User's Manual



# RMAC
# (R-Switch-3 Ethernet MAC)

Document number : 0.50
Date of issue : 2025/02/18

- This document and contents shall be strictly managed for all document users and used for specific/limited purpose only, issued under NDA between Renesas Electronics and the company of this document holders
- Under development
- Preliminary document
  Specification in this document are tentative and subject to change
- Renesas Electronics Confidential - issued under NDA



Printed in Japan

# Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(2012.4)

General precautions for handling of product

The following notes are applicable to entire CBIC with CPU core. For detailed usage notes, refer to the relevant sections of the manual. If the description under General precautions and in the body of the manual differs from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.

- The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flow internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

The state of the product is undefined at the moment when power is supplied.

- The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses Access to reserved addresses is prohibited.

- The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Regarding Clock

After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized. When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

# Contents

**Contents**

## List of Figure

## List of Table

# 1.Overview

This IP is an Ethernet controller that conforms to the definition of the MAC (Media Access Control) layer for Ethernet in the IEEE 802.3 standard.

When connected with a physical-layer LSI chip (PHY-LSI) that complies with the standard, it is able to transmit and receive Ethernet (IEEE 802.3) frames. It has a single MAC layer interface.

## 1.1 Features

Table 1-1 lists the specifications.

**Table 1-1 Specifications (Functions)**

| Item | Description |
|---|---|
| Protocol | Flow control conforming with the IEEE 802.3x standard |
| | Priority-based Flow Control with 802.1Qbb |
| | Interspersing Express Traffic (preemption) conforming with IEEE 802.3br |
| | Support for IEEE 1588 time synchronization,<br>Support for IEEE AS and IEEE AS-Rev acceleration |
| Transfer speed | Supports transfer at 10,100,1000, 2500, 5000, and 10000Mbps |
| Mode | Full-duplex mode |
| | Energy Efficient Ethernet (Low Power Idle) |
| Interface | Supports the IEEE 802.3 standard MII (Media Independent Interface) , GMII (Gigabit Media Independent Interface), and XGMII<br>Supports also USXGMII |
| | PHY MDIO programming Interface |
| | Supports internal register interface (32bit data width)<br>Supports internal data interface (64bit data width) |
| Original function | Configurable frame data padding<br>A variety of flexible RX address filtering modes<br>Configurable CRC pass through for transmit and receive<br>Network statistic counter (IEEE 802.3, ROMN RFC2819) |
| Magic Packet™ | Detection of Magic Packets™* and output of a detected signal |

Note: * Magic Packet™ is a trademark of Advanced Micro Devices, Inc.

**Table 1-2 References**

| Title | Contents |
|---|---|
| IEEE Std 802.3™-2008 | Carrier sense multiple access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications |
| IEEE 802.3az™-2010 | Media Access Control Parameters, Physical Layers, and Management Parameters for Energy-Efficient Ethernet |
| IEEE Std 802.3br™-2016 | Specification and Management Parameters for Interspersing Express Traffic |
| P802.1AS-Rev™ | Timing and Synchronization for Time Sensitive Applications |
| P802.1Qbu™ | Frame Preemption |
| IEEE Std 1588™-2008 | Precision Clock Synchronization Protocol for Networked Measurement and Control Systems |
| IEEE Std 802.1BA™ - 2011 | Audio Video Bridging (AVB) Systems |
| USXGMII | Universal Serial 10GE Media Independent Interface |
| IEEE802.1Qbb | Priority-based Flow Control |

The terms used in this RMAC section is listed below.

**Table 1-3 Terms**

| Terms | Explanation |
|---|---|
| eMDIO | Extension Management Data Input/Output interface<br>Management Data Input/Output interface defined in Clause 45 of IEEE802.3 specification. |
| MHD | MHD or MHD I/F means interface between RMAC layer and upper layer. Tx MHD I/F is used to indicate frame transmission request, Rx MHD I/F is used to transfer received data derived from received ethernet frame. |

## 1.2 RMAC Block Diagram

Figure 1-1 represents RMAC block diagram.

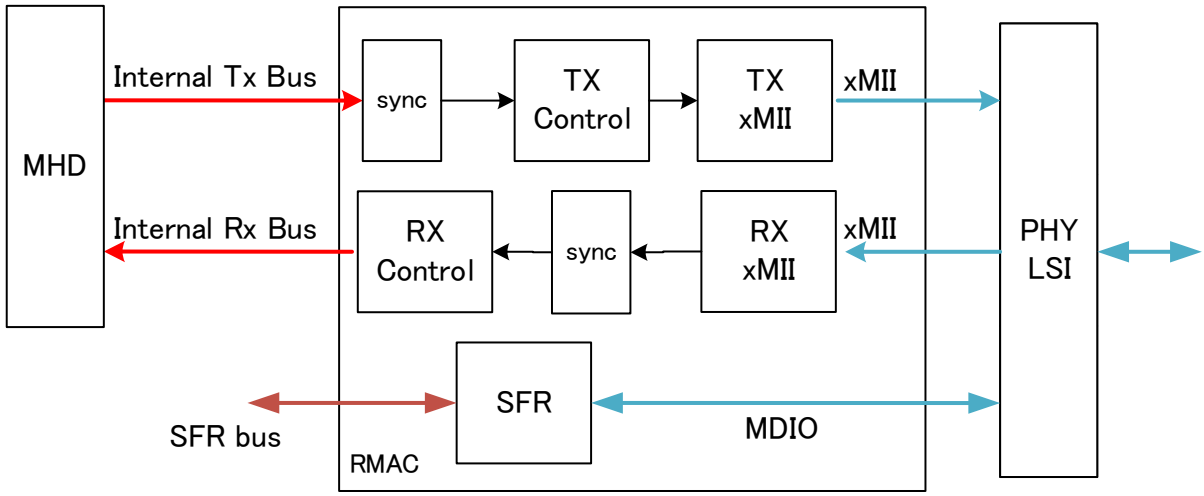



**Figure 1-1 Block Diagram**

| Block | Function |
|---|---|
| MHD | Message HanDler (Upper layer module).<br>In R-Switch, this is “TSNA(ETHA)”. |
| SFR | Contains all the RMAC SFRs and the bridging system to access PHY registers.<br>RMAC registers are on clk. |
| Rx Control | Consists of the receive FSM and Data path logic, which disassembles the received Ethernet frame.<br>It controls functions like checking and removing FCS, Destination Address filtering, Flow Control etc. |
| Tx Control | Consists of the transmission FSM and Data path logic, which facilitates the transmitted Ethernet frame.<br>It controls functions like adding the Preamble and SFD, adding PAD bytes, calculating the FCS etc. |
| Rx xMII / Tx xMII | Provides the connectivity to the PHY.<br>It supports several types of xMII (MII, GMII, XGMII). |
| RMAC | IP described in this documentation. |
| Sync | Synchronized between bus clock domain and mii clock domain. |

# 2.Parameter list

RMAC global parameter list is shown in table2-1.

RMAC local parameter list is shown in table 2-2.

Table 2-1 Global parameter list

| **Parameter Name** | **RSW3 Values** | **Explanation** |
|---|---|---|
| **Grobal** | | |
| PORT_TSNA_N | 13 | TSNA(ETHA) port number |
| PORT_GWCA_N | 2 | GWCA port number |
| PORT_N | 15 | GWCA and ETHA port number |
| PORT_W | 4 | GWCA and ETHA port width |
| **Frame** | | |
| PAS_LVL_N | 2 | Pause level number handled by COMA |
| FRM_PRIO_N | 8 | Priority number handled by GWCA |
| **gPTP timer** | | |
| PTP_TN | 2 | gPTP timer number connected to the switch |
| PTP_TUNES | 3 | Timestamp unique number extension size |

Table 2-2 Local parameter list

| **Parameter Name** | **Value** | **Explanation** |
|---|---|---|
| **Frame** | | |
| FRM_SMD_E | 8‘hd5 | E-frame SMD value |
| FRM_SMD_V | 8‘h07 | V-frame SMD value |
| FRM_SMD_R | 8‘h19 | R-frame SMD value |
| FRM_SMD_S0 | 8‘he6 | S0-frame SMD value |
| FRM_SMD_S1 | 8‘h4c | S1-frame SMD value |
| FRM_SMD_S2 | 8‘h7f | S2-frame SMD value |
| FRM_SMD_S3 | 8‘hb3 | S3-frame SMD value |
| FRM_SMD_C0 | 8‘h61 | C0-frame SMD value |

| Parameter Name | Value | Explanation |
|---|---|---|
| FRM_SMD_C1 | 8‘h52 | C1-frame SMD value |
| FRM_SMD_C2 | 8‘h9e | C2-frame SMD value |
| FRM_SMD_C3 | 8‘h2a | C3-frame SMD value |
| FRM_FC0 | 8’he6 | FRAG_COUNT0 value |
| FRM_FC1 | 8’h4c | FRAG_COUNT1 value |
| FRM_FC2 | 8’h7f | FRAG_COUNT2 value |
| FRM_FC3 | 8’hb3 | FRAG_COUNT3 value |
| TPLW | 16 | Total payload length width |
| MCRC_MAGIC_NUMBER | 32'hff48647d | MCRC magic number |
| FCS_MAGIC_NUMBER | 32'hc704dd7b | FCS magic number |
| **gPTP timer** | | |
| PTP_TN_W | 1 | gPTP Timer number Width |
| **SFR parameter** | | |
| COUNT_LOW_W | 32 | Counter SFR width |
| COUNT_MED_W | 32 | Counter SFR width |
| COUNT_HIGH_W | 64 | Counter SFR width |
| **Buffer parameter** | | |
| TXDEPTH | 8 | Buffer depth for tx async static buffer |
| RXDEPTH | 8 | Buffer depth for rx async static buffer |

# 3.Pin Function

## 3.1 Pin Description

### 3.1.1 Clock/Reset Interfaces

RMAC has three clocks as inputs. The required frequencies depending on the PHY type are described in the below table.

**Table 3-1 Summary of clock frequency**

| PHY Interface | Baud Rate [Mbps] | clk_phy_rx [MHz] | clk_phy_tx [MHz] | Clk minimum value [MHz] |
|---|---|---|---|---|
| MII | 10 | 2.5 | 2.5 | 0.8 |
| | 100 | 25 | 25 | 8 |
| GMII | 1000 | 125 | 125 | 80 |
| | 2500 * | 312.5 | 312.5 | 80 |
| XGMII *2 | 2500 | 78.125 | 78.125 | 80 |
| | 5000 | 156.25 | 156.25 | 160 |
| | 10000 | 312.5 | 312.5 | 320 |

* : GMII 2.5Gbps is a special setting for interfaces connected on-chip with TSN-ES. Do not use it for external interfaces.

### 3.1.2 PHY Interfaces

PHY interface supports several Media Interfaces. Each media interface has its own interface signals.

PHY interface are described in “IEEE Std 802.3™-2008”.

IPG(inter packet gap) of RMAC is defined as follows.

| MII TYPE | Tx/Rx | IPG |
|---|---|---|
| GMII | Tx | Fixed 12 byte |
| | Rx | Minimum 12 byte |
| MII | Tx | Fixed 12 byte |
| | Rx | Minimum 12 byte |
| XGMII | Tx | Minimum 9 byte |
| | Rx | Minimum 5 byte |

### 3.1.3 MDIO interface

Refer to MDIO standard “IEEE Std 802.3™-2018”.

### 3.1.4 MHD TX timestamp capture interface

If a timestamp capture is request to RMAC on TX side (by SW or HW), this interface is used to transmit the timestamp.

This interface is based on a handshake, the interface time diagram is described in the figure bellow.

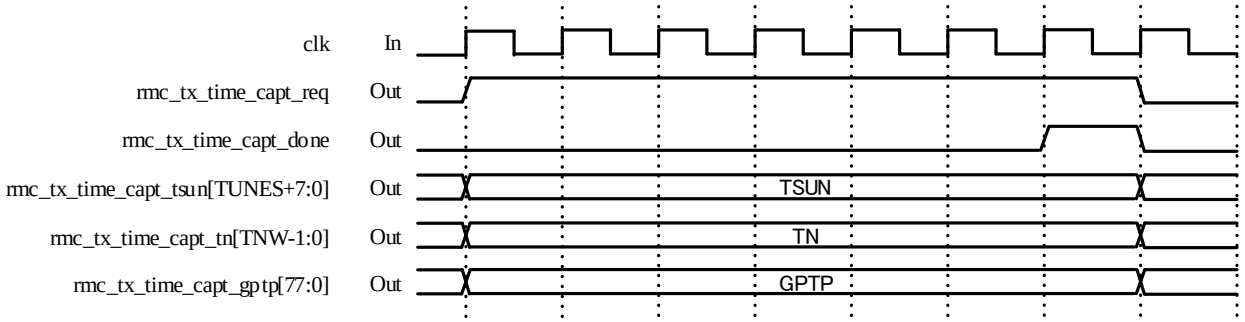



**Figure 3-1 TX capture interface time chart**

# 4.Registers

## 4.1 Register list

The RMAC register list is described in Table **4-1**. RMRO (RMAC Register Offset) indicates base address of address space allocated to RMAC by the system. RMAC uses only the 12-lower address bits. If the RMAC is not use with default parameters, it should be taken in account by the user while reading the SFR representation.

All registers have the -P attribute when connected with RSW2 or RSW3. (-P means “Protected” refer to Register attributes of [TOP])

Access Mode:

- Any: Register can be accessed in any mode.
- D: Register can be accessed in DISABLE mode
- C: Register can be accessed in CONFIG mode
- O: Register can be accessed in OPERATION mode

Note:

- All registers can be read in any mode.

Table 4-1: List of RMAC registers

| Offset/Address | Register name | Abbreviation | Write Access Mode |
|---|---|---|---|
| RMRO + 0000H | MAC PHY station management | MPSM | D,C,O |
| RMRO + 0004H | MAC PHY interfaces configuration | MPIC | C |
| RMRO + 0008H | MAC PHY interfaces monitoring | MPIM | - - |
| RMRO + 0010H | MAC IO Configuration | MIOC | D,C,O |
| RMRO + 0020H | MAC transmission frame format configuration | MTFFC | C |
| RMRO + 0024H | MAC transmission pause or PFC frame configuration | MTPFC | C |
| RMRO + 0028H | MAC transmission pause or PFC frame configuration2 | MTPFC2 | C,O |
| RMRO + 0030H + 4*t | MAC transmission pause or PFC frame configuration 3 t (t=0..PAS_LVL_N-1) | MTPFC3t (t=0..PAS_LVL_N-1) | C |
| RMRO + 0060H | MAC transmission interfaces monitoring | MTIM | - - |
| RMRO + 0080H | MAC reception general configuration | MRGC | C |
| RMRO + 0084H | MAC reception MAC address configuration 0 | MRMAC0 | C,O |
| RMRO + 0088H | MAC reception MAC address configuration 1 | MRMAC1 | C,O |
| RMRO + 008CH | MAC reception address filter configuration | MRAFC | C,O |
| RMRO + 0090H | MAC reception storm configuration e-frame | MRSCE | C,O |
| RMRO + 0094H | MAC reception storm configuration p-frame | MRSCP | C,O |
| RMRO + 0098H | MAC reception storm counter control | MRSCC | D,C,O |
| RMRO + 009CH | MAC reception frame size configuration for e-frames | MRFSCE | C |
| RMRO + 00a0H | MAC reception frame size configuration for p-frames | MRFSCP | C |
| RMRO + 00a4H | MAC Timestamp reception configuration | MTRC | C |
| RMRO + 00a8H | MAC reception interfaces monitoring | MRIM | - - |
| RMRO + 00aCH | MAC reception Pause or PFC frame monitoring | MRPFM | - - |
| RMRO + 0100H + 4*t | MAC PTP filtering configuration t (t=0.. PTP_TN*8-1) | MPFCt (t=0..PTP_TN*8-1) | C |
| RMRO + 0180H | MAC link verification configuration | MLVC | O |
| RMRO + 0184H | MAC energy efficient ethernet configuration | MEEEC | D,C,O |
| RMRO + 0188H | MAC loopback configuration | MLBC | C |
| RMRO + 0190H | MAC XGMII configuration | MXGMIIC | O |
| RMRO + 0194H | MAC XGMII PCH configuration | MPCH | C |
| RMRO + 0198H | MAC XGMII Auto-neg configuration | MANC | O |
| RMRO + 019CH | MAC XGMII Auto-neg message | MANM | D,C,O |

| Offset/Address | Register name | Abbreviation | Write Access Mode |
|---|---|---|---|
| RMRO + 0200H | MAC error interrupt status | MEIS | D,C,O |
| RMRO + 0204H | MAC error interrupt enable | MEIE | D,C,O |
| RMRO + 0208H | MAC error interrupt disable | MEID | D,C,O |
| RMRO + 0210H | MAC monitoring interrupt status 0 | MMIS0 | D,C,O |
| RMRO + 0214H | MAC monitoring interrupt enable 0 | MMIE0 | D,C,O |
| RMRO + 0218H | MAC monitoring interrupt disable 0 | MMID0 | D,C,O |
| RMRO + 0220H | MAC monitoring interrupt status 1 | MMIS1 | D,C,O |
| RMRO + 0224H | MAC monitoring interrupt enable 1 | MMIE1 | D,C,O |
| RMRO + 0228H | MAC monitoring interrupt disable 1 | MMID1 | D,C,O |
| RMRO + 0230H | MAC monitoring interrupt status 2 | MMIS2 | D,C,O |
| RMRO + 0234H | MAC monitoring interrupt enable 2 | MMIE2 | D,C,O |
| RMRO + 0238H | MAC monitoring interrupt disable 2 | MMID2 | D,C,O |
| RMRO + 0300H | MAC manual pause frame transmit counter | MMPFTCT | - - |
| RMRO + 0304H | MAC automatic pause frame transmit counter | MAPFTCT | - - |
| RMRO + 0308H | MAC pause frame receive counter | MPFRCT | - - |
| RMRO + 030cH | MAC false carrier indication counter | MFCICT | - - |
| RMRO + 0310H | MAC energy efficient ethernet counter | MEEECT | - - |
| RMRO + 0320H + 4*t | MAC manual PFC frame transmit counter t (t=0..PAS_LVL_N-1) | MMPCFTCTt (t=0..PAS_LVL_N-1) | - - |
| RMRO + 0330H + 4*t | MAC automatic PFC frame transmit counter t (t=0..PAS_LVL_N-1) | MAPCFTCTt (t=0..PAS_LVL_N-1) | - - |
| RMRO + 0340H + 4*t | MAC PFC frame receive counter t (t=0..FRM_PRIO_N -1) | MPCFRCTt (t=0..FRM_PRIO_N-1) | - - |
| RMRO + 0360H | Receive overflow Counter | MROVFC | - - |
| RMRO + 0364H | Receive Header-CRC (PCH-CRC) error Counter | MRHCRCEC | - - |
| RMRO + 0408H | Received good frame counter E-frames | MRGFCE | - - |
| RMRO + 040CH | Received good frame counter P-frames | MRGFCP | - - |
| RMRO + 0410H | Received good broadcast frame counter | MRBFC | - - |
| RMRO + 0414H | Received good multicast frame counter | MRMFC | - - |
| RMRO + 0418H | Received good unicast frame counter | MRUFC | - - |
| RMRO + 041CH | Received PHY error frame counter | MRPEFC | - - |
| RMRO + 0420H | Received nibble error frame counter | MRNEFC | - - |
| RMRO + 0424H | Received FCS/mCRC error frame counter | MRFMEFC | - - |
| RMRO + 0428H | Received final fragment missing error frame count | MRFFMEFC | - - |
| RMRO + 042CH | Received C-fragment count error frame counter | MRCFCEFC | - - |
| RMRO + 0430H | Received fragment count error frame counter | MRFCEFC | - - |
| RMRO + 0434H | Received RMAC filter error frame counter | MRRCFEFC | - - |
| RMRO + 0438H | Received frame counter | MRFC | - - |
| RMRO + 043CH | Received good undersize error frame counter | MRGUEFC | - - |
| RMRO + 0440H | Received bad undersize error frame counter | MRBUEFC | - - |
| RMRO + 0444H | Received good oversize error frame counter | MRGOEFC | - - |
| RMRO + 0448H | Received bad oversize error frame counter | MRBOEFC | - - |
| RMRO + 044CH | Received byte counter E-frames Upper | MRXBCEU | - - |
| RMRO + 0450H | Received byte counter E-frames Lower | MRXBCEL | - - |
| RMRO + 0454H | Received byte counter P-frames Upper | MRXBCPU | - - |
| RMRO + 0458H | Received byte counter P-frames Lower | MRXBCPL | - - |
| RMRO + 0508H | Transmitted good frame counter E-frames | MTGFCE | - - |
| RMRO + 050CH | Transmitted good frame counter P-frames | MTGFCP | - - |
| RMRO + 0510H | Transmitted broadcast frame counter | MTBFC | - - |

| Offset/Address | Register name | Abbreviation | Write Access Mode |
|---|---|---|---|
| RMRO + 0514H | Transmitted multicast frame counter | MTMFC | - - |
| RMRO + 0518H | Transmitted unicast frame counter | MTUFC | - - |
| RMRO + 051CH | Transmitted error frame counter | MTEFC | - - |
| RMRO + 0520H | Transmitted byte counter E-frames Upper | MTXBCEU | - - |
| RMRO + 0524H | Transmitted byte counter E-frames Lower | MTXBCEL | - - |
| RMRO + 0528H | Transmitted byte counter P-frames Upper | MTXBCPU | - - |
| RMRO + 052CH | Transmitted byte counter P-frames Lower | MTXBCPL | - - |
| RMRO + 0530H + 4*i | PHY Broken Link Transmitted Frame Counter E-frame per Source Port i ( i=0..PORT_N-1) | MPBLTFCESPi | - - |
| RMRO + 0570H + 4*i | PHY Broken Link Transmitted Frame Counter P-frame per Source Port i ( i=0..PORT_N-1) | MPBLTFCPSPi | - - |
| RMRO + 05B0H | PHY Broken Link Transmitted Frame Counter E-frame | MPBLTFCE | - - |
| RMRO + 05B4H | PHY Broken Link Transmitted Frame Counter P-frame | MPBLTFCP | - - |
| RMRO + 0600H - RMRO + 08B4H | Monitor of Counter register from RMRO + 0300H to RMRO + 05B4H All counters are RC, but monitors are RO. | ****M | - - |

## 4.2 Register detailed explanation

### 4.2.1 RMAC Function registers

#### 4.2.1.1 PHY configuration registers

(1) **MPSM**

MAC PHY station management.

Provides an access to the PHY-LSI internal registers via the MDIO.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PRD[15:0] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | POP[1:0] | | PRA[4:0] | | | | | PDA[4:0] | | | | | MFF | RSV | PSME |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | PRD | R!=W | 0H | PHY Register Data<br>These bits specify following value according to **MPSM.MFF.**<br>Refer to 8.5 for more details.<br>Value<br>- **MPSM.MFF**=0 : data field of the PHY register<br>- **MPSM.MFF**=1 : data/address filed of the PHY register<br>Function<br>- write operation: HW send register value to PHY in write operation.<br>- read operation: HW write read data to this register from PHY in read operation. |
| 15 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 14:13 | POP | RW | 0H | PHY OPeration code<br>These bits specify the operation code.<br>Refer to 8.5 for more details.<br>Value<br>These bits specify following value according to **MPSM.MFF.**<br>- **MPSM.MFF**=0<br>10 Read Frame<br>01 Write Frame<br>Others reserved<br>- **MPSM.MFF**=1<br>00 Address Frame<br>01 Write Frame<br>11 Read Frame<br>10 Post-read-increment-address frame<br>Restriction<br>- SW: Value 00 or 11 should not be set when **MPSM.MFF**=0. |
| 12:8 | PRA | RW | 0H | PHY Register Address<br>These bits specify following value according to **MPSM.MFF.**<br>Refer to 8.5 for more details.<br>Value<br>- **MPSM.MFF**=0: Register Address<br>- **MPSM.MFF**=1: device address |
| 7:3 | PDA | RW | 0H | PHY Device Address<br>These bits specify following value according to **MPSM.MFF.**<br>Refer to 8.5 for more details.<br>Value<br>- **MPSM.MFF**=0 : the PHY Address field<br>- **MPSM.MFF**=1 : port address filed |

| | | | | |
|---|---|---|---|---|
| 2 | MFF | RW | 0B | Management frame format<br>This bit selects the format of management frame.<br>Value<br>- 0: normal management frame format defined in Clause 22 of IEEE802.3<br>Describe “MDIO” in this document.<br>- 1: extension management frame format defined in Clause 45 of IEEE802.3<br>Describe “eMDIO” in this document. |
| 1 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 0 | PSME | R!=W | 0B | PHY Station Management Enable<br>Set this bit to access PHY registers.<br>Set condition:<br>- SW : By writing 1 to this register. It starts phy access.<br>Clear condition<br>- HW : When PHY register access is completed, this bit returns automatically to 0.<br>Restriction<br>- SW : Accessing to MPSM register while this bit is asserted has no effect.<br>- SW : This bit cannot be set when **MPIC.PSMCS** = 0.<br>- SW : Before resetting RMAC, this register should always be checked. If this register is set, no reset should be performed.<br>Value<br>- 0: PHY register access is not on progress.<br>- 1: PHY register access is requested and on progress. |

## (2) **MPIC**

MAC PHY interfaces configuration.

Configures MDIO and xMII interface.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PSMCT[3:0] | | | | PSMHT[3:0] | | | | PSMDP | PSMCS[6:0] | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PSMCS[9:7] | | | RSV | | PLSPP | PIPP | PIP | RSV | | LSC[2:0] | | | PIS[2:0] | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:28 | PSMCT | RW | 0H | PHY Station Management Capture time adjustment<br>The configured value will shift the data capture timing of MDI (rmc_mdi) with respect to the rising edge of MDC (rmc_mdc) clock.<br>Value<br>- 0x0: No adjusted capture time (i.e. capture MDI at the rising edge of MDC)<br>- 0x1: capture before 1 clk cycle<br>- :<br>- 0xF: capture before 15 clk cycle<br>Restriction<br>- SW : This register cannot be set if MPSM.PSME is set<br>- SW : The configured value must be less than MDC clock half cycle ( less than **MPIC.PSMCS**+1 ). |
| 27:24 | PSMHT | RW | 0H | PHY Station Management Hold time adjustment<br>The configured value will shift the data change timing of MDO (rmc_mdo and miimg_mde) with respect to the rising edge of MDC (rmc_mdc) clock.<br>Value:<br>- 0x0: No adjusted hold time (i.e. change MDO at the rising edge of MDC)<br>- 0x1: 1 clk cycle extra hold time<br>- :<br>- 0xF: 15 clk cycle extra hold time<br>Restriction<br>- SW : This register cannot be set if MPSM.PSME is set<br>- SW : The configured value must be less than MDC clock half cycle ( less than **MPIC.PSMCS**+1 ). |
| 23 | PSMDP | RW | 0B | PHY Station Management Disable Preamble<br>This bit could cut the preamble pattern (sequence of 32 contiguous one bits). When it is asserted, the preamble pattern is skipped and the management frame starts with the ST (Start of Frame) pattern.<br>Restriction<br>- SW : This register cannot be set if **MPSM.PSME** is set<br>Value<br>- 0: PHY Station Management Preamble enabled<br>- 1: PHY Station Management Preamble disabled |
| 22:16 | PSMCS [6:0] | RW | 0H | PHY Station Management Clock Selection<br>These bits configure the cycle of MDC clock. MDC clock is divided from clk clock. The duty cycle of MDC clock is 50%. MDC clock have to max 2.5MHz.<br>Refer to section 8.4 for detail.<br>Restriction<br>- SW : This register cannot be set if MPSM.PSME is set<br>- HW : MAX<br>Value<br>- ‘d0: MDC clock is off<br>- ‘d1: clk / ((1+1)*2)<br>- ‘d2: clk / ((2+1)*2)<br>- :<br>- ‘d1022: clk / ((1022+1)*2)<br>- ‘d1023: clk / ((1023+1)*2) |
| 15:13 | PSMCS [9:7] | RW | 0H | PHY Station Management Clock Selection |
| 12:11 | RSV | R0 | 0H | - Reserved area. On read, 0 will be returned |

| | | | | |
|---|---|---|---|---|
| 10 | PLSPP | RW | 0B | PHY Link Status Pin Plugged<br>Value<br>- 0: Unplugged<br>- 1: Plugged |
| 9 | PIPP | RW | 0B | PHY Interrupt Pin Plugged<br>Value<br>- 0: Unplugged<br>- 1: Plugged |
| 8 | PIP | RW | 0B | PHY Interrupt Polarity<br>This bit sets the active level of the PHY interrupt pin (rmc_phy_int).<br>Value<br>- 0: PHY interrupt is active low (default)<br>- 1: PHY interrupt is active high<br>Note<br>- For the active level, refer to the specifications of PHY-LSI to be connected. |
| 7:6 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 5:3 | LSC | RW | 0H | Link Speed Configuration<br>Value<br>- 3'b000: 10mbps<br>- 3'b001: 100mbps<br>- 3'b010: 1gbps<br>- 3'b011: 2.5gbps (Case of more than 300MHz clk)<br>- 3'b100: 5gbps<br>- 3'b101: 10gbps<br>- 3'b110-111: reserved<br>Note<br>- Reserved value cannot write and forced to 3'b101 (10gbps)<br>- The value for LSC and PIS are only available shown in Table 4-2. |
| 2:0 | PIS | RW | 0H | PHY Interface Select<br>These bits select the xMII interface connected to the PHY.<br>Value<br>- 000: MII<br>- 001: Reserved<br>- 010: GMII<br>- 011: Reserved<br>- 100: XGMII<br>- 101-111: Reserved.<br>Note<br>- Reserved value cannot write and forced to 3'b100 (XGMII)<br>- The value for LSC and PIS are only available shown in Table 4-2. |

### (3) **MPIM**

MAC PHY interfaces monitoring register.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | LPIA | PLS |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:2 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 1 | LPIA | R | 0B | LPI active<br>This bit indicates that LPI is active on RX side<br>Value<br>- 0: No LPI<br>- 1: LPI active<br>Set condition<br>- HW : When LPI code is appeared on detected on xMII Rx data signals (rmc_rx_data_valid_in, rmc_rx_data_data_in[31:0],rmc_rx_data_err_in). Refer xMII specification for more delatils.<br>Clear condition<br>- HW : When LPI code is not appeared on detected on xMII Rx data signals. |
| 0 | PLS | R | 0B | PHY Link Status<br>This bit indicates PHY link state. It could be used only when the link status pin (rmc_phy_link) is connected to the link signal from PHY-LSI.<br>For the active level, refer to the specifications of PHY-LSI to be connected.<br>Value<br>- 0: PHY link signal is at the low level.<br>- 1: PHY link signal is at the high level. |

### (4) **MIOC**

RMAC IO configuration registers

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MIOC [31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MIOC [15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | MIOC | RW | 00000000H | RMAC IO Configuration register<br>Value<br>- Bit [31:6] : No function.<br>- Bit [5] : Force function for PHY LINK input<br>0 : No function.<br>1 : **PHY_LINK** set to 1'b1.<br>- Bit [4] : Interrupt function for TSN-ES[0-7] PHY INT output<br>This function is enabled only TSNA/RMAC[5-12]<br>0 : Output value 1'b0.<br>1 : Output value 1'b1.<br>- Bit [3] : Connection change PHY or TSN-ES<br>This function is enabled only TSNA/RMAC[5-7]<br>0 : Connected TSN-ES[0-2]<br>1 : Connected PHY<br>- Bit [2:0] : No function. Don't write 1'b1.<br>Restriction<br>- SW can write 1 to bit[0] when clock derivation from PHY is unexpectedly stopped. In this case, write 1 to bit[0] to flush all internal condition in RMAC. After waiting to flush and change DISABLE mode, write 0 to bit[0] and change RESET mode. . |

### 4.2.1.2 Transmission configuration registers

#### (1) MTFFC

MAC transmission frame format configuration.

The register is used to configure the frame format to be transmitted.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | FCM | DPAD |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 30:2 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 1 | FCM | RW | 0H | Flow Control Mode<br>This bit specifies the flow control mode<br>Value<br>- 0 : PAUSE<br>- 1 : PFC |
| 0 | DPAD | RW | 0B | Data Padding Disable<br>Disables the automatic padding of too short transmit frames (less than 60 bytes) with no FCS.<br>Value<br>- 0: Padding is inserted<br>- 1: Padding is not inserted<br>Note<br>- If a too short transmit frame (less than 60 bytes) with no FCS is received when this bit is set, RMAC adds FCS and sends such a frame.<br>Restriction<br>- HW : A Verify frame, a Pause frame and a PFC frame are not affected by this bit.<br>- HW : Padding is not executed when “FCS included(FI==1)” indication is applied by upper layer. But FI will be cleared by upper layer based on the formula: A & !(B & C).where,<br>A: Corresponding transmit frames as an FCS and if the frame hasn’t been modified by the switch.<br>B: MTTFC.DPAD == 1’b0 [RMAC].<br>C: “Payload length of frame (including FCS)” less than 64 bytes.<br>- SW : This feature is not enabled when transmitting using MACsec. (Padding of encrypted frames is not possible.) Please perform padding with SW. |

## (2) **MTPFC**

MAC transmission Pause or PFC frame configuration.

The register is used to configure the TIME parameter value for pause frames.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PFRLV[4:0] | | | | | PFM | RSV | | PFRT[7:0] | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PT[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:27 | PFRLV | RW | 0H | Pause or PFC frame retry limit value<br>Value of pause frame sent without pause frame request signal de-assertion from which at which RMAC will emit an interrupt<br>Value<br>- H'00: No interrupt<br>- H'01: Interrupt at 1 pause frame<br>:<br>- H'1F: Interrupt at 31 pause frames<br>Restriction<br>- SW: This register is only writable in CONFIG mode<br>Note<br>- This bit is common bit for PAUSE and PFC.<br>- The number of pause(pfc) frames in this field indicates the total number of transmission attempt. |
| 26 | PFM | RW | 0H | Pause frame mode:<br>Value<br>- 0: Automatic pause frame. Sending a pause frame will be triggered by a hardware pause from [COMA] request.<br>- 1: Manual pause frame. Sending a Pause frame will be triggered by MTPFC2.MPFR or MTPFC2. MPFCFR.<br>Restriction<br>- SW: This register is only writable in CONFIG mode<br>Note<br>- This bit is common bit for PAUSE and PFC. |
| 25: 24 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 23:16 | PFRT | RW | 0H | Pause frame retransmission time<br>These bits set the TIME parameter value of pause frame retransmission.<br>Function<br>- A pause frame will be retransmitted when the previous frame has been sent more than (**MTPFC.PT**-**MTPFC.PFRT**) ± 10 cycles bit-time ago. However, this retransmission is delayed by the current transmission of another frame.<br>Value<br>- The unit for the setting is 512 bit-time. 1 bit-time is equal to 1ns in 1000Mbps, 10ns in 100Mbps, 100ns in 10Mbps.<br>- H'00: 512 x 0 bit-time<br>- H'01: 512 × 1 bit-time<br>:<br>- H'FF: 512 × 255 bit-time<br>Restriction<br>- SW: This register should be set to a smaller value than one of **MTPFC.PT**.<br>- SW: This register is only writable in CONFIG mode<br>Note<br>- These bits are common bit for PAUSE and PFC.<br>- MTPFC.PFRT (Pause frame retransmission time) is applicable for the 2nd **AutoPFC/PAUSE** frame, when the PAUSE Request of the 1st frame is still HIGH (Asserted). |
| 15:0 | PT | RW | 0H | Pause Time<br>These bits set the TIME parameter value for pause frames.<br>Function<br>- The value of these bits is set to Time field of a PAUSE or PFC frame.<br>- If the value of these bits is 0, PAUSE of PFC is not sent. |

| | | | | |
|---|---|---|---|---|
| | | | | Value<br>- The unit for the setting is 512 bit-time. 1 bit-time is equal to 1ns in 1000Mbps, 10ns in 100Mbps, 100ns in 10Mbps.<br>- H'0000: Pause frame disable<br>- H'0001: 512 × 1 bit-time<br>:<br>- H'FFFF: 512 × 65,535 bit-time<br>Restriction<br>- SW: This register is only writable in CONFIG mode<br>Note<br>- These bits are common bit for PAUSE and PFC. |

## (3) **MTPFC2**

MAC transmission pause or PFC frame configuration2.

The register is used to set the TIME parameter value for PFC frames.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | MPFR | PFTTZ |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | MPFCFR [PAS_LVL_N-1:0] | | RSV | | | | | | | PFCTTZ [PAS_LVL_N-1:0] | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: 18 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 17 | MPFR | RW | 0B | Manual pause frame request<br>Function<br>- Writing 1'b1 to this bit requests a manual pause frame.<br>- If a PAUSE frame is requested so fast that the hardware didn't have time to send the previous one (because a long frame is ongoing or because the software requests too many PAUSE frames), the hardware will ignore the PAUSE frame requests.<br>Set condition<br>- SW: Writing 1 to this bit will set the bit<br>Clear condition<br>- SW: Writing 0 to this bit will clear the bit.<br>- HW: Going out of OPERATION mode will clear this register.<br>Restriction<br>- SW: This register is only writable in OPERATION mode<br>Note<br>- This bit is used for a PAUSE frame only. |
| 16 | PFTTZ | RW | 0B | Pause Frame Transmission with TIME = 0<br>Function<br>- This bit enables the transmission of automatic Pause frames with TIME value zero. It is transmitted when the MHD negates the pause request in automatic pause frame mode or when the software negates **MTPFC2.MPFR** register in manual pause frame mode and the remaining pause time is more than PFRT.<br>- This bit has no effect for Manual Pause frame transmission.<br>Value<br>- 0: The transmission of Pause frames with TIME value zero is disabled.<br>- 1: The transmission of Pause frames with TIME value zero is enabled<br>Restriction<br>- SW : This register is only writable in CONFIG mode<br>Note :<br>- This bit is used for PAUSE frame only. |
| 15: 8+PAS_LVL_N | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 8+ PAS_LVL_N -1:8 | MPFCFR | RW | 0H | Manual PFC frame request t<br>Function<br>- Writing 1'b1 to this bit request manual PFC frame.<br>- If a PFC frame is requested so fast that the hardware didn't have time to send the previous one (because a long frame is ongoing or because the software requests too many PFC frames), the hardware will ignore the PFC frame requests. After completion of non-PFC frame transmission, the next attempt for PFC frame will be only when timer expires although request for PFC frame is already disabled.<br>Set condition<br>- SW: Writing 1 to this bit will set.<br>Clear condition<br>- SW: Writing 0 to this bit will clear.<br>- HW: Going out of OPERATION mode will clear this register.<br>Restriction<br>- SW: This register is only writable in OPERATION mode |
| 7: PAS_LVL_N | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| PAS_LVL_N-1:0 | PFCTTZ | RW | 0H | PFC Frame Transmission with TIME = 0 |

| | | | | |
|---|---|---|---|---|
| | | | | Function<br>- This bit enables the transmission of automatic PFC frames with TIME value zero. It is transmitted when the MHD negated the PFC request in automatic PFC frame mode or when the software negate **MTPFC2.MPFCFR** register in manual PFC frame mode and the remaining PFC time is more than PFRT.<br>Restriction<br>- SW : This bit has no effect for Manual PFC frame transmission.<br>Value<br>- 0: The transmission of PFC frames with TIME value zero is disabled.<br>- 1: The transmission of PFC frames with TIME value zero is enabled<br>Restriction<br>- SW: This register is only writable in CONFIG mode |

### (4) **MTPFC3t (t=0:PAS_LVL_N-1)**

MAC transmission pause or PFC frame configuration 3 t.

The register is used to set the TIME parameter value for PFC frames.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | PFCPGt[FRM_PRIO_N-1:0] | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:8 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| FRM_PRIO_N-1:0 | PFCPGt | RW | 0H | PFC Priority Group t<br>Function when **MTFFC.FCM**=1(PFC)<br>- When PFC frame transmission request (hardware request for automatic request, MTPFC2.MPFCFR for manual request) is set, PFC frame is generated. Then, the Field of Class-Enable vector is set to this register value.<br>- When all bits of PFCPGt = 0, the transmission request is ignored.<br>Function when **MTFFC.FCM**=0 (PAUSE)<br>- PAUSE frame transmission request is set by hardware pause from [COMA] request or MTPFC2.MPFR.<br>- Please set bit[0] to 1'b1. When PFCPG0[0] = 0, the transmission request is ignored.<br>Value<br>- 0 : priority x is not assigned to priority group t<br>- 1 : priority x is assigned to group t<br>Restriction<br>- SW: This register is only writable in CONFIG mode |

### 4.2.1.3 Reception configuration registers

#### (1) MRGC

MAC reception general configuration

The resister is used to configure the frame reception.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | PFCRC [FRM_PRIO_N-1:0] | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | RFCFE | MPDE | PFRTZ | PFRC | RCPT |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:24 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 23:16 | PFCRC | RW | 0H | PFC Frame Reception Control<br>This bit enables the Flow Control (Pause frame detection) function.<br>Value<br>- 0: Flow control for the receiving priority level x(FRM_PRIO_N-1:0) is disabled.<br>- 1: Flow control for the receiving priority level x(FRM_PRIO_N-1:0) is enabled.<br>Restriction<br>- HW : Only for e-frames<br>- SW : When **MRGC.PFRC** is set, SW should not set this bit. |
| 15:5 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 4 | RFCFE | RW | 0B | Reception Flow Control Forwarding Enable<br>This bit enables the Flow Control Frame (PAUSE or PFC frame) forwarding function.<br>Value<br>- 0: Flow Control Frame (PAUSE or PFC frame) is not forward to Rx MHD I/F.<br>- 1: Flow Control Frame (PAUSE or PFC frame) is forward to Rx MHD I/F. |
| 3 | MPDE | RW | 0B | Magic Packet™ Detection Enable<br>This bit enables Magic Packet™ detection by hardware to allow activation via the Ethernet connection. (Only for e-frames)<br>Value<br>- 0: Magic Packet™ detection is disabled.<br>- 1: Magic Packet™ detection is enabled. |
| 2 | PFRTZ | RW | 0B | Pause or PFC Frame Reception with Time = 0<br>When this bit is set to 1, the reception of Pause or PFC frame<br>with the Timer value 0 releases the transmission wait state.<br>Value<br>- 0: Reception of Pause or PFC frames with the TIME value 0 is disabled.<br>- 1: Reception of Pause or PFC frames with the TIME value 0 is enabled. |
| 1 | PFRC | RW | 0B | Pause Frame Reception Control<br>This bit enables the Flow Control (Pause frame detection) function.<br>Value<br>- 0: Flow control for the receiving port is disabled.<br>- 1: Flow control for the receiving port is enabled.<br>Restriction<br>- HW : Only for e-frames<br>- SW : When any bits of **MRGC.PFCRC** is set, SW should not set this bit. |
| 0 | RCPT | RW | 0B | Receive CRC pass through<br>This bit selects to pass FCS field of the received frame. The CRC is checked regardless of this configuration.<br>Value<br>- 0: Correct FCS is not passed to the MHD. Incorrect FCS is passed to the MHD.<br>- 1: Both Correct and Incorrect FCS are passed to the MHD. |

### (2) **MRMAC0**

MAC reception MAC address configuration 0.

Specifies the 16 upper-order bits of the 48-bit unicast MAC address. This register is also used as source address filed when the IP transmits a Pause frame or a PFC frame

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MAUP[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 15:0 | MAUP | RW | 0H | MAC Address Upper Part<br>These bits are used to set the 16 upper-order bits of the MAC address.<br>For example, if the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'0123 in this register.<br>Restrictions:<br>- SW: This register must write before MAMAC1 writing. |

### (3) **MRMAC1**

MAC reception MAC address configuration 1.

The register is to use to specify the 32 lower-order bits of the 48-bit unicast MAC address. This register is also used as source address filed when the IP transmits a Pause frame or a PFC frame.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MADP[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MADP[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | MADP | RW | 0H | MAC Address Downer(Lower) Part<br>These bits are used to set the 32 lower-order bits of the MAC address.<br>For example, if the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'456789AB in this register.<br>Restrictions:<br>- SW: This register must write after MAMAC0 writing. |

## (4) **MRAFC**

MAC reception address filter configuration.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | MSAREP | NSAREP | SDSFREP | NDAREP | BCACP | MCACP | BSTENP | MSTENP | BCENP | MCENP | UCENP |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | MSAREE | NSAREE | SDSFREE | NDAREE | BCACE | MCACE | BSTENE | MSTENE | BCENE | MCENE | UCENE |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:27 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 26 | MSAREP | RW | 0B | Multicast source address reception enable p<br>This bit enables the reception of p-frame with multicast and broadcast source address<br>- 0: disable<br>- 1: enable |
| 25 | NSAREP | RW | 0B | Null source address reception enable p<br>This bit enables the reception of p-frames with a destination MAC address null.<br>- 0: disable<br>- 1: enable |
| 24 | SDSFREP | RW | 0B | Same DA and SA frames reception enable p<br>This bit enables the reception of p-frame with equal destination address and source address.<br>- 0: disable<br>- 1: enable |
| 23 | NDAREP | RW | 0B | Null destination address reception enable p<br>This bit enables the reception of p-frames with destination MAC address null.<br>- 0: disable<br>- 1: enable |
| 22 | BCACP | RW | 0B | Broadcast Storm AutoClear p frames<br>This bit enables auto-clear of broadcast frame storm counter. When it is asserted, broadcast frames storm counter is cleared by detecting a frame with non-broadcast destination address. After the counter is cleared, the IP can receive Broadcast frame again.<br>- 0: Broadcast Storm AutoClear disabled<br>- 1: Broadcast Storm AutoClear enabled |
| 21 | MCACP | RW | 0B | Multicast Storm AutoClear p frames<br>This bit enables auto-clear of multicast frame storm counter. When it is asserted, multicast frame storm counter is cleared by detecting a frame with non-multicast destination address. After the counter is cleared, the bit can receive multicast frame again.<br>- 0: Multicast Storm AutoClear disabled<br>- 1: Multicast Storm AutoClear enabled<br>Restriction<br>- HW : Broadcast frames are not taken in account by multicast storm counter. |
| 20 | BSTENP | RW | 0B | Broadcast storm filter reception enable p-frames<br>This bit enables the broadcast storm filtering. When it is asserted, Broadcast frames are rejected after the number of received broadcast p-frames reaches the configured threshold (**MRSCP.CBFP**).<br>- 0: Broadcast storm filter reception disabled<br>- 1: Broadcast storm filter reception enabled<br>Restriction<br>- SW : It is valid only when Broadcast reception is enabled (**MRAFC.BCENP**=1). |
| 19 | MSTENP | RW | 0B | Multicast storm filter reception enable p-frames<br>This bit enables the multicast storm filtering. When it is asserted, Multicast frames are rejected after the number ofreceived multicast p-frames reaches the configured threshold (**MRSCP.CMFP**).<br>- 0: Multicast storm filter reception disabled<br>- 1: Multicast storm filter reception enabled<br>Restriction<br>- SW : It is valid only when Multicast reception is enabled (**MRAFC.MCENP**=1). |
| 18 | BCENP | RW | 1B | Broadcast reception enable p-frames<br>This bit enables the broadcast filtering.<br>- 0: Broadcast reception disabled<br>- 1: Broadcast reception enabled |

| | | | | |
|---|---|---|---|---|
| 17 | MCENP | RW | 1B | Multicast reception enable p-frames<br>This bit enables the multicast filtering.<br>0: Multicast reception disabled<br>1: Multicast reception enabled |
| 16 | UCENP | RW | 1B | Unicast reception enable p-frames<br>This bit enables the unicast filtering based on MAC address (MRMAC0/1).<br>0: Unicast reception disabled<br>1: Unicast reception enabled |
| 15:11 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 10 | MSAREE | RW | 0B | Multicast source address reception enable e<br>This bit enables the reception of e-frame with multicast and broadcast source address<br>0: disable<br>1: enable |
| 9 | NSAREE | RW | 0B | Null source address reception enable e<br>This bit enables the reception of e-frames with destination MAC address null.<br>0: disable<br>1: enable |
| 8 | SDSFREE | RW | 0B | Same DA and SA frames reception enable e<br>This bit enables the reception of e-frame with equal destination address and source address.<br>0: disable<br>1: enable |
| 7 | NDAREE | RW | 0B | Null destination address reception enable e<br>This bit enables the reception of e-frames with destination MAC address null.<br>0: disable<br>1: enable |
| 6 | BCACE | RW | 0B | Broadcast Storm AutoClear e frames<br>This bit enables auto-clear of broadcast frame storm counter. When it is asserted, broadcast frames storm counter is cleared by detecting a frame with non-broadcast destination address. After the counter is cleared, the IP can receive Broadcast frame again.<br>- 0: Broadcast Storm AutoClear disabled<br>- 1: Broadcast Storm AutoClear enabled |
| 5 | MCACE | RW | 0B | Multicast Storm AutoClear e frames<br>This bit enables auto-clear of multicast frame storm counter. When it is asserted, multicast frame storm counter is cleared by detecting a frame with non-multicast destination address. After the counter is cleared, the IP can receive multicast frame again.<br>- 0: Multicast Storm AutoClear disabled<br>- 1: Multicast Storm AutoClear enabled<br>Restriction<br>- HW : Broadcast frames are not taken in account by multicast storm counter. |
| 4 | BSTENE | RW | 0B | Broadcast storm filter reception enable e-frames<br>This bit enables the broadcast storm filtering. When it is asserted, Broadcast frames are rejected after the number of received broadcast e-frames reaches to the configured threshold (MRSCE.CBFE).<br>- 0: Broadcast storm filter reception disabled<br>- 1: Broadcast storm filter reception enabled<br>Restriction<br>- SW : It is valid only when Broadcast reception is enabled (**MRAFC.BCENE**=1). |
| 3 | MSTENE | RW | 0B | Multicast storm filter reception enable e-frames<br>This bit enables the multicast storm filtering. When it is asserted, Multicast frames sre rejected after the number of received multicast e-frames reaches the configured threshold (MRSCE.CMFE).<br>- 0: Multicast storm filter reception disabled<br>- 1: Multicast storm filter reception enabled<br>Restriction<br>- SW : It is valid only when Multicast reception is enabled (**MRAFC.MCENE**=1). |
| 2 | BCENE | RW | 1B | Broadcast reception enable e-frames<br>This bit enables the broadcast filtering.<br>0: Broadcast reception disabled<br>1: Broadcast reception enabled |
| 1 | MCENE | RW | 1B | Multicast reception enable e-frames<br>This bit enables the multicast filtering.<br>0: Multicast reception disabled<br>1: Multicast reception enabled |
| 0 | UCENE | RW | 1B | Unicast reception enable e-frames<br>This bit enables the unicast filtering based on MAC address (MRMAC0/1).<br>0: Unicast reception disabled<br>1: Unicast reception enabled |

## (5) **MRSCE**

MAC reception storm configuration for e-frames

The register is to use to configure the number of times how many broadcast/multicast frames can be received consecutively.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CBFE[15:0] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CMFE[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | CBFE | RW | 0H | Consecutive Broadcast Frame Reception Count Setting for e-frames<br>These bits specify the upper-limitation of received consecutive broadcast frames . After the number of received consecutive broadcast frames reaches the set value, the subsequent broadcast frames are discarded.<br>H'0000: 1 Broadcast frame accepted until broadcast frame counter reset.<br>H'0001: 2 Broadcast frame accepted until broadcast frame counter reset<br>: :<br>H'FFFE: 65535 Broadcast frames accepted until broadcast frame counter reset<br>H'FFFF: Reserved |
| 15:0 | CMFE | RW | 0H | Consecutive Multicast Frame Reception Count Setting for e-frames<br>These bits specify the upper-limitation of received consecutive multicast frames. After the number of received consecutive multicast frames reaches the set value, the subsequent multicast frames are discarded.<br>H'0000: 1 Multicast frame accepted until multicast frame counter reset.<br>H'0001: 2 Multicast frame accepted until multicast frame counter reset.<br>: :<br>H'FFFE: 65535 Multicast frames accepted until frame counter reset.<br>H'FFFF: Reserved |

## (6) **MRSCP**

MAC reception storm configuration for p-frames

Configures the number of times how many broadcast/multicast frames can be received consecutively.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CBFP[15:0] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CMFP[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | CBFP | RW | 0H | Consecutive Broadcast Frame Reception Count Setting for p-frames<br>These bits specify the upper-limitation of received consecutive Broadcast frames. After the number of received consecutive broadcast frames reaches the set value, the subsequent broadcast frames are discarded.<br>H'0000: 1 Broadcast frame accepted until broadcast frame counter reset.<br>H'0001: 2 Broadcast frame accepted until broadcast frame counter reset.<br>: :<br>H'FFFE: 65535 Broadcast frames accepted between two broadcast frame counter resets.<br>H'FFFF: Reserved |
| 15:0 | CMFP | RW | 0H | Consecutive Multicast Frame Reception Count Setting for p-frames<br>These bits specify the upper-limitation of received consecutive consecutive multicast frames. After the number of received consecutive multicast frames reaches the set value, the subsequent multicast frames are discarded.<br>H'0000: 1 Multicast frame accepted until multicast frame counter reset.<br>H'0001: 2 Multicast frame accepted until multicast frame counter reset.<br>: :<br>H'FFFE: 65535 Multicast frames accepted until multicast frame counter reset.<br>H'FFFF: Reserved |

## (7) **MRSCC**

MAC reception storm counter configuration

Register to clear Broadcast and Multicast Storm Counters.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | BSCCP | MSCCP |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | BSCCE | MSCCE |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:18 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 17 | BSCCP | R0W | 0B | Broadcast Storm Counter Clear p-frames<br>This bit requests to clear the broadcast frames storm counter.<br>After the counter is cleared, the IP can receive broadcast frames again until the upper-limitation of received consecutive broadcast frames (MRSCP.CBFP).<br>It should be used with Broadcast Storm AutoClear disabled (MRAFC.BCACP=0).<br>Writing 1 to this bit clears the broadcast storm counter |
| 16 | MSCCP | R0W | 0B | Multicast Storm Counter Clear p-frames<br>This bit requests to clear the multicast frames storm counter.<br>After the counter is cleared,the IPt can receive multicast frames again until the upper-limitation of received consecutive multicast frames (MRSCP.CMFP).<br>It should be used with multicast Storm AutoClear disabled (MRAFC.MCACP=0).<br>Writing 1 to this bit clears the multicast storm counter |
| 15:2 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 1 | BSCCE | R0W | 0B | Broadcast Storm Counter Clear e-frames<br>This bit requests to clear the broadcast frames storm counter.<br>After the counter is cleared, the IP can receive Broadcast frames again until the upper-limitation of received consecutive Broadcast frames (MRSCE.CBFE).<br>It should be used with Broadcast Storm AutoClear disabled (MRAFC.BCACE=0).<br>Writing 1 to this bit clear the broadcast storm counter |
| 0 | MSCCE | R0W | 0B | Multicast Storm Counter Clear for e-frames<br>This bit requests to clear the multicast frames storm counter.<br>After the counter is cleared, the IP can receive multicast frames again until the upper-limitation of received consecutive multicast frames (MRSCE.CMFE).<br>It should be used with multicast Storm AutoClear disabled (MRAFC.MCACE=0).<br>Writing 1 to this bit clear the multicast storm counter |

## (8) **MRFSCE**

MAC reception frame size configuration for e-frames

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EMNS | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EMXS | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | EMNS | RW | 0H | E-frame minimum size |
| 15:0 | EMXS | RW | FFFFH<br>(65,535) | E-frame maximum size<br>a part of received frame which exceeds the set value are truncated.<br>Don’t set a value under 16 bytes |

Note:

- The size limitations concern the frame size RMAC outputs to an upper layer (e.g. MHD), and so the FCS removal impacts the register values.

## (9) **MRFSCP**

MAC reception frame size configuration for p-frames

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PMNS | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PMXS | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | PMNS | RW | 0H | P-frame minimum size |
| 15:0 | PMXS | RW | FFFFH | P-frame maximum size<br>. A part of received frame which exceeds the set value are truncated.<br>Don’t set a value under 16 bytes |

Note:

- The size limitations concern the frame size RMAC outputs to an upper layer (e.g. MHD), and so the FCS removal impacts the register values.

## (10) **MTRC**

MAC Timestamp reception configuration register

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | DTN[PTP_TN_W:0] | TCTSP | TCTSE | TRDDP | TRDDE | RSV | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | TRHFME[PTP_TN-1:0] | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:PTP_TN_W+28 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| PTP_TN_W+27:28 | DTN | RW | 0H | Default timer number<br>The bit set Timer used for timestamp forwarding on RX MHD I/F when no timer is detected. |
| 27 | TCTSP | RW | 0B | Timestamp capture on TX side P-frame<br>The bit enables the timestamp capture on TX side for p-frames<br>0: disable<br>1: enable |
| 26 | TCTSE | RW | 0B | Timestamp capture on TX side E-frame<br>The bit enables the timestamp capture on TX side for e-frames<br>0: disable<br>1: enable |
| 25 | TRDDP | RW | 0B | Timestamp reception default disable P-frame<br>A timestamp for every frame will be transmitted to the MHD. This timestamp will be taken on the default timer.<br>0: enable<br>1: disable |
| 24 | TRDDE | RW | 0B | Timestamp reception default disable E-frame<br>A timestamp for every frame will be transmitted to the MHD. This timestamp will be taken on the default timer except when TRHFME is set and the hardware filter matches.<br>0: enable<br>1: disable |
| 23:PTP_TN | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| PTP_TN-1:0 | TRHFME | RW | 0H | Timestamp reception hardware filter match enable<br>Setting bit i to 1'b1 in this register will enable hardware filtering for timer i.<br>When bit i is set to 1'b1, at least one of the bit i in MPFCt registers should be set to one.<br>When hardware filter matches, the timestamp from the matched timer will be sent along with the frame.<br>0: disable<br>1: enable |

## (11) **MRPFM**

MAC reception Pause or PFC frame monitoring register.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | PFCTCA [FRM_PRIO_N-1:0] | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | | PTCA |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:24 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| [16+FRM_PRIO_N-1:16] | PFCTCA | R | 0B | PFC time Counting Active<br>This bit indicates that the Pause time of a received PFC frame has not been completed.<br>0: No pause state<br>1: Pause state |
| 15:1 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 0 | PTCA | R | 0B | Pause time Counting Active<br>This bit indicates that the Pause time of a received Pause frame has not been completed.<br>0: No pause state<br>1: Pause state |

### 4.2.1.4 PTP configuration registers

#### (1) MPFCt(t=0.. PTP_TN*8-1)

MAC PTP filtering register configuration t

The register is used to configure the PTP frame filtering for time domain selection.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | TEFt[PTP_TN-1:0] | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PFBVt[7:0] | | | | | | | | PFBNt[7:0] | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:PTP_TN+16 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| PTP_TN+15:16 | TEFt | RW | 0H | Timer enable for filtering<br>Setting bit i to 1'b1 in this register will apply the filter to timer i. |
| 15:8 | PFBVt | RW | 0H | PTP filtering byte value<br>When the filtered byte has a value equal to this register, the byte will be eligible for the corresponding timers |
| 7:0 | PFBNt | RW | 0H | PTP filtering byte number<br>0: Byte number 0 will be filtered<br>: :<br>255: Byte number 255 will be filtered |

Note:
- This register is used for RX time domain selection
- When an overlapping configuration is done, HW will always choose the timer with the smaller number.

### 4.2.1.5 Link verification configuration registers

#### (1) MLVC

MAC link verification configuration.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | PLV |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | PASE | RSV | LVT[6:0] | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:17 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 16 | PLV | R!=W | 0B | Preemption Link Verification<br>By asserting this bit, link verification is requested.<br>This bit requests the Link Verification process. When link verification is completed, this bit is cleared.<br>When Link Verification succeeded, Preemption status bit (MMIS0.LVSS) are set. While it failed, the status bit (MMIS0.LVFS) is set.<br>0: Preemption Link Verification not requested<br>1: Preemption Link Verification requested<br>Going out of OPERATION mode will clear this register.<br><br>NOTE:<br>As described in 7.2.2, this bit should only be used in initialization.<br>SW set this bit when another transmission is not requested. |
| 15:9 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 8 | PASE | RW | 0B | Preemption auto response enable<br>This bit enables the automatic response to verify frames. |
| 7 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 6:0 | LVT | RW | 9H | Link Verification Timer<br>These bits define the nominal wait time between verification attempts in unit of milliseconds. It means the time of retransmitting SMD-V frame.<br>000 0000: 1ms wait time<br>000 0001: 2ms wait time<br>:<br>000 1001: 10ms wait time (default specified by IEEE 802.3)<br>:<br>111 1111: 128ms wait time |

Note:

- When PLV is set to 1, do not make write access continuously. Please wait completion of Link Verification Procedure for re-write access.

### 4.2.1.6 Energy efficient ethernet configuration registers

#### (1) MEEEC

MAC energy efficient ethernet configuration

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | | LPITR |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:1 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 0 | LPITR | RW | 0B | LPI Transmit Request<br>This bit requests to enter Low Power Idle (LPI) mode. RMAC enters to LPI state after all ongoing transmissions have been completed. When this bit is negated, this IP leaves LPI state.<br>0: LPI mode is not requested<br>1: LPI mode is requested<br>LPI signal should be transmitted only when TX is disabled.<br><br>Restriction<br>SW: LPITR bit can only be set when transmission request to RMAC is stopped,<br>and transmission request can only be allowed after LPITR is cleared.<br>It is prohibited to request transmission from HMD when LPITR bit is set.<br><br>Clear conditions:<br>- HW: Being in RESET mode will clear this register. |

### 4.2.1.7 Loopback register

(1) **MLBC**

MAC Loopback configuration

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | | LBME |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:1 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 0 | LBME | RW | 0B | Loopback mode enable<br>0: disable<br>1: enable<br>Transmit xMII signals are connected to reception xMII signals internally.<br><br>Restriction<br>- When loopback mode is enabled, frame received by external PHY Interface are ignored.<br>- If this bit is set, clk_phy_tx and clk_phy_rx must be synchronized with each other. |

### 4.2.1.8 XGMII Control

#### (1) MXGMIIC

XGMII configuration

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | | LFS_TXIDLE | LFS_TXRFS |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:2 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 1 | LFS_TXIDLE | RW | 0H | IDLE code transmission request<br>Set condition<br>- SW : Writing 1 to this bit will set the bit.<br>Clear condition<br>- SW : Writing 0 to this bit will clear it.<br>Function<br>- When this bit is set, IDLE code is transmitted<br>Restriction<br>- SW : this bit is only available when MPIC.PIS = 3'b100 (XGMII) |
| 0 | LFS_TXRFS | RW | 0H | Remote Fault signal transmission request<br>Set condition<br>- SW : Writing 1 to this bit will set the bit.<br>Clear condition<br>- SW : Writing 0 to this bit will clear it.<br>Function<br>- When this bit is set, Remote Fault orders set is transmitted repeatedly.<br>Restriction<br>- SW : this bit is only available when MPIC.PIS = 3'b100 (XGMII) |

#### (2) MPCH

XGMII PCH configuration

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | RPHCRCD | RXPCH_TSM |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | CTRIOD | IETRIOD | CTPTE | IETPTE | TXPCH_PID[3:0] | | | | TXPCH_ETYPE [1:0] | | RSV | TXPCH_M |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:18 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 17 | RPHCRCD | RW | 0H | Rx PCH HCRC Error Disable<br>This bit selects the internal operation about Rx PCH HCRC check.<br>Value<br>- 0: Enable. RMAC check Rx PCH HCRC.<br>- 1: Disable. RMAC does not check Rx PCH HCRC. HCRC is ignored. |
| 16 | RXPCH_TSM | RW | 0H | RX PCH Timestamp Mode<br>RMAC generates receive timestamp according to this register when ethernet frame is received.<br>Value<br>- 0 : Disable. Rx Timestamp is generated by RMAC.<br>- 1 : Enable. Rx Timestamp is extracted from RxPCH. |

| | | | | |
|---|---|---|---|---|
| 15:12 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 11 | CTRIOD | RW | 0H | CT for RMAC Internal Operation Disable<br>This bit selects the internal operation about CT.<br>Value<br>- 0: Timestamp capturing operation is enable<br>- 1: Timestamp capturing operation is disable.<br>In this case, timestamp is captured internally but captured timestamp is not outputted by rmc_tx_time_capture_req signal evenwhen CT==1. |
| 10 | IETRIOD | RW | 0H | IET for RMAC Internal Operation Disable<br>This bit selects the internal operation about IET.<br>Value<br>- 0: Timestamp/ResidenceTime insertion operation is enable<br>- 1: Timestamp/ResidenceTime insertion operation is disable.<br>In this case insertion operation is not executed even when IET==1.<br>Restriction<br>- This bit is only available for e-frame. |
| 9 | CTPTE | RW | 0H | CT for PCH Timestamp Enable<br>This bit enables to set the value of Tx PCH bit31 (PTPTimestamp enable bit) by CT.<br>Value<br>- 0: CT value is not used for generating Tx PCH bit31.<br>- 1: If CT==1 set Tx PCH bit31 to 1. |
| 8 | IETPTE | RW | 0H | IET for PCH Timestamp Enable<br>This bit enables to set the value of Tx PCH bit31 (PTPTimestamp enable bit) by IET.<br>Value<br>- 0: IET value is not used for generating Tx PCH bit31.<br>- 1: If IET==1 set Tx PCH bit31 to 1.<br>Restriction<br>- This bit is only available for e-frame. |
| 7:4 | TXPCH_PID | RW | 0H | TXPCH sub Port ID [3:0]<br>Function<br>This register value is used to Port ID [3:0] of PCH data structure. |
| 3:2 | TXPCH_ETYPE | RW | 0H | TXPCH Extension Type<br>Value<br>- 00 : e-frame is sent by Extension Type = 00<br>- 01 : e-frame is sent by Extension Type = 01<br>- 10 : e-frame is sent by Extension Type = 10<br>- 11 : reserved.<br>Restriction<br>- SW : This SFR is only valid when MPCH.TXPCH_M = 1(PCH).<br>- HW : p-frame is sent by Extension Type = 10 when MPCH.TXPCH_M = 1 regardless this SFR setting.<br>- SW : 2'b00 is selected when SW set 2'b11 to this register.<br>- |
| 1 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 0 | TXPCH_M | RW | 0H | TX PCH Mode<br>Value<br>- 0 : Format A (preamble)<br>- 1 : Format B (PCH)<br>Function<br>- RMAC selects preamble format of transmission frame. |

* If 1st byte of received frame is 8'hfb, RMAC decode first 8 bytes as a preamble.

- RMAC decode USXGMII TYPE A preamble (Normal preamble but 1st byte is 8'hfb) if pch[47:46] (frame type) is 2'b01.

In this case, if 8th byte is not SMDx, RMAC ignore this frame.

- RMAC decode USXGMII TYPE B preamble (PCH)) if pch[47:46] (frame type) is not 2'b01.

If 1st byte of received frame is not 8'hfb, RMAC wait valid SMDx symbol is detected.

### (3) **MANM**

Auto-negotiation message

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RX_AN_MES | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:16 | RSV | R0 | 0H | - Reserved area. On read, 0 will be returned |
| 15:0 | RX_AN_MES | R | 0H | Rx Auto-nego message<br>This register is used to control Auto-negotiation described in USXGMII specification.<br>Function<br>- HW: When Auto-nego message defined in USXGMII specification received, Config[15:0] is written in this register. |

### 4.2.2 RMAC Counter registers

Counter registers
Note:
- All counter registers in RMAC are read clear registers.
- When counters reach their own maximum value, they stop.

#### (1) MMPFTCT

MAC manual pause frame transmit counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MPFTC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | MPFTC | RC | 0H | Manual Pause frame Transmit Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when manual Pause frame is transmitted. |

## (2) **MAPFTCT**

MAC automatic pause frame transmit counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| APFTC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | APFTC | RC | 0H | Automatic pause Frame Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when automatic Pause frame is transmitted. |

## (3) **MPFRCT**

MAC pause frame receive counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PFRC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | PFRC | RC | 0H | Pause Frame Receive Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when automatic Pause frame is received.<br>Restriction<br>- SW: Set condition is only available when Flow Control is enabled in reception (**MRGC.PFRC** = 1'b1). |

## (4) **MFCICT**

MAC false carrier indication counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| FCIC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | FCIC | RC | 0H | False Carrier Indication Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when False Carrier Indication pattern is detected. |

Note:

- This register is counter of “False Carrier Indication” event. This event goes through the synchronizer between clk and clk_phy_rx. If the event occurs continuously, there is a possibility of miss counting.

## (5) **MEEECT**

MAC energy efficient ethernet counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EEERC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | EEERC | RC | 0H | Energy Efficient Ethernet Receive Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when LPI signal is detected on xMII reception bus. |

## (6) **MMPCFTCTt (t=0..PAS_LVL_N-1)**

MAC manual PFC frame transmit counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MPCFCTC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | MPCFCTC | RC | 0H | Manual PFC frame Transmit Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when manual PFC frames is transmitted. |

## (7) **MAPCFTCTt (t=0..PAS_LVL_N-1)**

MAC automatic PFC frame transmit counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| APCFCTC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | APCFCTC | RC | 0H | Automatic PFC Frame Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when automatic PFC frames is transmitted. |

## (8) **MPCFRCTt (t=0..FRM_PRIO_N-1)**

MAC PFC frame receive counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PCFCRC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | PCFCRC | RC | 0H | PFC Frame Receive Counter<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when PFC frames is received.<br>Restriction<br>- SW: Set condition is only available when Flow Control is enabled in reception (**MRGC.PFRC** = 1'b1). |

## (9) **MROVFC**

Receive overflow Counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ROVFC[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ MED_W-1:0 | ROVFC | RC | 0H | Receive overflow counter<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when frame receive overflow occurred ( when set condition for MEIS.REOES or MEIS.RPOES are met). |

## (10) **MRHCRCEC**

Reception Header-CRC(PCH CRC) error Counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RHCRCEC[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_ LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ LOW_W-1:0 | RHCRCEC | RC | 0H | Receive Header-CRC (Receive PCH CRC) Error Counter<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- HW: Incremented by 1 when receive HCRC (Receive PCH CRC) Error occurred. |

## (11) **MRXBCEU**

RMAC Received byte counter E-frames upper side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RBNEU[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBNEU[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | RBNEU | RC | 0H | Received byte number E-frames upper side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented each time the received data of e-Frame is sent to the RX MHD I/F regardless error is detected.<br>Note:<br>- When the RMAC does not pass the FCS to the upper layer, FCS is not taken in account in total byte calculation.<br>- Read before reading MRXBCEL. |

## (12) **MRXBCEL**

RMAC Received byte counter E-frames lower side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RBNEL[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBNEL[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | RBNEL | RC | 0H | Received byte number E-frames lower side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: None. (Reading MRXBCEL register holds it.)<br>Increment condition<br>- Incremented each time the received data of e-frame is sent to the RX MHD I/F regardless the error detection.<br>Note:<br>- When the RMAC does not pass the FCS to the upper layer, FCS is not taken in account in total byte calculation.<br>- Read after reading MRXBCEU. |

## (13) **MRXBCPU**

RMAC Received byte counter P-frames upper side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RBNPU[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBNPU[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | RBNPU | RC | 0H | Received byte number P-frames upper side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented each time the received data of p-frame is sent to the RX MHD I/F regardless the error detection.<br>Note:<br>- If the RMAC does not pass the FCS to the upper layer, FCS is not taken in account in total byte calculation.<br>- Read before reading MRXBCPL |

## (14) **MRXBCPL**

RMAC Received byte counter P-frames lower side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RBNPL[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBNPL[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | RBNPL | RC | 0H | Received byte number P-frames lower side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: None. (Reading MRXBCPL register holds it.)<br>Increment condition<br>- Incremented each time the received data of p-frame is sent to the RX MHD I/F regardless the error detection.<br>Note:<br>- When the RMAC does not pass the FCS to the upper layer, FCS is not taken in account in total byte calculation.<br>- Read after reading MRXBCPU. |

## (15) **MRGFCE**

RMAC Received good frame counter E-frames

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RGFNE[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | RGFNE | RC | 0H | Received good frame number E-frames<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when the reception of e-frame is completed without error. |

## (16) **MRGFCP**

RMAC Received good frame counter P-frames

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RGFNP[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | RGFNP | RC | 0H | Received good frame number P-frames<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when the reception of p-frame is completed without error. |

## (17) **MRBFC**

RMAC Received good broadcast frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | RBFN | RC | 0H | Received good broadcast frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when destination address filed of broadcast frame is sent to RX MHD I/F without error. |

## (18) **MRMFC**

RMAC Received good multicast frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RMFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | RMFN | RC | 0H | Received good multicast frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when destination address filed of multicast frame is sent to RX MHD I/F without error. |

## (19) **MRUFC**

RMAC Received good unicast frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RUFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | RUFN | RC | 0H | Received good unicast frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when destination address filed of unicast frame is sent to RX MHD I/F without error. |

## (20) **MRPEFC**

RMAC Received PHY error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RPEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | RPEFN | RC | 0H | Received PHY error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with phy error. |

## (21) **MRNEFC**

RMAC Received nibble error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RNEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | RNEFN | RC | 0H | Received nibble error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with nibble error. |

## (22) **MRFMEFC**

RMAC Received FCS/mCRC error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RFMEFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | RFMEFN | RC | 0H | Received FCS/mCRC error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with FCS/mCRC error. |

## (23) **MRFFMEFC**

RMAC final fragment missing error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RFFMEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | RFFMEFN | RC | 0H | Received final fragment missing error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with final fragment error. |

## (24) **MRCFCEFC**

RMAC Received C-fragment count error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RCFCEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | RCFCEFN | RC | 0H | Received C-fragment count error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with C-fragment count error. |

## (25) **MRFCEFC**

RMAC Received fragment count error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RFCEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | RFCEFN | RC | 0H | Received fragment count error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with fragment count error. |

## (26) **MRRCFEFC**

RMAC Received RMAC filter error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RRCFEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | RRCFEFN | RC | 0H | Received RMAC filter error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with RMAC filtered. |

## (27) **MRFC**

RMAC Received frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | RFN | RC | 0H | Received frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received frame is sent to RX MHD I/F. |

## (28) **MRGUEFC**

RMAC Received good undersize error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RGUEFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W<br>-1:0 | RGUEFN | RC | 0H | Received good undersize error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with undersize error but without fcs/mcrc error.<br>Restriction:<br>- Pause/PFC frames are out of scope. |

## (29) **MRBUEFC**

RMAC Received bad undersize error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBUEFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W<br>-1:0 | RBUEFN | RC | 0H | Received bad undersize error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when received data is sent to RX MHD I/F with undersize error and fcs/mcrc error.<br>Restriction:<br>- Pause/PFC frames are out of scope. |

## (30) **MRGOEFC**

RMAC Received good oversize error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RGOEFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | RGOEFN | RC | 0H | Received good oversize error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with oversize error, but without fcs/mcrc error.<br>Restriction:<br>- Pause/PFC frames are out of scope. |

## (31) **MRBOEFC**

RMAC Received bad oversize error frame counter.

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RBOEFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | RBOEFN | RC | 0H | Received bad oversize error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Incremented when received data is sent to RX MHD I/F with oversize error and fcs/mcrc error.<br>Restriction:<br>- Pause/PFC frames are out of scope. |

## (32) **MTXBCEU**

RMAC Transmitted byte counter E-frames upper side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TBNEU[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TBNEU[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | TBNEU | RC | 0H | Transmitted byte number E-frames upper side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment every time transmit data for e-frame is received on TX MHD I/F.<br>- Read before reading MTXBCEL<br>Notes:<br>- Padding bytes are not applied (counted) to tx-byte-count statistic counters. |

Note:
- If the FCS is not included from the upper layer, FCS is not taken in account in total byte calculation.

## (33) **MTXBCEL**

RMAC Transmitted byte counter E-frames lower side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TBNEL[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TBNEL[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | TBNEL | RC | 0H | Transmitted byte number E-frames lower side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: None. (Reading MTXBCEL register holds it.)<br>Increment condition<br>- Increment every time transmit data for e-frame is received on TX MHD I/F.<br>- Read after reading MTXBCEU.<br>Notes:<br>- Padding bytes are not applied (counted) to tx-byte-count statistic counters. |

Note:
- If the FCS is not included from the upper layer FCS is not taken in account in total byte calculation.

## (34) **MTXBCPU**

RMAC Transmitted byte counter P-frames upper side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TBNPU[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TBNPU[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | TBNPU | RC | 0H | Transmitted byte number P-frames upper side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment every time transmit data for p-frame is received on TX MHD I/F.<br>- Read before reading MTXBCPL<br>Notes:<br>- Padding bytes are not applied (counted) to tx-byte-count statistic counters. |

Note:

- If the FCS is not included from the upper layer, FCS is not taken in account in total byte calculation.

## (35) **MTXBCPL**

RMAC Transmitted byte counter P-frames lower side

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TBNPL[31:16] | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TBNPL[15:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:0 | TBNPL | RC | 0H | Transmitted byte number P-frames lower side<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: None. (Reading MTXBCPL register holds it.)<br>Increment condition<br>- Increment every time transmit data for p-frame is received on TX MHD I/F.<br>- Read after reading MTXBCPU.<br>Notes:<br>- Padding bytes are not applied (counted) to tx-byte-count statistic counters. |

Note:

- If the FCS is not included from the upper layer, FCS is not taken in account in total byte calculation.

## (36) **MTGFCE**

RMAC Transmitted good frame counter E-frames

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TGFNE[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | TGFNE | RC | 0H | Transmitted good frame number E-frames<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when transmission of e-frame is completed on TX MHD I/F with (**PHY_LINK & MPIC.PLSPP)** and not error. |

## (37) **MTGFCP**

RMAC Transmitted good frame counter P-frames

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TGFNP[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_ MED _W-1:0 | TGFNP | RC | 0H | Transmitted good frame number P-frames<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when transmission of p-frame is completed on TX MHD I/F with (**PHY_LINK & MPIC.PLSPP)** and not error. |

## (38) **MTBFC**

RMAC Transmitted broadcast frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TBFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W -1:0 | TBFN | RC | 0H | Transmitted broadcast frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when destination address field of broadcast frame is sent on TX MHD I/F. |

## (39) **MTMFC**

RMAC Transmitted multicast frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TMFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:<br>COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W<br>-1:0 | TMFN | RC | 0H | Transmitted multicast frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when destination address field of multicast frame is sent on TX MHD I/F. |

## (40) **MTUFC**

RMAC Transmitted unicast frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TUFN[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | TUFN | RC | 0H | Transmitted unicast frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when destination address field of unicast frame is sent on TX MHD I/F. |

## (41) **MTEFC**

RMAC Transmitted error frame counter

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TEFN[COUNT_LOW_W-1:0] | | | | | | | | | | | | | | | |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31: COUNT_LOW_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_LOW_W-1:0 | TEFN | RC | 0H | Transmitted error frame number<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment condition<br>- Increment when frame transmission is completed on TX MHD I/F with error. |

## (42) **MPBLTFCESPi ( i=0..PORT_N-1)**

PHY Broken Link Transmitted Frame Counter E-frame per Source Port i ( i=0..PORT_N-1)

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PBLTFCESPi[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| **Bits** | **Bit name** | **R/W** | **Initial value** | **Function description** |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | PBLTFCESPi | RC | 0H | PHY Broken Link Transmission Frame Counter E-frame per Source Port i<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment conditions:<br>- HW: Incremented by 1 when a E-frame transmission error occurs with the broken PHY link (PHY link down) from source port i and if this register has a value different than {{COUNT_MED_W}{1'b1}}.<br>"E-frame transmission error occurs with the broken PHY link (PHY link down)" means "transmission of e-frame is completed on TX MHD I/F with !(**PHY_LINK & MPIC.PLSPP)"**<br>Restrictions:<br>- HW: This counter is incremented on **MTGFCE.TGFNE** count condition + PHY link down. **MTGFCE.TGFNE** counter do not increment when this counter increments.<br>- HW: This counter will be incremented with **MPBLTFCE.PBLTFCE**.<br>- HW: **MTXBCEU** and **MTXBCEL** will count independently. (Regardless of this register) |

## (43) **MPBLTFCPSPi ( i=0..PORT_N-1)**

PHY Broken Link Transmitted Frame Counter P-frame per Source Port i ( i=0..PORT_N-1)

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PBLTFCPSP[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| **Bits** | **Bit name** | **R/W** | **Initial value** | **Function description** |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | PBLTFCPSPi | RC | 0H | PHY Broken Link Transmission Frame Counter P-frame per Source Port i<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment conditions:<br>- HW: Incremented by 1 when a P-frame transmission error occurs with the broken PHY link (PHY link down) from source port i and if this register has a value different than {{COUNT_MED_W}{1'b1}}.<br>"P-frame transmission error occurs with the broken PHY link (PHY link down)" means "transmission of e-frame is completed on TX MHD I/F with !(**PHY_LINK & MPIC.PLSPP)"**<br>Restrictions:<br>- HW: This counter is incremented on **MTGFCP.TGFNP** count condition + PHY link down. **MTGFCP.TGFNP** counter do not increment when this counter increments.<br>- HW: This counter will be incremented with **MPBLTFCP.PBLTFCP**.<br>- HW: **MTXBCPU** and **MTXBCPL** will count independently. (Regardless of this register) |

## (44) **MPBLTFCE**

PHY Broken Link Transmitted Frame Counter E-frame

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PBLTFCE[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| **Bits** | **Bit name** | **R/W** | **Initial value** | **Function description** |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | PBLTFCE | RC | 0H | PHY Broken Link Transmission Frame Counter E-frame<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment conditions:<br>- HW: Incremented by 1 when a E-frame transmission error occurs with the broken PHY link (PHY link down) and if this register has a value different than {{COUNT_MED_W}{1'b1}}.<br>"E-frame transmission error occurs with the broken PHY link (PHY link down)" means "transmission of e-frame is completed on TX MHD I/F with !(**PHY_LINK & MPIC.PLSPP)"**<br>Restrictions:<br>- HW: This counter is incremented on **MTGFCE.TGFNE** count condition + PHY link down. **MTGFCE.TGFNE** counter do not increment when this counter increments.<br>- HW: This counter will be incremented with **MPBLTFCESPi.PBLTFCESPi**.<br>- HW: **MTXBCEU** and **MTXBCEL** will count independently. (Regardless of this register) |

## (45) **MPBLTFCP**

PHY Broken Link Transmitted Frame Counter P-frame

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PBLTFCP[COUNT_MED_W-1:0] | | | | | | | | | | | | | | | |

| **Bits** | **Bit name** | **R/W** | **Initial value** | **Function description** |
|---|---|---|---|---|
| 31: COUNT_MED_W | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| COUNT_MED_W-1:0 | PBLTFCP | RC | 0H | PHY Broken Link Transmission Frame Counter P-frame<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Reading this register clears it.<br>Increment conditions:<br>- HW: Incremented by 1 when a P-frame transmission error occurs with the broken PHY link (PHY link down) and if this register has a value different than {{COUNT_MED_W}{1'b1}}.<br>"P-frame transmission error occurs with the broken PHY link (PHY link down)" means "transmission of e-frame is completed on TX MHD I/F with !(**PHY_LINK & MPIC.PLSPP)"**<br>Restrictions:<br>- HW: This counter is incremented on **MTGFCP.TGFNP** count condition + PHY link down. **MTGFCP.TGFNP** counter do not increment when this counter increments.<br>- HW: This counter will be incremented with **MPBLTFCPSPi.PBLTFCPSPi**.<br>- HW: **MTXBCPU** and **MTXBCPL** will count independently. (Regardless of this register) |

### 4.2.3 RMAC Interrupt registers

## 4.2.3.1 Error interrupt registers

### (1) MEIS

MAC error interrupt status

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | FOES | FUES | FFS | RPOOMS | FRCES | CFCES | FFMES | FCMCES | PNAES | PDES | RSV | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | CTLES [PTP_TN-1:0] | | RPCRES | RPOES | REOES | FCES | BFES | TBCIS | TCES | FCDS | PFRROS | PRES | RSV | TSLS |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:30 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 29 | FOES | R!=W | 0B | Oversize error Status<br>Set condition<br>- HW: When too big frame is received. ( when the received frame size is bigger than the setting in MRFSCE.EMXS and MRFSCP.PMXS )<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction:<br>- Pause/PFC frames are out of scope. |
| 28 | FUES | R!=W | 0B | Undersize error Status<br>Set condition<br>- HW: When too short frame is received. ( when the received frame size is shorter than the setting in MRFSCE.EMNS and MRFSCP.PMNS )<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction:<br>- Pause/PFC frames are out of scope. |
| 27 | FFS | R!=W | 0B | Frame filtered Status<br>Set condition<br>- HW: When frame didn't pass the RMAC filters<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it. |
| 26 | RPOOMS | R!=W | 0B | Reception partially out of operation mode status<br>Set condition<br>- HW: when a frame is being received while RMAC is moving out of operation mode (suspending).<br>(1) initial or intermediate fragment of p-frame is receiving.<br>(2) e-frame is receiving when initial p-frame is already received and last fragment of p-frame is not received.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Notes:<br>- This error has nothing to do with partial rejection due to buffer overflow. E-Frames can only be fragmented/partialized by buffer overflows and it is not affected by partial out of operation. |

| | | | | |
|---|---|---|---|---|
| 25 | FRCES | R!=W | 0B | Fragment count error Status<br>Set condition<br>- HW: when a C-fragment is received with a bad fragment count field<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it. |
| 24 | CFCES | R!=W | 0B | C fragment count error Status<br>Set condition<br>- HW: when a C fragment not corresponding to the current frame is received.<br>For example: For S0-C0-C0-C1 and the P-Frame did not complete on all C0s. This error was asserted by last of C1 that expected as C0.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it. |
| 23 | FFMES | R!=W | 0B | Final fragment missing error Status<br>Set condition<br>- HW: When a S-fragment is received even though the previous fragment has been terminated by a mCRC<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it. |
| 22 | FCMCES | R!=W | 0B | FCS/mCRC error Status<br>Set condition<br>- HW: When unexpected FCS/mCRC are received<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it. |
| 21 | PNAES | R!=W | 0B | PHY nibble alignment error Status<br>Set condition<br>- HW: When residual-bit is received.<br>Clear conditions:<br>- SW: Writing 1 to this bit will clear it. |
| 20 | PDES | R!=W | 0B | PHY data error Status.<br>Set condition<br>- HW: When frame is received with error.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it. |
| 19:12+PTP_TN | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| [12+PTP_TN-1:12] | CTLES | R!=W | 0B | Captured timestamp lost error status<br>Set condition<br>- HW: When captured timestamp is lost.<br>This bit is fixed to 0.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Error recovery<br>- No recovery. Timestamp is lost. |
| 11 | RPCRES | R!=W | 0B | Rx PCH CRC Error Status<br>Set condition<br>- HW: When CRC error is detected in PCH period.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Error recovery<br>- No recovery. Frame is lost.<br>Restriction<br>- SW: This error is only happened when MPIC.PIS = 3'b100. |

| | | | | |
|---|---|---|---|---|
| 10 | RPOES | R!=W | 0B | Rmac pFrame Overflow Error Status<br>Value<br>- 0: No overflow error<br>- 1: overflow error occurred during p-frame reception.<br>Set condition<br>- HW: The acknowledge signal was not received successfully from an upper layer (e.g. MHD) during p-frame reception.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- HW: Pulse is generated per 1 frame although multiple de-assertion is generated.<br>Notes:<br>- When frame reception is already ongoing and buffer overflow occurs i.e CABPPCM.RPC = 0, RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. When new frame is received, during buffer overflow i.e CABPPCM.RPC = 0, the incoming frame will be rejected completely. |
| 9 | REOES | R!=W | 0B | Rmac eFrame Overflow Error Status<br>Value<br>- 0: No overflow error<br>- 1: overflow error occurred during e-frame reception.<br>Set condition<br>- HW: The acknowledge signal was not received successfully from an upper layer (e.g. MHD) during e-frame reception.<br>Clear conditions:<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- HW: Pulse is generated per 1 frame although multiple de-assertion is generated.<br>Notes:<br>- When frame reception is already ongoing and buffer overflow occurs i.e CABPPCM.RPC = 0, RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. When new frame is received, during buffer overflow i.e CABPPCM.RPC = 0, the incoming frame will be rejected completely. |
| 8 | FCES | R!=W | 0B | Frame count error status<br>Set condition<br>- When S fragment has been received in the wrong order.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it.<br>Notes:<br>- When C-fragment is received at very first time before any valid S-fragment is received or C-fragment is received after frame termination is occurred before any valid S-fragment is received, then RMAC will discard the fragment and this error status flag will not be set. For Error status flag to get set, RMAC should have received one valid S-fragment before error condition (frame count error) has occurred. |
| 7 | BFES | R!=W | 0B | Bad fragment error status<br>Set condition<br>- HW: C fragment has been detected without receiving S fragment<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it.<br>Notes:<br>- When C-fragment is received at very first time before any valid S-fragment is received or C-fragment is received after frame termination is occurred before any valid S-fragment is received, then RMAC will discard the fragment and this error status flag will not be set. For Error status flag to get set, RMAC should have received one valid S-fragment before error condition (bad fragment error) has occurred. |
| 6 | TBCIS | R!=W | 0B | TX bad CRC insertion status<br>Set condition<br>- HW: A bad CRC is inserted from the upper layer (e.g. MHD).<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it. |

| | | | | |
|---|---|---|---|---|
| 5 | TCES | R!=W | 0B | TX CRC error status<br>Set condition<br>- HW: When CRC error occurred<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it. |
| 4 | FCDS | R!=W | 0B | False Carrier Detection Status<br>Set condition<br>- HW: When PHY-LSI has detected a false carrier on the line in case of MII and GMII.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it.<br>Notes:<br>- This status flag is not applicable in case of XGMII. |
| 3 | PFRROS | R!=W | 0B | Pause or pfc Frame Retransmit Retry Over Status<br>Set condition<br>- HW: when the total number of Pause (pfc) transmit count reaches the configured upper limit and Pause time expires.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it. |
| 2 | PRES | R!=W | 0B | Pause or pfc frame Reception Error Status<br>Set condition<br>- HW: When Pause or PFC frame is received with lower than minimum pause/pfc frame length.<br>Minimum PAUSE Length : 18Byte (MRGC.RCPT==0) or 22 Byte ((MRGC.RCPT==1)<br>Minimum PFC Length : 34Byte (MRGC.RCPT==0) or 38 Byte ((MRGC.RCPT==1)<br>- HW: When PAUSE/PFC frame is received with fcs error in MRGC.RFCFE==0,<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it.. |
| 1 | RSV | R0 | 0H | - Reserved area. On read, 0 will be returned |
| 0 | TSLS | R!=W | 0B | Transmission Stream Lost Status<br>Set condition<br>- HW: when MHD do not provide the transmitted data timely. In this case an invalid FCS is transmitted. (only for e-frames)<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this register will clear it.<br>[Note and Restriction]<br>When using Cut through, this flag set condition may be due to the state ([COMA] CAEIS0.BPOPS) where all Buffer pointer are used. Please use the Watermark function so that Buffer Pool Out of Pointer Status does not occur. If get this error with Buffer Pool Out of Pointer Status using Cut through (it may get hung-up in the transmit state), initialize the state with Emergency reset flow [TOP]. |

Note:
- This interrupt register is mapped to rmc_meis_int interrupt wire.

**Table 4-1 Transmission error list**

| Error type | Event | Status Bit |
|---|---|---|
| TX bad CRC insertion status | A packet has been received for transmission by RMAC with an error and a wrong FCS has been inserted. | MEIS.TBCIS |
| TX CRC error status | A packet which included a presupposed valid FCS had a faulty FCS | MEIS.TCES |
| Pause frame Retransmit Retry Over Status | A Pause or PFC retransmit count reaches the configured upper limit(MTPFC.APFRLV) and Pause time expires. | MEIS.PFRROS |
| Transmission Stream Lost Status | An e-frame data hasn't been provided on time to RMAC | MEIS.TSLS |

**Table 4-2 Reception error list**

| No | Error type | Event | Frame status bit | Status register |
|---|---|---|---|---|
| 1 | PHY data error | Received a data with error.<br>RMAC handle frame normally with error status, and generate interrupt. | Bit 0 | MEIS.PDES |
| 2 | PHY nibble alignment error | Received residual-bit.<br>RMAC adds four bits padding to the reception error frames for completing the byte, handle frame normally with error status, and generate interrupt. | Bit 1 | MEIS.PNAES |
| 3 | FCS/mCRC error | Received unexpected FCS field value.<br>RMAC handle frame normally with error status, and generate interrupt. | Bit 2 | MEIS.FCMCES |
| 4 | Final fragment missing error | Occurs when a S-fragment is received even though the previous fragment has been terminated by a MCRC<br>RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. | Bit 3 | MEIS.FFMES |
| 5 | C fragment count error (SMD Error) | Occurs when a C fragment not corresponding to the current frame is received.<br>For example: For S0-C0-C0-C1 and the P-Frame did not complete on all C0s. This error was asserted by last of C1 that expected as C0.<br>RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. | Bit 4 | MEIS.CFCES |
| 6 | Fragment count error<br>(FRAG_COUNT Error) | Occurs when a C-fragment is received with a bad fragment count field<br>RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. | Bit 5 | MEIS.FRCES |
| 7 | Reception partially out of operation mode | Occurs when a frame is being received while RMAC is moving out of operation mode<br>RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. | Bit 6 | MEIS.RPOOMS |
| 8 | Frame filtered | The frame didn't pass the RMAC filters<br>RMAC handle frame normally with error status, and generate interrupt. | Bit 7 | MEIS.FES |
| 9 | Undersize error | The frame is too short<br>RMAC handle frame normally with error status, and generate interrupt. | Bit 8 | MEIS.FUES |
| 10 | Oversize error | The frame is too big<br>RMAC handle frame normally with error status, and generate interrupt. | Bit 9 | MEIS.FOES |
| 11 | Rx PCH CRC Error Status | PCH CRC Error of received frame<br>RMAC handle frame normally with error status, and generate interrupt. | Bit 11 | MEIS.RPCRES |
| 12 | Buffer overflow error | Occurs when the RX done signal has been de-asserted during frame reception<br>RMAC adds one byte padding to the reception error Frames to terminate the protocol internally. | Bit 12 | MEIS.REOES<br>MEIS.RPOES |
| 13 | Frame count error status | Occurs when a wrong S-frame order is received.<br>For example: S0-S1-S2-**S1**<br>RMAC handle frame normally, and generate interrupt. | - | MEIS.FCES |
| 14 | Bad fragment error status | Occurs when a C fragment has been detected without receiving S fragment<br>RMAC discard frame, and generate interrupt. | - | MEIS.BFES |
| 15 | False Carrier Detection Status | Occurs when PHY-LSI has detected a false carrier on the line.<br>RMAC handle frame normally, and generate interrupt. | - | MEIS.FCDS |

## (2) **MEIE**

MAC error interrupt enable

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | FOEE | FUEE | FFE | RPOOME | FRCEE | CFCEE | FFMEE | FCMCEE | PNAEE | PDEE | RSV | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | CTLEE [ PTP_TN-1:0] | | RPCREE | RPOEE | REOEE | FCEE | BFEE | TBCIE | TCEE | FCDE | PFRROE | PREE | RSV | TSLE |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:30 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 29 | FOEE | R!=W | 0B | Oversize error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- SW: Writing 1 to **MEID.FOED** will clear this bit. |
| 28 | FUEE | R!=W | 0B | Undersize error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FUED** will clear this bit. |
| 27 | FFE | R!=W | 0B | Frame filtered Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FFD** will clear this bit. |
| 26 | RPOOME | R!=W | 0B | Reception partially out of operation mode Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.RPOOMD** will clear this bit. |
| 25 | FRCEE | R!=W | 0B | Fragment count error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FRCED** will clear this bit. |
| 24 | CFCEE | R!=W | 0B | C fragment count error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>Writing 1 to **MEID.CFCED** will clear this bit. |
| 23 | FFMEE | R!=W | 0B | Final fragment missing error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FFMED** will clear this bit. |
| 22 | FCMCEE | R!=W | 0B | FCS/mCRC error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FCMCED** will clear this bit. |
| 21 | PNAEE | R!=W | 0B | PHY nibble alignment error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.PNAED** will clear this bit. |
| 20 | PDEE | R!=W | 0B | PHY data error Enable.<br>Set condition<br>- Writing 1 to one of this bit will set it. |

| | | | | |
|---|---|---|---|---|
| | | | | Clear condition<br>- Writing 1 to **MEID.PDED** will clear this bit. |
| 19:12+PTP_TN | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| [12+PTP_TN-1:12] | CTLEE | R!=W | 0B | Captured timestamp lost error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.CTLED** will clear this bit. |
| 11 | RPCREE | R!=W | 0B | Rx PCH CRC Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.RPIDED** will clear this bit. |
| 10 | RPOEE | R!=W | 0B | Rmac pFrame Overflow Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.RPOED** will clear this bit. |
| 9 | REOEE | R!=W | 0B | Rmac eFrame Overflow Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.REOED** will clear this bit. |
| 8 | FCEE | R!=W | 0B | Frame Count Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FCED** will clear this bit. |
| 7 | BFEE | R!=W | 0B | Bad Fragment Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.BFED** will clear this bit. |
| 6 | TBCIE | R!=W | 0B | TX bad CRC Insertion Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.TBCID** will clear this bit. |
| 5 | TCEE | R!=W | 0B | Tx Crc Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.TCED** will clear this bit. |
| 4 | FCDE | R!=W | 0B | False Carrier Detection Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.FCDD** will clear this bit. |
| 3 | PFRROE | R!=W | 0B | Pause or pfc Frame Retransmit Retry Over Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.PFRROD** will clear this bit. |
| 2 | PREE | R!=W | 0B | Pause or pfc frame Reception Error Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>Writing 1 to **MEID.PRED** will clear this bit. |
| 1 | RSV | R0 | 0H | - Reserved area. On read, 0 will be returned |
| 0 | TSLE | R!=W | 0B | Transmission Stream Lost Enable<br>Set condition<br>- Writing 1 to one of this bit will set it.<br>Clear condition<br>- Writing 1 to **MEID.TSLD** will clear this bit. |

## (3) **MEID**

MAC error interrupt disable

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | FOED | FUED | FFD | RPOOMD | FRCED | CFCED | FFMED | FCMCED | PNAED | PDED | RSV | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | CTLED [ PTP_TN-1:0] | | RPCRED | RPOED | REOED | FCED | BFED | TBCID | TCED | FCDD | PFRROD | PRED | RSV | TSLD |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:30 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 29 | FOED | R!=W | 0B | Oversize error disable<br>Function<br>Writing 1 to this register will clear **MEID.FOEE** register. |
| 28 | FUED | R!=W | 0B | Undersize error disable<br>Function<br>Writing 1 to this register will clear **MEIE.FUEE** register. |
| 27 | FFD | R!=W | 0B | Frame filtered disable<br>Function<br>Writing 1 to this register will clear **MEIE.FFE** register. |
| 26 | RPOOMD | R!=W | 0B | Reception partially out of operation mode disable<br>Function<br>Writing 1 to this register will clear **MEIE.RPOOME** register. |
| 25 | FRCED | R!=W | 0B | Fragment count error disable<br>Function<br>Writing 1 to this register will clear **MEIE.FRCEE** register. |
| 24 | CFCED | R!=W | 0B | C fragment count error disable<br>Function<br>Writing 1 to this register will clear **MEIE.CFCEE** register. |
| 23 | FFMED | R!=W | 0B | Final fragment missing error disable<br>Function<br>Writing 1 to this register will clear **MEIE.FFMEE** register. |
| 22 | FCMCED | R!=W | 0B | FCS/mCRC error disable<br>Function<br>Writing 1 to this register will clear **MEIE.FCMCEE** register. |
| 21 | PNAED | R!=W | 0B | PHY nibble alignment error disable<br>Function<br>Writing 1 to this register will clear **MEIE.PNAEE** register. |
| 20 | PDED | R!=W | 0B | PHY data error disable.<br>Function<br>Writing 1 to this register will clear **MEIE.PDEE** register. |
| 19: 12+PTP_TN | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| [12+PTP_TN-1:12] | CTLED | R!=W | 0B | Captured timestamp lost error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.CTLEE** register. |
| 11 | RPCRED | R!=W | 0B | Rx PCH CRC Error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.RPCREE** register. |
| 10 | RPOED | R!=W | 0B | Rmac pFrame Overflow Error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.RPOEE** register. |
| 9 | REOED | R!=W | 0B | Rmac eFrame Overflow Error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.REOEE** register. |
| 8 | FCED | R!=W | 0B | Frame Count Error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.FCEE** register. |
| 7 | BFED | R!=W | 0B | Bad Fragment Error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.BFEE** register. |

| | | | | |
|---|---|---|---|---|
| 6 | TBCID | R!=W | 0B | TX bad CRC Insertion Disable<br>Function<br>- Writing 1 to this register will clear **MEIE.TBCIE** register. |
| 5 | TCED | R!=W | 0B | Tx Crc Error Disable<br>Function<br>- Writing 1 to this register will clear **MEIE.TCEE** register. |
| 4 | FCDD | R!=W | 0B | False Carrier Detection Disable<br>Function<br>- Writing 1 to this register will clear **MEIE.FCDE** register. |
| 3 | PFRROD | R!=W | 0B | Pause or pfc Frame Frame Retransmit Retry Over Disable<br>Function<br>- Writing 1 to this register will clear **MEIE.PFRROE** register. |
| 2 | PRED | R!=W | 0B | Pause or pfc frame Reception Error Disable<br>Function<br>Writing 1 to this register will clear **MEIE.PREE** register. |
| 1 | RSV | R0 | 0H | - Reserved area. On read, 0 will be returned |
| 0 | TSLD | R!=W | 0B | Transmission Stream Lost Disable<br>Function<br>- Writing 1 to this register will clear **MEIE.TSLE** register. |

### 4.2.3.2 Monitoring registers

#### (1) MMIS0

MAC monitoring interrupt status 0

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | XLISDS | XRFSDS | XLFSDS | XLFES | XLFDS | RSV | ANDETS | RSV | VFRS | LVFS | LVSS | PIDS | PLSCS |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:13 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 12 | XLISDS | R!=W | 0B | XGMII Link Interruption Signal detect status<br>Set condition<br>- HW: When Link Interruption Signal is detected. (32'h03_00_00_9c is detected on XGMII bus)<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has not effect.<br>- It's no effect when suspending, , e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 11 | XRFSDS | R!=W | 0B | XGMII Remote Fault Signal detect status<br>Set condition<br>- HW: When Remote Fault Signal is detected. (32'h02_00_00_9c is detected on XGMII bus)<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has not effect.<br>- It's no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 10 | XLFSDS | R!=W | 0B | XGMII Local Fault Signal detect status<br>Set condition<br>- HW: When Local Fault Signal is detected. (32'h01_00_00_9c is detected on XGMII bus)<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has not effect.<br>- It's no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 9 | XLFES | R!=W | 0B | XGMII Link Fault Exit Status (exit from "FAULT" state defined in 802.3 Clause46 XGMII link fault signaling state machine)<br>Set condition<br>- HW: 128 cycle is elapsed after Link Fault Signal is detected on "FAULT" state and no Link Fault Signal is detected after last Link Fault Signal is detected.<br>- HW: Link Fault Signal is detected on "FAULT" state but current and previous received Link Fault Signal is different type.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.Restriction<br>- Writing 0 to this bit has not effect.<br>- It's no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state.<br>- |

| | | | | |
|---|---|---|---|---|
| 8 | XLFDS | R!=W | 0B | XGMII Link Fault Detect Status (enter to “FAULT” state defined in 802.3 Clause46 XGMII link fault signaling state machine)<br>Set condition<br>HW: When same fault signaling are detected 4 continuous clock cycles on RX XGMII and enter to “FAULT” state.<br>Clear Condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has not effect.<br>- It’s no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 7 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 6 | ANDETS | R!=W | 0B | Auto-Nego message detection or XGMII Link Interruption Signal detect status<br>Set condition<br>- HW: When Signal is detected.<br>(32’h03_**_**_9c is detected on XGMII bus)<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has not effect.<br>- It’s no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 5 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 4 | VFRS | R!=W | 0B | Verify frame reception Status<br>Set condition<br>- HW: When valid verify frame has been received.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has not effect.<br>- It’s no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 3 | LVFS | R!=W | 0B | Link Verification Failed Status<br>Set condition<br>- HW: When Link Verification process failed. It means that the remote node is not able to handle Preemptable frames.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has not effect.<br>- It’s no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 2 | LVSS | R!=W | 0B | Link Verification Succeed Status<br>Set condition<br>- HW: When Link Verification process succeed.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has not effect.<br>- It’s no effect when suspending, e.g. an upper layer (e.g. MHD) is not in OPERATION state. |
| 1 | PIDS | R!=W | 0B | PHY interrupt detection Status<br>Set condition<br>- HW: PHY interrupt pin (rmc_phy_int port) is active level.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has not effect.<br>- SW: Enabling **MPIC.PIPP** to check this status.<br>- HW: This flag could be asserted unintentionally before configuring PHY interrupt polarity (**MPIC.PIP**) into the right value. In the phase, this flag should be ignored. |

| | | | | |
|---|---|---|---|---|
| 0 | PLSCS | R!=W | 0B | PHY Link signal change Status<br>Set condition<br>- HW: When the value of link status pin (rmc_phy_link port) is changed.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has not effect.<br>- SW: Enabling **MPIC.PLSPP** to check this status.<br>- SW: To check the current link state, refer to the link status pin state bit in the PHY link status register (**MPIM.PLS**). |

Note:
- This interrupt register is mapped to rmc_mmis0_int interrupt wire.

## (2) **MMIE0**

MAC monitoring interrupt enable 0

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | XLISDE | XRFSDE | XLFSDE | XLFEE | XLFDE | RSV | ANDETE | RSV | VFRE | LVFE | LVSE | PIDE | PLSCE |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:13 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 12 | XLISDE | R!=W | 0B | XGMII Link Interruption Signal Detect Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 11 | XRFSDE | R!=W | 0B | XGMII Remote Fault Signal Detect Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 10 | XLFSDE | R!=W | 0B | XGMII Local Fault Signal Detect Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 9 | XLFEE | R!=W | 0B | XGMII Link Fault Exit Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 8 | XLFDE | R!=W | 0B | XGMII Link Fault Detect Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 7 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 6 | ANDETE | R!=W | 0B | Auto-Nego message or XGMII Link Interruption Signal detection Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 5 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 4 | VFRE | R!=W | 0B | Verify frame reception Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 3 | LVFE | R!=W | 0B | Link Verification Failed Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 2 | LVSE | R!=W | 0B | Link Verification Succeed Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 1 | PIDE | R!=W | 0B | PHY interrupt detection Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |
| 0 | PLSCE | R!=W | 0B | PHY Link signal change Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has not effect. |

## (3) **MMID0**

MAC monitoring interrupt disable 0

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | XLISDD | XRFSDD | XLFSDD | XLFED | XLFDD | RSV | ANDETD | RSV | VFRD | LVFD | LVSD | PIDD | PLSCD |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:13 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 12 | XLISDD | R0W | 0B | XGMII Link Interruption Signal Detect Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.XLISDE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 11 | XRFSDD | R0W | 0B | XGMII Remote Fault Signal Detect Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.XRFSDE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 10 | XLFSDD | R0W | 0B | XGMII Local Fault Signal Detect Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.XLFSDE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 9 | XLFED | R0W | 0B | XGMII Link Fault Exit Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.XLFEE** register.<br>Restriction<br>Writing 0 to this register has no effect. |
| 8 | XLFDD | R0W | 0B | XGMII Link Fault Detect Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.XLFDE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 7 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 6 | ANDETD | R0W | 0B | Auto-Nego message or XGMII Link Interruption Signal detection Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.ANDETE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 5 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 4 | VFRD | R0W | 0B | Verify frame reception Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.VFRE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 3 | LVFD | R0W | 0B | Link Verification Failed Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.LVFE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 2 | LVSD | R0W | 0B | Link Verification Succeed Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.LVSE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |

| | | | | |
|---|---|---|---|---|
| 1 | PIDD | R0W | 0B | PHY interrupt detection Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.PIDE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 0 | PLSCD | R0W | 0B | PHY Link signal change Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE0.PLSCE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |

## (4) **MMIS1**

MAC monitoring interrupt status 1

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | PPRACS | PAACS | PWACS | PRACS |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:4 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 3 | PPRACS | R!=W | 0B | PHY Post-Read access completed Status<br>Set condition<br>- HW: When PHY post-read access has completed.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>Writing 0 to this bit has no effect. |
| 2 | PAACS | R!=W | 0B | PHY Address access completed Status<br>Set condition<br>- HW: When PHY address access has completed.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has no effect. |
| 1 | PWACS | R!=W | 0B | PHY Write access completed Status<br>Set condition<br>- HW: When PHY write access has completed.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has no effect. |
| 0 | PRACS | R!=W | 0B | PHY Read access completed Status<br>Set condition<br>- HW: When PHY read access has completed.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- Writing 0 to this bit has no effect. |

Note:
- This interrupt register is mapped to rmc_mdio_int interrupt wire.

## (5) **MMIE1**

MAC monitoring interrupt enable 1

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | PPRACE | PAACE | PWACE | PRACE |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:4 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 3 | PPRACE | R!=W | 0B | PHY Post-Read access completed Enable<br>Set Condition<br>- SW: Writing 1 to this bit will set it.<br>Clear Condition<br>- HW: Writing 1 to **MMID1.PPRACD** register will clear this bit.<br>Restriction<br>SW: Writing 0 to this bit has no effect. |
| 2 | PAACE | R!=W | 0B | PHY 107runcat access completed Enable<br>Set Condition<br>- SW: Writing 1 to this bit will set it.<br>Clear Condition<br>- HW: Writing 1 to **MMID1.PAACD** register will clear this bit.<br>Restriction<br>- SW: Writing 0 to this bit has no effect. |
| 1 | PWACE | R!=W | 0B | PHY Write access completed Enable<br>Set Condition<br>- SW: Writing 1 to this bit will set it.<br>Clear Condition<br>- HW: Writing 1 to **MMID1.PWACD** register will clear this bit.<br>Restriction<br>- SW: Writing 0 to this bit has no effect. |
| 0 | PRACE | R!=W | 0B | PHY Read access completed Enable<br>Set Condition<br>- SW: Writing 1 to this bit will set it.<br>Clear Condition<br>- HW: Writing 1 to **MMID1.PRACD** register will clear this bit.<br>Restriction<br>- SW: Writing 0 to this bit has no effect. |

## (6) **MMID1**

MAC monitoring interrupt disable 1

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | PPRACD | PAACD | PWACD | PRACD |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:4 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 3 | PPRACD | R0W | 0B | PHY Post-Read access completed Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE1.PPRACE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 2 | PAACD | R0W | 0B | PHY Address access completed Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE1.PAACE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 1 | PWACD | R0W | 0B | PHY Write access completed Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE1.PWACE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 0 | PRACD | R0W | 0B | PHY Read access completed Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE1.PRACE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |

## (7) **MMIS2**

MAC monitoring interrupt status 2

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | LPIDIS | LPIAIS | MPDIS |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:3 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 2 | LPIDIS | R!=W | 0B | LPI de-assertion interrupt status<br>Set condition<br>- HW: When LPI request de-assertion.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has no effect.<br>- HW: It's no effect when suspending.<br>- HW: This bit have high priority for set condition. |
| 1 | LPIAIS | R!=W | 0B | LPI assertion interrupt status<br>Set condition<br>- HW: When LPI request is detected. This register is continuously asserted when LPI request continues<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has no effect.<br>- HW: It's no effect when suspending.<br>- HW: This bit have high priority for set condition. |
| 0 | MPDIS | R!=W | 0B | Magic Packet Detection Interrupt Status<br>Set condition<br>- HW: When a Magic Packet has been detected.<br>Clear condition<br>- HW: Being in RESET mode will clear this register.<br>- SW: Writing 1 to this bit will clear it.<br>Restriction<br>- SW: Writing 0 to this bit has no effect.<br>- HW: Magic Packet in Preemptable frames is ignored.<br>- HW: CRC of magic packet is not checked.<br>- HW: It's no effect when suspending.<br>- HW: This bit have high priority for set condition. |

Note:

- This interrupt register is mapped to rmc_mp_int , rmc_lpi_start_int and rmc_lpi_stop_int interrupt wire.

- This register's value goes through the synchronizer between clk and clk_phy_rx.
  The result of write access is reflected to each SFR bit after "5clk + 4clk_phy_rx" in Wrost Case in RMAC.
  Do not make write access continuously. Please wait following time for re-write access.

## (8) **MMIE2**

MAC monitoring interrupt enable 2

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | LPIDIE | LPIAIE | MPDIE |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:3 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 2 | LPIDIE | R!=W | 0B | LPI de-assertion Interrupt Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has no effect. |
| 1 | LPIAIE | R!=W | 0B | LPI assertion Interrupt Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has no effect. |
| 0 | MPDIE | R!=W | 0B | Magic Packet™ Detection Interrupt Enable<br>Writing 1 to this register will set it.<br>Writing 0 to this register has no effect. |

Note:

- This register's value goes through the synchronizer between clk and clk_phy_rx.
  The result of write access is reflected to each SFR bit after "5clk + 4clk_phy_rx" in Wrost Case in RMAC.
  Do not make write access continuously. Please wait following time for re-write access.

## (9) **MMID2**

MAC monitoring interrupt disable 2

| B31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RSV | | | | | | | | | | | | | | | |
| B15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSV | | | | | | | | | | | | | LPIDID | LPIAID | MPDID |

| Bits | Bit name | R/W | Initial value | Function description |
|---|---|---|---|---|
| 31:3 | RSV | R0 | 0H | Reserved area. On read, 0 will be returned |
| 2 | LPIDID | R0W | 0B | LPI de-assertion Interrupt Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE2.LPIDIE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 1 | LPIAID | R0W | 0B | LPI assertion Interrupt Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE2.LPIAIE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |
| 0 | MPDID | R0W | 0B | Magic Packet™ Detection Interrupt Disable<br>Function:<br>- Writing 1 to one of this bit will clear **MMIE2.MPDIE** register.<br>Restriction<br>- Writing 0 to this register has no effect. |

Note:

- This register's value goes through the synchronizer between clk and clk_phy_rx.
  The result of write access is reflected to each SFR bit after "5clk + 4clk_phy_rx" in Wrost Case in RMAC.
  Do not make write access continuously. Please wait following time for re-write access.

# 5.Register utilizations

## 5.1 Operation Modes

The operation mode is linked with the upper layer EAMS.OPS of [TSNA]. For example, “out of OPERATION means “EAMS.OPS != 2’b11”

### 5.1.1 Frame Handling

In transmission, this IP transmits all frames from an upper module (e.g. MHD) when this IP is in operation state,e.g. when MHD is in OPERATION state. This IP receives an e-frame and a p-frame from the MHD.

In reception, this IP only receives frames from PHY when this IP is in operation state, e.g. when MHD is in OPERATION state and outputs them to the upper module(e.g. MHD). In other modes, this IP discards all received frames from PHY internally, and doesn’t output them to the upper module.
When the system supports preemption and the Mode change is executed, this IP of the Rx side behaves like the following figures.

**[Mode change from DISABLE to OPERATION on Rx]**

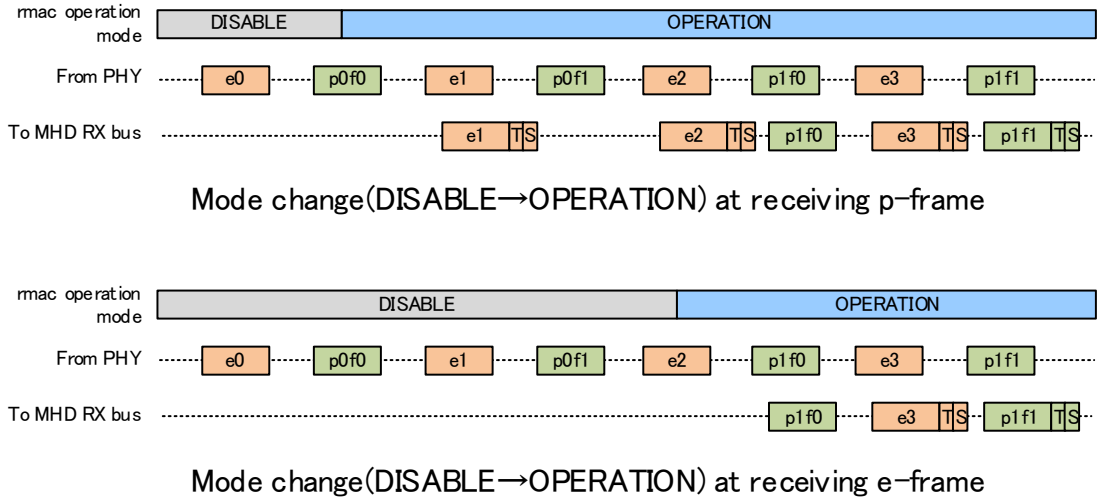



Figure 5-1 Mode change(DISABLE→OPERATION) on Rx

**[Mode change from OPERATION to DISABLE on Rx]**

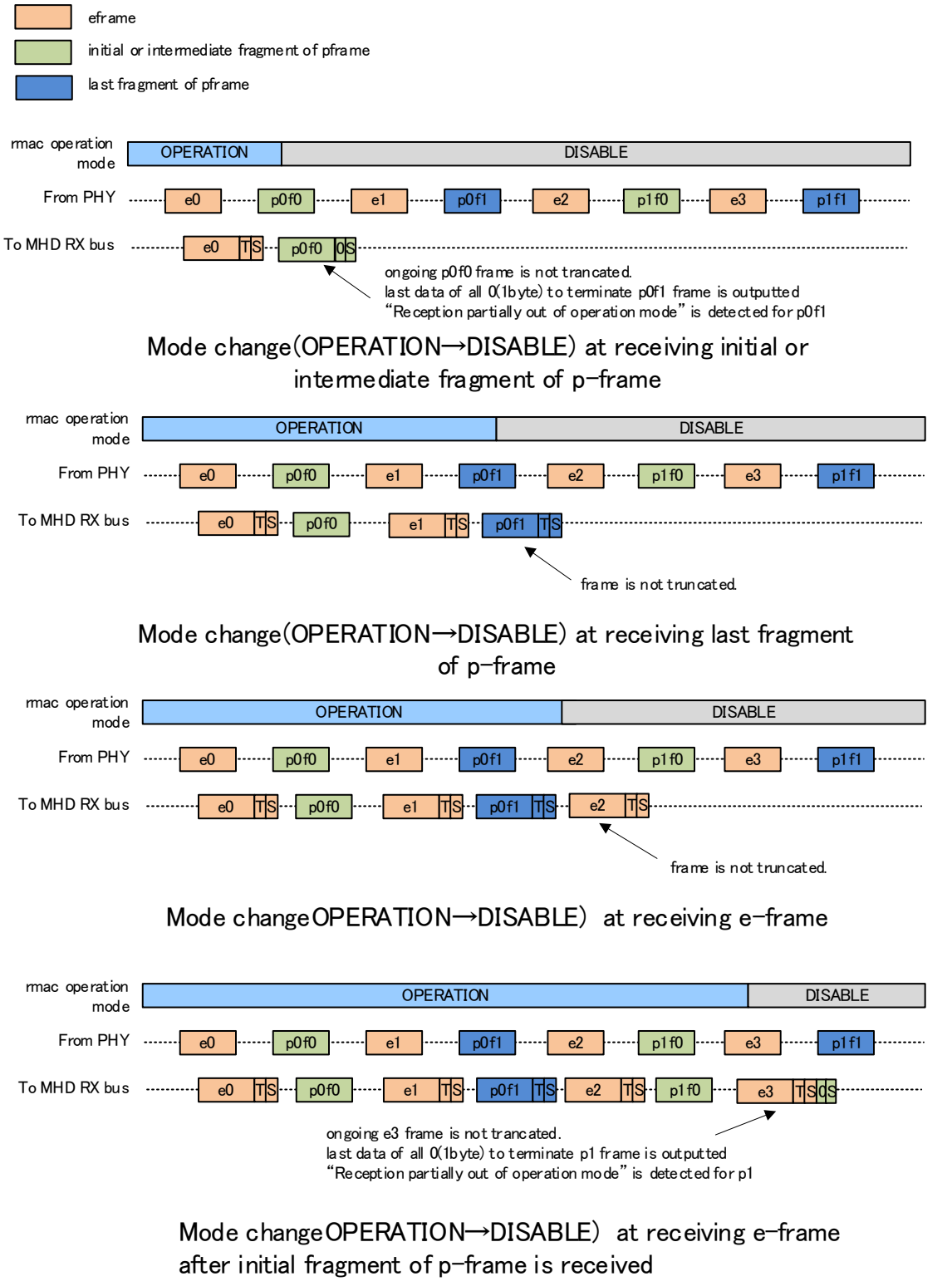



Figure 5-2 Mode change(OPERATION→DISABLE) on Rx

## 5.2 Software flows

Restrictions:

- SW: Please follow to the flow in this section.

### 5.2.1 Software flow legend

Software flow legend is described in Figure 5-3 .

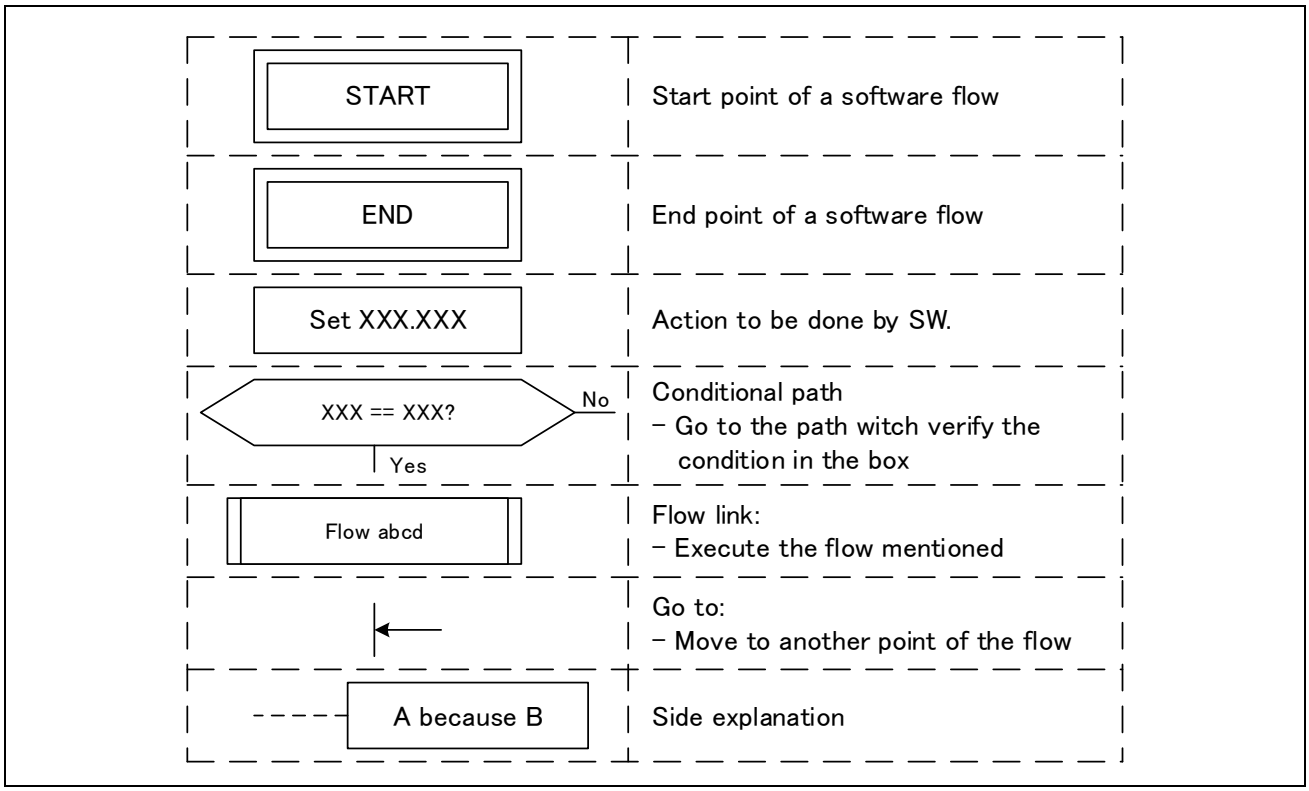



Figure 5-3 **Software flow legend**

### 5.2.2 Initialization flow

The figure below shows the initialization flow.

Restrictions:

- This flow is referenced by upper layer and is usable in CONFIG modes.

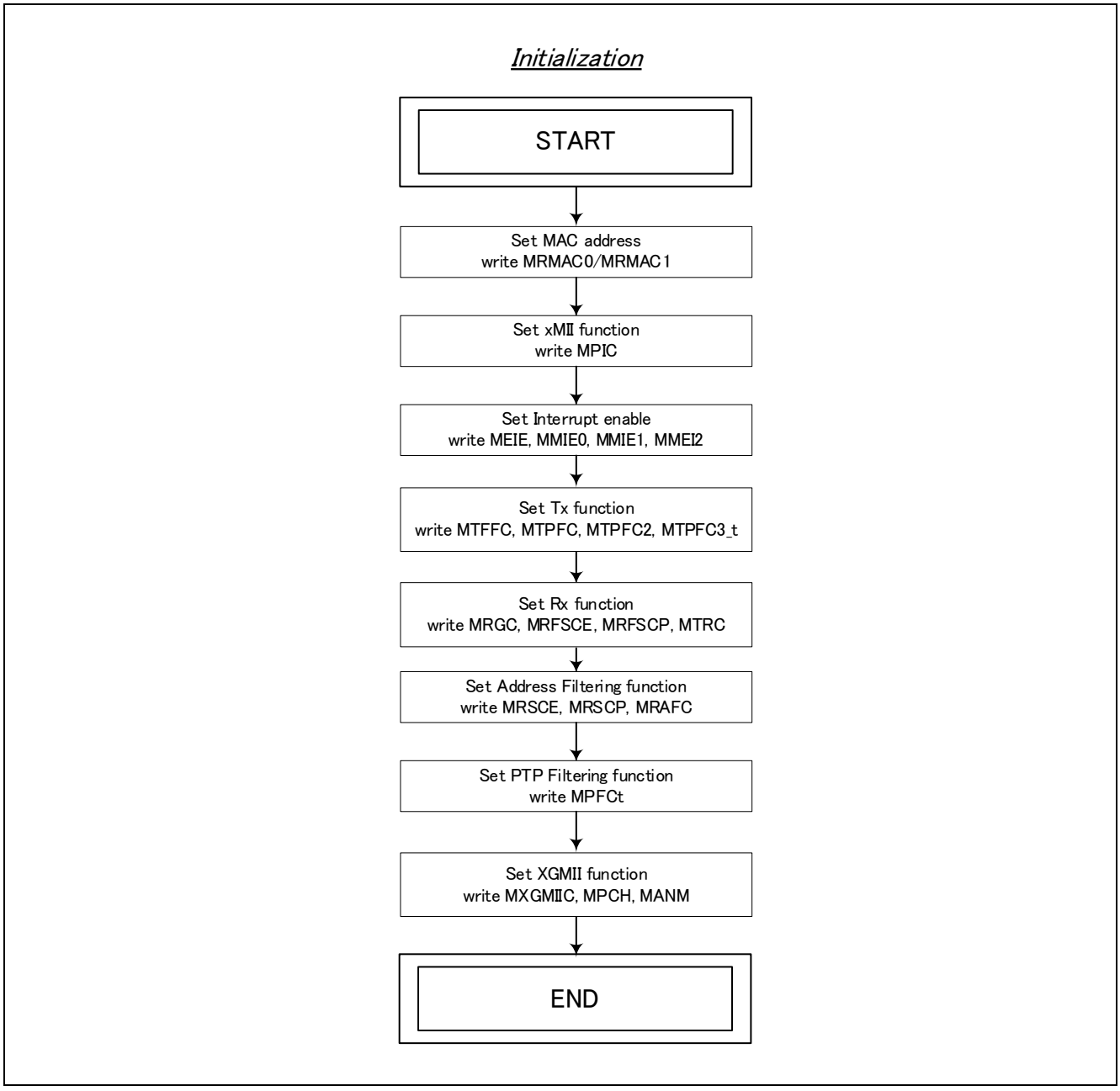



Figure 5-4 **Initialization flow**

### 5.2.3 Interrupt handling flow

Interrupt handling flow can be used in any mode except RESET mode.
The figure below shows the interrupt handling flow.

Restrictions:
- This flow is not usable in RESET modes.

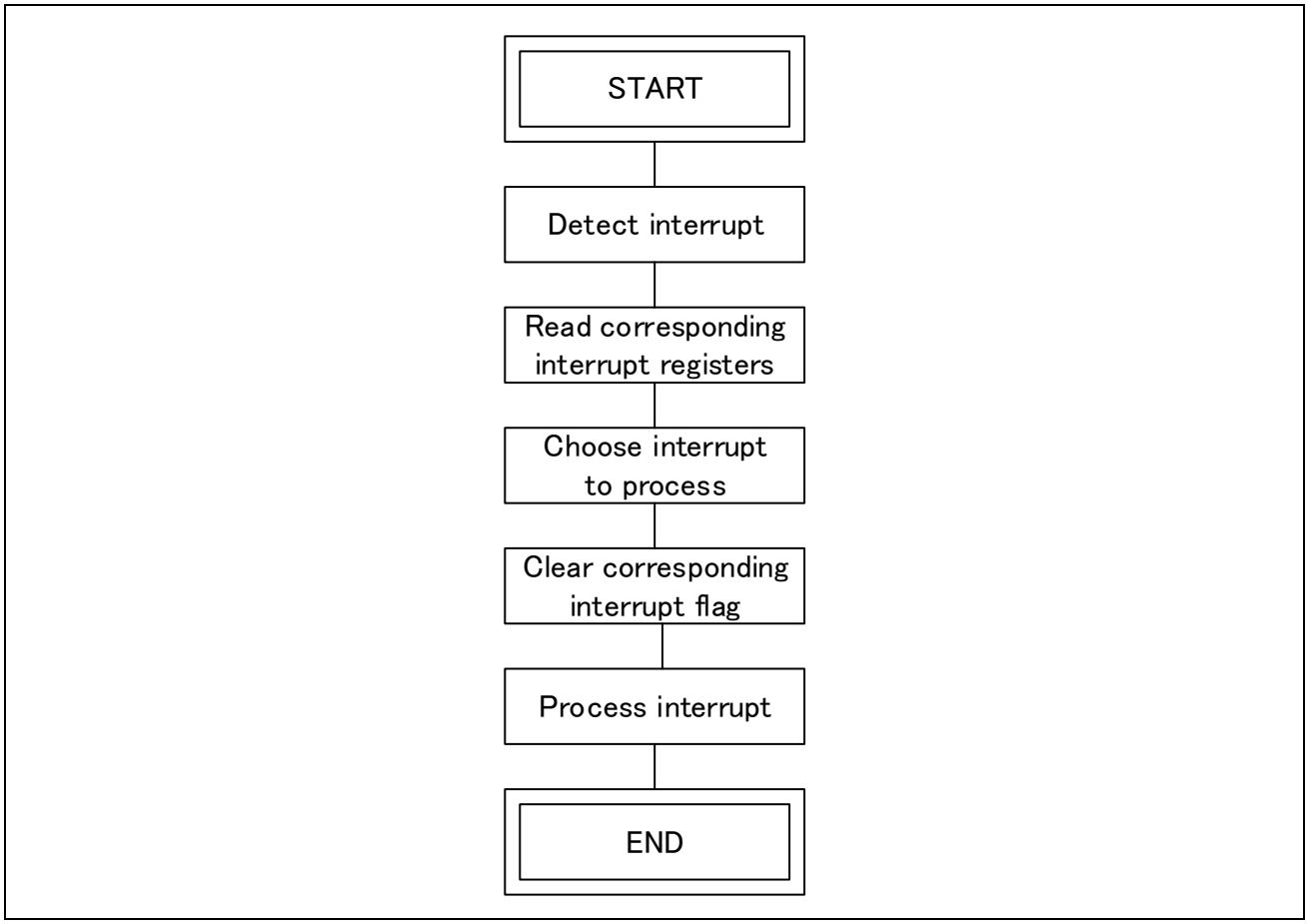



Figure 5-5 **Interrupt handling flow**

### 5.2.4 PHY MDIO and eMDIO Access flows

The figure below shows the PHY MDIO and eMDIO access flow.

Restrictions:
- This flow is not usable in RESET modes.

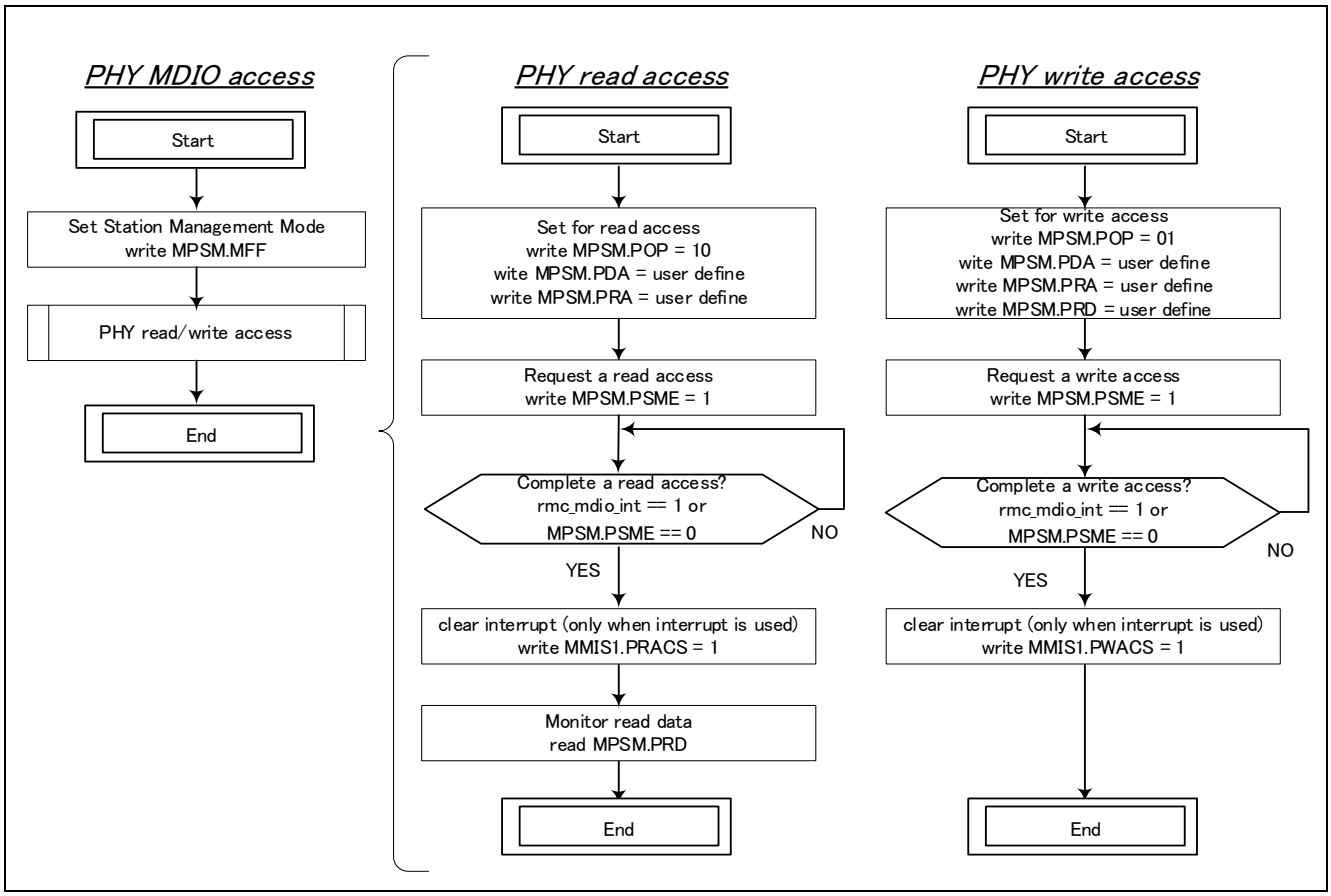



Figure 5-6 **PHY MDIO access flow**

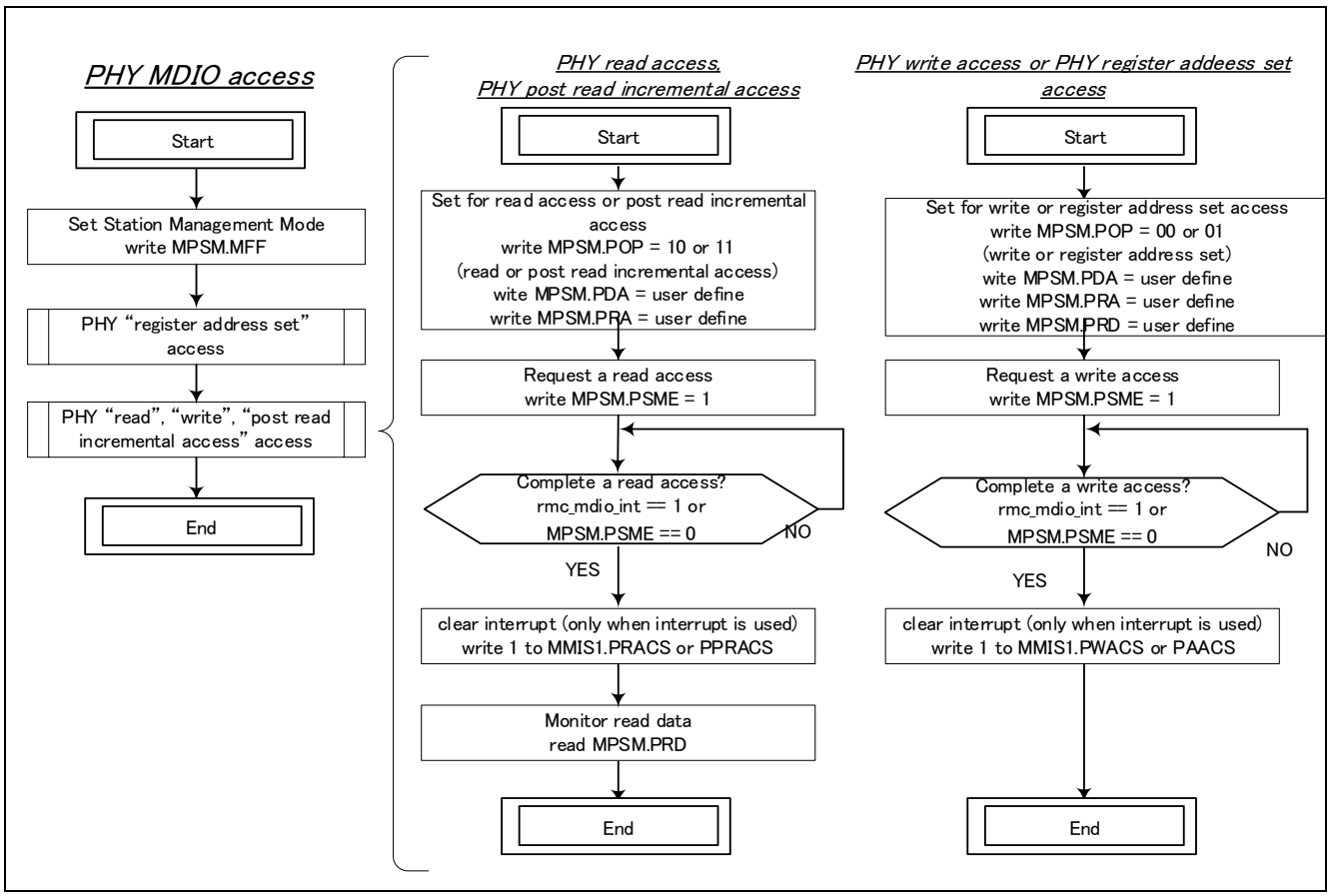



Figure 5-7 **PHY eMDIO access flow**

### 5.2.5 LPI transition for Tx port

The figure below shows the LPI transition flow. In case of resumed from LPI, the transmission should be requested until PHY wake-up completed.

Restrictions:
- This flow is not usable in RESET modes.

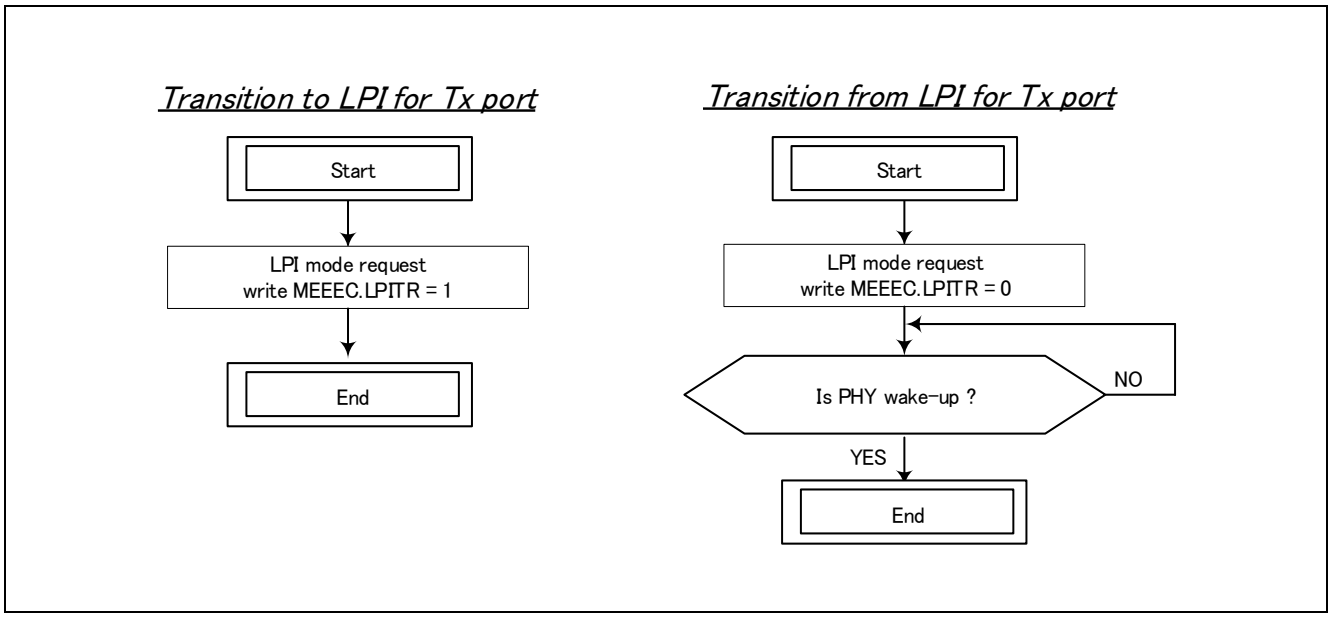



Figure 5-8 LPI transition for Tx port

### 5.2.6 LINK Verification

The figure below shows Link Verification flow.

Restrictions:
- This flow is usable in OPERATIONAL mode.

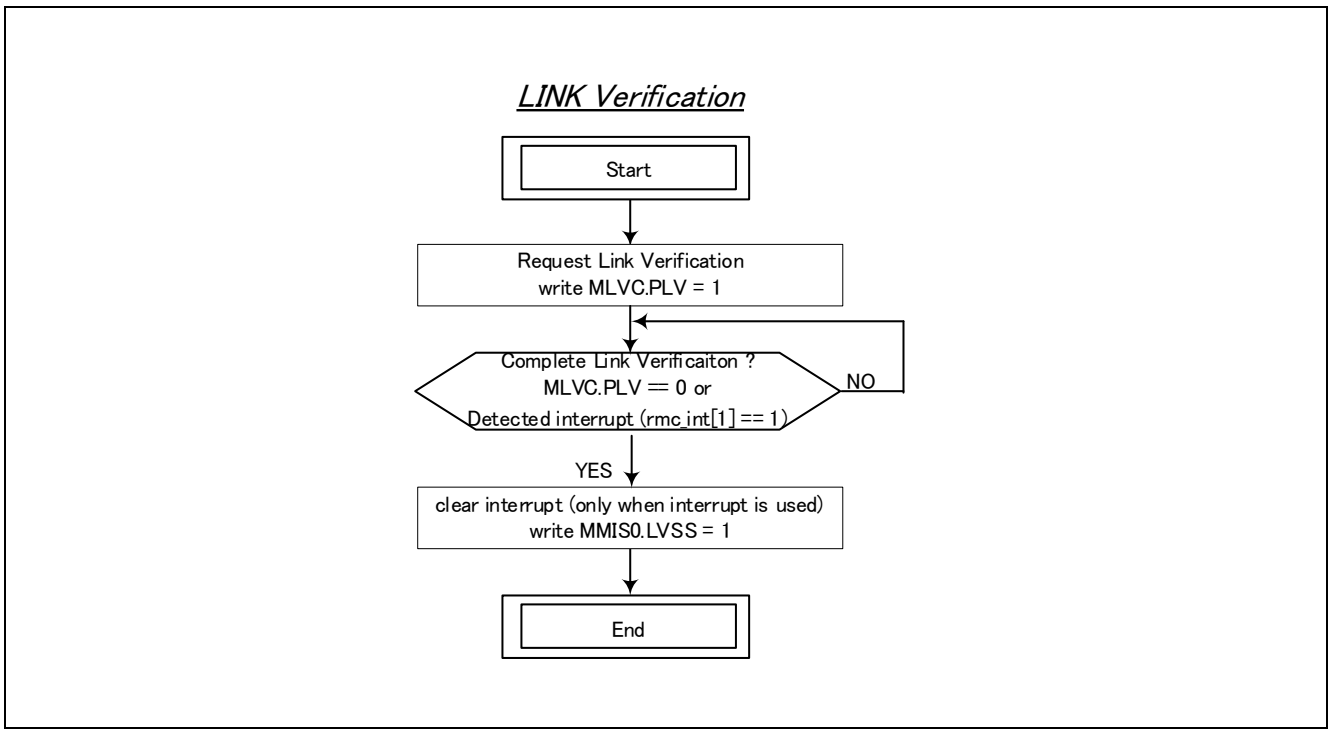



Figure 5-9 **Link Verification flow**

### 5.2.7 Called software flows

The flows described in this section can only be called from other flows thanks to a “flow link” box (Figure 5-3 ) and cannot be used alone.

## 5.3 Register writable without software flow

These registers can be changed dynamically. (However, it is necessary that the initial settings such as the clock enabling have been completed.)

- MRMAC0 and MRMAC1 (if in CONFIG)
- Other (not been described so far)

# 6.Functional details

## 6.1 Preemption

"Preemption" is the function specified in IEEE802.3.br. This function separates an ethernet frame to several fragments like Figure 6-1. This figure shows an example of preemption of 3 fragments. Preemption for only two fragments or over three fragments is possible.

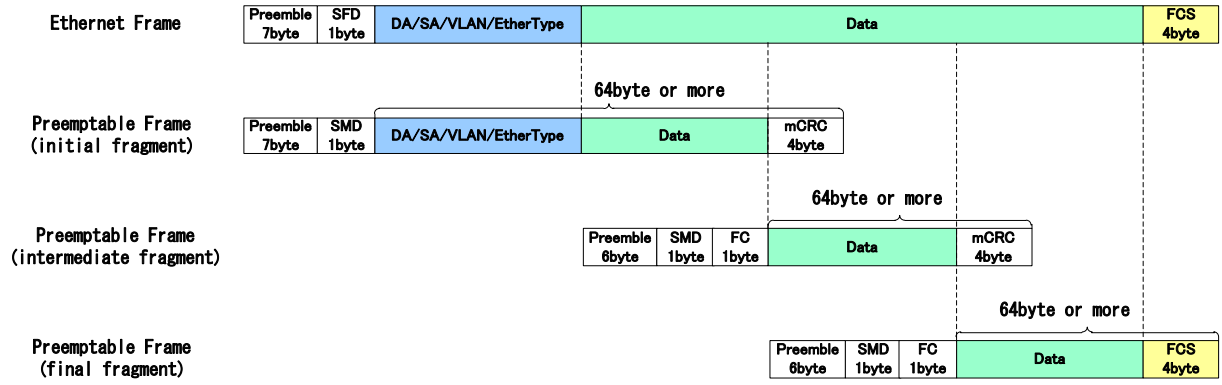



Figure 6-1 Outline of preemption

This function is handled by this IP and an upper module (e.g. MHD).

On the Tx Side, This IP only generates the fields of "Preamble", "SMD"," FC" and "mCRC" of the Preemptable frame complying with the rule of IEEE802.3.br. The upper module (e.g. MHD) has to separate a transmit frame appropriately.

On the Rx Side, this IP removes the fields of "Preamble" , "SMD" , "FC" and "mCRC" of the Preemptable frame and outputs the fields of "DA/SA/VLAN/EtherType", "Data" and "FCS"(To outputs FCS part whether or not is selectable by MRGC.RCPT) to upper module. The upper module (e.g. MHD) has to re-assemble a received frame.

"Preemption" is effective to prevent decreasing of throughput for high priority frame by transferring of low priority frames. Figure 6-2 shows one of examples. The software doesn't care the length of Low priority frames.

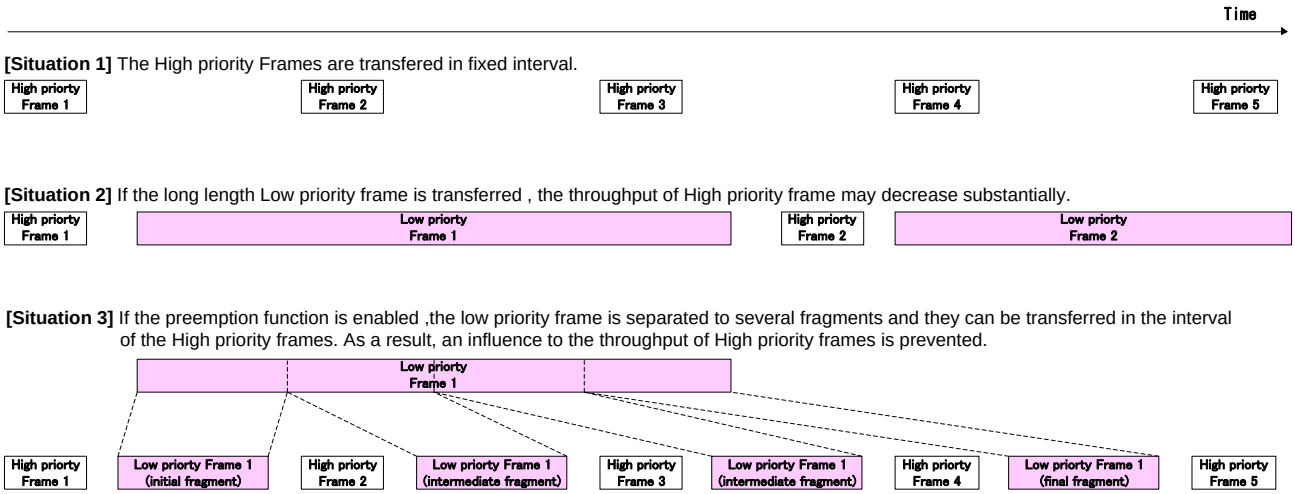



Figure 6-2 Effective of Preemption

To use “Preemption”, the Link Verification process should be executed on the system level.

Note: This IP doesn’t hold an information whether the preemption function is valid or not in the system.

### 6.1.1 Link Verification

This IP supports Link Verification process of Figure 6-3.

By setting MLVC.PASE this IP answers to a Verify frame by a Response frame to authorize the linked IP that preemption is possible. When a valid Verify frame is received, MMIS.VFRS is asserted as an interrupt.

When MLVC.PLV is set to 1, this IP starts to transmit Verify frame. The transmission attempts 3 times with the configured interval (MLVC.LVT) until Response frame is received from remote node. If the received Response frame is valid, MMIS.LVSS is asserted as an interrupt.

If not receiving Response frame after attempted 3 times, MMIS.LVFS is asserted as an interrupt.

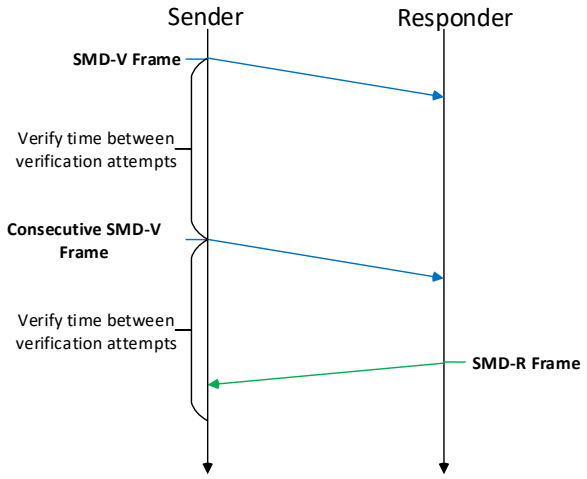



Figure 6-3 Example of Link Verification flow

### 6.1.2 PCH (Packet Control Header)

This IP support “PCH” function specified in “USXGMII” specification. [Restriction] PCH cannot use (as timestamping) in Non-XGMII interfaces.

On Tx side, this IP selects a Tx Frame format by the configuration of MPCH.TXPCH_M whether PCH frames or non PCH frames like following Figure 6-4.

On Rx side, this IP receives both PCH frames and non PCH frames without any configurations. If CRC of PCH field in received frames is incorrect, “Rx PCH CRC Error” field in follow is set to 1.

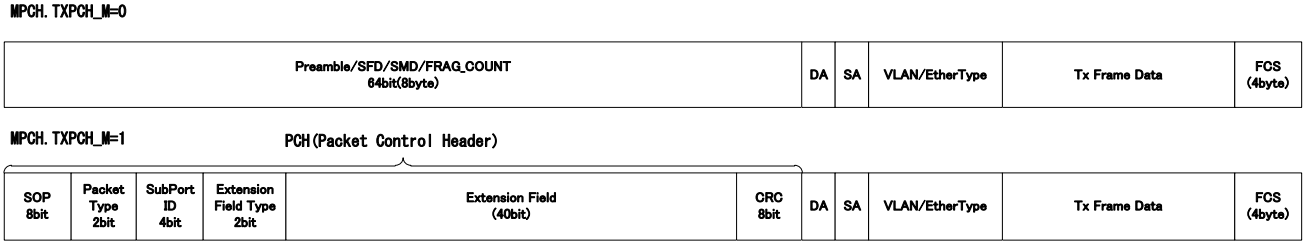



Figure 6-4 Packet Control Header

When MPCH.TXPCH_M is set to 1, the specific frame formats for any kind of frame are followings.

eframe without Extension field(MPCH.TXPCH_ETYPE[1:0]=2'b00)

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | DA | SA | VLAN/EtherType | Tx Frame Data | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | | | | |
| 0xfb | 2'b00 | 4bit | 2'b00 | 2'b00 | 2'b00 | 2'b00 | 2'b00 | 1'b0 | All 0 | All 0 | 8bit | | | | | (4byte) |

MPCH.TXPCH_PID[3:0]

eframe without preemption function(MPCH.TXPCH_ETYPE[1:0]=2'b01)

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | DA | SA | VLAN/EtherType | Tx Frame Data | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | | | | |
| 0xfb | 2'b00 | 4bit | 2'b01 | 2'b00 | 2'b00 | 2'b00 | 2'b00 | 1'b0 | All 0 | | 8bit | | | | | (4byte) |

MPCH.TXPCH_PID[3:0]

"Timestamp unique number" of tx_status from Tx Descriptor

eframe with preemption fucntion(MPCH.TXPCH_ETYPE[1:0]=2'b10)

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | DA | SA | VLAN/EtherType | Tx Frame Data | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | | | | |
| 0xfb | 2'b00 | 4bit | 2'b10 | 2'b00 | 2'b00 | 2'b00 | 2'b00 | | All 0 | | 8bit | | | | | (4byte) |

MPCH.TXPCH_PID[3:0]

depend on SFR (MPCH.IETPTE, MPCH.CTPTE) and TxStatusInfo(Bit2,3)

"Timestamp unique number" of tx_status from Tx Descriptor

pframe(initial fragment)

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | DA | SA | VLAN/EtherType | Tx Frame Data | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | | | | |
| 0xfb | 2'b00 | 4bit | 2'b10 | 2'b01 | 2'b00 | 0 to 3 | 0 to 3 | | All 0 | | 8bit | | | | | (4byte) |

MPCH.TXPCH_PID[3:0]

depend on SFR(MPCH.CTPTE) and TxStatusInfo(Bit2)

"Timestamp unique number" of tx_status from Tx Descriptor

pframe(intermediate or last fragment)

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | DA | SA | VLAN/EtherType | Tx Frame Data | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | | | | |
| 0xfb | 2'b00 | 4bit | 2'b10 | 2'b10 | 2'b00 | 0 to 3 | 0 to 3 | 1'b0 | All 0 | All 0 | 8bit | | | | | (4byte) |

MPCH.TXPCH_PID[3:0]

Verify Packet

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | 60byte | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | |
| 0xfb | 2'b00 | 4bit | 2'b10 | 2'b11 | 2'b00 | 2'b00 | 2'b00 | 1'b0 | All 0 | All 0 | 8bit | All 0 | (4byte) |

MPCH.TXPCH_PID[3:0]

Respond Packet

| SOP | Packet Type | SubPort ID | Extension Field Type | Extension Field (40bit) | | | | | | | CRC | 60byte | FCS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | [39:38] | [37:36] | [35:34] | [33:32] | [31] | [30:16] | [15:0] | | | |
| 0xfb | 2'b00 | 4bit | 2'b10 | 2'b11 | 2'b01 | 2'b00 | 2'b00 | 1'b0 | All 0 | All 0 | 8bit | All 0 | (4byte) |

MPCH.TXPCH_PID[3:0]

Figure 6-5 Tx Frame formats

## 6.2 Flow Control

This IP supports the following two types of Flow Control functions.

1) Flow Control specified in IEEE 802.3 for a full-duplex operation using a PAUSE frame.
2) Priority Based Flow Control specified in IEEE 802.1Qbb for a full-duplex operation using a PFC frame.

### 6.2.1 Flow Control by PAUSE frame

This Flow Control is only for a full-duplex operation and this consists of Transmission and Reception of a PAUSE frame.

**Transmission of PAUSE frame**

During MTFFC.FCM=0 and MTPFC.PT[15:0]≠0, the transmission function of a PAUSE frame is enabled. This IP transmits a PAUSE frame as described in Figure 6-6 by asserting a transmission request. The value of MTPFC.PT [15:0] is used for “PAUSE_TIME” field. And MRMAC0 and MRMAC1 are used for “SA” field.

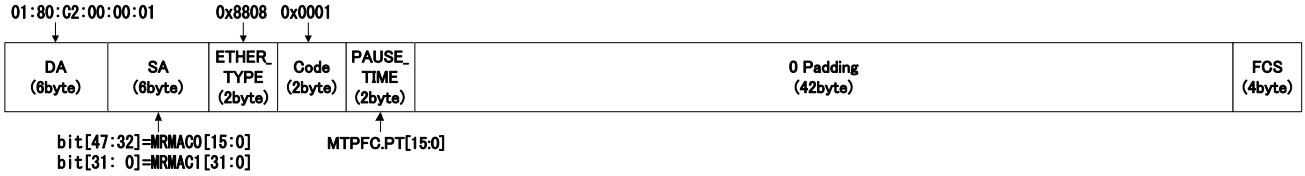



Figure 6-6 PAUSE Frame Format

■Auto PAUSE Frame

In MTPFC.PFM=0, this IP uses a hardware pause (**PAUSE[0]** interface) from [COMA] transmission requests a transmit request. After transmitting the PAUSE Frame, MAPFTCT is incremented.

■Manual PAUSE Frame

In MTPFC.PFM=1, this IP uses MTPFC2.MPFR as a transmit request. After transmitting the PAUSE Frame, MMPFTCT is incremented.

When the transmit request of the PAUSE frame is asserted while the IP is transmitting the other frame, the PAUSE frame is transmitted after the transmitting frame.

(1) While the transmit request is asserted, a PAUSE frame is continuously re-transmitted at a fixed interval of time configured by MTPFC.PFRT. If the number of re-transmission reaches the value of MTPFC.PFRLV, this IP sets MEIS.PFRROS to 1 for an interrupt. Even if the interrupt is detected, this IP continues re-transmission of a PAUSE frame while the transmit request is asserted.

(2) After de-asserting the transmission request, this IP stops a transmission of PAUSE frame. At that time, when MTPFC2.PFTTZ=1 and an internal counter to count a pausing-period exceeds the value of MTPFC.PFRT (*) , this IP transmits one more a PAUSE frame with “PAUSE_TIME” field as 0.

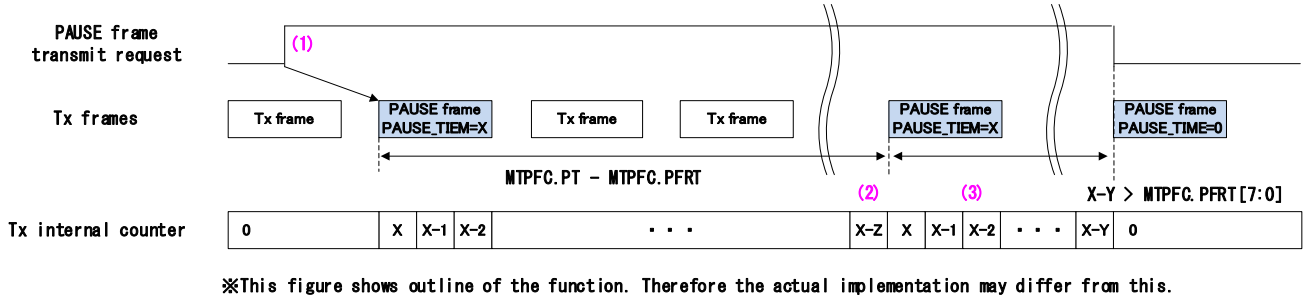



**Figure 6-7** Transmission of PAUSE frame

**Reception of PAUSE frame**

During MRGC.PFRC=1, this IP receives a PAUSE frame and it starts internally counting a pausing-period specified by “PAUSE_TIME” field of a received PAUSE frame. While counting the pausing-period, notification of pause frame reception to an upper layer (e.g. MHD) and MRPFM.PTCA is asserted. And any transmissions except for a Verify_packet , a Respond_packet and a PAUSE frame are inhibited. When this IP receives a PAUSE frame while transmitting any frames, transmissions are inhibited after the transmission completion.

(1) When a PAUSE frame with “PAUSE_TIME≠0” is received again before expiring the internal pausing-period, the internal counter is updated to that value.

(2) When the internal pausing-period is expired, notification of pause frame reception to an upper layer (e.g. MHD) and MRPFM.PTCA is de-asserted.

(3) When MRGC.PFRTZ=1 and PAUSE frame with “PAUSE_TIME=0” is received before expiring the internal pause-period, the internal counter is cleared. Then notification of pause frame reception to an upper layer (e.g. MHD)and MRPFM.PTCA is de-asserted.

In any case, MPFRCT is incremented after receiving a PAUSE frame in MRGC.PFRC=1. And the received PAUSE frames are discarded internally.

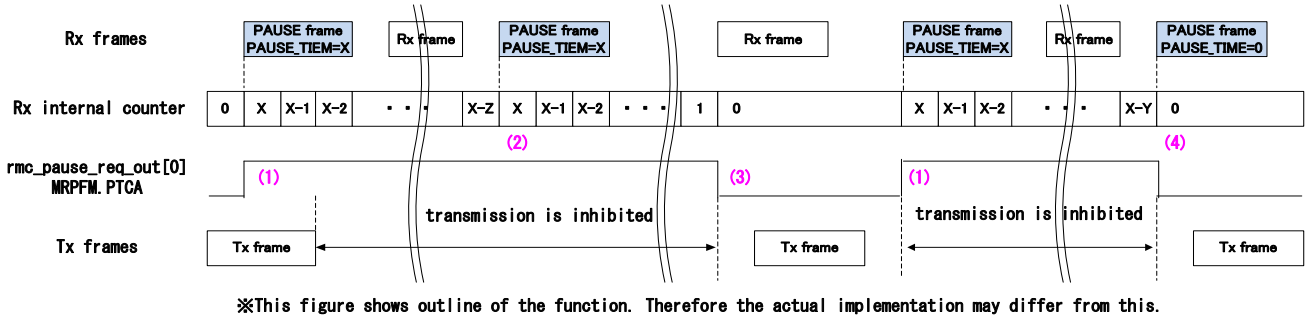



Figure 6-8 Reception of PAUSE Frame

### 6.2.2 Flow Control by PFC frame

This Flow Control is for a full-duplex operation and this consists of Transmission and Reception of a PFC frame.

**Transmission of PFC frame**

InMTFFC.FCM=1 and MTPFC.PT[15:0]≠0, the transmission function of a PFC frame is enabled.
This IP treats several priorities as one priority-group to transmission of the PFC frame. The number of priority-group is configured by hardware parameter “PAS_LVL_N” (pause level) and all descriptions in this section are written as PAS_LVL_N=2. Each priority-group is configured by MTPFC30 and MTPFC31.

This IP transmits a PFC frame as described in Figure 6-9 for each priority-group by asserting a transmission request. The value of MTPFC.PT [15:0] is used for all “PAUSE_TIME” fields. “Priority_Enable” field value is different for each priority-group and the value of MTPFC30.PFCPG and MTPFC31.PFCPG are used for these. And MRMAC0 and MRMAC1 are used for “SA” field. When MTPFC30.PFCPG (or MTPFC31.PFCPG) is set to all 0,this IP doesn’t transmit a PFC frame for the priority-group0 (or priority-group1) even when a transmission request is asserted.

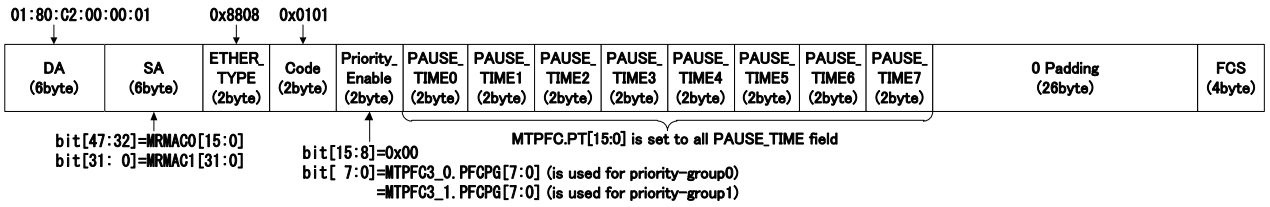



Figure 6-9 PFC frame Format

■Auto PFC Frame

InMTPFC.PFM=0, this IP uses a hardware pause (**PAUSE[PAS_LVL_N-1:0]** interface) from [COMA] transmission request as a transmit request. The [COMA] PAUSE interface bit[0] is for priority-group0 and the [COMA] PAUSE interface bit[1] is for priority-group1. MAPCFTCT0 and MAPCFTCT1 are incremented respectively after transmitting a PFC Frame of each priority-group.

■Manual PFC Frame

InMTPFC.PFM=1, this IP uses MTPFC2.MPFCFR[1:0] as a transmit request. The [COMA] PAUSE interface bit[0] is for priority-group0 and the[COMA] PAUSE interface bit[1] is for priority-group1. MMPCFTCT0 and MMPCFTCT1 are incremented respectively after transmitting a PFC Frame of each priority-group.

If several transmit requests are asserted at same time, one PFC frame which has merged priority_enable information is transmitted.

If several transmit requests are asserted while transmitting other frames, an internal counter is cleared and one PFC frame which has merged priority_enable information is transferred.

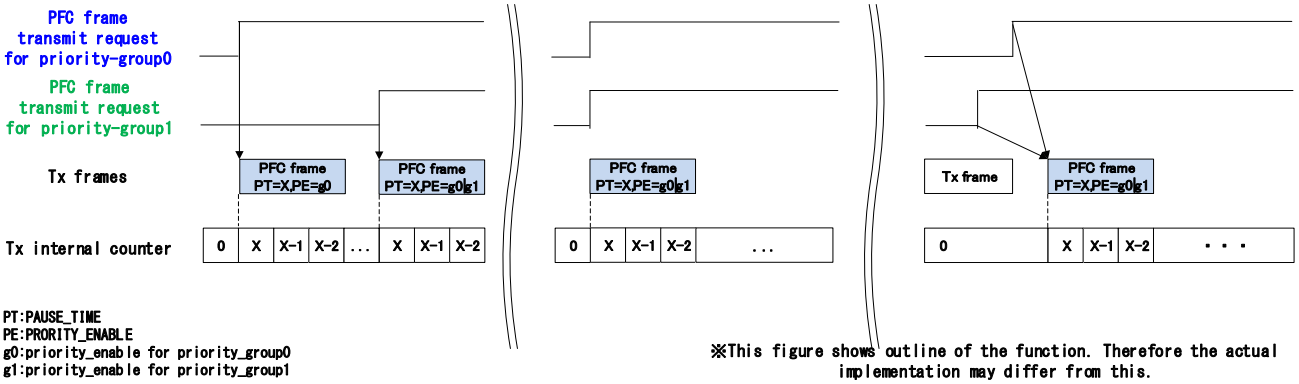



**Figure 6-10 Transmitting PFC frame (part1)**

While the transmit request is asserted, a PFC frame corresponding to that request is continuously re-transmitted at a fixed interval configured by MTPFC.PFRT. When the number of re-transmissions for one of priority-groups reaches the value of MTPFC.PFRLV, this IP sets MEIS.PFRROS to 1 for an interrupt. Even when the interrupt is detected, this IP continues re-transmission of a PFC frame while the transmit request is asserted.

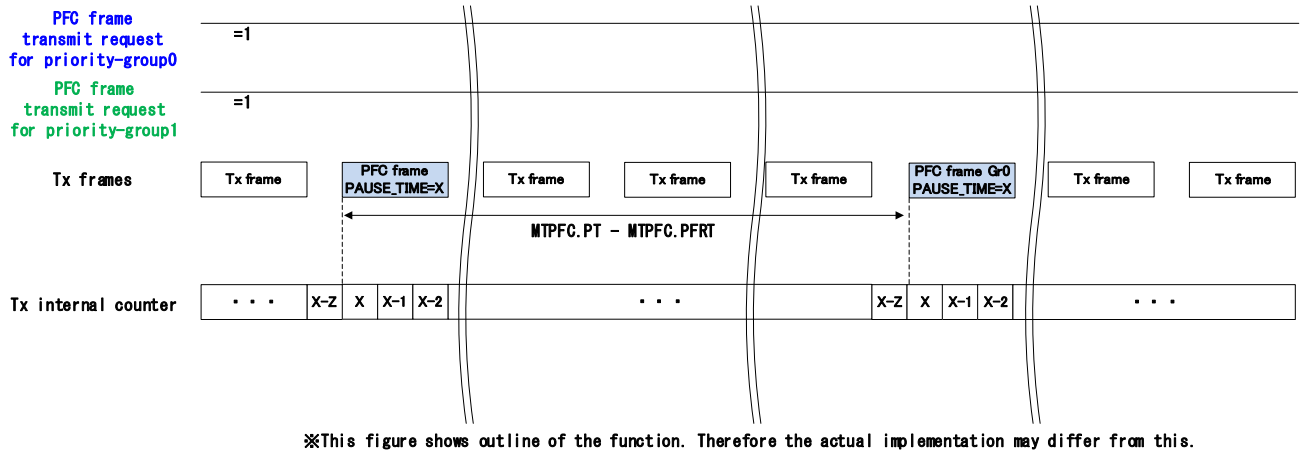



Figure 6-11 Transmitting PFC frame (part2)

After de-asserting the transmission request, this IP stops the transmission of the PFC frame corresponding to that request. At that time, if the corresponding bit of MTPFC2.PFCTTZ[1:0] for de-asserted priority-group request is 1 and an internal counter to count the pausing-period exceeds MTPFC.PFRT (*), this IP transmits one more PFC frame with all “PAUSE_TIME” fields as 0 for that priority-group.

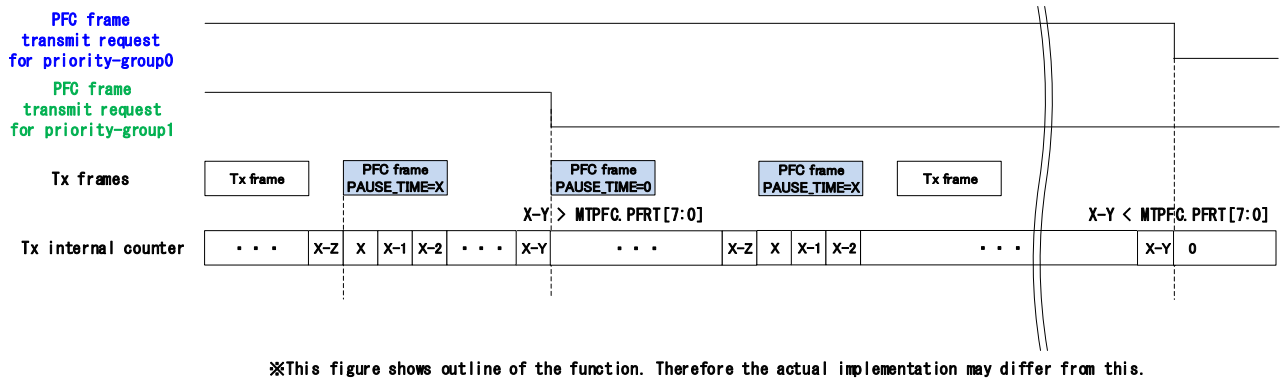



Figure 6-12 Transmitting PFC frame (part3)

**Reception of PFC frame**

This IP treats several priorities specified in received PFC frames. The number of supported priorities is configured by hardware parameter “FRM_PRIO_N” and all descriptions in this section are written as FRM_PRIO_N=8.

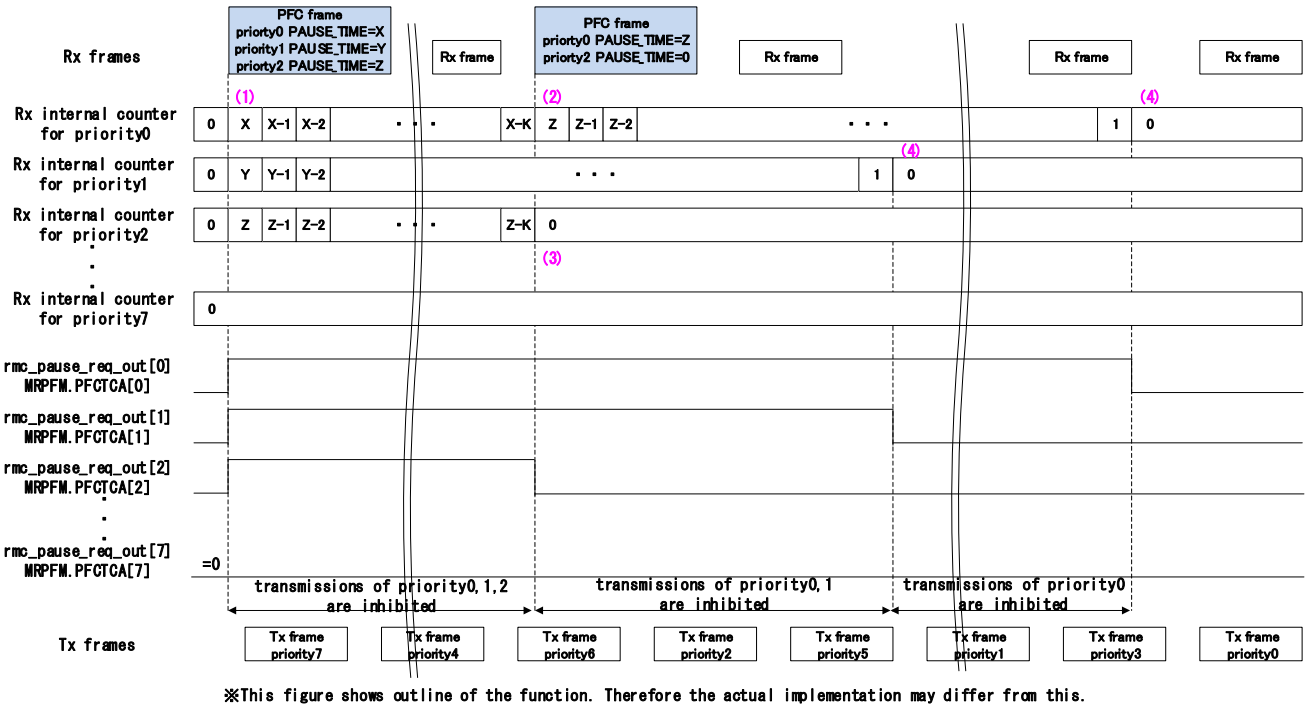
(1) When this IP receives a PFC frame with a priority-enable field which is corresponding to apriority configured by MRGC.PFCRC[7:0], it starts internally counting a pausing-period for each priority respectively specified by each “PAUSE_TIME” fields of a received PFC frame. During counting the pausing-period, notification of PFC frame reception to an upper layer (e.g. MHD)and MRPFM.PFCTCA[7:0] corresponding to each priority are asserted. The bit number is corresponding to priority number.

(2) When PFC frame with “PAUSE_TIME≠0” with corresponding to valid “priority_enable” field is received again before expiring the internal pausing-period, the internal counter for that priority is updated to that value.

(3) When MRGC.PFRTZ=1 and PFC frame with “PAUSE_TIME=0” with corresponding to valid “priority_enable” field is received before expiring the internal pause-period, the internal counter for that priority is cleared and notification of pause frame reception to an upper layer (e.g. MHD) and MRPFM.PFCTCA[7:0] for that priority is de-asserted.

(4) When the internal pausing-period is expired, notification of pause frame reception to an upper layer (e.g. MHD) and MRPFM.PFCTCA[7:0] for that priority are de-asserted.

In any case, MPCFRCT0 to MPCFRCT7 are incremented respectively after receiving a PFC frame for enabled priority by MRGC.PFCRC[7:0]. And all received PFC frames are discarded internally.




※This figure shows outline of the function. Therefore the actual implementation may differ from this.

Figure 6-13 Reception of PAUSE Frame

## 6.3 Timestamping

This IP supports time synchronization for IEEE 1588. It supports time synchronization accuracy in the sub-microsecond range.

### 6.3.1 Timestamp capture

This IP can capture timestamps on both TX and RX side. gPTP timer value includes two 62bit timers, bit[61:0] is a Timer0, bit[123:62] is a Timer1. PTP_TN and PTP_TUNES are Hardware parameters which are used for this function. All descriptions in this section are written as PTP_TN=2, PTP_TUNES=3.

**【Timestamp capture on TX side】**
“Timestamp capture” and “Timer number” fields in follow. control this function for both e-frame and p-frame. These are configured by a Tx Descriptor. If “Timestamp capture” =1, the timer specified by “Timer number” field is captured.

The captured timestamp is sent to an upper module(e.g. MHD) with “Timestamp unique number” in follow via MHD Tx timestamp capture Interface..

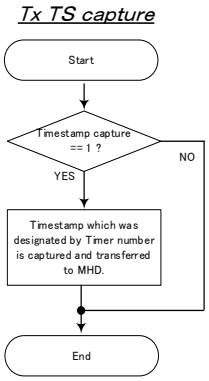



Figure 6-14 Tx Timestamp capture flow

In MPCH.TXPCH_M=1, PHY will do timestamping used by signature information in PCH. This is a “PCH” function specified in USXGMII specification. Please refer to PHY specification.

The length of the captured timestamp is fixed as 4bytes.
“Timestamp unique number” in follow is used as a “Signature” information of “PCH”.

Note: On the TX Side, the timestamp capture function by this IP and PHY can be enabled at same time.

**【Timestamp capture on RX side】**

InMPCH.RXPCH_TSM=0, this IP outputs a timestamp captured by this IP.
MTRC.TRDDE, MTRC.TRDDP, and MTRC.TRHFME[1:0] control this function.
The configuration of the timestamp capture is following.

- MTRC.TRDDE controls this function for e-frames.
  ・MTRC.TRDDE=0
  the default timer configured by MTRC.DTN is captured for a received e-frame.

- MTRC.TRDDP controls this function for p-frames.
  ・MTRC.TRDDP=0
  the default timer configured by MTRC.DTN is captured for a received p-frame.

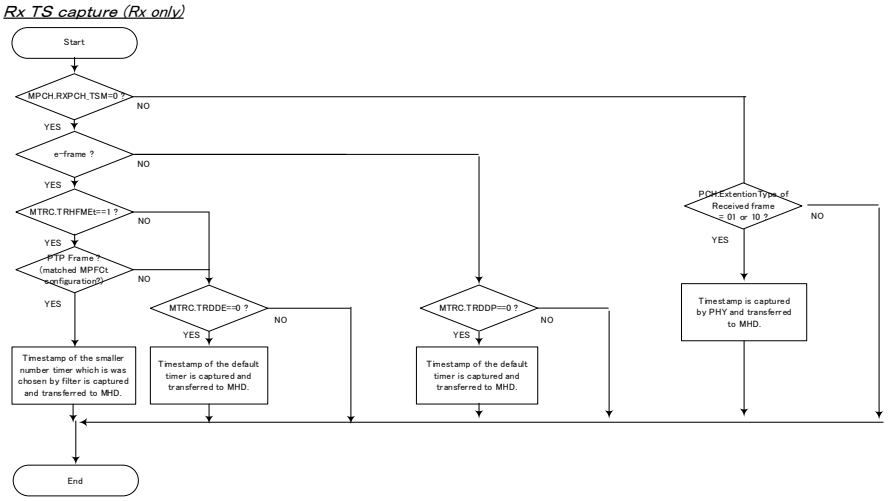
- MTRC.TRHFME[1:0] controls this function for e-frames using the HW PTP message filtering.
  The bit[0] is for Timer0 and bit[1] is for Timer1.
  ・TRHFME[1:0]=2'bx1 and the received e-frame matched the HW PTP messaged filter for Timer0
  The Timer0 is captured for that received e-frames.

  ・TRHFME[1:0]=2'b10 and the received e-frame matched the HW PTP messaged filter for Timer1
  The Timer1 is captured for that received e-frames.

  ・TRHFME[1:0]=2'b11 and the received e-frame matches the HW PTP messaged filter for Timer0 and 1
  The Timer0 is captured for that received e-frames.

A captured timestamp is outputted from this IP with the received frame.

InMPCH.RXPCH_TSM=1, this IP outputs a timestamp captured by PHY. This is a function of "PCH" specified in "USXGMII" specification.
The length of the captured timestamp is fixed as 4bytes.
A captured timestamp is outputted from this IP with the received frame.



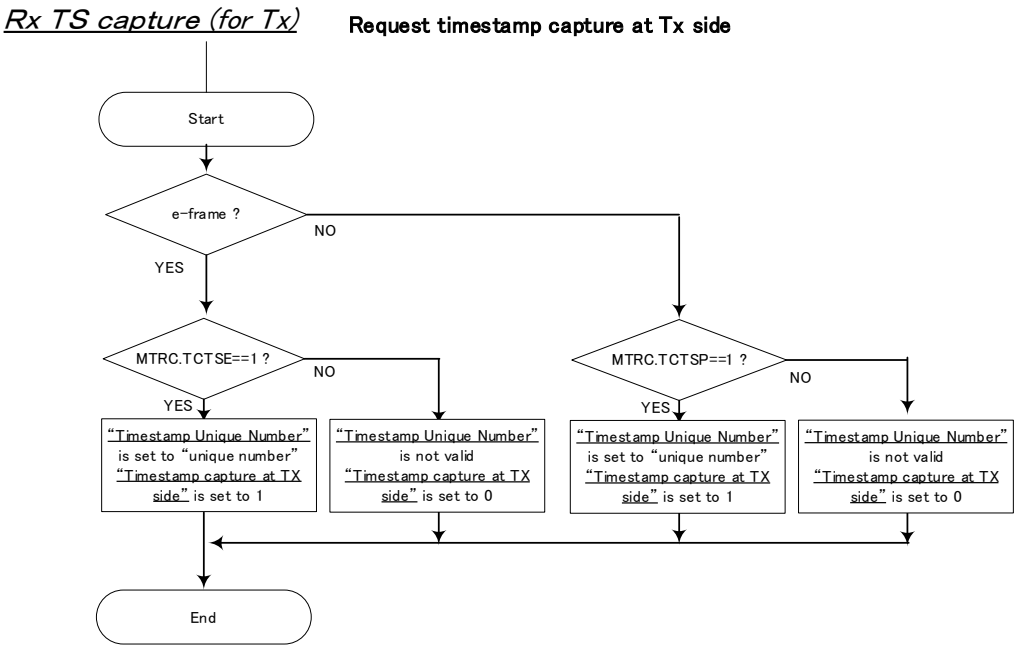




Figure 6-15 Rx Timestamp capture flow

**【Configuration of HW PTP Filter】**

HW PTP message filter is configured by using MPFC0-15. The Figure 6-16 shows an example configuration of PTP Message Filter. This is a case of MTRC.TRHFME[1:0] = 2'b11.

In this case:

MPFC0-7 are used as the filters for both Timer0 and Timer1. Because TEF field is set to 0x03.

MPFC8 is used as the filter for Timer0. Because TEF [8] field is set to 0x01.

MPFC9 is used as the filter for Timer1. Because TEF [9] field is set to 0x02.

MPFC10-15 are not used. Because TEF field is set to 0x00.

If an e-frame matches all MPFC0-7 and MPFC8, it is judged as a frame of Timer0 domain.

If an e-frame matches all MPFC0-7 and MPFC9, it is judged as a frame of Timer1 domain.

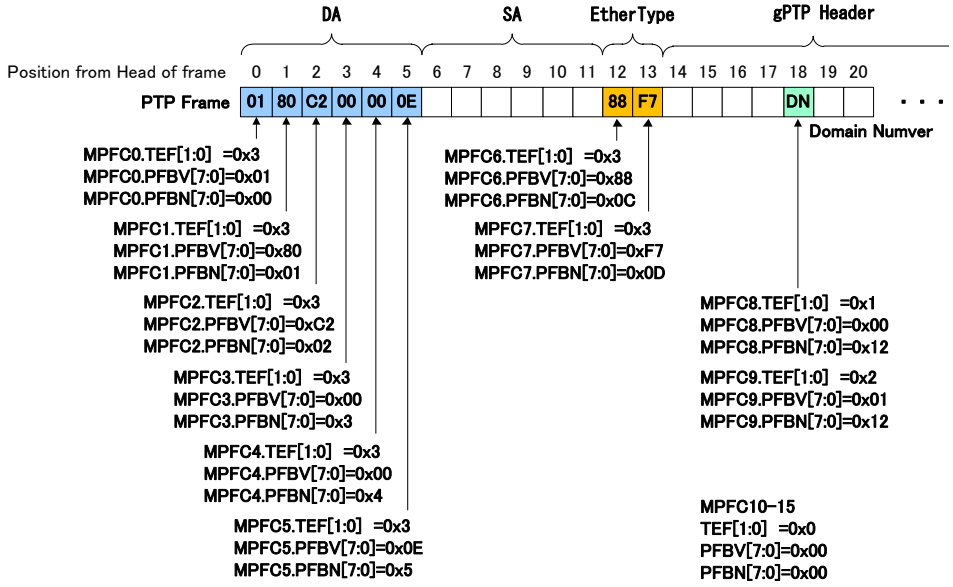



Figure 6-16 Example configuration of PTP Message Filter

[Note]

- Depending on the configuration of the filter, a frame may match both filters. In this case, Timer0 is selected.
- PFBN should not select a position that exceeds the end of frame. Therefore, this value less than or equal to the minimum frame size is recommended.

## 6.4 Station Management

The Station Management controls the access to Management registers in PHY.
rmc_mdi, rmc_mdo, rmc_mde, rmc_mdc and rmc_mdio_int are used for this function.

### 6.4.1 Management register access

This IP has two types of access methods, one is specified by clause 22 the other is specified by clause 45 in IEEE802.3.

#### Access Method specified in clause 22

To write or read of Management register of PHY which supports this access method using management frame format, execute the following steps.

**【write access】**
Set MPSM as following.
PRD[15:0]=”Write Data”
POP[1:0]=2’b01
PRA[4:0]=”Register Address”
PDA[4:0]=”PHY Address ”
MFF=0
PSME=1
After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PWACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
At this time, when MMIE1.PWACE is set to 1, the interrupt appears on rmc_mdio_int.
Set MMIS1.PWACS to 0 to clear the interrupt.

**【Read access】**
Set MPSM as following.
PRD[15:0]=”16’h0000”
POP[1:0]=2’b10
PRA[4:0]=”Register Address ”
PDA[4:0]=”PHY Address”
MFF=0
PSME=1
After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PRACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
At this time, when MMIE1.PRACE is set to 1, the interrupt appears on rmc_mdio_int.
After the completion of read access, the read data is stored in MPSM.PRD[15:0]. So read it.
Set MMIS1.PRACS to 0 to clear the interrupt.

Note:
When MPSM.PSME is set to 1, this IP outputs preamble code of fixed length specified in standard on rmc_mdo before write or read access. IEEE802.3 clause 22 allows skipping preamble code. When the preamble code isn’t needed and PHY also supports to skip the preamble code, this IP doesn’t output the preamble code by setting MPIC.PSMDP to 1.

### Access Method specified in clause 45

To write or read Management register of PHY which supports this access method using extended management frame format, execute the following steps.

**【write access】**

1) Set MPSM as following. (“Register Address” is set to a PHY)
   PRD[15:0]=”Register Address”
   POP[1:0]=2’b00
   PRA[4:0]=”Device Address ”
   PDA[4:0]=”Port Address”
   MFF=1
   PSME=1
   After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PAACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
   At this time, when MMIE1.PAACE is set to 1, the interrupt appears on rmc_mdio_int.
   Set MMIS1.PWACS to 0 to clear interrupt.

2) Set MPSM as following. (“Write Data” is set to the “Register Address” which was configured by step 1) )
   PRD[15:0]=”Write Data”
   POP[1:0]=2’b01
   PRA[4:0]=”Device Address ”
   PDA[4:0]=”Port Address”
   MFF=1
   PSME=1
   After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PWACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
   At this time, if MMIE1.PWACE is set to 1, the interrupt appears on rmc_mdio_int.
   Set MMIS1.PWACS to 0 to clear interrupt.

**【Read access】**

1) Set MPSM as following. (“Register Address” is set to a PHY)
   PRD[15:0]=”Register Address”
   POP[1:0]=2’b00
   PRA[4:0]=”Device Address ”
   PDA[4:0]=”Port Address”
   MFF=1
   PSME=1
   After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PAACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
   At this time, when MMIE1.PAACE is set to 1, the interrupt appears on rmc_mdio_int.
   Set MMIS1.PWACS to 0 to clear interrupt.

2) Set MPSM as following. (Read access of the Register Address which was pointed by previous step)
   PRD[15:0]=”16’h0000”
   POP[1:0]=2’b11
   PRA[4:0]=”Device Address ”
   PDA[4:0]=”Port Address”
   MFF=1
   PSME=1
   After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PRACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
   At this time, when MMIE1.PRACE is set to 1, the interrupt appears on rmc_mdio_int.
   After the completion of read access, the read data is stored in MPSM.PRD[15:0]. So read it.

Set MMIS1.PRACS to 0 to clear interrupt.

**【Post read incremental access】**

Read access needs setting ”Register Address” every time. Only when continuous Register Address fields are read continuously, using this access (POP[1:0]=10) is efficient.

1) Set MPSM as following. (“Register Address” is set to a PHY)
   PRD[15:0]=”Register Address”
   POP[1:0]=2’b00
   PRA[4:0]=”Device Address ”
   PDA[4:0]=”Port Address”
   MFF=1
   PSME=1
   After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PAACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
   At this time, when MMIE1.PAACE is set to 1, the interrupt appears on rmc_mdio_int.
   Set MMIS1.PWACS to 0 to clear the interrupt.

2) Set MPSM as following. (Read access of the Register Address which was pointed by previous step)
   PRD[15:0]=”16’h0000”
   POP[1:0]=2’b10
   PRA[4:0]=”Device Address ”
   PDA[4:0]=”Port Address”
   MFF=1
   PSME=1
   After setting MPSM, PSME is kept to 1 during the access. Wait an interrupt is detected (MMIS1.PPRACS=1) or MPSM.PSME turns to 0 as a sign of an access completion.
   At this time, when MMIE1.PPRACE is set to 1, the interrupt appears on rmc_mdio_int.
   After the completion of read access, the read data is stored in MPSM.PRD[15:0]. So read it.
   And ”Register Address” set by step 1) is automatically changed in the PHY to point a next register address field. So don’t need to re-configuration of “Register Address” for next read access.
   Set MMIS1.PPRACS to 0 to clear the interrupt.

3) Repeat step 2) for the read access from the Next Register Address field.

Note:
When MPSM.PSME set to 1, this IP outputs preamble code of fixed length specified in standard on rmc_mdo before write or read access. IEEE802.3 clause 45 doesn’t allow skipping the preamble code. But skipping preamble code is also supported in this access method by setting MPIC.PSMDP to 1. To use this function, check a PHY specification.

### 6.4.2 Management Data Clock (MDC) setting

To access Management registers in PHY, this IP outputs MDC. This is generated by dividing CHI clock (clk) in this IP and outputted from rmc_mdc. The division ratio can be set by the MPIC.PSMCS.

The following table shows the programmed MDC frequency depending on MPIC.PSMCS. MDC frequency is calculated by clk[MHz] / ((MPIC.PSMCS +1)*2).

**Table 6-1 Programming example for MPIC.PSMCS**

| clk frequency | MPIC.PSMCS | MDC frequency |
|---|---|---|
| 100MHz | 0x13 | 2.5MHz |
| 200MHz | 0x27 | 2.5MHz |
| 300MHz | 0x3B | 2.5MHz |
| 400MHz | 0x4F | 2.5MHz |
| 500MHz | 0x63 | 2.5MHz |

Note: IEEE 802.3 specifies as the clock should be slower than 2.5MHz. But some PHY could operate with a higher frequency. Refer to the PHY specification for detail.

### 6.4.3 Management Data timing adjustment

This IP can adjust the drive timing of MDO (rmc_mdo and rmc_mde) and the capture timing of MDI (rmc_mdi). It may be used for the propagation delay on the PCB.

The drive timing could be adjusted by PHY Station Management Hold time adjustment (MPIC.PSMHT). The capture timing could be adjusted by PHY Station Management Capture time adjustment (MPIC.PSMCT). The figure below is an example shifted by 2clk cycles.

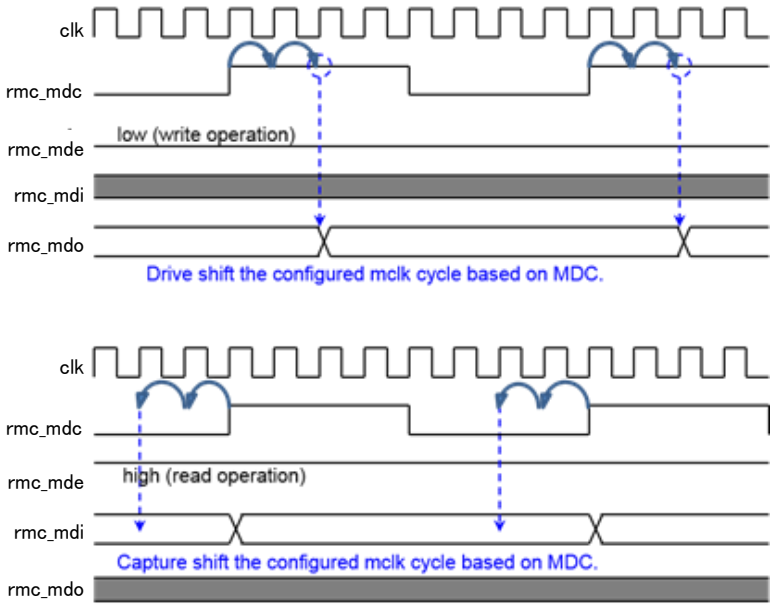



**Figure 6-17 Management Data timing adjustment**

## 6.5 Energy Efficient Ethernet

This IP supports Energy Efficient Ethernet (EEE) which is standardized inIEEE 802.3. Tx-LPI mode and Rx-LPI mode can be activated independently

### 6.5.1 LPI Control

This IP can request LPI mode for Tx port to the PHY when the LPI request bit (MEEEC.LPITR) is asserted. LPI mode is released when the bit is negated. The operation flow is showed in section 7.2.2.5 in detail.

And LPI mode for Rx port can be recognized by the reception LPI interrupt status bit (MMIS2.LPIAIS). When the bit (MMIS2.LPIDIS) is asserted, it means RMAC detected de-assertion of LPI.

The table below is LPI specification on each xMII interface. The specification for MII, GMII or XGMII is specified by IEEE802.3.

**Table 6-2 Rx LPI specification on each xMII interface**

| PHY IF | RX_DV | RX_ER | RXD |
|---|---|---|---|
| MII | rmc_rx_valid_gmii = 0 | rmc_rx_err_gmii = 1 | rmc_rx_data_gmii[3:0] = 0001 |
| GMII | rmc_rx_valid_gmii= 0 | rmc_rx_err_gmii = 1 | rmc_rx_data_gmii[7:0] = 00000001 |

**Table 6-3 Tx LPI specification on each xMII interface**

| PHY IF | TX_EN | TX_ER | TXD |
|---|---|---|---|
| MII | rmc_tx_valid_gmii = 0 | rmc_tx_err_gmii = 1 | rmc_tx_data_gmii [3:0] = 0001 |
| GMII | rmc_tx_valid_gmii = 0 | rmc_tx_err_gmii = 1 | rmc_tx_data_gmii [7:0] = 00000001 |

**Table 6-4 Tx / Rx LPI specification on XGMII interface**

| PHY IF | DATA | CMD |
|---|---|---|
| XGMII (Tx) | rmc_tx_data_xgmii[31:0] = 32’h06060606 | rmc_tx_cmd_xgmii [3:0] = 3’b1111 |
| XGMII (Rx) | rmc_rx_data_xgmii[31:0] = 32’h06060606 | rmc_rx_cmd_xgmii [3:0] = 3’b1111 |

## 6.6 MAC Address filtering on Reception

This IP supports the frame reception in a full-duplex operation mode.

This IP can pass all received Ethernet frames data to an upper layer via xMII Interface.

This IP strips off automatically SFDs/SMDs, preambles, FRAG_COUNTs and mCRCs from all the frames.

The FCS reception by the upper layer can be set though MRGC.RCPT register.

This IP identifies the frames with the special format and informs to the upper layer. It is indicated with the dedicated flag in status information followed payload data. The table below indicates the frames with the special format.

**Table 6-5 Operation for special format frame**

| Frame | Format | Operation |
|---|---|---|
| Broadcast frame | DA: FF_FF_FF_FF_FF_FF | Filtering possible. |
| Multicast frame | DA: B'xxxx_xxx1_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx | Filtering possible. |
| Pause frame | DA: 01_80_C2_00_00_01<br>TYPE: 88_08<br>OPCODE: 00_01 | This IP handles Flow Control, refer to section 8.2.<br>Pause information is not sent to upper layer. |
| PFC frame | DA: 01_80_C2_00_00_01<br>TYPE: 88_08<br>OPCODE: 01_01 | This IP handles Flow Control, refer to section 8.2.<br>Pause information is not sent to upper layer. |
| Magic packet | When the MAC address (MRMAC.MA) is 00-11-22-33-44-55h, this IP would scan for the data sequence:<br><br>Destination-Address Source-Address<br>.....MISC..... FF FF FF FF FF FF<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>00 11 22 33 44 55 00 11 22 33 44 55<br>.....MISC..... FCS<br>This sequence can be located anywhere in the Magic Packet payload. The Magic Packet is formed with a standard Ethernet header and FCS. | Interrupt.<br>Provide information to upper layer.<br>Provide magic packet interrupt (MMIS2.MPDIS).<br>Magic packets are only detected as E-frame. |

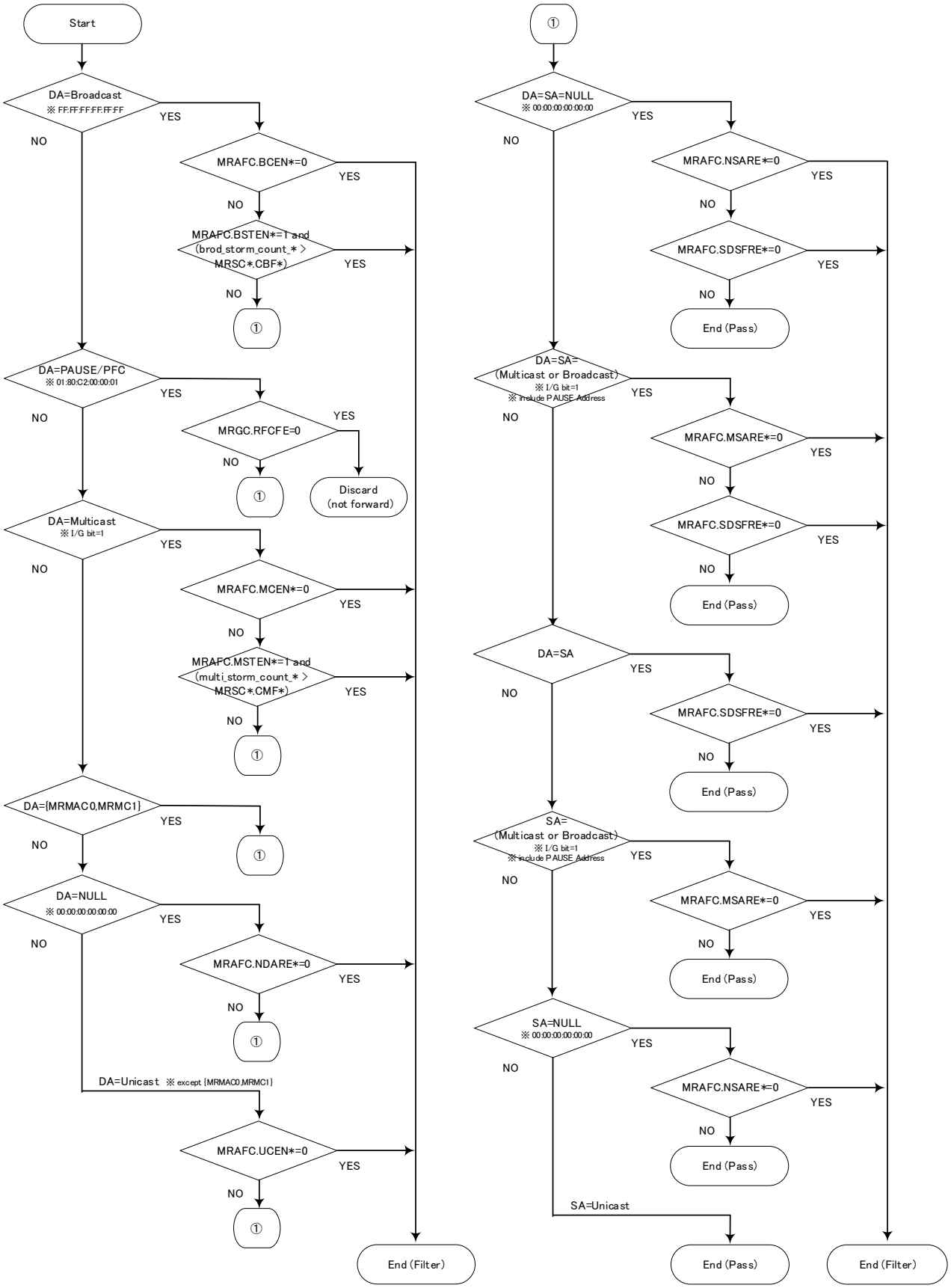
This IP supports the MAC address filtering for Unicast, Multicast, Broadcast, Broadcast storm(continuous Broadcast frame reception) and Multicast storm (continuous Multicast frame reception). It uses the destination address (DA) and source Address(SA) of a received frame for the address filtering. When the filter condition is satisfied, the received frames are transferred to an upper layer. If not satisfied, the received frames are transferred to the upper layer with the error flag of “Frame filtered” in follow.

The filter condition is configured in Reception filter configuration register (MRAFC). It can be individually configured for Unicast address, Multicast address, and Broadcast address.

And this IP also supports Promiscuous mode.
When MRAFC.UCEN*, MCEN*, BCEN*, NDARE*, SDSFRE*, NSARE* and MSARE* are set to 1, all frames are transferred to the upper layer without error flag of “Frame filtered”.
Note : Regardless of MRAFC configuration, Verify and Respond are not transferred to the upper layer.

The flow of next page shows the MAC Address filtering.

*MAC Address Filtering*




[Note1] "*" means "p" or "e"
[Note2] "brod_storm_count_*" and "multi_storm_count_*" are internal counters for storm filter

**Figure 6-19 MAC Address Filtering**

# 7.Precautions

## 7.1 Precautions

NA.

## 7.2 Restrictions (Including known problems)

NA.

Published by Renesas Electronics Corporation