FLEXIBILIS ETHERNET SWITCH (FES)
FLEXIBILIS REDUNDANT SWITCH (FRS)
Manual

Revision History

Rev	Date	Description
1.0	13.6.2011	First release.
1.1	14.10.2011	Minor improvements.
1.2	14.3.2012	- HSR configuration register modified - Configuration pins defined to give system clock frequency - PRP added - HSR updates
2.0	23.3.2012	Approved
2.1	1.6.2012	HSR_CFG register clarifications (Table 11)
2.2	11.6.2012	Added IEEE 1588 Layer 2 functionality and Peer-to-peer transparent clock
2.3	13.6.2012	Support for link local traffic without HSR tag or PRP trailer HSR_CFG register updates (Table 11)
2.4	28.6.2012	HSR_CFG register correction (Table 11)
2.5	29.6.2012	Changed PTP mode bits
2.6	15.8.2012	Minor changes
2.7	25.9.2012	Clock interface changed.
2.8	14.12.2012	Inbound Policy (IPO) registers changed.
2.9	29.1.2013	FES-HSR renamed to FRS
2.10	7.2.2013	IP License Authentication added
2.11	18.4.2013	Avalon interface, Ordinary/Boundary Clock support and host interrupts added. Management mode now possible for all the ports.
2.12	27.9.2013	HSR/PRP Interlink support added. And up to four FRS instances can now share a single security chip.
2.13	3.12.2013	New features: - Virtual LAN tagging and priority tagging - MII clock enable signals - 12-port support (HSR_PORT_OPT 3)
2.14	11.12.2013	New counter registers
2.15	29.1.2014	Added MAC address table read functionality and Time Trailer
2.16	22.4.2014	New features: Per port MAC table clear functionality IPO match to source address Generics changed: VLAN_OPT removed MGMT_PORT_OPT removed PORT_STATE_DEFAULT now 0x0120.
2.17	26.6.2014	Document moved to a new updated document template. Register tables changed from 16bit word addressing to byte addressing (just a documentation change, no change in actual register locations).
2.18	10.9.2014	New features: Cut-Through between HSR redundant ports Management Trailer length and bit offset now configurable Possibility to leave priority untouched in case of IPO match New generics: CUT_THROUGH enable and disable COUNTERS enable and disable

Rev	Date	Description
2.19	18.6.2015	HSR mode and management mode both allowed for a port at the same time. Specified more precisely what happens in case of two IPO matches. Added a warning about unnecessary clearing of the MAC address table. Address lifetime configuration change
2.20	10.3.2016	Added register settings for compensating TX and RX delay caused by PHYsical layer devices, interface adapters, SFP modules, etc.
3.0	30.6.2016	FES (Flexibilis Ethernet Switch), FRS (Flexibilis Redundant Switch) and FDS (Flexibilis Deterministic Switch) integrated into the same document. Filename changed from FRS_Manual to FES_Manual. New licensing model, new generics, HSR mode X support, MACsec support and other smaller changes.
3.1	8.12.2017	Added register bit for enabling/disabling support for independent VLANs. (Bit 7 in GENERAL register)
3.2	15.2.2021	FDS (Flexibilis Deterministic Switch) removed from the document and related features added as optional features for FRS.

This document could contain technical inaccuracies or typographical errors. Flexibilis Oy may make changes in the product described in this document at any time.

Please, email comments about this document to support@flexibilis.com.

Trademarks

Contents

Figures

Tables

1 About This Document

This document describes Flexibilis Ethernet Switch (FES). Flexibilis Redundant Switch (FRS) is FES with redundant networking features. FES is an Ethernet switch Intellectual Property (IP) core targeted at programmable hardware platforms. It is also possible to integrate FES into an ASIC. FES is written in VHDL language.

This document is targeted for current and potential users of FES. This includes those who are designing software employing the functions of FES, and those who are evaluating the usability of FES in their system or project. This documentation is not targeted at those who want to make changes in the implementation of FES, and therefore this document does not cover detailed information about the internal implementation of FES. The internal implementation details are documented in Flexibilis Ethernet Switch/Router Specification [1]. FPGA tools and their usage is also out of the scope of this document.

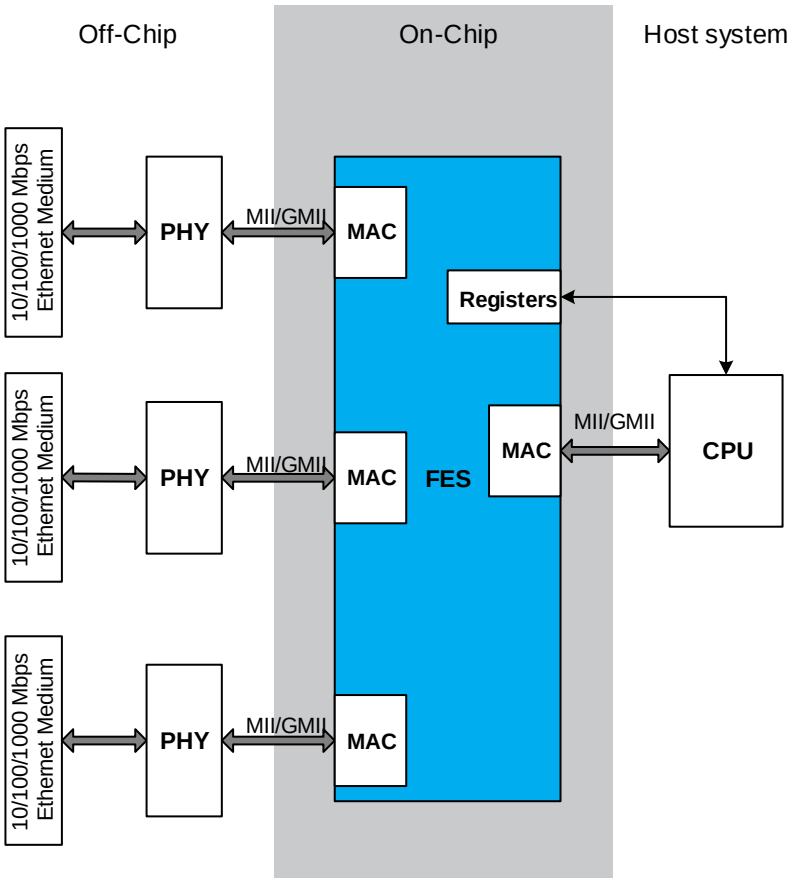
Chapter 2 gives a general overview of FES: what it is, what can be done with it, and what are the main features of it. Chapter 3 contains a short introduction to Ethernet and switching in general and chapter 4 describes FES functionality in more detail. Chapter 5 defines the register map of FES and Chapter 6 defines the external signals. Chapter 7 contains a glossary and chapter 8 contains the references.

In this document signal names are written in *signal_name* style. Block names are written with Capital first letter. Pseudo code is written in `PseudoCode` style and command line commands are written in `CommandLine` style. References are marked in the document in [brackets].

2 General Overview

FES is an Ethernet switch IP block designed to be used in programmable environments. FES includes multiple Ethernet Media Access Controller (MAC) functional entities and provides MII/GMII interfaces for Ethernet PHY devices and optionally for a host system CPU (see Figure 1).

Figure 1. FES Overview



2.1 FES Features

FES standard and optional features include:

- 10/100/1000 Mbit/s Full-Duplex Ethernet interfaces (IEEE 802.3x [2])
- Compatible with IEEE standard 802.1D Media Access Control (MAC) Bridges [3]
- Ethernet packet forwarding at wire-speed, non-blocking
- Media Independent Interfaces (MII) and Gigabit Media Independent Interfaces (GMII) for attaching to external Physical Layer devices (PHY) and host system CPU. Other interface types including SGMII, RGMII, RMII, 1000BASE-X and 100BASE-FX can be provided with optional interface adapters on the FPGA [4]
- Avalon slave interface for register access
- PTPv2 end-to-end one-step transparent clock [5] processing at hardware
- PTPv2 peer-to-peer transparent clock [5] support functions
- PTPv2 boundary and ordinary clock [5] support functions
- Ethernet packet filtering and prioritization on each of the ports

- Compatible with IEC 62439-3 ”High-availability Seamless Redundancy (HSR)”
- Compatible with IEC 62439-3 ”Parallel Redundancy Protocol (PRP)”
- Virtual LAN tagging and priority tagging
- Traffic shaping and policing
- Frame encryption and decryption with MACsec

2.2 FRS

FES with features for redundant communication is called Flexibilis Redundant Switch (FRS). FRS is targeted for applications requiring high availability. The standard and optional features of FES and FRS are listed in Table 1. Optional features need to be licensed separately.

Table 1. Features of FES and FRS

Feature:	**FES**	**FRS**
Ethernet switch	x	x
Number of ports*	3…12	3…8
Priority Queues per port	4	4
IEEE 1588	x	x
HSR	-	x
PRP	-	x
Static MAC table	-	o
Traffic Policing	-	o
Traffic Shaping	-	o
MACsec	-	o

“x” = standard, “o” = optional, “-“ = not available

* When redundancy (HSR/PRP) is enabled, the maximum port count is 8. Otherwise the maximum port count is 12.

It is possible to adjust the selected feature set during product development by licensing optional features and/or adjusting the generics (Table 16).

2.2.1 Interface between MAC and PHY

FES supports Media Independent Interface (MII) and Gigabit Media Independent Interface (GMII). GMII is used only when an Ethernet port is in 1000 Mbps operating mode. In 10 Mbps and 100 Mbps operating modes FES uses MII interface.

GMII interface contains more signals than MII interface, for example receive and transmit data paths are four bits wider. The signals that exist in both GMII interface and MII interface share the same signals in FES.

Reduced pin count interfaces (RMII and RGMII) are supported by using external to FES RMII and RGMII adapter blocks at the FPGA. Serial SGMII interfaces are supported with SGMII adapters, and for connecting to fiber optic transceivers there are 100BASE-FX and 1000BASE-X adapters. The available Ethernet interface adapters are described in a separate document [4].

2.2.2 Host System Interface

FES has a special feature in its Ethernet interfaces that is helpful when the interface is connected to a host system CPU. This feature of FES is called management mode. By using the management mode it is possible for the CPU to send/receive packets to/from any other port independent from the MAC address table and other configurations.

3 Introduction to Ethernet and Switching

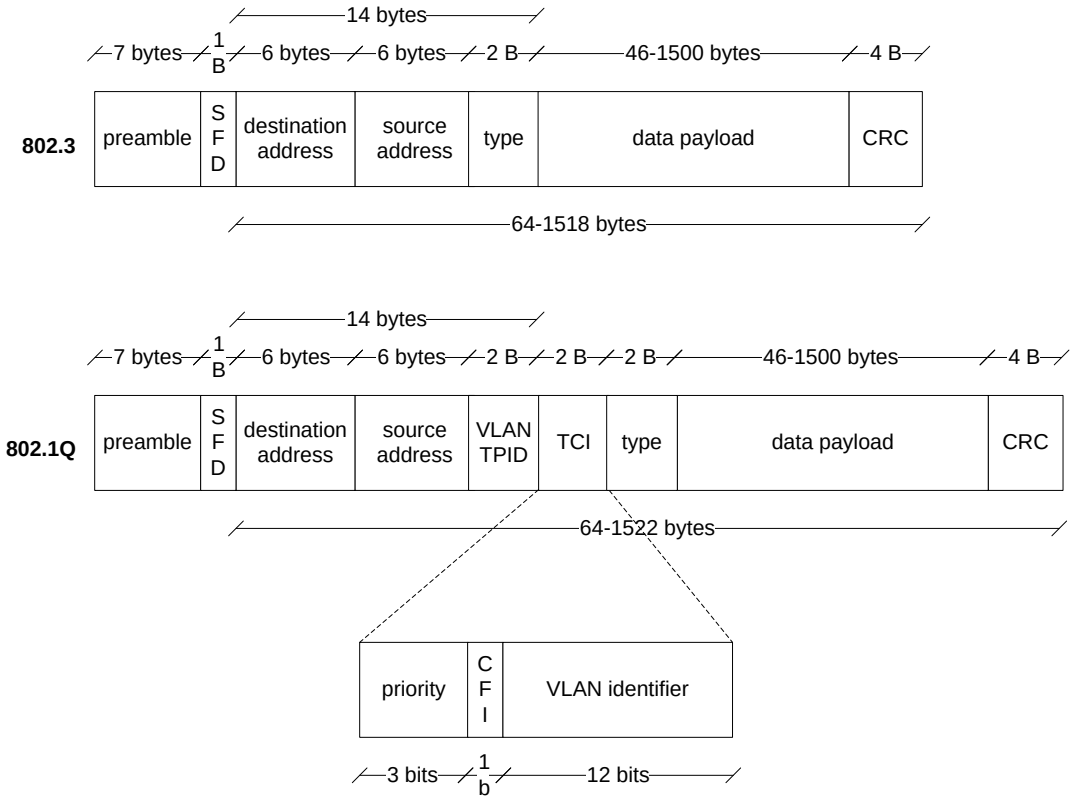
This chapter contains an introduction to Ethernet, Ethernet switching, Precision Time Protocol (IEEE 1588v2) and High-availability Seamless Redundancy (HSR).

3.1 Ethernet Media Access Control (MAC)

The Media Access Control (MAC) protocol is used to provide the data link layer of Ethernet protocol. The MAC protocol encapsulates data by adding a 14 byte header before the payload and a 32 bit Cyclic Redundancy Check (CRC) checksum after the payload. In addition to this, there is a 7 byte preamble and a 1 byte Start Frame Delimiter (SFD) before the header, see Figure 2.

In case of Virtual LAN (VLAN) tagging being used, the header is 4 bytes longer because of an additional type field and a VLAN tag. This also increases the Ethernet frame's maximum length from 1518 bytes to 1522 bytes (without preamble and SFD). This means that an Ethernet Bridge should support forwarding of up to at least 1522 byte frames.

Figure 2. Ethernet Frame Formats



3.1.1 Preamble and Start Frame Delimiter (SFD)

Between every frame, there is a small idle time called an Interframe Gap. The length of the Interframe Gap is 96 bits. After an Interframe Gap a node starts its transmission by sending a preamble sequence consisting of 56 alternating 1's and 0's. The purpose of the preamble sequence is to synchronize the receiver(s) before the actual data arrives. After the preamble comes a Start Frame Delimiter ("0x5D") that indicates the starting point of the actual frame (the header).

3.1.2 Header

MAC header consists of three parts:

- Destination Address field (6 bytes)
  The Destination Address specifies whether the recipient is a single node (unicast), a group of nodes

(multicast), or all the nodes at the medium (broadcast). If the first bit is zero, then the recipient is a single node and this field contains the physical address of the receiver's interface.

- Source address field (6 bytes)
  The physical address of the interface that originated the frame.
- Type/Length field (2 bytes)
  The type field indicates the protocol being carried. In the case of IEEE 802.3 LLC, this field can also be used to indicate the length of the data. If the value of the field is less than or equal to 1500, then it is indicating the length of the frame.

Ethernet nodes are usually capable of choosing whether they want to receive only frames sent to their own Ethernet address and the broadcast address, or also frames sent to some, or all of the multicast addresses. The nodes may also choose to receive all the frames, including those destined for other nodes. This is called promiscuous mode.

3.1.3 CRC

The 32-bit cyclic redundancy check (CRC) checksum at the end of the frame provides an error detection mechanism. The CRC checksum is calculated and added to the frame when the frame is sent, and it is checked when the frame is received at the opposite end. Frames received with an invalid CRC checksum contain an error and should be discarded by the receiver.

3.1.4 Media Independent Interface (MII/GMII)

Media Independent Interface (MII) is a standardized interface between MAC and PHY. MII interface can exist in different forms, it can for example be a physical connector between two devices or it can be just signals between two devices on a circuit board. It is also possible that a MII interface exists inside a single chip, between two functional blocks. The idea of the MII interface is that it is independent of the physical medium. In practise this means that the same Ethernet MAC can be used with various kinds of Ethernet media. The MII interface standard supports both 10 Mbps and 100 Mbps transfer rates. For gigabit media there is GMII (Gigabit Medium Independent Interface).

3.2 Media Access Control (MAC) Bridge

MAC Bridges allow communications between end stations attached to separate LANs (network segments). A bridge has an own separate MAC for each LAN it connects to and it is able to bridge traffic between the LANs transparent to logical link control (LLC) and network layer protocols, just as if the stations were attached to the same LAN. MAC Bridges with more than two ports are commonly called as switching hubs or Ethernet switches.

Functional entities of an Ethernet switch can be divided roughly into three parts: Forwarding Process, Address Learning Process and MAC address table (forwarding database). The Forwarding Process forwards Ethernet frames between the ports according to MAC address table updated by the Address Learning Process. The Address Learning Process observes all the received frames and learns which stations are in which network segments by storing their MAC addresses into the MAC address table.

Ethernet frames whose destination is known by the switch to be in the network segment behind another port are forwarded into that port. If the destination is known to be behind the same port where from the frame was received, the switch discards the frame. Ethernet frames whose destination is unknown (the address is not yet stored into the MAC address table, or the address information is deleted from the MAC address table because of being too old) are forwarded to every other port than the source port. Also frames with broadcast (and multicast) destination addresses are forwarded by switches to every other port that the source port as they are supposed to reach every station in the network.

3.2.1 MAC Address Table

The operations on the MAC address table can be broke down to three different processes; a learning process, a lookup process and an aging process.

The learning process compares the source addresses of the received frames to the addresses in the MAC address table. If the entry is found to be already in the table, the port mapping information is updated if it has changed (the network topology has changed), and the entry is refreshed so that the aging process does not

remove it prematurely. If the address entry cannot be found from the table, it is added there. If the table is already full it may be necessary to remove some other entry from the table. Note that if the network topology changes and nodes are moved from a switch port to another, the switch will not have correct information on the whereabouts of the node until it transmits something or the aging process removes the entry. Until then frames destined to the node will be forwarded to wrong port and they will never reach their destination.

The lookup process compares the destination addresses of the received frames to the addresses in the MAC address table. According to this information the forwarding process either discards the frame or forwards it to another or to all other ports. Note that frames are never forwarded to the port where from they were received, as that would cause duplicate frames to the network.

The aging process removes entries from the MAC address table. The entries are removed when they have not been refreshed by the learning process for some time. In many switches this aging time is configurable, but not always, in which case the default value of 5 minutes is typically used. The aging process helps to keep the MAC address table small, which may in some cases affect the time taken by address lookup (depends on the search algorithm used). The other reason for aging process to exist is to be able to react to network topology changes in case there are nodes that do not transmit too often or nodes that do not initiate any communication by themselves.

3.2.2 Shared Media versus Dedicated Media

“There are no shared-medium implementations of Ethernet at data rates above 10Mb/s.” [6]

Unfortunately still today many Ethernet tutorials present Ethernet as a shared medium despite the fact that typical Ethernet network has not been a shared medium for a decade. Shared Ethernet, where technologies like Carrier Sense Multiple Access (CSMA) and Collision Detection (CD) are used in gaining access to the physical medium, is today legacy technology. Today’s Ethernet is switched, which means that every end node has a dedicated port in an Ethernet switch. In switched Ethernet every network segment consists of an end-node, an Ethernet switch and a point-to-point link between them.

Switched Ethernet has many advantages over a shared media:

- Full-Duplex links offer double the capacity of a Half-Duplex link at the same nominal speed
- Different nodes are able to operate with different data rates
- Network capacity is used more efficiently and latency minimized as all the frames are not forwarded to all the nodes and links
- Full-Duplex operation allows links to be longer in distance because CSMA/CD protocol does not limit the maximum length. Today fiber optic Ethernet links can have range of over 100 kilometres.

3.2.3 Cut-through versus Store-and-Forward Operation

According to the IEEE standard 802.1D “Media Access (MAC) Bridges” [3] every port of the MAC bridge has an individual MAC entity examining all the frames transmitted to the medium by the other node(s). The CRC checksums of the received frames are checked by calculating CRC of the received frames and comparing it to the CRC checksum at the end of the frame. Frames with erroneous CRC checksums should be silently discarded.

The so called cut-through switches that start to forward the frame before it is fully received, do not conform with the standard in this respect, because they are not able to discard the received frames if they have an erroneous CRC checksum. The opposite of cut-through operation is store-and-forward operation, in which case frames are fully received and their CRC checksums are checked before they are forwarded. This is standard compliant method of operation.

3.2.4 Spanning Tree Protocol

Spanning Tree Protocol (STP) is a protocol that automatically removes loops from a switched Ethernet network. A loop-free topology is achieved by disabling some of the network links, to form a tree topology from the mesh topology. Loop-free topology is needed, because if there was a loop in the network Ethernet frames would circulate in the loop for infinite time.

According to the standard (IEEE 802.1D) the Spanning Tree Protocol configures full, simple, and symmetric connectivity throughout a Bridged Local Area Network that comprises individual LANs interconnected by Bridges. The Spanning Tree Protocol (STP) configures the Port State of each Bridge Port in the Bridged Local

Area Network. STP ensures that the stable connectivity provided by each Bridge between its Ports and by the individual LANs to which those Ports attach is predictable, manageable, full, simple, and symmetric. STP further ensures that temporary loops in the active topology do not occur if the network has to reconfigure in response to the failure, removal, or addition of a network component, and that erroneous station location information is removed from the Filtering Database after reconfiguration.[3]

STP and its improved version Rapid Spanning Tree Protocol (RSTP)[7] use configuration messages sent to a specific multicast MAC address to create and maintain the network topology. In addition to the STP protocol stack, support for STP requires the Bridge to support certain port states (Disabled, Learning and Forwarding) and capture the STP configuration messages for the STP protocol stack.

3.3 Precision Time Protocol

Precision Time Protocol (PTP), defined in IEEE standard 1588 [5], is a protocol enabling precise synchronization of device clocks in packet based networks, for example Ethernet. Devices running PTP are automatically synchronized to the most accurate clock in the network. The protocol supports system wide synchronization accuracy in sub microsecond range with minimal network and local clock computing resources. It is used for example by test and measurement, power-line management, industrial automation and telecom applications.

PTP accuracy is based on an assumption that the delay in Ethernet is approximately constant and symmetric. Because of the packet based traffic in Ethernet, Ethernet switches in the path of the packets cause variable delay for packet throughput, thus degrading PTP synchronization accuracy. Transparent PTP clock functionality removes these problems and enables precise synchronization of clocks in switched Ethernet.

3.3.1 Transparent Clock

Transparent Clock timestamps PTP event frames in receive and in transmission ports and calculates the delay caused by the switch by subtracting the receive timestamp from the transmission timestamp. In PTP end-to-end transparent one step switch, the calculated delay is added to the correction field of the PTP event frame. A PTP slave that receives the PTP event message corrects the delay calculation by removing the effect of the PTP transparent clocks by subtracting the correction field value from the calculated total packet transmission delay. PTP peer-to-peer transparent clock measures the delay of the ingress path and includes that also in the correction field value.

3.4 HSR (High-availability Seamless Redundancy)

Standard IEC62439-3 [9] deals with redundancy in Ethernet networks. The HSR concept introduces network ring(s), where each possible source and destination pair is always connected via two routes. In case of a fault, the ring breaks, but still provides connection between source and destination(s) via second path, as shown in Figure 3 and Figure 4. HSR can also be used with double LAN topology as depicted in Figure 7 (if SANs are attached directly to LAN_A or LAN_B, they need to have HSR support). The standard is developed for demanding and mission-critical applications such as substation automation.

Figure 3. HSR Ring Example with Unicast Message

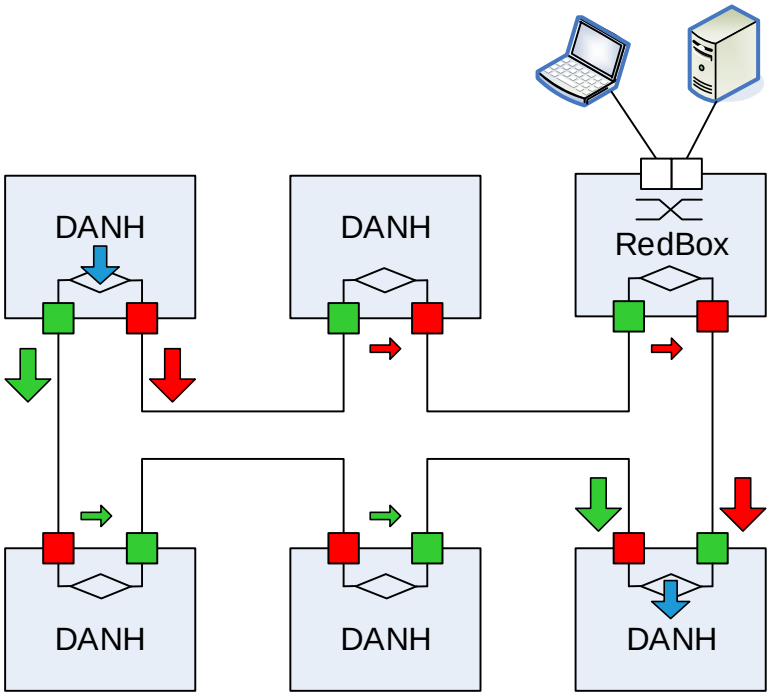
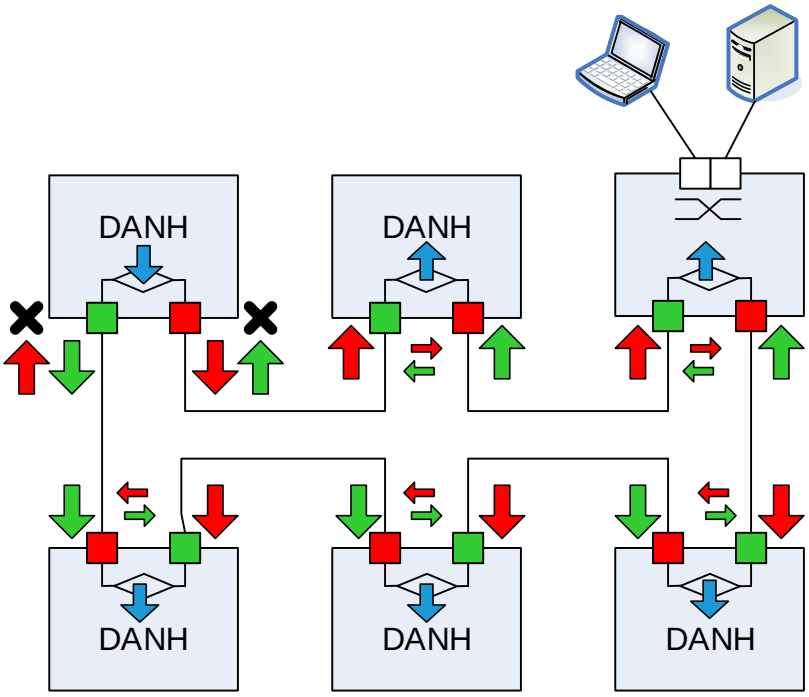
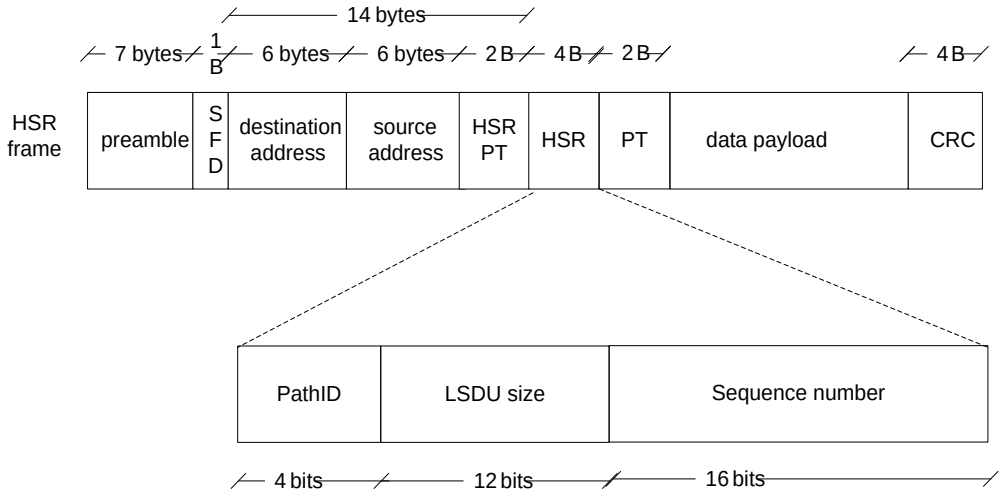


Figure 4. HSR Ring Example with Multicast Message



A frame in a ring is always HSR tagged. HSR tag in an Ethernet frame is presented in Figure 5. Tags are added / removed by the nodes connected to ring and ring exterior. Such nodes are called Redundancy Boxes (RedBox). Source nodes send always two copies (red and green arrows in Figure 4 and Figure 5) of the original frame (blue arrow) to the ring. The intermediate nodes in the ring forward the frames and the destination node discards the duplicate (the frame that arrives later). The duplicate frames are identified by having the same source MAC address and sequence number. In case the frame travels full round (in case of unicast this happens when destination is not found, in case of multicast this happens always), the source node takes care of removing the frame from ring (X-marking in Figure 4).

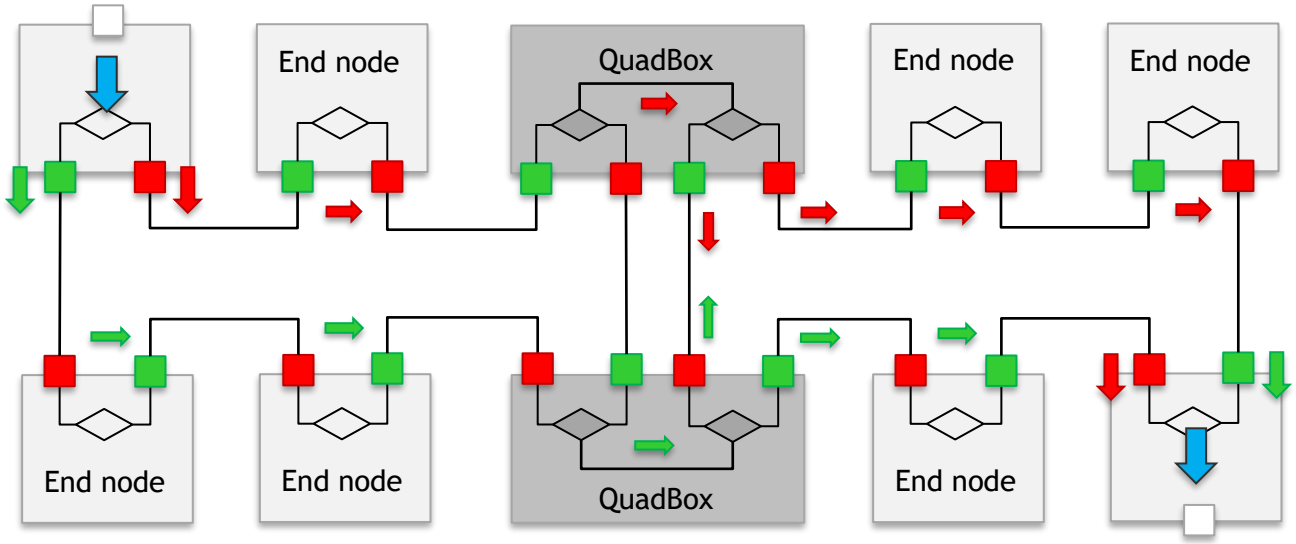
Figure 5. HSR Tagged frame



3.4.1 HSR Node Types

Normal nodes that connect only to one ring and therefore have only two ports are called HSR End-nodes. Also name Double Attached Node Implementing HSR (DANH) is used [9]. RedBox is a device that connects non-HSR node(s) or subnetwork(s) to the ring (see Figure 4). Two HSR rings can be connected together using a device named QuadBox. To prevent single point of failure, two QuadBoxes are needed as presented in Figure 6. With these building blocks also more complex network topologies, such as rings of rings etc., can be built.

Figure 6. QuadBox

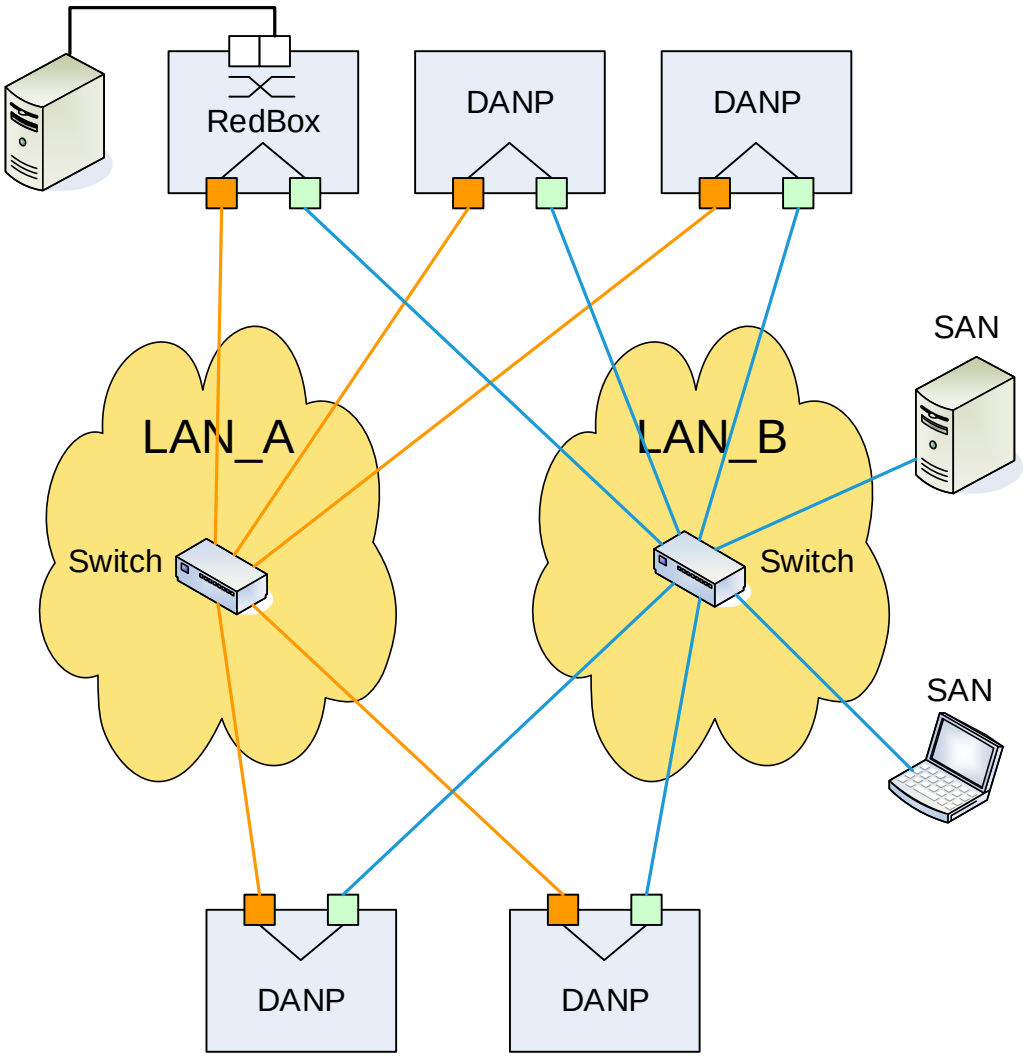


FES can be used as a part of HSR RedBox and HSR QuadBox design as well as HSR End-node.

3.5 Parallel Redundancy Protocol (PRP)

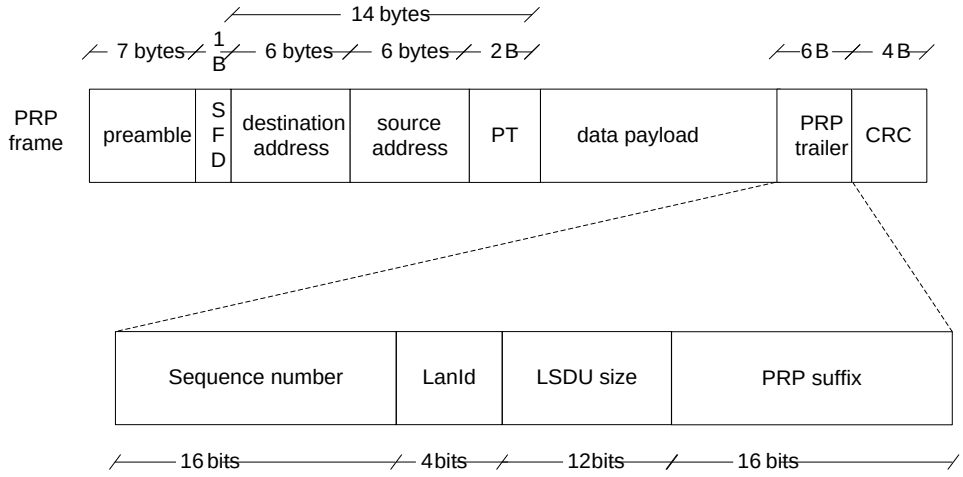
In addition to HSR standard IEC62439-3 [9] defines also PRP redundancy method. The PRP concept introduces double LAN networks, where each possible source and destination pair is always connected via two routes. In case of a single fault, the network still provides connection between source and destination(s) via second path, as shown in Figure 7. The standard is developed for demanding and mission-critical applications such as substation automation.

Figure 7. PRP Double LAN



Each PRP frame contains a PRP trailer as depicted in Figure 8. Trailers are added / removed by the PRP nodes connected to a PRP network. Normal non-PRP Ethernet switches can be used to form PRP LAN_A and LAN_B network segments. Redundancy Box (RedBox) can be used to connect non-PRP (SAN) devices to a PRP network. In PRP protocol the source node sends always two copies of the original frame, one for both network segments (LAN_A and LAN_B), and the destination node(s) discards the duplicate (the frame that arrives later). The duplicate frames are identified by having the same source MAC address and sequence number. Non-PRP (SAN) devices are also allowed to connect to LAN_A and LAN_B directly without a RedBox. In that case SANs do not benefit from the redundancy.

Figure 8. PRP Frame



3.5.1 PRP Node Types

Normal nodes that connect to PRP network are called PRP End-nodes. Also name Double Attached Node Implementing PRP (DANP) is used [9]. RedBox is a device that connects non-PRP node(s) or subnetwork(s) to the PRP network (see Figure 7).

FES can be used as a part of a PRP RedBox design as well as PRP End-node.

3.6 Traffic Shaping and Policing

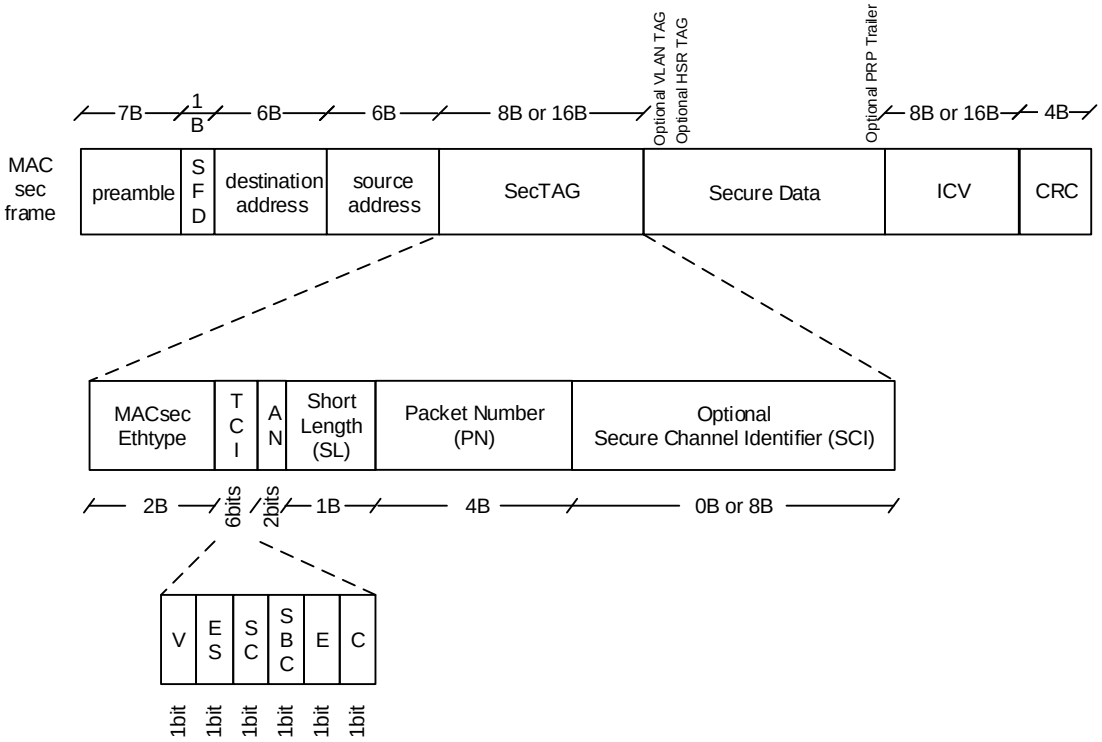
Traffic shaping and traffic policing are techniques for rate limiting. Traffic rate is measured and if the rate exceeds the configured rate, actions are taken to force the rate to comply with the configured profile.

The difference between shaping and policing is that a shaper is able to delay frames and a policer just drops the excess traffic. Therefore shaping also consumes more hardware resources than policing as it requires buffering of the frames. As no buffer is infinite in size shaping may also lead into dropping of frames if the traffic rate exceeds the configured rate continuously.

3.7 Data Authenticity

In Ethernet the authenticity of the data origin can be guaranteed by using the protocol specified in IEEE standard 801.AE [10] also known as MACsec. The authentication ensures that unauthorized devices cannot access the network. The MACsec protocol encapsulates the user data by adding a MAC Security Tag (SecTAG) before the data and Integrity Check Value (ICV) after the data, see Figure 9.

Figure 9. MACsec Frame



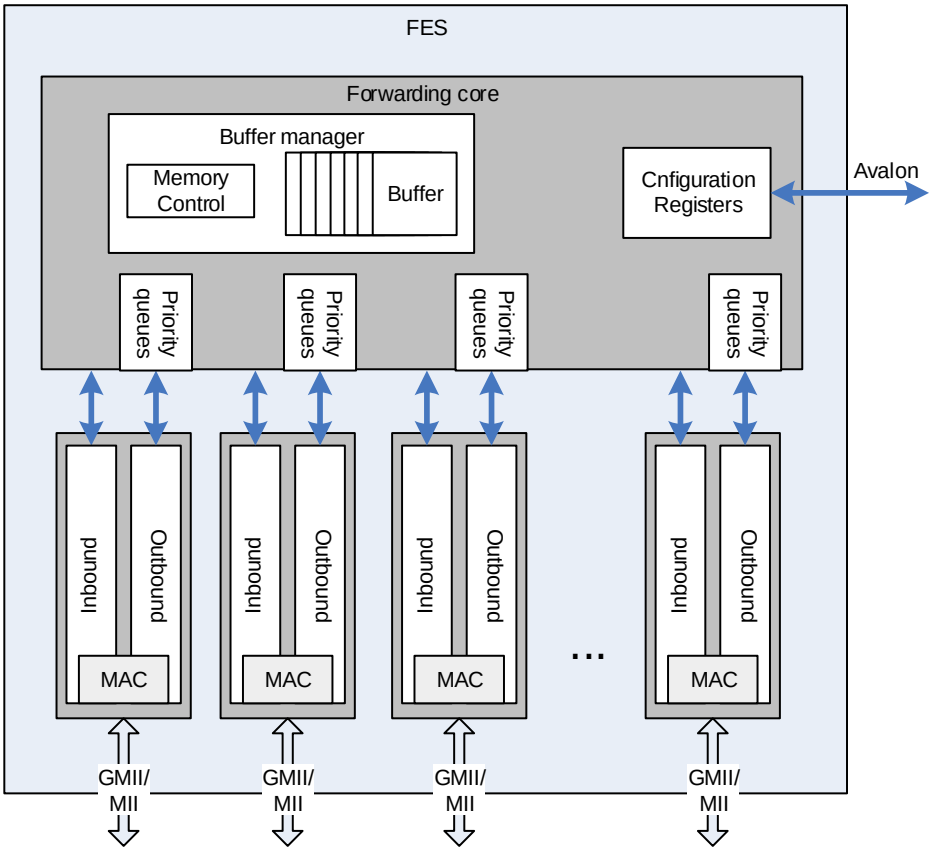
The explanation of the bit-fields in the SecTAG is the following:

- V bit: Version. Must be 0.
- ES bit: Indicates End Station. If ES bit is not set source MAC address cannot be used for determining SCI.
- SC bit: Indicates whether SCI is present or not (8 byte or 16 byte SecTAG).
- SCB bit: For Single Broadcast Copy functionality of Ethernet Passive Optical Networks (EPON).
- E bit: Indicates if Secure Data is encrypted.
- C bit: If not set, indicates that Secure Data in the MACsec frame is exactly the same as the original User Data and if the ICV is 16 bytes long.

4 Functional Description

The functional blocks of FES are presented in Figure 10. The functionality of FES can be controlled via configuration registers accessed using Avalon Slave interface. The configuration registers are defined in Chapter 5.

Figure 10. FES Block Diagram



FES 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. The forwarding core does the memory management for the buffer memory and it is also responsible for queuing of the frames.

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 11. FES Forwarding Path

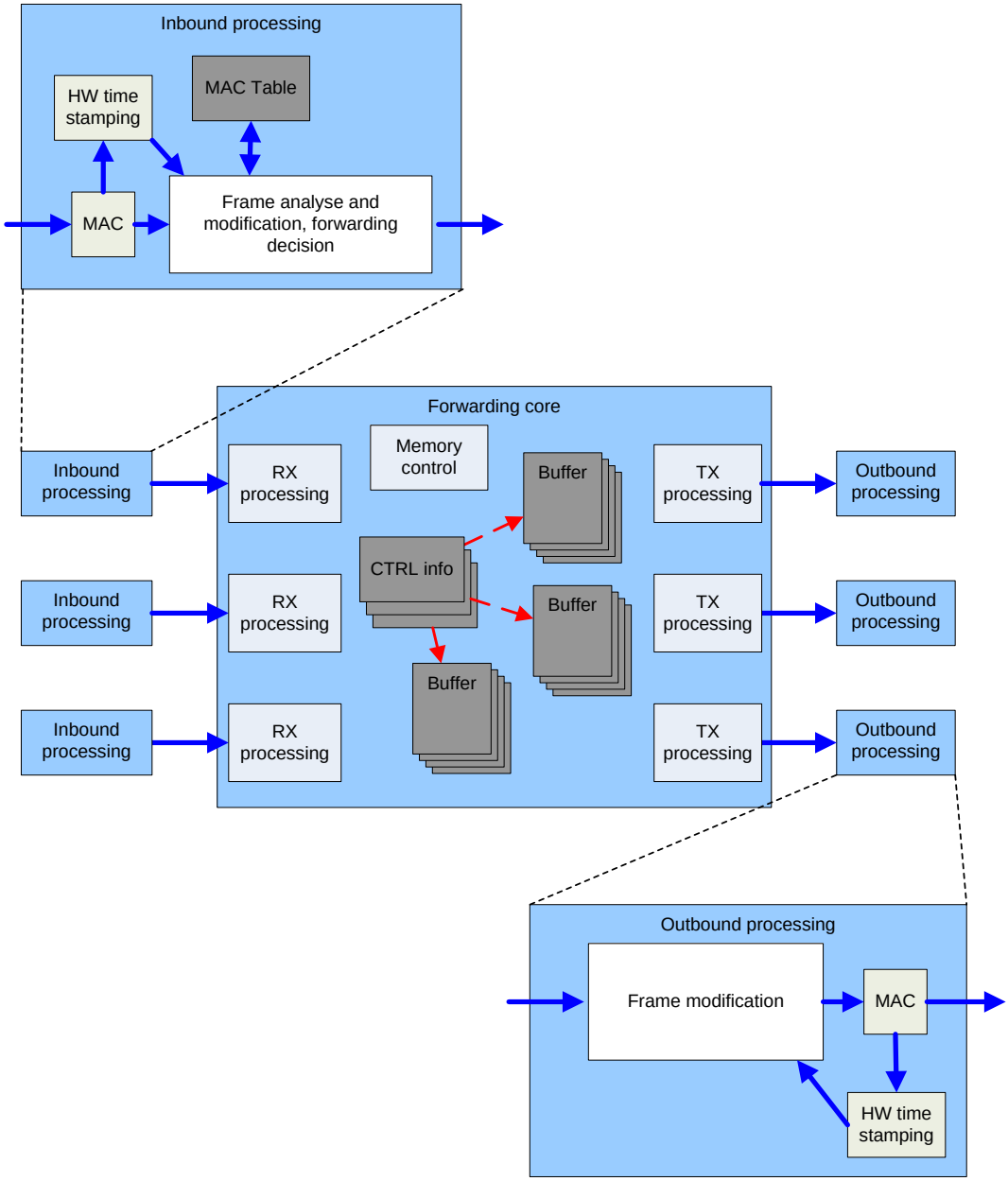


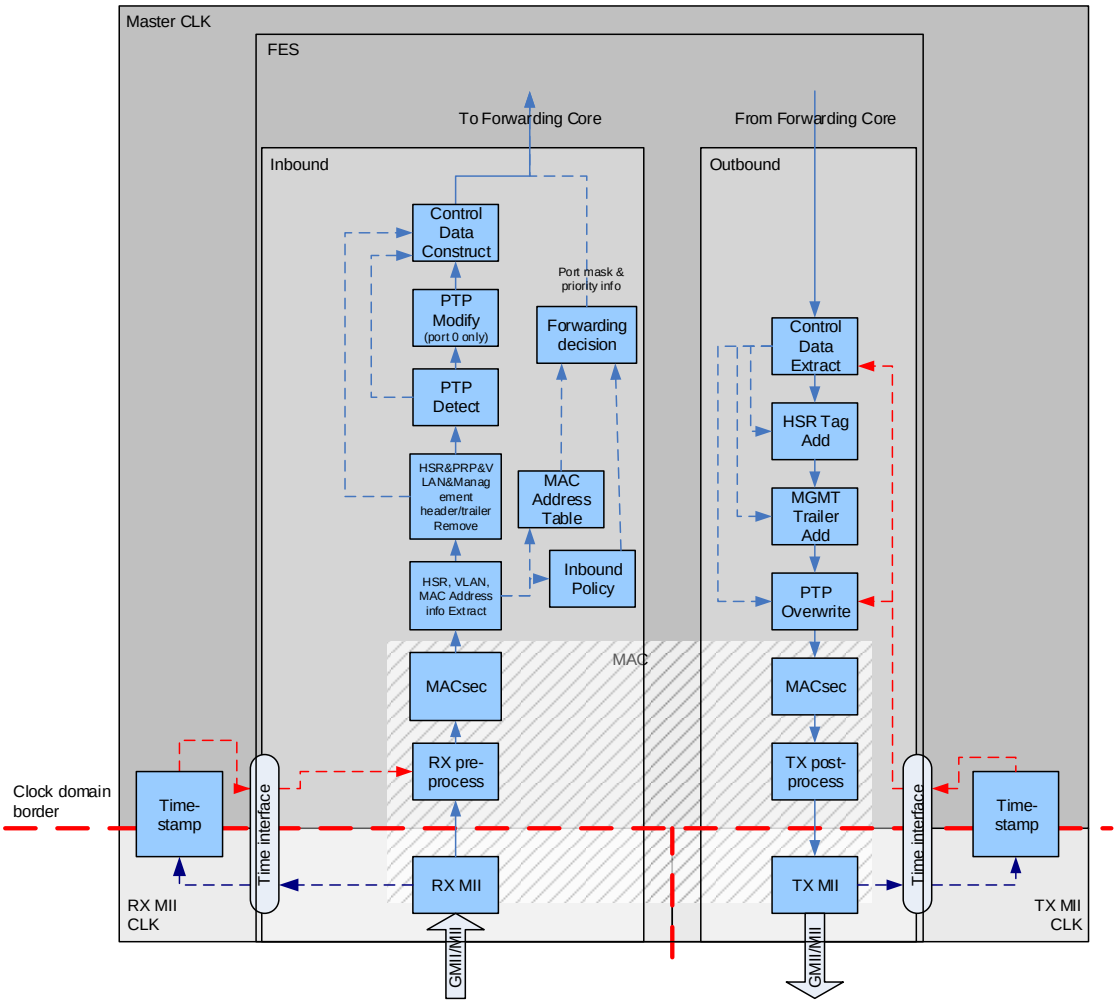
Figure 11 presents the processing path of a forwarded frame inside FES. The path contains Inbound processing, Outbound processing and Forwarding core functions.

Inbound processing contains Ethernet MAC and data processing blocks that are able to analyze and modify the frame. Every port has its own Inbound processing functionality, that uses MAC address table common to all ports.

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 that what its capacity is. Frame data is stored into a buffer memory, and the state of the frame is managed via a CTRL info data structure. Every frame is associated with one CTRL info structure. CTRL info data structure contains all the needed information about the state of the frame, including the destination port and the whereabouts of the memory buffers where the frame data is stored. 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 from 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 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 12. FES Inbound and Outbound Processing Blocks



FES Inbound and Outbound processing paths are depicted in Figure 12. The Inbound processing and Outbound processing functionality are completely independent of each other. The Inbound processing and Outbound processing data paths are designed so that it is relatively easy to add more processing blocks to them. Such blocks could collect information from the frames, modify them and/or affect their forwarding decision. It is also possible to forward the data path outside of FES, to allow adding of customer's own Inbound and Outbound processing blocks.

4.1 Inbound Processing

Inbound processing receives frames from Ethernet and transfers them to the buffer memory (see Figure 12). The functionality of inbound processing blocks are described in sub-paragraphs in this chapter.

During reception, Inbound processing does:

- Detect frame errors
- Check frame validity (MACsec)
- Timestamp frames
- Filter and recognize frames

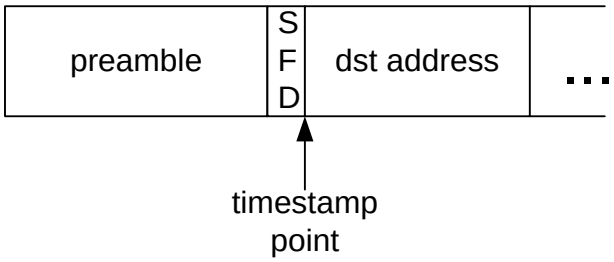
- Determine the destination port(s) for every frame
- Perform MAC address learning
- Modify frames

4.1.1 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. Frames with invalid CRC are discarded by the forwarding core and MAC RX error counter is incremented.

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 13.

Figure 13. 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 incrementing MAC RX Error Counter.

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 13). Also frames whose size is less than 64 bytes are discarded.

4.1.2 Timestamp

The Timestamp block uses the start-of-frame indication from RX MII block 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).

The Timestamp block is external to FES. FES communicates with the Timestamp block using Time interface. Time interface uses time presentation that has accuracy of 2^-16 nanoseconds and timestamp length of 96 bits. The actual accuracy depends on the system clock frequency used, and the type of clock. Time interface is presented in Chapter 6.4.

Timestamp block is external to FES because it may be hardware dependent. The implementation of the Timestamp block can depend on the FPGA family used, available FPGA resources (PLLs for example) and the timing related hardware external to FPGA; different kinds of boards have different kinds of clocks, with different accuracies and different frequencies, fixed and adjustable.

4.1.3 RX Pre-process

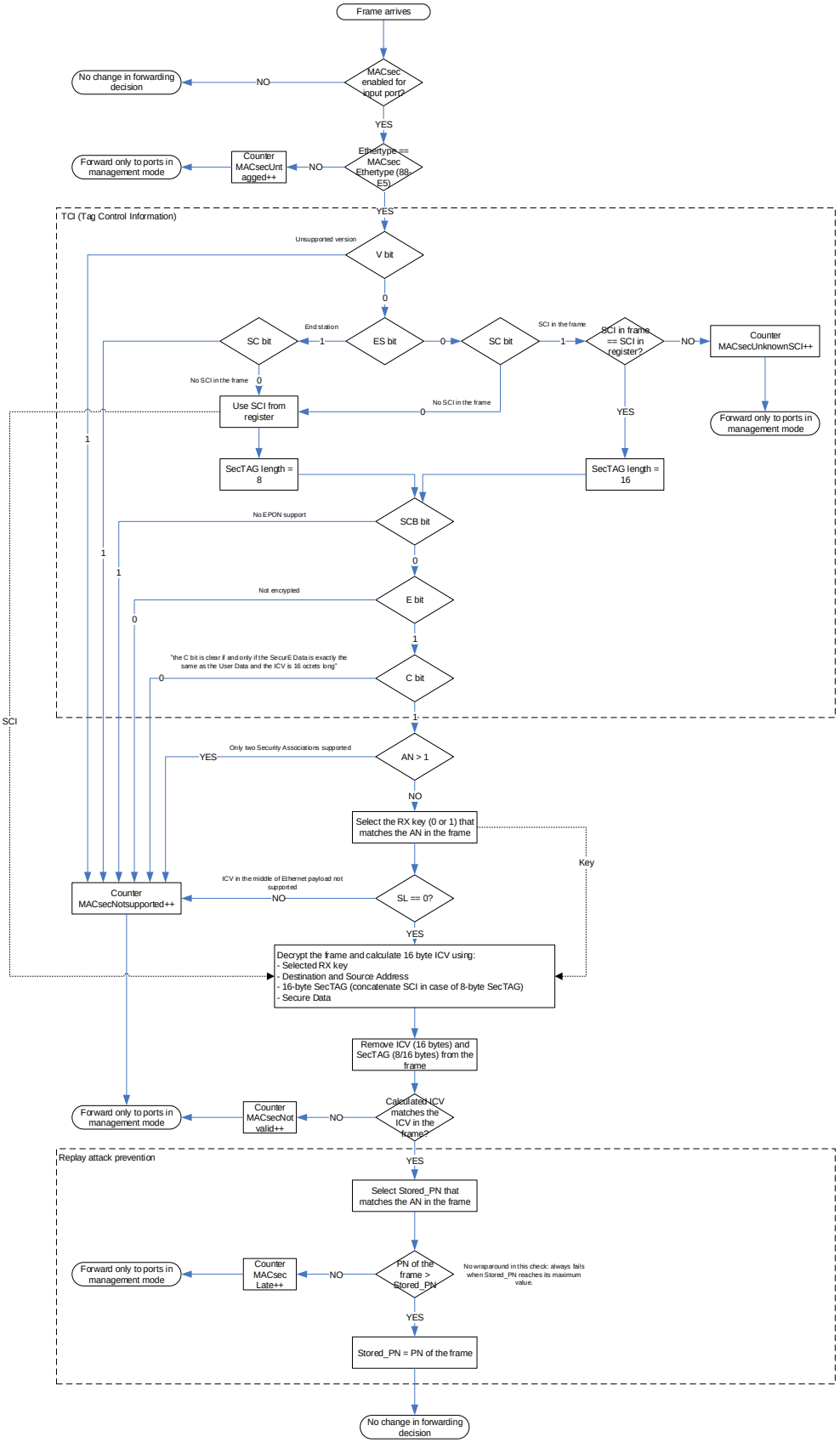
The RX pre-process block provides the received frames to the rest of the Inbound processing blocks. The Inbound processing blocks are chained in a row between the RX pre-process and the Forwarding core. The inbound processing blocks are connected to each other with an interface called EIF (Extended Interface). The RX pre-process is the source of the EIF bus and it generates EIF-cycles for each frame it receives from the RX MII.

4.1.4 MACsec

When MACsec is enabled for a port (generic MACSEC, see Table 16, and enable bit in MACsec Configuration Register, see Table 15) MACsec inbound processing block checks the validity of all the incoming frames, the MACsec header and ICV (Integrity Check Value) are removed and the payload is decrypted. If a frame passes the checks, it is forwarded normally. If a frame does not pass all of the checks, for example because of missing MACsec header, incompatibility, miscalculated ICV, non-increasing Packet Number (PN) or for some other reason, the frame is forwarded only to those ports that are in Management Mode. When Management Trailer is added to these frames, the MACsec bit in the Management Trailer is enabled to show that the frame requires special handling. As these frames are potentially dangerous, no MAC addresses are learned from these frames, nor are they considered as duplicate frames by the duplicate removal algorithm. For security reasons MAC addresses are not learned from non-MACsec frames either when MACsec is enabled for a port.

The functionality of MACsec inbound processing block is presented in Figure 14.

Figure 14. MACsec Inbound Processing Block



4.1.5 Forwarding Decision

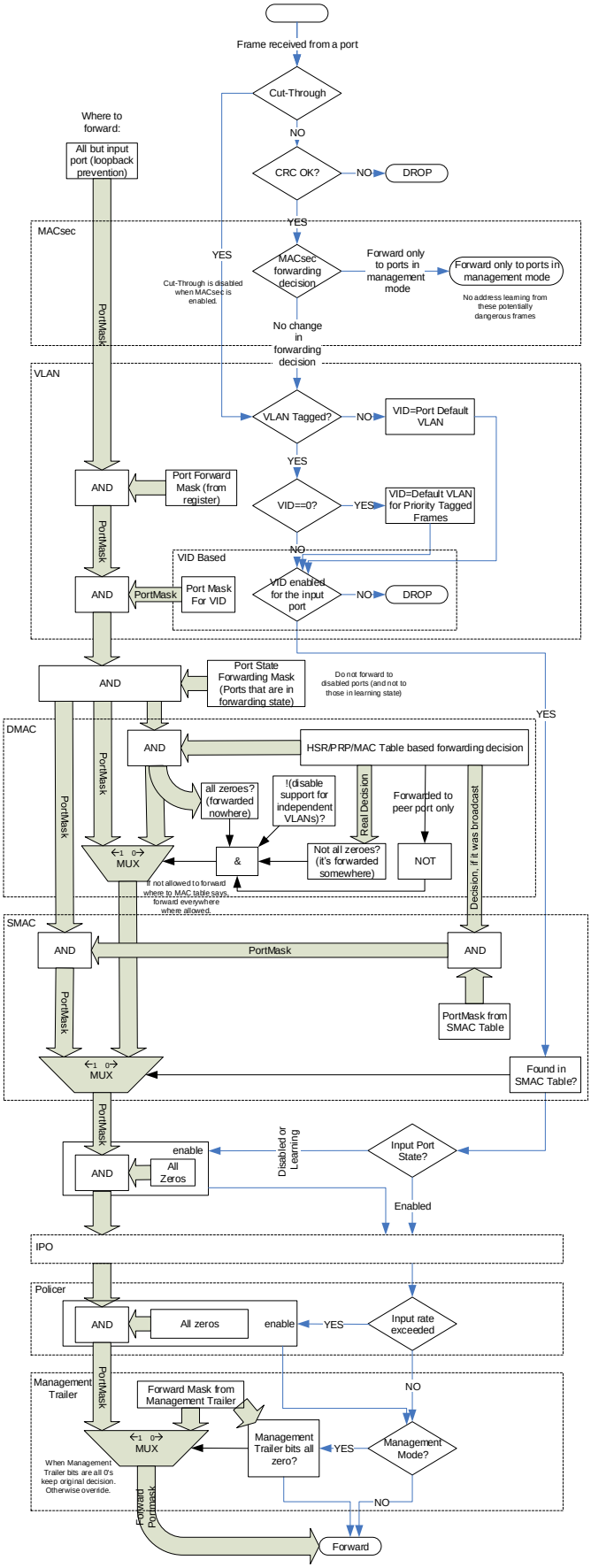
A Forwarding decision in FES is made based on information from the following sources:

- MAC address table (Chapter 4.1.11.5)
- Management Trailer (Chapter 4.1.10)
- Inbound Policy (Chapter 4.1.6)
- VLAN configuration and VLAN ID (Chapter 4.1.12)
- Port state (Port State Register Table 10)
- HSR tag (Chapter 4.4)
- PRP trailer (Chapter 4.5)
- Traffic Policer (Chapter 4.1.8)

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 15. 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 4.1.6).

Figure 15. Forwarding Decision



4.1.6 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. The alternatives for individual MAC addresses are the following:

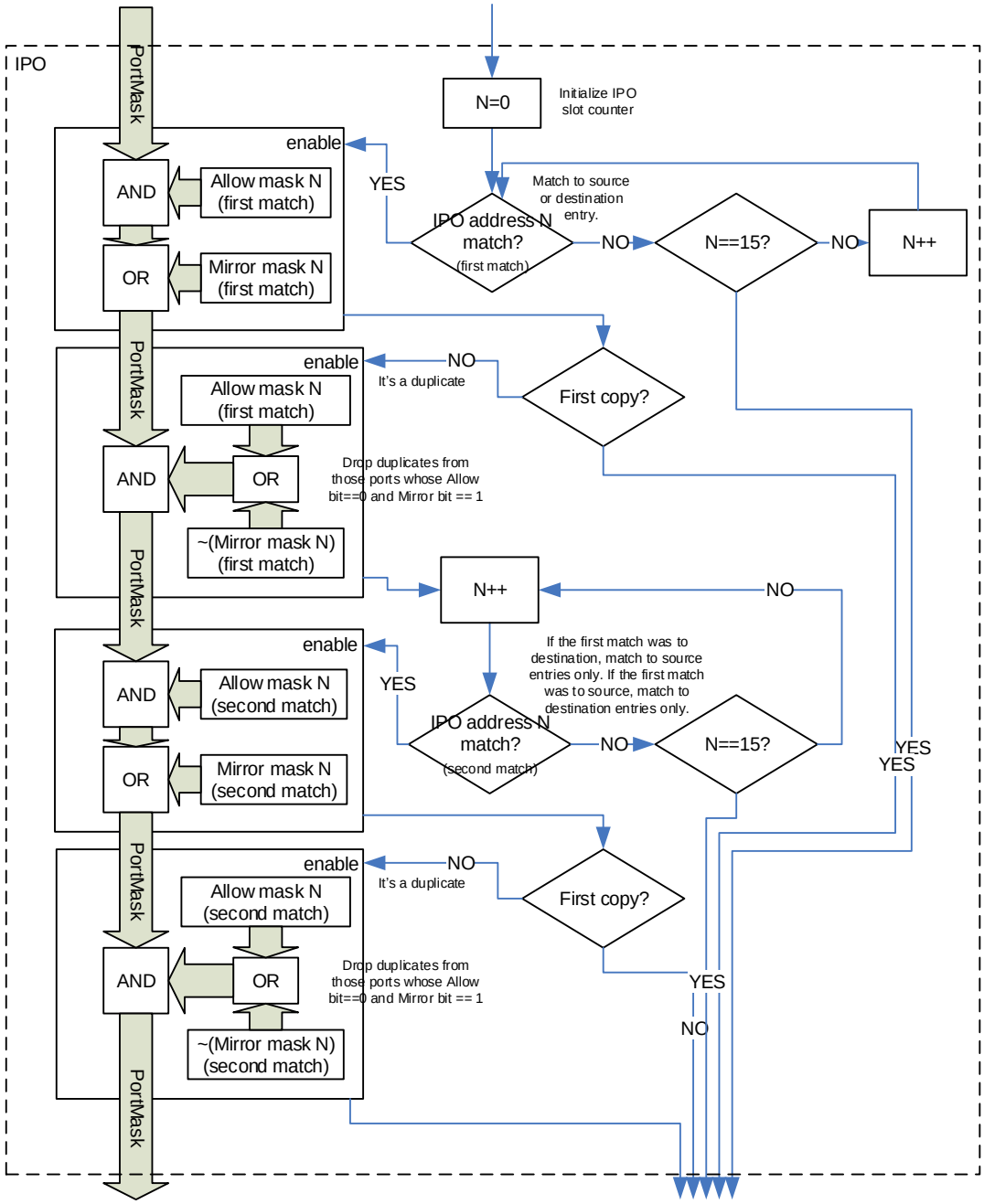
- Drop
- Allow forwarding only to certain ports
- Forced forwarding (mirroring) to certain ports
- Forward without adding HSR tag or PRP trailer

It is also possible to enable or disable:

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

See Figure 16 how Inbound Policy affects the forwarding decision.

Figure 16. Inbound Policy



The MAC addresses for the inbound policy are configured using Inbound Policy (IPO) registers presented in Table 14. 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 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 IPO matches to the forwarding decision can be seen in Figure 15. The other settings for the matching frame (priority, whether to send without HSR/PRP tag) are taken from the latest match (if any).

4.1.7 Priority-setting

When an incoming frame is VLAN tagged 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, Table 10). 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.

Inbound policy can override the priorities for the frames according to the priority setting in Inbound Policy Configuration Register (Table 14). 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 port default PCP of the input port.

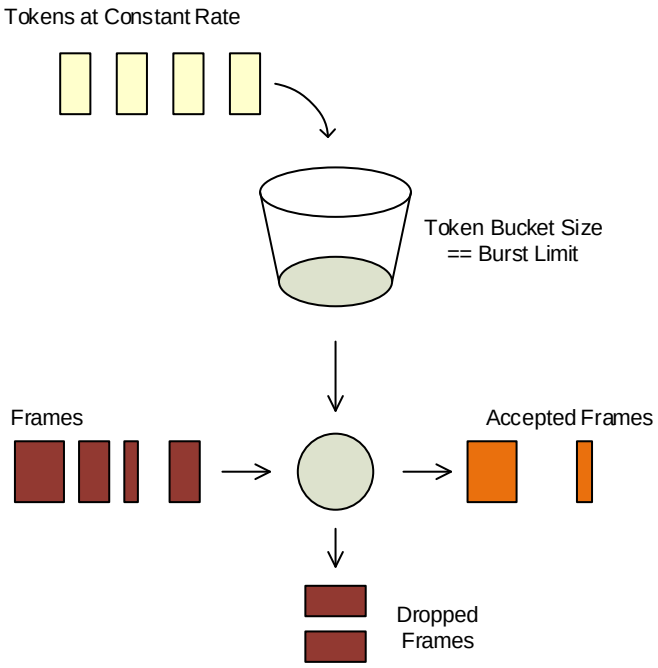
4.1.8 Traffic Policing

Traffic Policing provides means for limiting the rate of an incoming data stream or streams by dropping the frames that exceed the configured rate.

FES has a configurable amount of counters that can be used to measure rates of incoming streams. The amount of counters is configurable from 128 to 4096 per port, see generics in Table 16. Each port has its own set of counters that are independent from the counters of other ports. Using the inbound Policy (Chapter 4.1.6) or the Static MAC Address Table (Chapter 4.1.11.2) the user can configure which MAC addresses are mapped to which counter. If the traffic rate configured for a counter is exceeded the frame is dropped by forwarding it nowhere (see forwarding decision in Figure 15).

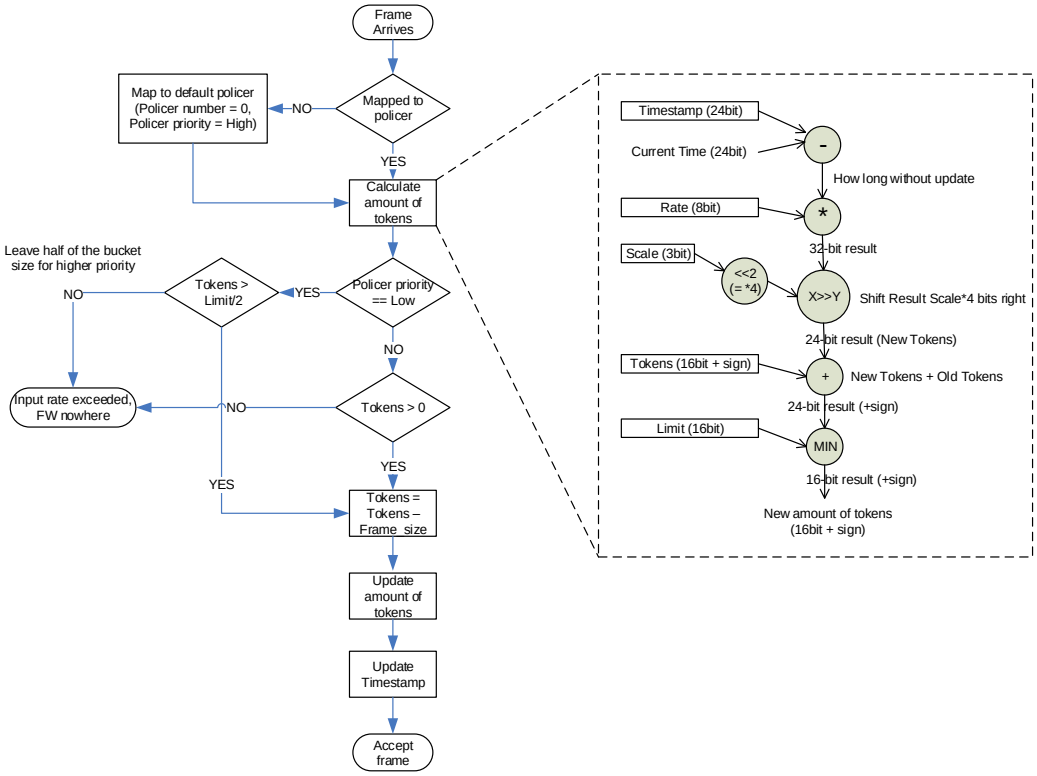
Each counter forms a policer that is known in the literature as Token Bucket [11]. Tokens arrive into the bucket (counter) at a constant rate and tokens are removed from the bucket for every frame that arrives. If a frame arrives and there is no tokens in the bucket, the frame is dropped. The rate at which tokens arrive into the bucket is configurable and it defines the maximum continuous rate the incoming stream is policed to. The size of the bucket defines the maximum length of a burst the policer allows to exceed the configured rate. Figure 17 illustrates the principle of a Token Bucket.

Figure 17. Token Bucket Principle



In FES the bucket is implemented as a 16 bit counter that counts tokens. The value in the counter is equivalent to number of bytes; for each accepted frame the counter value is decremented with the size of the frame. If the counter value is not positive, the frame is dropped. The functionality is presented in Figure 18. When calculating the length of a frame, preamble and SFD are not counted, CRC is counted (refer to Figure 2).

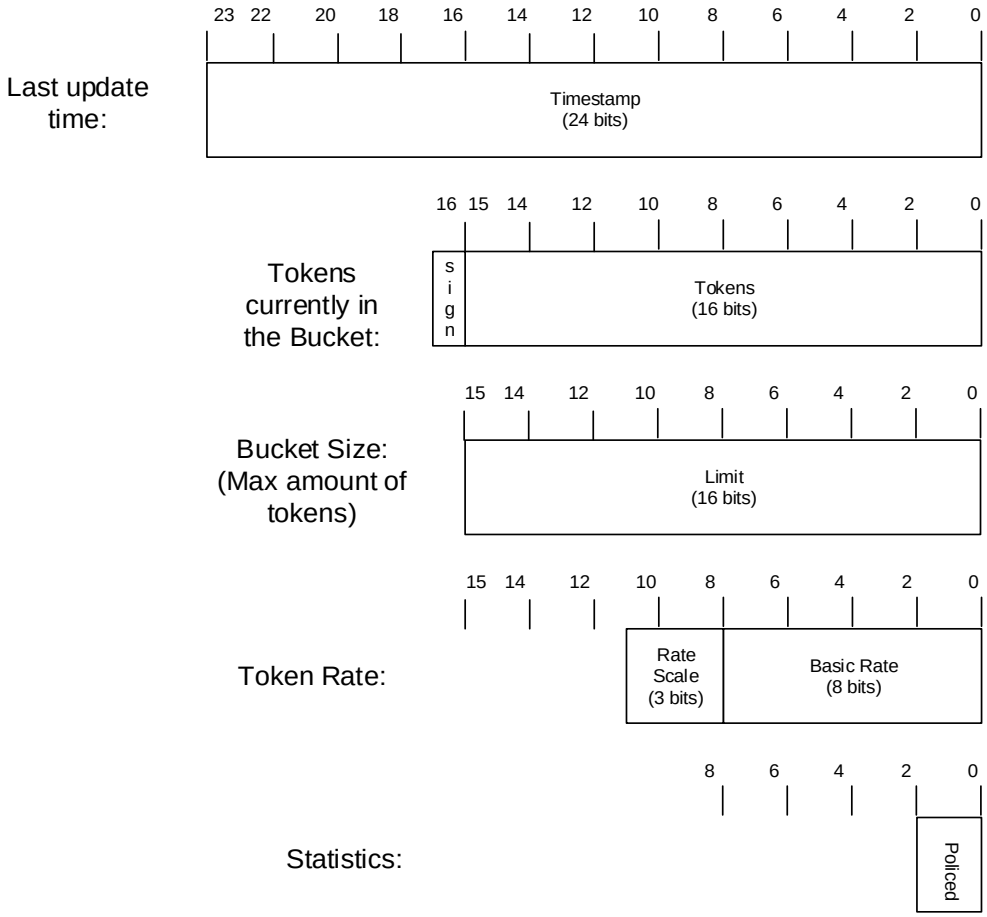
Figure 18. Token Bucket Operation for Frames



The amount of tokens in the buckets is updated only when the information is needed, not all the time. For this reason there is a Timestamp field in the bucket that indicates the last time the bucket was accessed. Based on the time of the bucket access and the timestamp of the last access, it can be calculated how many tokens has arrived to the bucket since the last access.

Figure 19 presents the data stored for each Token Bucket. The Bucket Size and Token Rate are separately configurable for each individual Bucket. See configuration registers (Table 10) on how to use indirect access to configure these values. The default values for Token rate and Bucket size are such that no frames are ever dropped by the policers.

Figure 19. Data Stored for each Token Bucket



The token rate (the policed rate) is configured with the parameters Basic Rate and Rate Scale. The resulting rate can be calculated using the formula:

Rate (Bytes/s) = Basic_Rate * clk / 16^Rate_Scale

, where

clk = system clock speed (typically 125 MHz)

Basic_Rate = configured value for Basic Rate in the register

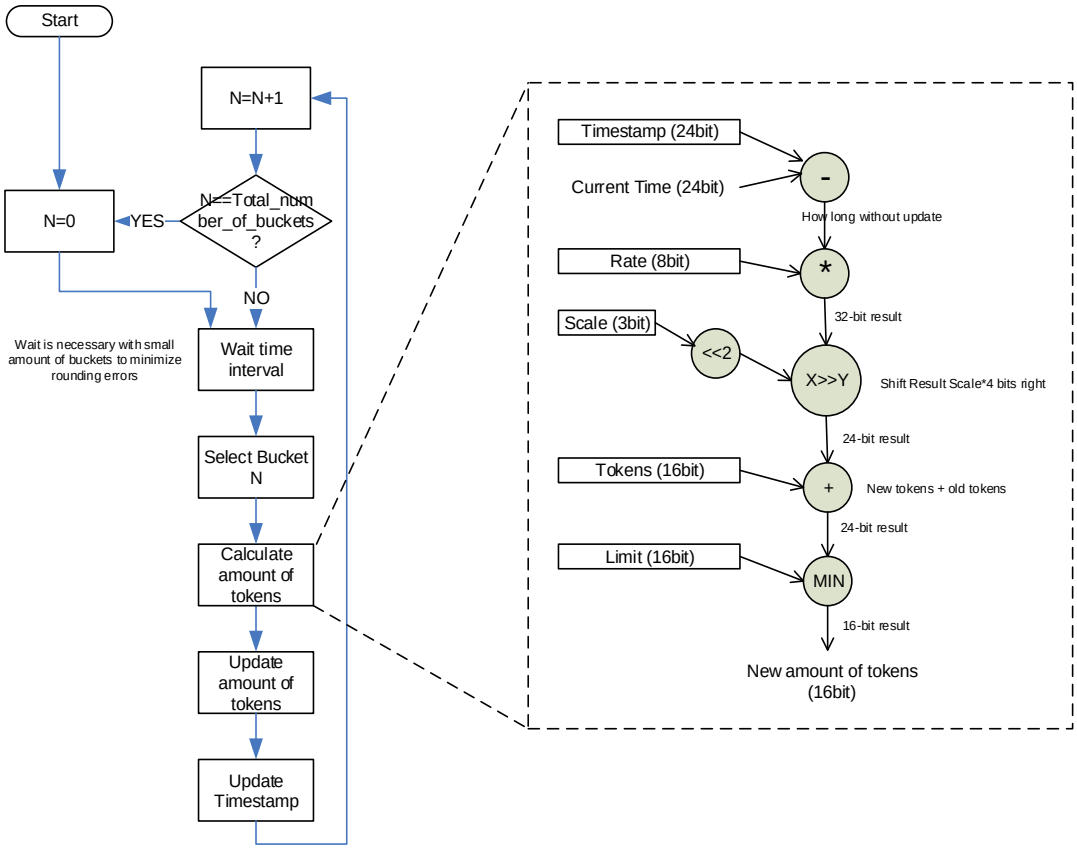
Rate_Scale = configured value for Rate Scale in the register

Rate_Scale values 0, 1 and 7 are reserved. Therefore the maximum rate that can be configured is with the typical 125 MHz clk clock speed:

Max policed rate = 255 * 125 000 000 / 16^2 = 124.5 MB/s = 0.996 Gbps, which is more than the maximum speed of gigabit Ethernet. (The maximum speed of gigabit Ethernet is less than gigabit because of the overhead caused by the interframe gap, preamble and start frame delimiter.)

To update the amount of tokens in the buckets when no frames arrive there is a slow background process that goes through the buckets updating the Token counters and Timestamps. This is needed because of the finite length of the Timestamp. The process is presented in Figure 20.

Figure 20. Bucket Update Background Process



As the Timestamp is 24 bits, it can present up to 16 777 215 cycles of local clock (clk). With typical 125 000 000 Hz local clock this means that the token count in a bucket has to be updated (Figure 20) every 0.134 seconds. If we want to keep rounding errors under 10% (the amount of tokens is rounded down), the token rate has to be at least 10 tokens in the update period, which corresponds to minimum possible policed rate of 10 Bytes in 0.134 seconds = 74.5 Bytes/second = 596 bps.

4.1.8.1 Default Policer

All the frames that are not mapped to any policer by Static MAC address table or IPO configuration are mapped to the default policer of the port. The default policer for all the ports is the Policer 0. If the user wants to drop all the frames that not mapped to another policer by IPO or by the Static MAC address table, they can set the rate of the default policer to zero. If the user does not want to drop any of these frames, the rate of the default policer can be set to maximum.

4.1.9 Precision Time Protocol

FES supports PTP message transportation directly over Ethernet (IEEE 1588-2008 [5] Annex F) and over User Datagram Protocol (UDP) over Internet Protocol version 4 (IEEE 1588-2008 [5] Annex D). PTP Mode setting in General Register (see Table 6) selects which one of the two modes is selected.

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 12.

4.1.9.1 End-to-end Transparent Clock Functionality

FES 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 [5] 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 FES 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.

4.1.9.2 Peer-to-Peer Transparent Clock Support

From FES 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. For this purpose there are registers (PTP_DELAY_SUBNS, PTP_DELAY_NS_LOW, PTP_DELAY_NS_HIGH) for the link delay (see Chapter 5.2.3). 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 FES is able to make the corrections automatically.

4.1.9.3 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, FES port 0 has the features built-in.

4.1.9.3.1 Time Stamp Recording

At inbound and outbound of port 0 timestamps of PTP event message (Sync, Delay_Req, Pdelay_Req and Pdelay_Resp) frames 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.

FES 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 FES is able to record more of them (Transfer bit in Transmit Timestamp Control register, see Table 7).

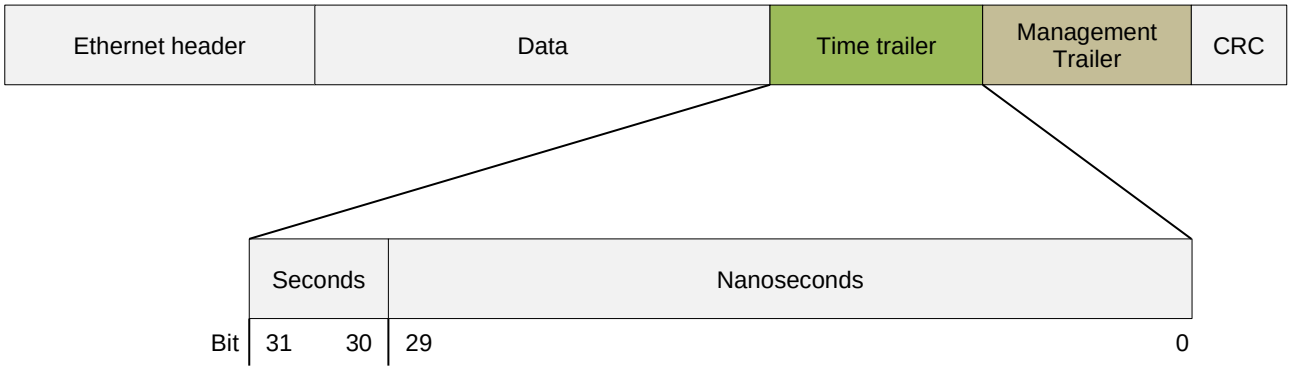
4.1.9.3.2 Frame Modification

At inbound of port 0 the PTP Sync messages are modified when PTP frame modification feature is enabled (Modify Sync Frames bit in General register, see Table 6). 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).

4.1.9.3.3 Time Trailer

At outbound of port 0 timestamps of PTP event message (Sync, Delay_Req, Pdelay_Req and Pdelay_Resp) frames can be added to the frames themself. When the feature is enabled (see General Register Table 6), a Time Trailer (see Figure 21) is added between the Ethernet frame payload data and Management Trailer (4.1.10). If Management Trailer is not enabled, the Time Trailer is added between the Ethernet frame payload data and the CRC. The 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). A 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



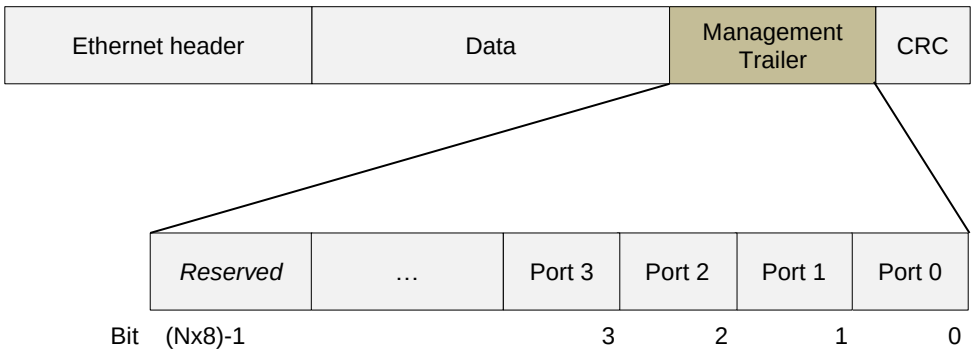
4.1.10 Management Trailer

The FES ports can be configured to a special management mode. In management mode the port supports the following feature:

- From the management port, it is possible to send 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. FES inserts a management trailer to every frame when it sends the frame out of the port and FES expects every frame received from the port to have a management trailer. The 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



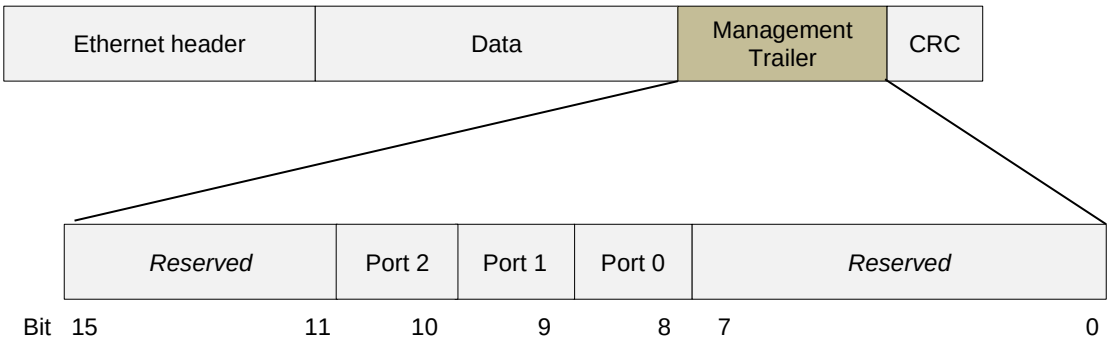
The length of the Management Trailer is one or two octets depending on the Management Trailer Length setting in GENERAL register (Table 6). For up to eight ports, the length of the management trailer can be 8 bits (one octet, in Figure 22 N=1) or 16 bits (two octets, in Figure 22 N=2). For nine to twelve port FES the length of the management trailer is 16 bits (two octets, in Figure 22N=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 FES and if FES forwards a frame from a management port to another management port FES does not alter them.

When the host CPU wants to send a frame from port number 0 to port number 1 (CPU attached to port0, port0 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 FES 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 FES 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 FES.

When FES sends a frame to host CPU and the CPU port is in management mode, FES 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 FES. When FES forwards frames 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 6) which allows different FES instances to use different bits (8 bit offset) in the trailer (see example in Figure 23). This feature makes it possible to send frames from a CPU to a certain port of an FES instance through another FES instance, which can be useful for example when two FES instances are used in a QuadBox design.

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



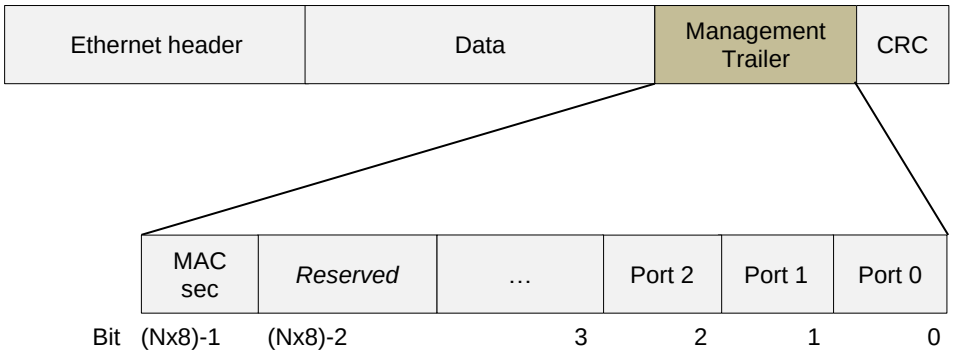
4.1.10.1 Management Trailer with MACsec

When MACsec is enabled (generic MACSEC, see generics in Table 16) the most significant bit of the Management Trailer becomes MACsec bit. This means that when MACsec functionality is enabled with the generic, the maximum supported port count with 8 bit Management trailer is 7.

When FES sends a frame out of a port that is in Management Mode, it sets the MACsec bit if the frame came in using a port that had MACsec enabled and the MACsec inbound block decided to forward the frame only to the ports that are in management mode (see Figure 14). This happens for example when there is something wrong with the frame, or if the frame is for a special purpose, for example for negotiation of a new key. In these cases the MACsec header is not removed so that the CPU that is connected to the Management port can see the contents of the whole frame.

If the frame came in using a port that did not have MACsec enabled, or if the frame passed all the checks of the MACsec inbound block (see Figure 14), the MACsec bit is always unset.

Figure 24. Management Trailer when MACsec enabled



When sending a frame to FES port that is in Management Mode the user (CPU) unsets the MACsec bit if they want the frame to be sent without MACsec header & ICV from MACsec enabled ports (see Figure 31).

4.1.11 MAC Address Table

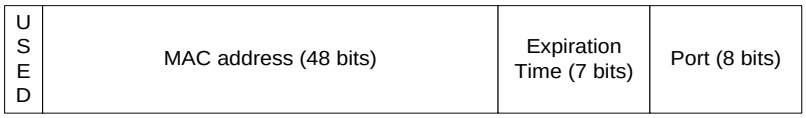
Forwarding decisions of FES are typically made by an address search to the MAC Address Table. Note that Inbound policy, Management trailer or VLAN settings may override the decision made based on the contents of the MAC Address Table.

The MAC Address Table of FES is divided into two parts, Static MAC Address Table and Dynamic MAC Address Table. The Static MAC Address Table contains only user configured MAC Addresses. Typically these are multicast addresses, but also unicast addresses can be configured in the static MAC Address Table. The Dynamic MAC Address Table contains only automatically learned MAC addresses that are therefore all unicast addresses. If the same destination address can be found in both Static and Dynamic MAC Address Table, the user configured information in the Static MAC Address Table is used.

4.1.11.1 Dynamic MAC Address Table Entry

The structure of a MAC Address Table entry in Dynamic MAC Address Table is depicted in Figure 25.

Figure 25. MAC Address Entry



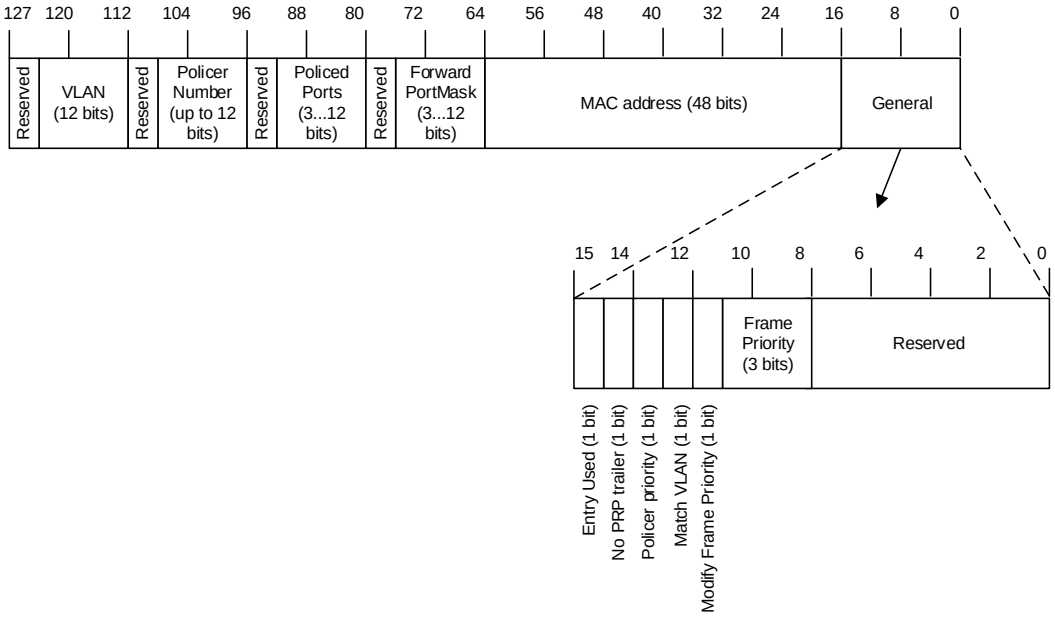
The MAC address entry contains a USED bit, the actual MAC Address, Expiration time and a Port number. USED bit set to '1' indicates that the entry is currently in use. The port where the frame was received from is stored into the Port field.

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 6). The Expiration time value is stored in multiples of 16 seconds.

4.1.11.2 Static MAC Address Table Entry

Static MAC Address Entry is presented in Figure 26. These entries are configured by the user by using the register interface. The static entries are not automatically aged.

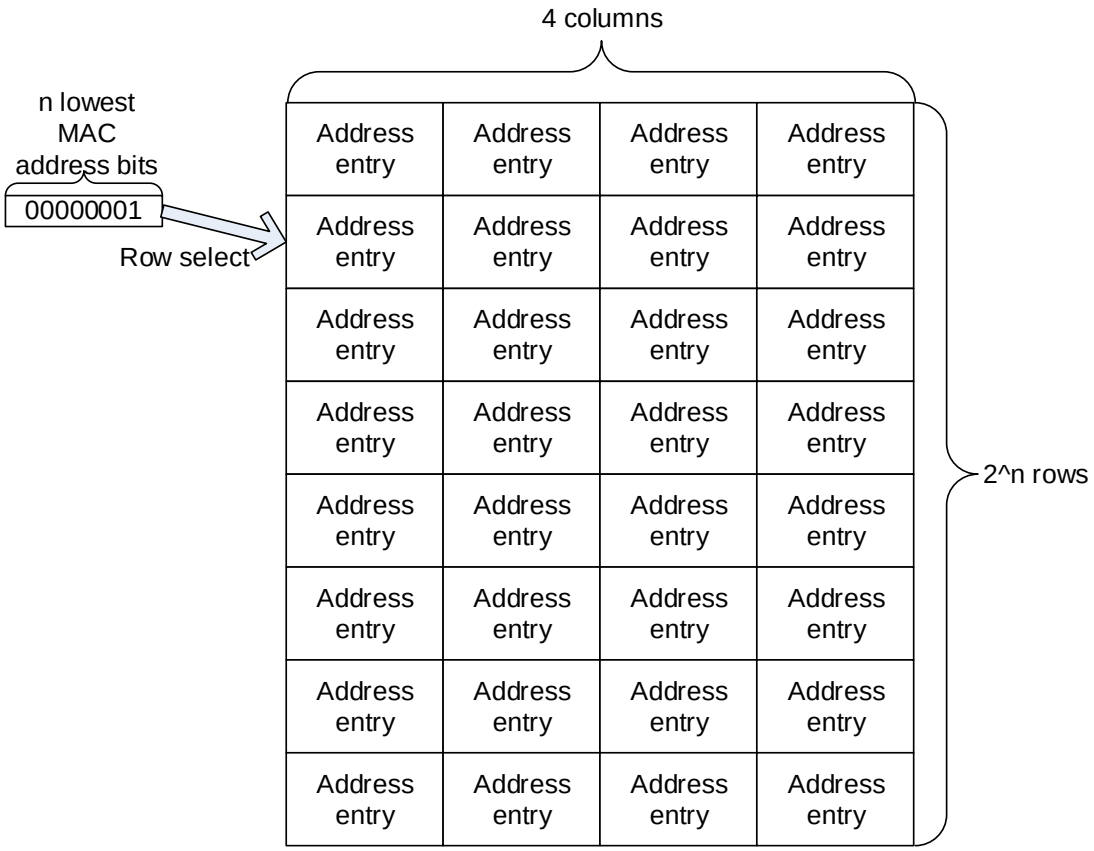
Figure 26. Static MAC Address Entry



When compared to entries of Dynamic MAC Address Table, an entry of the Static MAC address table includes more information. The most important is Forward PortMask instead of one single destination port. By using the Forward PortMask the user is able to configure more than one port where to frames with certain destination address are forwarded. It is also possible to map frames to certain inbound policers and put them to certain output queues (Frame Priority). The VLAN field allows defining a Static MAC Address entry for certain Virtual LAN only.

The organization of the Static MAC Address Table is presented in Figure 27. The lowest bits of the MAC address define the row in the table that can be used to store the MAC address. The MAC address can be stored in any Address entry (column) on that row.

Figure 27. Static MAC Address Table



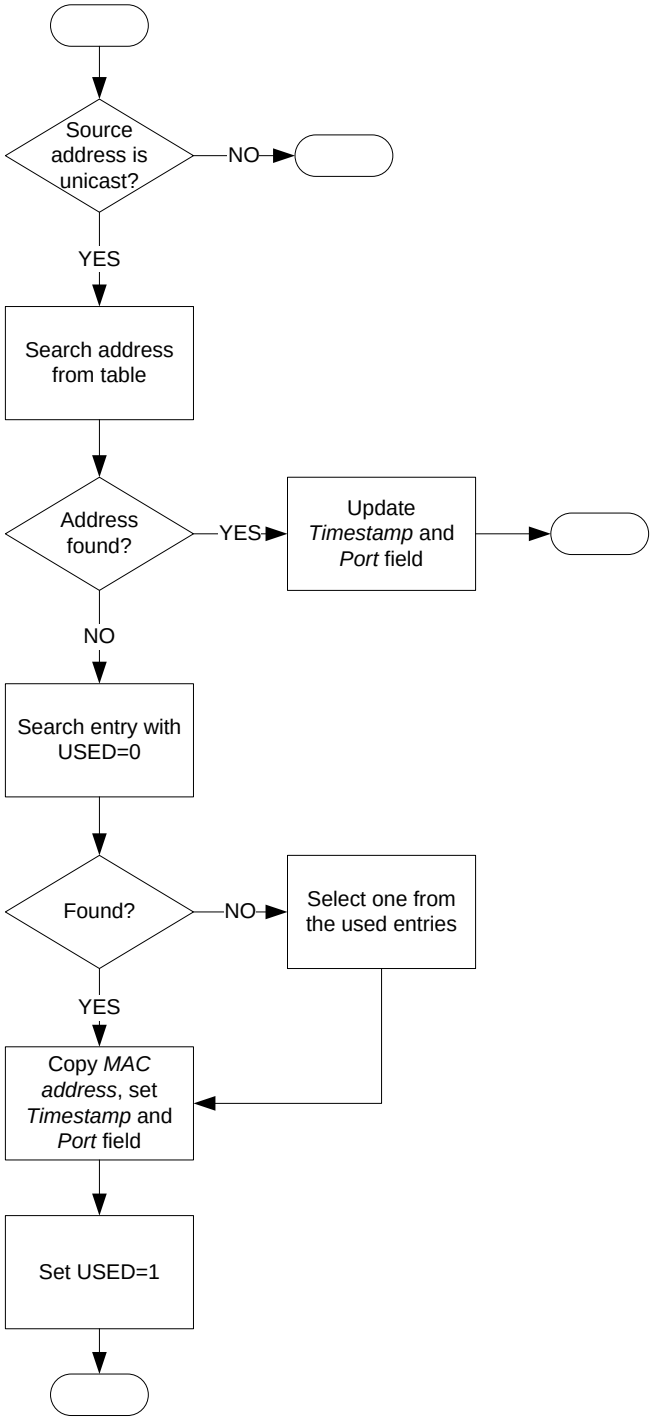
The number of rows (2^n) is configurable from 128 (2^7) to 4096 (2^12). See Generics in Table 16.

4.1.11.3 Address Learning

FES updates the Dynamic MAC address table automatically according to the source MAC address information in the received Ethernet frames. The Address learning process updates the Dynamic MAC Address Table when the receive port of the frame is in Forwarding or Learning state (See port state in Table 10). 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 28.

Figure 28. Address Learning



The Address learning process of FES is the following:

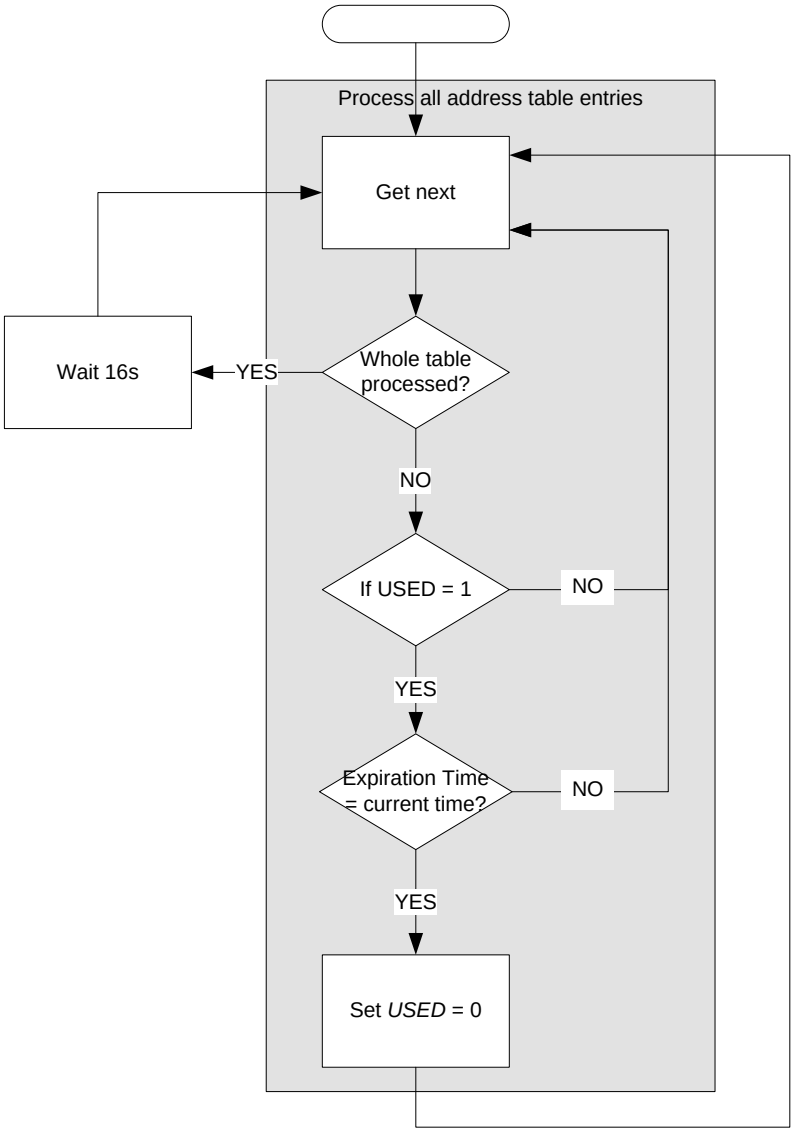
1. Check that the source address is a unicast Ethernet address.
2. Search for the address in the Dynamic MAC Address table.
3. If the address is found, update *Expiration time* and *port* fields with expiration time and source port and exit.
4. If the address is not found, search for an address entry 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'.

4.1.11.4 Address Aging

Address aging processes for the Dynamic MAC address table removes entries that are found to be expired. The Address aging process is depicted in Figure 29.

Figure 29. Address Aging



The Address Aging process goes through all the MAC Address Table entries every 16 seconds. All the expired entries are removed, in other words, their *USED* bit is set to '0'. Expiration is detected from the Expiration time field of the MAC Address entry: If the Expiration time is the same as the current time, the entry is to be removed. The address entry expiration time is updated by the Address Learning process, so only the entries that have not been refreshed for a certain time (Address lifetime) are removed by the Address Aging process.

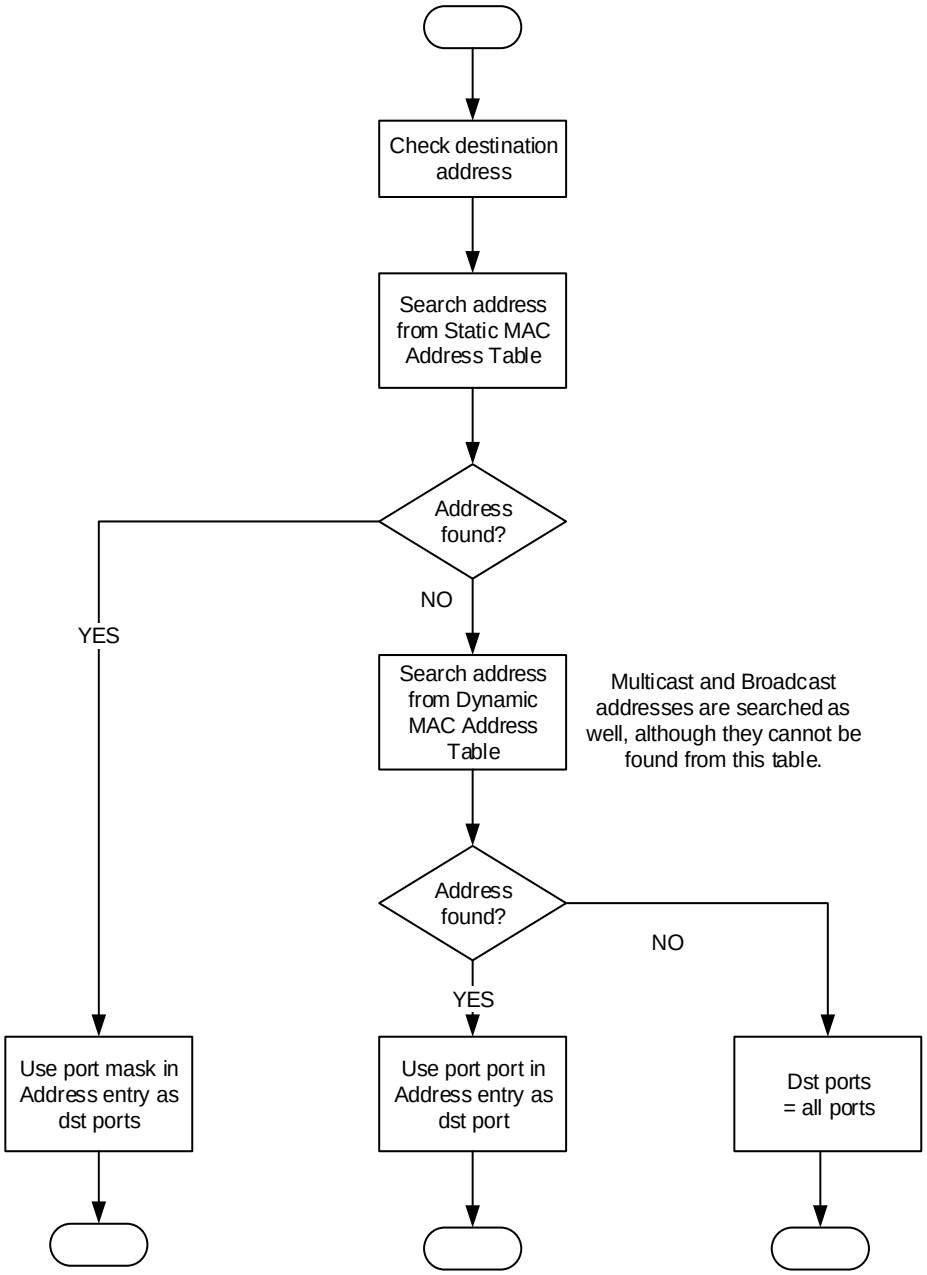
Entries in the Static MAC Address table are user configured and they are not automatically aged.

4.1.11.5 MAC Address and Forward Decision

The Address Search process searches MAC addresses from the two MAC Address tables. Static MAC Address Table and Dynamic 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 a MAC Address table, it means that in case of a unicast address 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 30. The whole forwarding decision process is presented in Figure 15.

Figure 30. MAC Address Search & Forwarding Decision



Note that in case of HSR and PRP also the port registered for the source MAC address affects the forwarding decision (see Figure 38 and Figure 39).

4.1.12 Virtual LANs (VLANs)

FES supports 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 10). See Figure 15 on how VLAN configuration affects the forwarding decision.

4.1.13 Forward Portmask

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

Forward portmask configuration is made via port configuration registers (see Table 10). With Forward portmask the user defines to which ports it is possible to forward frames from a port. Note that the forwarding can be configured to be different in different directions. 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 other ports to the CPU port only.

See Figure 15 on how Forward Portmask affects the forwarding decision.

4.2 Outbound Processing

Outbound processing block transfers frames from forwarding core (buffer memory) to Ethernet medium (refer to Figure 12). Every port has its own individual Outbound processing entity. During transmitting the Outbound processing does the following:

- Timestamp frames
- Modify frames
- Validate frames (MACsec)

4.2.1 TX Post-process

The Outbound processing blocks are chained in a row between the Forwarding core and the TX post-process. The Outbound processing blocks are connected to each other with a standard interface called EIF (Extended Interface). 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.

4.2.2 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 Timestamp block via Time interface (see Figure 12). The timestamp point of a frame is defined in Figure 13.

4.2.3 Timestamp

The Timestamp block 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.

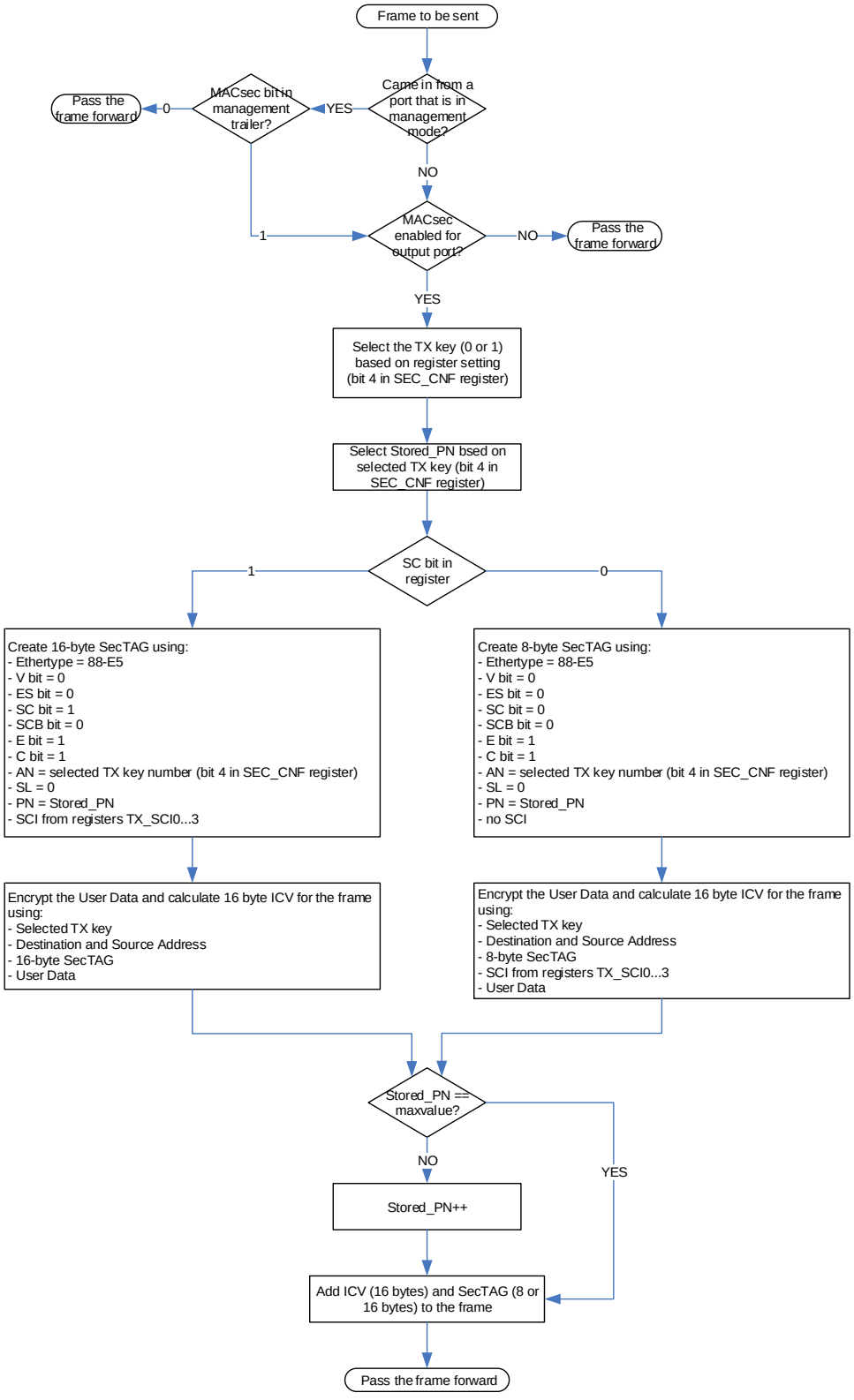
The Time-stamp block is external to FES. FES communicates with the Time-stamp block using Time interface. The timestamp block and Time interface at Outbound processing are similar to the Timestamp block and Time interface at Inbound processing. See more information at Chapter 4.1.2.

4.2.4 MACsec

The functionality of MACsec outbound processing block is presented in Figure 31.

When a frame is sent, the MACsec outbound processing block encrypts the frame and adds a MACsec header and ICV (Integrity Check Value) to the frame if MACsec is enabled for the output port. If the frame came in using a port that is in Management Mode it is also required that the MACsec bit was set. Otherwise the frame is sent unencrypted and without MACsec header and ICV.

Figure 31. MACsec Outbound Processing block



4.2.5 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 the 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 4.1.9.

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.

4.3 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.

4.3.1 Memory Controller

Memory controller block (see Figure 10) 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 4.3.5) 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.

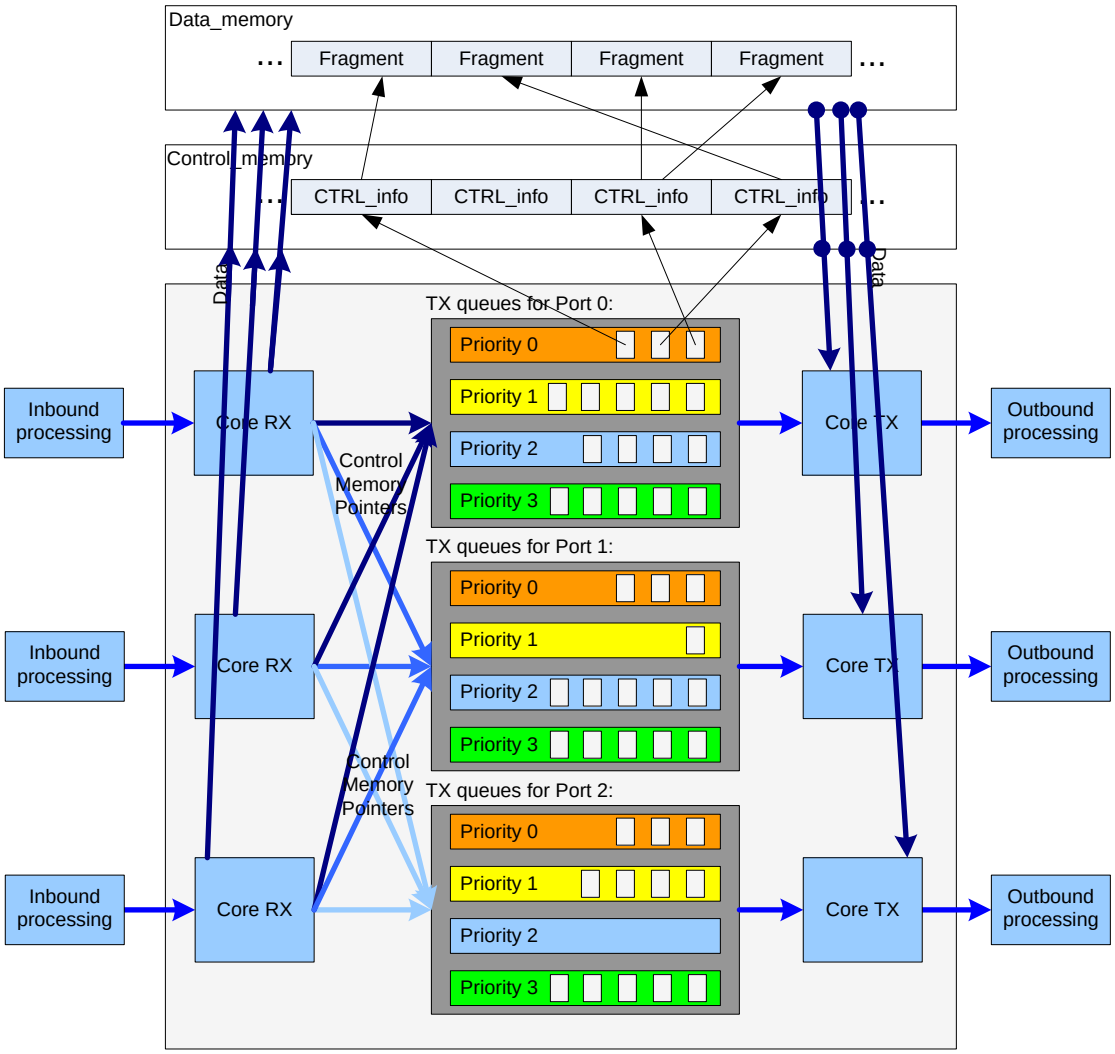
4.3.2 Priority Queues

FES has four or eight priority queues for every output interface depending on the generics settings (Chapter 6.1). 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 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 32). The data of the frame is not moved, when the pointer moves forward in the queue.

If traffic shaping (Chapter 4.3.2.1) is not active, 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. Traffic shaping may affect this behavior by not giving permission to transmit from certain queue(s) due to insufficient credit. If there is not enough credit to be able to transmit from certain queue, a frame from lower priority queue is transmitted instead, or if no such lower priority frame exists, the transmitter is kept idle.

All the priority queues have fixed length of 32 frames. If a frame arrives and the destination priority queue is full, the frame is dropped. In case the frame is to be forwarded to multiple ports, the frame is dropped only from the queue(s) that are full. If a frame is dropped because the destination priority queue is full, the corresponding error counter is incremented (see Table 10).

Figure 32. Priority Queues (only three ports shown)

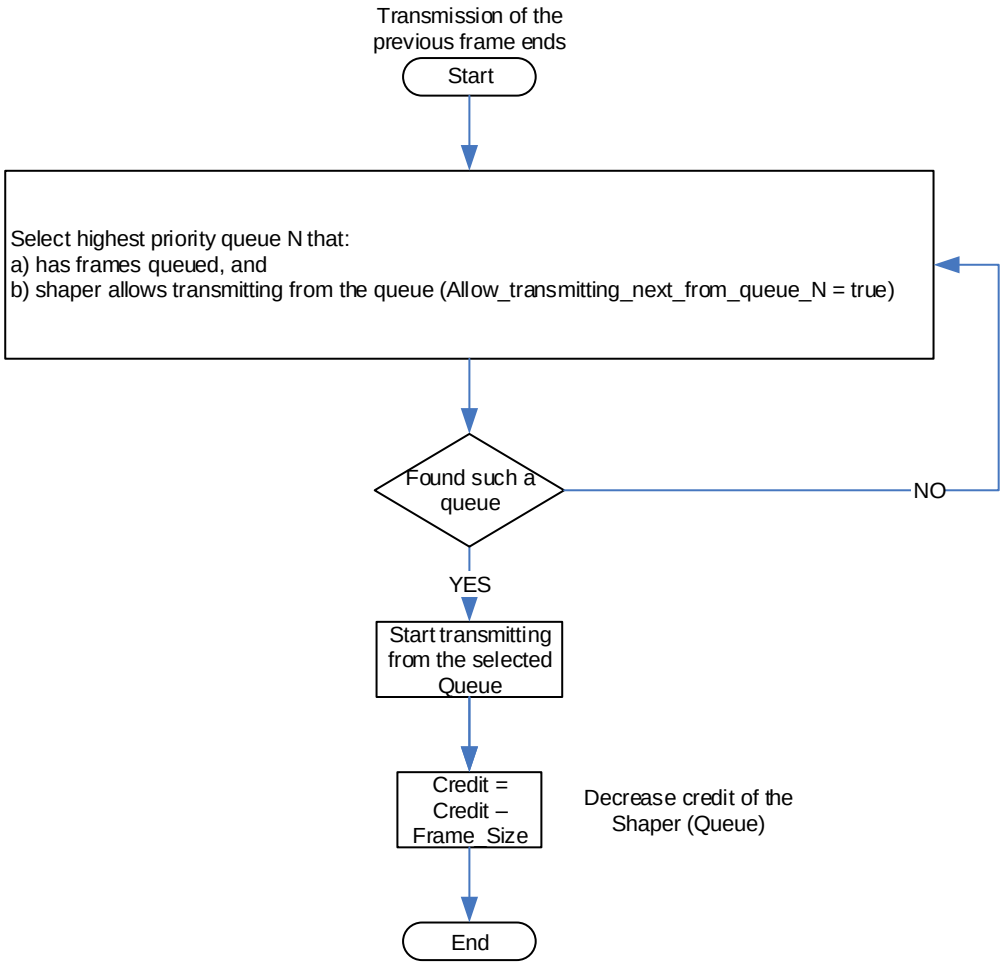


4.3.2.1 Traffic Shaping

Traffic Shaping means limiting of the transmit rate to meet certain preconfigured maximum allowed transmit rate; the frames that exceed the configured rate are delayed increasing the latency they experience. In FES shaping is done per Priority Queue basis which means that the rate at which frames can be transmitted from certain Priority Queue can be configured. If frames come into the queue faster than they can be sent out, the queue fill level increases. This means that when traffic shaping is active the frames have to wait longer in the queue than without shaping. On the other hand the fill levels of the lower priority queues decrease when higher priority queues are shaped.

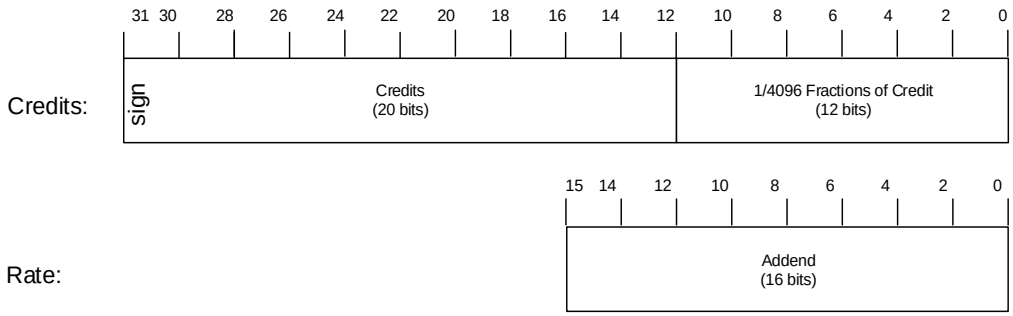
FES has a counter for each priority queue that can be used to control the transfer rate of the queue. Each counter forms a shaper whose functionality is equivalent to those known in the literature as Credit Based Shaper or Credit-based fair queuing. Credit (corresponds to tokens in Token Bucket) is added into the counter (corresponds to bucket in Token Bucket) at constant rate and credit is removed from the counter for every frame that is sent from the queue. A frame can be sent from the queue only when the credit is zero or above. If credit is negative, the frame (queue) has to wait for credit to increase. The process that selects the queue the next frame is transmitted from is presented in Figure 33.

Figure 33. Selection of Transmit Queue



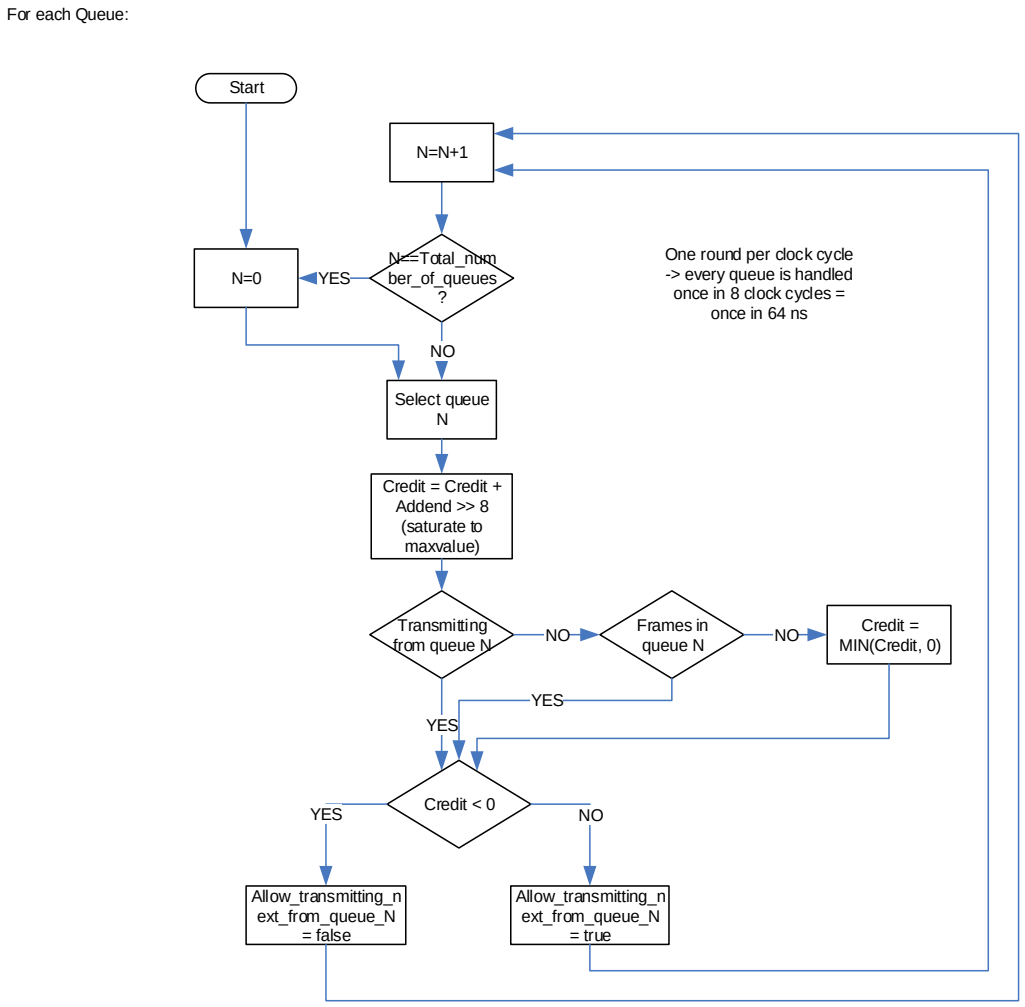
The rate at which credit increases defines the rate the output stream is shaped to. In a Credit Based Shaper the maximum amount of credit should be never reached, so the maximum value for credit is not configurable (unlike the bucket size in case of Token Bucket). The Data saved for every shaper (Queue) is presented in Figure 34. The Addend is the only configurable value for a shaper and it defines the rate at which credit increases which is also the rate the queue is shaped to.

Figure 34. Data Saved for a Shaper



The amount of credit saturates to 0 if the line is idle; if there are frames queued and they have to wait for another transmission to finish, the credit is allowed to increase over 0. The process is presented in Figure 35.

Figure 35. Increasing of Credit



The reset default is that the rate of all the shapers of all the ports and queues is above line speed; the user does not have to configure shapers if they are not used.

4.3.2.2 Configuring Shapers

The only user configurable parameter in a shaper is the rate to which the shaper shapes the traffic of the queue. It can be configured by setting the Addend value in the SHAPER0 register (see Table 10). The Addend value defines the speed by defining how many credits are added to the internal credit counter at every time interval of 8 clock cycles. The Addend value is 16 bits, 12 of which are fractions of credit, so the maximum value for Addend is 15,9998 and the minimum is 0. The resolution of the adjustment is 1/4096 (about 0.00024).

The configured rate can be calculated using the formula:

Rate = Addend / (8 * (1 / clk)) = Addend * clk /8

, where Addend is the value whiten in the shaper internal register and clk is the clock speed.

The value for Addend can be calculated from the desired data rate by using the formula:

Addend = (Rate * 8) / clk

, where Rate is the desired data rate in Bytes per second and clk is the clock speed. When calculating the rate, preamble and SFD are not counted to frame length, CRC is counted (refer to Figure 2).

With the typical 125 MHz clock the maximum configured rate is:

Rate (max) = 15.9998 Bytes * 125 000 000 Hz / 8 = 250 MB/s which is well above the maximum line speed.

The resolution of the rate adjustment is:

Rate (resolution) = 1/4096 Bytes * 125 000 000 Hz / 8 = 3.8 kB/s = 30.5 kb/s

4.3.3 Core RX

Core RX blocks terminate the Inbound processing paths (EIF buses, see Chapter 4.1.3). The Core RX blocks reserve a CTRL_info structures for the frames from the Control memory and handle the writing of the frame data to the Data memory. A CTLR_info structure contains the information on where the data fragments are located in the Data memory. The Core RX block places a pointer to the CTRL_info struct into correct priority queue of the correct output port(s) based on the information from Inbound processing. If the queue is full the frame is not forwarded to that port at all. Note that it is not allowed to place the frame into a priority queue with lower or higher priority, because the frames belonging to the same data stream in Ethernet have to preserve their order.

If a frame cannot be forwarded to any destination port because their TX queues are full, the frame is dropped. The frame is dropped also in case Control memory is full (out of CTRL_info structures) or Data memory is full. In case Inbound processing indicates that there was an error in the frame, Core RX block drops the frame and increments the corresponding error counter. When frames are dropped by Core RX, both the CTRL_info stucture and the reserved Data_memory are deallocated.

4.3.4 Core TX

Core TX blocks source the Outbound processing paths (EIF buses). A Core TX block gets information of which frame to transfer to the Outbound processing path from the TX queues of the corresponding port. The Core TX block then transfers the data from the Data_memory to the Outbound processing path at the rate the Outbound processing path is able to handle. This rate depends on the speed setting of the output interface (10/100/1000 Mbit/s).

After transferring the frame to the Outbound processing, it is checked if the frame has to be transferred to Outbound processing of other ports as well. If this was the last port to forward to, both the CTRL_info stucture and the reserved Data memory are freed.

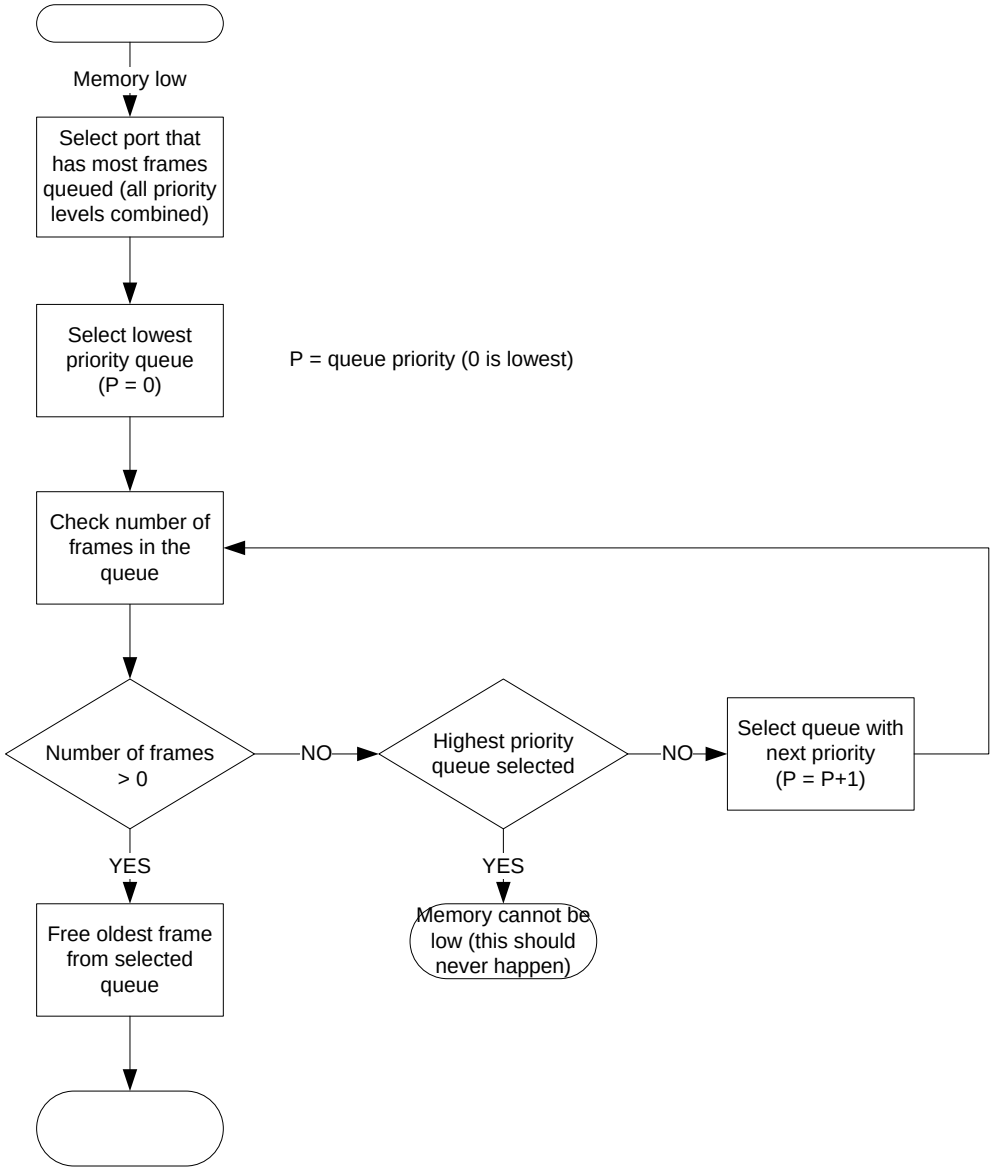
4.3.5 Frame Early Drop

Frame Early Drop (FED) algorithm deallocates buffer memory when FES 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 36. Frame Early Drop Algorithm



The Frame Early Drop algorithm is presented in Figure 36. 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.

4.4 HSR (High-availability Seamless Redundancy)

HSR specific features of FES include:

- Automatic insertion of HSR tag
- Automatic removal of HSR tag
- Automatic duplicate generation for HSR redundant ports
- Automatic duplicate detection and removal for HSR redundant ports

In inbound processing after making the forwarding decision for a frame the HSR tag (see Figure 5) is always removed if the port is in HSR mode. In outbound processing a HSR tag is added if the output port is in HSR mode.

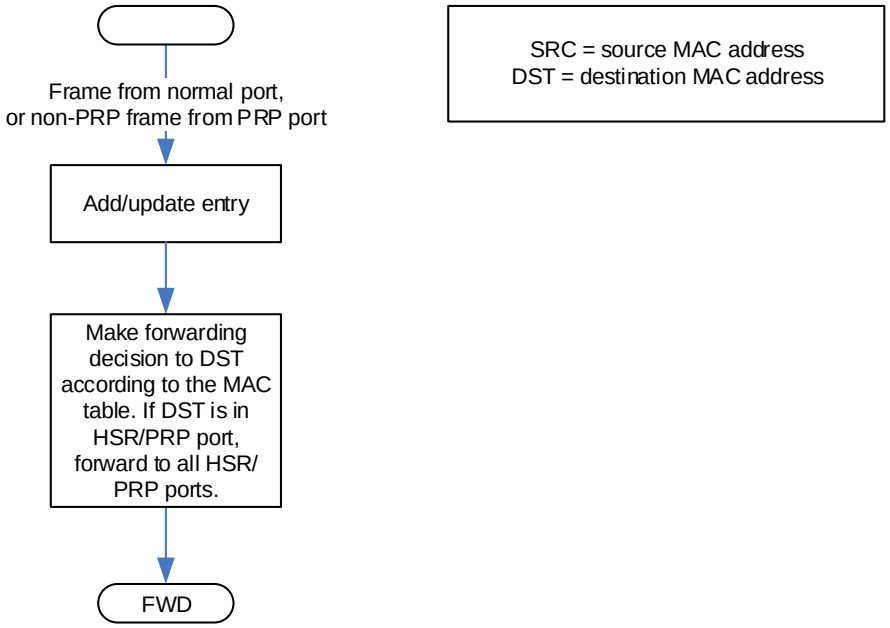
The HSR mode for a port and the other HSR specific setting are configured using HSR/PRP registers (see Table 11).

4.4.1 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 (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.

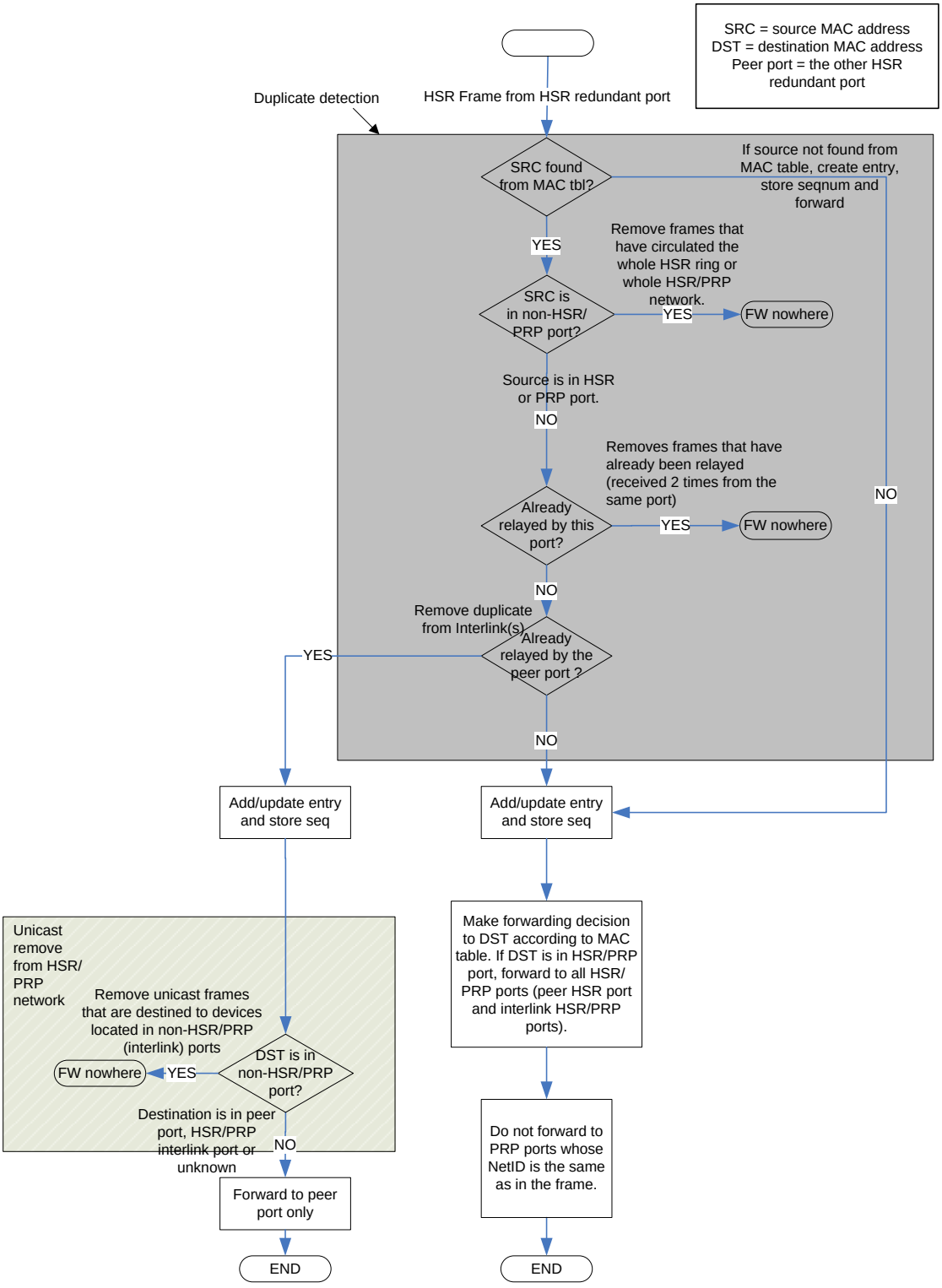
If a frame comes in from a normal (non-HSR, non-PRP) interlink port it is forwarded as presented in Chapter 4.1.5, but when it is to be forwarded to HSR-redundant port the frame is duplicated and sent out from the both redundant ports (see Figure 37).

Figure 37. 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 38.

Figure 38. 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-redundant (interlink) port. The duplicate detection is then done by looking at the MAC address entry's stored HSR sequence numbers for the other HSR redundant port: if it 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, coming 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) and the corresponding entry is updated.

Multicast and broadcast frames (see Figure 4) 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.

4.4.2 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 6) by disabling forwarding between HSR redundant ports.

The configuration procedure is the following:

1. Disable both redundant ports (Table 10, PORT_STATE register)
2. Change mode (Table 11, HSR_CFG register) for both redundant ports
3. Configure PORT_FWD_MASK (, PORT_FWD_MASK register)
4. Enable both configured ports (Table 10, PORT_STATE register)

4.4.2.1 HSR Mode X

Mode X is a new HSR mode introduced in HSR Standard [9] Edition 3 (Jan 8th, 2016). When operating in HSR Mode X duplicates are not sent out from ports that have already received the duplicate. This decreases the total amount of frames sent in the network. Introduction of Mode X can also be seen as a step towards mesh topology.

In a HSR ring Mode X cannot be used for HSR Supervision frames or IEEE 1588 PTP frames, and therefore in FES it is possible to force Mode H operation for selected frames with Inbound Policy (Chapter 4.1.6) when Mode X is enabled.

4.5 PRP (Parallel Redundancy Protocol)

PRP specific features of FES 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

In inbound processing after making the forwarding decision for a frame the PRP trailer is removed if the port is in PRP mode and the frame has one. In outbound processing 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).

The PRP mode for a port and the other PRP specific settings are configured using HSR/PRP registers (see Table 11). To enable PRP functionality, two ports must be configured as PRP redundant. Also PORT_FWD_MASK must be configured to prevent forwarding between PRP redundant ports.

The configuration procedure is the following:

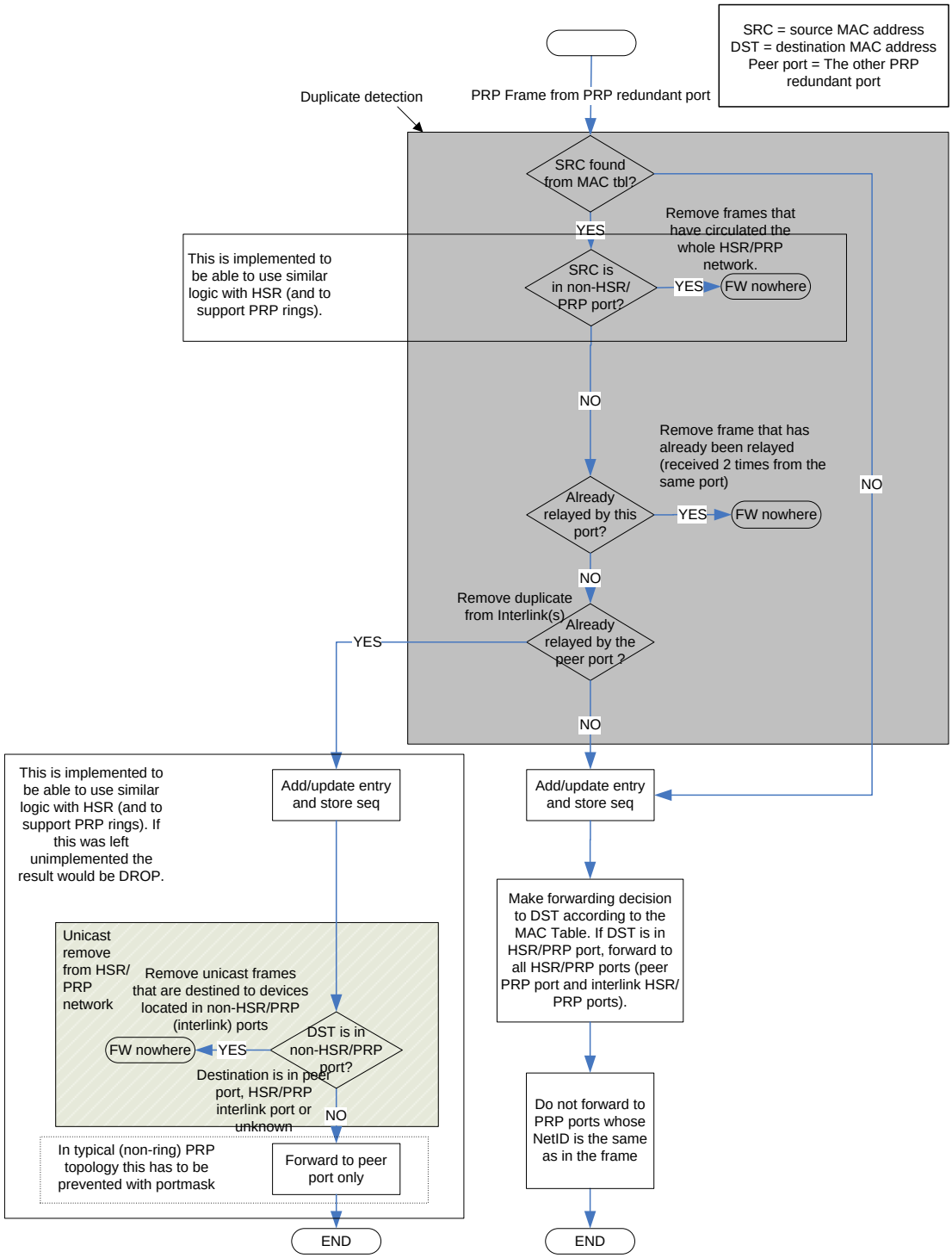
1. Disable both redundant ports (Table 10, PORT_STATE register)
2. Change mode (Table 11, HSR_CFG register) for both redundant ports
3. Configure PORT_FWD_MASK to prevent forwarding between PRP ports (Table 10, PORT_FWD_MASK register)
4. Enable both configured ports (Table 10, PORT_STATE register)

4.5.1 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 4.1.5, but if it is to be forwarded to a PRP redundant port, the frame is duplicated and sent out from 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 frames coming in from a PRP redundant port is presented in Figure 39.

Figure 39. 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 MAC address entry's stored PRP sequence numbers for the other PRP port: if it 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) and the corresponding entry is updated.

4.6 HSR/PRP interoperability

4.6.1 LanID, NetID and PathID

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 2. Table 3 presents how the identifier fields are handled in FES.

Table 2. 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 3. 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 or LanId configured for the output port, depending on LanId_modify bit (see Table 11).	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	-	-	-
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.	-

4.7 IP License Authentication

To prevent unauthorized use of the IP block, FES includes an Authentication Interface (see Authentication Interface signals in Figure 40). An external Security Chip is connected to the Interface. FES sends authentication requests approximately once in four seconds to the connected Security Chip and examines the replies it gets. Valid replies from the Security Chip indicate to FES that the user has a valid license to use the IP.

If FES does not get valid replies to its requests, for example when there is no Security Chip connected, FES will operate for approximately 2 hours and 15 minutes from reset, after which it stops forwarding frames. This behavior makes it possible for customers to evaluate FES IP core without purchasing Security Chips.

The register map (see Table 6, AUTH_STATUS register) includes an Authentication failure counter that counts failed and succeeded authentication requests. From this register the user can see whether the IP license authentication is working or not, which can be useful for example in product development phase and in production tester environment.

4.8 Reset

4.8.1 Software Reset

Software reset is made by writing value “1” to the Software reset bit (see Table 6) in General Register. After reset command FES 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, FES clears the Software reset bit in the register. After software reset FES is in the same state as after hardware reset.

4.8.2 Hardware Reset

Hardware reset is made by setting the global asynchronous reset input signal to its active state. All of the flip-flops of FES are immediately returned to their reset states and internal memories are initialized. The state after hardware reset is equivalent to power-up condition.

5 Configuration Registers

Functionality of FES is controlled through registers. The registers are accessible using the Avalon slave interface of FES.

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

FES register map consist of two sets of registers: FES Switch configuration registers (Chapter 5.1) and FES Port configuration registers (Chapter 5.2). The FES Switch configuration registers are used to configure the operation of the FES switch core. The FES Port configuration registers are used to configure the operation of the input/output ports of FES. There is a separate set of FES Port configuration registers for every Ethernet port (see Table 4). The location of the registers at Avalon bus can be selected by the user. An example configuration is presented in Table 3.

Table 4. FES Example Register Map

Address Offset	Register group description	Section	Table
0x0000 0000	FES Port Configuration Registers, port0 base	5.2	Table 9
0x0001 0000	FES Port Configuration Registers, port1 base	5.2	Table 9
0x0002 0000	FES Port Configuration Registers, port2 base	5.2	Table 9
0x0003 0000	FES Port Configuration Registers, port3 base	5.2	Table 9
0x0004 0000	FES Port Configuration Registers, port4 base	5.2	Table 9
0x0005 0000	FES Port Configuration Registers, port5 base	5.2	Table 9
0x0006 0000	FES Port Configuration Registers, port6 base	5.2	Table 9
…	…	…	…
0x0010 0000	FES Switch Configuration Registers base	5.1	Table 5

5.1 FES Switch Configuration Registers

Table 5 presents the FES switch configuration register groups. These registers configure the operation of the FES switch core.

Table 5. FES Switch Configuration Register Groups

Address Offset	Acronym	Register group description	Section	Table
0x0000	SWITCH	General switch configuration registers	5.1.1	Table 6
0x2000	TS	Frame Timestamp registers	5.1.2	Table 7
0x4000	VLAN	Virtual LAN Configuration Registers	5.1.3	Table 8

5.1.1 General Switch Configuration Registers

The General switch configuration registers are presented in Table 6.

Table 6. General Switch Configuration Registers

Address	Register	Description
SWITCH+ 0x0000	ID0	Reset: 0 x 00 00 IP core identification register 0. Bits 15-0: RO — Device ID bits 23 to 8. Device ID of FES (and FRS) is 0x000040.
SWITCH+ 0x0002	ID1	Reset: 0 x 40 00 IP core identification register 1. Bits 7-0: RO — Reserved Bits 15-8: RO — Device ID bits 7 to 0. Device ID of FES (and FRS) is 0x000040.
SWITCH+ 0x0004	FES_CFG_ID	Reset: 0 x XX XX FES Configuration ID Bits 15-0: RO — FES configuration ID. For Flexibilis internal use only.
SWITCH+ 0x0006	FES_SVN_ID0	Reset: 0 x XX XX FES SVN ID0 Bits 15-0: RO — FES configuration SVN ID. Revision number from Flexibilis version control system.
SWITCH+ 0x0008	FES_SVN_ID1	Reset: 0 x XX XX FES SVN ID1 Bits 15-0: RO — FES body SVN ID. Revision number from Flexibilis version control system.
SWITCH+ 0x000A … SWITCH+ 0x000E	Reserved	Reserved
SWITCH+ 0x0010	GENERAL	Reset: 0 x 00 00 General control bits for FES. Bit 0: R/W — General purpose output control Controls the general purpose output signal of FES. If the signal is connected to MDIO_to_Avalon Bridge [8], this bit behaves as “Disable MDIO polling” bit. In that case writing ‘1’ to this bit disables MDIO polling. When polling is disabled CPU may access registers of connected PHY devices. Bit 1: RO — General purpose input state Shows the status of the general purpose input signal of FES. If this signal is connected to MDIO_to_Avalon Bridge [8], this bit behaves as “MDIO polling state” bit: After CPU writes ‘1’ to Disable MDIO polling -bit, it should wait until this bit is ‘0’ before accessing the external PHY devices. ‘0’ = disabled ‘1’ = enabled 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 (can be used only with port counts 3…8) ‘1’ = 16 bits ‘2’ = reserved ‘3’ = reserved

Address	Register	Description			
		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 FES instances are connected to each other using interfaces that are in management mode, as it makes different FES instances to use different bits in the trailer. Use only with port counts 3…8 and when Management Trailer Length is configured to 16 bits. '2' = reserved '3' = reserved
		Bit	6:	RO	Reserved
		Bit	7:	R/W	Disable support for independent VLANs '0' = Support for independent VLANs enabled. If destination MAC is registered to a port where 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 forwarding is not allowed. See also forwarding decision in Figure 15.
		Bit	8	R/W	Policer configuration '0' = Policing based on IPO. No Static MAC Address Table based policing. Policer bits in Static MAC Address Table are reserved. '1' = Policing based on Static MAC Address Table. No policing based on IPO. Policer bits in IPO registers are reserved. This bit is reserved if either generic SMAC_TABLE_ROWS = '0' or generic POLICING is not '3'. See generics in Table 16.
		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 '01' = reserved '10' = PTP over Ethernet '11' = reserved
		Bit	13:	R/W	Cut-Through Enables Cut-Through operation between HSR redundant ports. Cut-through operation is possible only between ports defined by generic CUT_THROUGH (see Table 16). When enabling Cut-Through, exactly two ports must be configured to HSR mode, they must not be in management mode, and none of the other ports is allowed to be in PRP mode. '0' = Store-and-Forward operation '1' = Cut-Through operation

Address	Register	Description			
					All the other ports are always Store-and-Forward. To prevent broken frames from looping indefinitely, the 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. FES clears this bit when done.
		Bit	15:	R/SC	Software reset Writing '1' to this bit starts a software reset. FES clears the bit when reset is completed.
SWITCH+ 0x0012	MT_CLEAR_MASK	Reset: 2^(FES_PORT_HIGH+1)-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 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 is down. Clearing the entries may cause frames loss or duplicates.			
		Bits	15-0:	R/W	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 '1' = Entries registered to this port are cleared
SWITCH+ 0x0014	Reserved	Reserved			
SWITCH+ 0x0016	DMEM_FILL_LEVEL	Reset: 0 x XX XX Data Memory Fill Level			
		Bits	N-0:	RO	Number of available fragment pointers. N depends on the size of the data memory. Each port keeps two fragment pointers reserved all the time. This register is for Flexibilis internal use only.
		Bits	15-M:	RO	Reserved
SWITCH+ 0x0018	SEQ_MEM_FILL_LEVEL	Reset: 0 x 00 XX HSR Sequence Number Memory Fill Level			
		Bits	7-0:	RO	Number of available sequence number memory pointers. Each port keeps one pointer reserved all the time. This register is for Flexibilis internal use only.
		Bits	15-8:	RO	Reserved
SWITCH+ 0x001A … SWITCH+ 0x001E	Reserved	Reserved			
SWITCH+ 0x0020	ADDRESS_AGING	Reset: 0 x 00 12 Configuration register for address aging functionality of the MAC Address table			
		Bits	6-0:	R/W	Address LifeTime Lifetime of automatically learned addresses. 0=16s, 1=32s, 2=48s, … , 127=2048s. This setting defines also the ProxyNodeTableForgetTime for HSR/PRP.
		Bit	7:	RO	Reserved
		Bits	10-8:	R/W	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.
		Bits	15-11:	RO	Reserved

Address	Register	Description
SWITCH+ 0x0022	AGING_BASE_TIME_LO	Reset: 0 x A1 1F Aging Base Time Value low Bits 15-0: R/W — Aging base time value bits (15:0). Defines Aging Base Time in number of system clock cycles. The value written into this register should result 4 milliseconds. The default value gives 4 milliseconds with 125 MHz system clock frequency.
SWITCH+ 0x0024	AGING_BASE_TIME_HI	Reset: 0 x 00 07 Aging Base Time Value high Bits 7-0: R/W — Aging base time value bits (23:16). Defines Aging Base Time in number of system clock cycles. The value written into this register should result 4 milliseconds. The default value gives 4 milliseconds with 125 MHz system clock frequency. Bits 15-8: RO — Reserved
SWITCH+ 0x0026	AUTH_STATUS	Reset: 0 x 00 00 IP License Authentication Status Counter Bit 10-0: RO — Authentication failure counter The value of this counter is incremented by one when IP license authentication request fails; the value is decremented by one when authentication succeeds. Saturates to 2047 (0x7FF) and to 0. Authentication requests are made approximately once in four seconds. In case value 2047 is reached FES stops forwarding frames. Bits 15-11: RO — Reserved
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 FES 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. FES clears this bit after storing the information into 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 FES 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. FES clears this bit after storing the information into 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

Address	Register	Description
SWITCH+ 0x002C	INT_MASK	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. Bit 0: R/W — TX Timestamp. Indicates that information of a PTP event message has been stored into TX timestamp registers. Bit 1: R/W — RX Timestamp. Indicates that information of a PTP event message has been stored into RX timestamp registers. Bit 2: R/W — 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 13). RX_WRONGLAN does not cause an interrupt. Bit 3: R/W — 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. Bits 15-4: RO — Reserved
SWITCH+ 0x002E	INT_STATUS	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. Bit 0: R/C — TX Timestamp. Indicates that information of a PTP event message has been stored into TX timestamp registers. Bit 1: R/C — RX Timestamp. Indicates that information of a PTP event message has been stored into RX timestamp registers. Bit 2: R/C — 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 13) RX_WRONGLAN does not cause an interrupt. Bit 3: R/C — 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 Bits 15-4: RO — Reserved
SWITCH+ 0x0030 … SWITCH+ 0x01FE	Reserved	Reserved

Address	Register	Description
SWITCH+ 0x0200	MAC_TABLE0	Reset: 0 x 00 00 Dynamic MAC Table Read 0. 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 00 00 Dynamic 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. 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 00 00 Dynamic 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. 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 00 00 DynamicMAC 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. Bits 7-0: RO — 5th octet (XX:XX:XX:XX:XX:XX) Bits 15-8: RO — 6th octet (XX:XX:XX:XX:XX:XX)
SWITCH+ 0x0208 … SWITCH+ 0x021E	Reserved	Reserved
SWITCH+ 0x0220	SMAC_CMD	Reset: 0 x 00 00 Static MAC Table R/W Command register. Bits 11-0: R/W — Row. The number of the row of the Static MAC Table Entry that is read or written Bits 13-12: R/W — Column. The number of the column of the Static MAC Table Entry that is read or written Bit 14: R/W — Read/Write. 0 = Read values from Static MAC Table Entry to registers SMAC_TABLE0…SMAC_TABLE7. 1 = Write values from registers SMAC_TABLE0…SMAC_TABLE7 to Static MAC Table Entry. Bit 15: R/SC — Transfer. Write '1' to this bit to initiate read or write access to Static MAC Address Table. Value '0' indicates that the access is completed.
SWITCH+ 0x0222 … SWITCH+ 0x022E	Reserved	Reserved
SWITCH+ 0x0230	SMAC_TABLE0	Reset: 0 x 00 00 Static MAC Table Read/Write 0. Bits 7-0: RO — Reserved

Address	Register	Description			
		Bits	10-8:	R/W	Frame Priority New priority for frames with destination address match. Defines into which output queue the frames are put into.
		Bit	11:	R/W	Modify Priority 0 = Do not change frame priority 1 = Change frame priority to the value in Bits 10-8 of this register
		Bit	12:	R/W	Match VLAN 0 = Do not care about the VLAN of the frame. The MAC Table entry is for all the VLANs. 1 = Check that the value in VLAN field (register SMAC_TABLE7) matches to VLAN of the frame. The MAC Table entry is for one single VLAN.
		Bit	13:	R/W	Policer Priority 0 = Normal Priority 1 = Low Priority, frames with lower policer priority use only the upper half of the token bucket
		Bit	14:	R/W	No PRP Trailer 0 = Normal operation. 1 = Frames received from PRP network with matching source address are assumed not to have PRP trailer (PRP trailer is not removed, even if there seems to be one). Frames sent to PRP network with matching destination address are sent without PRP trailer.
		Bit	15:	R/W	Entry Used 0 = The entry is free (not used) 1 = The entry is in use
SWITCH+ 0x0232	SMAC_TABLE1	Reset: 0 x 00 00 Static MAC Table Read/Write 1. The first two bytes of the MAC address.			
		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)
SWITCH+ 0x0234	SMAC_TABLE2	Reset: 0 x 00 00 Static MAC Table Read/Write 2. The two bytes in the middle of the address.			
		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)
SWITCH+ 0x0236	SMAC_TABLE3	Reset: 0 x 00 00 Static MAC Table Read/Write 3. The last two bytes of the MAC address.			
		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)
SWITCH+ 0x0238	SMAC_TABLE4	Reset: 0 x 00 00 Static MAC Table Read/Write 4.			
		Bits	11-0:	R/W	Forward Port Mask Bit 0 corresponds to output port 0, bit 1 to port 1, bit 2 to port 2 and so on. 0 = Do not forward frames whose destination address matches to the port 1 = Forward frames whose destination address matches to the port
		Bits	15-12:	RO	reserved
SWITCH+ 0x023A	SMAC_TABLE5	Reset: 0 x 00 00 Static MAC Table Read/Write 5.			
		Bits	11-0:	R/W	Policed Ports Bit 0 corresponds to input port 0, bit 1 to port 1, bit 2 to port 2 and so on. 0 = Keep the default policer for this input port 1 = Change policer mapping (to Policer Number in register SMAC_TABLE6) for this input port

Address	Register	Description
		Bits 15-12: RO reserved
SWITCH+ 0x023C	SMAC_TABLE6	Reset: 0 x 00 00 Static MAC Table Read/Write 6. Bits 11-0: R/W Policer Number The number of the policer the matching frame is mapped to. The policer mapping is not changed if Policed Port bit = 0 for the input port (in register SMAC_TABLE5) Bits 15-12: RO reserved
SWITCH+ 0x023E	SMAC_TABLE7	Reset: 0 x 00 00 Static MAC Table Read 7. Bits 11-0: R/W VLAN The VLAN for which the entry is enabled. Valid only when bit 12 in register SMAC_TABLE0 is set. Bits 15-12: RO reserved

The exact formula for the minimum and maximum of the duplicate aging time is the following:

Duplicate aging time (min) = 2 * (AGING_BASE_TIME * Tclk) * (2 ^ HSR_ENTRY_FORGET_TIME);

Duplicate aging time (max) = 3 * (AGING_BASE_TIME * Tclk) * (2 ^ HSR_ENTRY_FORGET_TIME);

, where AGING_BASE_TIME is the Aging Base Time in registers AGING_BASE_TIME_LO and AGING_BASE_TIME_HI, HSR_ENTRY_FORGET_TIME is the value in HSR EntryForgetTime bits in ADDRESS_AGING register and Tclk is the period of the Main Clock (*clk*).

The exact formula for the minimum and maximum of the MAC entry aging time is the following:

MAC entry aging time (min) = 6 * (AGING_BASE_TIME * Tclk) * 600 * ADDRESS_LIFETIME;

MAC entry aging time (max) = 7 * (AGING_BASE_TIME * Tclk) * 600 * ADDRESS_LIFETIME;

, where AGING_BASE_TIME is the Aging Base Time in registers AGING_BASE_TIME_LO and AGING_BASE_TIME_HI, ADDRESS_LIFETIME is the value in Address Lifetime bits in ADDRESS_AGING register and Tclk is the period of the Main Clock (*clk*).

5.1.2 Frame Timestamp Registers

The Frame Timestamp registers are presented in Table 7. 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. FES 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 7) to 1 before FES is able to use the register set. After filling the information to the register set, FES clears the Transfer bit. If the Transfer bit corresponding to the register set that FES is going to use next is not set, the timestamp information of the next event message is lost (the frame itself is forwarded normally).

The timestamps of PTP messages are stored only for port 0.

Table 7. 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).

Address	Register	Description
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 (next lowest bits).
TS+ 0x0008… TS+ 0x000C	Reserved	Reserved
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 (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
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

Address	Register	Description
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
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

Address	Register	Description
TS+ 0x0384	RX _TS_3_S_LO	Receive TimeStamp 3 Seconds low. See TX_TS_1_S_LO
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

5.1.3 Virtual LAN Configuration Registers

Virtual LAN configuration registers are presented in Table 8.

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 in FES 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 8. Virtual LAN Configuration Registers

Address	Register	Description
VLAN+ 0x0000	VLAN0	Reset: 2^(FES_PORT_HIGH+1)-1 VLAN ID0 Mask 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^(FES_PORT_HIGH+1)-1 VLAN ID1 Mask 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^(FES_PORT_HIGH+1)-1 VLAN ID4095 Mask			
		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

5.2 FES Port Configuration Registers

Table 9 presents the FES port configuration register groups. FES port configuration registers are used to configure port-specific features of FES.

Table 9. FES Port Configuration Register Groups

Address Offset	Acronym	Register group description	Section	Table
0x0000	GEN	General configuration and state registers	5.2.1	Table 10
0x2000	HSR	HSR configuration registers	5.2.2	Table 11
0x4000	PTP	PTP configuration registers	5.2.3	Table 12
0x6000	CNT	Counter Registers	5.2.4	Table 13
0x8000	IPO	Inbound policy registers	5.2.5	Table 14
0xA000	SEC	MACsec configuration registers	5.2.6	Table 15

5.2.1 General Configuration and State Registers

General configuration and state registers are presented in Table 10.

Table 10. General Configuration and State Registers

Address	Register	Description			
GEN+ 0x0000	PORT_STATE	Reset: The reset value is defined by the generic PORT_STATE_DEFAULT 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. 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. 10 = Reserved 11 = Reserved
		Bits	5-4:	R/W	Port HW mode When Speed select = 00 (bits 8 and 9), the MII/GMII selection is automatic and these bits are not in use. 00 = MII 01 = Reserved 10 = GMII 11 = Reserved
		Bits	7-6:	RO	Reserved

Address	Register	Description			
		Bits	9-8:	R/W	Speed select 00 = selected by external signals (speed_sel) 01 = 1000 Mb/s 10 = 100 Mb/s 11 = 10 Mb/s
		Bits	11-10:	RO	Current speed Updated only when the speed is selected using external speed selection signals (speed_sel). 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. 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 FES. 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 FES. '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).
GEN+ 0x0012	VLAN0_MAPPING	Reset: 0 x 00 00 VLAN ID0 Mapping 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 from 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_LO	Reset: 0 x FA50 Port VLAN Priority Register, lowest bits Defines priorities for VLAN Priority Code Points (PCP) together with register PORT_VLAN_PRIO_HI. The priority for each PCP is configured with two or three bits, depending on how many queues there are per port. If there are 8 queues per port, the highest bit is from register PORT_VLAN_PRIO_HI and the two lowest bits are from this register. The lowest priority is 0, the highest is 3 or 7 depending on how many queues there are per port. The priority the frame gets defines into which output priority queue the frame is put into. Bits 1-0: R/W — Priority, PCP0, lowest two bits. Frames with Priority Code Point 0 in their VLAN tag get this priority inside FES. Bits 3-2: R/W — Priority, PCP1, lowest two bits. Frames with Priority Code Point 1 in their VLAN tag get this priority inside FES. Bits 5-4: R/W — Priority, PCP2, lowest two bits. Frames with Priority Code Point 2 in their VLAN tag get this priority inside FES. Bits 7-6: R/W — Priority, PCP3, lowest two bits. Frames with Priority Code Point 3 in their VLAN tag get this priority inside FES. Bits 9-8: R/W — Priority, PCP4, lowest two bits. Frames with Priority Code Point 4 in their VLAN tag get this priority inside FES. Bits 11-10: R/W — Priority, PCP5, lowest two bits. Frames with Priority Code Point 5 in their VLAN tag get this priority inside FES. Bits 13-12: R/W — Priority, PCP6, lowest two bits. Frames with Priority Code Point 6 in their VLAN tag get this priority inside FES. Bits 15-14: R/W — Priority, PCP7, lowest two bits. Frames with Priority Code Point 7 in their VLAN tag get this priority inside FES.

Address	Register	Description
GEN+ 0x0018	PORT_VLAN_PRIO_HI	Reset: 0 x 0000 Port VLAN Priority Register, highest bits. Defines priorities for VLAN Priority Code Points (PCP) together with register PORT_VLAN_PRIO_LO. If the number of queues per port (Generic QUEUES) is 8, the priority for each PCP is configured with three bits, two lowest bits are from register PORT_VLAN_PRIO_LO and the highest bit is from this register. If the number of queues per port is 4, the priority is presented solely with the bits in PORT_VLAN_PRIO_LO register and this register is reserved. The priority the frame gets defines into which output priority queue the frame is put into. Bit 0: R/W Priority, PCP0, highest bit Frames with Priority Code Point 0 in their VLAN tag get this priority inside FES. Bit 1: RO Reserved Bit 2: R/W Priority, PCP1, highest bit Frames with Priority Code Point 1 in their VLAN tag get this priority inside FES. Bit 3: RO Reserved Bit 4: R/W Priority, PCP2, highest bit Frames with Priority Code Point 2 in their VLAN tag get this priority inside FES. Bit 5: RO Reserved Bit 6: R/W Priority, PCP3, highest bit Frames with Priority Code Point 3 in their VLAN tag get this priority inside FES. Bit 7: RO Reserved Bit 8: R/W Priority, PCP4, highest bit Frames with Priority Code Point 4 in their VLAN tag get this priority inside FES. Bit 9: RO Reserved Bit 10: R/W Priority, PCP5, highest bit Frames with Priority Code Point 5 in their VLAN tag get this priority inside FES. Bit 11: RO Reserved Bit 12: R/W Priority, PCP6, highest bit Frames with Priority Code Point 6 in their VLAN tag get this priority inside FES. Bit 13: RO Reserved Bit 14: R/W Priority, PCP7, highest bit Frames with Priority Code Point 7 in their VLAN tag get this priority inside FES. Bit 15: RO Reserved
GEN+ 0x001A … GEN+ 0x001E	Reserved	Reserved
GEN+ 0x0020	POLICER_CMD	Reset: 0 x 00 00 Policer Read/Write command register. Bits N-0: R/W Policer Number The number of the policer whose registers are read or written when Transfer bit is set. N=(POLICERS-1), see Generics in Table 16. Bits 13-M: RO Reserved, (M= POLICERS, see Generics in Table 16) Bit 14: R/W Read/Write 0 = Read values from policer internal registers to registers POLICER0 and POLICER1. 1 = Write values from registers POLICER0 and POLICER1 to policer internal registers.

Address	Register	Description
		Bit 15: R/SC — Transfer Write '1' to this bit to enable read or write to policer internal registers. FES clears this bit when done.
GEN+ 0x0022 … GEN+ 0x0026	Reserved	Reserved
GEN+ 0x0028	POLICER0	Reset: 0 x 00 00 Policer Read/Write 0. Bits 15-0: R/W — Limit The maximum amount of Tokens in the Bucket
GEN+ 0x002A	POLICER1	Reset: 0 x 00 00 Policer Read/Write 1. Bits 7-0: R/W — Basic Rate Policed rate (Bytes/s) = Basic_Rate * clk / 16^Rate_Scale Bits 10-8: R/W — Rate Scale Values 0, 1 and 7 are reserved. Policed rate (Bytes/s) = Basic_Rate * clk / 16^Rate_Scale Bits 13-11: RO — Reserved Bits 15-14: RO — Policed '0' = Policer did not police (drop) frames since the last read access to policer registers '1' = Policer policed (dropped) only frames with low policer priority since the last read access to policer registers '2' = Policer policed (dropped) only frames with normal policer priority since the last read access to policer registers '3' = Policer policed (dropped) frames with both low policer priority and normal policer priority since the last read access to policer registers
GEN+ 0x002C … GEN+ 0x002E	Reserved	Reserved
GEN+ 0x0030	SHAPER0	Reset: 0 x 00 00 Shaper configuration for priority queue 0 (lowest priority). Bits 15-0: R/W — Addend The amount of credit added every time interval of 8 system clocks. The lowest 12 bits of this Addend present 1/4096 parts of a credit.
GEN+ 0x0032	SHAPER1	Reset: 0 x 00 00 Shaper configuration for priority queue 1. Bits 15-0: R/W — Addend The amount of credit added every time interval of 8 system clocks. The lowest 12 bits of this Addend present 1/4096 parts of a credit.
GEN+ 0x0034	SHAPER2	Reset: 0 x 00 00 Shaper configuration for priority queue 2.
GEN+ 0x0036	SHAPER3	Reset: 0 x 00 00 Shaper configuration for priority queue 3.
GEN+ 0x0038	SHAPER4	Reset: 0 x 00 00 Shaper configuration for priority queue 4.
GEN+ 0x003A	SHAPER5	Reset: 0 x 00 00 Shaper configuration for priority queue 5.
GEN+ 0x003C	SHAPER6	Reset: 0 x 00 00 Shaper configuration for priority queue 6.
GEN+	SHAPER7	Reset: 0 x 00 00

Address	Register	Description
0x003E		Shaper configuration for priority queue 7 (highest priority).

5.2.2 HSR/PRP Registers

HSR/PRP registers are presented in Table 11.

Table 11. HSR Registers

Address	Register	Description
HSR+ 0x0000	HSR_CFG	Reset: 0 x 00 00 HSR/PRP Configuration Register FES Generics (see Table 16) affect which features in this register are available and which ones are not. This register can be updated only when the port is disabled via PORT_STATE (Table 10) register. Configuration change procedure: - Disable both redundant ports (Table 10, 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	5-1:	RO	Reserved
Bit	6:	R/W	HSR Mode Selection ‘0’ = HSR Mode H ‘1’ = HSR Mode X This selection is valid only when the port mode is HSR redundant port (see bits 0, 8 and 9 of this register).
Bit	7:	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.
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 2 and Table 3). Note that these bits have also another meaning (LandId/NetId) described above.

		Bit	14:	R/W	LanId_modify If this port is in HSR mode, defines for frames that came in from non-HSR/PRP port whether LanId bit in the frame is set according to LanId of the input port or according to the LanId of the output port (see Table 2) 0 = LanId according to input port 1 = LanId according to output port
		Bits	15:	RO	Reserved

5.2.3 PTP Registers

PTP registers are presented in Table 12.

Table 12. PTP Registers

Address	Register	Description
PTP+ 0x0000	PTP_RX_SYNC_DELAY_SUBNS	Reset: 0 x 00 00 Contains the subnanoseconds part 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 (the time the frame spent inside FES) and TX delay. This compensation consist of delay from device input to FES input (includes for example PHY delay and delay caused by any interface adapters) and link delay for peer-to-peer transparent clock. For end-to-end transparent clock this register must be set to same value as PTP_RX_EVENT_DELAY_SUBNS. The value in this register is taken into use after writing to register PTP_RX_SYNC_DELAY_NS_LOW. Bits 15-0: R/W Sync delay, subnanoseconds Presented in 2^16 parts of a nanosecond
PTP+ 0x0002	PTP_RX_SYNC_DELAY_NS_LOW	Reset: 0 x 00 00 Contains the lowest 16 bits of the nanoseconds part 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 (the time the frame spent inside FES) and TX delay. This compensation consist of delay from device input to FES input (includes for example PHY delay and delay caused by any interface adapters) and link delay for peer-to-peer transparent clock. 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 values in registers PTP_RX_SYNC _DELAY_SUBNS and PTP_RX_SYNC _DELAY_NS_HIGH. Bits 15-0: R/W Sync delay, nanoseconds
PTP+ 0x0004	PTP_RX_SYNC_DELAY_NS_HIGH	Reset: 0 x 00 00 Contains the highest 8 bits of the nanoseconds part 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 (the time the frame spent inside FES) and TX delay. This compensation consist of delay from device input to FES input (includes for example PHY delay and delay caused by any interface adapters) and link delay for peer-to-peer transparent clock. For end-to-end transparent clock this register must be set to value 0x0. The value in this register is taken into use after writing to register PTP_RX_SYNC _DELAY_NS_LOW. Bits 7-0: R/W Sync delay, nanoseconds Bits 15-8: RO Reserved
PTP+ 0x0006	Reserved	Reserved

Address	Register	Description
PTP+ 0x0008	PTP_RX_EVENT_DELAY _SUBNS	Reset: 0 x 00 00 Contains the subnanoseconds part of the Receive delay compensation for all the PTP Event messages except Sync messages (who have an own compensation value). This value is added to the correction field of every received Event message (except for Sync messages) in addition to residence time (the time the frame spent inside FES) and TX delay. This compensation is the delay from device input to FES input (includes for example PHY delay and delay caused by any interface adapters). The value in this register is taken into use after writing to register PTP_ RX_EVENT DELAY_NS Bits 15-0: R/W Event delay, subnanoseconds Presented in 2^16 parts of a nanosecond
PTP+ 0x000A	PTP_RX_EVENT_DELAY _NS	Reset: 0 x 00 00 Contains the nanoseconds part of the Receive delay compensation for all the PTP Event messages except Sync messages (who have an own compensation value). This value is added to the correction field of every received Event message (except for Sync messages) in addition to residence time (the time the frame spent inside FES) and TX delay. This compensation is the delay from device input to FES input (includes for example PHY delay and delay caused by any interface adapters). Write access to this registers takes into use the value in register PTP_RX_EVENT _DELAY_SUBNS. Bits 14-0: R/W Event delay, nanoseconds Bit 15: RO Reserved
PTP+ 0x000C… PTP+ 0x000E	Reserved	Reserved
PTP+ 0x0010	PTP_TX_EVENT_DELAY _SUBNS	Reset: 0 x 00 00 Contains the subnanoseconds part of 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 (the time the frame spent inside FES) and RX delay. This compensation is the delay from FES output to device output (includes for example PHY delay and delay caused by any interface adapters). The value in this register is taken into use after writing to register PTP_ TX_EVENT DELAY_NS. Bits 15-0: R/W Event delay, subnanoseconds Presented in 2^16 parts of a nanosecond
PTP+ 0x0012	PTP_TX_EVENT_DELAY _NS	Reset: 0 x 00 00 Contains the nanoseconds part of 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 (the time the frame spent inside FES) and RX delay. This compensation is the delay from FES output to device output (includes for example PHY delay and delay caused by any interface adapters). Write access to this registers takes into use the value in register PTP_TX_EVENT _DELAY_SUBNS. Bits 14-0: R/W Event delay, nanoseconds Bits 15: RO Reserved

5.2.4 Counter Registers

Counter registers are presented in Table 13.

Table 13. Counter Registers

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. FES clears this bit when the counter registers can be read. 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). 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
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). 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) 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) 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.
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).

Address	Register	Description
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.
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.
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.

Address	Register	Description
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.
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.
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.

Address	Register	Description
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.
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.
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.

Address	Register	Description
CNT+ 0x0250 … CNT+ 0x0256	Reserved	Reserved
CNT+ 0x0258	RX_POLICED_L	Reset: 0 x 00 00 RX Policed 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 were policed because the input rate limit was exceeded. The counter counts also frames that would have been dropped anyway, and frames still forwarded because of management trailer.
CNT+ 0x025A	RX_POLICED_H	Reset: 0 x 00 00 RX Policed 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 were policed because the input rate limit was exceeded. The counter counts also frames that would have been dropped anyway, and frames still forwarded because of management trailer.
CNT+ 0x25C… CNT+ 0x25F	Reserved	Reserved
CNT+ 0x0260	RX_MACSEC_UNTAGGED_L	Reset: 0 x 00 00 RX MACsec Untagged 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 that were received while MACsec was enabled and that had no MACsec Ethertype (88-E5)
CNT+ 0x0262	RX_MACSEC_UNTAGGED_H	Reset: 0 x 00 00 RX MACsec Untagged 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 that were received while MACsec was enabled and that had no MACsec Ethertype (88-E5)
CNT+ 0x0264	RX_MACSEC_NOTSUPP_L	Reset: 0 x 00 00 RX MACsec Not Supported 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 that were received while MACsec was enabled and that had unsupported bit combination in the SecTAG.
CNT+ 0x0266	RX_MACSEC_NOTSUPP_H	Reset: 0 x 00 00 RX MACsec Not Supported 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 that were received while MACsec was enabled and that had unsupported bit combination in the SecTAG.

Address	Register	Description
CNT+ 0x0268	RX_MACSEC_UNKOWN SCI_L	Reset: 0 x 00 00 RX MACsec Unknown SCI 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 that were received while MACsec was enabled and that had SCI value that did not match to the value in registers RX_SCI0…3
CNT+ 0x026A	RX_MACSEC_UNKOWN SCI _H	Reset: 0 x 00 00 RX MACsec Unknown SCI 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 that were received while MACsec was enabled and that had SCI value that did not match to the value in registers RX_SCI0…3
CNT+ 0x026C	RX_MACSEC_NOTVALID _L	Reset: 0 x 00 00 RX MACsec Not Valid 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 that were received while MACsec was enabled and that failed the validity check
CNT+ 0x026E	RX_MACSEC_NOTVALID _H	Reset: 0 x 00 00 RX MACsec Not Valid 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 that were received while MACsec was enabled and that failed the validity check
CNT+ 0x0270	RX_MACSEC_LATE_L	Reset: 0 x 00 00 RX MACsec Late 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 that were received while MACsec was enabled and that arrived late thus failing the replay attack check
CNT+ 0x0272	RX_MACSEC_LATE _H	Reset: 0 x 00 00 RX MACsec Late 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 that were received while MACsec was enabled and that arrived late thus failing the replay attack check
CNT+ 0x274… 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.

Address	Register	Description
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.
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.
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.

Address	Register	Description
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.
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 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..

FES does not forward frames whose length is more than 1536 octets or under 64 octets (under 70 octets with HSR). Frames who come to PRP port but are too small to be PRP frames are not recognized as PRP frames and their PRP tag is not removed even if they seem to have one. 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 the VLAN tag.

5.2.5 Inbound Policy Registers

Inbound policy registers are presented in Table 14.

Table 14. 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. With value 0 every frame matches. Values over 48 are reserved. Bit 8: R/W — HSR Mode H HSR Mode H operation is forced in cases where output port is in HSR Mode X. In case of two IPO matches (source address match and destination address match) the information is taken from the second match Bit 9: R/W — Policer Priority '0' = Normal Priority '1' = Low Priority, frames with lower policer priority use only the upper half of the token bucket If Police bit = '0', this bit has to be '0' as well. In case of two IPO matches (source address match and destination address match) the information is taken from the first match. 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 Bits 13-12: R/W — New Priority, MSBs If Preserve Priority bit = '0' the priority of the frame is set to this value. Overrides priority defined by VLAN PCP. The lowest priority is 0, the highest is 3 or 7 depending on how many queues there are per port. If the number of queues per port (Generic QUEUES) is 4, the priority is presented with two bits (bit 12 and 13 of this register). If the number of queues per port is 8, the priority is presented with three bits (bits 12, 13 and 15 of this register), so that MSB is bit 13 and LSB is bit 15. In case of two IPO matches (source address match and destination address match) the information is taken from the second match

Address	Register	Description			
		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
		Bits	15:	R/W	New Priority, LSB If Preserve Priority bit = '0' the priority of the frame is set to the value in New Priority bits. Overrides priority defined by VLAN PCP. The lowest priority is 0, the highest is 3 or 7 depending on how many queues there are per port. In case of two IPO matches (source address match and destination address match) the information is taken from the second match This bit is reserved if the number of queues per port is 4. (Generic QUEUES, see Table 16)
IPO + 0x0002	ETH_ADDR0_FWD _ALLOW	Reset: 0 x 00 00 Ethernet address 0 allow forwarding. Defines into 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 into 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	ETH_ADDR0_POLICER	Reset 0 x 00 00 Ethernet address 0 policer configuration Defines into which policer matching frames are mapped to.			
		Bits	6-0:	R/W	Policer The number of the policer that is used to measure and police matching frames. In case of two IPO matches (source address match and destination address match) the Policer number is taken from the first match.
		Bits	15-7:	R/O	Reserved
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)

Address	Register	Description
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	ETH_ADDR1_POLICER	Ethernet address 1 policer configuration. See ETH_ADDR0_POLICER.
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.

5.2.6 MACsec Configuration Registers

MACsec configuration registers are presented in Table 15.

Table 15. MACsec Configuration Registers

Address	Register	Description			
SEC+ 0x0000	SEC_CNF	Reset: 0 x 00 00 MACsec Configuration Register			
		Bit	0:	R/W	MACsec enable ‘0’ = Disabled ‘1’ = Enabled When enabled, payload is encrypted and decrypted and MACsec SecTAG and ICV are added at outbound and removed at inbound. It is not allowed to enable MACsec and Management mode (PORT_STATE register, see Table 10) for a port at the same time.
		Bits	2-1:	RO	Reserved
		Bit	3:	R/W	SC bit ‘0’ = unset SC bit in transmitted frames and do not include SCI in the SecTAG ‘1’ = set SC bit in transmitted frames and include SCI in the SecTAG
		Bit	4:	R/W	TX key used ‘0’ = TX key 0 (registers TX_KEY0_0…15) ‘1’ = TX key 1 (registers TX_KEY1_0…15) When this setting is changed, it may take up to one second before the TX key changes, during which period it is not allowed to write to TX_KEY registers. It is also not allowed to write to the key registers of the TX key currently in use.
		Bits	7-5:	RO	Reserved

Address	Register	Description			
		Bit	8:	R/SC	Reset Packet Number (PN) counter for TX Security Association 0. The user should reset the PN counter before taking new TX key 0 into use.
		Bit	9:	R/SC	Reset Packet Number (PN) counter for TX Security Association 1. The user should reset the PN counter before taking new TX key 1 into use.
		Bits	11-10:	RO	Reserved
		Bit	12:	R/SC	Reset Packet Number (PN) counter for RX Security Association 0. The user should reset the PN counter after changing the RX key 0.
		Bit	13:	R/SC	Reset Packet Number (PN) counter for RX Security Association 1. The user should reset the PN counter after changing the RX key 1.
		Bits	15-14:	RO	Reserved
SEC+ 0x0002… SEC+ 0x00FE	Reserved	Reserved			
SEC+ 0x0100	TX_SCI0	Reset: 0 x 00 00 The first 16 bits of 64-bit Secure Channel Identifier, used for encryption and calculating the ICV for transmitted frames.			
		Bits	15-0:	R/W	TX SCI, bits 0…15
SEC+ 0x0102	TX_SCI1	Reset: 0 x 00 00 16 bits of 64-bit Secure Channel Identifier. Used for encryption and calculating the ICV for transmitted frames.			
		Bits	15-0:	R/W	TX SCI, bits 16…31
SEC+ 0x0104	TX_SCI2	Reset: 0 x 00 00 16 bits of 64-bit Secure Channel Identifier. Used for encryption and calculating the ICV for transmitted frames.			
		Bits	15-0:	R/W	TX SCI, bits 32…47
SEC+ 0x0106	TX_SCI3	Reset: 0 x 00 00 The last 16 bits of 64-bit Secure Channel Identifier. Used for encryption and calculating the ICV for transmitted frames.			
		Bits	15-0:	R/W	TX SCI, bits 48…63
SEC+ 0x0108… SEC+ 0x017E	Reserved	Reserved			
SEC+ 0x0180	RX_SCI0	Reset: 0 x 00 00 The first 16 bits of 64-bit Secure Channel Identifier, used for decryption and calculating the ICV for received frames. If received frame contains SCI, it is also checked against this value.			
		Bits	15-0:	R/W	RX SCI, bits 0…15
SEC+ 0x0182	RX_SCI1	Reset: 0 x 00 00 16 bits of 64-bit Secure Channel Identifier, used for decryption and calculating the ICV for received frames. If received frame contains SCI, it is also checked against this value.			
		Bits	15-0:	R/W	RX SCI, bits 16…31
SEC+ 0x0184	RX_SCI2	Reset: 0 x 00 00 16 bits of 64-bit Secure Channel Identifier, used for decryption and calculating the ICV for received frames. If received frame contains SCI, it is also checked against this value.			
		Bits	15-0:	R/W	RX SCI, bits 32…47
SEC+ 0x0186	RX_SCI3	Reset: 0 x 00 00 The last 16 bits of 64-bit Secure Channel Identifier, used for decryption and calculating the ICV for received frames. If received frame contains SCI, it is also checked against this value.			
		Bits	15-0:	R/W	RX SCI, bits 48…63
SEC+ 0x0188… SEC+ 0x01FE	Reserved	Reserved			

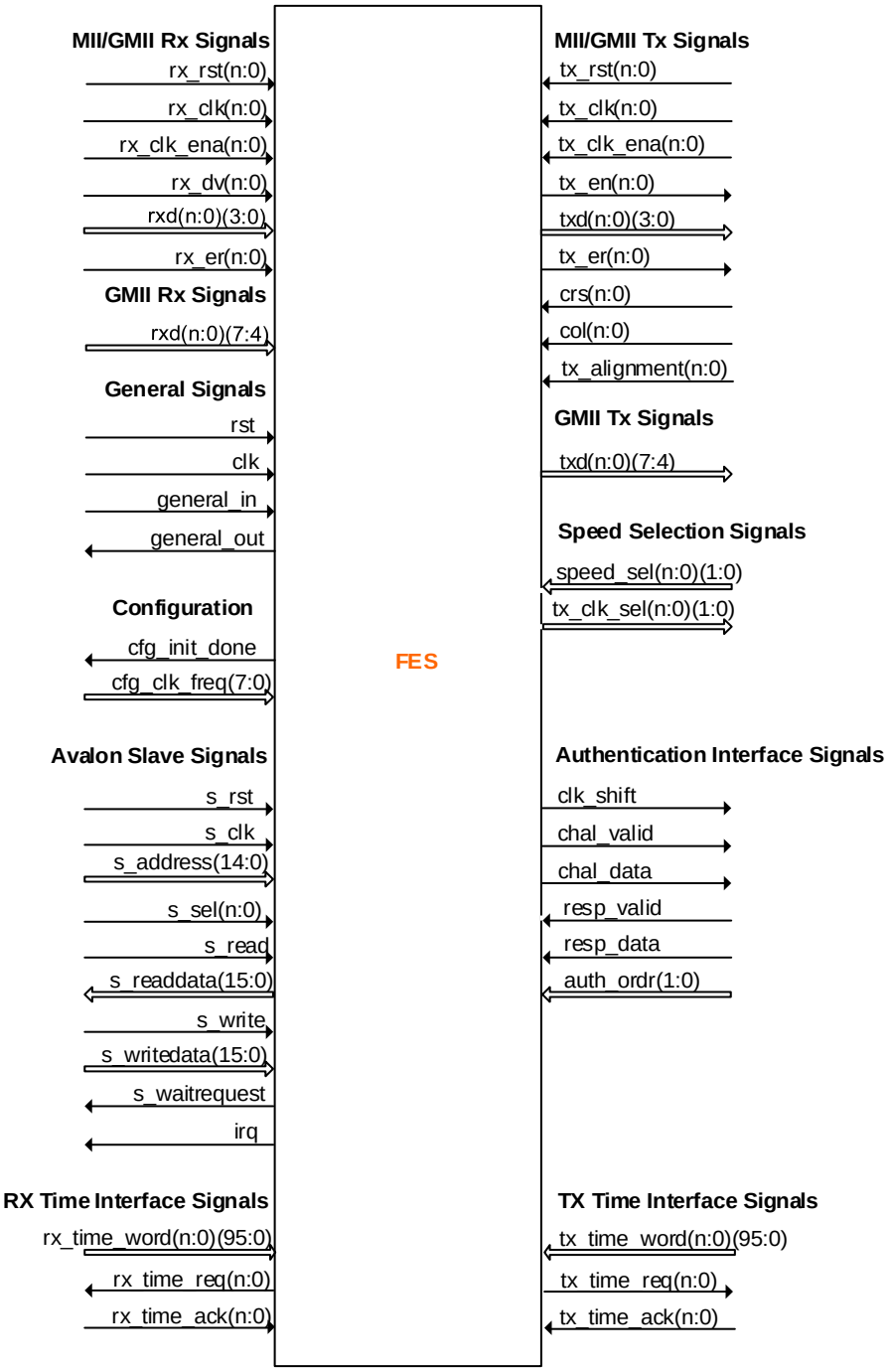
Address	Register	Description
SEC+ 0x0200	TX_KEY0_0	Reset: 0 x 00 00 The first 16 bits of TX key 0. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '0' in SEC_CNF register. Bits 15-0: R/W TX Key 0, bits 0…15
SEC+ 0x0202	TX_KEY0_1	Reset: 0 x 00 00 Bits 16…31 of TX key 0. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '0' in SEC_CNF register. Bits 15-0: R/W TX Key 0, bits 16…31
SEC+ 0x0204	TX_KEY0_2	Reset: 0 x 00 00 Bits 32…47 of TX key 0. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '0' in SEC_CNF register. Bit 15-0: R/W TX Key 0, bits 32…47
…	…	…
SEC+ 0x021E	TX_KEY0_15	Reset: 0 x 00 00 The last 16 bits of TX key 0. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '0' in SEC_CNF register. In case of AES-128 the key is only 128 bits and the extra registers must contain 0x0. Bits 15-0: R/W TX Key 0, bits 240...255
SEC+ 0x0220… SEC+ 0x023E	Reserved	Reserved
SEC+ 0x0240	TX_KEY1_0	Reset: 0 x 00 00 The first 16 bits of TX key 1. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '1' in SEC_CNF register. Bits 15-0: R/W TX Key 1, bits 0…15
SEC+ 0x0242	TX_KEY1_1	Reset: 0 x 00 00 Bits 16…31 of TX key 1. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '1' in SEC_CNF register. Bits 15-0: R/W TX Key 1, bits 16…31
SEC+ 0x0244	TX_KEY1_2	Reset: 0 x 00 00 Bits 32…47 of TX key 1. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '1' in SEC_CNF register. Bit 15-0: R/W TX Key 1, bits 32…47
…	…	…
SEC+ 0x025E	TX_KEY1_15	Reset: 0 x 00 00 The last 16 bits of TX key 1. Used for encryption and calculating the ICV for transmitted frames when Used TX Key = '1' in SEC_CNF register. In case of AES-128 the key is only 128 bits and the extra registers must contain 0x0. Bits 15-0: R/W TX Key 1, bits 240...255
SEC+ 0x0260… SEC+ 0x02FE	Reserved	Reserved
SEC+ 0x0300	RX_KEY0_0	Reset: 0 x 00 00 The first 16 bits of RX key 0. Used for decryption and validation of received frames. Bits 15-0: R/W RX Key 0, bits 0…15
SEC+ 0x0302	RX _KEY0_1	Reset: 0 x 00 00 Bits 16…31 of RX key 0. Used for decryption and validation of received frames. Bits 15-0: R/W RX Key 0, bits 16…31
SEC+ 0x0304	RX _KEY0_2	Reset: 0 x 00 00 Bits 32…47 of RX key 0. Used for decryption and validation of received frames. Bit 15-0: R/W RX Key 0, bits 32…47
…	…	…

Address	Register	Description
SEC+ 0x031E	RX _KEY0_15	Reset: 0 x 00 00 The last 16 bits of RX key 0. Used for decryption and validation of received frames. In case of AES-128 the key is only 128 bits and the extra registers must contain 0x0. Bits 15-0: R/W RX Key 0, bits 240...255
SEC+ 0x0320… SEC+ 0x033E	Reserved	Reserved
SEC+ 0x0340	RX _KEY1_0	Reset: 0 x 00 00 The first 16 bits of RX key 1. Used for decryption and validation of received frames. Bits 15-0: R/W RX Key 1, bits 0…15
SEC+ 0x0342	RX _KEY1_1	Reset: 0 x 00 00 Bits 16…31 of RX key 1. Used for decryption and validation of received frames. Bits 15-0: R/W RX Key 1, bits 16…31
SEC+ 0x0344	RX _KEY1_2	Reset: 0 x 00 00 Bits 32…47 of RX key 1. Used for decryption and validation of received frames. Bit 15-0: R/W RX Key 1, bits 32…47
…	…	…
SEC+ 0x035E	RX _KEY1_15	Reset: 0 x 00 00 The last 16 bits of RX key 1. Used for decryption and validation of received frames. In case of AES-128 the key is only 128 bits and the extra registers must contain 0x0. Bits 15-0: R/W RX Key 1, bits 240...255

6 External Signals

The external signals of FES are presented in Figure 40. The external signals consist of MII/GMII signals, Avalon slave signals, Time interface signals, Authentication interface signals and General signals.

Figure 40. External Signals of FES



6.1 Generics

The compile time configuration is presented in Table 16.

Table 16. FES Generics

Generic name	Default value	Description
RST_ACTIVE	1	Defines whether the reset input signal is active low or active high. '0': Active low reset signal '1': Active high reset signal
FES_PORT_HIGH	3	Number of Ethernet ports minus one. '2': 3-port FES '3': 4-port FES '4': 5-port FES … Allowed values: 2 to 7 when there are HSR or PRP enabled ports (Generics HSR_PORTS and PRP_PORTS). When there are no HSR or PRP ports values from 2 to 11 are allowed.
PORT_STATE_DEFAULT	0x0	Initial value for PORT_STATE registers. This generic can be used for example for defining whether the ports are open or closed in the startup.
COUNTERS	1	'0': Counter registers (Table 13) do not exist, the memory area is reserved. '1': Counter registers are as presented in Table 13.
CT_PORTS	0	Cut Through port vector. Defines between which two ports Cut Through operation is possible, if possible at all. Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. '0': Cut-Through feature does not exist for the port '1': Cut-Through is possible for the port Cut through can be configured only for zero or for two ports with this generic (only two of the bits can be '1' or all of them '0'). Enabling Cut-Through (bit 13 in GENERAL register) requires also that the two ports are in HSR redundant mode.
SMAC_TABLE_ROWS	0	Number of rows in Static MAC Address Table. Presented in powers of two: '0': Static MAC Table Disabled, SMAC_TABLE registers are reserved '7': 128 rows '8': 256 rows '9': 512 rows '10': 1024 rows '11': 2048 rows '12': 4096 rows Allowed values '0', '7'…'12' (disabled, 128…4096 rows).
POLICING	0	'0': Policing disabled, Registers POLICER0…POLICER2 are read-only and reserved. '1': Policing based on Inbound Policy (IPO) only '2': Policing based on Static MAC Table (SMAC) only '3': Policing based on either IPO or SMAC, selectable by register setting (see Policer Configuration in GENERAL register, Table 6)
POLICERS	7	Number of Policers per port. Presented in powers of two: '7': 128 policers per port '8': 256 policers per port '9': 512 policers per port '10': 1024 policers per port '11': 2048 policers per port '12': 4096 policers per port Allowed values '7'…'12' (128…4096 policers per port).

Generic name	Default value	Description
QUEUES	4	Number of output queues per port. Allowed values ‘4’ and ‘8’.
SHAPERS	0	Shapers enable. ‘0’: Shapers disabled, shaper registers are reserved ‘1’: Shapers enabled for all ports and queues
GIGABIT	0	Gigabit enable ‘0’: All ports support 10 Mbit/s and 100 Mbit/s only ‘1’: All ports support 10/100/1000 Mbit/s
HSR_PORTS	0	HSR enable port vector. Defines for which ports HSR functionality exists. Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. ‘0’: HSR functionality disabled for the port ‘1’: HSR functionality enabled for the port
PRP_PORTS	0	PRP enable port vector. Defines for which ports PRP functionality exists. Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. ‘0’: PRP functionality disabled for the port. ‘1’: PRP functionality enabled for the port.
MACSEC	0	MACsec enable port vector. Defines for which ports MACsec functionality exists. Bit 0 corresponds to port 0, bit 1 corresponds to port 1, and so on. ‘0’: MACsec disabled for the port. MACsec registers are reserved. ‘1’: MACsec enabled for the port.
MACSEC_CIPHER	1	MACsec Cipher Suite selection. ‘1’: GCM-AES-128 Other values are reserved.

6.2 General Signals

The General Signals consist of global reset, clocks and configuration signals. The General Signals are presented in Table 17.

Table 17. General Signals

Name	Direction	Description
rst	Input	This is a global asynchronous signal that resets all the flip-flops of FES to their initial values. Active high or low depending on generic RST_ACTIVE (see Table 16).
clk	Input	This signal is the FES main clock and its frequency must be greater than or equal to 125 MHz. Only the rising edge of the signal is used. This signal should be connected to a low skew clock buffer. If there is a shortage of low skew clock buffers at the FPGA, this signal should have priority over the rx_clk(n:0) and tx_clk(n:0) signals. If FES is going to be used only in 10/100 Mbit modes, the frequency must be greater than or equal to 50 MHz.
cfg_init_done	Output	When high, FES initialization is done. Goes low only during HW and SW reset.

cfg_clk_freq(7:0)	Input	Informs FES about main clock (clk) frequency used. This is given in 1 MHz increments as follows: 0 = N/A 1 = 1 MHz 2 = 2 MHz … 255= 255 MHz
general_in	Input	This general purpose input signal can be connected to other FPGA blocks to monitor their functionality. The status of the signal can be read from GENERAL register (see Table 6). This signal can be used to monitor whether PHY polling of the MDIO-to-Avalon Bridge is enabled or not [8].
general_out	Output	This general purpose output signal can be connected to other FPGA blocks to control their functionality. The signal can be controlled using GENERAL register (see Table 6). This signal can be used to control whether PHY polling of the MDIO-to-Avalon Bridge is enabled or not [8].

6.3 Avalon Slave Signals

Avalon Slave signals are presented in Table 18. The Avalon interface is used for register access only. See Avalon specification [12] for more information on Avalon interface and signals.

Table 18. Avalon Slave Signals

Name	Direction	Description
s_rst	Input	Reset for Avalon clock domain. Active high or low depending on generic RST_ACTIVE (see Table 16).
s_clk	Input	Avalon Clock
s_address(14:0)	Input	Avalon word address.
s_sel	Input	Selects FES internal Avalon Slave. 0 = FES Switch Configuration Registers, 1 = FES Port Configuration Registers for port 0, 2 = FES Port Configuration Registers for port 1, …
s_read	Input	Indicates read transfer.
s_readdata(15:0)	Output	Data read in read transfer.
s_write	Input	Indicates write transfer.
s_writedata(15:0)	Input	Data written in write transfer.
s_waitrequest	Output	Asserted when the slave is unable to respond to a read or write request. Forces the master to wait until ready to proceed with the transfer.
irq	Output	When an interrupt is activated, the corresponding bit in the Interrupt Status register is given value 1. FES activates this interrupt signal when any one of the interrupt status bits in the Interrupt Status register is set and the corresponding Interrupt Mask bit has value 1. The signal is level sensitive, active high.

6.4 Time Interface Signals

Time Interface consists of signals used for time information exchange between FES and the rest of the system. The handling of the time information is left outside of FES because its implementation may vary significantly depending on the hardware.

The clock time information is used in time stamping of the Ethernet frames received and transmitted. An exact receive and transmit time information is essential for the IEEE 1588 Precision Time Protocol to achieve its best accuracy.

Table 19. Time Interface Signals

Name	Direction	Description
tx_alignment	Input	This transmit alignment control signal can be used to adjust transmit start timing to achieve optimal symbol alignment when using fiber connection. This signal can be connected to tx alignment signal of an external SGMII adapter block. **If used**, this signal must toggle on each clock cycle. **If not used**, this signal must be tied to '0'. This signal is synchronous to the MII/GMII interface transmit clock.
rx_time_req	Output	Receive Time Request. When a frame is received, at the moment when the Frame Timestamp Point is in MII/GMII interface, this signal toggles. This signal is synchronous to the MII/GMII interface receive clock. If time interface is not used, this signal must be tied to rx_time_ack.
rx_time_ack	Input	Receive Time Acknowledge. This signal should toggle to follow the `rx_time_req` signal when data is updated on the `rx_time_word(95:0)` signal. This signal is synchronized to the clk clock. If time interface is not used, this signal must be tied to rx_time_req.
rx_time_word(95:0)	Input	Receive Time Word. This signal should be updated when the `rx_time_req` signal has opposite state to the `rx_time_ack` signal. The format is the following: rx_time_word(15:0): subnanoseconds (2^16 parts of a nanosecond) rx_time_word(45:16): nanoseconds rx_time_word(47:46): reserved (=0) rx_time_word(95:48): seconds This signal is synchronous to the MII/GMII interface receive clock. If time interface is not used, this bus must be tied to zero.
tx_time_req	Output	Transmit Time Request. This signal toggles to request up to date timestamp data. This signal is synchronous to the MII/GMII interface transmit clock. If time interface is not used, this signal must be tied to tx_time_ack.
tx_time_ack	Input	Transmit Time Acknowledge. This signal should toggle to follow the `tx_time_req` signal when data is updated on the `tx_time_word(95:0)` signal. This signal is synchronized to the clkclock. If time interface is not used, this signal must be tied to tx_time_req.
tx_time_word(95:0)	Input	Transmit Time Word. This signal should be updated when the `tx_time_req` signal has opposite state to the `tx_time_ack` signal. rx_time_word(15:0): subnanoseconds (2^16 parts of a nanosecond) rx_time_word(45:16): nanoseconds rx_time_word(47:46): reserved (=0) rx_time_word(95:48): seconds This signal is synchronous to the MII/GMII interface transmit clock. If time interface is not used, this bus must be tied to zero.

6.5 Authentication Interface Signals

Authentication Interface Signals are connected to respective signals of external (to FPGA) security chip, if such chip exists. If there is no security chip, input signals should be tied to '0' and output signals left unconnected.

Auth_ordr(1:0) signal is connected to "00" if there is just one FES core on the FPGA. In case of multiple FES instances see Appendix 2.

6.6 MII/GMII Signals

The Media Independent Interface (MII) and the Gigabit Media Independent Interface (GMII) signals of FES MAC are presented in Table 20. The MII/GMII signals consist of the MII/GMII RX and the MII/GMII TX signals. Every MAC entity of FES has its own set of MII/GMII signals: the MII/GMII signals are presented as vectors in Table 20, where n is the number of MAC blocks in FES minus one.

The MII/GMII RX signals are synchronous to rx_clk and the MII/GMII TX signals are synchronous to tx_clk. Crs and col signals can be asynchronous with respect to tx_clk (and rx_clk). For further information about the MII interface see IEEE Std 802.3 [13].

The MII/GMII signals are presented in Table 20. Note that in the table, unless otherwise stated, all the signals that can activate or enable something are active when their state is high and inactive when their state is low.

Figure 41. (G)MII signal timing for signal port

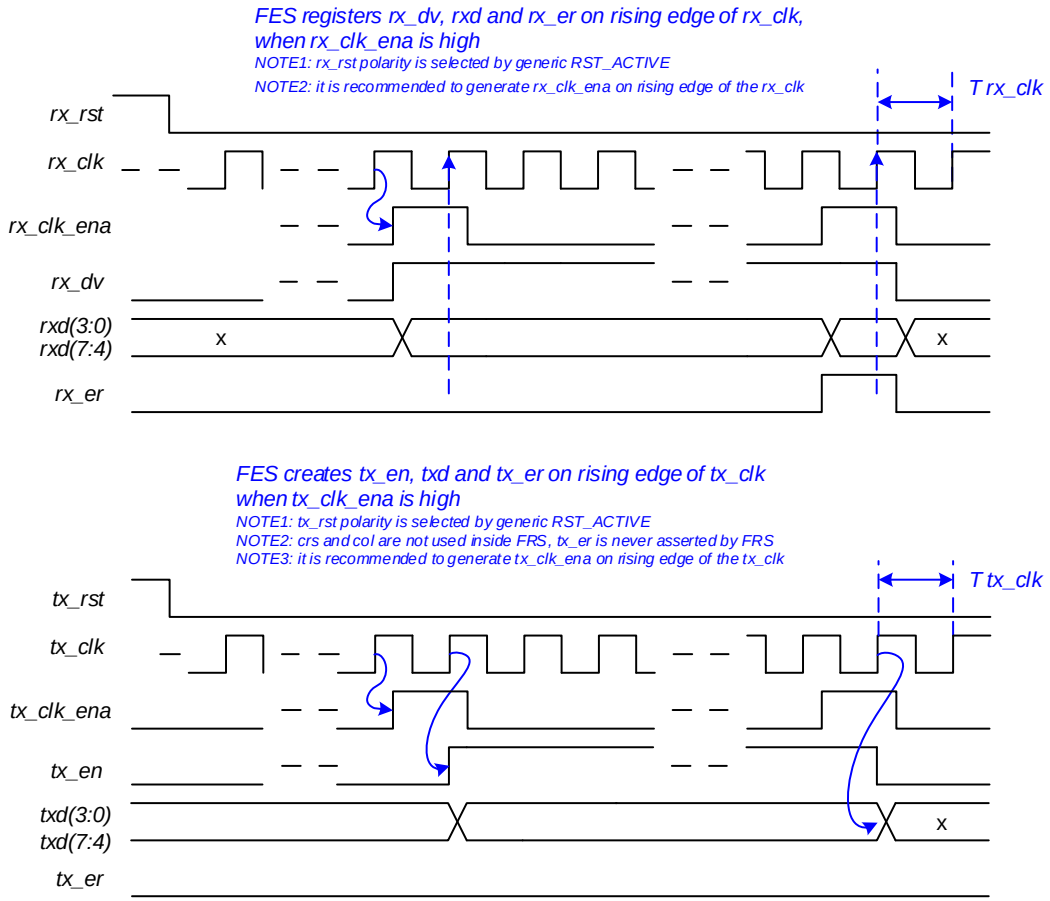


Table 20. MII/GMII Signals

Name	Direction	Description
rx_rst(n:0)	Input	MII/GMII Receive Reset. Resets the corresponding clock domain logic. Active high or active low depending on generic RST_ACTIVE (see Table 16).
rx_clk(n:0)	Input	MII/GMII Receive Clock. The frequency is 2.5 MHz in the 10 Mbps mode, 25 MHz in the 100 Mbps mode and 125MHz in 1000 Mbps mode. It is allowed that the clock is not running continuously. Discontinuous receive clock does not cause malfunction in FES MAC. This signal should be connected to a low skew clock buffer.

rx_clk_ena(n:0)	Input	MII/GMII Receive Clock Enable. The Receive MII signals are registered only when this signal is at high state. This signal can be used to generate lower MII data rates using a single clock source connected to rx_clk. In case rx_clk signal already has correct frequency, tie this signal to logic '1'.
rx_dv(n:0)	Input	MII/GMII Receive Data Valid. Indicates when the data lines are carrying valid data from the PHY.
rxd(n:0)(3:0)	Input	MII/GMII Receive Data. Data lines from the PHY.
rxd(n:0) (7:4)	Input	GMII Receive Data. Data lines from the PHY.
rx_er(n:0)	Input	MII/GMII Receive Error. Indicates that an error has been detected while receiving a frame.
tx_rst(n:0)	Input	MII/GMII Transmit Reset. Resets the corresponding clock domain logic. Active high or active low depending on generic RST_ACTIVE (see Table 16).
tx_clk(n:0)	Input	MII/GMII Transmit Clock. The frequency is 2.5 MHz in the 10 Mbps mode, 25 MHz in the 100 Mbps mode and 125MHz in 1000 Mbps mode. It is allowed that the clock is not running continuously. Discontinuous transmit clock does not cause malfunction in FES MAC. This signal should be connected to a low skew clock buffer.
tx_clk_ena(n:0)	Input	MII/GMII Transmit Clock Enable. The Transmit MII signals are generated only when this signal is at high state. This signal can be used to generate lower MII data rates using a single clock source connected to tx_clk. In case tx_clk signal already has correct frequency, tie this signal to logic '1'.
tx_en(n:0)	Output	MII/GMII Transmit Enable. Indicates that the data lines are carrying valid data to the PHY.
txd(n:0) (3:0)	Output	MII/GMII Transmit Data. Data lines to the PHY.
txd(n:0) (7:4)	Output	GMII Transmit Data. Data lines to the PHY.
tx_er(n:0)	Output	MII/GMII Transmit Error. Indicates that an error has occurred while transmitting the frame.
crs(n:0)	Input	MII Carrier Sense. Indicates that the medium is busy.
col(n:0)	Input	MII Collision Detection. Indicates that a collision is detected on the medium.

6.7 Speed Selection Signals

Speed selection signals are presented in Table 21.

Table 21. Speed Selection Signals

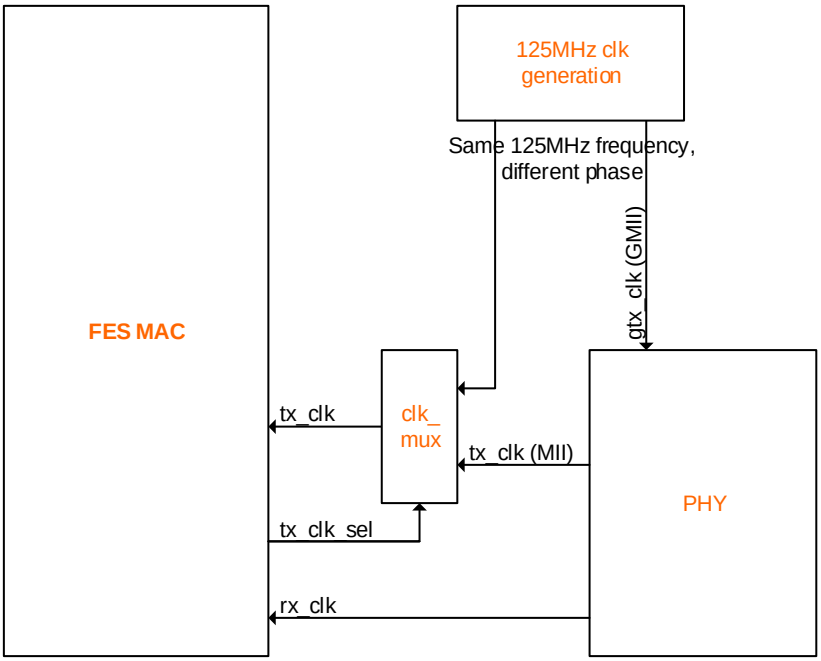
Name	Direction	Description
speed_sel(n:0)(1:0)	Input	MII/GMII interface speed selection Selects the speed of the interface in case the speed is selectable by external signals. This is configured in PORT_STATE register (see Table 10). 0 0 = reserved 0 1 = 1000 Mb/s 1 0 = 100 Mb/s 1 1 = 10 Mb/s

tx_clk_sel(n:0)(1:0)	output	MII/GMII interface current speed information Reports the line speed of the interface. Indicates also whether GMII or MII interface is used, which affects to the direction of the PHYsical layer chip transmit clock. 0 0 = reserved 0 1 = 1000 Mb/s (GMII) 1 0 = 100 Mb/s (MII) 1 1 = 10 Mb/s (MII)

6.8 Transmit Clock Multiplexer

In gigabit (GMII) mode the receive and the transmit clock speeds are higher, and the direction of the transmit clock signal towards PHY is opposite than in 10 Mbps and 100 Mbps (MII) modes. For this reason, and also because it is reasonable to generate the high-speed GMII transmit clock (gtx_clk) outside of FES, FES MAC employs an external clock multiplexer. The basic connection of the clock multiplexer (clk_mux) with FES MAC is presented in Figure 42. The clock multiplexer is not needed if FES MAC is used only in 10/100 Mbps mode, or only in 1000 Mbps mode.

Figure 42. External Transmit Clock Multiplexer



6.8.1 Interfacing to CPU with MII/GMII

Connecting MII/GMII between FES and a CPU is illustrated in Figure 43.

Figure 43. Host Interface Connection

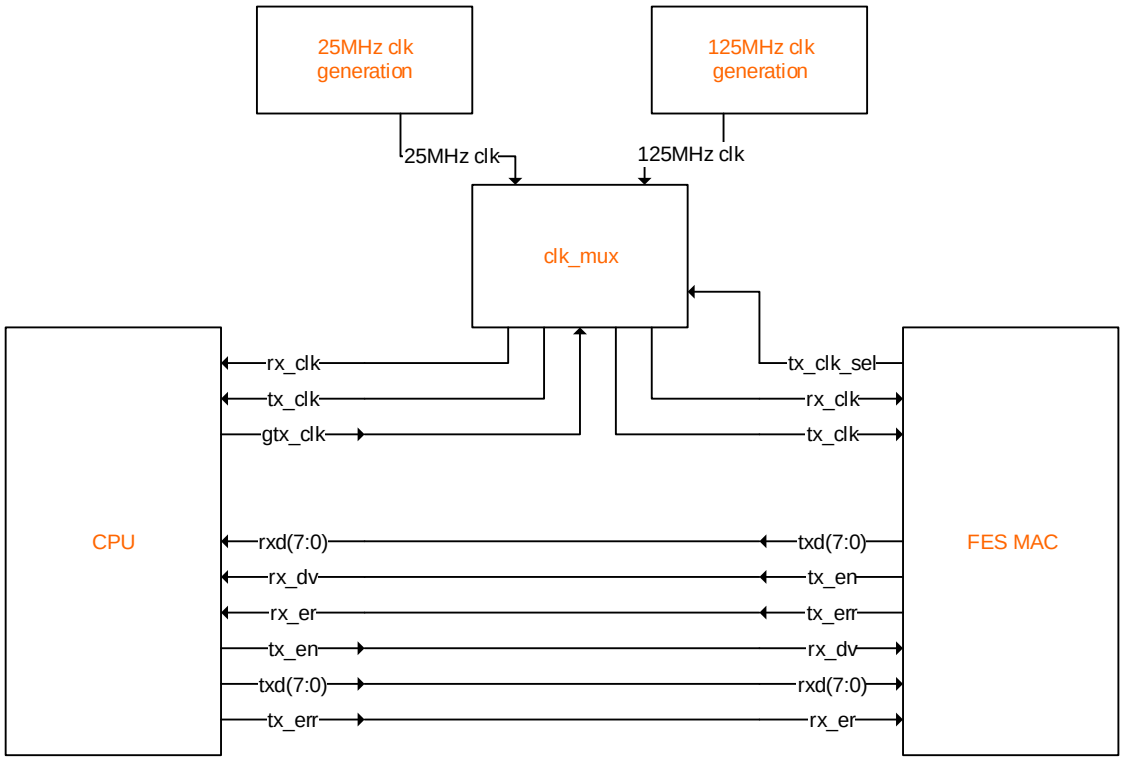


Table 22 presents functionality of clk_mux for MII/GMII clocks in MII and GMII modes.

Table 22. MII/GMII Clock Mux Configuration

Signal	Source (MII mode)	Source (GMII mode)
CPU rx_clk	25MHz clk	125MHz clk
CPU tx_clk	25MHz clk	-
FES rx_clk	25MHz clk	CPU gtx_clk
FES tx_clk	25MHz clk	125MHz clk

7 Abbreviations

ASIC	Application Specific Integrated Circuit
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DST	Destination (Address)
FED	Frame Early Drop
FES	Flexibilis Ethernet Switch
FRS	Flexibilis Redundant Switch
FIFO	First In First Out
FPGA	Field Programmable Gate Array
GMII	Gigabit Media Independent Interface
HSR	High-availability Seamless Redundancy
ICV	Integrity Check Value
IEEE	Institute of Electrical & Electronics Engineers, Inc
IEC	International Electrotechnical Commission
LAN	Local Area Network
MAC	Media Access Control
MACsec	MAC Security
MDC	Management Data Clock
MDIO	Management Data Input/Output
MII	Media Independent Interface
MMD	MDIO Managed Device
MUX	MUltipleXer
PHY	Physical layer device
PN	Packet Number
PRP	Parallel Redundancy Protocol
PTP	Precision Time Protocol
RSTP	Rapid Spanning Tree Protocol
RX	Receive
SC	Secure Channel
SCI	Secure Channel Identifier
SecTAG	MAC Security TAG
SFD	Start Frame Delimiter
SRC	Source (Address)
STA	Station Management Entity
STP	Spanning Tree Protocol
TX	Transmit
VHDL	Very high speed integrated circuits Hardware Description Language

VLAN	Virtual LAN

8 References

[1] “Flexibilis Ethernet Switch/Router Specification”, Flexibilis Oy, 2011

[2] IEEE Std 802.3x-1997, “Specification for 802.3 Full Duplex Operation and Physical Layer Specification for 100 Mb/s Operation on Two Pairs of Category 3 Or Better Balanced Twisted Pair Cable (100BASE-T2)”, 1997

[3] IEEE Std 802.1D, “Media Access Control (MAC) Bridges”, 2004.

[4] “Ethernet Interface Adapters Specification”, Flexibilis Oy, 2015

[5] IEEE Std 1588-2008: IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 2008.

[6] Rich Seifert, “The Switch Book, The complete guide to LAN switching technology”, John Wiley & Sons inc, 2000, page 299

[7] IEEE Std 802.1w-2001, “Media Access Control (MAC) Bridges: Amendment 2 – Rapid Reconfiguration“, 2001

[8] “MDIO-to-Avalon Bridge Manual”, Flexibilis Oy, 2016

[9] IEC62439-3 , “Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)”, 2010, Clause 5

[10] IEEE Std 802.1AE-2006, “Media Access Control (MAC) Security”, 2006

[11] “MEF Technical Specification 10.3”, The MEF Forum, 2013, Chapter C.3.1

[12] “Avalon Interface Specifications”, Altera Corporation, May 2011, http://www.altera.com/literature/manual/mnl_avalon_spec.pdf

[13] IEEE Std 802.3-2008, “Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications”, 2008

Appendix 1. Authentication Interface Multiplexer

Authentication Interface Multiplexer makes it possible to have more than one FES core on an FPGA without having multiple Security Chips; the Authentication Interface Multiplexer makes it possible for two to four FES cores to share one single Security Chip.

Figure 44. Authentication Interface Multiplexer

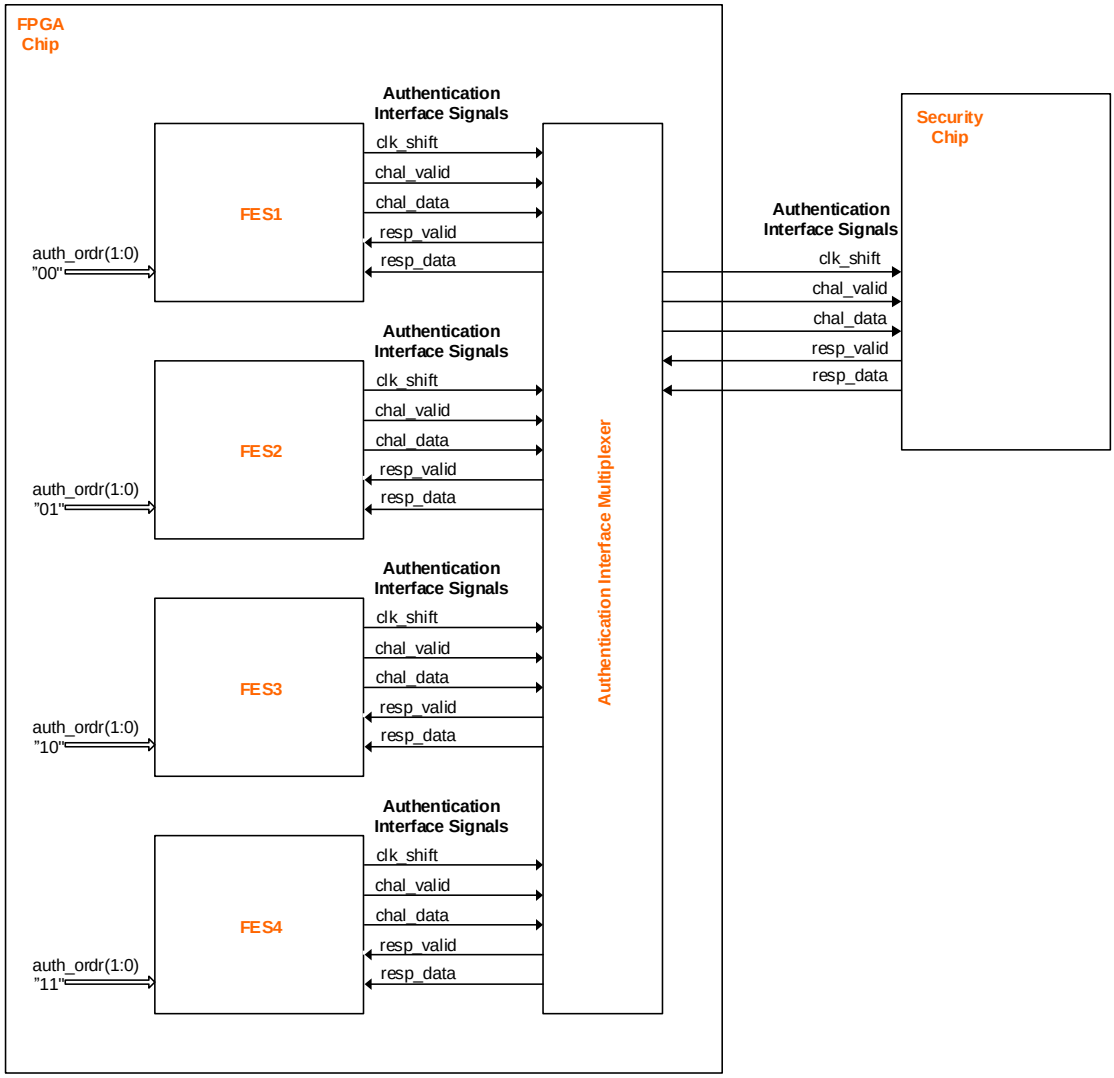


Figure 44 presents how to connect Authentication Interface Multiplexer. The Authentication Interface Multiplexer forwards authentication requests from the connected FES cores to the Security Chip one at a time. Authentication response from the Security Chip is then forwarded to the FES who made the request. To prevent FES cores from making the authentication requests at the same time, there is signal auth_ordr(1:0). auth_ordr(1:0) is set to a different value for each FES core (see Figure 44).

If less than four FES cores are connected to Authentication Interface Multiplexer, the unused inputs of Authentication Interface Multiplexer are connected to “0” and the unused outputs are left unconnected.