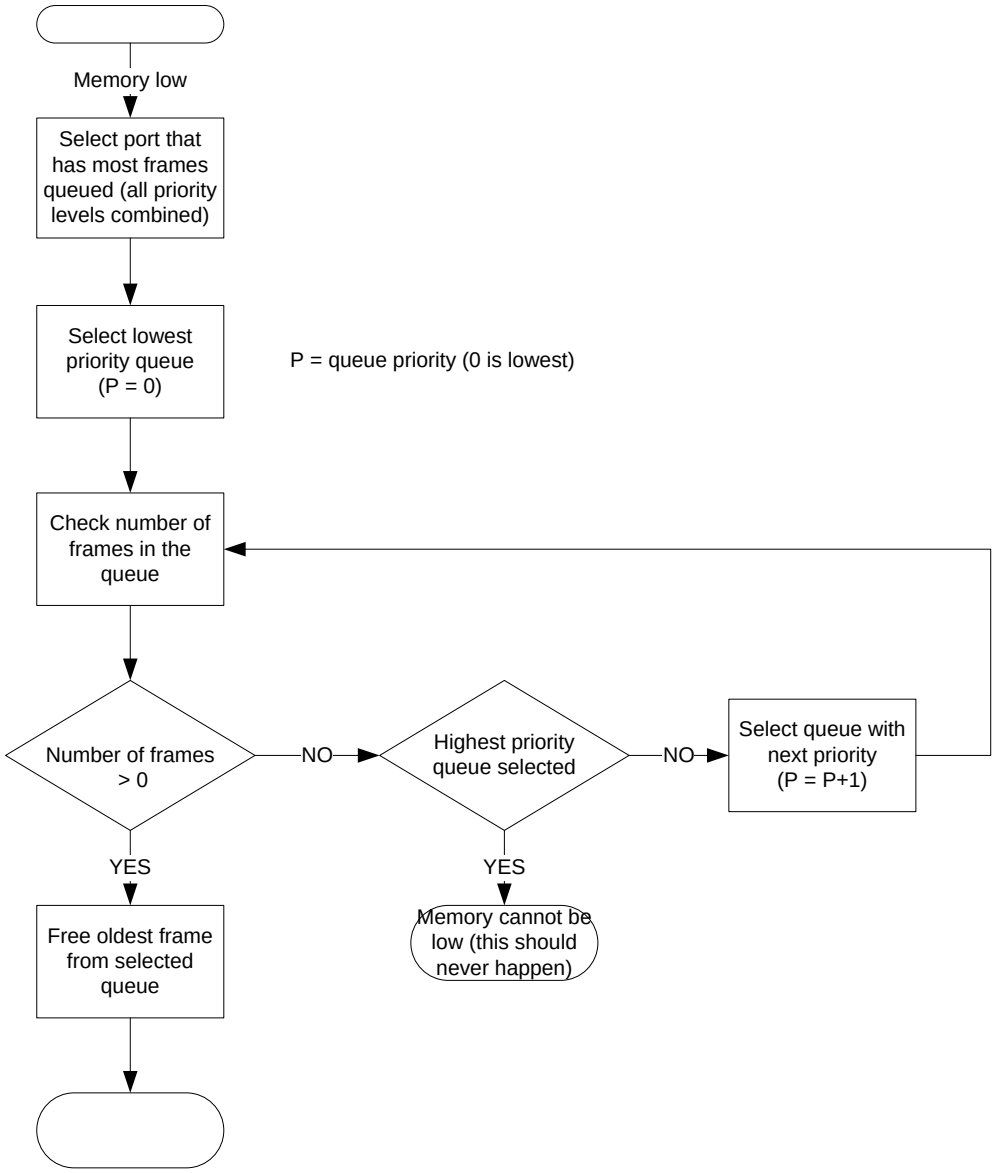
6.3.3 *Frame Early Drop*

Frame Early Drop (FED) algorithm deallocates buffer memory when RS encounters heavy load and buffer memory is about to cease. An indication that the memory is going to be fully used soon is got from Memory controller block.

One reason for using FED algorithm is to ensure that frames buffered for some output ports do not block traffic for the other ports. Buffering too much frames for some ports would in worst case cause all the incoming frames to be dropped. This is because the buffer memory is common to all ports. So the Memory controller block triggers the Frame Early Drop algorithm early and frequently enough to ensure that there will always be enough buffer memory to be able to receive all the incoming frames.

The FED algorithm used also guarantees that lower priority frames for a port are dropped before higher priority ones for the port. Another effect is that when buffer memory consumption is high, frames are dropped early enough to slow down TCP connections (selective dropping for oldest frames) to prevent total congestion.

Figure 29. Frame Early Drop Algorithm



The Frame Early Drop algorithm is presented in Figure 29. The algorithm is run when buffer memory resources are running low. The algorithm selects one frame to be dropped and drops it.

The Frame Early Drop algorithm selects first the output port that has the most frames queued. Then it selects the lowest priority queue of that port that has queued frames it, and removes the oldest frame from that queue. After dropping the frame, the corresponding error counter is incremented.

6.4 HSR (High-availability Seamless Redundancy)

HSR specific features of RS include:

- Automatic insertion of HSR tag
- Automatic removal of HSR tag
- Automatic duplicate generation for HSR ports

- Automatic duplicate detection and removal for HSR ports

At input the HSR tag is always removed if the port is in HSR mode. At output a HSR tag is added if the output port is in HSR mode.

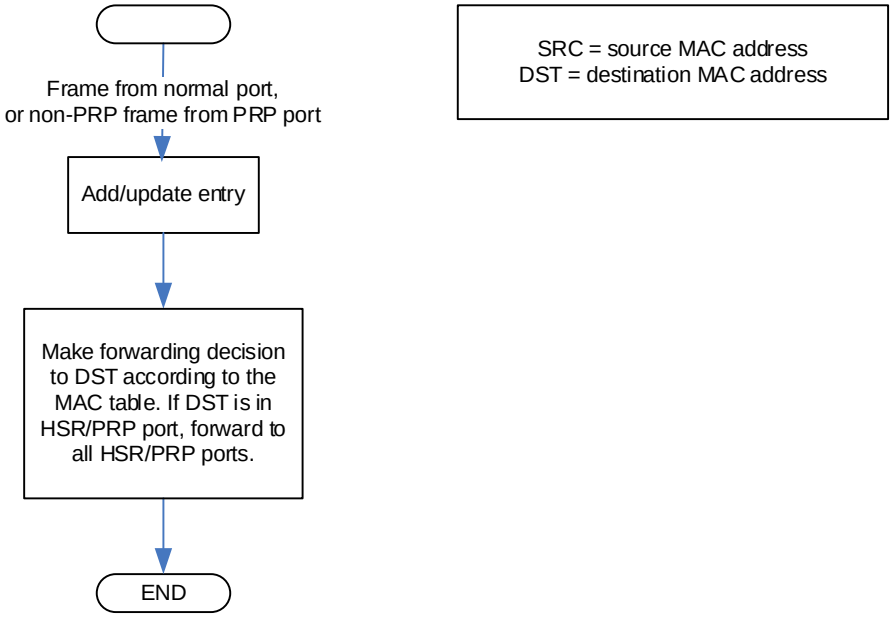
The HSR mode for a port and the other HSR specific setting are configured using HSR/PRP registers (see Table 32).

6.4.1 *Forwarding of HSR Frames*

When forwarding frames in HSR-enabled switch there are basically two different cases: the frame is either coming in from a HSR redundant port (typically ring port) or it is coming in from an interlink port. The interlink port can be either in HSR, PRP or normal (non-HSR, non-PRP) mode.

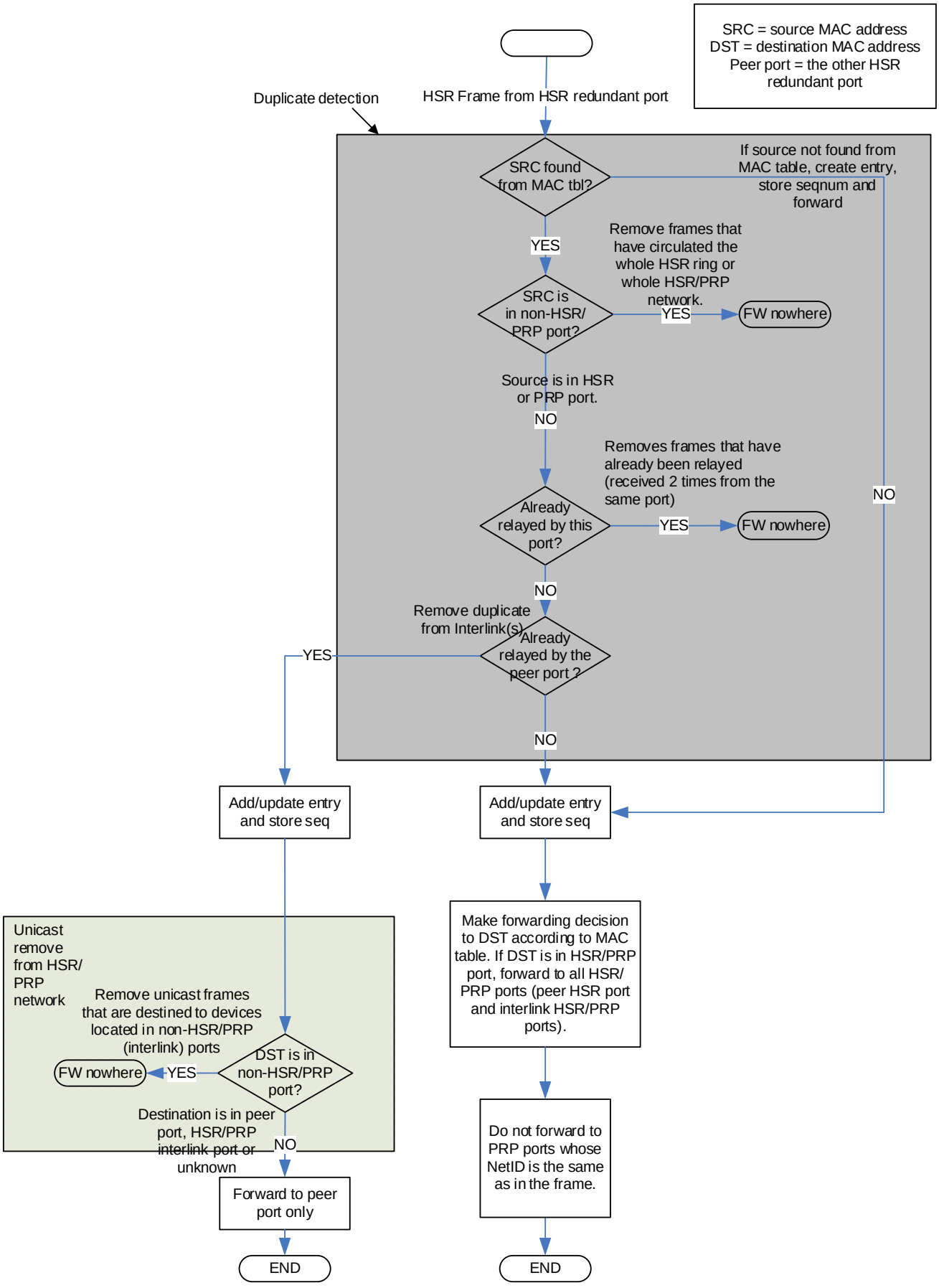
If a frame comes in from a normal (non-HSR, non-PRP) interlink port it is forwarded as presented in Chapter 6.1.4, but when it is to be forwarded to HSR-redundant port the frame is duplicated and sent out from the both redundant ports (see Figure 30).

Figure 30. Frame from normal port (or non-PRP frame from PRP port)



The forwarding logic for frames coming in from a redundant port is more complicated, because of duplicate detection and removal. The forwarding logic for frames coming in from an HSR redundant port is presented in Figure 31.

Figure 31. HSR Redundant Port Forwarding Logic



For HSR frames received from a HSR port, it is first checked if the source MAC address exists in the MAC address table and if the source node is located in non-HSR/PRP port. The duplicate detection is then done by first looking at the stored HSR sequence numbers for the other HSR redundant port: if one matches with the incoming frame's HSR Tag's sequence number, we have a duplicate. Additionally, it is checked whether a frame with this same sequence number and source MAC address, that in from this same port has already been forwarded, in which case the frame is circulating in the ring/network and has to be deleted. If the frame is neither duplicate nor circulating, it is forwarded towards its destination(s).

Multicast and broadcast frames always circulate the whole ring. The duplicate detection for multicast and broadcast frames is made in two phases: by first looking if the frame has already been forwarded into this direction. If the answer is yes, the frame already circulated the whole ring and it is dropped (note that checking whether the source address is behind a non-HSR port probably drops the frame earlier). The next step is to see whether the frame was already received from the other redundant port and has therefore already been forwarded to the interlink ports. If not, the frame is forwarded to all the other ports (except the input port). Otherwise the frame is forwarded only to the other redundant port.

6.4.2 *HSR Port Modes*

HSR standard defines one mandatory operation mode and four optional modes. The default mode is called mode H, which is normal HSR tagged forwarding.

In the optional mode N, traffic is not forwarded between HSR redundant ports. The mode N can be configured using PORT_FWD_MASK register (see Table 31) by disabling forwarding between HSR redundant ports.

The configuration procedure is the following:

1. Disable both redundant ports (Table 31, PORT_STATE register)
2. Change mode (Table 32, HSR_CFG register) for both redundant ports
3. Configure PORT_FWD_MASK (Table 31, PORT_FWD_MASK register)
4. Enable both configured ports (Table 31, PORT_STATE register)

6.5 PRP (Parallel Redundancy Protocol)

PRP specific features of RS include:

- Automatic insertion of PRP trailer
- Automatic removal of PRP trailer
- Automatic duplicate generation for PRP ports
- Automatic duplicate detection and removal for PRP ports

During inbound processing the PRP trailer is removed if the port is in PRP mode and if the frame has one. In output a PRP trailer is added if the output port is in PRP mode. Frames coming in from a PRP enabled port that do not have a PRP trailer are accepted, because they can be from a SAN (Singly Attached Node) that does not support PRP.

The PRP mode for a port and the other PRP specific settings are configured using HSR/PRP registers (see Table 32). Also PORT_FWD_MASK must be configured to prevent forwarding between PRP redundant ports.

The configuration procedure is the following:

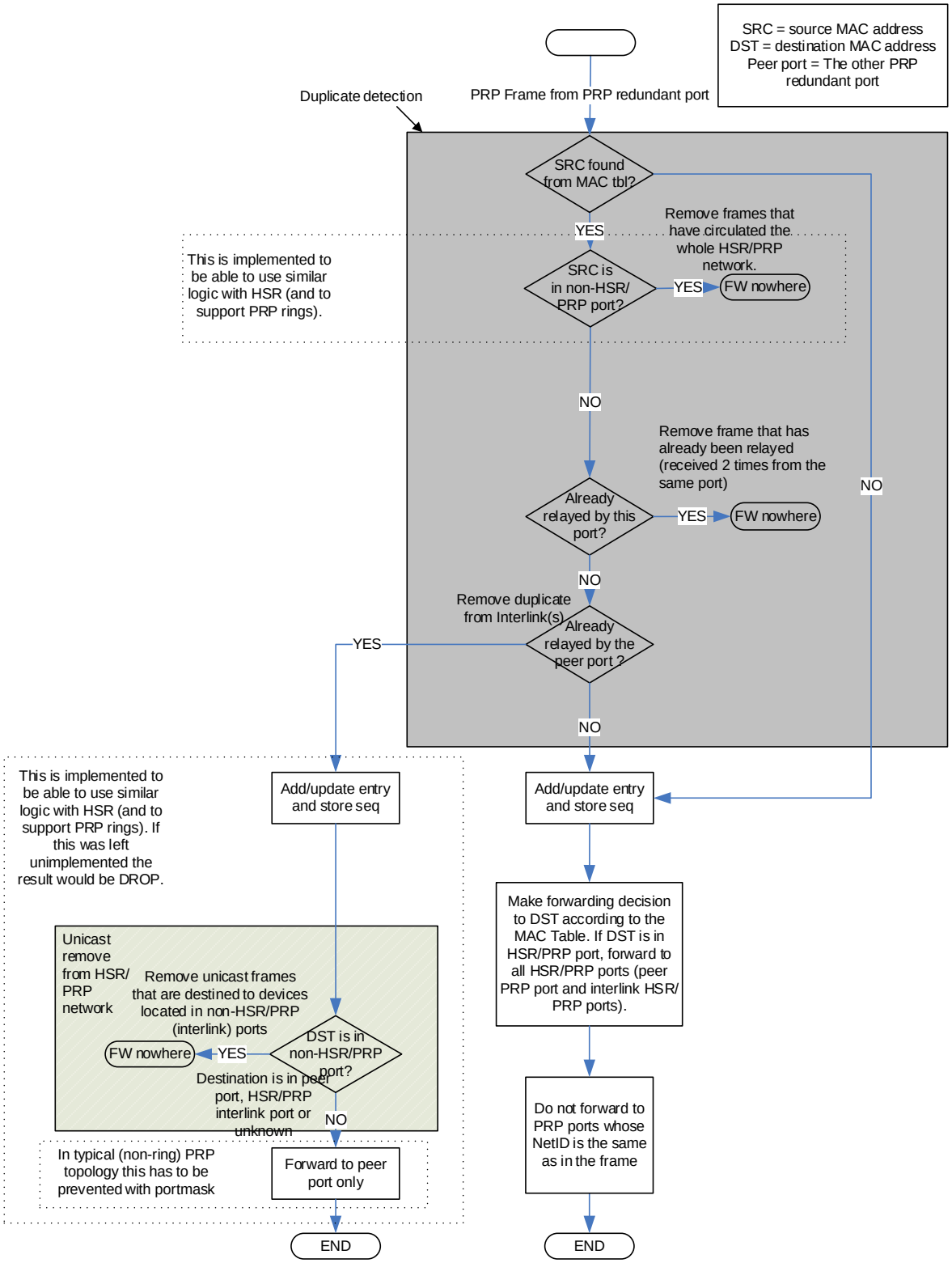
1. Disable both redundant ports (Table 31, PORT_STATE register)
2. Change mode (Table 32, HSR_CFG register) for both redundant ports
3. Configure PORT_FWD_MASK to prevent forwarding between PRP ports (Table 31, PORT_FWD_MASK register)
4. Enable both configured ports (Table 31, PORT_STATE register)

6.5.1 *Forwarding of PRP Frames*

When a frame comes in from a normal (non-PRP, non-HSR) interlink port it is forwarded as presented in Chapter 6.1.4, but if it is to be forwarded to a PRP redundant port, the frame is duplicated and sent out of the both PRP redundant ports.

The forwarding logic for frames coming in from a PRP redundant port is more complicated, because of duplicate detection and removal. The forwarding logic for PRP frames coming in from a PRP redundant port is presented in Figure 32.

Figure 32. PRP Redundant Port Forwarding Logic



For PRP frames received from a PRP port, it is first checked if the source MAC address exists in the MAC address table. The duplicate detection is then done by looking at the stored PRP sequence numbers for the other PRP port:

if one matches with the incoming frame's PRP trailer's sequence number, we have a duplicate. Additionally, it is checked whether a frame with this same sequence number and source MAC address, coming in from this same port has already been forwarded. If the frame is not duplicate it is forwarded towards its destination(s).

6.6 HSR/PRP interoperability

In PRP frame LanId identifies into which LAN (LAN_A, LAN_B) the PRP frame is sent. In HSR frame LanId identifies from which PRP LAN (LAN_A, LAN_B) the frame came into the HSR Network. In addition to LanId, in HSR there is NetId identifying the PRP network where from the frame originated. The idea of NetId is that the frame is not forwarded from the HSR network back to the PRP network it came originally from. The definitions of the different identifiers used in HSR and PRP can be found in Table 23. Table 24 presents how the identifier fields are handled in RS.

Table 23. Definition of LanID, NetId, ring NetId and PathId

Term	HSR	PRP
LanId	Lowest bit of the PathId. Identifies whether connected to PRP LAN A or LAN B.	4-bit field in PRP tag identifying the LAN. Either A (1010) or B (1011).
NetId	3-bit identification number for attached PRP network. 3 highest bits of PathId.	-
ring NetId	NetId for the frames originated from the RedBox itself.	-
PathId	4-bit field in HSR header. NetId + LanId.	-

Table 24. Resulting LanId and NetId

Input port mode	Output port mode	Affect to the Forwarding Decision	Resulting LanID in the frame	Resulting NetID in the frame
normal	normal	-	-	-
normal	HSR	-	LanId configured for the input port.	NetID configured for the input port.
normal	PRP	-	0xA or 0xB, according to LanId bit configured for the output port.	-
HSR	normal	-	-	-
HSR	HSR	-	unchanged	unchanged
HSR	PRP	if NetID of the frame matches the NetID of the output port, drop the frame.	0xA or 0xB, according to LanId bit configured for the output port.	-
PRP	normal	-	-	-

Input port mode	Output port mode	Affect to the Forwarding Decision	Resulting LanID in the frame	Resulting NetID in the frame
PRP	HSR	-	LanId configured for the input port.	NetID configured for the input port.
PRP	PRP	-	0xA or 0xB, according to LanId bit configured for the output port.	-

6.7 Allowed Port Modes

Not every port of XRS is allowed to be configured in every mode, Table 25 lists the allowed modes for each port.

Table 25. Allowed Port Modes

Device	Port	non-HSR, non-PRP	HSR Redundant	PRP Redundant	HSR Interlink	PRP Interlink	Management
XRS7004	0	Yes	-	-	-	-	Yes
	1	Yes	Yes	Yes	-	-	Yes
	2	Yes	Yes	Yes	-	-	Yes
	3	Yes	-	-	Yes	Yes	Yes
XRS7003	0	Yes	-	-	-	-	Yes
	1	Yes	Yes	Yes	-	-	Yes
	2	Yes	Yes	Yes	-	-	Yes
XRS3003	0	Yes	-	-	-	-	Yes
	1	Yes	Yes	Yes	-	-	-
	2	Yes	Yes	Yes	-	-	-

6.8 Software Reset

Software reset is made by writing value 1 to the Software reset bit (see Table 27) in General Register. After reset command RS cancels all of its current operations and waits until all of its state machines have returned to their reset states. After the reset has completed, RS clears the Software reset bit in the register. After software reset RS is in the same state as after hardware reset.

6.9 Switch Configuration Registers

Table 26 presents the switch configuration register groups. These registers configure the operation of the switch core.

Table 26. Switch Configuration Register Groups

Address Offset	Acronym	Register group description	Section	Table
0x0000	SWITCH	General switch configuration registers	0	Table 27
0x2000	TS	Frame Timestamp registers	6.9.2	Table 28
0x4000	VLAN	Virtual LAN Configuration Registers (not in XRS3003)	6.9.3	Table 29

6.9.1 *General Switch Configuration Registers*

The General switch configuration registers are presented in Table 27.

Table 27. General Switch Configuration Registers

Address	Register	Description			
SWITCH+ 0x0000 … SWITCH+ 0x000E	*Reserved*	*Reserved*			
SWITCH+ 0x0010	GENERAL	Reset: 0 x 00 00 General control bits for RS.			
		Bits	1-0:	RO	*Reserved*
		Bits	3-2:	R/W	Management Trailer Length Defines the length of the management trailer for ports that are in management mode. 0 = 8 bits 1 = 16 bits (reserved in XRS3003) 2 = reserved 3 = reserved
		Bits	5-4:	R/W	Management Trailer Offset 0 = normal operation 1 = 8 bit offset. This causes bit 8 in the management trailer to relate to port 0 (instead of port 8), bit 9 to relate to port 1 (instead of port 9), and so on. Bits 0 to 7 in the trailer are ignored. This feature is useful (only) when two XRS devices are connected to each other using interfaces that are in management mode, as it makes different XRS devices to use different bits in the trailer. Use only when Management Trailer Length is configured to 16 bits. 2 = reserved 3 = reserved
		Bit	6:	RO	*Reserved*

Address	Register	Description			
		Bit	7:	R/W	Disable support for independent VLANs 0 = Support for independent VLANs enabled. If destination MAC is registered to a port where to forwarding is not allowed, the frame is forwarded to all ports where allowed. Do not use this mode when HSR/PRP is enabled. 1 = Support for independent VLANs disabled. Frame is dropped when destination MAC is registered to a port where to forwarding is not allowed. See also forwarding decision in Figure 19.
		Bit	8:	RO	*Reserved*
		Bit	9:	R/W	Time Trailer When enabled, Time Trailer (Chapter 4.1.7.3.3) is added to PTP event message frames at outbound of port 0. 0 = disabled 1 = enabled It is not allowed to enable Time Trailer and HSR/PRP mode (HSR_CFG register) for port 0 at the same time.
		Bit	10:	R/W	Modify Sync frames When enabled, originTimestamp field in the Sync messages in port 0 inbound are modified to contain the receive time of the frame. 0 = disabled 1 = enabled
		Bits	12-11:	R/W	PTP Mode 00 = PTP over UDP/IPv4 (reserved in XRS3003) 01 = reserved 10 = PTP over Ethernet 11 = reserved

Address	Register	Description			
		Bit	13:	R/W	Cut-Through Enables Cut-Through operation between HSR redundant ports. Cut-through operation is possible only between port 1 (RGMII1) and port 2 (RGMII2). When enabling Cut-Through, the ports 1 and 2 must be configured to HSR redundant mode, they must not be in management mode and port 3 cannot be configured as HSR/PRP interlink port. 0 = Store-and-Forward operation 1 = Cut-Through operation To prevent broken frames from looping indefinitely, this bit self-clears in case of a receive error (RX_UNDERSIZE, RX_FRAGMENTS, RX_OVERSIZE, RX_JABBER, RX_ERR or RX_CRC).
		Bit	14:	R/SC	Clear MAC address table Writing 1 to this bit clears entries from the MAC address table. The value in register MT_CLEAR_MASK defines which entries are cleared. RS clears this bit when done. This bit is reserved in XRS3003.
		Bit	15:	R/SC	Software reset Writing 1 to this bit starts a software reset. RS clears the bit when reset is completed.
SWITCH+ 0x0012	*MT_CLEAR_MASK*	Reset: 2^(NUMBER_OF_PORTS)-1 MAC Table Clear Mask - Defines which entries are cleared from the MAC address table when MAC address table clear command is given (bit 14 in GENERAL register) - The reset value is such that all the entries in the table are cleared - If entries of a HSR port (redundant or HSR-interlink) are cleared, entries of all the HSR ports must be cleared at the same time - If entries of a PRP port are cleared, entries of all the PRP ports must be cleared at the same time. - There is no need for clearing entries of HSR/PRP ports if one of the HSR/PRP ports goes down. Clearing the entries may cause frames loss or duplicates. - This register is reserved in XRS3003.			

<table>
<tr><th>Address</th><th>Register</th><th colspan="4">Description</th></tr>
<tr><td></td><td></td><td>Bits</td><td>15-0:</td><td>R/W</td><td>MAC Table Clear Mask.
Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on.
0 = Entries registered to this port are not cleared or the port does not exist
1 = Entries registered to this port are cleared</td></tr>
<tr><td>SWITCH+ 0x0014 … SWITCH+ 0x001E</td><td>Reserved</td><td colspan="4">Reserved</td></tr>
<tr><td>SWITCH+ 0x0020</td><td>ADDRESS_AGING</td><td colspan="4">Reset: 0 x 00 12
Configuration register for address aging functionality of the MAC Address table.
<table><tr><td>Bits</td><td>6-0:</td><td>R/W</td><td>Address LifeTime
Lifetime of automatically learned addresses. 0=16s, 1=32s, 2=48s, … , 127=2048s
This setting defines also the ProxyNodeTableForgetTime for HSR/PRP.</td></tr><tr><td>Bit</td><td>7:</td><td>RO</td><td>Reserved</td></tr><tr><td>Bits</td><td>10-8:</td><td>R/W</td><td>EntryForgetTime
Timer value for HSR/PRP duplicate discard algorithm, see HSR/PRP specification [7].
0=10ms, 1=20ms, 2=40ms, 3=80ms, 4=160ms, 5=320ms, 6=640ms, 7=1280ms.</td></tr><tr><td>Bits</td><td>15-11:</td><td>RO</td><td>Reserved</td></tr></table></td></tr>
<tr><td>SWITCH+ 0x0022… SWITCH+ 0x0026</td><td>Reserved</td><td colspan="4">Reserved</td></tr>
</table>

Address	Register	Description			
SWITCH+ 0x0028	*TS_CTRL_TX*	Reset: 0 x 00 00 Timestamper Control, Transmit Side			
		Bit	0:	R/SC	Transfer TXTS0 User sets this bit to allow RS to store information of a PTP Event message to registers TX_TS_0_NS_LO, TX_TS_0_NS_HI, TX_TS_0_S_LO, TX_TS_0_S_HI and TX_TS_0_HDR_0…29. RS clears this bit after writing the information to the registers.
		Bit	1:	R/SC	Transfer TXTS1
		Bit	2:	R/SC	Transfer TXTS2
		Bit	3:	R/SC	Transfer TXTS3
		Bits	15-4:	RO	*Reserved*
SWITCH+ 0x002A	*TS_CTRL_RX*	Reset: 0 x 00 00 Timestamper Control, Receive Side			
		Bit	0:	R/SC	Transfer RXTS0 User sets this bit to allow RS to store information of a PTP Event message to registers RX_TS_0_NS_LO, RX_TS_0_NS_HI, RX_TS_0_S_LO, RX_TS_0_S_HI and RX_TS_0_HDR_0…29. RS clears this bit after writing the information to the registers.
		Bit	1:	R/SC	Transfer RXTS1
		Bit	2:	R/SC	Transfer RXTS2
		Bit	3:	R/SC	Transfer RXTS3
		Bits	15-4:	RO	*Reserved*

<table>
<tr><th>Address</th><th>Register</th><th>Description</th></tr>
<tr><td>SWITCH+
0x002C</td><td>INT_MASK</td><td>Reset: 0 x 00 00
Interrupt mask. An external interrupt is activated when at least one of the following Interrupt Mask bits is set and the corresponding Interrupt Status bit is 1.
<table>
<tr><td>Bit</td><td>0:</td><td>R/W</td><td>TX Timestamp
Indicates that information of a PTP event message has been written to TX timestamp registers.</td></tr>
<tr><td>Bit</td><td>1:</td><td>R/W</td><td>RX Timestamp
Indicates that information of a PTP event message has been written to RX timestamp registers.</td></tr>
<tr><td>Bit</td><td>2:</td><td>R/W</td><td>RX Error
Indicates that an RX error has happened and that the corresponding counter has been incremented. The number of erroneous frames can be read from counter registers RX_UNDERSIZE, RX_FRAGMENTS, RX_OVERSIZE, RX_JABBER, RX_ERR and RX_CRC (see Table 35). RX_WRONGLAN does not cause an interrupt.</td></tr>
<tr><td>Bit</td><td>3:</td><td>R/W</td><td>Congested
Indicates that at least one frame was dropped due to congestion. Number of frames dropped can be read from registers PRIQ_DROP and EARLY_DROP.</td></tr>
<tr><td>Bits</td><td>15-4:</td><td>RO</td><td>Reserved</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Address</th><th>Register</th><th>Description</th></tr>
<tr><td>SWITCH+
0x002E</td><td>INT_STATUS</td><td>Reset: 0 x 00 00
Interrupt Status. An external interrupt is activated when at least one of the following Interrupt Status -bits is set and the corresponding Interrupt Mask bit is 1.The interrupt status bit in this register is updated independent from the corresponding Interrupt Mask bit.
<table>
<tr><td>Bit</td><td>0:</td><td>R/C</td><td>TX Timestamp
Indicates that information of a PTP event message has been written to TX timestamp registers.</td></tr>
<tr><td>Bit</td><td>1:</td><td>R/C</td><td>RX Timestamp
Indicates that information of a PTP event message has been written to RX timestamp registers.</td></tr>
<tr><td>Bit</td><td>2:</td><td>R/C</td><td>RX Error
Indicates that an RX error has happened and that the corresponding counter has been incremented. The number of erroneous frames can be read from counter registers RX_UNDERSIZE, RX_FRAGMENTS, RX_OVERSIZE, RX_JABBER, RX_ERR and RX_CRC (see Table 35) RX_WRONGLAN does not cause an interrupt.</td></tr>
<tr><td>Bit</td><td>3:</td><td>R/C</td><td>Congested
Indicates that at least one frame was dropped due to congestion. Number of frames dropped can be read from registers PRIQ_DROP and EARLY_DROP</td></tr>
<tr><td>Bits</td><td>15-4:</td><td>RO</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>SWITCH+
0x0030 …
SWITCH+
0x01FE</td><td>Reserved</td><td>Reserved</td></tr>
</table>

Address	Register	Description
SWITCH+ 0x0200	MAC_TABLE0	Reset: 0 x 00 00 MAC Table Read 0. In XRS3003 this register is reserved. Bits 3-0: RO — Port Number. The port where to the address is registered. Bits 14-4: RO — *Reserved* Bit 15: R/SC — Transfer. Write 1 to this bit to enable fetching of the next MAC address entry to registers MAC_TABLE0… MAC_TABLE3. Value 0 indicates that the fetch is completed. The first entry fetched is the first entry in the table. Fetched MAC address value FF:FF:FF:FF:FF:FF indicates that the latest entry fetched was the last entry in the table and that the next entry to be fetched is the first entry in the table.
SWITCH+ 0x0202	MAC_TABLE1	Reset: 0 x FF FF MAC Table Read 1. The first two bytes of the MAC address. MAC address value FF:FF:FF:FF:FF:FF indicates the end of the table. In XRS3003 this register is reserved. Bits 7-0: RO — 1st octet (XX:XX:XX:XX:XX:XX) Bits 15-8: RO — 2nd octet (XX:XX:XX:XX:XX:XX)
SWITCH+ 0x0204	MAC_TABLE2	Reset: 0 x FF FF MAC Table Read 2. The two bytes in the middle of the MAC address. MAC address value FF:FF:FF:FF:FF:FF indicates the end of the table. In XRS3003 this register is reserved. Bits 7-0: RO — 3rd octet (XX:XX:XX:XX:XX:XX) Bits 15-8: RO — 4th octet (XX:XX:XX:XX:XX:XX)
SWITCH+ 0x0206	MAC_TABLE3	Reset: 0 x FF FF MAC Table Read 3. The last two bytes of the MAC address. MAC address value FF:FF:FF:FF:FF:FF indicates the end of the table. In XRS3003 this register is reserved. Bits 7-0: RO — 5th octet (XX:XX:XX:XX:XX:XX) Bits 15-8: RO — 6th octet (XX:XX:XX:XX:XX:XX)

6.9.2 *Frame Timestamp Registers*

The Frame Timestamp registers are presented in Table 28. There are four sets of registers for TX and four sets of registers for RX into which information of four PTP event messages at a time can be stored. RS uses the register sets (consisting of registers TX_TS_X_NS_LO, TX_TS_X_ NS_HI, TX_TS_X_S_LO, TX_TS_X_ S_HI, TX_TS_X_HDR 0… TX_TS_X_HDR 29) in ascending order 0,1,2,3,0,1,2,3,0,1,2,3… The user has to set the

corresponding Transfer bit in the Control register (TS_CTRL_TX, TS_CTRL_RX, see Table 28) to 1 before RS is able to use the register set. After filling the information to the register set, RS clears the Transfer bit. If the Transfer bit corresponding to the register set that RS is going to use next is not set, the timestamp information of the next event message is lost (the frame itself is forwarded normally).

Timestamps of PTP messages are stored only for port 0.

Table 28. Frame Timestamp Registers

Address	Register	Description
TS+ 0x0000	*TX_TS_0_NS_LO*	Reset: 0 x 00 00 Transmit TimeStamp 0 Nanoseconds low. Bits 15-0: RO — Nanoseconds, bits 15:0 Nanoseconds of the timestamp of the frame (lowest bits).
TS+ 0x0002	*TX_TS_0_NS_HI*	Reset: 0 x 00 00 Transmit TimeStamp 0 Nanoseconds high. Bits 13-0: RO — Nanoseconds, bits 29:16 Nanoseconds of the timestamp of the frame (highest bits). The nanoseconds value can at some cases be more than 999 999 999. The software using the timestamp can handle the situation by subtracting 1 000 000 000 from the nanoseconds value and adding 1 to seconds. Bits 15-14: RO — *Reserved*
TS+ 0x0004	*TX_TS_0_S_LO*	Reset: 0 x 00 00 Transmit TimeStamp 0 Seconds low. Bits 15-0: RO — Seconds, bits 15:0 Seconds of the timestamp of the frame (lowest bits).
TS+ 0x0006	*TX_TS_0_S_HI*	Reset: 0 x 00 00 Transmit TimeStamp 0 Seconds high. Bits 15-0: RO — Seconds, bits 31:16 Seconds of the timestamp of the frame (the next lowest bits).
TS+ 0x0008… TS+ 0x000C	*Reserved*	*Reserved*

Address	Register	Description
TS+ 0x000E	*TX_TS_0_HDR 0*	Reset: 0 x 00 00 Transmit TimeStamp 0 Message Header 0. Bits 15-0: RO PTP Message bytes 1:0 First two bytes of the PTP message header.
TS+ 0x0010	*TX_TS_0_HDR 1*	Reset: 0 x 00 00 Transmit TimeStamp 0 Message Header 1. Bits 15-0: RO PTP Message bytes 3:2 Bytes 3:2 of the PTP message header.
TS+ 0x0012	*TX_TS_0_HDR 2*	Reset: 0 x 00 00 Transmit TimeStamp 0 Message Header 2. Bits 15-0: RO PTP Message bytes 5:4 Bytes 5:4 of the PTP message header.
TS+ 0x0014… TS+ 0x0046	*TX_TS_0_HDR N*	Reset: 0 x 00 00 Transmit TimeStamp 0 Message Header N. Bits 15-0: RO PTP Message bytes ((N*2)+1):(N*2)
TS+ 0x0048	*TX_TS_0_HDR 29*	Reset: 0 x 00 00 Transmit TimeStamp 0 Message Header 29. Bits 15-0: RO PTP Message bytes 59:58 Bytes 59:58 of the PTP message header.
TS+ 0x004A… TS+ 0x007E	*Reserved*	*Reserved*
TS+ 0x0080	*TX_TS_1_NS_LO*	Reset: 0 x 00 00 Transmit TimeStamp 1 Nanoseconds low. Bits 15-0: RO Nanoseconds, bits 15:0 Nanoseconds of the timestamp of the frame (lowest bits).
TS+ 0x0082	*TX_TS_1_NS_HI*	Reset: 0 x 00 00 Transmit TimeStamp 1 Nanoseconds high. Bits 13-0: RO Nanoseconds, bits 29:16 Nanoseconds of the timestamp of the frame (highest bits). The nanoseconds value can at some cases be more than 999 999 999. The software using the timestamp can handle the situation by subtracting 1 000 000 000 from the nanoseconds value and adding 1 to seconds. Bits 15-14: RO *Reserved*

Address	Register	Description
TS+ 0x0084	*TX_TS_1_S_LO*	Reset: 0 x 00 00 Transmit TimeStamp 1 Seconds low. Bits 15-0: RO Seconds, bits 15:0 Seconds of the timestamp of the frame (lowest bits).
TS+ 0x0086	*TX_TS_1_S_HI*	Reset: 0 x 00 00 Transmit TimeStamp 1 Seconds high. Bits 15-0: RO Seconds, bits 31:16 Seconds of the timestamp of the frame (the next lowest bits).
TS+ 0x0088… TS+ 0x008C	*Reserved*	*Reserved*
TS+ 0x008E	*TX_TS_1_HDR 0*	Reset: 0 x 00 00 Transmit TimeStamp 1 Message Header 0. Bits 15-0: RO PTP Message bytes 1:0 First two bytes of the PTP message header.
TS+ 0x0090	*TX_TS_1_HDR 1*	Reset: 0 x 00 00 Transmit TimeStamp 1 Message Header 1. Bits 15-0: RO PTP Message bytes 3:2 Bytes 3:2 of the PTP message header.
TS+ 0x0092	*TX_TS_1_HDR 2*	Reset: 0 x 00 00 Transmit TimeStamp 1 Message Header 2. Bits 15-0: RO PTP Message bytes 5:4 Bytes 5:4 of the PTP message header.
TS+ 0x0094… TS+ 0x00C6	*TX_TS_1_HDR N*	Reset: 0 x 00 00 Transmit TimeStamp 1 Message Header N. Bits 15-0: RO PTP Message bytes ((N*2)+1):(N*2)
TS+ 0x00C8	*TX_TS_1_HDR 29*	Reset: 0 x 00 00 Transmit TimeStamp 0 Message Header 29. Bits 15-0: RO PTP Message bytes 59:58 Bytes 59:58 of the PTP message header.
…	…	…
TS+ 0x0180	*TX_TS_3_NS_LO*	Transmit TimeStamp 3 Nanoseconds low. See TX_TS_1_NS_LO
TS+ 0x0182	*TX_TS_3_NS_HI*	Transmit TimeStamp 3 Nanoseconds high. See TX_TS_1_NS_HI
TS+ 0x0184	*TX_TS_3_S_LO*	Transmit TimeStamp 3 Seconds low. See TX_TS_1_S_LO

Address	Register	Description
TS+ 0x0186	*TX _TS_3_S_HI*	Transmit TimeStamp 3 Seconds high. See TX_TS_1_S_HI
TS+ 0x0188… TS+ 0x018C	*Reserved*	*Reserved*
TS+ 0x018E	*TX _TS_3_HDR 0*	Transmit TimeStamp 3 Message Header 0. See TX_TS_1_HDR 0
TS+ 0x0190	*TX _TS_3_HDR 1*	Transmit TimeStamp 3 Message Header 1. See TX_TS_1_HDR 1
TS+ 0x0192	*TX _TS_3_HDR 2*	Transmit TimeStamp 3 Message Header 2. See TX_TS_1_HDR 2
TS+ 0x0194… TS+ 0x01C6	*TX _TS_3_HDR N*	Transmit TimeStamp 3 Message Header N. See TX_TS_1_HDR N
TS+ 0x01C8	*TX _TS_3_HDR 29*	Transmit TimeStamp 3 Message Header 29. See TX_TS_1_HDR 29
TS+ 0x01CA … TS+ 0x01FE	*Reserved*	*Reserved*
TS+ 0x0200	*RX _TS_0_NS_LO*	Receive TimeStamp 0 Nanoseconds low. See TX_TS_1_NS_LO
TS+ 0x0202	*RX _TS_0_NS_HI*	Receive TimeStamp 0 Nanoseconds high. See TX_TS_1_NS_HI
TS+ 0x0204	*RX _TS_0_S_LO*	Receive TimeStamp 0 Seconds low. See TX_TS_1_S_LO
TS+ 0x0206	*RX _TS_0_S_HI*	Receive TimeStamp 0 Seconds high. See TX_TS_1_S_HI
TS+ 0x0208… TS+ 0x020C	*Reserved*	*Reserved*
TS+ 0x020E	*RX _TS_0_HDR 0*	Receive TimeStamp 0 Message Header 0. See TX_TS_1_HDR 0
TS+ 0x0210	*RX _TS_0_HDR 1*	Receive TimeStamp 0 Message Header 1. See TX_TS_1_HDR 1
TS+ 0x0212	*RX _TS_0_HDR 2*	Receive TimeStamp 0 Message Header 2. See TX_TS_1_HDR 2

Address	Register	Description
TS+ 0x0214… TS+ 0x0246	*RX _TS_0_HDR N*	Receive TimeStamp 0 Message Header N. See TX_TS_1_HDR N
TS+ 0x0248	*RX _TS_0_HDR 29*	Receive TimeStamp 0 Message Header 29. See TX_TS_1_HDR 29
TS+ 0x024A … TS+ 0x027E	*Reserved*	*Reserved*
TS+ 0x0280	*RX _TS_1_NS_LO*	Receive TimeStamp 1 Nanoseconds low. See TX_TS_1_NS_LO
TS+ 0x0282	*RX _TS_1_NS_HI*	Receive TimeStamp 1 Nanoseconds high. See TX_TS_1_NS_HI
TS+ 0x0284	*RX _TS_1_S_LO*	Receive TimeStamp 1 Seconds low. See TX_TS_1_S_LO
TS+ 0x0286	*RX _TS_1_S_HI*	Receive TimeStamp 1 Seconds high. See TX_TS_1_S_HI
TS+ 0x0288… TS+ 0x028C	*Reserved*	*Reserved*
TS+ 0x028E	*RX _TS_1_HDR 0*	Receive TimeStamp 1 Message Header 0. See TX_TS_1_HDR 0
TS+ 0x0290	*RX _TS_1_HDR 1*	Receive TimeStamp 1 Message Header 1. See TX_TS_1_HDR 1
TS+ 0x0292	*RX _TS_1_HDR 2*	Receive TimeStamp 1 Message Header 2. See TX_TS_1_HDR 2
TS+ 0x0294… TS+ 0x02C6	*RX _TS_1_HDR N*	Receive TimeStamp 1 Message Header N. See TX_TS_1_HDR N
TS+ 0x02C8	*RX_TS_1_HDR 29*	Receive TimeStamp 1 Message Header 29. See TX_TS_1_HDR 29
…	…	…
TS+ 0x0380	*RX _TS_3_NS_LO*	Receive TimeStamp 3 Nanoseconds low. See TX_TS_1_NS_LO
TS+ 0x0382	*RX _TS_3_NS_HI*	Receive TimeStamp 3 Nanoseconds high. See TX_TS_1_NS_HI
TS+ 0x0384	*RX _TS_3_S_LO*	Receive TimeStamp 3 Seconds low. See TX_TS_1_S_LO

Address	Register	Description
TS+ 0x0386	*RX _TS_3_S_HI*	Receive TimeStamp 3 Seconds high. See TX_TS_1_S_HI
TS+ 0x0388… TS+ 0x038C	*Reserved*	*Reserved*
TS+ 0x038E	*RX _TS_3_HDR 0*	Receive TimeStamp 3 Message Header 0. See TX_TS_1_HDR 0
TS+ 0x0390	*RX _TS_3_HDR 1*	Receive TimeStamp 3 Message Header 1. See TX_TS_1_HDR 1
TS+ 0x0392	*RX _TS_3_HDR 2*	Receive TimeStamp 3 Message Header 2. See TX_TS_1_HDR 2
TS+ 0x0394… TS+ 0x03C6	*RX _TS_3_HDR N*	Receive TimeStamp 3 Message Header N. See TX_TS_1_HDR N
TS+ 0x03C8	*RX _TS_3_HDR 29*	Receive TimeStamp 3 Message Header 29. See TX_TS_1_HDR 29
TS+ 0x03CA … TS+ 0x03FE	*Reserved*	*Reserved*

6.9.3 *Virtual LAN Configuration Registers*

Virtual LAN configuration registers are presented in Table 29.

When a frame comes in from a port, its VLAN ID is checked against the VLAN configuration in the VLAN configuration registers. If the port is not a member of the VLAN the frame belongs to, the frame is dropped. Frames without a VLAN tag are mapped to the port's default VLAN (configured in PORT_VLAN_ID register). Untagged frames can be dropped by setting the default VLAN to a VLAN the port is not a member of.

A frame can be forwarded only to the ports that are members of the VLAN the frame belongs to. If the frame is a unicast frame and the destination port (according to the MAC address table) is not a member of the VLAN, the frame is forwarded to all the other ports that are members of the VLAN. Note that the reserved VLAN IDs 0x1, 0x2 and 0xFFF are handled the same way as the other VLAN IDs. Frames with the reserved VLAN ID 0x0 (priority tagged frames) can be mapped to any other VLAN.

Table 29. Virtual LAN Configuration Registers

Address	Register	Description
VLAN+ 0x0000	VLAN0	Reset: 2^(NUMBER_OF_PORTS)-1 VLAN ID0 Mask. The register is reserved in XRS3003. Bits 15-0: R/W VLAN mask for VLAN ID 0 Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. 0 = The port is not a member of this VLAN 1 = The port is a member of this VLAN
VLAN+ 0x0002	VLAN1	Reset: 2^(NUMBER_OF_PORTS)-1 VLAN ID1 Mask. The register is reserved in XRS3003. Bits 15-0: R/W VLAN mask for VLAN ID 1 Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. 0 = The port is not a member of this VLAN 1 = The port is a member of this VLAN
…	…	…
VLAN+ 0x1FFE	VLAN4095	Reset: 2^(NUMBER_OF_PORTS)-1 VLAN ID4095 Mask. The register is reserved in XRS3003. Bits 15-0: R/W VLAN mask for VLAN ID 4095 Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. 0 = The port is not a member of this VLAN 1 = The port is a member of this VLAN

6.10 Port Configuration Registers

Table 30 presents the port configuration register groups. The port configuration registers are used to configure port-specific features of RS.

Table 30. Port Configuration Register Groups

Address Offset	Acronym	Register group description	Section	Table
0x0000	GEN	General configuration and state registers	6.10.1	Table 31
0x2000	HSR	HSR configuration registers	6.10.2	Table 32
0x4000	PTP	PTP configuration registers	6.10.3	Table 34
0x6000	CNT	Counter Registers (not in XRS3003)	6.10.4	Table 35
0x8000	IPO	Inbound policy registers	6.10.5	Table 36

6.10.1 *General Configuration and State Registers*

General configuration and state registers are presented in Table 31.

Table 31. General Configuration and State Registers

Address	Register	Description			
GEN+ 0x0000	PORT_STATE	Reset: 0 x 02 02 Port State Register			
		Bits	1-0:	R/W	Port Forwarding state 00 = Forwarding. Port learns MAC addresses and forwards data. 01 = Learning. Port learns MAC addresses, but does not forward data. (for STP/RSTP) 10 = Disabled. Port neither learns MAC addresses nor forwards data. 11 = Reserved
		Bits	3-2:	R/W	Port management state 00 = Normal mode 01 = Management mode. Only frames with a management trailer can be sent to this port and it sends all frames with a management trailer. It is not allowed to configure a port to Management mode and PRP mode (HSR_CFG register) at the same time. In XRS3003 only port 0 can be configured in Management mode. 10 = Reserved 11 = Reserved
		Bits	9-4:	R/W	Speed select 000000 = automatic (RGMII ports only) 010010 = 1000 Mb/s (RGMII ports only) 100000 = 100 Mb/s 110000 = 10 Mb/s
		Bits	11-10:	RO	Current speed Updated only when the speed selection is automatic 00 = Reserved 01 = 1000 Mb/s 10 = 100 Mb/s 11 = 10 Mb/s
		Bits	15-12:	RO	*Reserved*

Address	Register	Description
GEN+ 0x2 … GEN+ 0xE	*Reserved*	*Reserved*
GEN+ 0x0010	PORT_VLAN	Reset: 0 x 8F FF Port VLAN Configuration Register. The register is reserved in XRS3003. Bits 11-0: R/W — Port default VLAN Incoming untagged frames are mapped to this VLAN. Outgoing frames with this VLAN are sent untagged. Value 0x0 = reserved. Bits 14-12: R/W — Port default PCP Priority Code Point (PCP) for the frames coming in from the port without a VLAN tag. This setting together with Port VLAN Priority Register defines also the priority for the frames inside RS. Bit 15: R/W — Tagged/untagged Defines whether frames are sent out with or without a VLAN tag. Does not affect handling of incoming frames. 0 = no VLAN tags are added to the frames at exit. Because one VLAN tag per frame is always removed in ingress, the number of VLAN tags in frames decreases by one when a frame goes through RS. 1 = a VLAN tag is added to every frame at exit, except for default VLAN. Because one VLAN tag per frame is always removed in ingress, the number of VLAN tags in the frame increases by one for frames that did not have a VLAN tag and stays the same for frames that had one or more VLAN tags (except for default VLAN for which the number of VLAN tags decreases by one).

Address	Register	Description
GEN+ 0x0012	VLAN0_MAPPING	Reset: 0 x 00 00 VLAN ID0 Mapping. The register is reserved in XRS3003. Bits 11-0: R/W Default VLAN for Priority Tagged Frames. Incoming frames with priority tag (VLAN ID 0) are mapped to this VLAN. This setting does not affect frames going out of this port. Bits 15-12: RO *Reserved*
GEN+ 0x0014	PORT_FWD_MASK	Reset: 0 x 00 00 Forward Portmask Configuration Register. Configures to which ports forwarding is allowed from this port. Bits 15-0: R/W Port forward mask Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. 0 = Enable forwarding to the port 1 = Disable forwarding to the port

Address	Register	Description
GEN+ 0x0016	PORT_VLAN_PRIO	Reset: 0 x FA50 Port VLAN Priority Register. Contains priorities for VLAN Priority Code Points (PCP). Lowest priority is 0, highest is 3. The priority the frame gets defines into which output priority queue the frame is put into. The register is reserved in XRS3003.

Bits	1-0:	R/W	Priority, PCP0 Frames with Priority Code Point 0 in their VLAN tag get this priority inside RS.
Bits	3-2:	R/W	Priority, PCP1 Frames with Priority Code Point 1 in their VLAN tag get this priority inside RS.
Bits	5-4:	R/W	Priority, PCP2 Frames with Priority Code Point 2 in their VLAN tag get this priority inside RS.
Bits	7-6:	R/W	Priority, PCP3 Frames with Priority Code Point 3 in their VLAN tag get this priority inside RS.
Bits	9-8:	R/W	Priority, PCP4 Frames with Priority Code Point 4 in their VLAN tag get this priority inside RS.
Bits	11-10:	R/W	Priority, PCP5 Frames with Priority Code Point 5 in their VLAN tag get this priority inside RS.
Bits	13-12:	R/W	Priority, PCP6 Frames with Priority Code Point 6 in their VLAN tag get this priority inside RS.
Bits	15-14:	R/W	Priority, PCP7 Frames with Priority Code Point 7 in their VLAN tag get this priority inside RS.

6.10.2 *HSR/PRP Registers*

HSR/PRP registers are presented in Table 32.

Table 32. HSR Registers

Address	Register	Description
HSR+ 0x0000	HSR_CFG	Reset: 0 x 00 00 HSR/PRP Configuration Register This register can be updated only when the port is disabled via PORT_STATE (Table 31) register. Configuration change procedure: - Disable both redundant ports (PORT_STATE register) - Change mode (HSR_CFG) for both redundant ports - Enable both configured ports (PORT_STATE) Bit 0: R/W — Port Mode 0 = HSR/PRP disabled for the port 1 = HSR/PRP enabled for the port Bits 7-1: RO — *Reserved* Bit 8: R/W — HSR/PRP mode select 0 = Port is in HSR mode 1 = Port is in PRP mode The selection is valid only when HSR/PRP is enabled (bit0=1). Bit 9: R/W — Redundant/Interlink mode select 0 = Port is HSR/PRP redundant port 1 = Port is HSR/PRP interlink port The selection is valid only when HSR/PRP is enabled (bit0=1). Note that in a valid configuration there is either two or zero redundant ports. Bit 10: R/W — LanId for the port If this port is in PRP mode and a frame is output from this port, this bit determines the LanId of the frame 0 = 0xA 1 = 0xB Note that this bit has also another meaning (PathId) described below. For HSR/PRP redundant ports this bit must be 0 for the other redundant port and 1 for the other redundant port.

Address	Register	Description			
		Bits	13-11:	R/W	NetId for the port If this port is in PRP mode and a frame from a HSR port is forwarded into this port and the NetID in the frame matches this NetID, the frame is dropped. Note that these bits have also another meaning (PathId) described below.
		Bits	13-10:	R/W	PathId for the port If this port is in non-HSR mode and a frame from this port goes out from an HSR port, the PathID of the HSR tag is given this value (see Table 23 and Table 24). Note that these bits have also another meaning (LandId/NetId) described above.
		Bits	15-14:	RO	*Reserved*

6.10.3 *PTP Registers*

PTP registers are presented in Table 34. Interface type specific PTP Delay corrections are presented in Table 33. Adding the values presented in Table 33 to the values written to registers PTP_RX_SYNC_DELAY_NS_LOW, PTP_RX_EVENT_DELAY_NS and PTP_TX_EVENT_DELAY_NS make PTP corrections more accurate. For example in case of peer-to-peer transparent clock, if the link delay is 50 ns, RX latency of the PHY is 220 ns, the interface type is RGMII and the port is in gigabit mode, the value written to register PTP_RX_SYNC_DELAY_NS_LOW is 50 + 220 + 40 = 310.

Table 33. Delay Correction Values for PTP Registers

Interface type:	**RMII**		**RGMII**	
Direction:	**RX**	**TX**	**RX**	**TX**
1000Mbit/s	-	-	40	16
100Mbit/s	100	120	200	80
10Mbit/s	260	440	2000	800

Table 34. PTP Registers

Address	Register	Description
PTP+ 0x0000	*Reserved*	*Reserved*
PTP+ 0x0002	PTP_RX_SYNC_ DELAY_NS_LOW	Reset: 0 x 00 00 Contains the lowest 16 bits of the Receive delay compensation for PTP Sync messages. This value is added to the correction field of every received Sync message in addition to residence time and TX delay. This compensation consist of delay from device input to XRS input (includes for example PHY delay), link delay for peer-to-peer transparent clock and a constant correction value that depends on interface type and speed (see Table 33). For end-to-end transparent clock this register must be set to same value as PTP_RX_EVENT_DELAY_NS. Write access to this registers takes into use the value in register PTP_RX_SYNC _DELAY_NS_HIGH. Bits | 15-0: | R/W | Sync delay, nanosecond bits 15:0
PTP+ 0x0004	PTP_RX_SYNC_ DELAY_NS_HIGH	Reset: 0 x 00 00 Contains bits 16…23 of the Receive delay compensation for PTP Sync messages. This value is added to the correction field of every received Sync message in addition to residence time and TX delay. This compensation consist of delay from device input to XRS input (includes for example PHY delay), link delay for peer-to-peer transparent clock and a constant correction value that depends on interface type and speed (see Table 33). For end-to-end transparent clock this register must be set to value 0x0. XRS starts using the value in this register after writing to register PTP_DELAY_NS_LOW. Bits | 7-0: | R/W | Sync delay, nanosecond bits 23:16 Bits | 15-8: | RO | *Reserved*
PTP+ 0x0006… PTP+ 0x0008	*Reserved*	*Reserved*

<table>
<tr><th>Address</th><th>Register</th><th>Description</th></tr>
<tr><td>PTP+
0x000A</td><td>PTP_RX_EVENT
_DELAY_NS</td><td>Reset: 0 x 00 00
Contains the Receive delay compensation for all the PTP Event messages except Sync messages (who have an own compensation register).
This value is added to the correction field of every received Event message (except for Sync messages) in addition to residence time and TX delay.
This compensation value consist of delay from device input to XRS input (includes for example PHY delay) and a constant correction value that depends on interface type and speed (see Table 33).
<table><tr><td>Bits</td><td>14-0:</td><td>R/W</td><td>Event delay, nanoseconds</td></tr><tr><td>Bits</td><td>15:</td><td>RO</td><td>Reserved</td></tr></table></td></tr>
<tr><td>PTP+
0x000C…
PTP+
0x0010</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>PTP+
0x0012</td><td>PTP_TX_EVENT
_DELAY_NS</td><td>Reset: 0 x 00 00
Contains the Transmit delay compensation for all the PTP Event messages.
This value is added to the correction field of every transmitted Event message in addition to residence time and RX delay.
This compensation value consist of delay from XRS output to device output (includes for example PHY delay) and a constant correction value that depends on interface type and speed (see Table 33).
<table><tr><td>Bits</td><td>14-0:</td><td>R/W</td><td>Event delay, nanoseconds</td></tr><tr><td>Bits</td><td>15:</td><td>RO</td><td>Reserved</td></tr></table></td></tr>
</table>

6.10.4 *Counter Registers*

Counter registers are presented in Table 35. The counters are designed to support the Ethernet Statistics Group of Remote Network MONitoring (RMON) SNMP MIB (RFC 2819). The register area is reserved in XRS3003.

Table 35. Counter Registers (not in XRS3003)

Address	Register	Description
CNT+ 0x0000	CNT_CTRL	Reset: 0 x 00 00 Counter control. This register controls the functionality on the other counter registers. Bit 0: R/SC Capture Writing 1 to this bit captures all the counters after which the counter values are written to the corresponding registers. After capture the counters are reset. So the registers contain the number of events that happened between two successive captures. Bits 15-1: RO *Reserved*
CNT+ 0x0002… CNT+ 0x01FE	*Reserved*	*Reserved*
CNT + 0x0200	RX_GOOD_ OCTETS_L	Reset: 0 x 00 00 RX Good Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Preamble and SFD are not counted to frame length. FCS (CRC) is counted. Frames are considered as good frames if they have valid CRC, there was no RX error during the receiving and their length is 64 to 1536 octets (70 to 1536 with HSR/PRP). If a frame is not considered as a good frame, it's considered as a bad frame. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of octets in good frames received

Address	Register	Description
CNT + 0x0202	RX_GOOD_ OCTETS_H	Reset: 0 x 00 00 RX Good Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Preamble and SFD are not counted to frame length. FCS (CRC) is counted. Frames are considered as good frames if they have valid CRC, there was no RX error during the receiving and their length is 64 to 1536 octets (70 to 1536 with HSR/PRP). If a frame is not considered as a good frame, it's considered as a bad frame. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of octets in good frames received
CNT + 0x0204	RX_BAD_ OCTETS_L	Reset: 0 x 00 00 RX Bad Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Preamble and SFD are not counted to frame length. FCS (CRC) is counted. Frames are considered as bad frames if they do not have a valid CRC, an RX error happened during reception or their length is under 64 octets (under 70 with HSR/PRP) or over 1536 octets. If a frame is not considered as a good frame, it's considered as a bad frame. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received octets in bad Ethernet frames
CNT + 0x0206	RX_BAD_ OCTETS_H	Reset: 0 x 00 00 RX Bad Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Preamble and SFD are not counted to frame length. FCS (CRC) is counted. Frames are considered as bad frames if they do not have a valid CRC, an RX error happened during reception or their length is under 64 octets (under 70 with HSR/PRP) or over 1536 octets. If a frame is not considered as a good frame, it's considered as a bad frame. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received octets in bad Ethernet frames

Address	Register	Description
CNT + 0x0208	RX_UNICAST_L	Reset: 0 x 00 00 RX Unicast Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received good unicast frames
CNT + 0x020A	RX_UNICAST_H	Reset: 0 x 00 00 RX Unicast High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received good unicast frames
CNT + 0x020C	RX_BROADCAST_L	Reset: 0 x 00 00 RX Broadcast Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received good broadcast frames
CNT + 0x020E	RX_BROADCAST_H	Reset: 0 x 00 00 RX Broadcast High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received good broadcast frames
CNT + 0x0210	RX_MULTICAST_L	Reset: 0 x 00 00 RX Multicast Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received good multicast frames. Does not include broadcast frames counted in RX_BROADCAST_L/H.

Address	Register	Description
CNT + 0x0212	RX_MULTICAST_H	Reset: 0 x 00 00 RX Multicast High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received good multicast frames. Does not include broadcast frames counted in RX_BROADCAST_L/H.
CNT + 0x0214	RX_UNDERSIZE_L	Reset: 0 x 00 00 RX Undersize Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Undersized frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with valid CRC and under 64 octets in length (under 70 octets with HSR/PRP).
CNT + 0x0216	RX_UNDERSIZE_H	Reset: 0 x 00 00 RX Undersize High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Undersized frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with valid CRC and under 64 octets in length (under 70 octets with HSR/PRP).
CNT + 0x0218	RX_FRAGMENTS _L	Reset: 0 x 00 00 RX Fragments Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Fragments are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with invalid CRC and length under 64 octets.

Address	Register	Description
CNT + 0x021A	RX_FRAGMENTS_H	Reset: 0 x 00 00 RX Fragments High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Fragments are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with invalid CRC and length under 64 octets.
CNT + 0x021C	RX_OVERSIZE_L	Reset: 0 x 00 00 RX Oversize Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Oversized frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with valid CRC and length over 1536 octets.
CNT + 0x021E	RX_OVERSIZE_H	Reset: 0 x 00 00 RX Oversize High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Oversized frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with valid CRC and length over 1536 octets.
CNT + 0x0220	RX_JABBER_L	Reset: 0 x 00 00 RX Jabber Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Jabber frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with invalid CRC and length over 1536 octets.
CNT + 0x0222	RX_JABBER_H	Reset: 0 x 00 00 RX Jabber High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. Jabber frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with invalid CRC and length over 1536 octets.

Address	Register	Description
CNT + 0x0224	RX_ERR_L	Reset: 0 x 00 00 RX Error Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. RX Erroneous frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with an error, with size from 64 to 1536 octets, with or without a valid CRC. Includes frames with non-integral number of bytes and those received with RX Error signal from the PHY.
CNT + 0x0226	RX_ERR_H	Reset: 0 x 00 00 RX Error High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. RX Erroneous frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with an error, with size from 64 to 1536 octets, with or without a valid CRC. Includes frames with non-integral number of bytes and those received with RX Error signal from the PHY.
CNT + 0x0228	RX_CRC_L	Reset: 0 x 00 00 RX CRC Error Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. CRC Erroneous frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size from 64 to 1536 octets and without a valid CRC, but not counted in RX_ERR_L/H.

Address	Register	Description
CNT + 0x022A	RX_CRC_H	Reset: 0 x 00 00 RX CRC Error High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. CRC Erroneous frames are considered as bad frames, and they are not forwarded. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size from 64 to 1536 octets and without a valid CRC, but not counted in RX_ERR_L/H.
CNT + 0x022C	RX_64_L	Reset: 0 x 00 00 RX 64 Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size of exactly 64 octets, including frames with errors.
CNT + 0x022E	RX_64_H	Reset: 0 x 00 00 RX 64 Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size of exactly 64 octets, including frames with errors.
CNT + 0x0230	RX_65_127_L	Reset: 0 x 00 00 RX 65 to 127 Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size of 65 to 127 octets, including frames with errors.

Address	Register	Description
CNT + 0x0232	RX_65_127_H	Reset: 0 x 00 00 RX 65 to 127 Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size of 65 to 127 octets, including frames with errors.
CNT + 0x0234	RX_128_255_L	Reset: 0 x 00 00 RX 128 to 255 Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size of 128 to 255 octets, including frames with errors.
CNT + 0x0236	RX_128_255_H	Reset: 0 x 00 00 RX 128 to 255 Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size of 128 to 255 octets, including frames with errors.
CNT + 0x0238	RX_256_511_L	Reset: 0 x 00 00 RX 256 to 511 Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size of 256 to 511 octets, including frames with errors.
CNT + 0x023A	RX_256_511_H	Reset: 0 x 00 00 RX 256 to 511 Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size of 256 to 511 octets, including frames with errors.

Address	Register	Description
CNT + 0x023C	RX_512_1023_L	Reset: 0 x 00 00 RX 512 to 1023 Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size of 512 to 1023 octets, including frames with errors.
CNT + 0x023E	RX_512_1023_H	Reset: 0 x 00 00 RX 512 to 1023 Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size of 512 to 1023 octets, including frames with errors.
CNT + 0x0240	RX_1024_ 1536_L	Reset: 0 x 00 00 RX 1024 to 1536 Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received frames with size of 1024 to 1536 octets, including frames with errors.
CNT + 0x0242	RX_1024_ 1536_H	Reset: 0 x 00 00 RX 1024 to 1536 Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received frames with size of 1024 to 1536 octets, including frames with errors.

Address	Register	Description
CNT + 0x0244	RX_HSRPRP_L	Reset: 0 x 00 00 RX HSR/PRP Frames Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of good HSR frames received while in HSR mode and number of good PRP frames received while in PRP mode. The counter does not count HSR frames in non-HSR mode or PRP frames in non-PRP mode. The counter is not reset when changing modes.
CNT + 0x0246	RX_HSRPRP_H	Reset: 0 x 00 00 RX HSR/PRP Frames High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of good HSR frames received while in HSR mode and number of good PRP frames received while in PRP mode. The counter does not count HSR frames in non-HSR mode or PRP frames in non-PRP mode. The counter is not reset when changing modes.
CNT+ 0x0248	RX_WRONGLAN_L	Reset: 0 x 00 00 RX Wrong LAN Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of received good frames with wrong LAN identifier. The counter is used only when the port is in PRP mode.
CNT+ 0x024A	RX_WRONGLAN _H	Reset: 0 x 00 00 RX Wrong LAN High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of received good frames with wrong LAN identifier. The counter is used only when the port is in PRP mode.

Address	Register	Description
CNT+ 0x024C	RX_DUPLICATE_L	Reset: 0 x 00 00 RX Duplicate Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of frames received that are detected as copies of frames received earlier from this or another HSR/PRP port. The counter is used only when the port is in HSR or PRP mode.
CNT+ 0x024E	RX_DUPLICATE_H	Reset: 0 x 00 00 RX Duplicate High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of frames received that are detected as copies of frames received earlier from this or another HSR/PRP port. The counter is used only when the port is in HSR or PRP mode.
CNT+ 0x250… CNT+ 0x27E	*Reserved*	*Reserved*
CNT + 0x0280	TX_OCTETS_L	Reset: 0 x 00 00 TX Octets Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Total number of octets in frames transmitted.
CNT + 0x0282	TX_OCTETS_H	Reset: 0 x 00 00 TX Octets High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Total number of octets in frames transmitted.

Address	Register	Description
CNT + 0x0284	TX_UNICAST_L	Reset: 0 x 00 00 TX Unicast Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of transmitted unicast frames.
CNT + 0x0286	TX _UNICAST_H	Reset: 0 x 00 00 TX Unicast High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of transmitted unicast frames.
CNT + 0x0288	TX _BROADCAST_L	Reset: 0 x 00 00 TX Broadcast Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of transmitted broadcast frames.
CNT + 0x028A	TX _BROADCAST_H	Reset: 0 x 00 00 TX Broadcast High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of transmitted broadcast frames.
CNT + 0x028C	TX _MULTICAST_L	Reset: 0 x 00 00 TX Multicast Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of transmitted multicast frames. Does not include broadcast frames counted in TX_BROADCAST_L/H.
CNT + 0x028E	TX _MULTICAST_H	Reset: 0 x 00 00 TX Multicast High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of transmitted multicast frames. Does not include broadcast frames counted in TX_BROADCAST_L/H.

Address	Register	Description
CNT + 0x0290	TX_HSRPRP_L	Reset: 0 x 00 00 TX HSR/PRP Frames Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of HSR frames transmitted while in HSR mode and number of PRP frames transmitted while in PRP mode. The counter does not count HSR frames in non-HSR mode or PRP frames in non-PRP mode. The counter is not reset when changing modes.
CNT + 0x0292	TX_HSRPRP_H	Reset: 0 x 00 00 TX HSR/PRP Frames High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of HSR frames transmitted while in HSR mode and number of PRP frames transmitted while in PRP mode. The counter does not count HSR frames in non-HSR mode or PRP frames in non-PRP mode. The counter is not reset when changing modes.
CNT+ 0x0294… CNT+ 0x02BE	*Reserved*	*Reserved*
CNT + 0x02C0	PRIQ_DROP_L	Reset: 0 x 00 00 Priority Queue Drop Counter Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of frames dropped by this output port because TX priority queue was full. Note that a frame can be forwarded to several ports, so one single frame can increment the counter for more than one port.

Address	Register	Description
CNT + 0x02C2	PRIQ_DROP_H	Reset: 0 x 00 00 Priority Queue Drop Counter High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of frames dropped by this output port because TX priority queue was full. Note that a frame can be forwarded to several ports, so one single frame can increment the counter for more than one port.
CNT+ 0x02C4	EARLY_DROP_L	Reset: 0 x 00 00 Early Frame Drop Counter Low The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 15:0 Number of frames dropped by this output port from TX priority queues because internal memory was becoming full. Note that a frame can be forwarded to several ports, so one single frame can increment the counter for more than one port.
CNT+ 0x02C6	EARLY_DROP_H	Reset: 0 x 00 00 Early Frame Drop Counter High The value in this register is updated after writing 1 to Capture bit in CNT_CTRL register. The counter saturates to 0x FFFF FFFF. Bits 15-0: RO Counter value, bits 31:16 Number of frames dropped by this output port because internal memory was becoming full. Note that a frame can be forwarded to several ports, so one single frame can increment the counter for more than one port.

Note that RS does not forward frames whose length is more than 1536 octets or under 64 octets (under 70 octets with HSR/PRP). Frames who came in with a VLAN tag and whose length was under 68 octets (under 74 octets with HSR/PRP) are padded to 64 octets (70 octets with HSR/PRP) when sent out without a VLAN tag.

6.10.5 *Inbound Policy Registers*

Inbound policy registers are presented in Table 36.

Table 36. Inbound Policy Registers

Address	Register	Description			
IPO + 0x0000	ETH_ADDR0_CFG	Reset: 0 x 00 00 Configuration for Ethernet address 0 (ETH_ADDR0) of inbound policy filter.			
		Bit	0:	R/W	Enable Enable this entry. The other bits in this register are functional only when this bit is set to 1.
		Bit	1:	R/W	Source/Destination Match Defines whether to compare the address in registers ETH_ADDR0_0…ETH_ADDR0_2 to the source or to the destination MAC address of the incoming frame. 0 = Match to destination address 1 = Match to source address
		Bits	7-2:	R/W	Compared Length Defines how many bits from the start of the MAC addresses of the incoming frames are compared to the value in registers ETH_ADDR0_0...2. If the value is 48 the whole MAC address is compared. With value 1 only the unicast/multicast bit is compared (note that the bit order here is transmission order). With value 0 every frame matches. Values over 48 are reserved.
		Bits	9-8:	RO	*Reserved*
		Bit	10:	R/W	No HSR-tag Do not add a HSR-tag to the frame even when output from HSR enabled port. In case of two IPO matches (source address match and destination address match) the information is taken from the second match.
		Bit	11:	R/W	No PRP-trailer Do not add a PRP-trailer to the frame even when output from PRP enabled port. In case of two IPO matches (source address match and destination address match) the information is taken from the second match.

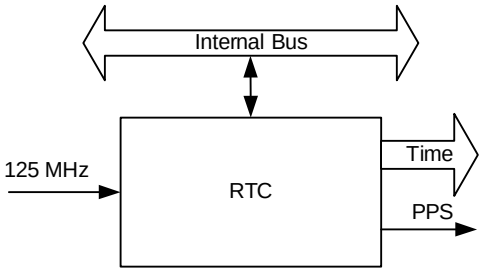
Address	Register	Description			
		Bits	13-12:	R/W	New Priority If Preserve Priority bit = 0 the priority of the frame is set to this value. Overrides priority defined by VLAN PCP (in XRS3003 the default priority is 0). The lowest priority is 0, the highest is 3. In case of two IPO matches (source address match and destination address match) the information is taken from the second match.
		Bit	14:	R/W	Preserve Priority 0 = Set priority of the frame to the value in "New Priority" bits. 1 = Do not alter priority of the frame In case of two IPO matches (source address match and destination address match) the information is taken from the second match.
		Bit	15:	RO	*Reserved*
IPO + 0x0002	ETH_ADDR0_FWD_ALLOW	Reset: 0 x 00 00 Ethernet address 0 allow forwarding. Defines to which ports the matching frame is allowed to be forwarded.			
		Bits	15-0:	R/W	Forward allow mask Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. 0 = Forwarding is not allowed to this port 1 = Forwarding is allowed to this port.
IPO + 0x0004	ETH_ADDR0_FWD_MIRROR	Reset: 0 x 00 00 Ethernet address 0 mirror port. Defines to which ports the matching frames are mirrored to.			
		Bits	15-0:	R/W	Forward mirror mask Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. 0 = Frame is not mirrored to this port 1 = Frame is mirrored to this port. If the corresponding allow bit is 1, also duplicate frames are mirrored. If the corresponding allow bit is 0 HSR/PRP duplicate discard is applied to mirrored frames and duplicates are not mirrored.
IPO+ 0x0006	*Reserved*	*Reserved*			

Address	Register	Description
IPO + 0x0008	ETH_ADDR0_0	Reset: 0 x 00 00 Ethernet address 0 part 0. First part of the Ethernet address ETH_ADDR0. Bits 7-0: R/W 1st octet (XX:XX:XX:XX:XX:XX) Bits 15-8: R/W 2nd octet (XX:XX:XX:XX:XX:XX)
IPO + 0x000A	ETH_ADDR0_1	Reset: 0 x 00 00 Ethernet address 0 part 1. Second part of the Ethernet address ETH_ADDR0. Bits 7-0: R/W 3rd octet (XX:XX:XX:XX:XX:XX) Bits 15-8: R/W 4th octet (XX:XX:XX:XX:XX:XX)
IPO + 0x000C	ETH_ADDR0_2	Reset: 0 x 00 00 Ethernet address 0 part 2. Third part of the Ethernet address ETH_ADDR0. Bits 7-0: R/W 5th octet (XX:XX:XX:XX:XX:XX) Bits 15-8: R/W 6th octet (XX:XX:XX:XX:XX:XX)
IPO+ 0x000E… IPO+ 0x001E	*Reserved*	*Reserved*
IPO + 0x0020	ETH_ADDR1_CFG	Ethernet address 1 filter configuration. See ETH_ADDR0_CFG.
IPO + 0x0022	ETH_ADDR1_FWD_ALLOW	Ethernet address 1 allow forwarding. See ETH_ADDR0_FWD_ALLOW.
IPO + 0x0024	ETH_ADDR1_FWD_MIRROR	Ethernet address 1 mirror port. See ETH_ADDR0_FWD_MIRROR.
IPO+ 0x0026	*Reserved*	*Reserved*
IPO + 0x0028	ETH_ADDR1_0	Ethernet address 1 part 0. See ETH_ADDR0_0.
IPO + 0x002A	ETH_ADDR1_1	Ethernet address 1 part 1. See ETH_ADDR0_1.
IPO + 0x002C	ETH_ADDR1_2	Ethernet address 1 part 2. See ETH_ADDR0_2.
IPO+ 0x002E… IPO+ 0x003E	*Reserved*	*Reserved*
IPO + 0x0040	ETH_ADDR2_CFG	Ethernet address 2 filter configuration. See ETH_ADDR0_CFG.
…	…	…
IPO + 0x01EC	ETH_ADDR15_2	Ethernet address 15 part 2. See ETH_ADDR0_2.

7. REAL-TIME CLOCK (RTC)

RTC (Real-Time Clock) keeps track of the internal time of XRS7004/XRS7003/XRS3003. RTC provides clock time for the other blocks including RS and TS. RTC provides also a Pulse per Second (PPS) output that can be used for synchronizing other devices. The time format of RTC is compatible with the time format defined in IEEE1588 standard.

Figure 33. RTC Interfaces



The time is presented in seconds and nanoseconds and there are 48 bits for seconds and 30 bits for nanoseconds.

The internal implementation is an accumulator that counts nanoseconds. Seconds value is updated and nanoseconds value wraps around always when nanoseconds value exceeds 1 000 000 000 (corresponds to one second). Because of this, the nanoseconds value can never be more than 30 bits in length.

7.1 Registers

Registers of RTC are presented in Table 37.

Table 37. RTC Registers

Address	Register	Description
0x0000 … 0x1002	*Reserved*	*Reserved*
0x1004	CUR_NSEC0	Reset: 0 x 00 00 Current time nanoseconds. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Nanoseconds Nanoseconds part of the current time, bits 15:0.

Address	Register	Description
0x1006	CUR_NSEC1	Reset: 0 x 00 00 Current time nanoseconds. Updated with Read Time command (see TIME_CMD register). Bits 13-0: RO Nanoseconds Nanoseconds part of the current time, bits 29:16. Bits 15-14: RO *Reserved*
0x1008	CUR_SEC0	Reset: 0 x 00 00 Current time seconds. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Seconds Seconds part of the current time, bits 15:0.
0x100A	CUR_SEC1	Reset: 0 x 00 00 Current time seconds. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Seconds Seconds part of the current time, bits 31:16.
0x100C	CUR_SEC2	Reset: 0 x 00 00 Current time seconds. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Seconds Seconds part of the current time, bits 47:32.
0x100E	*Reserved*	*Reserved*
0x1010	TIME_CC0	Reset: 0 x 00 00 Clock cycle counter. Free-running counter running at 125 MHz. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Free-running clock cycle counter Clock cycles since the reset, bits 15:0.. The value wraps around automatically. Time adjustments have no effect on this value.
0x1012	TIME_CC1	Reset: 0 x 00 00 Clock cycle counter. Free-running counter running at 125 MHz. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Free-running clock cycle counter Clock cycles since the reset, bits 31:16.. The value wraps around automatically. Time adjustments have no effect on this value.

Address	Register	Description
0x1014	TIME_CC2	Reset: 0 x 00 00 Clock cycle counter. Free-running counter running at 125 MHz. Updated with Read Time command (see TIME_CMD register). Bits 15-0: RO Free-running clock cycle counter Clock cycles since the reset, bits 47:32.. The value wraps around automatically. Time adjustments have no effect on this value.
0x1016 … 0x101E	*Reserved*	*Reserved*
0x1020	STEP_SIZE0	Reset: 0 x 00 00 Step size. The value added to the internal accumulator at the frequency of 125 MHz. Defines the speed of the clock time. Bits 15-0: R/W Subnanoseconds Subnanoseconds step size, bits 15:0. The new value is taken into use with Adjust Step command (see TIME_CMD register).
0x1022	STEP_SIZE1	Reset: 0 x 00 00 Step size. The value added to the internal accumulator at the frequency of 125 MHz. Defines the speed of the clock time. Bits 15-0: R/W Subnanoseconds Subnanoseconds step size, bits 31:16. The new value is taken into use with Adjust Step command (see TIME_CMD register).
0x1024	STEP_SIZE2	Reset: 0 x 00 08 Step size. The value added to the internal accumulator at the frequency of 125 MHz. Defines the speed of the clock time. Bits 3-0: R/W Nanoseconds Nanoseconds step size. The new value is taken into use with Adjust Step command (see TIME_CMD register). Bits 15-4: RO *Reserved*
0x1026 … 0x1032	*Reserved*	*Reserved*

Address	Register	Description
0x1034	ADJUST_NSEC0	Reset: 0 x 00 00 Adjust time (nanoseconds part). This value is added to nanoseconds part with Adjust Time command (see TIME_CMD register). The nanosecond adjustment value is unsigned and therefore always positive. To go backwards use negative seconds adjustment (ADJUST_SEC registers). For example, to go 500 ns backwards, adjust 999 999 500 ns forward and one second backward. The nanosecond adjustment is not allowed to be more than 999 999 999. Bits | 15-0: | R/W | Nanoseconds, bits 15:0.
0x1036	ADJUST_NSEC1	Reset: 0 x 00 00 Adjust time (nanoseconds part). This value is added to nanoseconds part with Adjust Time command (see TIME_CMD register). The nanosecond adjustment value is unsigned and therefore always positive. To go backwards use negative seconds adjustment (ADJUST_SEC registers). For example, to go 500 ns backwards, adjust 999 999 500 ns forward and one second backward. The nanosecond adjustment is not allowed to be more than 999 999 999. Bits | 13-0: | R/W | Nanoseconds, bits 29:16. Bits | 15-14: | RO | *Reserved*
0x1038	ADJUST_SEC0	Reset: 0 x 00 00 Adjust time (seconds part). This value is added to seconds part with Adjust Time command (see TIME_CMD register). The seconds adjustment value is signed and can therefore be also negative. To go backwards, use two's complement arithmetics. Bits | 15-0: | R/W | Seconds, bits 15:0.
0x103A	ADJUST_SEC1	Reset: 0 x 00 00 Adjust time (seconds part). This value is added to seconds part with Adjust Time command (see TIME_CMD register). The seconds adjustment value is signed and can therefore be also negative. To go backwards, use two's complement arithmetics. Bits | 15-0: | R/W | Seconds, bits 31:16.
0x103C	ADJUST_SEC2	Reset: 0 x 00 00 Adjust time (seconds part). This value is added to seconds part with Adjust Time command (see TIME_CMD register). The seconds adjustment value is signed and can therefore be also negative. To go backwards, use two's complement arithmetics. Bits | 15-0: | R/W | Seconds, bits 47:32. The highest bit (bit 47) is the sign.
0x103E	*Reserved*	*Reserved*

Address	Register	Description			
0x1040	TIME_CMD	Reset: 0 x 00 00 Time command register.			
		Bit	0:	R/SC	Adjust Time Causes (for one time only) the values in ADJUST_NSEC and ADJUST_SEC registers to be added to the internal time (accumulator) instead of the values in STEP_SIZE registers. Causes the clock time to jump forward or backward.
		Bit	1:	R/SC	Adjust Step Take the new values in STEP_SIZE registers into use. Adjusting the step makes the clock run faster or slower.
		Bit	2:	R/SC	Read Time Updates the CUR_NSEC, CUR_SEC and TIME_CC registers with the values in the accumulator (current time).
		Bits	15-3:	RO	*Reserved*

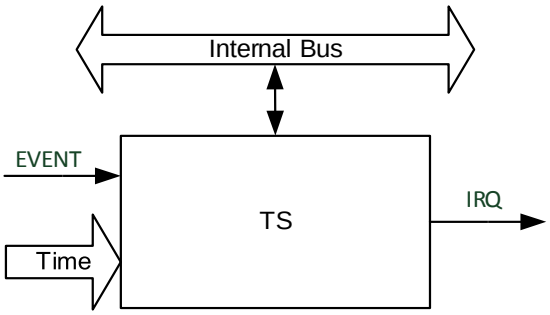
8. TIME STAMPER (TS)

Time Stamper (TS) time stamps external input events. This functionality can be used for transferring the internal clock time of XRS7004/7003 to other devices and/or systems. It is also possible to use TS for synchronizing XRS7004/7003 to an external time source. XRS3003 does not have TS functionality.

TS timestamps rising edges of its EVENT input and stores the timestamps into its internal registers where the user can read them. Input signal frequencies up to 25 MHz are supported. One typical use of TS is synchronizing to a 1PPS signal.

Figure 34 presents the interfaces of TS.

Figure 34. TS Interfaces



The time interface (see Figure 34) provides the current time information for TS as TS does not have internal timekeeping functionality. The Time Interface is connected to the RTC (Chapter 7). The IRQ line can be connected to an IRQ input of an attached CPU so that the CPU does not necessarily need to poll the registers of TS.

8.1 Registers

Registers of TS are presented in Table 38. XRS3003 does not have TS functionality so the register area is reserved in XRS3003.

Table 38. TS Registers (not in XRS3003)

Address	Register	Description
0x0000… 0x0FFE	*Reserved*	*Reserved*

<table>
<tr><th>Address</th><th>Register</th><th>Description</th></tr>
<tr><td>0x1000</td><td>TS_CTRL</td><td>Reset: 0 x 00 00<table>
<tr><td>Bit</td><td>0:</td><td>R/SC</td><td>Get Timestamp
Writing 1 to this bit enables recording timestamp of the next event (rising edge) to registers TS_SEC, TS_NSEC and TS_SNSEC. Also the amount of events (rising edges) arrived so far is written to register PCNT. This bit is cleared when the timestamp and the counter value are available.</td></tr>
<tr><td>Bits</td><td>15-1:</td><td>RO</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>0x1002…
0x1006</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>0x1008</td><td>INT_MASK</td><td>Interrupt Mask
Reset 0 x 00 00
An external interrupt is activated when an Interrupt Mask bit is set and the corresponding Interrupt Status bit is 1.<table>
<tr><td>Bit</td><td>0:</td><td>R/W</td><td>Time Stamp
Indicates that an event (rising edge) has arrived and a time stamp and pulse counter value are recorded to registers TS_SEC, TS_NSEC, TS_SNSEC and PCNT</td></tr>
<tr><td>Bits</td><td>15-1:</td><td>RO</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>0x100A…
0x100E</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>0x1010</td><td>INT_STATUS</td><td>Interrupt Status
Reset 0 x 00 00
An external interrupt is activated when an Interrupt Status bit is set and the corresponding Interrupt Mask bit is 1. The interrupt status bit in this register is updated independent from the corresponding Interrupt Mask bit.<table>
<tr><td>Bit</td><td>0:</td><td>R/C</td><td>Time Stamp
Indicates that an event (rising edge) has arrived and a time stamp and pulse counter value are recorded to registers TS_SEC, TS_NSEC, TS_SNSEC and PCNT</td></tr>
<tr><td>Bits</td><td>15-1:</td><td>RO</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>0x1012…
0x1102</td><td>Reserved</td><td>Reserved</td></tr>
</table>

Address	Register	Description
0x1104	TS_NSEC0	Time Stamp Nanoseconds Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 15-0: RO Nanoseconds Nanoseconds part of the time of the previous event, bits 15:0.
0x1106	TS_NSEC1	Time Stamp Nanoseconds Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 13-0: RO Nanoseconds Nanoseconds part of the time of the previous event, bits 29:16. Bits 15-14: RO *Reserved*
0x1108	TS_SEC0	Time Stamp Seconds Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 15-0: RO Seconds Seconds part of the time of the previous event, bits 15:0.
0x110A	TS_SEC1	Time Stamp Seconds Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 15-0: RO Seconds Seconds part of the time of the previous event, bits 31:16.
0x110C	TS_SEC2	Time Stamp Seconds Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 15-0: RO Seconds Seconds part of the time of the previous event, bits 47:32.

Address	Register	Description
0x110E	*Reserved*	*Reserved*
0x1110	PCNT0	Pulse Counter Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 15-0: RO Pulse Counter Value in the free running 32 bit counter that counts incoming events (rising edges). This counter can be used for finding out how many events there has been between certain Time Stamp captures. Useful with fast input signals. Bits 15:0.
0x1112	PCNT1	Pulse Counter Reset: 0 x 00 00 Updated only with “Get Timestamp” command (see TS_CTRL register). When “Get Timestamp” bit in TS_CTRL register is 1 the value in this register is not valid. Bits 15-0: RO Pulse Counter Value in the free running 32 bit counter that counts incoming events (rising edges). This counter can be used for finding out how many events there has been between certain Time Stamp captures. Useful with fast input signals. Bits 31:16.

9. GENERAL-PURPOSE IO (GPIO)

With the GPIO block the user can control the functionality of the General Purpose Input/Output pins of the chip. Each GPIO pin can be configured to be either an input or an output.

9.1 Registers

The register map is presented in Table 39.

Table 39. GPIO Block Register Map

Address	Register	Description
0x0000 … 0x0FFE	*Reserved*	*Reserved*
0x1000	CONFIG0	Reset: 0 x 00 00 General Purpose Input/Output Configuration Bits 1-0: R/W Configuration of GPIO pin 0 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high Bits 3-2: R/W Configuration of GPIO pin 1 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high Bits 5-4: R/W Configuration of GPIO pin 2 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high Bits 7-6: R/W Configuration of GPIO pin 3 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high Bits 9-8: R/W Configuration of GPIO pin 4 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high

Address	Register	Description			
		Bits	11-10:	R/W	Configuration of GPIO pin 5 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high
		Bits	13-12:	R/W	Configuration of GPIO pin 6 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high
		Bits	15-14:	R/W	Configuration of GPIO pin 7 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high
0x1002	INPUT_ STATUS0	Reset: 0 x 00 00 General Purpose Input Status.			
		Bits	1-0:	RO	Status of GPIO pin 0. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
		Bits	3-2:	RO	Status of GPIO pin 1. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
		Bits	5-4:	RO	Status of GPIO pin 2. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
		Bits	7-6:	RO	Status of GPIO pin 3. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved

Address	Register	Description			
		Bits	9-8:	RO	Status of GPIO pin 4. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
		Bits	11-10:	RO	Status of GPIO pin 5. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
		Bits	13-12:	RO	Status of GPIO pin 6. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
		Bits	15-14:	RO	Status of GPIO pin 7. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved
0x1004	CONFIG1	Reset: 0 x 00 00 General Purpose Input/Output Configuration			
		Bits	1-0:	R/W	Configuration of GPIO pin 8 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high
		Bits	3-2:	R/W	Configuration of GPIO pin 9
		Bits	5-4:	R/W	Configuration of GPIO pin 10
		Bits	7-6:	R/W	Configuration of GPIO pin 11
		Bits	9-8:	R/W	Configuration of GPIO pin 12
		Bits	11-10:	R/W	Configuration of GPIO pin 13
		Bits	13-12:	R/W	Configuration of GPIO pin 14
		Bits	15-14:	R/W	Configuration of GPIO pin 15

Address	Register	Description
0x1006	INPUT_ STATUS1	Reset: 0 x 00 00 General Purpose Input Status. Bits 1-0: RO — Status of GPIO pin 8. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved Bits 3-2: RO — Status of GPIO pin 9 Bits 5-4: RO — Status of GPIO pin 10 Bits 7-6: RO — Status of GPIO pin 11 Bits 9-8: RO — Status of GPIO pin 12 Bits 11-10: RO — Status of GPIO pin 13 Bits 13-12: RO — Status of GPIO pin 14 Bits 15-14: RO — Status of GPIO pin 15
0x1008	CONFIG2	Reset: 0 x 00 00 General Purpose Input/Output Configuration Bits 1-0: R/W — Configuration of GPIO pin 16 00 = Input 01 = reserved 10 = Output, driven low 11 = Output, driven high Bits 3-2: R/W — Configuration of GPIO pin 17 Bits 5-4: R/W — Configuration of GPIO pin 18 Bits 7-6: R/W — Configuration of GPIO pin 19 Bits 15-8: RO — Reserved
0x100A	INPUT_ STATUS2	Reset: 0 x 00 00 General Purpose Input Status. Bits 1-0: RO — Status of GPIO pin 16. Valid only when configured as input. 00 = low 01 = high 10 = reserved 11 = reserved Bits 3-2: RO — Status of GPIO pin 17 Bits 5-4: RO — Status of GPIO pin 18 Bits 7-6: RO — Status of GPIO pin 19 Bits 15-8: RO — *Reserved*

10. ELECTRICAL SPECIFICATION

10.1 Absolute Maximum Ratings

Table 40. Absolute Maximum Ratings, EQFP144 Package

Symbol	Parameter	Min	Max	Unit
V_{CC}	Supply Voltage for core and input and output buffers	-0.5	3.9	V
V_{CC_PLL}	Supply Voltage for internal PLLs	-0.5	3.9	V
V_I	DC input voltage	-0.5	4.12	V
I_{OUT}	DC output current per pin	-25	25	mA
T_{STG}	Storage temperature	-65	150	°C
T_J	Operating junction temperature	-40	125	°C

Table 41. Absolute Maximum Ratings, FBGA256 Package

Symbol	Parameter	Min	Max	Unit
V_{CC}	Supply voltage for core	-0.5	1.63	V
V_{CC_PLL}	1.2V supply voltage for internal PLLs	-0.5	1.63	V
V_{CCA}	2.5V supply voltage for internal PLLs	-0.5	3.41	V
V_{CCIO33}	Supply voltage for 3.3V input and output buffers	-0.5	3.9	V
V_{CCIO25}	Supply voltage for 2.5V input and output buffers	-0.5	3.9	V
V_{I33}	DC input voltage for 3.3V IO	-0.5	4.12	V
V_{I25}	DC input voltage for 2.5V IO	-0.5	4.12	V
I_{OUT}	DC output current per pin	-25	25	mA
T_{STG}	Storage temperature	-65	150	°C
T_J	Operating junction temperature	-40	125	°C

10.2 Recommended Operating Conditions

Table 42. Recommended Operating Conditions, EQFP144 Package

Symbol	Parameter	Min	Typ	Max	Unit
V_{CC}	Supply voltage for core and input and output buffers	3.135	3.3	3.45	V
V_{CC_PLL}	Supply voltage for internal PLLs	3.135	3.3	3.465	V
V_I	DC input voltage	-0.3	-	3.6	V
V_O	Output voltage	0	-	V_{CC}	V
T_J	Operating junction temperature	-40	-	100	°C
t_{RAMP}	Power supply ramp time	200 µs	-	3 ms	-

Table 43. Recommended Operating Conditions, FBGA256 Package

Symbol	Parameter	Min	Typ	Max	Unit
V_{CC}	Supply voltage for core	1.15	1.2	1.25	V

V_{CC_PLL}	1.2V supply voltage for internal PLLs	1.15	1.2	1.25	V
V_{CCA}	2.5V supply voltage for internal PLLs	2.375	2.5	2.625	V
V_{CCIO33}	Supply voltage for 3.3V input and output buffers	3.135	3.3	3.465	V
V_{CCIO25}	Supply voltage for 2.5V input and output buffers	2.375	2.5	2.625	V
V_{I33}	DC input voltage, 3.3V IO	-0.3	-	3.6	V
V_{I25}	DC input voltage, 2.5V IO	-0.3	-	2.8	V
V_O	Output voltage	0	-	V_{CCIO}	V
T_J	Operating junction temperature	-40	-	100	°C
t_{RAMP}	Power supply ramp time	200 µs	-	3 ms	-

10.3 Package Thermal Information

Table 44. Thermal Resistance, EQFP144 Package

Symbol	Parameter	Typ	Unit
θ_{JA}	Thermal resistance, junction to ambient, no air flow	21.1	°C/W
θ_{JA}	Thermal resistance, junction to ambient, air flow 0.5 m/s	17.3	°C/W
θ_{JA}	Thermal resistance, junction to ambient, air flow 1.0 m/s	16.2	°C/W
θ_{JA}	Thermal resistance, junction to ambient, air flow 2.0 m/s	13.9	°C/W
θ_{JC}	Thermal resistance, junction to case	5.5	°C/W
θ_{JB}	Thermal resistance, junction to board	7.6	°C/W

Table 45. Thermal Resistance, FBGA256 Package

Symbol	Parameter	Typ	Unit
θ_{JA}	Thermal resistance, junction to ambient, no air flow	24.3	°C/W
θ_{JA}	Thermal resistance, junction to ambient, air flow 0.5 m/s	21.5	°C/W
θ_{JA}	Thermal resistance, junction to ambient, air flow 1.0 m/s	19.6	°C/W
θ_{JA}	Thermal resistance, junction to ambient, air flow 2.0 m/s	18.0	°C/W
θ_{JC}	Thermal resistance, junction to case	4.5	°C/W
θ_{JB}	Thermal resistance, junction to board	12.1	°C/W

10.4 DC Electrical Characteristics

10.4.1 *Current consumption*

Table 46. Current Consumption, XRS7004E

Symbol	Parameter	Typ	Unit
I_{VCC} +I_{VCC_PLL}	Total current consumption, 3.3V, full traffic all ports	0.73	A

Table 47. Current Consumption, XRS7003E

Symbol	Parameter	Typ	Unit
I_{VCC} +I_{VCC_PLL}	Total current consumption, 3.3V, full traffic all ports	0.62	A

Table 48. Current Consumption, XRS7004F

Symbol	Parameter	Typ	Unit
I_{VCC}	Current consumption, VCC (1.2V), full traffic all ports	0.72	A
I_{VCC_PLL}	Current consumption, VCC_PLL (1.2V), full traffic all ports	0.02	A
I_{VCCA}	Current consumption, VCCA (2.5V), full traffic all ports	0.03	A
$I_{VCCIO33}$	Current consumption, VCCIO33, full traffic all ports	0.02	A
$I_{VCCIO25}$	Current consumption, VCCIO25, full traffic all ports	0.01	A

Table 49. Current Consumption, XRS7003F

Symbol	Parameter	Typ	Unit
I_{VCC}	Current consumption, VCC (1.2V), full traffic all ports	0.61	A
I_{VCC_PLL}	Current consumption, VCC_PLL (1.2V), full traffic all ports	0.02	A
I_{VCCA}	Current consumption, VCCA (2.5V), full traffic all ports	0.03	A
$I_{VCCIO33}$	Current consumption, VCCIO33, full traffic all ports	0.02	A
$I_{VCCIO25}$	Current consumption, VCCIO25, full traffic all ports	0.01	A

Table 50. Current Consumption, XRS3003F

Symbol	Parameter	Typ	Unit
I_{VCC}	Current consumption, VCC (1.2V), full traffic all ports	0.49	A
I_{VCC_PLL}	Current consumption, VCC_PLL (1.2V), full traffic all ports	0.02	A
I_{VCCA}	Current consumption, VCCA (2.5V), full traffic all ports	0.03	A
$I_{VCCIO33}$	Current consumption, VCCIO33, full traffic all ports	0.02	A
$I_{VCCIO25}$	Current consumption, VCCIO25, full traffic all ports	0.01	A

10.4.2 *I/O Characteristics*

Table 51. I/O Characteristics, EQFP144 Package

Symbol	Parameter	Min	Typ	Max	Unit
V_{IL}	Low level input voltage	-0.3	-	0.8	V
V_{IH}	High level input voltage	1.7	-	3.6	V
V_{OL}	Low level output voltage	0	-	0.45	V
V_{OH}	High level output voltage	2.4	-	V_{CC}	V
C_{IO}	Input capacitance	6	8	9	pF
I_I	Input pin leakage current (V_I=0V…V_{CC})	-10	-	10	µA
R_{PU}	Resistance of internal pull-up resistors	7	12	18	kΩ

Table 52. I/O Characteristics, FBGA256 Package

Symbol	Parameter	Min	Typ	Max	Unit
V_{IL33}	Low level input voltage, 3.3V IO	-0.3	-	0.8	V
V_{IL25}	Low level input voltage, 2.5V IO	-0.3	-	0.7	V
V_{IH33}	High level input voltage, 3.3V IO	1.7	-	3.6	V
V_{IH25}	High level input voltage, 2.5V IO	1.7	-	2.8	V
V_{OL33}	Low level output voltage, 3.3V IO	0	-	0.45	V
V_{OL25}	Low level output voltage, 2.5V IO	0	-	0.4	V
V_{OH33}	High level output voltage, 3.3V IO	2.4	-	V_{CCIO33}	V
V_{OH25}	High level output voltage, 2.5V IO	2.0	-	V_{CCIO25}	V
C_{IO}	Input capacitance	6	8	9	pF
I_I	Input pin leakage current (V_I=0V…VCCIO)	-10	-	10	µA
R_{PU}	Resistance of internal pull-up resistors	7	12	18	kΩ

10.5 Interface Timing

10.5.1 *Input Clock Timing*

Table 53. Input Clock Timing (CLK)

Symbol	Parameter	Min	Typ	Max	Unit
f_{CLK}	CLK clock frequency	25 -50 ppm	25	25 + 50 ppm	MHz
f_{CLK_DUTY}	CLK clock duty cycle	40	50	60	%
T_{J_CC}	CLK clock cycle-to-cycle jitter	-	-	200	ps
T_{J_RMS}	CLK RMS jitter 12 kHz to 20 MHz	-	-	3	ps

10.5.2 *Reset Timing*

Figure 35. Reset Timing

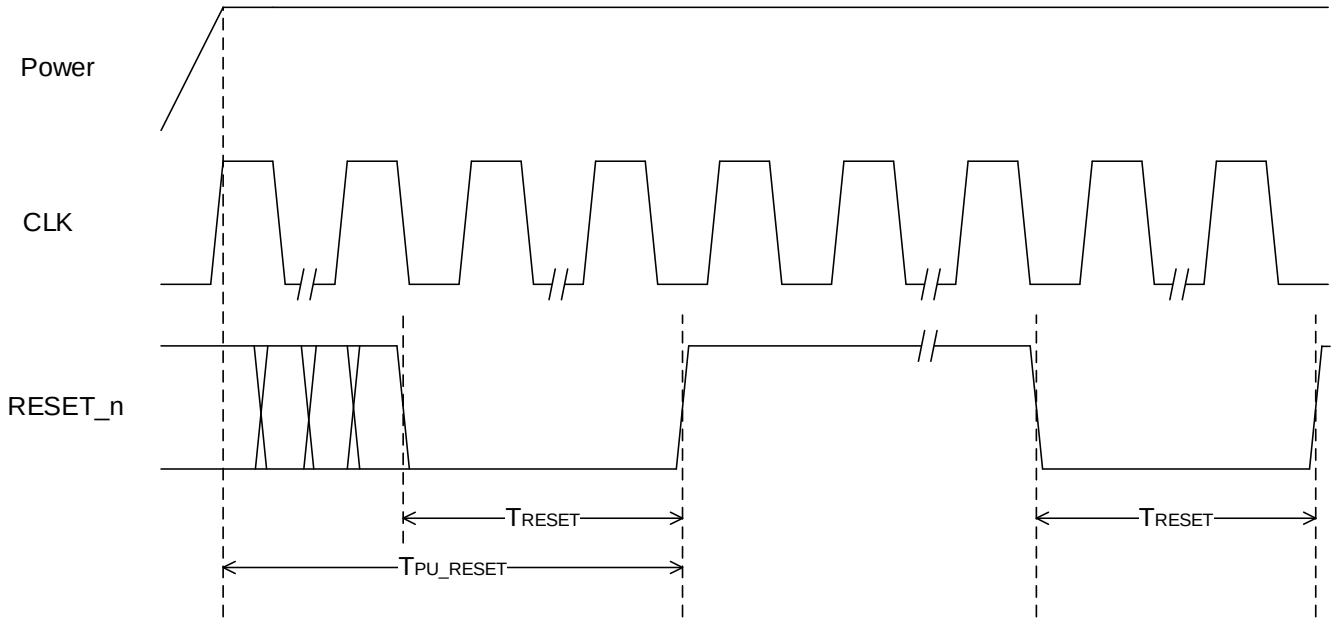


Table 54. Reset Timing

Symbol	Parameter	Min	Typ	Max	Unit
T_{RESET}	Reset pulse width	1	-	-	µs
T_{PU_RESET}	Time from power-up to reset deasserted	300	-	-	ms

10.5.3 *RGMII Timing*

Figure 36. RGMII Timing

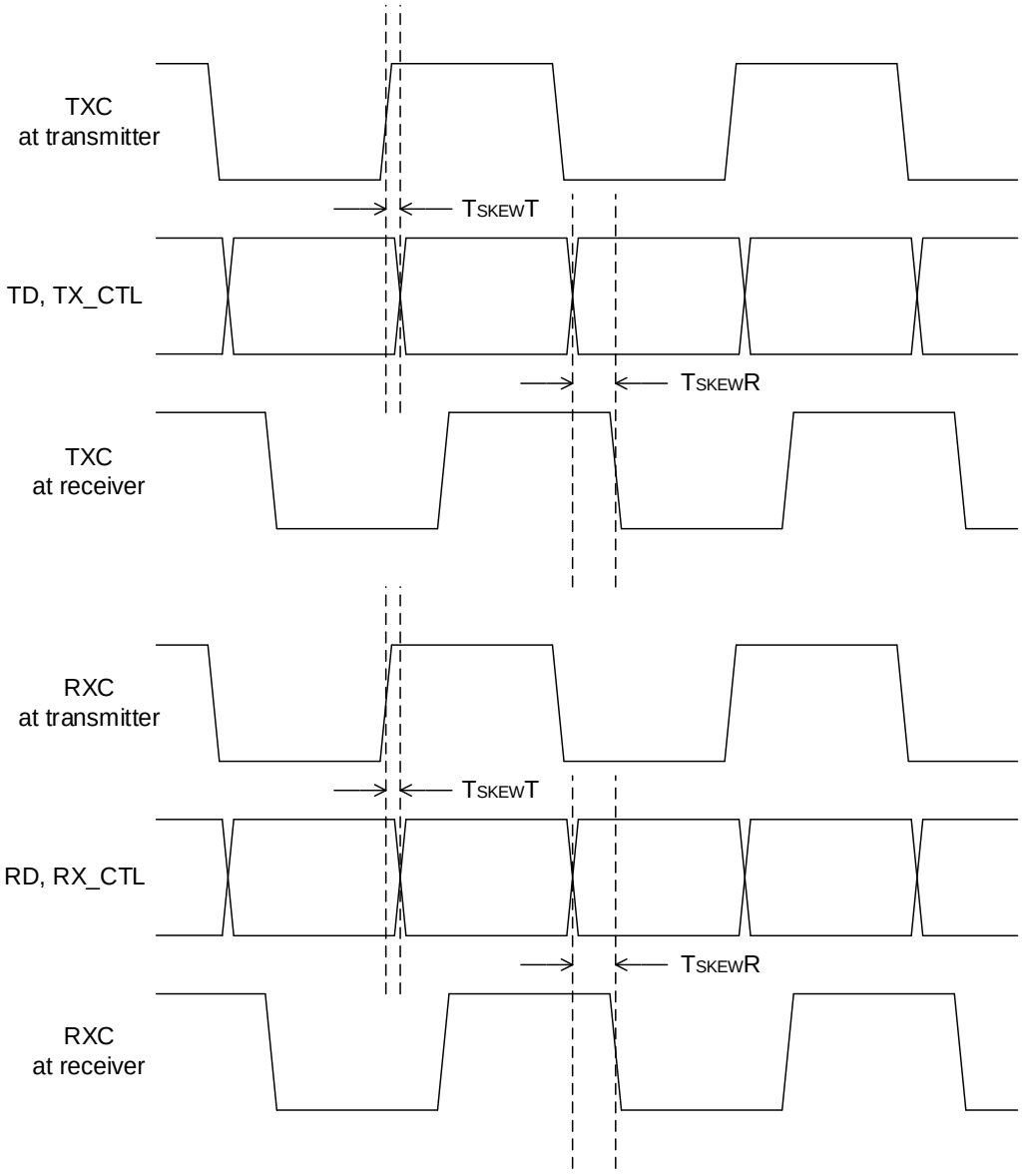


Table 55. RGMII Timing

Symbol	Parameter	Min	Typ	Max	Unit
$T_{SKEW}T$	Data to clock output at transmitter	-500	0	500	ps
$T_{SKEW}R$	Data to clock input at receiver	1	-	2.6	ns

Note that the PCB design needs to add 1.5 ns to 2.1 ns more delay to the RGMII clock signals unless there are adjustable clock delays in the other end (PHY).

10.5.4 *RMII Timing*

Figure 37. RMII Clocking Option 1

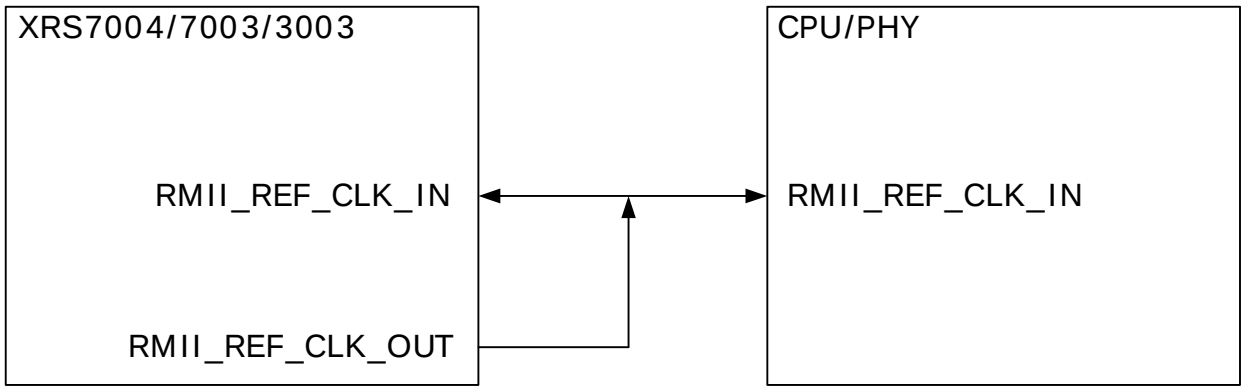


Figure 38. RMII Clocking Option 2

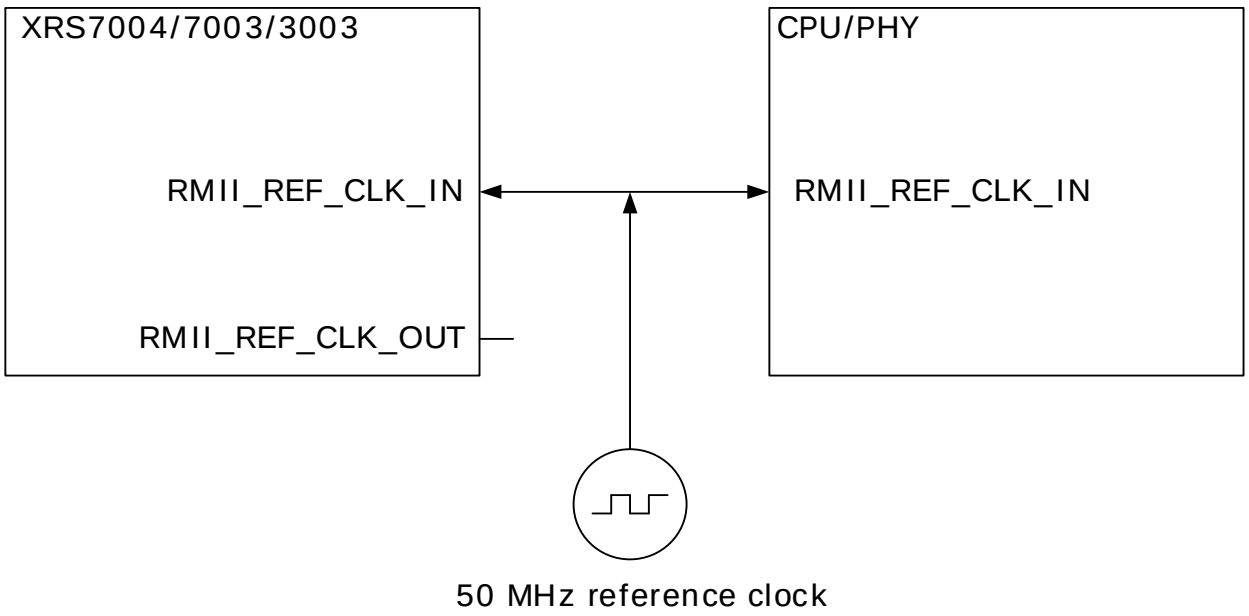


Figure 39. RMII Timing

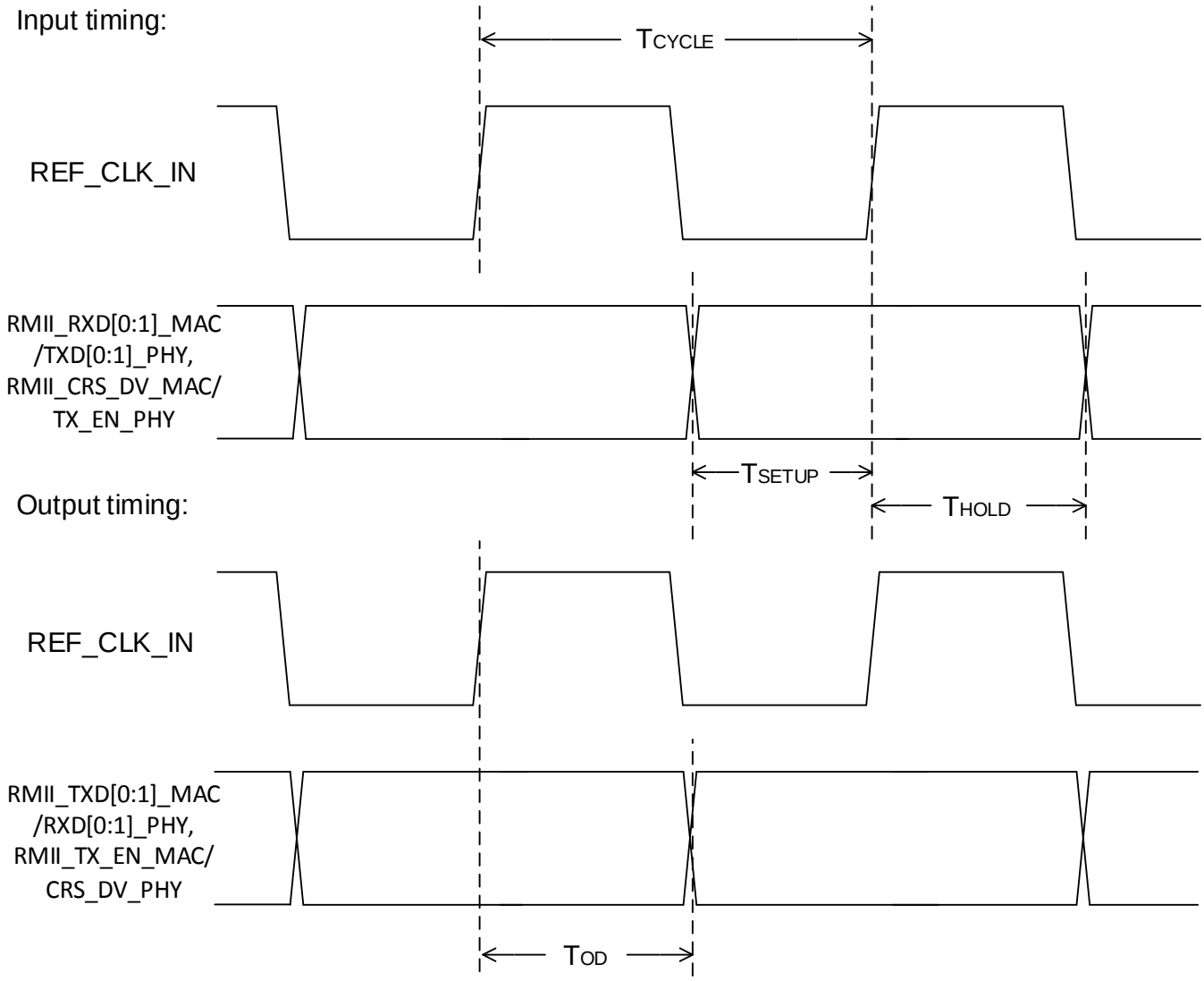


Table 56. RMII Timing

Symbol	Parameter	Min	Typ	Max	Unit
T_{CYCLE}	50 MHz Reference clock period	-	20	-	ns
T_{SETUP}	Input data setup time	2	-	-	ns
T_{HOLD}	Input data hold time	2	-	-	ns
T_{OD}	Output Delay, Time from rising edge of clock to output data valid	4	-	13	ns

10.5.5 *I2C Timing*

Figure 40. I2C Timing

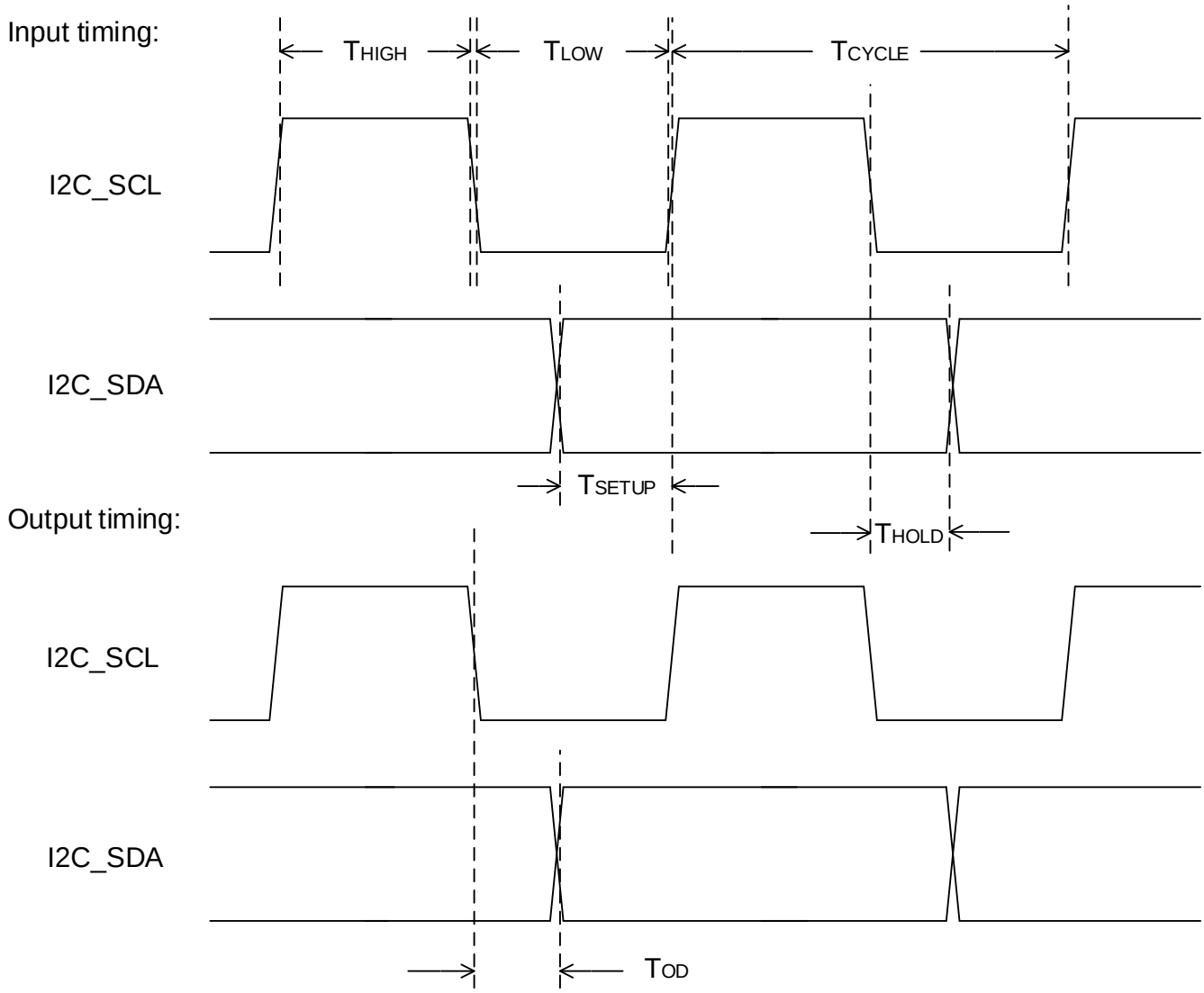


Table 57. I2C Timing

Symbol	Parameter	Min	Typ	Max	Unit
T_{HIGH}	Clock high period	600	-	-	ns
T_{LOW}	Clock low period	600	-	-	ns
T_{CYCLE}	Clock cycle period	2500	-	-	ns
T_{SETUP}	Input data setup time	50	-	-	ns
T_{HOLD}	Input data hold time	0	-	-	ns
T_{OD}	Output Delay, Time from falling edge of clock to output data valid	100	180	260	ns

10.5.6 *MDIO Timing*

Figure 41. MDIO Timing

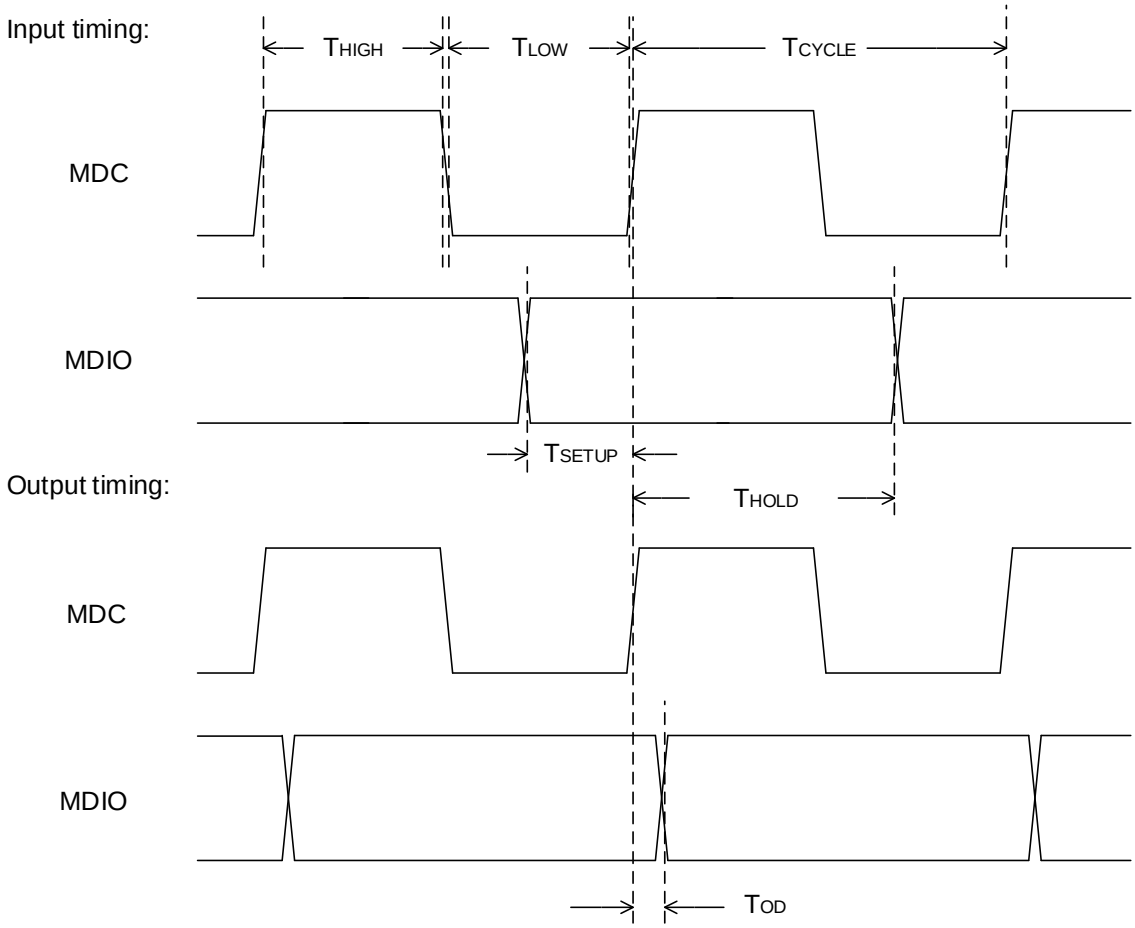


Table 58. MDIO Timing

Symbol	Parameter	Min	Typ	Max	Unit
T_{HIGH}	Clock high period	150	-	-	ns
T_{LOW}	Clock low period	150	-	-	ns
T_{CYCLE}	Clock cycle period	400	-	-	ns
T_{SETUP}	Input data setup time	10	-	-	ns
T_{HOLD}	Input data hold time	10	-	-	ns
T_{OD}	Output Delay, Time from rising edge of clock to output data valid	10	-	100	ns

11. MECHANICAL SPECIFICATION

11.1 EQFP144 Package

Package information is presented in Table 59 and package dimensions in Table 60 and Figure 42.

Table 59. EQFP144 Package Information

Leadframe material:	Copper
Lead finish (plating):	Matte Sn
JEDEC outline reference:	MS-026 variation: BFB-HD
Lead coplanarity:	0.08 mm
Weight:	1.4 g (Typ.)

Figure 42. EQFP144 Package Dimensions

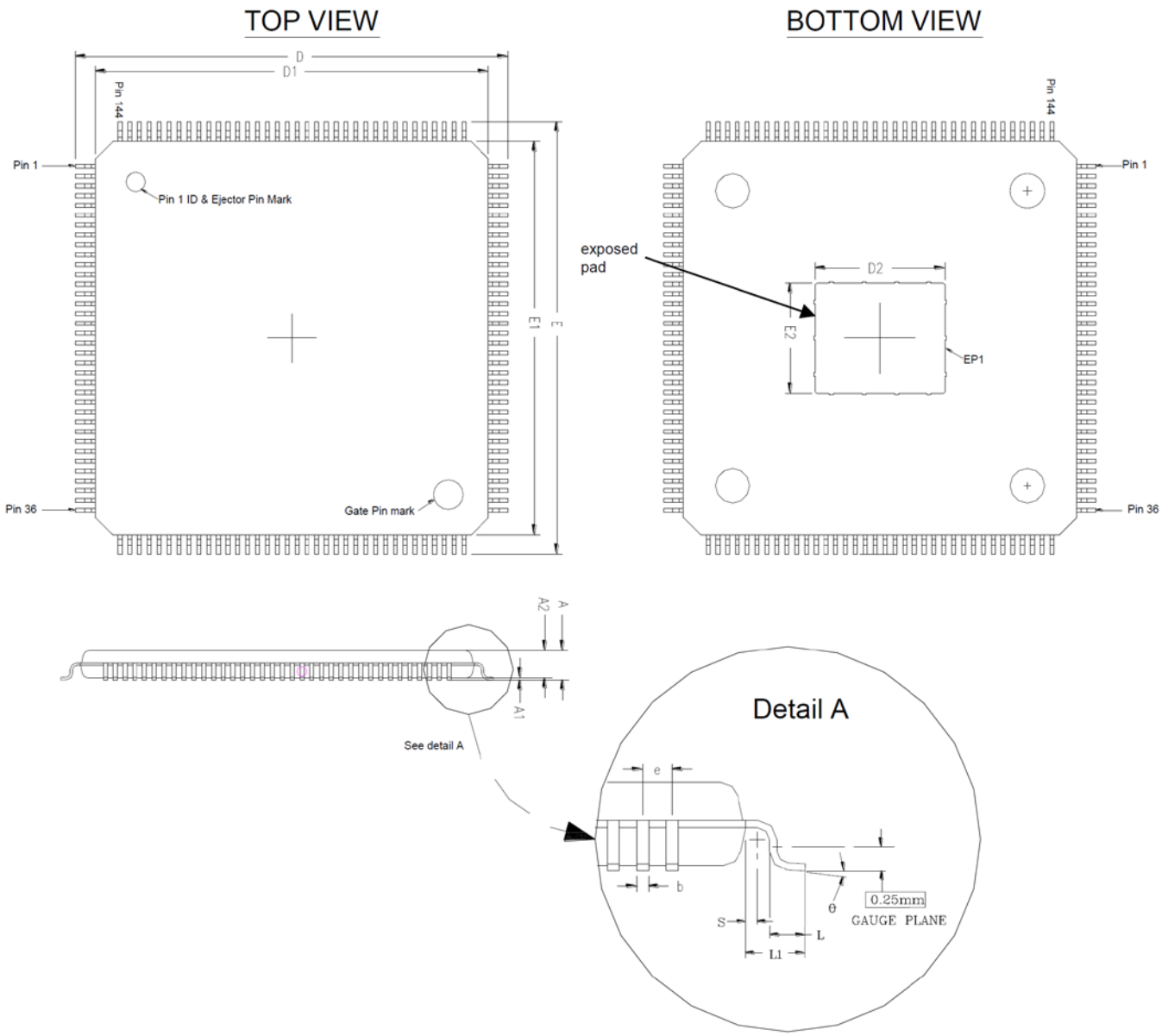


Table 60. EQFP144 Package Dimensions in Millimeters

Symbol	Min	Nom	Max
A	1.45	1.55	1.65
A1	0.05	0.10	0.15
A2	1.30	1.45	1.60
D		22.00 BSC	
D1		20.00 BSC	
D2	8.78	8.93	9.08

Symbol	Min	Nom	Max
E		22.00 BSC	
E1		20.00 BSC	
E2	8.55	8.70	8.85
L	0.45	0.60	0.75
L1		1.00 REF	
S	0.20	-	-
b	0.17	0.22	0.27
c	0.09	-	0.20
e		0.50 BSC	
θ	0°	3.5°	7°

11.2 FBGA256 Package

Package information is presented in Table 61Table 59 and package dimensions in Table 62 and Figure 43.

Table 61. FBGA256 Package Information

Solder ball composition	Sn:3Ag:0.5Cu (Typ.)
JEDEC outline reference	MO-192 variation: DAF-1
Lead coplanarity	0.20 mm
Weight	0.93 g (Typ.)

Figure 43. FBGA256 Package Dimensions

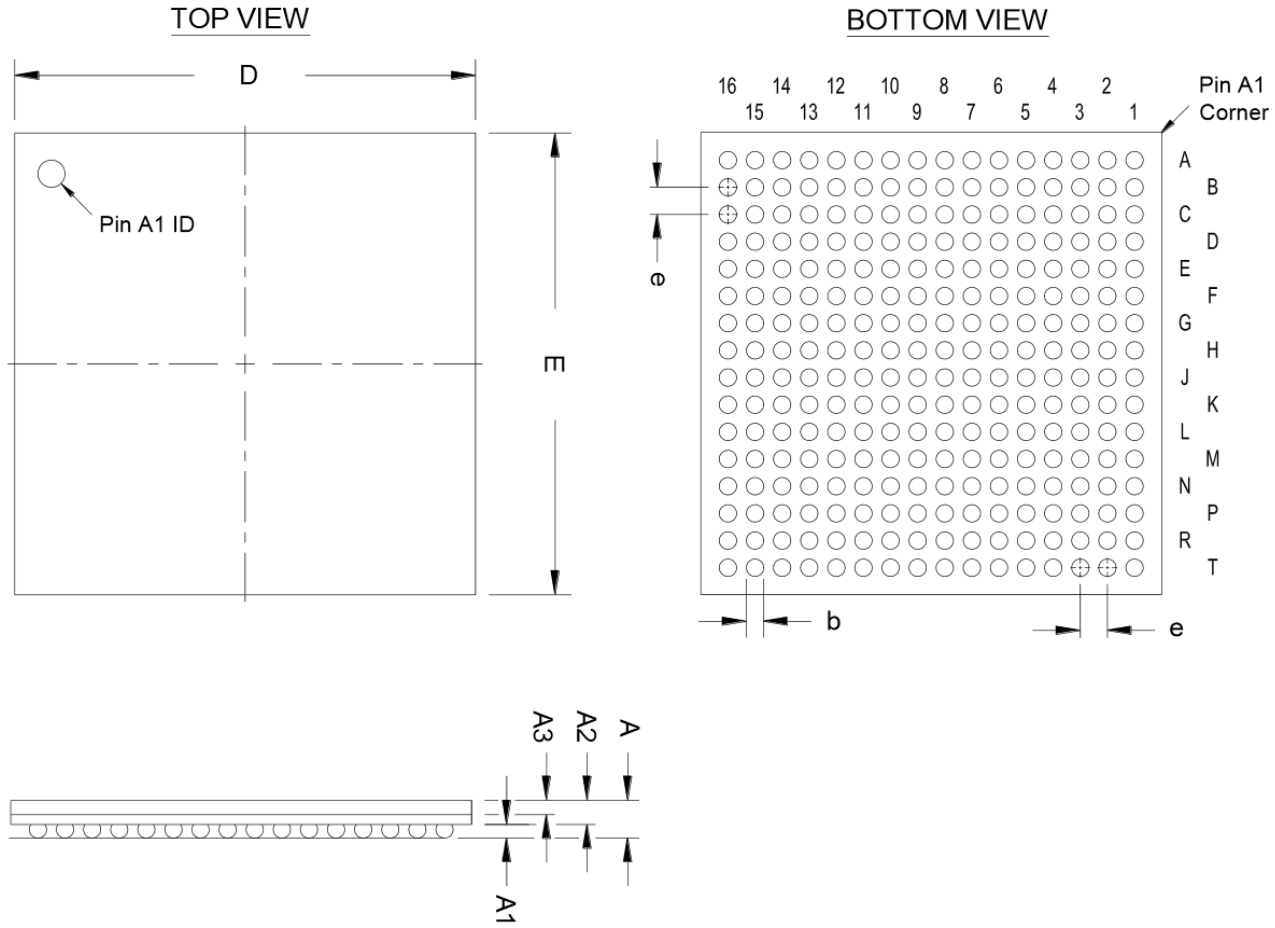


Table 62. FBGA256 Package Dimensions in Millimeters

Symbol	Min	Nom	Max
A	1.35	1.45	1.55
A1	0.30	0.40	0.50
A2	0.85	1.05	1.25
A3	0.65	0.70	0.75
D	17.00 BSC		
E	17.00 BSC		
b	0.40	0.50	0.60
e	1.00 BSC		

12. ORDERING INFORMATION

12.1 Part Ordering Numbers

Ordering numbers for parts are presented in Table 63. All parts are lead-free.

Table 63. Part Ordering Numbers

Device	Package	Part Ordering Number
XRS7003E	EQFP144	ARWSC-XRS7003E
XRS7004E	EQFP144	ARWSC-XRS7004E
XRS3003F	FBGA256	ARWSC-XRS3003F
XRS7003F	FBGA256	ARWSC-XRS7003F
XRS7004F	FBGA256	ARWSC-XRS7004F

12.2 Sales Offices

Arrow has more than 460 locations worldwide. An up to date list of Arrow sales offices and contact information is available at Arrow web page:

http://arrow.com/office_locations/

Arrow International telephone number (Europe, South America, Africa, Asia and Oceania): +800-8000-1010

Arrow telephone number (Canada, United States): 1-855-326-4757

Sales Hotlines:
China: (86) 400-886-1880
Hong Kong: (852) 2484-2112
Korea: (82) 2 2650 9700